The general problem
===================

The "Year 2000' or y2k problem is the failure to represent the century
part of the date correctly, or at all, in computer hardware, operating
systems and software packages, in date-aware embedded microcontrollers
and microprocessors, and in digital data.

It has already shown up in the commercial world in credit card readers,
hotel and rentacar companies checking credit or drivers' licenses by
computer, and in much commercial software such as E-mail, spreadsheets
and project management packages.

Computer operating systems now in use at the NRAO (some scheduled for
upgrade or replacement) contain non y2k-compliant utilities. For
example, although UNIX in principle has no clock problems until 2038
and a very robust calendar facility, IBM AIX 3.2.5 (which is running
Charlottesville's main server right now), has a dozen non-compliant
utilities associated with account management, timed shutdown, etc.

Another NRAO example is the time in the VLA on-line computer system,
in which the 16-bit signed integer system date derived from the VLA
clock is currently fixed algorithmically to a "cosmetic" 19YY format!!
(This is not the date used to control the array, but would be the one
assigned by the Modcomps to any error messages, time-stamping
printouts etc. Its operational consequences are therefore minimal,
but it's a home-grown example of the sort of problem that could be
widespread.)

47% of all PC's purchased from major suppliers in 1997 have firmware
(RTC+BIOS) that is not fully y2k-compliant and may affect some
date-aware applications even when their clocks are set properly.

Companies with "smart" office buildings are reporting failures in
tests of automated thermostats, date-aware power supplies and other
"smart" equipment containing date-aware microcontrollers or
microprocessors. According to the US Air Force, only 10% of all
date-aware microprocessors and microcontrollers sold in 1995 were
y2k-compliant. Billions are sold every year!

Neither of the NRAO examples I gave above is a show-stopper. The
Modcomp problem at the VLA doesn't directly affect array control, and
the server in C'ville is scheduled to be replaced; its os can also be
patched or upgraded even if we don't replace it. But they point to
the sorts of questions that we should be asking NOW about ALL our
critical systems.

We won't know for sure how big, or small, the NRAO's exposure is to
y2k problems until we look.

What's "critical"?
==================================

"Looking" involves taking an inventory of all our possible exposures,
starting with the most critical areas first.

We need to identify the most critical areas of activity. This should
be done by AD's NOW.

I suggest that we should begin by focussing on areas that are
essential to keeping the telescopes operating, to keeping the
essential infrastructure of the observatory functional, and to
time-deadlined business (fiscal and personnel) functions.

Evaluation
==========

Then we need to take inventory of our y2k exposures, starting with these
most critical areas.

The exposures may come in:

Computers -- real time clocks, firmware, operating systems, languages
             and compilers, applications software (in-house and
             commercial), databases, interfaces and device drivers.

Networks -- routers, switches, protocols, scripts and tools

Infrastructure -- all date-aware microprocessors and microcontrollers,
             including HVAC, office equipment, telephone systems,
             security systems

Telescope ops -- online computers, monitor and control software,
             microprocessor-controlled electronics, correlators,
             communications with other systems, essential materiel
             suppliers

Business -- payroll, personnel records, purchasing, insurance,
             in hardware, software and databases. Where outsourced,
             when will supplier be y2k compliant and what upgrades etc.
             will be needed for us to retain compatibility with them?
             Interlocking databases could be a big problem, defining
             when to cut over from old to new versions.

We need to evaluate the NRAO exposures in every critical date-aware
area. This can be done through enquiry to suppliers or designers, by
reading code, etc. and by testing.

Evaluation should eventually be by testing wherever possible. Tests
may not be trivial, they will require design (both so that they are
extensive enough to be valid and so that we can be reasonably sure
that we can back out of them) and they will take time away from normal
operations.

Systems and codes that are "y2k ready" in principle may never have had
their y2k logic exercised in our operational environments, and so do
have to be tested.

For complex or inaccessible systems evaluation may only be possible
through testing. y2k problems in embedded processors cannot be
simulated.

Testing will take planning, and take time and resources away from
normal operations. We have to make this investment.

Testing can also be dangerous as well as time-consuming. Tests have
to be designed so you can back out of them, or done on exact replicas
of the systems.

We can't afford to wait to start this. The deadline is immovable and
bears no relation to size of the problem. Everyone is faced with the
same deadline and supplies of remediation may become scarce or
expensive, perhaps including personnel!

For vendor-supplied software, we will have to develop some basis for
making our own assessments, just as we do of antenna contractors.

There is a problem in evaluation of vendor statements in that there is
no standard for "y2k compliance". Vendors may say something is "y2k
compliant" if it is possible to make the product comply, but not
necessarily compatibly with the compliance conditions of other products.

Priorities for Remediation
===========================

If it turns out that we have a large y2k problem anywhere, it may be
very important to distinguish things that are functionally
non-compliant but only have nuisance or cosmetic value. A
non-compliant date appearing in an obvious output display is not as
dangerous as one that is used for date arithmetic, for sorting or for
scheduling of real-time events, for example.

Off-line astronomical data processing is likely to be in good shape
once FITS is fixed, but is also less critical to the observatory's
operations.

Action Needed
=============

Every division at every site should NOW appoint a y2k responsible
person, preferably someone with computer and/or engineering expertise,
to act as the lead person for this assessment. An observatory-wide
steering committee should assist these efforts,

We should define critical operations areas NOW and start y2k asseement
in them immediately, to the point that we can budget $$, time and
people for testing and remediation. If we have a large y2k problem in
any critical area, it will need much more careful management than
usual, as the deadline is immovable.

We should extend this effort to less critical operations areas SOON.

We should decide what can be tested, and design and schedule tests.

We should defer working on y2k things that are a mere nuisance until
we have fixed potential show-stoppers, if we have any.

Then we should decide whether to retire, replace, or refurbish the
critical non-compliant items.

We should also prepare to develop contingency plans for our own y2k
problems, and those of essential outside suppliers, in case everything
is not done in time. We need to be aware of y2k compliance at all
entities with which we exchange vital data, or on whom we depend for
vital supplies and services.

We are already late starting y2k assessment compared with many

organisations, including the much-maligned federal government! In our
favor we may have an unusually skilled workforce and a limited
incidence of the problem in our home-grown codes and systems because
we are used to being careful about times and dates. But this didn't
mean that FITS was y2k-compliant, and we should take nothing for
granted!

Most organisations that have started y2k assessments discover they
have a much bigger problem than they thought, with unexpected budget
and personnel implications.

This can only get worse as the deadline approaches.

We need to start NOW.

It's not just software
Darrell: stuff we've bought isn't our problem
        our stuff is okay.

Tony:
There are people going through our ~~fits~~ software

Drick Svanek: GPS Rx is Aug 22 1999
            12 × $10,000

Offline is not critical but okay!

JDesmond
We have assurances for JDE & ADP

Personnel Cent is looking into it.
        CIGNA / IAA

Tucson: Jeff Hagen
Socorro: Beesley
GB   : Bob Vance
Bus. : J. Desmond
CV   : Simon,