From: Alan Bridle <abridle@NRAO.EDU>
To: rbrown@polaris.cv.nrao.edu
Subject: A y2k checklist and commentary
Date: Thu, 17 Jul 1997 12:23:28 -0400

Potential Year2000 (Millennium Bug) Exposure at the NRAO

Telescopes

Operating systems of old on-line computers (e.g. ModComps)
On-line software
Embedded chips with date/timing functions

- computers
- receivers and electronics with microprocessors
- power supplies
- any microprocessor-controlled subassemblies with timing functions - will clocks "hang" on 2000?
- computer controlled machine tools, especially if PC chipsets used and any date-awareness

Off-line Computers (hardware and os)

- all pre-1998 PCs will likely need century bit reset
- SunOs and Win3.1 must be replaced
- AIX uncertain, rev-dependent, will we still have it?
- Solaris, Mac and Win95 os are alleged okay (Win95 fully compliant only with to-be-released upgrade)
- Linux?
- testing may be difficult in presence of licensed software and automated file operations (have to roll dates back after tests) and on networks; but alternative may be real-time failures in Jan 2000

Software

- no package that does date calculations should be trusted, all packages critical to operations, purchasing, management and accounting should be tested for year arithmetic, sorting and ability to write y2k dates in output files intelligibly to other software
- any time or date-aware backup procedures should be reviewed, and any time-aware auto-deletion scripts should be disabled until proven y2k-compliant
- critical time/date functions that use old assembler calls should not be trusted, but tested
- any mission-critical databases using 2-digit years that go into calculations may need to be both backed up and overhauled (data screen problems are an obvious "nuisance" but may be less critical)
- PC software that interrogates date and time through the BIOS may still produce anomalies even when century bit is set correctly in hardware. Some commercial packages do this, and effects are BIOS-dependent. Tests are needed in actual environments that will be used post-2000 for truly mission-critical work.
- Unix systems should have no clock problems, but software running in them may still be defective, including parts of os or network

management. (One standard C library functions says year is input-1900, fine if used exactly as in standard doc, dangerous otherwise). Calls to old libraries, or to code for which there is no source available, or to assembler, are possible trouble spots.

- Much off-the-shelf software from commercial vendors is not yet compliant, upgrades or patches will need to be installed and paid for
- old in-house software is likely to suffer from same problems only up to us to fix, or replace with y2k-compliant outside packages, then retrain personnel

Networking

- anything depending on pre-1998 PC's, SunOS or AIX should be tested
- internet access may be "unusual" for a while, a few internet protocols are not fully y2k compliant (but probably will be by 20000, but effect of possible failures in worldwide telecom systems and lack of y2k readiness in many countries connecting into internet is very hard to predict. We must expect to place high reliance on our intranet and not assume internet is going to be available and unsaturated
- parts of our intranet use old Cisco routers that are not likely y2k compliant and may have to be replaced
- full testing of our net may be very difficult as many os are very unhappy about roll-forward, roll-back date testing (not designed to accommodate parts of file systems being dated in the future, or remote past) and because of the real-time license verification issues. Partial simulation in a dedicated "sacrificial subnet" may be needed.

Buildings/Physical Plant

Biggest exposure is date-aware embedded chips with PC-type clocks, "smart" systems that think they know day of week from calendar but which may go into strange states when calendar function has 00 in year. We should look into y2k issues re:

- automated thermostats, heating/ventilation systems
- elevators with date-aware controls
- "Smart" doorlocks with date awareness (AOC?)
- Telephone systems/PBX
- Faxes
- "Smart" copiers with date/time accounting functions in chips
- Any mission-critical time/date recording systems
- Safes (do we have any that are date-aware?)

Business preparation

I gather that our business division currently uses a very old rev. of J.D.Edwards software for purchasing/accounting. J.D.Edwards is one of the more y2k-aware companies but it may be imperative to upgrade our operation to their latest revs. and run it for some time to ensure full y2k compliance.

I am told that programs written in dbase III (used by personnel divn.) can be made compliant by turning on a "century awareness" feature, but existing databases and codes may need conversion to run with this feature. Will conversion of old records have impact for auditing purposes?

Even if the observatory becomes fully y2k compliant in time by upgrading and fully testing critical applications, the business division will need to be aware of the state of y2k preparation at any outside agencies or suppliers who are critical to our operation, including:

Funding agencies
Banks
Payroll contractors (ADP)
Power companies (we can expect 'unusual" frequency of power outages at
  all sites if parts of the national power grid are unstable to millennium bug,
  especially as 2000 is also year of solar maximum and biggest outage in US
  history was flare-related)
Essential materiel suppliers, esp. fuel
Telephone companies
Insurers

Very few companies or organizations are now y2k-compliant. Most should be ready to make some statement about future intentions, very few are likely to describe their plans in detail for competitive or legal reasons. But we may need to be as critical about the state of preparedness of critical suppliers (e.g. ADP) as we would be of any major telescope contractor. Or to have contingency plans ready in case they do not comply with our needs in time.

Some of our biggest problems could turn out to be in the business/fiscal side of the observatory, which has traditionally been insulated, perhaps with good reason, from the CIS knowledge in the rest of the observatory. We might however get into a situation where "all hands are needed" across some traditional organizational boundaries.

By comparison, I do not regard anything in astronomical data analysis as mission critical. We won't care if the dates on our FITS files are correct if we can't meet payroll, the phones are out, the intranet is down and we can't drive the telescopes. Assuming that we have power, communications, banks and an NSF that can still talk to us in January 2000, we need to be sure that the NRAO can ensure the safe operation of its telescopes by employees who will get paid, during what may be a very bumpy ride for some months in many sectors of commerce and high-tech society.

We are looking at a "come-as-you-are" problem with an immovable deadline that is the same everywhere in the world. A few unusual priority shifts may therefore be needed to handle it!

The ability to replace vulnerable systems or services closer to 2000 at reasonable cost should NOT be assumed, as everyone in the world will be trying to do this at once! The sellers' market for remediation will like start in early 1998 as many agencies are setting Jan 1, 1999 as time to test compliance. 1999 will likely see the available supply of remediation (hardware, software, and programming services) fall far behind worldwide demand. The NRAO is vulnerable to losing programmer services as salaries in commercial shops may skyrocket as the full scope of y2k problems is revealed.

Information readily available on the Web and in congressional testimony shows that many organizations that have started y2k work find that their problems are much more thorough-going than they realized. Fixing everything date-related in the short time available (we are now catching up on 30 years or more of noncompliance across a huge array of systems and services in less than 900 days) turns out to be impossible. Some old systems simply have to be junked. Some databases forgotten. Some services curtailed.

The key for late starters (and we will be one) will be triage by management, targeting the (say) 20% of all possible problems that affect (say) 80% of the mission-critical work, and finding ways to work-around or temporarily ignore the y2k failures in the rest. Without some inventory of possible exposure areas, this triage can't be done. We should have that inventory already, but we don't even have a mechanism for getting it right now.

Finally, the deadline is not only immovable but bears no relation to the size of the problem. If we find that the NRAO has a big y2k problem in a critical area, we may have no alternative but to commit all available resources to remediating it. The longer we wait to start, the worse this will get, both in terms of cost, lack of time for testing, and perhaps even availability of hardware, software and personnel to do the work. Right now my understanding is that no people or dollars have been budgeted explicitly for y2k activity at the NRAO. The calendar does not know this, and will not be forgiving of it.

Alan Bridle