

y2k

Is T_EX Y2K-compliant?

Erin Frambach
Rijksuniversiteit Groningen
email: E.H.M.Frambach@eco.rug.nl

abstract

Will T_EX and related programs continue to work properly after the year 2000? This article describes how T_EX deals with dates and how this affects its behavior after the year 2000.

keywords

millennium, Y2K

What's the problem?

Many programs that have been running perfectly well for a long time could crash or otherwise cause problems at the turn of the century. The reason is that these programs internally represent years in only two digits. This means that, e.g., the year 1999 is represented as '99' and the year 2000 is represented as '00'.

Any program that attempts to do arithmetics with these truncated values will fail. My age, e.g., could be determined by subtracting my year of birth from the current year. In 1999 that would be '99' minus '60' equals '39'. But in the year 2000 it would be '00' minus '60' equals '-60'. Depending on the kind of application that makes such errors, the consequences could be devastating.

If a program behaves decently, using 'large' numbers such as '1999' for internal representations, it is called Y2K-compliant (Y2K means 'Year 2 Thousand'). Many companies are currently testing all their mission-critical software and hardware for Y2K-compliance. Naturally T_EX is also put to the test.

The testing procedure

There are several ways to test if a program is Y2K-compliant. The best way to test software is to scan the source code carefully for any potentially dangerous statements. Secondly, practical tests can be done to verify if a program really behaves correctly when the year turns.

In the case of T_EX both tests are needed. The sources of T_EX have been publically available for about 20 years and they are exceptionally well documented. This makes it

easy to find any potentially dangerous program statements. However, the proof of the pudding is in the eating, so let's set up some tests to see how T_EX behaves before, 'during' and after the turn of the century.

Reviewing the sources

T_EX's sources are set up as so-called WEB files: ASCII files that contain both the source code and the documentation. The source code can be extracted from the WEB file by a program called Tangle. This general code can be used on any operating system. A 'change file' is provided for operating system specific details, such as a call to obtain the current date. These sources can be compiled into an executable.

In the file TEX.WEB (version 3.14159) we can find the following lines that contain source code dealing with the year value:

```
4922: @d year_code=23
      {current year of our Lord}
4985: @d year==int_par(year_code)
5049: year_code:print_esc("year");
5137: primitive("year",assign_int,
      int_base+year_code);@/
5138: @!@:year_}{\.{\year} primitive@>
10329: print_char(" "); print_int(year);
      print_char(" ");
12262: print(" TeX output "); print_int(year);
      print_char(".");
24125: print_int(year mod 100);
      print_char(".");
```

The T_EX primitive command `\year` is defined as a 4-byte signed integer with a range of `-2147483648` to `+2147483647`, which should be quite sufficient. The modulo function applied in line 24125 turns out not to be harmful. It is only used to write non-vital information about the preloaded format to the log file. When running the emT_EX 4b implementation a typical log file will show, e.g.:

```
(preloaded format=texput 97.9.3)
```

The (newer) Web2c 7.2 implementation avoids any possible confusion by supplying the full year:

```
(format=tex 1998.3.26)
```

Assuming that the operating system parses the correct year value to the compiler, any integer operation that involves the year value should work correctly.

Practical experiments

In order to prove that T_EX outputs the year value correctly we can set up a few tests. The program below uses T_EX's variable `\year` to display the current year and a little more information. This program can be run on a computer on which you change the year a number of times. This way you can determine how the program would behave in, e.g. **1990, 1998, 1999, 2000, 2001** and **3000**. In any case it should return the same year value as the operating system of the computer dictates.

```
\newcount\YtwoK
\YtwoK = 2000
\advance \YtwoK by -\year
\def\countdown
{It's \the\year: % display current year
 \ifnum \YtwoK > 0 % before 2000:
 \ifcase \YtwoK % 1999
 Time to panic!
 \or % 1998
 Relax, still so much time.
 \or % 1997
 Millennium?
 \else % before 1997
 Don't worry, be happy.
 \fi
 \else % 2000 or later:
 \ifcase -\YtwoK % 2000
 You've survived!
 \or % 2001
 Business as usual.
 \else % after 2001
 When's the next deadline?
 \fi
 \fi}
\countdown
```

Next we can run a program just before midnight **1999** to see what happens if the year turns during the run of a T_EX program. The program below will display the current time (in minutes since last midnight), day, month and year in an endless loop.

```
\loop \message{It's now
 \the\time,
 \the\day,
 \the\month,
 \the\year}
\iftrue \repeat
```

From the sources it could be determined that T_EX polls the operating system only once to find the current time and date. In a programmer's words: these are not functions but variables that are initialized at the start of the program. Therefore, the test program should write the same line over and over.

We have run these tests on the following implementations of T_EX 3.14159:

- emT_EX 386 [4b] (MS-DOS, OS/2)
- Web2c 6.1 (Unix)
- Web2c 7.0 (Unix)
- Web2c 7.2 (Windows 95/98/NT)
- Web2c 7.2 (Unix)
- Visual T_EX 6.23 (Windows 95/98/NT)
- MiK_TE_X 1.11 (Windows 95/98/NT)

The test results were identical in all cases.

Related programs

A typical T_EX implementation contains many other programs that perform specific tasks. Almost all of them do not perform any task that could induce errors because of an erroneous date value. In fact, dates are hardly used at all in these programs. Nevertheless, a few remarks about these programs are listed below.

Metafont This program supports a date primitive, much like T_EX. However, the implementation is slightly different. In MF.WEB we can read that the date value is represented as a 2-byte signed integer. This means that its value can range from **-32768** to **+32767**, which is still quite sufficient.

MetaPost This program is identical to Metafont with respect to date values.

BibT_EX This program may be involved in sorting by date. Internally BibT_EX stores dates as integers with a more than sufficient range. If a BibT_EX user writes a date as, say **'98'** instead of **'1998'** then sorting problems may occur. However, it's extremely unusual to refer to an article's year in two digits. In any case, this would be a *user's* problem, not a *program's* problem.

MakeIndex This program does not use dates at all, so it's unlikely to cause problems.

DVI drivers In general DVI drivers only use dates in output to the console or to a log file, if at all. This should be completely harmless.

Statements

Naturally the Y2K issue has been discussed at the tex-implementors discussion list.

Several T_EX implementors have made official Y2K statements about their products. On the Internet these statements are usually easy to find.

Some T_EX implementations for which I couldn't find a statement on the Internet I contacted by email. Their responses are also listed below.

tex-implementors discussion list

From: Barbara Beeton

Subject: message for prof. knuth don,

after a lengthy discussion on the tex-implementors discussion list, it seems advisable for me to write and ask you to consider another change to tex and metafont.

due to the way bureaucracies work, a number of people in the tex community have been asked to sign pieces of paper stating that tex is Y2K compliant.

one of them, chris rowley, had the temerity to point out that there is one place where tex now prints out a year in two-digit format and that perhaps changing this would be very simple and would save a lot of people the time otherwise needed to explain this phenomenon in the current climate of anxiety about such matters.

the code in question is

```
print_int(year mod 100)
```

it is used only in a message giving the date of a format file. chris [Rowley, EF] has further pointed out that in the years 2000–2009, only a single digit will be printed out, a result that may not be clearly recognizable as a year.

the most compelling technical discussion of the details has (as usual) been given by peter breitenlohner:

So much has been said to this subject, that I can't resist...

1. The only place where `print_int(year mod 100)` is used in `tex.web` (and something very similar and functionally completely equivalent in `metafont` and `metapost`) is when producing the "format_ident" string. This string is created by `initex` and written to the `.fmt` files and displayed (to the terminal and/or transcript file aka logfile). When `tex` loads a format that string is retrieved from the `.fmt` file and displayed again. The only purpose of this "format_ident" string is to somehow identify the format, not nearly in a unique fashion because all formats created on the same day will have the same `format_ident` string. Knuth has chosen to use `(year mod 100)` as part of that string – probably for human readability – but some sort of hash code would serve the same purpose and might provide a unique identification.

*No one in his clear mind – except possibly a bureaucrat – should take this use of `(year mod 100)` as an indication of lacking Y2K compliance!! It would, however, be nicer to use the full year. PT has, however, argued that this is not allowed, since only Don Knuth is allowed to make such changes. I think this point is not quite clear, although **it would certainly be best if Knuth either makes or at least approves such a change.** (emphasis added by Barbara Beeton) Otherwise such a modification must certainly live in all the system dependent change files.*

Let me quote from `tripman.tex`:

If somebody claims to have a correct implementation of T_EX, I will not believe it until I see that `\.{TRIP.TEX}` is translated properly. I propose, in fact, that a program must meet two criteria before it can justifiably be called T_EX: (1) The person who wrote it must be happy with the way it works at his or her installation; and (2) the program must produce the correct results from `\.{TRIP.TEX}`.

...

\item{5.} Compare the `\.{TRIP.LOG}` file from step 4 with the "master" `\.{TRIP.LOG}` file of step 0. (Let's hope you put that master file in a safe place so that it wouldn't be clobbered.) There should be perfect agreement between these files except in the following respects:

\itemitem{a)} The dates and possibly the file names will naturally be different.

The last sentence could be interpreted in the sense that a four digit year in the `format_ident` string is legitimate.

2. It has been said that grepping `{tex,mf,mp}.tex` for "year" is not enough. This is, in principle, certainly correct. Since I know, however, these files fairly well, let me assure you that grepping is good enough in this particular case.

3. `tex.web` initializes `\year`, `\month`, and `\day` and `\time` to the 4th July 1776, 12.00 noon. Since `tex.web` is Pascal based and `std. Pascal` does not provide such services this is certainly one way to do it. It is then left to the respective implementation (via the system dependent change file) to retrieve the correct date from the operating system and initialize `\year`, etc. accordingly. `\year` is supposed to be initialized to the current year, e.g., 1998, no truncation!!!

But: `\year` et al. are just some of T_EX's integer parameters initialized such that they reflect the

current date and time. They are otherwise completely under the user's control. Using the sequence `"\year=1940 \dump"` today yields "40.10.20" as part of `format_ident`, `"\year=2000 \dump"` yields "0.10.10", and `"\year=-143 \dump"` yields either "-43.10.20" or "57.10.20" – depending on how the modulo operation works for negative values (this is not specified in the Pascal Standard).

with best regards, Peter Breitenlohner

if a change to remove this pseudo-problem is delayed until 2002, then tex is liable to have to be subjected to a lot of unnecessary scrutiny, taking attention away from things that really *are* problems, and even potentially causing some sites to ban its use. i don't think that is desirable – even less desirable than my writing to you at this date about the problem.

i hope you agree.

Subject: note from Don Knuth

Dear Barbara,

Thanks for your note. It spurred me to make update my macro files for errata in TAOCP; now they are Y2K compliant!

Peter Breitenlohner's comments are 100% correct. Also, I might note that the Metafont documentation already mentions that Metafont (as it stands) cannot be run after the year 32767; the latter "problem" is not applicable however to TeX.

I agree that it would now be best to remove the "mod 100" from T_EX module 1328 and from MF module 1200 (and from MetaPost in the corresponding place). I hereby give permission to implementors to make such changes in their change files. No change to the version numbers are needed.

I've actually made the changes in my personal copies of `tex.web` and `mf.web` and `tex82.bug` and `mf85.bug` and `errata.eight` and `errata.tex`; but those sources won't be updated at labrea until 2002.

I'm not changing page 23 of *The T_EXbook* — where the example was probably my original motivation for cutting to two digits, since the example wouldn't fit on a single line otherwise — nor the corresponding page of *The Metafontbook*. The format-identifier details are not an essential part of T_EX's actions.

Incidentally, I'm not considering this to be the "final bug" in T_EX. But it may well turn out to be the final change ever made.

Cordially, Don

On the tex-implementors discussion list Nelson Beebe raised the issue of macro packages that may use truncated year values:

Since \year is accessible to T_EX packages, date truncation could also appear elsewhere in T_EX distributions. I therefore checked all of the files in the T_EX Live 3 tex/tex/latex tree for references to \year, and turned up just one another year truncation, in:

`tex/latex/dinbrief/dinbrief.cls`

In this L^AT_EX class file the macro `\ntoday` displays the current date as "day.month.year" in which the year is represented as two digits. Since no calculations are applied on this value, it should be harmless.

PCT_EX

Their official statement can be found on <http://www.pctex.com/techsupp/itechinf.htm> and this is what it says:

"Is PCT_EX Year 2000 Compatible?"

Because PCT_EX and accompanying macros etc. are operating system/date dependant; as long as you have taken proper precautions to ensure your operating system is year 2000 compliant, you will experience no change in operation, come January 1, 2000. If you need a verification letter for your files, click [HERE](#)"

The word *HERE* above links to <ftp://ftp.crl.com/users/rw/98765/yr2000.dvi> which reads:

"June 16, 1998

Year 2000 Manager

To Whom it May Concern,

The vendor warrants that each item of hardware, software, and/or firmware delivered, developed or modified under this contract shall be able to accurately process date data (including, but not limited to calculating, comparing and sequencing) from, into, and between the twentieth and twenty-first centuries, including leap year calculations, when used in accordance with the item documentation provided by the contractor, provided that all items (e.g. hardware, software, firmware) used in combination with other designated items properly exchange date data with it. The duration of this warranty and the remedies available to the customer for breach of this warranty shall be as

defined in, and subject to, the terms and limitation of any general warranty provision(s) of this contract, provided that notwithstanding any provision to the contrary in such warranty provision(s), or in the absence of any such warranty provision(s), the remedies available to the customer under this warranty shall include repair or replacement of any item whose non-compliance is discovered and made known to the contractor in writing within ninety (90) days after acceptance. Nothing in the warranty shall be construed to limit any rights or remedies the customer may otherwise have under this contract with respect to defects other than Year 2000 performance.

Sincerely,

Lance Carnes
President”

TrueT_EX

Their statement can be found on <http://www.truetex.com/y2k.htm> and it reads:

“T_EX Year 2000 (Y2K) Issues in Summary

This page summarizes the year 2000 issues surrounding T_EX and METAFONT, based on discussions in the news-group *comp.text.tex* and among the members of the mailing list *tex-implimentors@ams.org*. We attempt herein to merely set forth the chief matters at hand, without engaging the controversial aspects of what solutions should be taken.

1. **Crashing:** The programs T_EX and METAFONT themselves will not crash due to dates. (However, each executable implementation depends on a run-time library and an operating system, which should be evaluated in this regard.)
2. **Timestamps:** A 2-digit year is (a) printed in logfiles, and (b) stored in format file and base file time stamps. These items should not be of general concern, because they are intended for human readers and not as input to other programs.
On November 24, 1998, Donald Knuth granted an unusual permission to modify T_EX and METAFONT to use 4-digit timestamps (nearly all implementations, such as *web2c*, had already been doing so), saying:

I agree that it would now be best to remove the “mod 100” from T_EX module 1328 and from MF module 1200 (and from METAPOST in the corresponding place). I hereby give permission to implementors to make such changes in their change files. No change to the version numbers are

needed. [As reported by Barbara Beeton on the tex-implimentors e-mail list.]

- This permission means that 4-digit timestamps, while changing the output of T_EX and METAFONT slightly from the current autographs, still meet Knuth’s authoritative standards required of software calling itself T_EX or METAFONT.
3. **The \year primitive:** T_EX TRIP certification, in the strictest sense, does not require that \year return a meaningful value (T_EX may be certifiably implemented on platforms that do not even supply date reporting, such as standard Pascal). The T_EXbook does define \year as “the current year of our Lord”, which is the only correct meaning of \year for those implementations which can supply a meaningful value, which is to say nearly all of them.
In short, T_EX implementations should provide a value in \year giving the 4-digit year Anno Domini, or the value 1776 if the platform does not support a date function. T_EX does not provide any state variables to indicate whether \year contains a meaningful value, and while 1776 could have been considered a signal value for a lack of meaning to \year, this is not a standardized requirement.
 4. **External software:** The T_EX corpus embodies many accessory programs, such as macro packages and DVI translators, which may compute dates from the value of \year (or rarely, from timestamps). Such accessories should be checked individually for correct behavior when \year is assumed to return a correct 4-digit \year value before and after 2000. Accessories with an additional “defensive level” of correctness will behave reasonably when \year contains a two-digit value or a meaningless value such as 1776.”

Y&Y

Their statement can be found on <http://www.YandY.com/Y2K.htm> which reads:

“Year 2000 (Y2K)

Relative to the Year 2000, there are no date issues with Y&Y T_EX System release 2.1

Y&Y T_EX uses only standard calls to the Microsoft C++ library and WIN32 API to manage the dates and times for files. The operating system sets the date and time arithmetic values based on what the PC’s realtime clock provides. It is the operating system, the underlying BIOS, and the realtime clock upon which Y&Y software depends for Year 2000 compliance (*)

We do not embed or implement any algorithm of our own or any third party's to compute dates in any of our products.

TeX itself does not do any computation based on dates. It simply provides informational access to the dates and times obtained from the underlying operating system. The year is not truncated to two digits when offered to the user from inside TeX source code via \year, or in information output on screen or in the log file. This information is provided "AS IS" without warranty of any kind, either express or implied. Any further question may be addressed to the attention of support@YandY.com

() Some C libraries on Unix, DOS, Windows and Macintosh will have an overflow problem in the year 2038 or 2040. Unlike some other TeX systems, some browsers, and some email programs, the Y&Y TeX System release 2.1 will not crash when run on a machine with the year set to these values."*

In an email Bertold Horn, representing Y&Y, wrote to me:

"We some time ago switched to 4 digit years in the comment string in the format file. Which is the only place where there is anything even remotely related in TeX itself. We did not feel any need for wait for Knuth to make a statement on this. In fact we have made numerous such small changes to improve TeX.

[...]

IMHO, the Y2K problem is vastly overrated. The year 2038 and 2040 bugs are more serious. Unfortunately I won't be around to have to worry about that.

Regards, Berthold Horn."

Visual TeX

Through email MicroPress Inc. stated:

"No statement, besides "VT_EX is fully Y2K compliant"

A statement of Y2K compliance will appear on the Web site in the near future. (With all the new things, the site has fallen behind a bit.)"

Unfortunately the full statement is still not available.

BlueSky

BlueSky's official statement can be found on <http://www.bluesky.com/y2knote.html> and it reads:

"Year 2000 Compliance

As you may or may not know, the Macintosh in general has been Y2K compliant since it was first introduced in 1984. Contrast this with Microsoft's Windows software—even the Windows 95 version is not Y2K compliant, nor is their Windows NT 4.0! See Apple's year 2000 page on the web.

Likewise, most Macintosh software is built to handle the year 2000, and Textures is no exception. Don Knuth, the author of the TeX typesetting system and the multi-volume set "The Art of Computer Programming" was well ahead of the curve, and TeX has been Y2K compliant since the 70's!

In short: Yes, Textures is Y2K compliant.

Apple Macintosh: the choice for the year 2000 and beyond ;-)"

emTeX

In an email Eberhard Mattes wrote the following about the millennium problem:

"AFAIK, the only problem is TeX printing only two digits of the year in the .log file. (Blame DEK.) Completely harmless."

Conclusions

Studying the TeX sources shows that a Y2K-problem is very unlikely to appear. The practical tests revealed no problems either. TeX jobs that do calculations on dates should not be run around midnight because TeX only polls for the current time and date once, at the start of the job. But that is only natural for a program that's not supposed to run continuously: this limitation has no relation to Y2K-compliance.

TeX is a programming language that can be used to program just about anything, including applications that are clearly not Y2K-compliant. TeX itself does not do any calculations that involve date values. The value of the variable \year is an integer (with a range more than sufficient for any year value) that is provided by the operating system on which TeX runs. Therefore, we have every confidence that, when TeX is run under a Y2K-compliant operating system, it will itself be compliant. However, being freeware, no warranty (express or implied) accompanies its sources, and TeX users are therefore advised to perform their own examination and testing.