

Isolation Inspiration Interview

Do Not Disturb During Interview

Dan Mills

From: millsdan@kc.rr.com
Sent: Tuesday, April 14, 2020 11:46 AM
To: 'Peter de Jager'
Cc: Dan Mills
Subject: RE: My Y2K Experience

Wow, there is vast difference between John and myself - top to bottom rungs!

1. When did you realize there was a Y2K problem?

About 1997 (late) someone (not me initially) asked if we might be in trouble with the century changing. I started researching and came across some of Peter's works and writings. I've essentially followed his writing ever since. 'A Pocketful of Change' and such - well you probably know all about his writings if you are here now.

2. How were you involved after that?

I worked for two different organizations dealing with the Y2K Bug - one processing homeowners insurance and another processing annuities (one relatively minor impact and the other immensely impacted).

3. What was the major way in which you dealt with the situation?

There were essentially two methods most people dealt with the problem: 1) expand the year to 4 digits; or 2) pivot - select a date that anything on or before that date is treated as 1900 while anything after is treated as 2000. We chose 12/31/1959 I think.

4. So, simply expanding the date fields just couldn't be done?

If you think about thousands of lines of code - and automated code changes (make the same change repetitively through automation) were in their infant states then - and the reason there were so many instances (you assume there is something unique each time you deal with date). So, the idea that the "simplest" and "known unknown" was better - and doable to be tested and validated before that day made what we did the easiest choice.

5. Were there any sort of shortcut methods you used?

Many of our "sort" processes used things called "control cards" where the same sort PROGRAM was called but different "procs" controlled how the sort was executed. We had a few thousand of those control cards that I (not a coder - just a business analyst at the time) could change, with supervisory review of course worked on. That freed up a coder for more complex tasks.

6. Do you know about the 400 rule for leap year?

Every year that is evenly divisible by four is a leap year except for years that are evenly divisible by 100 and not evenly divisible by 400. For example, 2014, 2015, 2017, and 2018 are not leap years because they are not evenly divisible by four, 2012, 2016, 2020, and 2024 are leap years because they are evenly divisible by four and not evenly divisible by 100, 1800, 1900, 2100, and 2200 are not leap years because they are evenly divisible by 100 and not evenly divisible by 400, and 1600, 2000, and 2400 are leap years because they are evenly divisible by 400.

So, every 4 years except those divisible by 100, except those divisible by 400.
an interesting example of nested if "created over 2000 years ago!"

From <<https://www.quora.com/What-is-the-fact-for-dividing-a-leap-year-by-400>>

7. The interesting aspect of Y2K that you experienced dealt with leap year then?

Yes. We used FTP (File Transfer Protocol) extensively in communicating by swapping files between our clients and ourselves, like price files used to support trades between annuity funds. We also produced a fund value file at the end of processing showing the new balance of each fund our clients had post processing. Needless to say, FTP processes were fundamental to our ability to process.

8. There was a problem then that you encountered?

We used third party software to support our FTP process and there was a patch that was mis-categorized as "cosmetic" when in fact it dealt with the 400 year rule. Without it, the calendar for the software did not keep the date in sync - and that fundamentally meant, if the host system and the connected system did not match dates, the software disconnected - immediately.

9. That seems easy enough to diagnose! What happened?

Well, because the speed with which the disconnect happened, we didn't really notice that we actually did connect and that there was a handshake attempted - all we saw was we tried and "couldn't connect." We thought we had all sorts of different problems - VPN, telephone, our "slice of the telecom pie" - all sorts of problems were thought of, checked, and ruled out. Finally, on a group call, someone (me) asked "have you talked to the software vendor to see if anyone else has this problem?" While I expected, "Of course!" the response was "No." When they checked, they discovered the patch should be applied and that fixed it. Some thought I was the hero, but I thought I was the goat for not asking that question from the beginning. Similar questions though became a trademark among all those on the call when problems rose to that level. *TO BE CLEAR, I WAS THE LOWEST PERSON ON THE CALL. VPs AND SENIOR TECHNICIANS WERE ON*

10. What was the "real" lesson to be learned?

There are several: 1) you don't know as much as you think when someone gives you the "expert" moniker; 2) sometimes asking your software partners for help early is better than not asking; 3) the least experienced person in the room sometimes asks "dumb" questions that can cause you to rethink assumptions and that is sometimes a good thing; 4) crises shape character in ways you never can foresee. *- BOTH GOOD AND BAD*

11. Are there any parallels from Y2K (and the aftermath) that you can foresee?

We COBOL folks became expendable after Y2K since everything was "up to date" and maintenance was offshored to India for the most part. I recognize the "Made in America" thought processes of today will mean one thing that most Americans will have some difficulty dealing with - higher costs. Over time, the people we entrust to make decisions about "commodities" will select the least expensive product available (that's their job). Some things are designated as "trade secrets" or "military necessity" and by law are not allowed to be manufactured outside the U.S.. Before situations like the current pandemic, nobody is clairvoyant enough to understand exactly what will be critical - but there are many people who are perceptive enough to give pretty good guesses. In the world of corporate expense controls though, the perception will rarely become actionable - and that is where we will continue to be in trouble.

I'VE ALSO NOTICED THAT TRADESMEN HAVE BECOME MORE RESPECTED. AS THE SONS OF AN ELECTRICIAN MY BROTHERS AND I WERE EXPOSED TO "THE TRADES" ENOUGH TO "GET BY" ON OUR OWN. WE DIDN'T NEED TO CALL SOMEONE WHEN SOMETHING BROKE - THAT IS NOT SOMETHING MY KIDS LEARNED - THEY CALL SOMEONE

From: Peter de Jager <pdejager@technobility.com>

Sent: Tuesday, April 14, 2020 9:15 AM

To: millsdan@kc.rr.com

Subject: Re: My Y2K Experience

Hi Dan

You have more than enough to do an interview. esp the leap year story... here's what I've been send out - take a read - and a listen - and get back to me - the key thing? I'd like you to send me 10-12 questions you'd LIKE to answer... ie... you have a story for each particular question.

Thanks for expressing interest in being interviewed for my 'Y2K An Autobiography' podcast.

I'll start off by providing you several bits of information.

And, my apologies in advance because this is a long read - but then it needs to be I guess.

First: I've completed the Interview with John Koskinen and it is the best example - so far - of the type of interview I'm hoping for.

You might take a look at it to see how I'm approaching these.

John Koskinen's interview:

Link: <https://vimeo.com/381177127>

Password is: john751

***I'd ask that you NOT share this with anyone please.

Second: An overview of what I'm attempting to do with this project.

Starting about mid-2000 I stopped giving ANY Y2K Interviews with the media, as I had grown weary of both the 'it was a scam/hoax - nothing happened' and 'It was all about survivalists and hoarding food' narratives that the media had grown fond of.

As we approached 2020 I realized that the phone would start ringing again as the media did a backward look at this, now historical, event. It struck me that If someone had been born when I wrote the 'Doomsday 2000' article in 1993... they would be pushing 30 today... and to have been involved in Y2K in any real manner, they'd be almost 50 years old...(feeling old I am) ANYTHING less than that? And their view of Y2K is acquired entirely from the media narratives mentioned earlier. Which is a pity.

So... I decided to take the narrative back. I've started a podcast on Jan 1st 2020 entitled 'Y2K: An Autobiography' ---

You can access this from two sites:

iTunes: <https://podcasts.apple.com/ca/podcast/y2k-an-autobiography/id1455676429>

Podbean: <https://yy2k.podbean.com/>

And... several episodes are already posted.

The podcast is intended to be an objective as possible historical record of what this project entailed. It will run as two streams --- the first? 10-12 audio components from focused webinars - freely available to any who wish to listen.

The second stream will be an on-demand premium service providing the full webinars that provided the audio for the first stream AND a series of 'from the trenches' interviews from those who actually worked on some aspect of Y2K and know whereof they speak --- unlike MOST of the media coverage. This is where I'll place your viewpoint. Each episode is taking about 40 hours of work - and I hope to recoup at least a fraction of that personal cost via the ondemand part. at some point? My hope it to pass the entire collection off to a library system/research centre of some sort.

Frankly? While I HOPE it will gain some traction and a following, I'm mainly doing this to offset the media coverage and create an historical resource for the future.

The interviews are not going to be typical interviews. I'm not going to ask a series of preselected questions - I'm merely going to let each individual decide for themselves what they think was important - and then ask a question or two, or comment on what they've chosen to talk about. The KEY idea here, is that YOU have a story to tell, here's the chance for you to tell it.

When John Koskinen -- the US Y2K Czar was on, he spoke to these points (This is the first link I provided

you above)

Concept behind the creation of the President's Council on Year 2000 Conversion

How we decided that the best way to proceed was to create partnerships with the companies in each of the critical parts of the country's infrastructure

How we came up with the 25 working groups

The consensus of the private sector on the most important thing we could do

The concern about public panic as we got closer to the transition date

The purpose of our "community conversations" held across the country

How I got drafted into worrying about the rest of the world

The structure of our international effort

Our ongoing outreach with a wide range of individuals and organizations, including those who thought we weren't doing enough to warn people about the upcoming disaster

Ultimately, how certain were we that there was a real problem to be solved and how did the entire effort go so well?

Third: If you'd like to have your voice on record? Then I'd love to have you on.

1 - I'd need a series of points, and the sequence in which you'd address them. I'll convert those into a visual guide as a background when you're talking.

2 - we'd need to find a 90 minute slot where you could be at your desktop PC... microphone is preferred, but there's a toll-free number you can use. I don't think we'll need that long, but it's best to carve that out in case of technical difficulties.

I'll setup an online Gotowebinar session for the agreed upon time slot, and send you the link.

Lastly: My promise to you is twofold - It'll be as painless as humanly possible --- and unlike other interviews? Once I have the recording - I'll send you a link to it BEFORE I make it available, and you can decide whether you want me to use it or not. ie. Unlike any media interview, YOU control your words and thoughts. I am so tired of being misquoted and I'd loved to have been given THAT option by the media back in the day.

Let me know if you're still interested AND when you might be ready to do the interview. And we can move ahead.

Cheers

Peter

On Tue, Apr 14, 2020 at 9:13 AM <millsdan@kc.rr.com> wrote:

Peter,

I think I'll just write down a synopsis of my Y2K experience and let you decide whether there is merit in interviewing me for your Y2K endeavors. I'm thinking my story is not much different than quite a few others who were involved at the time.

In the mid-1990s I worked for a relatively small homeowners and property and casualty insurance cooperative here in Leavenworth, KS as a Business Analyst (title was something that allowed less pay!!) Eventually your name and ideas became known to myself and others and an analysis of how our "home grown" software and databases handled dates. Of course, it had been written in the 50s and 60s with an expected shelf life of about 15 years – it was well past time to be replaced, but as they say "it works, lets keep using it." I'm afraid that was / is a common thought pattern among "prudent" cost benefit analysts today. In any event, a judgement that acquiring replacement software would contain two constraints that were unsurmountable – 1) there was insufficient time to get it evaluated, purchased, installed in a test environment, a conversion process gone through (with the incumbent phases [develop, test, etc.]), the new software tested [with real converted data], users trained and finally get the new software moved to a production environment; 2) Cost was prohibitive for such a small operation. Of course, we had all the same logical reasons for being where we were – it seemed as if everyone had made the same assumptions about our code not living long enough to matter and data storage space was at a premium in the previous decades.

A decision was taken then to modify the method where date handling occurred using a "pivot date" which was also a common practice at the time (get along until a proper conversion can occur). Of course there were a number of tables where expanding the date and using the full "8 digit year" was more practical than the pivot date and for those tables and processes we simply expanded the data. I'm sure the topic of why and how the pivot date process had been adequately covered by yourself and others extensively.

Part way through that process, I was offered a position at Computer Science Corporation (CSC) in downtown Kansas City working on the VAS (Variable Annuities System) that processed annuity contracts for a number of top tier insurance companies. VAS had been developed by DST Inc and spun off in the 1980s or early 1990s, eventually being purchased by CSC and used extensively. CSC had a successor system, Vantage One, that many companies moved on to, but some didn't. Eventually the VAS system was down to only a few clients, eventually just one.

VAS had two major divisions, one for fixed annuities and one for variable annuities. I worked on the fixed side initially and in the end of my time at CSC I was the Account Manager for our client and knew quite a lot about the variable processes and team as well (the differences were relatively small).

In any event, I started as a Business Analyst and eventually worked up (in 3 years) to be Project Manager and then Account Manager for our client. I worked with some of the best COBOL programmers in Kansas City in some of the toughest situations possible. Everyone on the team knew that the platform of VAS was dying, that our client (eventually bought it from CSC) was using it as a bridge to something else, and that their jobs would soon (nekulous as to exactly when) disappear.

Anyway, the changes to VAS were also based on converting what made sense to the 8 digit date format where possible and pivoting in those situations where it would not work. As I recall we decided 12/31/99 was the date where we decided that any date before then meant 2000 while anything after meant 1900 (plus the last two digits of course).

I remember that a good majority of our sorting routines used "proc cards" where I was able to prepare changes to those procs in a test environment quickly, like perhaps 30 or 40 per day, leaving the actual code changes to the developers. I plowed through some 2000 procs in a few weeks while developers handled the "real" issues.

I was asked to put together a Y2K plan for the night of Dec 31 in detail. At first I wrote this long introduction speculating on everything that could go wrong, including telephone and electrical grid. My boss then suggested, "just worry about what we can control!" and so I wound up writing a 60 page, basically minute by minute plan showing what we expected to happen, step by step, in our batch processes. We (I was the person who asked questions and wrote down responses – I did not "design" the result) included pauses so that we could review database updates in real time and stop if we discovered problems. The plan made our batch process expand from an hour to 10 hours, followed by our weekly, monthly, quarterly and annual processes (each had their own delays).

That night (12/31/1999), our first step was a price file would be transmitted to us from the client with an expected time of 6:00 PM – it was 2 hours late – of course they wanted to make sure it was correct. We worked in an 8 story building with a glass enclosed top lounge. At midnight, a fireworks show happened in the city. We were able to see some of it, but it was "back to work" very quickly.

Afterwards there were the expected layoffs, a few here, a few more there, and then leap year happened – our FTP software had a patch that the manufacturer marked as "cosmetic" and we didn't install anything not marked essential or the equivalent. Turns out this little patch kept the software from functioning past 2/28/2000 - it would connect, check the date on the receiving end, and disconnect immediately. Unfortunately, it took us several hours to decide to check with the vendor – we thought we had a telephone problem, a VPN problem, all sorts of issues were checked and rechecked until finally someone (me) asked "have you checked with the vendor to see if they know something?" Of course the answer was no, and when they did, the patch took seconds to install. As you might imagine, delaying processing trades for several hours impacted our financial situation where we encountered stop loss situations due to the over / under results.

A few months later, while offshoring of maintenance occurred, the VAS team was disbanded and I moved to a Vantage One team – eventually that team was disbanded as well and I was out of work for a while – and then life began again!

Peter de Jager 905-792-8706 Speaker/Writer/Consultant
Books on Kindle: www.vimeo.com/technobility
So? You're a Manager... Now What?
Truth Picks
A Pocketful of Change

Dan Mills

From: millsdan@kc.rr.com
Sent: Tuesday, April 14, 2020 8:13 AM
To: pdejager@technobility.com
Cc: Dan Mills
Subject: My Y2K Experience

Peter,

I think I'll just write down a synopsis of my Y2K experience and let you decide whether there is merit in interviewing me for your Y2K endeavors. I'm thinking my story is not much different than quite a few others who were involved at the time.

In the mid-1990s I worked for a relatively small homeowners and property and casualty insurance cooperative here in Leavenworth, KS as a Business Analyst (title was something that allowed less pay!!) Eventually your name and ideas became known to myself and others and an analysis of how our "home grown" software and databases handled dates. Of course, it had been written in the 50s and 60s with an expected shelf life of about 15 years - it was well past time to be replaced, but as they say "it works, lets keep using it." I'm afraid that was / is a common thought pattern among "prudent" cost benefit analysts today. In any event, a judgement that acquiring replacement software would contain two constraints that were unsurmountable - 1) there was insufficient time to get it evaluated, purchased, installed in a test environment, a conversion process gone through (with the incumbent phases [develop, test, etc.]), the new software tested [with real converted data], users trained and finally get the new software moved to a production environment; 2) Cost was prohibitive for such a small operation. Of course, we had all the same logical reasons for being where we were - it seemed as if everyone had made the same assumptions about our code not living long enough to matter and data storage space was at a premium in the previous decades.

A decision was taken then to modify the method where date handling occurred using a "pivot date" which was also a common practice at the time (get along until a proper conversion can occur). Of course there were a number of tables where expanding the date and using the full "8 digit year" was more practical than the pivot date and for those tables and processes we simply expanded the data. I'm sure the topic of why and how the pivot date process had been adequately covered by yourself and others extensively.

Part way through that process, I was offered a position at Computer Science Corporation (CSC) in downtown Kansas City working on the VAS (Variable Annuities System) that processed annuity contracts for a number of top tier insurance companies. VAS had been developed by DST Inc and spun off in the 1980s or early 1990s, eventually being purchased by CSC and used extensively. CSC had a successor system, Vantage One, that many companies moved on to, but some didn't. Eventually the VAS system was down to only a few clients, eventually just one.

VAS had two major divisions, one for fixed annuities and one for variable annuities. I worked on the fixed side initially and in the end of my time at CSC I was the Account Manager for our client and knew quite a lot about the variable processes and team as well (the differences were relatively small).

In any event, I started as a Business Analyst and eventually worked up (in 3 years) to be Project Manager and then Account Manager for our client. I worked with some of the best COBOL programmers in Kansas City in some of the toughest situations possible. Everyone on the team knew that the platform of VAS was dying, that our client (eventually bought it from CSC) was using it as a bridge to something else, and that their jobs would soon (nebulous as to exactly when) disappear.

Anyway, the changes to VAS were also based on converting what made sense to the 8 digit date format where possible and pivoting in those situations where it would not work. As I recall we decided 12/31/59 was the date where we decided that any date before then meant 2000 while anything after meant 1900 (plus the last two digits of course).

I remember that a good majority of our sorting routines used "proc cards" where I was able to prepare changes to those procs in a test environment quickly, like perhaps 30 or 40 per day, leaving the actual code changes to the developers. I plowed through some 2000 procs in a few weeks while developers handled the "real" issues.

I was asked to put together a Y2K plan for the night of Dec 31 in detail. At first I wrote this long introduction speculating on everything that could go wrong, including telephone and electrical grid. My boss then suggested, "just worry about what we can control!" and so I wound up writing a 60 page, basically minute by minute plan showing what we expected to happen, step by step, in our batch processes. We (I was the person who asked questions and wrote down responses - I did not "design" the result) included pauses so that we could review database updates in real time and stop if we discovered problems. The plan made our batch process expand from an hour to 10 hours, followed by our weekly, monthly, quarterly and annual processes (each had their own delays).

That night (12/31/1999), our first step was a price file would be transmitted to us from the client with an expected time of 6:00 PM - it was 2 hours late - of course they wanted to make sure it was correct. We worked in an 8 story building with a glass enclosed top lounge. At midnight, a fireworks show happened in the city. We were able to see some of it, but it was "back to work" very quickly.

Afterwards there were the expected layoffs, a few here, a few more there, and then leap year happened - our FTP software had a patch that the manufacturer marked as "cosmetic" and we didn't install anything not marked essential or the equivalent. Turns out this little patch kept the software from functioning past 2/28/2000 - it would connect, check the date on the receiving end, and disconnect immediately. Unfortunately, it took us several hours to decide to check with the vendor - we thought we had a telephone problem, a VPN problem, all sorts of issues were checked and rechecked until finally someone (me) asked "have you checked with the vendor to see if they know something?" Of course the answer was no, and when they did, the patch took seconds to install. As you might imagine, delaying processing trades for several hours impacted our financial situation where we encountered stop loss situations due to the over / under results.

A few months later, while offshoring of maintenance occurred, the VAS team was disbanded and I moved to a Vantage One team - eventually that team was disbanded as well and I was out of work for a while - and then life began again!