United States
Environmental Protection
Agency

Office of Information
Resources Management

EPA Year 2000 Guidance Document

Contract No. 68-W1-0055
Delivery Order No. 094
Product Control No. SDC-0055-094-DM-6006B

February 14, 1997

February 14, 1997

YEAR 2000 GUIDANCE DOCUMENT

CONTRACT NO. 68-W1-0055
DELIVERY ORDER NO. 094

Prepared for:

United States Environmental Protection Agency
Office of Information Resources Management
401 M Street, SW.
Washington, DC 20460

Alternate Delivery Order Project Officer:

Dwayne Aydlett

Prepared by:

EPA Systems Development Center
(A Contractor Operated Facility)
Science Applications International Corporation
200 North Glebe Road, Suite 300
Arlington, VA 22203

PREFACE

"All systems in the planning, analysis, design, and development stages and that use a date need to provide for the four-digit year. All existing systems that are in the maintenance stage and that must deal with years beyond 1999 need to plan and implement the year change before they can handle those dates. Each office should ensure that the impact of the date change is assessed for planning and budgeting purposes and the appropriate system and data base changes are made by the time they are needed."

Alvin Pesachowitz, EPA Chief Information Officer,
May 22, 1995, Memorandum to SIRMOs, System
Managers, and Regional IRM Chiefs.

EXHIBITS

CONTENTS

SECTION 1.0 - EXECUTIVE SUMMARY

The *Year 2000 Guidance Document* provides an overview of the Year 2000 problem; addresses the issues; reviews standards for Year 2000-compliant systems, including the Environmental Protection Agency's *EPA Date Standard*; discusses the Agency's approach to the problem; and answers questions system managers might ask. The document provides guidance and strategies for solving the Year 2000 date problem in EPA systems at headquarters and in the regions, laboratories, and states.

Guidance for EPA System Managers

The guidance is specifically designed for EPA system managers, as it is their responsibility to fix the Year 2000 problem—also referred to as the "two-digit year problem"— in systems under their supervision. Fixing the problem means locating all occurrences of two-digit year fields and expanding or replacing them with four-digit year fields, or using other acceptable solutions. Finding all of these two-digit year fields within computer code is no small feat, because dates have been coded in numerous permutations over many years.

Year 2000 Issues

At EPA the Year 2000 problem means that many Agency computer systems will generate incorrect data because they cannot accurately process information containing a date beyond 1999. For many EPA systems, the deadline looms closer than January 1, 2000 because of an early event horizon—the date a system will first experience a Year 2000 problem. Systems projecting dates, such as those computing interest for enforcement actions or accounting for future fiscal years (FY), starting with FY 2000 (October 1, 1999), will reach their event horizons long before the calendar reads 2000. Therefore, the effort to make critical EPA information systems Year 2000-compliant must proceed immediately.

These are the Year 2000 issues EPA management faces to ensure the continuity of critical Agency functions:

- All EPA systems must be evaluated to determine the impact of the two-digit year problem on processing dates in the 1900s and 2000s.

- System triage must be performed to prioritize systems for repair:

  - Critical systems (currently designated as Level 1 and Level 2 systems) containing date-related problems must be on schedule for repair as a first priority.

All other systems designated for repair must also be on schedule to meet the Year 2000 deadline.

EPA's Approach

EPA's approach for identifying and solving the Year 2000 problem is presented in six stages: planning, inventory, assessment, repair, testing, and implementation. The first stage uses a project plan approach, identifying the who, what, when, where, and how of meeting Year 2000 project goals and discussing risks and dependencies, unique features, obtaining help, and selecting tools. The remaining five stages use standard system lifecycle procedures to identify the scope of the problem; assess the impact on the system and the organization that depends on the system; identify a cost-effective solution; and repair, test, and implement corrected applications and data.

Year 2000 Key Awareness Factors for EPA Program Management

Several important factors make Year 2000 repair different from most maintenance projects; it is critical that EPA program management be aware of these key factors and provide support for system managers when required:

- Recognize the following unique aspects of the Year 2000 maintenance effort:
  - Managing to a fixed deadline.
  - Establishing priorities for repairing interfacing systems.
  - Leveraging outreach efforts with external data partners.
  - Understanding the magnitude of the problem, which extends beyond software to microcode, operating systems, compilers, database management systems (DBMS), and data.
- Promote communication among managers of interfacing systems.
- Ensure that contingency planning is part of the overall strategy.
- Be prepared to make program-level decisions on technical approach, system priority, and resource allocation.

SECTION 2.0 - INTRODUCTION

Will your system work on January 1, 2000? The rollover from the year 1999 to the year 2000 will be disastrous for many automated information systems. Systems will fail or generate incorrect data because they cannot accurately process information containing dates that go beyond 1999.

> "At 12:01 a.m., Jan. 1, year 2000—or the 'Y2K,' as computer aficionados refer to it—the world won't exactly 'stand still.' But it could come to a close unless the world's major governments and businesses start to fix their computer systems right away."[1]

2.1 BACKGROUND ON THE YEAR 2000 PROBLEM

What Is the Problem? The problem is that most software has been programmed to store the year as a two-digit field. The year 1996 is stored as "96"; the year 2000 would be stored as "00." Two-digit years, used as the basis of date operations such as calculations, comparisons, and sequencing, will result in incorrect values when the years cross a 100-year boundary, i.e., 1996 and 2000.

What Is Wrong With Using Two-Digit Years?

Consider a simple age calculation for someone born in 1960 completed in a system using two-digit years. The computer application would use the current year, 97, and subtract the birth year, 60. The age would be 37 because the software logic "assumes" the year will always begin with "19." If this same calculation is completed in 2000, the age would be computed as -60.[2] The basic assumption that the first two digits of the year are always "19" also causes major problems when sorting or ordering data based on two-digit year fields. Records with dates in the 2000s will show up in the wrong sequence because they will only sort by the last two digits, e.g., 2000, 2001, 1998, 1999, etc.

Where Did This Problem Come From?

Why was software programmed in a way that could cause such problems? The answer is: to save valuable resources. Obvious, repeatable values, such as the first two digits of the year, were never stored because processing and storing data was very expensive in the early days of computing. In addition, automated data processing in the 1950s and 1960s was based on the use of "punch cards" limited to 80-digit columns. When developers coded

these early programs, they never envisioned that application software would remain in use through the end of the 1900s.

Where Can This Problem Be Found?

The Year 2000 problem is pervasive; the problem is not limited to legacy systems. As stated by the Information Technology Association of America (ITAA), two-digit year fields can be found in "microcode, operating systems, software compilers, applications, queries, procedures, screens, data bases, and data." The problem may even occur in applications on the Internet that access older databases.

2.2 DOCUMENT PURPOSE

The purpose of the *Year 2000 Guidance Document* is to provide system managers with guidance and strategies for solving the Year 2000 problem. The guidance provided in this document is intended to assist system managers in answering the following questions:

- Where do I start? Where do I get help?
- Which of my systems has a Year 2000 problem?
- How big is my problem? Can I fix all my affected systems? Which systems do I fix first?
- How do I go about fixing my systems?
- How do I determine when a system is fixed?
- How will I cross over to my repaired systems?

The guidance addresses Year 2000 issues and provides an overview of system activities, such as repair, testing, and implementation. The document presents a project approach for identifying Year 2000 problems and repairing systems and data; this approach is similar to standard system maintenance projects, but highlights issues unique to the Year 2000 repair effort. The information provided in this document does not replace EPA's system lifecycle management approach but rather assumes that EPA's lifecycle approach will be used as the foundation for Year 2000 repair activities.

2.3 DOCUMENT SCOPE

Specifically designed for EPA system managers, *the Year 2000 Guidance Document assumes that system managers are responsible for getting the job done—that is, they have to repair the Year 2000 problem in their systems.* The document focuses on the Year 2000 project at the EPA system manager level.

The Year 2000 approach presented in the document is based on methodologies in Year 2000 literature, much of which is accessible on the Internet. This document synthesizes the best information currently available on approaches to the Year 2000 problem and presents the logical Year 2000 project stages of most use to the EPA system manager.

2.4 DOCUMENT ORGANIZATION

The *Year 2000 Guidance Document* contains nine sections, a bibliography, and Appendices A through G, containing supplementary information. The document is structured as follows:

Section 1.0 - Executive Summary. Executive level summary of the *Year 2000 Guidance Document.*

Section 2.0 - Introduction. Background on the Year 2000 problem, and purpose, scope, and organization of the *Year 2000 Guidance Document.*

Section 3.0 - EPA's Approach. Overview of EPA's approach to the Year 2000 problem including system evaluation and compliance, the *EPA Date Standard*, special management issues, and introduction to the six Year 2000 repair stages to follow.

Section 4.0 - Year 2000 Planning. Strategies for planning the Year 2000 project.

Section 5.0 - Inventory. Guidance on completing and documenting the system inventory.

Section 6.0 - Assessment. Guidance on identifying date problems within the system inventory and completing the system triage.

Section 7.0 - Year 2000 Repair. Guidance on identifying cost-effective solutions and repairing affected applications and data.

Section 8.0 - Test Year 2000 Changes. Guidance for completing system tests.

Section 9.0 - Implementation. Discussion of Year 2000 implementation and post-implementation activities.

Bibliography. Literature researched in preparing the *Year 2000 Guidance Document*.

Appendix A Level 1 and Level 2 Systems.

Appendix B *EPA Date Standard* (pending).

Appendix C Additional Information on Year 2000 Software Tools.

Appendix D Year 2000 Process Flowchart.

Appendix E Where to Get Help.

Appendix F Acronym List.

Appendix G Glossary.

Endnotes

1. Lanny J. Davis, "Countdown to a Meltdown," *The Washington Post*, September 15, 1996.

2. Actually, the value is more likely to be a positive 60, as many of these types of date calculations use an unsigned field for the calculation; obviously, a person's age could not be negative.

SECTION 3.0 - EPA'S APPROACH

EPA's approach to the Year 2000 effort centers on two facts: 1) all EPA systems must be evaluated to determine the impact of using two-digit years on system processing and 2) date-related problems discovered in critical systems must be repaired within a very short time frame. Solving the Year 2000 problem within this tight schedule means identifying the full scope of the problem and using a standard, cost-effective solution to repair all affected software code and data.

> "While the magnitude of the problem in the Federal government is difficult to assess, it is clear that some systems will fail unless they are fixed. . . . Addressing the problem will require a concerted effort by our managers and computer professionals over the next three years."
> *—John A. Koskinen, OMB Year 2000 Memorandum, April 11, 1996.*

3.1 EPA'S APPROACH TO THE YEAR 2000 PROBLEM

EPA's approach to solving the Year 2000 problem is to identify the scope of the problem; assess the impact on the system and the organization that depends on the system; establish priorities for system repair; identify a cost-effective solution(s); and repair, test, and implement corrected applications and data.

EPA's Year 2000 approach is divided into the following six stages:[1] planning, inventory, assessment, repair, testing, and implementation. EPA's approach is based on the premise that 1) all systems must be evaluated to determine if there is a Year 2000 problem and 2) the event horizon—the date a system will experience a Year 2000 problem—must be identified for every critical system.

All Systems Must Be Evaluated

All systems, including applications software and operating system components containing "date" coding, must be evaluated to identify date references and determine if the references are based on two-digit years. Any data that is output, calculated, stored, etc., based on two-digit year comparisons, will be incorrect when dates cross the year 2000 boundary. In addition, even as systems are repaired to correctly process dates, if these systems receive data from other sources, there is no guarantee that the incoming data will have dates in the same date format.

Event Horizons Must Be Identified

While many systems will not experience an event horizon until the year 2000, many others are likely to have an event horizon well before the year 2000. Projected dates, such as those for insurance purposes, mortgages, and expiration dates, are common future dates that may result in date processing errors. Some EPA systems, such as planning and budgeting systems that project dates, are already experiencing Year 2000 problems. Other EPA systems likely to reach an event horizon before the year 2000 are those that compute interest for enforcement actions. Many EPA systems will reach an event horizon when they must process data containing dates for fiscal year 2000.

3.2 YEAR 2000 COMPLIANCE

Year 2000-compliant information technology is defined in the Federal Acquisition Circular 90-45 (December 1996) as follows:

> *Information technology that accurately processes date/time data (including, but not limited to, calculating, comparing, and sequencing) from, into, and between the twentieth and twenty-first centuries and the years 1999 and 2000 and leap-year calculations. Furthermore, year 2000 compliant information technology, when used in combination with other information technology, shall accurately process date/time data if the other information technology properly exchanges date/time data with it.*

All critical Agency information systems must be made Year 2000-compliant. EPA critical systems, categorized as Level 1 and Level 2 systems (Appendix A), must be given priority in the Year 2000 repair process to ensure that critical processing functions continue. However, this does not mean that Level 3 and Level 4 systems should be ignored—if a system's functionality is needed, the system must be repaired.

EPA systems must accurately process dates as follows:

- All date-related calculations must execute correctly.
- All system-specific event horizons (e.g., FY 1998, FY 1999, FY 2000, etc.) must be processed correctly, not just the rollover to the year 2000.

- Successful transition to Year 2000 processing must occur without human intervention.
- Calculations that determine and process leap years must be accurate.
- Correct results in forward and backward date data calculations that span century boundaries, including the year 2000, must be provided, including the conversion of previous years' data currently stored as two digits, if necessary.
- Date data output for data exchange with other systems must be in the format specified by the *EPA Date Standard* (see below).

EPA Standard for Calendar Dates

To address the Year 2000 problem within the Agency, EPA has issued a date standard (pending), *EPA Data Standard for Representation of Calendar Date* (Appendix B) that sets forth the required date format and prescribes the format for ***date data that are output for data interchange.*** This EPA calendar date standard (referred to in this document as *EPA Date Standard*) adopts the American National Standards Institute (ANSI) date standard that was issued as a Federal Information Processing Standard (FIPS) by the National Institute of Standards and Technology (NIST).

(1) "CCYY" represents the four-digit year.
(2) "MM" represents the calendar month of the year.
(3) "DD" represents the calendar day of the month.

3.3 MANAGEMENT ISSUES

The Year 2000 problem may be one of the most complex and time-consuming system repair efforts ever attempted. System maintenance to repair identified problems is nothing new. But the Year 2000 project faces many problems different from most maintenance projects. For this effort, it is not simply the large number of potential problems that must be addressed. The difficulty is that so many of these problems are outside the control of the system staff. System managers must maintain awareness of the Year 2000 issues and be proactive in raising these issues to management when required. These issues include the project deadline, interfacing systems, external data partners, and Year 2000 software tools.

Project Deadline. A unique aspect of the Year 2000 project deadline is that it cannot be extended even if the project schedule slips. Managing to a fixed deadline will be critical in getting systems repaired before they hit an event horizon. A key factor that can affect the schedule is the repair of interfacing systems.

Interfacing Systems. *What happens if your system has been expanded to accept four-digit years, but the system that provides you with information is still in the old two-digit year format?* Coordinating the schedule for repairing interfacing systems is important to ensure that systems are ready to send and receive correctly formatted date data at the same time.

External Data Partners.** With the move toward electronic reporting, many EPA systems now accept electronic filings from the regulated community and state and local governments. Regional offices have already begun outreach efforts to coordinate with states and local offices. When EPA cannot change the date format for incoming data, system managers may need to develop mechanisms, i.e., bridges, for converting data from the old format to the new format before the data are input to the system. ***See definition in the Glossary.*

Year 2000 Software Tools. If the decision is made to use software tools for some aspects of the Year 2000 process, be sure to thoroughly research tools and vendors before purchasing any tools. Section 4.6 discusses factors to consider when selecting Year 2000 tools. In addition, Appendix C includes excerpts from an EPA document on Year 2000 tools. This appendix discusses selecting tools and vendors and lists tools supporting various stages of the Year 2000 repair effort.

3.4 THE SIX STAGES TO YEAR 2000 REPAIR

The remaining sections in this document are devoted to EPA's six-stage approach for addressing the Year 2000 problem. Each section attempts to answer questions system managers might ask, as shown in Exhibit 3-1 below. Appendix D, Year 2000 Process Flowchart, shows the flow of activities associated with each stage in the Year 2000 system repair process.

Exhibit 3-1. Six Stages to Year 2000 Repair		
Question	**Answer**	**Section / *Stage***
• **Where Do I Start?** • **Where Do I Get Help?**	**Start by Planning the Year 2000 Project.** Identify tasks and resource requirements, and devise a schedule for the Year 2000 repair effort. Find sources of help on the Year 2000 problem here and in Appendix E, Where to Get Help.	**Section 4.0** *Year 2000 Planning*
• **Which of My Systems Has a Year 2000 Problem?**	**Complete a System Inventory.** Identify system components that could be affected by the Year 2000 problem. The inventory will be used as the foundation for the Year 2000 assessment—the components in the inventory will be assessed to identify the degree to which systems are affected by the Year 2000 problem.	**Section 5.0** *Inventory*
• **How Big is My Problem?** • **Can I Fix All My Affected Systems?** • **Which Systems Do I Fix First?**	**Conduct a Year 2000 Assessment.** Identify Year 2000 problems in the system inventory and estimate the cost to correct them. This stage includes a triage step: a systematic review of affected systems and their importance to determine which systems will be repaired and in what order.	**Section 6.0** *Assessment*
• **How Do I Go About Fixing My Systems?**	**Repair Two-Digit Years and Solve Related Date Problems.** Identify cost-effective solutions to the Year 2000 problem (and related date problems) and repair affected applications and data.	**Section 7.0** *Year 2000 Repair*
• **How Do I Determine When a System Is Fixed?**	**Test the Repaired Applications and Data.** Verify and validate the modified and repaired applications and data.	**Section 8.0** *Test Year 2000 Changes*
• **How Will I Cross Over to My Repaired Systems?**	**Implement the Tested Applications and Data.** Begin successfully operating the tested applications and data, within the implementation deadlines.	**Section 9.0** *Implementation*

Endnotes

1. Awareness is also a stage in the Year 2000 process. Because EPA started its Year 2000 awareness campaign in early 1996, the awareness stage is not covered in detail in this document.

SECTION 4.0 - YEAR 2000 PLANNING

The Year 2000 literature states that project planning and management are the key factors to successfully completing the Year 2000 project. A comprehensive Year 2000 project management plan is absolutely necessary to keep EPA mission-critical systems functional through the year 2000.

> The key is to start now. "The problems represented by the date change will only grow as the Year 2000 approaches. Organizations that decide to wait for a silver bullet solution are assuming an enormous risk. In fact, normal bureaucratic delay can be costly."[1]

What Is Year 2000 Planning?

Year 2000 planning is project planning—deciding *what* must be done, *how* and *when* it will be done, *who* will do it, and *how much* it will cost.

What Is the Goal?

The goal for the Year 2000 planning process is to ensure that work is carried out successfully to identify and repair all Year 2000 problems by the Year 2000 deadline.

Why Is Year 2000 Planning Important?

The resources needed to repair date problems—time, staff, and funding—will be an issue for many program and regional offices. Time is one key resource that is finite. Programming staff skilled in the software languages in which your system is coded may be difficult to find. Obtaining additional funding will become more difficult as more and more offices begin repairing affected systems. Good project planning and management will be key to completing the Year 2000 project on schedule and within budget.

This section is not a review of the project planning process. The intent is to review the key Year 2000 planning activities and examine the aspects of the Year 2000 project that are different from many typical system development and maintenance projects.

4.1 PREPARING THE YEAR 2000 PROJECT PLAN

In preparing the Year 2000 project plan, follow EPA's standard project planning procedures for accomplishing the five subsequent stages of the Year 2000 project: inventory, assessment, repair, testing, and implementation. Focus on the following components:

- *What* must be accomplished. Define the objectives, tasks, and milestones for each of the Year 2000 project stages.

- *How* the project will be completed. Develop an initial strategy for accomplishing each of the five subsequent stages of the Year 2000 project.

- *When* each activity will be completed. Develop the project schedule.

- *Who* will complete the work. Identify staff needed to complete the project.

- *How much* it will cost. Develop the initial budget for the Year 2000 project.

- *What is the planning network?* Using a project management software tool, create a critical path network based on tasks, durations, and dependencies. Any slippage in task schedules along the critical path will affect the project's end date.

In addition, identify project risks and develop a risk mitigation approach. Identify methods for assessing progress, and project assumptions and constraints.

Several key elements of a project plan are especially important for the Year 2000 project. These key elements include the following:

Key Year 2000 Project Planning Elements

- **Project Risk**. Because the deadline for repairing the system will not change, project risk becomes an important factor. The risk that the project will not be completed on time or that the system simply cannot be repaired (due to missing source code, unique programming languages, etc.) must be assessed and contingencies developed, especially if the system is classified as critical.

- **Dependencies and Critical Path**. The critical path and all project dependencies must be identified. Interconnected systems represent a key Year 2000 project management challenge—which system(s) must be completed first and will all interfacing systems choose compatible solutions to the Year 2000 problem?

- **Disaster Recovery Plans**. If disaster recovery plans are already in place, update them to address Year 2000 issues.

- **Contingency Plans**. The Year 2000 problem and its subsequent repair will introduce many new system vulnerabilities. These vulnerabilities include the potential for applications to stop functioning or produce incorrect results, data integrity to be compromised, and system hardware to fail. Don't assume that the plans need to cover only the time period until the system is repaired. Contingency plans must be in place to address the following contingencies:

  - Year 2000 repairs not being completed before the system's event horizon.

  - New errors or problems introduced into applications during the repair process.

  - Potential for system failure after the year 2000 due to occurrences of the Year 2000 problem that were not identified and repaired.

  Appendix E cites references providing information on preparing contingency plans.

To Be Effective, the Plan Cannot Be Static

"Planning is an iterative process that is only complete when the project is complete."[2] As with any project, the Year 2000 project plan must be continually reviewed throughout the Year 2000 effort. Because the project schedule is so critical, measuring progress against the plan, anticipating problems, and identifying solutions are essential project management functions. In addition, as the Year 2000 repair effort progresses, the initial strategy for each Year 2000 stage must be tailored, based on the scope of the Year 2000 problem within the system inventory, solutions chosen, and project experience.

4.2 UNIQUE FEATURES OF THE YEAR 2000 PROJECT

Many components of the Year 2000 project will be much the same as other system development and maintenance projects. However, features unique to the Year 2000 project should be carefully considered as the Year 2000 project plan takes shape. These factors include the following:

- The size and scope of the project may be enormous.

- The deadline will not change from January 1, 2000. For many systems, the event horizon will be well before the year 2000.
- The main benefit of the Year 2000 effort will be a system that "still works." Some Year 2000 literature refers to the Year 2000 project as one with no benefits to outweigh the cost.
- Few, if any, projects have ever required such an in-depth understanding of systems and their code. *All* system components and code must be evaluated to determine if they are affected by the Year 2000 problem.
- As the deadline approaches, competition for resources, such as computer space, computer time, and programmers, will increase. Everyone will be trying to repair systems at the same time.
- Many factors can affect the Year 2000 project schedule and/or cost. These factors include, but are not limited to, the following:
  - How and when interconnected systems will be repaired.
  - Whether external sources of data will be willing to reformat their date data.
  - Whether the commercial software the system requires is Year 2000-compliant, and, if not, when it will be repaired.
  - The extent to which system hardware is affected.

4.3 ADDRESSING THE UNIQUE FEATURES

Focus attention early in the planning process on the unique features of the Year 2000 project to ensure that the project is successful. Consider all potential problems and possible solutions, including size, scope, and deadline; resources; dependencies on other repair efforts; commercial software; and external data providers, discussed below.

- **Size, Scope, and Deadline**

Don't overlook existing project experience. Use seasoned project managers and a team approach. Take advantage of the lessons being learned from other Year 2000 projects, both inside and outside the Agency. Year 2000 Workgroups and Internet web sites provide a

wealth of information on Year 2000 issues, possible solutions, and software tools.

Look for ways to streamline the Year 2000 project, including the potential for grouping systems or subsystems for repair, and sharing system resources, such as programmers, code (common date subroutines), etc. Automated software tools will be helpful in many aspects of the Year 2000 project. Research the software tools available for assisting in the inventory, assessment, repair, and/or testing processes. The *right* automated software tool in the *right environment* can streamline the Year 2000 process. Subsection 4.6 discusses Year 2000 software tools.

- **Resources** — Line up the needed resources, such as staff and computer resources, now. Plan for how the project can continue should the level of resources change. Consider the required testing environment. Given the deadline for EPA Level 1 and 2 systems, many applications will be in the testing stage at the same time. Identify possible alternatives and ensure that the system contingency plan has been formally documented.

- **Dependencies on Other Repair Efforts** — Are there other Year 2000 projects that must be completed before this system can be repaired? Continually assess the status of interconnected systems and their Year 2000 projects. More importantly, evaluate the approach(es) they are using to repair their date problems and determine if their approach will have an impact on yours.

- **Commercial Software** — Is the system dependent on commercial software (operating systems, software utilities, compilers, etc.)? Decide how software vendors will be contacted to determine if the software they provided is Year 2000-compliant, or, if the software is not compliant, when it will be updated.

- **External Data Partners** — Does the system receive data from trading partners, state or local governments, etc.? Can the format be changed? If so, establish an outreach effort. Develop procedures for contacting external data providers to communicate changes to date data formats. Does the system share data (current or historical data) with other EPA

systems? Define procedures for repairing data shared by multiple groups or organizations.

4.4 USE A TEAM APPROACH

A team approach is important in addressing the full scope of the Year 2000 problem. Given the time constraints for fixing the Year 2000 problem, sharing both resources and knowledge will be key to meeting deadlines. Utilize a team approach that includes the following components:

- One lead team within each program or regional office to focus on coordinating Year 2000 conversion activities.
- Multiple project teams tasked with assessing and repairing individual systems or groups of related systems.

Lead Year 2000 Team

The lead Year 2000 team should communicate status and issues to upper management. The team should ensure that compatible, cost-effective solutions are used and the necessary budget, personnel, and technical resources are available. The lead team should include representatives from management, programs, and information technology staff.

Year 2000 Project Team

The Year 2000 project team will be responsible for defining the approach to be followed in repairing systems affected by the use of two-digit years. The Year 2000 project team should include the project manager, system managers, and programmers.

Do not consider these project teams to be static. Other team members may be assigned as the Year 2000 repair proceeds. Different categories of skilled personnel, such as system experts, testers, and documentation specialists may be needed in different stages of the Year 2000 effort.

4.5 GETTING HELP

An important factor to successfully solving the Year 2000 problem will be timely access to help. After all, if everyone has to learn the same technical and managerial lessons, progress for all will take longer. Appendix E (Where to Get Help) lists Year 2000 Internet web sites and key documents providing information on Year 2000 issues, solutions, and software tools.

Assistance will also be available from the Agency and other federal Year 2000 resources. The subsections below review organizations providing guidance to assist in the Year 2000 effort.

Agency Resources

EPA has established a Year 2000 Workgroup to assist offices and regions in addressing their Year 2000 problems. The EPA Year 2000 Workgroup is responsible for promulgating guidance and standards.

The Enterprise Technology Services Division (ETSD) is also a resource for Year 2000 assistance. ETSD is a key source of technical information on the operating systems, utilities, DBMSs, networks, and other technologies it supports.

Federal Year 2000 Resources

The Chief Information Officer's (CIO) Council Subcommittee on Year 2000 (formerly known as the Interagency Year 2000 Workgroup) is the primary source of Year 2000 information specific to federal agencies. The General Services Administration (GSA) offers an Information Technology Policy Clearinghouse and a Year 2000 Information Directory. These web sites are at the following Internet addresses:

- http://www.itpolicy.gsa.gov
- http://www.itpolicy.gsa.gov/mks/yr2000/y201toc1.htm

Appendix E lists other groups providing information and assistance on the Year 2000 problem.

Non-Federal Resources

Several commercial and professional organizations are providing information on the Year 2000 problem. One is the Information Technology Association of America (ITAA). ITAA offers general Year 2000 information, Year 2000 vendor and tool information, and links to other relevant Year 2000 Internet sites. The Internet address for ITAA is as follows:

- http://www.itaa.org

Other non-federal sources of Year 2000 information include computer industry journals and trade magazines. In addition, the Internet provides a wealth of information that is continuously growing or being updated. Internet sites providing Year 2000

information include vendor-maintained sites that may provide useful information on software tools for the Year 2000 process.

4.6 SELECTING TOOLS

Automated software tools are available to support many aspects of the Year 2000 process, including project management, system inventory, assessment, repair, and testing. Software tools can streamline the process by automating many portions of the Year 2000 effort. However, as stated by ITAA, "no tool is foolproof or a complete Year 2000 conversion solution."

Types of Tools

Various types of tools are available to support Year 2000 activities. Some tools determine total lines of code (LOC), LOC that are affected by date problems, application code complexity, and approximate cost to repair. Several knowledge-based software tools are available that can be preprogrammed to parse code, identify date references, and repair code. In general, the types of tools available to support the Year 2000 effort include the following:

- Date reference search and locate packages (include browsers, parsing technology, clock simulators, etc.).
- Code editors, debuggers, and code generators.
- Configuration Management (CM) tools, useful for managing the repair process and controlling versions of the software as they are modified, tested, and released.
- Cost estimating, which includes systems that determine costs and schedule based on metrics (such as number of LOC).
- Testing software to support unit, module, system, integration, stress, and environmental testing.

Factors to Consider When Selecting Tools

Select tools based on careful consideration of their potential benefits against cost or other resources required for their use. *As with any purchase, be an informed consumer.* The Year 2000 process cannot be fully automated for any system.[3] Tools do not "solve" the problem; they support and enhance the problem resolution process.

When selecting tools consider the following factors:

- Cost, availability, licensing issues, and technical support.
- Applicability to the technical environment; many tools are language-oriented and/or platform-specific.
- Ease of use. Is training or technical expertise required to use the tool?
- Versatility. Can the tool support multiple platforms and/or languages?
- Support for multiple stages of the Year 2000 conversion process, e.g., assessment and conversion, etc.
- "Pre-programming" requirements. Will the tool require customization? Who will do the customization?
- Vendor reputation. Will the vendor fully support the tool? Is the tool warrantied/guaranteed?
- Success stories. Who else has used the tool? How well does it work?

Decide which tool(s) will provide the most benefit and not seriously impact the Year 2000 budget. Several Internet sites provide information on tools and their vendors, including:

- http://www.itaa.org
- http://www.mitre.org/research/y2k/docs
- http://www.itpolicy.gsa.gov/mks/yr2000/y201toc1.htm

Appendix C includes excerpts from an EPA document on Year 2000 tools. The appendix includes considerations for selecting tools and vendors and provides a list of tools and vendors supporting different stages of the Year 2000 repair effort.

Note: Evaluation of specific tools may be obtained from ETSD via a Working Capital Fund (WCF) request for services.

4.7 REPORTING YEAR 2000 STATUS

All agencies are facing Year 2000 reporting requirements. The Office of Management and Budget (OMB) has required cost estimates for addressing the Year 2000 problem as part of the budgeting process. The Budget Division of OMB issues OMB Circular A-11, *Budget Preparation*, which includes Year 2000 reporting requirements. All agencies prepared estimates of the cost of Year 2000 activities as part of Exhibit 43C. It is anticipated that OMB will require quarterly monitoring and reporting by agencies.

Within EPA, offices will report status to the Year 2000 Workgroup. The Workgroup will be surveying system managers and collecting results of the inventory and planning efforts, tracking completion of tasks, and preparing Agencywide reports to EPA's CIO, oversight agencies such as OMB and the General Accounting Office (GAO), and Congress.

Endnotes

1. The Information Technology Association of America, *What are You Waiting For? Start Preparing for Your Year 2000 Software Conversion Today*, (Arlington, VA, March 1996).

2. Ian Summerville, *Software Engineering*, Addison-Wesley Publishing Company, (Wokingham, England, 1989).

3. An article published by Peter de Jager, a well-known Year 2000 consultant, discusses why there can be no fully automated solution, i.e., a "silver bullet," to the Year 2000 problem. The article, "*Biting the Silver Bullet,*" is available on the Internet at http://www.year2000.com/.

SECTION 5.0 - INVENTORY

A critical step in addressing the Year 2000 problem is identifying all systems that could have problems processing the year 2000 date. Date references will be found throughout EPA systems: application software contains date computations, comparisons, and code that outputs dates to databases and files. Historical databases contain two-digit year fields. In addition, system hardware components will be affected by the rollover to the year 2000.

> "Date references can appear in programs, data bases, functions, operations, queries, procedures, screens, libraries, datasets, data definitions, jobs, transactions, and load modules...."[1]

What Is a Year 2000 Inventory?

The Year 2000 system inventory is simply an itemized list of systems and their components. Creating the inventory is the first step toward identifying, evaluating, and solving date-related problems.

Why Is the Inventory So Crucial?

A comprehensive system inventory is absolutely essential. Failure to identify all systems and databases as part of the inventory process can result in significantly underestimating the time and resources required to repair Year 2000 problems. In addition, as interrelated systems are repaired and tested, systems and databases omitted in the inventory process will be discovered, resulting in significant delays. This section provides guidance for conducting an inventory.

Inventory Objective

Identify and document all components of the automated system.

Inventory Steps

- Conduct Pre-Triage.
- Identify Inventory Requirements.
- Complete the System Inventory.
- Screen Inventory.

5.1 CONDUCT PRE-TRIAGE

Pre-triage is the process of eliminating or postponing the assessment of "noncritical" systems without apparent date problems to save time and resources. Examine the system to determine if it does not require a detailed inventory. However, pre-triage must, at the very least, include a summary inventory to minimize the risk that date-related fields or problems will be overlooked.

Be forewarned that a hasty decision at this stage can have serious consequences down the line. A key factor in the breadth of the Year 2000 problem is that many systems may contain operations having, or affected by, the use of two-digit date fields; the length of each and every date field cannot be readily ascertained without examining the code. Keep in mind that any code or procedure (Job Control Language (JCL), sorts, etc.) that could perform a date (time) computation; contain logic based on date; output date to screens, reports, and files; or obtain date stamp information from an operating system clock, must be evaluated for date problems.

Potential Exclusions

Systems that fall into the following three categories can potentially be excluded from the inventory, assessment, and repair process:

- Systems that already have been repaired and certified as Year 2000-compliant.
- Systems scheduled for replacement or retirement before they reach their event horizon.
- Systems that will not experience Year 2000 problems *AND do not interface with other systems.*

A fourth category is a Level 3 or Level 4 system whose repair may be postponed.

Guidelines

The following guidelines are for use in determining whether a system fits into one of the above four categories and either does not need to be assessed and repaired, or can be postponed.

Year 2000-Compliant Systems

Carefully test any system that has been repaired to ensure that it meets the requirements established for Year 2000 processing (provided in Section 7.1). A system or third-party vendor software may claim to be Year 2000-compliant, but if it does not follow EPA's date standard—CCYYMMDD—compatibility issues may arise if the following conditions exist:

- The system interfaces with other systems.
- The system outputs date data for future use.

In addition, if the system uses another approach for Year 2000 compliancy, such as a logic workaround, keep in mind that Year 2000-type problems have not necessarily been repaired; they may just have been postponed.

Systems Scheduled for Replacement or Retirement

Review systems that will be replaced or retired prior to the Year 2000 to ensure that they will not experience a Year 2000 event horizon, such as FY 2000 processing on or before October 1999. In addition, if these systems store data for future use, they could be a potential source of "date contamination" for future systems. Remember that historical data may need to be accessed and may still require modification.

Systems That Will Not Experience Year 2000 Problems

Be absolutely sure that the system is free of all date problems. This category requires careful scrutiny. As stated earlier, date references or the problems caused by date references may not be readily apparent. Section 6.1 contains a list of some of the more common date-related problems found in application software.

Systems That Can Be Postponed

Given the time constraints in dealing with the Year 2000 problem, systems classified as Level 3 or Level 4 may have to be set aside in order to focus resources on more critical systems. If this is a Level 3 or Level 4 system AND does not interface with other systems, consider the other systems that must be inventoried, assessed, and repaired. Decide if the inventory and assessment for this system should be postponed in order to focus on other more critical systems.

5.2 IDENTIFY INVENTORY REQUIREMENTS

The primary objective of the Year 2000 inventory is to identify and characterize all components of the automated system (i.e., applications, application modules, communications, interfaces, devices, etc.) that could potentially be affected by the Year 2000 problem. System components must be clearly and uniquely documented to facilitate the Year 2000 assessment.

Level of Risk

The level of risk should be the primary factor in determining the level of detail that must be included in the inventory. A detailed inventory is necessary for systems where the risk of loss or harm due to undiscovered date problems would be significant. EPA systems classified as Level 1 and Level 2 systems fall into this category.

Identify Types of System Components to Be Included in the Inventory

The most obvious component types are customized applications, associated application interfaces, databases, and data files. Archived/historical data must also be identified in the inventory process so that it can be assessed for Year 2000 problems. The date problem can also affect hardware devices, operating system software, and software provided by third-party vendors (for example, Oracle, IBM, Microsoft, etc.).

Identify Information to Be Collected for Each System Component

The type of information to be collected for each system component will vary depending on the nature of the system and/or system component. Exhibit 5-1 lists system components that must be included in a comprehensive system inventory and descriptive information that should be included for each system component. System documentation, if it is current, will provide a source for much of this information.

The information gathered for the inventory will be used as the basis for identifying date-related problems. If the inventory information is unclear, duplicative, or incomplete, staff conducting subsequent assessment activities may have to recreate portions of the system inventory.

Develop Inventory Procedures

Deciding exactly how to conduct the inventory is as important as deciding what information to include in it. Given the time constraints for dealing with Year 2000 system changes, it is important to have a "roadmap" of written procedures to follow

Exhibit 5-1. Year 2000 System Inventory Components

Component	Descriptive Information
System Platform	• List the platform (hardware), operating system version and release, and point of contact (POC) for the system.
Application Software	Many systems consist of multiple applications and/or modules coded in different software languages. • List the application/application module name, software language, version/release (where applicable), and the system manager.
Databases, Files, and Indexes	• List the source of data and applications, or application modules, that use the data.
Historical/ Archived Data	• Document how often and across what time periods (i.e., 1 year, 5 years, etc.) historical data must be retrieved.
System Interfaces	• Document the type of interface (data, screens, reports, etc.). • Identify and list the person responsible for maintaining the system interface.
Peripherals	• Include all PCs, routers, and printers. • List hardware type, associated software, and version and release.
Software Utilities/Third-Party Vendor Software	This list of other system software should include compilers, access control software, and other third-party vendor software. • List software name/title, vendor, version and release, and point of contact.
Existing System Documentation	This documentation may include source code, design specifications, configuration diagrams, etc. • List the type of documentation and location.

during the inventory. The procedures must define how the inventory will be completed and establish who will be responsible for identifying particular systems or system components. Although this may sound trivial, examining system components that could cross platforms, languages, and organizational boundaries will be difficult.

Decide How the Inventory Will Be Documented

Keep a master list of the inventory information from each staff member in an automated format, such as a spreadsheet or database, if possible. This will facilitate updates, research, and future reporting. If the inventory is not thoroughly documented now, at least some of it may have to be recreated as part of the assessment phase. Even worse, undocumented system components could show up as problems during repair, testing, or implementation.

5.3 COMPLETE THE SYSTEM INVENTORY

During the inventory process, verify that system components are being documented at an appropriate level of detail. The inventory process may be iterative, requiring several rounds of investigation to identify all significant system components.

Remember that external system interfaces will be a critical issue in the Year 2000 effort. All system interfaces and external sources of data must be documented as part of the inventory process. External sources of data may represent significant risks to meeting Year 2000 deadlines; there is no guarantee if or when externally provided data will be made Year 2000-compliant. In addition, data from external sources containing dates in a two-digit year format will cause problems when added to a database that has already been changed to a four-digit year format.

5.4 SCREEN INVENTORY

When the master inventory list is complete, review the systems to determine if there are systems that should be excluded from the next stage—the assessment process. Any systems that can be excluded from the assessment process because they are not affected by the Year 2000 problem can save valuable time and resources. *Carefully review the criteria provided in Section 5.1, Conduct Pre-Triage, before excluding any systems from subsequent Year 2000 assessment activities.*

5.5 ADDITIONAL CONSIDERATIONS

In completing and documenting the inventory, situations may arise that have special implications for the inventory itself, or, that could affect subsequent steps in the Year 2000 process.

- **Missing or outdated source code.** During the inventory process, executing software may be discovered for which there is no source code. Experience within the computer industry has shown that a lack of source code is not uncommon. In addition, there may be executables that don't exactly match the source code. Identifying the full functionality of the executing code and recreating the source code will add additional time to the coding process.

- **EPA forms.** Pre-printed forms may have the first two digits of the year already printed—for example, "19__." In some cases, these forms may be scanned into the system resulting in "date contamination." These types of forms should be included in the inventory, assessment, and repair process.

- **Standard system software.** Do not ignore "standard" or packaged system software. Some standard system software may experience Year 2000 problems. For example, access control packages that do not accept 4-digit years may trip system intruder alarms. The March 1996 *CSL Bulletin* from NIST, "Millennium Rollover: The Year 2000 Problem," cites an example of tape management software attempting in 1995 to scratch tapes with a retention date of "12/31/00."

Endnotes

1. The Information Technology Association of America, *What are You Waiting For? Start Preparing for Your Year 2000 Software Conversion Today*, (Arlington, VA, March 1996).

SECTION 6.0 - ASSESSMENT

Which systems will fail or return inaccurate results when the year 2000 arrives or when dates are projected past December 31, 1999? This is the question to ask while assessing the system inventory. But, date references are so pervasive throughout automated systems, this question may not be easy to answer. It has been estimated that if the date references in the federal government's information systems are not found and repaired, as many as 80 percent of these systems will fail.[2]

> "There's no way you can get all your systems converted by the turn of the century, experts say. Instead, pick the most important ones, find out which will be most affected, and concentrate on them."[1]

What Is a Year 2000 System Assessment?	A Year 2000 system assessment is an in-depth review of all inventoried system components to identify date references and determine if these references will cause the system to fail or produce inaccurate results.
Why Is the Assessment So Important?	The resources needed to fix Year 2000 problems are significant, both in terms of dollars and knowledgeable staff. Consequently, it may be risky to assume there will be enough resources to fix all systems. Assessing the inventory for Year 2000 problems will identify which systems have date problems, which systems are most critical, and how resources can be best used. This section provides guidance for setting priorities and pinpointing the problems.
Assessment Objective	Identify Year 2000 problems and the resources to correct them.
Assessment Steps	• Formulate the Year 2000 Assessment Procedures. • Complete the Year 2000 Assessment. • Document the Assessment. • Evaluate the Results. • Triage the Systems.

Use a Risk-Based Approach

Think of the Year 2000 assessment in the same light as a risk assessment. Evaluate the risk and likelihood of failure, the impact of system failure on mission and day-to-day operations, and the cost to mitigate the impact of system failure.

6.1 FORMULATE THE YEAR 2000 ASSESSMENT PROCEDURES

Why not just jump in and start coding? Experience has shown that just "jumping in" and fixing the code is not the most efficient approach to repairing Year 2000 problems. Given the extent of the problem, just jumping in and coding could mean using valuable resources on not-so-valuable code. Comprehensive assessment procedures, in conjunction with the appropriate automated software "tools," will assist in identifying which code is critical and must be repaired.

To Meet the Goal, Answer the Date Questions

Assessment Goal. The goal for the assessment process is to identify how extensively the inventoried system is affected by the Year 2000 problem. To achieve this goal, find answers to the following questions:

- Where are the date references?
- Will a date reference cause a problem?
- Are there other date-related problems?
- Are historical archived data affected?
- Is any hardware or firmware affected?

- **Where Are the Date References?**

Date-related problems are due to the use of a two-digit year (i.e., "96" rather than "1996") in code that performs computations and comparisons or outputs dates to databases, files, screens, or reports. Identify all uses of a date and determine which have two-digit years.

- **Will a Date Reference Cause a Problem?**

As date references are found, each occurrence must be evaluated to determine if it will have an impact on a system's ability to operate correctly or interface with another system. Will the date, as coded

in the software, produce inaccurate data and results? Identify the effect of the date reference in software utilities and procedures, such as sort routines or JCL.

Examples of Date Problems

Examples of date problems resulting from the use of two-digit years include the following:

- **Incorrect elapsed time calculations**, such as age or permit expiration dates. This could have serious implications if the software controls access to computers, data, or facilities, or determines disposition schedules (tape library management software).

- **Incorrect data sequence**, such as "00" sorting before "99." This problem may not be limited to sort routines that order data for output to a screen. Data may be indexed on a date field, resulting in sequence problems for the index.

- **Are There Other Date-Related Problems?**

Year 2000 problems can show up as more than just a two-digit year field. Locate any other date-related problems in the inventory, including the following:

- **Hard-coded years**. Software may use a hard-coded "19" for the first two digits of the year field in date operations, or on a report, screen, or as output to a database.

- **Hard-coded dates used to signify special processing**. Years of "00" or "99," or such dates as "12/31/99," have been used by programmers to signify special processing conditions.

- **Software edit routines containing hard-coded values for verifying dates**, such as "year must be > 80."

- **Incorrect leap-year calculation**. The year 2000 is a leap year; however, it has been widely noted that the leap-year calculation is incorrect in many software applications. The *EPA Date Standard* defines a leap year as follows: "a year whose number is divisible by four an integral number of times, except that if it is a centennial year, it shall also be divisible by 400 an integral number of times."

- **Noncompliant date formats**. Software might have a four-digit year format but still not follow the *EPA Date Standard* (CCYYMMDD)—for example, Microsoft's default date format is in reverse order: MMDDYYYY (Microsoft includes an option to change to the YYYYMMDD standard).

- **Are Historical Data Affected?**

Review historical, or archived, data to determine if the data contains date fields with two-digit years. To assess the impact of date fields containing two-digit years, answer the following questions.

- Why are the data retained? Are historical data required for legal or regulatory purposes?
- How long must the historical data be retained (days, months, years, etc.)?
- How frequently are the historical data accessed (every day, monthly, annually, rarely retrieved, etc.)?
- What is the maximum age of the data that is accessed, i.e., how far back must the data be retrieved: 1 to 2 years, up to 5 years, etc.?

- **Is Hardware Affected?**

Identify date problems within the hardware components of the inventory. The firmware, or microcode, controlling the hardware may be based on two-digit years, or may not be able to correctly process four-digit years. PCs are an example of equipment in this category. For many PCs, the BIOS accepts four-digit years; but after December 31, 1999, the system date will roll back to January 1 or January 4, 1980 (the "birth date" for DOS). The date problem is not limited to older PCs; even some newer Pentium models do not roll over to the year 2000. There may be other hardware in the inventory that will need to be repaired, upgraded, or replaced to take care of Year 2000 problems.

Develop Assessment Procedures

Develop procedures for identifying date references and date problems. Because the Year 2000 assessment will most likely be completed through a combination of manual code review and the

use of automated software tools, the assessment process must be clearly documented.

6.2 COMPLETE THE YEAR 2000 ASSESSMENT

All date references must be identified and carefully evaluated at this stage to ensure successful repair, implementation, and testing. The assessment process should be based on formally documented procedures. This section discusses the following four key steps in Year 2000 assessments:

- Search for date references.
- Evaluate date problems.
- Revise contingency plans.
- Identify potential problems.

Search for Date References

Date references are identified by searching for "date" fields and date routines. Search through source code and data dictionaries. Use system documentation to verify that all date operations have been identified. IBM's *The Year 2000 and 2-Digit Dates: A Guide for Planning and Implementation* provides a list of character strings often used to represent date fields. Some of the more common character strings are included in Exhibit 6-1.

Exhibit 6-1. Sample Date Field Character Strings

DATE	DAT	DTE	DT
DAY	DA	DD	DDMMYY
DDDYY	DOB	DOH	CCYY
CCYYDDD	CURR	CURRENT	MDY
MMDDYY	MMYY	YEAR	YR
YY	YMD	YYMMDD	YYDDD
TIME	TIMESTAMP	TIMEDATE	THISDATE

This list is not inclusive. Other software languages may use special field names when requesting time, day, month, year, etc. from the system. In addition, because most field names are chosen by programmers, the list is limited only by the creativity of the programmer. For example, any field name ending with a "_dt"

could represent a date field. Furthermore, don't assume that a date field with a length of eight characters is correct. A field with a length of eight characters, defined as character or alphanumeric, may mean that the field is used to store the date as "MM/DD/YY" or "YY/MM/DD."

Evaluate Date Problems

As part of the assessment, evaluate the impact of the Year 2000 problem—the rollover from the year 1999 to 2000—or projected dates, such as FY 2000, based on the format of the date field. *For government systems, the year 2000 effectively begins on October 1, 1999.* In addition, evaluate the impact of other date-related problems identified, i.e., incorrect leap-year calculation, hard-coded date values, etc. Exhibit 6-2 outlines criteria for evaluating impacts.

Exhibit 6-2. Impact Criteria

Impact	Criteria
1. Fatal	The application will terminate abnormally or the system will fail due to Year 2000 hardware or operating system software problems.
2. Immediate	The software will execute but will experience one of the following problems: • Produce incorrect results (e.g., invalid expiration dates or ages). • Transfer or receive date data in a two-digit year format. • Contains a date format incompatible with interfacing system or third-party software.
3. "Cosmetic"	Contains a two-digit year where the year can be inferred AND the date is NOT used for computation, comparison, or output to other systems. For example, screens or hardcopy reports in which the user can infer the year. Before assuming this impact, ensure that reports are not expected to be transferred to files or scanned as input for another application. Keep in mind that some dates may not provide enough information from which to infer the year, e.g., "01/02/02."

Exhibit 6-2.
Impact Criteria, *cont.*

Impact	Criteria
4. Delayed	The date is **not** exchanged and the application or system interprets the first two digits for a two-digit year field based on logic, such as "if two-digit year > 69, then first two digits = 19" or "if two-digit year < 15, then first two digits = 20."
5. None	The date format is compliant with the *EPA Date Standard,* or another four-digit year format is used.

Revise Contingency Plans

Revise system contingency plans based on the problems identified. If the system will not function or will not function correctly, plan for how the functionality supported by the system can continue should the Year 2000 project not be completed in time. Tailor the contingency plan in accordance with the criticality of the system.

Identify Potential Problems

Review the information gathered in the inventory and assessment process to identify issues or potential problems that could increase the cost of the Year 2000 project or affect the ability to complete the project on time.

Such problems might include the following:

- **Third-party vendor software cost.** Software provided by third-party vendors (also referred to as commercial-off-the-shelf [COTS] software) may need to be upgraded or replaced. These Year 2000-compliant software fixes may not be free.

- **Delays by external partners or vendors.** There is no guarantee as to when data or software provided by external parties will be fixed.

- **Support for multiple date formats.** Identify whether it is necessary to maintain the capability to transmit or receive data in multiple date formats. This may entail the design and implementation of software (i.e., bridge programs) to fix date formats "on the fly."

6.3 DOCUMENT THE ASSESSMENT

The master inventory list can be used to record the results of the assessment. At this point, the assessment documentation must include the following information:

- Total Lines of Code (LOC) and total LOC containing date problems (list by application).
- Custom utilities or procedures with date problems.
- Hardware or third-party vendor software requiring replacement or upgrade to be Year 2000-compliant.
- Impact of the date problem within the system components, i.e., fatal, immediate, cosmetic, delayed, or no impact.

6.4 EVALUATE THE RESULTS

Evaluate the information gathered during the assessment process and identify the resources required to correct the date problems identified. The cost of the Year 2000 problem is calculated primarily for application software—for reviewing, repairing, and testing code. Other potential costs must be included to ensure that the Year 2000 project is adequately funded from the assessment stage through repair, testing, and implementation.

Calculate Cost per LOC

Year 2000 information sources report the cost as dollars per LOC. The costs range from a low of .40¢ per LOC to a high of $8.00, depending on the source. Review the Year 2000 cost sources (listed in Appendix E) and choose a cost factor that best suits the system environment. If the code is well-structured (not complex) and the system does not interface with other systems, the cost per LOC may be at the low end of the range. If the system is large, complex, and not well-structured, the cost will more likely be at the high end.

Cost Factors

Keep in mind that the costs developed during this stage are averages. When calculating total costs, consider the following factors that can increase the cost of the project:

- Total lines of code and percent of code affected.
- Complexity of code.

- Type and number of system interfaces.
- Language the application was coded in and availability of skilled programmers to repair the code.
- Availability of source code.
- Availability of system documentation.
- Availability of Year 2000 or other third-party software to support the assessment and repair process.

Calculate the total cost and document the cost factor(s) used in determining the cost.

Calculate Other Costs

Add to the costs calculated for the application software the costs that could be incurred for related software and hardware. These items can include:

- Third-party vendor software or hardware requiring replacement or upgrades.
- Software tools purchased for the inventory, assessment, repair, or testing stages.
- Software, such as bridge programs or manual data entry programs, to be developed to support modifications.
- Additional disk space or Direct Access Storage Device (DASD) required to support expanded file structures.
- Costs for the testing environment. Determine the costs for the additional testing space required or for special testing environments (typically a stand-alone or otherwise segmented platform).

Finding the staff and funds to fix all Level 1 and Level 2 systems (mission critical) in the limited time remaining may be difficult. Level 3 and Level 4 systems may need to be evaluated to determine if their repair can be postponed. The criteria presented

in Section 6.5 for identifying system priorities can also be used to determine whether a Level 3 or Level 4 system is so important that its repair should not be postponed.

6.5 TRIAGE THE SYSTEMS

Three factors that determine the ability to repair systems and data follow:

- The amount of **time required vs. time remaining** to fix, test, and implement the changes.
- The number of **knowledgeable staff members** available to make the changes.
- **Availability of funding.**

Conducting a triage exercise will assist in making the best possible use of Year 2000 resources: time, staff, and funding. The sections that follow provide insight on how to triage information systems.

Definition of "Triage"

Webster's Dictionary defines triage in a broad sense as: *the assigning of priority order to projects on the basis of where funds and resources can be best used or are most needed.*[3] Webster's also defines triage in a medical sense as: *the sorting and allocation of treatment to patients . . . according to a system of priorities designed to maximize the number of survivors.*[4]

Both definitions apply to the Year 2000 triage process. Applying the first definition to Year 2000 triage means that systems must be assigned a priority to ensure the best use of Year 2000 resources. Applying the second definition means that the Year 2000 triage must focus on sorting through the system inventory and allocating "treatment," i.e., resources, to systems to maximize the number of "survivors"—systems that will survive the transition to the year 2000. The concept of maximizing survivors is now an important factor to consider in the Year 2000 repair process because the amount of time remaining to fix, test, and implement changes is rapidly decreasing.

System Categories

To focus resources on systems most likely to survive the transition to the year 2000, determine into which of the following three categories the system fits.

(A) **Systems not requiring repair.** Systems that will not need resources because they do not have a Year 2000 problem or will be replaced either by modernization efforts or total replacement before the year 2000.

(B) **Systems likely to survive.** Systems that can be successfully repaired.

(C) **Systems that probably won't survive the Year 2000 problem.** Systems that may not be repairable—even if repair is attempted, the effort is not likely to be successful.

Systems likely to survive (B) is the only category of systems that must be reviewed and assigned a priority order; the systems in the first and the third categories will be ignored. The following subsection discusses how systems are placed in each of these categories.

Determining the System Category

Systems not requiring repair (A) have already been identified at several stages in the Year 2000 process: during the pre-triage and screening process in the inventory stage (Sections 5.1 and 5.4); and during the completion and documentation of the assessment process (Sections 6.2 and 6.3). Review these sections if any question remains as to whether the system needs to be repaired.

Systems that probably won't survive the Year 2000 problem (C) may be more difficult to identify. To categorize these systems, review the answers to the following questions:

- How many date references and date-related problems requiring repair were found during the assessment process? Is historical data affected? Are operating system software and hardware affected? What is the impact of the Year 2000 problem on the system?

- How complex is the code? Is the code well-structured? What software language(s) was used? How difficult will it be to repair?

- Are the resources (time and programming expertise) available to repair the system so that it will still function when it reaches its event horizon?

Systems permeated with date problems (in source code, operating system software, microcode, data, and hardware), and whose source code is complex and would be difficult to repair may fall into category C. But, ask one final question: *Can the organization continue functioning without the system?* If the answer is "No," then the system cannot be set aside; it must be assigned a priority along with the systems likely to survive (B). *However, give careful consideration to a potentially "unrepairable" system when assigning priorities for repair. Will the resources required to repair this system, including time, justify this system's repair over others in your inventory?*

All remaining systems fall into category B: **systems likely to survive**. Following the broad definition of the term "triage," assign a priority order to these systems based on the importance of the system and the best use of available Year 2000 resources.

Assigning Priorities

One approach to assigning priorities to systems is to review and assign weights to them based on the criteria shown in Exhibit 6-3. These numeric scores can assist in determining an order for repair. However, system priorities cannot be made solely on a numeric value.

Another approach to assigning priorities might be to prioritize systems based on level of risk or identifying systems with less severe date problems. For example, determine if "cosmetic" or "delayed" problems can be postponed. In addition, review other factors for identifying the most critical systems. EPA's systems are already classified in terms of criticality and cost, i.e., Level 1, 2, 3, and 4. Nonetheless, other factors may contribute to the importance of the system and assist in determining the order in which the systems will be repaired.

Exhibit 6-3. Triage Weighting Factors

Criteria	Weight
Level 1 System	40
Level 2 System	30
Level 3 System	20
Level 4 System	10
System Interface	10 (for each interface)
Add weights based on the evaluated impact of the date problem	
Fatal	40
Immediate	30
Cosmetic	20
Delayed	10

Additional Ranking Factors

The following list suggests additional ranking factors:

- How important is the system to the organization's day-to-day operations?
- Would the unavailability of the system have a significant impact on the public?
- When would the event horizon occur?
- How would it affect partners in state and local governments?
- What effect would inaccurate data have?
- Does the system support key Agency initiatives, such as electronic reporting?
- Are there security issues?

- Can a complete repair be accomplished? Is a lot of source code missing? Will it be very difficult to find staff with the skills to repair the systems?

Choose the most cost-effective use of Year 2000 budgets. All critical systems must be repaired, and those critical systems that interface with other systems may need to be repaired first. For non-critical systems, consider how important each system is and how much it will cost to repair. For non-critical systems that are equally important, it may be more cost effective to fix two (or more) less costly systems than one very costly system.

Repair, Test and Implementation Schedule

Establish a repair, test, and implementation schedule, with the most critical systems/components listed first. This schedule must account for the factors discussed at the beginning of this section:

- Time required to make the changes.
- Time remaining before the system will experience date-related problems.
- Staff available to make the changes.

Review system interfaces and identify which interfaces will be repaired first.

System Maintenance

The repair and implementation schedule must take into account system maintenance activities. Necessary maintenance must continue. However, Year 2000 repairs will take precedence over most system maintenance, because the problem must be repaired if the system is going to continue to function. It is strongly recommended that enhancements, i.e., new functionality, **not** be added in conjunction with the Year 2000 date changes, because they might have a significant impact on repair, testing, and implementation.

6.6 ADDITIONAL CONSIDERATIONS

The following are additional considerations pertinent to the assessment process.

- **Definition of "one line of code."** A line-by-line code review is the standard approach for determining Year 2000 costs. However, a standard definition does not exist as to what is actually considered one line of code. A line of code could be one physical statement or one logical statement. When using a tool, identify which method is used. Document which approach is used when calculating costs.

- **Other methods for determining software costs.** "Function points" can also be used in estimating the cost of repairing application code. Capers Jones discusses the use of function points in the document "*The Global Economic Impact of the Year 2000 Problem.*"

Endnotes

1. Richard Adhikari, "Approaching 2000," *Information Week, Oct. 7, 1996.*

2. EPA Executive Steering Committee for IRM, *"Agency Year 2000 Management Strategy and Budget Plans."*

3. Merriam-Webster, Inc., *Merriam-Webster's Collegiate Dictionary*, 10th ed., (Springfield, MA, 1995).

4. Ibid.

SECTION 7.0 - YEAR 2000 REPAIR

At this stage in the Year 2000 project, systems have been inventoried, date fields have been identified, and repair priorities have been established. In the repair stage, correcting all affected code and data without introducing new problems will be one of the key challenges.

> "The passing of data from one system to another, and from one agency to another, is one of the factors which gives the Year 2000 such complexity."[1]

What Is Meant by Year 2000 Repair?

Repairing the Year 2000 problem is the correction of application code and data affected by the use of two-digit years, as well as the correction of other date-related problems identified in the assessment process.

Isn't This Just Corrective System Maintenance?

In many ways, the process of repairing affected application code and data is the same as most corrective system maintenance efforts. Standard EPA system maintenance procedures must be followed for changing and documenting the code. However, the Year 2000 project faces two issues that most system maintenance efforts don't face: 1) the pervasiveness of the problem in application software, data, and system hardware; and 2) an immovable deadline. The key challenge to the Year 2000 repair is to stay on top of these two issues and, at the same time, not introduce new problems into repaired systems.

Year 2000 Repair Objective

Repair all affected applications and data, without introducing new problems, in time for thorough system testing.

Repair Steps

- Document the requirements based on the date problems identified.
- Identify cost-effective solutions.
- Modify affected applications and data.
- Document system changes.

7.1 DOCUMENT REQUIREMENTS BASED ON DATE PROBLEMS IDENTIFIED

In the assessment stage, all problematic date references were identified and evaluated. Review the date problems identified and document the requirements for the system's date processing and date exchange functions (per the *EPA Date Standard*, EPA systems must exchange data in the format "CCYYMMDD").

Date requirements will depend on the type of date processing, such as the use of date computations, projected dates, elapsed time calculations, etc., or date exchange functionality. Exhibit 7-1 lists typical Year 2000 application requirements.

Exhibit 7-1. Year 2000 Application Requirements

Note: Not every requirement will apply to every application (i.e., some systems may not calculate elapsed time, leap years, etc.).

• Does not terminate abnormally for pre- or post-year 2000 dates.
• Correctly performs the following date operations: leap year, elapsed time (i.e., age and number of days, weeks, months, etc.), and correct day-of-week, month-of-year, etc.
• Correctly compares dates across centuries.
• Does not contain hard-coded values for the first two digits of the year—either "19" or "20."
• The date, as input to the system, is stored correctly. For example, a date input as "1999" is stored as "1999," not "99."
• Screens output the date correctly (some occurrences of two-digit years in system **output** may have been identified as cosmetic problems only and may not need to be changed to four-digit years). Screens accept four-digit dates or an indicator that specifies what the first two digits of a two-digit year field should be, i.e., "19" or "20."
• Reports show correct dates (cosmetic dates may stay as two-digit dates).
• Correctly processes the system's event horizon (i.e, fiscal year 2000, 2010, etc.), not just the rollover to the year 2000.

Exhibit 7-1. Year 2000 Application Requirements, *cont.*

• All system interfaces exchange date fields according to the format defined in the *EPA Date Standard*: "CCYYMMDD."
• Software edit capabilities accurately flag incorrect years.
• Software does not use special dates, such as "12/31/99," or year values, such as "00" or "99," to signify special processing conditions.

Year 2000 Hardware and Operating System Requirements

Hardware and operating system requirements should include the following:

- The system date increments correctly, without human intervention, for the year 2000 and beyond.
- As the date increments, the date change does not result in an error condition for operating system software or system hardware.
- All sequencing operations, sorts, indexing, etc. must complete with date data in the correct (or otherwise specified) order.

Historical Data Requirements

Document access/retrieval requirements for historical, i.e., archived, data containing two-digit years. In the context of this document, historical data is defined as data that is not considered part of day-to-day system operations (not accessed, output, etc. on a daily basis).

Current Data Requirements

Date data containing a two-digit year that is output from the system must be expanded to four-digit years *if the data will be exchanged with another information system* (see discussion below). Many commercial DBMSs are Year 2000-compliant, using algorithms to store date data. Make the decision to expand other date data based on an assessment of the cost effectiveness of expanding the date data versus using the benefit/cost of other solutions to the problems inherent in using two-digit years. Section 7.2 discusses the alternative approaches to solving the Year 2000 problem, advantages, and disadvantages.

Requirements for the Interchange of Calendar Date Data

The *EPA Date Standard* prescribes the format for date data that are exchanged with other information systems: CCYYMMDD. In addition, the standard requires compliance with the following criteria **"for the interchange of calendar date data"**:

"Agency information management systems must include the intelligence to accurately process date according to the following criteria:

(1) Calculations must execute, using dates with a four-digit year.

(2) Functionality, both online and batch, including but not limited to entry, inquiry, maintenance and update, must support four-digit year interfaces.

(3) Interfaces and reports must support processing of four-digit year.

(4) Successful transition into the year 2000, using the correct system date, must occur without human intervention.

(5) After transition to the year 2000, processing with a four-digit year must occur without human intervention.

(6) Calculations that determine and process leap years must be accurate.

(7) Correct results in forward and backward data calculations that span Century boundaries must be provided, including the conversion of previous years currently stored as two digits.

(8) Date data that are output for data interchange must be in the format specified above for standard representation of calendar date [CCYYMMDD]."

Provision for Waiver

The *EPA Date Standard* does note that there may be some situations where exceptions to the standard would be warranted: "A waiver may be granted by the Agency's Chief Information Officer (CIO) for legacy applications that achieve compliance by means other than the use of this standard, or for systems where

the costs of implementing the standard are significantly higher than the benefits."

7.2 IDENTIFY COST-EFFECTIVE SOLUTIONS

The *EPA Date Standard* mandates the use of four-digit years for all date data exchanged with other systems. Because so many Agency systems exchange data, expansion to a four-digit year field will be at least one of the solutions used for many systems in solving the Year 2000 problem. There are various solutions currently being used to solve Year 2000 problem. These possible solutions include the following:

Year 2000 Possible Solutions

- Expansion to a four-digit year field.
- Use of a "windowing" approach: fixed-date or sliding-date windows.
- Conversion to an encoded format.
- No change for cosmetic dates.
- Use of bridge programs.
- Expansion to a three-digit year field or addition of a "year" indicator.

Carefully consider each of the possible solutions. Each has advantages and disadvantages. The following paragraphs provide an overview of these potential solutions to the Year 2000 problem.

- **Expansion to a Four-Digit Year Field**

In the expansion approach, all occurrences of a two-digit year are expanded to a four-digit year. This solution requires changes to both the data and the application(s) using the data. *Note that some systems do not support four-digit years (OS/VS COBOL and VS COBOL II).*

Advantages: The expansion approach is consistent with governmentwide standards and the Agency's standard for system interfaces. In addition, the use of four-digit years results in easier updates and code with simpler date logic. Changing to a four-digit

year field is considered by many to be the only permanent solution to the two-digit year problem.

Disadvantages: Application code must be updated to reference the correct field size and changes in logic for date computations. In addition, *all* applications that reference or use the updated data will have to be repaired at the same time. The cost and time required for converting data may be significant. If the other applications cannot be repaired at the same time, a bridge program can be developed to perform the conversion between old and new data formats for unrepaired programs. Carefully plan the approach to updating screens and reports. Expanding every date field may have a significant cost or timing impact if the extra two digits result in the need to redesign screens and/or reports.

- **Use of a "Windowing" Approach: Fixed-Date or Sliding-Date Windows**

The windowing approach is based on the use of date windows: an algorithm using a year value as the context for determining what a two-digit year should begin with—"19" or "20." Date windows either use a *fixed date* or *a sliding date*. For a *fixed-date* window approach, the year value used to determine whether the year begins with "19" or "20" does not change. An example of this is as follows: *if year value > 50, then two-digit year begins with 19; if year value < 50, then two-digit year begins with 20.* The *sliding-date* window uses a year value that increments automatically every year, i.e., *year value + 1.*

Advantages: This approach eliminates the need to change data. Windowing may require significantly fewer changes to code than an expansion approach and be much more cost-effective for systems that will be retired or replaced soon.

Disadvantages: The obvious disadvantage is that this approach will not work if the dates exceed a 100-year interval. If, in the near future, the application must process dates that are beyond 100 years, the two-digit year field problem will recur. In addition, system performance may suffer depending upon how often the value for the first two digits of the year must be determined.

- **Conversion to Encoded Format**

This option is based on the use of an encoding scheme, such as conversion to a hexadecimal format, to store the full year in two

digits. IBM's Year 2000 planning document, *The Year 2000 and 2-Digit Dates: A Guide For Planning and Implementation,* describes various encoding options.

Advantages: Conversion to an encoded format eliminates the need for data expansion and additional disk space.

Disadvantages: Encoding is noted to be the least desirable approach. In addition, this option still requires changes to both data and applications.

- **No Change for Cosmetic Dates**

This alternative only applies to dates with two-digit years that have been determined to be cosmetic. The date is output for viewing only and the user can easily determine the year. For example, a user viewing a date formatted as 10/01/99 can easily deduce that the year is 1999.

Advantages: This approach does not require changes to either applications or data.

Disadvantages: The date may cause date integrity problems later. For example, the date appears on a report that is scanned as input to an application in the future. In addition, consider any ambiguity problems that can arise. For example, the date 01/02/02 may be difficult to interpret, especially as users become used to seeing dates in the EPA standard date format—2002/02/01.

- **Use of Bridge Programs**

Bridge programs are software programs that expand date data from a two-digit year format to a four-digit year format by adding a value for the first two digits of the year, e.g., "19" or "20." Bridge programs can be used as the application executes, or for one-time or on-the-fly data changes.

Advantages: May be very useful for historical/archived data that is not frequently accessed. Eliminates the need to change the data, and, in many cases, the need to change the applications.

Disadvantages: This solution does not correct the two-digit year problem and may impact other applications needing the data in the

future. The key problem with a bridge program is the overhead that it will add to system operations.

- **Expansion to a Three-Digit Year Field or Addition of a "Year" Indicator**

Expansion to a three-digit year format will encompass the same activities as expansion to a four-digit year format, except that a single digit is used to represent the first two digits of the year. A similar approach is using a "year" indicator—a code in another field to represent the first two digits of the year.

Advantages: This may be an advantageous approach if space is a critical issue.

Disadvantages: The three-digit year approach does not conform to government or Agency standards, but requires much the same effort as expansion to four-digits. Use of a "year" indicator may also require expansion to add a field (an existing field could also be used). The use of an indicator will require the addition of program logic to decipher the first two digits of the year.

Additional Information on Year 2000 Solutions

More information on Year 2000 solutions can be found on the Internet. Appendix B lists Year 2000 web sites containing information on Year 2000 issues and possible solutions. Note that there is much discussion as to the "correct" approach to solving the Year 2000 problem. Suffice it to say that expansion to a four-digit year field solves the problem for the longest period of time (until the year has to go from four-digits to five-digits). For the most part, the other alternatives resolve the two-digit year field problem for a specified period of time or under a specific set of circumstances.

Choosing the Solution

In selecting Year 2000 options or solutions, consider the factors that could affect the cost of using a particular solution. The following list includes factors that should be considered in choosing the most appropriate solution.

How Would It Impact the System?

- Does the solution have a significant impact on system performance? Would this be a critical issue? What affect would the solution have on system users? Would additional training be required?

Does Code Have To Be Repaired Anyway?

- How many other date-related problems (leap-year calculation, hard-coded year values, such as "19" or "20," edits, etc.) were identified in the code? How much more effort would be involved in a solution requiring date logic changes and/or field expansion changes?

How Long Will the System Be in Operation?

- Will the application be operational for a significant period of time after the year 2000? If the application will not be in use for a significant period of time after the year 2000, then a less costly solution should be considered.

How Will Historical Data Be Handled?

- Is a bridge program needed? If an application that processes historical data is repaired so that it can process four-digit years, the historical data may also need to be changed. It may, however, be more cost effective to develop a bridge program to change data from two-digit to four-digit years on the fly, rather than modify the entire set of historical data.

  The determination of what is most cost effective must be made based on the amount of data requiring repair and the cost to repair and store the expanded data, versus the cost to develop and maintain a bridge program for changing the data on the fly each time it must be accessed. Note that system performance issues may also need to be considered, as bridge programs will add more overhead processing to the system.

  Note that there may be some legal issues regarding the modification of some types of historical data. If the historical data are maintained for legal purposes, it may not be appropriate to tamper with (repair) the data. For example, electronic records of regulatory submissions, some categories of audit trails, or forms of "electronic evidence."

Based on the Solution Identified, Are There Other Requirements?

Other processing requirements may be identified based on the solution chosen for the application, i.e., expansion to a four-digit year versus a procedural solution, such as the use of a date window in which the first two digits of the year are logically deciphered. These solution-oriented date requirements are:

- The application has the capability to store the four-digit year.

- For two-digit year fields, the application can correctly decipher the first two digits of the year with 100 percent accuracy.

7.3 MODIFY APPLICATIONS AND DATA

Complete the detailed design, incorporating the requirements identified for date processing. If date data will be expanded to four-digits or bridge programs need to be developed, additional code will need to be designed and tested. Follow EPA standard procedures for project and program reviews; now is not the time to code quickly and have new application problems as a result.

Design the mechanism for checking the accuracy and completeness of repaired code and expanded data. If manual input is required, design the edit routines to ensure data quality.

7.4 UPDATE SYSTEM DOCUMENT-ATION

Documenting changes to the application code is extremely important to keep system lifecycle documentation current and provide information for the testing process. Consider the effort spent reviewing application code and the existing system documentation you had to use. If it was not as helpful or complete, now is the time to update it. Year 2000 literature notes that a benefit of the Year 2000 project is an understanding of the need for clearly documented programs and complete source code listings.

7.5 ADDITIONAL CONSIDER-ATIONS

Maintenance and enhancement are two important issues to address during the Year 2000 project. Application and database maintenance should be continued when degradation of the system would result from lack of maintenance. However, enhancements completed at the same time the date problem is repaired are likely to cost more and experience more problems. Changing, testing, and implementing multiple "fixes" or additional functionality will be especially difficult due to the Year 2000 project deadline and the need for equipment and experienced personnel. It is recommended that most, if not all, enhancements be postponed until after the Year 2000 problems have been corrected.

Endnote

1. *The Year 2000 Problem.* A Social Security Administration (SSA) white paper presented to FIRMPOC members in July 1995. Available on the Internet at: http://www.itpolicygsa.gov/

SECTION 8.0 - TEST YEAR 2000 CHANGES

Testing is one of the largest and most critical phases in the Year 2000 repair process. Documented experience from completed Year 2000 projects shows that testing requires a substantial effort—as much as 40 to 50 percent of total project resources.[1] Testing is the last chance to ensure that all Year 2000 problems in both software and hardware have been repaired.

> "Make sure that the systems are thoroughly tested—not only the fixes—regression test everything. . . .You only get one shot at making this right. It isn't like a replacement system that you can deliver and have running in parallel with the original system for six months or until you feel sure that it works. It's now or never."[2]

What Is Different About Year 2000 Testing?

Testing for the Year 2000 project is different than testing for typical system maintenance activities because of the scope of the changes and timing of the repair effort. It has been stated that repairing the Year 2000 problem represents the largest single integration effort ever attempted. So too with testing; Year 2000 testing will be the largest testing effort ever attempted.

What About EPA's Standards for Testing?

This section provides guidance on tailoring test plans for optimal Year 2000 testing: testing the changes made to date references and date operations in applications, data, and system procedures. This testing guidance does not replace EPA standards for system testing but rather provides supplemental information for testing to ensure that applications and data are Year 2000-compliant.

Testing Objective

Verify and Validate Repaired Applications and Data.

Test Steps

- Refine the Test Strategy.
- Follow a Standard Testing Approach.
- Develop Test Plans.
- Test and Document the Test Results.

8.1 REFINE THE TEST STRATEGY

To prepare for testing, review the test strategy documented during the Year 2000 project planning process and refine the initial strategy based on the scope of actual Year 2000 changes and the systems that will be tested. This review and refinement should include the following steps:

- Complete the master test plan.
- Review contingency plans.

Completing these steps will ensure an optimal testing environment. Each of these steps is discussed in the next sections.

Complete the Master Test Plan

The master test plan defines the procedures for conducting and managing the overall test process. These procedures should be defined at a higher level than application and data test plans. Application and data test plans will focus on whether specific functional and operational requirements have been satisfied.

Review Contingency Plans

Contingency plans will be important as the Year 2000 process moves into testing. Review and test contingency plans. Contingency plans in this phase must address potential problems within the testing environment to ensure that software and data are not accidentally corrupted or overwritten. In addition, ensure that the contingency plans answer the following questions:

- What will happen if the application software and hardware are not completed on time?
- What if corrected data is "contaminated" by uncorrected data?
- What will happen if there are problems within the test environment or with the test configuration?
- How can errors in applications and data be handled after implementation?

Contingency plans are especially important for application testing where stand-alone systems cannot be used, or are not feasible.

Changing the System Clock

WARNING!

A key test for Year 2000 readiness is to test the revised software, data, and procedures in an environment that simulates the Year 2000 by advancing the system clock; i.e., when the computer system date (clock) is reset equal to, or greater than, 20000101 (January 1, 2000). Even in a stand-alone testing environment, take care when changing the system clock. Be sure to make backups prior to testing *and test the ability to retrieve from the backups.* Simply changing the system clock can have major ramifications for some system platforms. For instance, software licenses may expire or access-control mechanisms may no longer provide access. This will be serious if you are the system administrator and no longer have access to reset the clock.

Simulation software is available to provide a "Year 2000" testing environment. Evaluate the software available for simulating Year 2000 within the current system environment. Review third-party software or access control mechanisms that could be affected by incrementing the system date.

8.2 FOLLOW A STANDARD TESTING APPROACH

EPA's standards for testing and documenting test results apply to the Year 2000 testing process. The standard testing process includes four stages of testing: 1) unit testing, 2) integration testing, 3) system testing, and 4) acceptance testing. Even though time is short, these test stages cannot be rushed. Careful unit testing must be conducted; errors found in the early stages of testing will have less impact on the project schedule than errors found in later testing stages. Such late-stage errors must be fixed and carefully retested: from the unit test for the changed code all the way through system and acceptance testing. Even though this retest process takes time, it cannot be omitted. This is the only chance to get it right: the Year 2000 deadline is immoveable.

Regression testing, testing to ensure that changes to the code have not corrupted the original software functionality, is normally included in system and acceptance testing. Regression testing is especially critical for the Year 2000 project. Because date fields and date operations are not always readily apparent, verifying that repaired software produces the same results for pre- and post-year 2000 dates is key to ensuring that all affected date references were correctly repaired. Note that thorough regression testing may be

difficult to accomplish if system documentation is incomplete or not available.

Issue with Parallel Testing

Parallel testing cannot be the sole form of overall testing for the Year 2000 problem. Why? Because the old system cannot process the same data as a system corrected for the Year 2000 problem. By definition, parallel testing is running both the old system and the changed system with the same data and comparing output. However, since the **old** system cannot handle the same values for date fields or system date—dates in the year 2000—parallel testing cannot be used to test the **changed** system's ability to function using dates in the year 2000. Parallel testing does, however, have its place in the Year 2000 process as a useful form of regression testing—ensuring that no new problems were introduced into the repaired system.

8.3 DEVELOP TEST PLANS

Written test plans are important for each of the types of testing to be conducted: unit, integration, system, acceptance, and regression testing. Include the following information in application test plans:

- Year 2000-specific test requirements.
- Year 2000 test scenarios and test scripts.
- Test approach.
- Test results.

The next sections provide guidance on Year 2000-specific test requirements and developing test scenarios.

8.3.1 Identify Year 2000 Test Requirements

Identify the Year 2000-specific requirements that the application software, data, procedures, and hardware will be tested against. These requirements, identified during the system repair process (Section 7.1) form the basis of test scenarios. How stringently each of these requirements is tested may be influenced to some degree by the approach used in resolving Year 2000 problems. For example, EPA systems must exchange date data using the four-digit year format specified by the *EPA Date Standard* but may

use a solution other than expansion to four-digit years as the solution to internal date processing problems. Therefore, internal date processing must be carefully verified. The obvious requirement is that original system functionality is not compromised by system changes.

8.3.2 Create Year 2000 Test Scenarios

Create test scenarios based on the test requirements discussed above and define the expected results. IBM's *The Year 2000 and 2-Digit Dates* provides a basic list of Year 2000 system test scenarios, including tests for the following special dates:

Tests for Special Dates

- **1900/02/29**. The date should not work because it is not a leap year.
- **1996/02/29** and **2000/02/29**. Both dates should work because they are leap years.
- **1999/12/31** and **2000/01/01**. The dates should work as any regular date and not signify any special processing condition.

Generate test data or copy production data for use in testing. Take care when using data containing date fields changed from two-digits to four-digits. The repaired data must be throughly reviewed before use with the revised application, or it will become difficult to identify errors due to application code or errors caused by bad data.

While this section provides some insights on developing Year 2000-specific test plans, each test plan must be unique to the system. Not all systems use dates in the same manner. Each test plan must focus on testing and validating the date-related system components.

8.4 TEST AND DOCUMENT THE TEST RESULTS

Testing should be conducted with the system dates set prior to, at, and past the year 2000. Use test data that span 100 years. Verify that the date operations identified during the inventory and assessment process return the correct results. Verify that the original system functionality has not been damaged by date repairs.

- **Application testing.** For applications in which the date fields were expanded to four-digit years, testing may not need to be as detailed. In these cases, determine whether any application program logic was also changed. If so, ensure that the test scenarios test the logic, as well as the changes to date fields.

- **Data testing.** Testing expanded date fields in data files (flat files) may be accomplished by simply comparing the old files with the reformatted files. Test the applications using the files to verify field expansion. Databases must be tested both in terms of expanded date fields, if the date fields were expanded, and in terms of the database application (i.e., Oracle, etc.) used to access the database. The database application must be tested following standard application testing procedures to ensure that it stores and retrieves dates appropriately.

The final important step is to document the results of the testing: what was tested and problems identified. Test results indicating errors or problems must be tracked through the process of repairing and retesting. Documented test results may prove useful in implementation and post-implementation as any future problems are identified and repaired. Without the documentation, it will be more difficult to retrace the steps.

8.5 ADDITIONAL CONSIDERATIONS

In conducting system testing, situations may arise that will need to be taken into account to ensure systems are ready on time. One key issue for Year 2000 testing is whether the system can actually be set to simulate the year 2000. If the system cannot simulate the year, ensure that the contingency plan addresses any potential for system failure.

Endnotes

1. National Institutes of Standards and Technology (NIST), Computer Security Resource Center. "Millennium Rollover: The Year 2000 Problem." *CSL Bulletin 96-03*. March 1996.

2. Bruce Ragland, *The Year 2000 Problem Solver, A Five Step Disaster Prevention Plan*, McGraw-Hill, (New York, 1997).

9.0 - IMPLEMENTATION

Now that applications and data have been repaired, they can be moved into an operational environment. However, this may not be the last you see of the Year 2000 problem. Given the deadlines under which systems had to be repaired, it is very likely that some occurrences of the problem will still show up.

> **Q:** Now that you've spent xxx dollars, on this project, what do you have now that you didn't have before?
>
> **A:** Systems that still work.

You Have Won the Battle, Don't Lose the War

Be prepared—contingency planning for this stage is critical. Prior to phasing out the old system, plans must be made for all possible contingencies: unidentified date problems may show up, interfacing systems may have problems that affect system operations, and expanded data may be contaminated by date data containing two-digit years.

- Follow standard lifecycle procedures for implementing the repaired system.
- Running the repaired system in parallel with the old system may help identify any errors introduced as part of the repair process. Keep in mind that running in parallel cannot test the repaired system's ability to function after the system reaches its event horizon.
- Configuration management will be especially important because so many repaired applications will be in testing and implementation during the same periods of time.
- Review the system in its production environment to identify any performance issues. This will be an on-going process, since the system may not be at full capacity until later.

Remember Lessons Learned from the Software Crisis of 2000

Keep in mind the lessons learned from this software crisis. Keep inventories up-to-date, update system documentation, and keep source code listings for all executing applications. The Year 2000 problem has been fixed for your system, but what about all those systems that used workarounds just to get by?

BIBLIOGRAPHY

Adhikari, Richard. "Approaching 2000." *Information Week*, 7 Oct. 1996: 36-48.

Avatar2000(tm) Impact Assessment. Online. Internet. November 1996. Available: http://www.avatars.com/

Baum, David. "Tool Up for 2000." *Datamation* 42, no. 1 (1996). Online. Internet. October 1996. Available: http://www.datamation.com/

Building a Y2K Conversion Cost Estimate. Background Documents from the EPA Year 2000 Workgroup. Not dated.

Celko, Joe. "A Battle Plan for 2000." *Datamation* 42, no. 1 (1996). Online. Internet. October 1996. Available: http://www.datamation.com/

Committee for the Year 2000 Software Conversion. The Information Technology Association of America. *What Are You Waiting For? Start Preparing for Your Year 2000 Software Conversion Today. A Buyer's Guide*. Virginia: The Information Technology Association of America, March 1996.

de Jager, Peter. *Biting the Silver Bullet*. Online. Internet. December 31, 1996. Available: http://www.year2000.com/y2karchive.html

de Jager, Peter. *Systemic Triage*. Online. Internet. January 1997. Available: http://www.year2000.com/y2karchive.html

Eldridge, Andrew, CBSI; and Louton, Bob, GTE. *A comparison of Procedural and Data Change Options for Century Compliance."* Online. Internet. December 31, 1996. Available: http://www.year2000.com/y2karchive.html

Jones, Capers. Chairman, Software Productivity Research, Inc. *The Global Economic Impact of the Year 2000 Software Problem*. White Paper. Version 4. September 23, 1996. Online. Internet. October 1996. Available: http://www.spr.com/

Keuffel, Warren. "Coping with the Year 2000 Rollover." *Software Development* 4, no. 8 (1996): 23-28.

Memorandum for Deputy Heads of Departments and Agencies. Online. Internet. November 1996. Available: http://www.itpolicy.gsa.gov/mks/yr2000/

Memorandum for Senior Information Resource Management, Officials and Chief Information Officers. Online. Internet. November 1996. Available: h://www.itpolicy.gsa.gov/mks/yr2000/

Microsoft Lends Advice to Congress on Year 2000 Issue. Online. Internet. November 1996. Available: http://www.microsoft.com/cio/year2000final.htm

Mitre. *Y2K Compliance Strategies.* Online. Internet. November 1996. Available: http://www.mitre.org/research/y2k

Mitre. *Cost Estimation for Year 2000 Efforts.* Online. Internet. November 1996. Available: http://www.mitre.org/research/y2k

Olsen, Florence. "Beat the Clock." *Government Computer News.* October 21, 1996. Online. Internet. November 1996. Available: http://www.gcn.com/gcnnews/102196/

Olsen, Florence. "Govt. urged to perform triage in two-digit date field repairs." *Government Computer News.* March 18, 1996. Online. Internet. November 1996. Available: http://www.gcn.com/gcnnews/031896/

Olsen, Florence. "Microsoft 2000 plan already nonstandard, mm/dd/yyyy won't work with other fixes." *Government Computer News.* July 15, 1996. Online. Internet. November 1996. Available: http://www.gcn.com/gcnnews/071596/

Pesachowitz, Alvin M. Acting Chief Information Officer, U.S. Environmental Protection Agency. *Guidance on Preparing FY-1998 Budget Estimate for the Year 2000 Date Problem.* Washington: U.S. Environmental Protection Agency, Summer 1996.

———. Acting Assistant Administrator and Chief Information Officer, U.S. Environmental Protection Agency. Letter to Mr. John A. Koskinen, Deputy Director for Management, Executive Office of the President, Office of Management and Budget. Washington: U.S. Environmental Protection Agency, May 16, 1996.

———. Acting Assistant Administrator and Chief Information Officer, U.S. Environmental Protection Agency. Letter with Enclosure, *Response to Information Request Regarding EPA's Ability to Process Year 2000 Data*, to the Honorable Stephen Horn, Chairman, Subcommittee on Government Management, Information and Technology; House of Representatives. Washington: U.S. Environmental Protection Agency, May 30, 1996.

———. Director, EPA Office of Information Resources Management, Office of Administration and Resources Management, U.S. Environmental Protection Agency. *Year 2000 Date Change.* Internal Memorandum to EPA Senior Information Resource Officials, System Managers, and Regional Information Resources Management Chiefs. Washington: U.S. Environmental Protection Agency, May 22, 1995.

U.S. Congress. House. Committee on Government Reform and Oversight. Subcommittee on Government Management, Information and Technology. Letter from the Honorable Stephen Horn, Chairman, Subcommittee on Government Management, Information and Technology to the Honorable Carol M. Browner, Administrator, U.S. Environmental Protection Agency. Washington, D.C., April 29, 1996.

———. ———. ———. *U.S. Federal Government Year 2000 Survey.* Statement of the Honorable Stephen Horn, Chairman, Subcommittee on Government Management, Information and Technology. Washington, D.C., July 30, 1996.

U.S. Environmental Protection Agency. *EPA Data Standard For Representation of Calendar Date.* Washington, D.C., September 20, 1996.

———. *EPA Data Standard For Representation of Calendar Date.* Washington, D.C., Not dated.

———. *EPA Integrity Act Report, FY 1996, Century Change Date Processing.* Washington, D.C., March 14, 1996.

———. *EPA Y2K Work Plan. Draft.* Background Document from the EPA Year 2000 Workgroup. Washington, D.C., August 12, 1996.

———. *EPA Year 2000 Background Documents.* List of EPA Year 2000 Background Documents. Washington, D.C., Not dated.

———. Executive Steering Committee for Information Resources Management. *Agency Year 2000 Management Strategy and Budget Plans. Decision Support Paper.* Washington, D.C., Not dated.

———. *Information Resources Management Policy Manual. Chapter 4. Software Management.* Washington, D.C., November 2, 1995.

———. *Information Resources Management Policy Manual. Chapter 17. System Lifecycle Management.* Washington, D.C., November 2, 1995.

———. Internal Memorandum. *Comments on FIMFIA Draft*. Background Document from the EPA Year 2000 Workgroup. Washington, D.C., September 13, 1996.

———. Internal Memorandum. *Report of Review E1NMB5-15-3038-6400036, Major Information Systems are Vulnerable to Failure Due to the Upcoming Century Change.* Washington, D.C., September 13, 1996.

———. Internal Memorandum. *Response to Final Report of Review - Major Information Systems are Vulnerable to Failure Due to the Upcoming Century Change.* Washington, D.C., June 4, 1996.

———. Office of Inspector General. *Report of Review: Major EPA Information Systems are Vulnerable to Failure Due to the Upcoming Century Change. Audit Report E1NMB5-15-3038-6400036.* Washington, D.C., March 14, 1996.

———. Procurement Policy Notice Number 96-XX. *Statement of Work Language for Turn-of-the-Century Computer Compliance*. Background Document from the EPA Year 2000 Workgroup. Washington, D.C., Not dated.

———. *Promoting Year 2000 Awareness with EPA's External Partners*. Background Document from the EPA Year 2000 Workgroup. Washington, D.C., 1996.

———. *System Design and Development Guidance*. Washington, D.C., June 1989.

U.S. National Institutes of Standards and Technology, Computer Security Resource Center. "Millennium Rollover: The Year 2000 Problem." *CSL Bulletin 96-03*. March 1996. Online. Internet. Available: http://csrc.nist.gov/nistbul/

———. ———. "An Introduction to Computer Security: The NIST Handbook." *NIST Special Publication 800-12*. February 1996. Online. Internet. Available: http://csrc.nist.gov/nistpubs/

U.S. Office of Management and Budget. *Getting Federal Computers Ready for 2000, Report of the U.S. Office of Management and Budget*. Washington, D.C., February 6, 1997.

Widdowson, Gerald. Year 2000 Coordinator, EPA System Support Branch. EPA Memorandum to Ed Springer, Year 2000 Issues Coordinator, Office of Management and Budget. *EPA Year 2000 Background Documents*. Washington: U.S. Environmental Protection Agency, September 27, 1996.

Wohlleben, Paul. Acting Director, EPA Office of Information Resources Management, Office of Administration and Resources Management, U.S. Environmental Protection Agency. *Year 2000 Date Change: Call to Action.* Internal Memorandum to EPA Senior Resource Officials. Washington: U.S. Environmental Protection Agency, May 31, 1996.

The Year 2000 and 2-Digit-Dates: A Guide for Planning and Implementation. IBM Corporation. Fifth Edition. September 1996. Online. Internet. October 1996. Available: http://www.software.ibm.com/year2000

Year 2000 Cost Estimating. May 22, 1996. Online. Internet. November 1996. Available: http://cfcse.ncr.disa.mil/

The Year 2000 Information Center. Online. Internet. October 1996. Available: http://www.year2000.com/

The Year 2000 Issue: Is Your Enterprise Ready? Online. Internet. November 1996. Available: http://www.microst.com/com/cio/year.htm.

The Year 2000 Problem. A Social Security Administration (SSA) white paper presented to FIRMPOC members in July 1995. Online. Internet. November 1996. Available: http://www.itpolicygsa.gov/

The Year 2000 Problem, From A CFO's Perspective. Online. Internet. November 1996. Available: http://www.viasoft.com/

Year 2000 Task Group. Systems Integration Division. The Information Technology Association of America. *The Year 2000 Software Conversion: Issues and Observations.* A White Paper. Virginia: The Information Technology Association of America., January 1996.

APPENDIX A

LEVEL 1 AND LEVEL 2 SYSTEMS

February 14, 1997

LEVEL 1 AND LEVEL 2 SYSTEMS

Office	System Level/Name	Contact	Phone
AA-OARM	1 ARCS CONTRACT TRACKING SYSTEM	VAUGHAN, DORTHEA	(202) 260-9033
AA-OARM	1 ASBESTOS RECEIVABLE TRACKING SYSTEM	NORLAND, SHELL	(702) 798-2499
AA-OARM	1 COMBINED PAYROLL REDISTRIBUTION AND REPORTI	CLUCK, ROBERT	(202) 260-5107
AA-OARM	1 EPA PAYROLL SYSTEM	CLUCK, ROBERT	(202) 260-5107
AA-OARM	1 FACILITIES INDEX SYSTEM	BERLINGERI, DAISY	(703) 235-5576
AA-OARM	1 INTEGRATED CONTRACTS MANAGEMENT SYSTEM	KRANDA, JAMES L	(202) 260-8289
AA-OARM	1 INTEGRATED FINANCIAL MANAGEMENT SYSTEM	CLUCK, ROBERT	(202) 260-5107
AA-OARM	1 MANAGEMENT ACCOUNTING REPORTING SYSTEM	CLUCK, ROBERT	(202) 260-5107
AA-OARM	1 MANAGEMENT AUDIT TRACKING SYSTEM	PADGETT, MARGO F	(202) 260-1201
AA-OARM	1 NATIONAL LOCATOR	RODGERS, DWIGHT F	(202) 260-2082
AA-OARM	1 OFFICE OF ENVIRONMENTAL JUSTICE BULLETIN BOARD SYSTEM	TRIPLIN, JANICE R	(202) 260-6357
AA-OARM	1 PERSONAL PROPERTY ACCOUNTABILITY SYSTEM	DELLAPENTA, DAN	(202) 260-1523
AA-OARM	1 RESOURCES MANAGEMENT INFORMATION SYSTEM	BOONE, WILLIAM J	(202) 260-3367
AA-OARM	1 TIME & ATTENDANCE PERSONNEL & PAYROLL	CLUCK, ROBERT	(202) 260-5107
AA-OARM	1 TIME SHARING SERVICES MANAGEMENT SYSTEM ONLINE	WATSON, ERNEST C	(919) 541-2143
AA-OARM	2 ADP BUDGET PLANNING SYSTEM	CARPENTIER, MICHAEL A	(202) 260-2415
AA-OARM	2 CONTRACT PAYMENT SYSTEM	GRAY, ROBERT (MITCH) E	(919) 541-3016
AA-OARM	2 CONTRACTS INFORMATION SYSTEM	DOCKERY, KATHY R	(202) 260-5605
AA-OARM	2 ENVIRONMENTAL FINANCING INFORMATION NETWORK	HOTLINE	(202) 260-0420
AA-OARM	2 EPA INFORMATION SYSTEMS INVENTORY	ISI MANAGER,	(202) 260-2381
AA-OARM	2 ENVIRONMENTAL SPATIAL DATA LIBRARY SYSTEM	PARTINGTON, EDWARD	(703) 235-5595
AA-OARM	2 GRANTS INFORMATION & CONTROL SYSTEM	CODY, MARIAN	(202) 260-9273
AA-OARM	2 INFORMATION MAIL MANAGEMENT SYSTEM	MARTIN, MARGARET L	(202) 260-4605
AA-OARM	2 INFOTERRA INTERNATIONAL DIRECTORY OF SOURCES	MCNAMARA, EMMA	(202) 260-1522
AA-OARM	2 INTERNATIONAL REGISTER OF POTENTIALLY TOXIC	MCNAMARA, EMMA T	(202) 260-1522
AA-OARM	2 IRM BUDGET SYSTEM	CARPENTIER, MICHAEL A	(202) 260-2415
AA-OARM	2 MASTER INVENTORY SYSTEM	MCFARLAND, SHANNON	(513) 569-7762
AA-OARM	2 SMALL PURCHASE ELECTRONIC DATA INTERCHANGE	CLINE, DAVID M	(202) 260-1677
AA-OARM	2 SUPERFUND COST RECOVERY IMAGE PROCESSING SYSTEM	YOUNG, CHARLES	(202) 260-6890
AA-OARM	2 TRANSIT SUBSIDY SYSTEM	MARTIN, MARGARET L	(202) 260-4605
AA-OAR	1 ACID RAIN DATA SYSTEM -- ALLOWANCE TRACKING SYSTEM	SALPETER, ALEX	(202) 233-9157
AA-OAR	1 ACID RAIN DATA SYSTEM -- EMISSIONS TRACKING SYSTEM	MORITZ, LARRY	(202) 233-9144
AA-OAR	1 AEROMETRIC INFORMATION RETRIEVAL SYSTEM	AMBROSE, VIRGINIA	(919) 541-5454
AA-OAR	1 GRIDDED MODEL INFORMATION SUPPORT SYSTEM	BALDRIDGE, ELLEN	(919) 541-5684
AA-OAR	1 TECHNOLOGY TRANSFER NETWORK	ROREX, HERSCHEL W	(919) 541-5637
AA-OAR	2 AN OPERATION SYSTEM FOR PREDICTING MAXIMUM	HUNG, CHENG-YENG	(202) 233-9204
AA-OAR	2 AN OPERATION SYSTEM FOR PREDICTING THE POPULATION	HUNG, CHENG-YENG	(202) 233-9204
AA-OAR	2 CERTIFICATION FUEL ECONOMY INFORMATION SYSTEM	PARSONS, RICHARD	(313) 668-4324
AA-OAR	2 CONTROL TECHNOLOGY CENTER	BLASZCZAK, ROBERT J	(919) 541-5432
AA-OAR	2 KINETICS MODEL AND OZONE ISOPLETH PLOTTING PACKAGE	BALDRIDGE, ELLEN	(919) 541-5684
AA-OAR	2 MANAGEMENT AND ACCOUNTABILITY PROCESS SYSTEM	STEIGERWALD, JOE	(919) 541-2736
AA-OAR	2 PERSONAL COMPUTER CONTINUOUS EMISSIONS MONITORING	ANTELL, MARK	(202) 564-5003
AA-OAR	2 RACT/BACT/LAER CLEARINGHOUSE OR RBLC	STEIGERWALD, JOSEPH E	(919) 541-2736
AA-OAR	2 URBAN AIRSHED MODEL	BALDRIDGE, ELLEN	(919) 541-5684
AA-OECA	1 ENFORCEMENT CASE SUPPORT EXPERT RESOURCES	LAMBER, KURT	(202) 564-4009
AA-OECA	1 PERMIT COMPLIANCE SYSTEM	MUNDELL, MICHAEL	(202) 564-5031
AA-OECA	2 CONSENT DECREE TRACKING SYSTEM	MILLER, MERLE J	(202) 564-4114
AA-OECA	2 ENFORCEMENT DOCKET SYSTEM	MILLER, MERLE J	(202) 564-4114
AA-OECA	2 ENFORCEMENT DOCUMENT RETRIEVAL SYSTEM	MILLER, MERLE J	(202) 564-4114
AA-OECA	2 INTEGRATED DATA FOR ENFORCEMENT ANALYSIS	ROTHROCK, BRUCE	(202) 260-2504

Legend:
OARM - Office of Management and Resource Management
OECA - Enforcement and Compliance Assurance
OPPTS - Prevention, Pesticides and Toxic
OSWER - Solid Waste and Emergency Response
OAR - Air and Radiation
OPPE - Policy, Planning, and Evaluation
ORD - Research and Development
OW - Water

February 14, 1997

LEVEL 1 AND LEVEL 2 SYSTEMS

Office	System Level/Name	Contact	Phone
AA-OPPE	1 REGISTER OF LISTS	DALEY, JAMES M	(202) 260-2743
AA-OPPTS	1 TOXIC CHEMICAL RELEASE INVENTORY SYSTEM	BOYD, RUBY N	(202) 260-8387
AA-OPPTS	2 CHEMICALS IN COMMERCE INFORMATION SYSTEM	PETTY, EYVONE	(202) 260-1444
AA-OPPTS	2 NATIONAL PESTICIDE INFORMATION RETRIEVAL SYSTEM	WALTERS, VIRGINIA	(317) 494-6614
AA-OPPTS	2 PERSONAL COMPUTER GRAPHICAL EXPOSURE MODELING	DELPIRE, LYNN	(202) 260-3928
AA-OPPTS	2 PESTICIDE PRODUCT INFORMATION SYSTEM	BEECH, JAMES L	(703) 305-5439
AA-OPPTS	2 PROBABILISTIC DILUTION MODEL	ABEL, SID	(202) 260-3920
AA-OPPTS	2 TOXIC SUBSTANCES CONTROL ACT TEST SUBMISSIONS	NOWAK, GERALDINE D	(202) 260-2320
AA-ORD	2 A NATIONAL COMPENDIUM OF FRESHWATER FISH AND WATER	EATON, JOHN	(218) 720-5557
AA-ORD	2 AQUATIC TOXICITY INFORMATION RETRIEVAL	RUSSOM, CHRISTINE L	(218) 720-5709
AA-ORD	2 ASSESSMENT TOOLS FOR THE EVALUATION OF RISK	RUSSOM, CHRISTINE L	(218) 720-5709
AA-ORD	2 ECOTOXICOLOGY DATABASE RETRIEVAL SYSTEM	RUSSOM, CRISTINE L	(218) 720-5709
AA-ORD	2 EXPOSURE ANALYSIS MODELING SYSTEM	MODEL COORDINATOR	(706) 546-3549
AA-ORD	2 HYDROLOGIC EVALUATION OF LANDFILL PERFORMANCE	LANDRETH, ROBERT	(513) 569-7871
AA-ORD	2 INTEGRATED RISK INFORMATION SYSTEM	IRIS RISK HOTLINE,	(513) 569-7254
AA-ORD	2 LAKE ANALYSIS MANAGEMENT SYSTEM	KREIS, RUSSELL G	(313) 692-7615
AA-ORD	2 QUANTITATIVE STRUCTURE ACTIVITY RELATIONSHIPS	RUSSOM, CHRIS	(218) 720-5709
AA-ORD	2 STREAM QUALITY MODEL	MODEL COORDINATOR	(706)
AA-ORD	2 TREATABILITY DATA BASE	SHAUL, GLENN M	(513) 569-7408
AA-ORD	2 TWO DIMENSIONAL CONTAMINANT TRANSPORT UNDER THE	BURDEN, DAVID S	(405) 436-8606
AA-OWSER	1 BIENNIAL REPORTING SYSTEM	WATSON, STEPHEN P	(703) 308-7914
AA-OWSER	1 LIST OF CHEMICALS SUBJECT TO REPORTING UNDER EPC	TRK, KATHLEEN	(202) 260-8353
AA-OWSER	1 RESOURCE CONSERVATION AND RECOVERY INFORMATION SYSTEM	WATSON, STEPHEN P	(703) 308-7914
AA_OSWER	2 COMPREHENSIVE ENV. RESPONSE AND LIABILITY INFORMATION SYS		
AA-OWSER	2 COMPUTER AIDED DATA REVIEW AND EVALUATION	ENG, DAVID S	(703) 603-8827
AA-OWSER	2 VENDOR FIELD ANALYTICAL AND CHARACTERIZATION	MA, CARL	(703) 308-8805
AA-OWSER	2 VENDOR INFORMATION SYSTEM FOR INNOVATIVE TREATMENT	MA, CARL	(703) 308-8805
AA-OW	1 NEEDS SURVEY	FITCH, LEONARD	(202) 260-5858
AA-OW	1 OFFICE OF WATER CLEARINGHOUSE	MABBITT, MORRIS	(202) 260-3963
AA-OW	1 STORAGE AND RETRIEVAL OF WATER QUALITY INFORMATION	HOELMAN, LOUIS H	(800) 424-9067
AA-OW	2 GRANTS INFORMATION AND CONTROL SYSTEM	LATTA, JANNIE	(202) 260-5831
AA-OW	2 NATIONAL SEWAGE SLUDGE SURVEY	WHITE, CHUCK	(202) 260-5411
AA-OW	2 OCEAN DATA EVALUATION SYSTEM	HOELMAN, LOUIS H	(800) 424-9067
AA-OW	2 SAFE DRINKING WATER INFORMATION SYSTEM	DORSEY, TOWANA	(202) 260-2805
AA-OW	2 STATE REVOLVING FUND AWARD LIST	FARBER, KIT	(202) 260-3973
AA-OW	2 STATE REVOLVING FUND INFORMATION DATA BASE	FARBER, KIT	(202) 260-3973
OGC	1 BID PROTEST TRACKING SYSTEM	DEGRANDCHAMP, T.	(202) 260-7547
Regional System	2 RESEARCH LIBRARY FOR RCRA DATABASE	FRIEDMAN, FRED T	(617) 573-9687

Legend:
OARM - Office of Management and Resource Management
OECA - Enforcement and Compliance Assurance
OPPTS - Prevention, Pesticides and Toxic
OSWER - Solid Waste and Emergency Response
OAR - Air and Radiation
OPPE - Policy, Planning, and Evaluation
ORD - Research and Development
OW - Water

APPENDIX B

EPA DATE STANDARD (PENDING)

EPA DATA STANDARD FOR REPRESENTATION OF CALENDAR DATE

1. PURPOSE.

The purposes of this standard are to provide for the accurate use of dates during systems operations and to provide a consistent numeric representation of calendar date to facilitate interchange of date data among information systems.

2. SCOPE AND APPLICABILITY.

This standard is applicable wherever dates are included in an Agency information system and wherever data containing dates is interchanged among information systems. The scope of this standard for date includes the processing of date and date-dependent data, including but not limited to calculating, comparing, and sequencing. All software used for data entry, storage, computation, and retrieval of date in Agency systems must provide the intelligence to process date in compliance with this standard.

This standard does not assign any particular meaning or interpretation to any data element (e.g., a fiscal year) that uses date representations in accordance with this standard. Such meaning will be determined by the context of the application.

This standard applies to all new data processing applications, data management systems, and databases developed by the Agency or its agents, after the date of approval of this standard. The standard also applies to all existing Agency software applications where such applications perform computation of dates before and beyond December 31, 1999, when such operations would otherwise yield incorrect results because of the lack of a date format that includes Century year characters.

This standard is applicable to Agency contracts for acquisition of commercial off-the-shelf (COTS) software and development of custom software. Where relevant, Agency contracts will be written to require vendors to warrant merchantable performance by their product in processing of date and date-dependent data across the Year 2000 boundary.

This standard also is to be utilized by writers of Agency regulations and by persons commenting on, or developing proposed legislation that will result in information gathering, processing, or dissemination by EPA.

This date standard is *not* intended to prescribe the format for date on printed reports, correspondence, or other appearance of date when presented to users.

3. REFERENCES.

The following references were used as a basis for preparation of this standard:

a. American National Standard Representation for Calendar Date and Ordinal Date

for Information Interchange, **X3.30-1985** (R1991), American National Standards Institute (ANSI).

b. Federal Information Processing Standards Publications **(FIPS PUB) 4-1**, Representation of Calendar Date and Ordinal Date for Information Interchange, 1988, January 27. This publication adopts ANSI X.30-1985 (R1991). Change number one of March 25, 1996, highly recommends that four-digit year time elements be used, and that two-digit year time elements should not be used for the purposes of any data interchange among U.S. Government agencies.

c. **ISO 8601**, Data elements and interchange formats -- Information interchange -- Representation of dates and times. International Organization for Standards, First Edition, 1988, June 15.

4. BACKGROUND

Many data collections represent date with two digits for "year" in a "date" field, assuming that all dates pertain to the 20th century. Whenever a computer system or software application uses two digits to represent the year, a change of date from 1999 to 2000 and beyond may produce inaccurate date computations. A two-digit representation of date is not adequate to meet the needs for most hardware operating systems and data processing applications into the next century and beyond. This standard will help to facilitate exchange of information between systems and to mitigate problems created by the turn of the Century, which occurs at the start of the Year 2000.

EPA has not yet adopted an Agency-wide standard for representation of date for data processing or for data interchange. Established standards organizations (e.g., the American National Standards Organization [ANSI], the International Standards Organization [ISO] and the National Institute for Standards and Technology [NIST] provide accepted standards for date representation to meet the needs of government and industry to manage and communicate data. International standards specify the representation of dates in the Gregorian calendar, including both calendar and ordinal date formats. Calendar date is a representation composed of the time elements: year, month of year, and day of month. Ordinal date is a representation composed of the time elements: year and day of year.

Many Agency information systems use commercial database management systems to store and manage data. These products employ specific algorithms to store and manipulate date information. There is no need to establish standards for the way dates are physically stored and managed *within* such operating environments. There ***is***, however, a definite need for the Agency to adopt a standard for representation of date in data files to avoid errors during data interchange, particularly where one application directly accesses data files created by another application which may have been stored in a non-compliant format.

There also is a need to ensure that procurements of commercial off-the-shelf (COTS) software and custom-designed software products require suppliers to verify their products will be capable of importing, exporting and computing dates using a four-digit year representation that complies with this standard.

5. DEFINITIONS.

a. Date, calendar: A particular day of a Gregorian calendar year, identified by a single eight-digit numeric data element composed of the ordinal number for its calendar day, the ordinal number for its calendar month, and the ordinal number for its calendar year.

b. Day, calendar: A particular day within a month of a Gregorian calendar year, identified by a two-digit number with a leading zero where the number representing the day has only one digit. The first day of the month is represented by the ordinal number "01," and subsequent days are numbered in ascending sequence from "02" to the end of the month.

c. Gregorian calendar: A calendar in general use introduced in 1582 to correct an error in the Julian calendar. In the Gregorian calendar, common years have 365 days and leap years 366 days, divided into 12 sequential months.

d. Month, calendar: A period of time resulting from the division of a calendar year into 12 sequential periods of time, each with a specific name and containing a specified number of days. Month of the year is represented by two digits, with a leading zero where the number representing the month has only one digit. January is represented by the ordinal number "01," and subsequent months are numbered in ascending sequence from "02" to "12."

e. Year, calendar: A cyclic period of time in a calendar that is required for one revolution of the earth around the sun. In the Gregorian calendar, a calendar year is either a common year or a leap year.

f. Year, common: In the Gregorian calendar, a year that has 365 days.

g. Year, leap: In the Gregorian calendar, a year that has 366 days. A leap year is a year whose number is divisible by four an integral number of times, except that if it is a centennial year, it shall also be divisible by 400 an integral number of times. The year 2000 is a leap year.

6. STANDARD.

EPA's standard for representation of calendar date in Agency information systems requires compliance with the following, for interchange of calendar date data:

a. EPA's standard for representation of calendar date is an eight-digit sequence composed of numeric characters, in the format CCYYMMDD, where:

(1) "CC" represents the current Century
(2) "YY" represents the decade and year within the Century
(3) "MM" represents the calendar month of the year, and
(4) "DD" represents the calendar day of the month

And:

(5) The order of the time elements shall be high order to low order: Century/year; month; day of month (e.g. CCYYMMDD).

(6) No alphabetic characters are used to represent "month".

(7) The numbers that represent month of year and day of the month shall include leading zeros whenever their respective values contain only one digit. For example, the month of May is "05" and the 7th day of the Month is "07".

(8) No separators are used between the time elements (i.e. no slashes, hyphens, or spaces) for the interchange of date. The numbers shall be contiguous.

b. Examples of the standard date format are:

Standard Date Format:	Alphanumeric Format:
19970101	January 1, 1997
19980704	July 4, 1998
20001225	December 25, 2000

c. Agency information management systems must include the intelligence to accurately process date according to the following criteria:

(1) Calculations must execute, using dates with a four-digit year.

(2) Functionality, both online and batch, including but not limited to entry, inquiry, maintenance and update, must support four-digit year interfaces.

(3) Interfaces and reports must support processing of four-digit year.

(4) Successful transition into the year 2000, using the correct system date, must occur without human intervention.

(5) After transition to the year 2000, processing with a four-digit year must

occur without human intervention.

(6) Calculations that determine and process leap years must be accurate.

(7) Correct results in forward and backward data calculations that span Century boundaries must be provided, including the conversion of previous years currently stored as two digits.

(8) Date data that are output for data interchange must be in the format specified above for standard representation of calendar date.

d. EPA's standard for representation of calendar date conforms to the appropriate subset of each of the standards referenced in paragraph three above.

e. The format for date, where it is represented as a character string for data interchange, shall be binary coded character(s) using the American Standard Code for Information Interchange (ASCII).

7. RESPONSIBILITIES.

The **Chief Information Officer** (CIO) of the Agency shall be responsible for issuing waivers for compliance consistent with Par. 9, below.

The **Office of Information Resources Management** (OIRM) shall:

Lead the efforts to develop, implement, and ensure adherence to this data standard.

Lead the efforts to develop a management plan describing steps for implementation of the standard.

Provide guidance and technical assistance in meeting the requirements of this standard.

Oversee resolution of conflicts regarding applicability or other issues relating to this standard.

Senior Information Resources Management Officials **(SIRMOs) and Regional IRM Branch Chiefs** shall be responsible for assuring compliance with this standard within their information management environments.

8. COMPLIANCE DATES -- NEW ACQUISITIONS AND EXISTING APPLICATIONS

This standard applies to the acquisition of all **new** data processing applications,

data management systems and databases developed by the Agency or its agents. Beginning January 1, 1998, all **new** applications software acquired by the Agency will comply with this standard and must be certified by vendors to be merchantable in compliance with Year 2000 requirements

The standard also applies to **existing** Agency applications by the Year 2000 or earlier whenever such operations would otherwise yield incorrect results because of the lack of a date format that includes Century year characters. Wherever an application uses projected dates for accelerated computation of interest, retirement, or other values, more immediate attention to compliance with this standard may be required to avoid errors. It is the responsibility of the program operating the application to make such a determination.

9. PROVISION FOR WAIVER.

A waiver may be granted by the Agency's Chief Information Officer (CIO) for legacy applications that achieve compliance by means other than the use of this standard, or for systems or applications that will be terminated and retired before the Year 2000, or for systems where the costs of implementing the standard are significantly higher than the benefits. The guiding principles for a waiver will involve:

a. Developing an application for waiver to the CIO outlining the reasons why the date data standard should not be implemented in the information collection. The application will contain a complete risk assessment and cost-effectiveness evaluation of continued operation in a non-compliant mode.

b. Obtaining approval by the decision officials in the requesting office, as defined by EPA's System Lifecycle Management Policy and the organization's Senior Information Resources Management Official (SIRMO).

c. Submitting the application to the CIO, who has responsibility for final disposition.

d. The CIO notifying the applying office in writing of the disposition of the waiver.

Appeals may be brought to the ESC if outstanding issues cannot be resolved. The ESC may make recommendations to the CIO for final CIO decision.

APPENDIX C

ADDITIONAL INFORMATION ON YEAR 2000 SOFTWARE TOOLS

Excerpts from the EPA/ITAS Document:

YEAR 2000 (Y2K) SOFTWARE ASSESSMENT REPORT Part 2 - System Manager Y2K Project Plan and Tool Reference Discussion Draft. October 28, 1996

Prepared for the U.S. Environmental Protection Agency
Enterprise Technology Services Division

This appendix lists tools and vendors for illustrative purposes only. Inclusion in this list does not constitute endorsement or certification by EPA.

ADDITIONAL INFORMATION ON YEAR 2000 SOFTWARE TOOLS

Exhibit 1-A
Y2K Service/Methodology Provider

	Service Provider	Methodology/Toolkit Name
1.	Accelr8 Technology Corp.	Navig8 2000
2.	CAP Gemini	Transmillenium Services
3.	Computer Associates	CA Discovery 2000
4.	COMSYS	
5.	Data Dimensions	Millenium Services
6.	Dun & Bradstreet	
7.	Edge Information Group	
8.	FEDSIM	Century Date Change Services
9.	HCL James Martin	TSRM
10.	PARAGON	Odyssey 2000
11.	Platinum Technologies	System Vision 2000
12.	Prince Software Inc.	Portal 2000
13.	TRECOM	APM2000
14.	Viasoft Inc.	Enterprise 2000

[Note: Additional information can be found on the Internet at:

www.itaa.org
www.mitre.org/research/y2k/docs

These sites maintain lists of vendors providing Year 2000 tools and services.]

---

[7] This is not an exhaustive list. Please see Appendix 4 for additional vendors [see note above].

Exhibit 1-B
Inventory/Portfolio Tools and Vendors

Inventory/Portfolio Tools	Vendor
1. CA-Impact 2000	Computer Associates
2. Century File Conversion	Quintic Systems
3. Century Source Conversion	Quintic Systems
4. Challenge 2000-Revolve w/Y2K addon	MicroFocus
5. CHC Signature 2000	Computer Horizons
6. COBOL Analyst 2000	SEEC
7. Conversion Xpert	Compuware
8. EDGE Portfolio Analyzer	EDGE Information Group
9. File-AID	Compuware
10. GILES 2001	Global Software
11. Maintenance Workbench	Intersolv
12. PM/SS	ADPAC
13. SoftAudit/One	Isogon
14. SoftAudit/2000	Isogon
15. Survey 2000	Prince Software
16. System Vision U2	ADPAC
17. System Vision Year 2000	ADPAC
18. Vantage YR2000	Millenium Dynamics
19. VIA/Alliance	Viasoft

Exhibit 2
Impact Analysis Tools and Vendors

	Impact Analysis Tools	Vendor
1.	CA-Impact 2000	Computer Associates
2.	Century File Conversion	Quintic Systems
3.	Century Source Conversion	Quintic Systems
4.	Challenge 2000-Revolve w/Y2K addon	MicroFocus
5.	CHC Signature 2000	Computer Horizons
6.	COBOL Analyst 2000	SEEC
7.	Conversion Xpert	Compuware
8.	Date/2000	Advanced Software Products Group
9.	EDGE Portfolio Analyzer	EDGE Information Group
10.	Estimate 2000	Viasoft
11.	File-AID	Compuware
12.	GILES 2001	Global Software
13.	Maintenance Workbench	Intersolv
14.	Survey 2000	Prince Software
15.	System Vision Year 2000	ADPAC
16.	VIA/Alliance	Viasoft

Exhibit 3
Implementation Tools and Vendors

Implementation Tools	Vendor
1. CA-Impact 2000	Computer Associates
2. CA-Migrate/COBOL	Computer Associates
3. CA Optimize/II	Computer Associates
4. Century File Conversion	Quintic Systems
5. Century Source Conversion	Quintic Systems
6. Challenge 2000-COBOL Workbench	MicroFocus
7. Challenge 2000-MVS Workbench	MicroFocus
8. Challenge 2000-Resolve w/Y2K addon	Micro Focus
9. CHC Signature 2000	Computer Horizons
10. COBOL Analyst 2000	SEEC
11. Conversion Xpert	Compuware
12. Existing System Workbench	Viasoft
13. Fieldex	STA Inc.
14. File-AID	Compuware
15. GILES 2001	Global Software
16. Maintenance Workbench	Intersolv
17. Peritus Software Auto Enhancer/2000	Peritus
18. ReSource	The Source Recovery Company
19. System Vision Year 2000	ADPAC
20. Trans Century Date Logic Generator	Trans Century Data Systems
21. Translate 2000	Prince Software
22. Vantage YR2000	Millenium Dynamics
23. Xpediter/TSO and CICS	Compuware

[page 15]

Exhibit 4-1
Code Complexity Analysis Tools and Vendors

	Code Complexity Analysis Tools	Vendor
1.	Visual Quality Toolset	McCabe Associates
2.	Visual Testing Toolset	McCabe Associates
3.	Visual Reengineering Toolset	McCabe Associates

[page 18]

Exhibit 4-2A
Testing Tool Categories

The following is a list of the category types for Y2K testing tools. Each of the categories has a brief explanation of what the tools in this category do.

1. Execution Monitors - These tools follow the flow of program logic and determine how data flows between programs.

2. Playback and Record - These tools capture key strokes at the beginning of the process for playback late[r] for iterative testing.

3. Date Simulators - These tools simulate dates for the testing of the various Y2K date scenarios that must be tested.

4. File Reformatters - These tools reformat files into a Y2K renovated format.

5. Debuggers - These tools provide automatic debugging of errors. Interactive debuggers are advised.

6. File Comparators - These tools perform file comparisons to ensure that files match preselected comparison criteria.

7. Database Fast Loaders - These tools fastload databases. The number of databases that will have to be tested in the Y2K project will be high. These loaders will save considerable time in this process.

8. Date/Time Wrappers - These tools provide the ability to test internal file date fields.

9. Bridge Builders - These tools provide automated assistance in the building of date bridges from system to system when one system handles the year 2000 differently than another system to which it either passes data or receives data.

Exhibit 4-2B
Testing Tools and Vendors

Date Routines	
Date Routine Tool	**Vendor**
1. Bridge to the Next Century	Edge Information Group
2. TransCentury Calendar Routines	TransCentury Data Systems

System Date Simulators	
Date Simulation Tools	**Vendor**
1. Hourglass 2000	Mainware
2 . Simulate 2000	Prince Software
3. TICTOC	Isogon
4. VIA/ValidDate	Viasoft
5. Xpediter/Xchange	Compuware

[page 20]

Other Testing Tools	
Testing Tool	**Vendor**
1. CA-Datamacs II	Computer Associates
2. CA-Impact 2000	Computer Associates
3. CA-Intertest for Batch	Computer Associates
4. CA-Intertest for CICS	Computer Associates
5. CA-Traps	Computer Associates
6. CA-Verify for CICS	Computer Associates
7. CA Verify for VTAM	Computer Associates
8. Century File Conversion	Quintic Systems
9. Challenge 2000 COBOL Workbench	Micro Focus
10. Challenge 2000- MVS Workbench	Micro Focus
11. Date/2000	Advanced Software Products Group
12. Hiperstation/IMS/DC	Compuware
13. Auto Enhancer/2000	Peritus Software
14. Playback/CICS	Compuware
15. Simulate 2000	Prince Software
16. Year 2000 S390/Workstation	Tech-Beamers
17. Trans Century Date Logic Generator	Trans Century Data Systems
18. Version Merger	Princeton Softech
19. VIA/SmartTest	Viasoft
20. VIA/SmartTest TCA	Viasoft

3.0 Vendor/Tool Selection Guidelines and Cautions

Even in those instances where a specific type of tool is known to be needed, the selection of the most appropriate Y2K tool and vendor is not an easy process in and of itself. This section offers some general guidelines for selecting tools and vendors. This section also provides some cautionary notes concerning vendor claims and potential problem areas associated with these claims.

3.1 Vendor and Tool Selection Guidelines

Before you decide on a vendor or a tool, determine what tools are already in place and where they are located!

3.1.1 Vendor Selection

- Your requirements must drive the vendor selection not those of the vendor.
- Ensure that the following information is provided and is satisfactory for your requirements:
  - Client References
  - Statement of Financial Stability
  - Resumes of Assigned Staff
  - Warranties Offered
  - What Work is Done Onsite/Offsite
  - Specific Project Milestones
  - Formal Project/Milestone Schedules

3.1.2 Tool Selection

3.1.2.1 General Guidelines

There are some general guidelines that can help in the selection of Y2K tools. These criteria are divided into two categories: Environmental and Operational (or Usability)

Environmental

- Identify what platform is being evaluated (e.g. IBM MVS/XA Mainframe)
- Identify what platform the tool will run on. (The mainframe, a Workstation, a Server Platform?)
- Identify tools, if any, that are prerequisites.
- Identify other tools that this tool works with or is related to.
- Identify the languages/DBMS it supports.
- Identify library systems used by the tool.
- Determine what the import and export capabilities of the tool are.

Operational

- Know CPU time required for using the tool.
- Know how much storage (DASD) is required by the tool.
- Review Sample reports for their usefulness.
- How intuitive/user friendly is the software and whether it requires consulting support to use.
- Ask the vendor about the limitations and the latest features of the tool in relation to the Y2K Phase effort in which it will be applied.
- Determine whether the tool will be useful beyond the year 2000.
- Identify the cost of the tool, cost of consulting support, upgrade costs etc.

3.2 Cautionary Notes

Some cautionary notes are offered below to further assist in the selection of Y2K tools and vendors and in addressing the Y2K problem. These notes are derived from of the experiences of organizations that have been addressing the Y2K problem for some time now.

- There are tools that run on PC platforms that don't necessarily get all the data from the mainframe because the tool is not resident on the evaluation platform. For example if library names are greater than 8 characters they will not download to many PC platforms.
- Generic tools may not handle customized uses of a programming language or database. For example, in some organizations all I/O was removed from DB2 or IMS and replaced by an I/O driver. Some organizations use assembler routines to interface with other code.
- Tools that require consulting support are *generally* not as robust.
- There are tools that will just do portfolio/inventory and others that will do inventory and impact analysis.
- Software tools need to be able to find imbedded code. This code has to be examined for date occurrences as well.
- Very few tools have actually been written for Year 2000. Features have been added to existing software tools to address Y2K.
- **Warning**: Claims that tools will recognize a particular language may not be accurate. This needs to be verified by using abnormalities in code structure not straightforward COBOL or PL/1.
- You ***cannot*** decide to ***just*** expand the date field or select only one approach to solving the problem. A combination of approaches is required.
- You will have to be very specific in contracts as to what is considered to be Y2K compliant.
- IBM will not be testing subsystems against older versions. For example COBOL II is not compliant. You can write compliant COBOL II code but that code may not work with the new version of IBM subsystems.

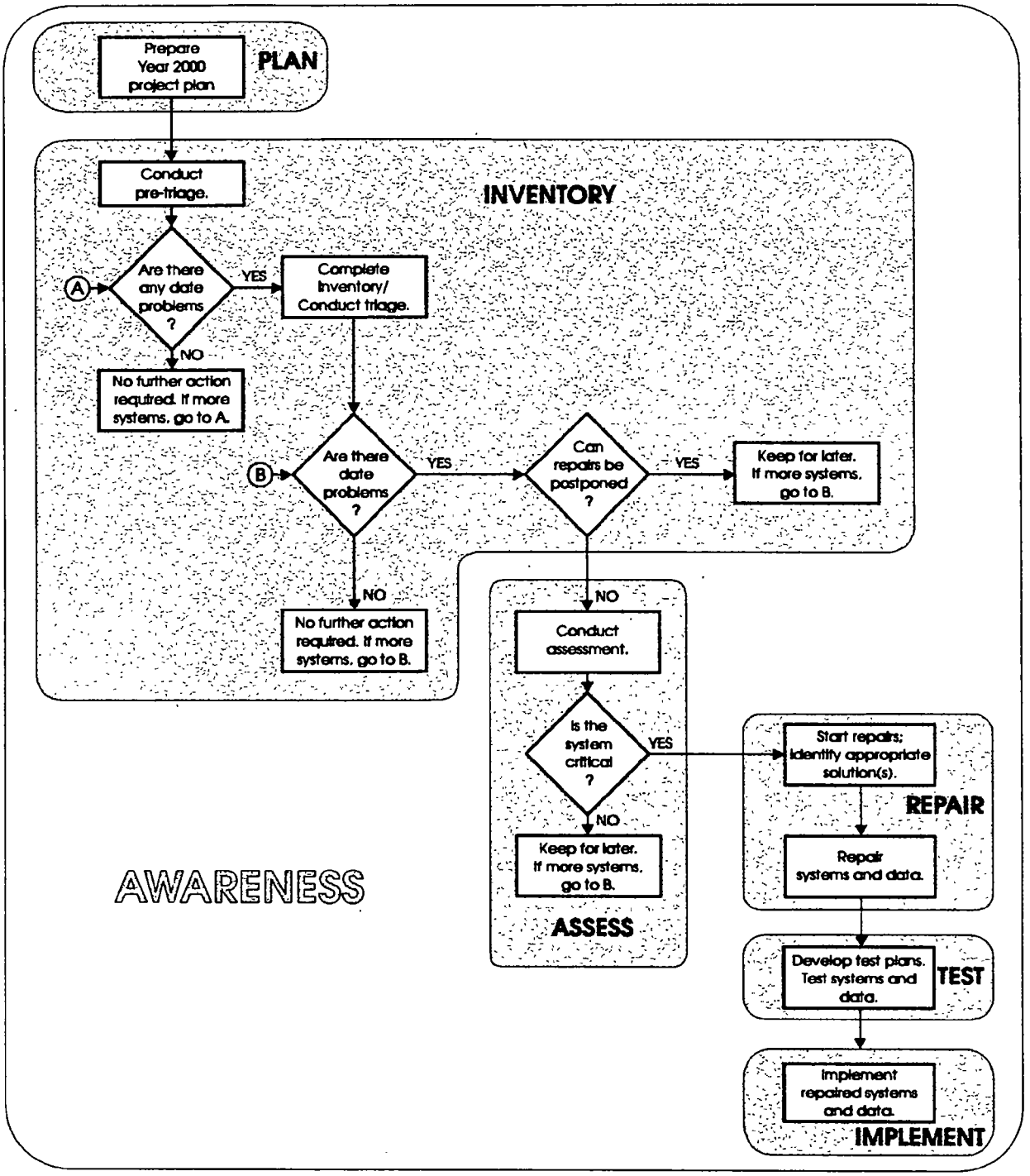
[page 27]

- IBM subsystems ***do not have*** compatible integer dates.
- Some date simulators can bring the system to a halt because the[y] intercept **all** SVC's.
- If you move the date forward to do testing and then decide to go back some software may go bananas because some "event" already happened that shouldn't have. Therefore the testing environment has to be repeatable.
- MIP requirements are a **critical** metric in testing.
- ***No matter how you test or what tools you use you must isolate files, software and DASD.***

APPENDIX D

YEAR 2000 PROCESS FLOWCHART

YEAR 2000 PROCESS FLOWCHART



APPENDIX E

WHERE TO GET HELP

This appendix includes web sites for Year 2000 tools and vendors. These commercial web sites are included for illustrative purposes only; inclusion of these sites does not constitute endorsement or certification by EPA.

WHERE TO GET HELP

This Appendix lists sources of information and guidance currently available on resolving Year 2000 and date-related problems, including Year 2000 Internet web sites, key documents, and EPA, federal, and professional/commercial resources. The web site addresses provided are curent as of February 1997. But please note that Internet addresses are subject to change.

Year 2000 Internet Web Sites
(As of February 1997)

Federal Sites	General Accounting Office (GAO)	**http://www.gao.gov/**
	General Services Administration (GSA)	**http://www.gsa.gov** **http://www.itpolicy.gsa.gov/mks/yr2000/**
	National Institute of Standards and Technology (NIST), Computer Security Resource Center (produces the *CSL Bulletin*).	**http://csrc.nist.gov/nistbul/**
Professional and/or Commercial Sites	Information Technology Association of America (ITAA)	**http://www.itaa.org**
	MITRE Year 2000 home page	**http://www.mitre.org/research/y2k**
	Year 2000 Information Center	**http://www.year2000.com/**
Examples of Web Sites Containing Year 2000 Vendor/Tool Information	Gartner Group	**http://www.gartner.com/** [Search on "year 2000"]
	CIO Council Sub-Committee on Year 2000 & General Services Administration Office of Governmentwide Policy (MKS)	**http://www.itpolicy.gsa.gov/mks/yr2000/y201toc1.htm** *Note: This site includes many links to web sites containing lists of vendors, tools, and Year 2000-compliant COTS.*

Year 2000 Internet Web Sites
(As of February 1997)

Examples of Web Sites Containing Year 2000 Vendor/Tool Information, *cont.*		**http://www.ssa.gov/year2000/y2klist.htm**
	ITAA vendor/tool list	**http://www.itaa.org/**
	MITRE	**http://www.mitre.org/research/y2k**
		http://www.mitre.org/research/cots/COMPLIANCE_CAT.html
		http://www.mitre.org/research/y2k/docs/TOOLS_CAT.html

Key Year 2000 Documents
(As of February 1997)

Professional and Commercial Year 2000 Documents

The Global Economic Impact of the Year 2000 Software Problem. Jones, Capers. Chairman, Software Productivity Research, Inc. Version 4. September 23, 1996. Available on the Internet: http://www.spr.com/

Proposed Criteria for Century Compliance. GTE Technology and Systems, Technology Program Office. June 6, 1996. Available on the Internet: http://www.year2000.com/y2karchive.html

What Are You Waiting For? Start Preparing for Your Year 2000 Software Conversion Today. A Buyer's Guide. Committee for the Year 2000 Software Conversion. The Information Technology Association of America. March 1996.

The Year 2000 and 2-Digit-Dates: A Guide for Planning and Implementation. IBM Corporation. Sixth Edition. December 1996. Available on the Internet: http://www.software.ibm.com/year2000

Key Year 2000 Documents
(As of February 1997)

The Year 2000 Software Conversion: Issues and Observations. Year 2000 Task Group. Systems Integration Division. The Information Technology Association of America. January 1996.

Federal Year 2000 Documents

EPA Data Standard for the Representation of Calendar Date. Pending Date Standard.

EPA Year 2000 Resources

Agency Points of Contact

Source	Information Provided
EPA Year 2000 Workgroup	Standards and guidance.
Enterprise Technology Services Division (ETSD)	Technical information.

Agency Documents

Document	Information Provided
Pending Date Standard, *EPA Data Standard for the Representation of Calendar Date.*	EPA standard for representation of calendar date.
EPA *Information Resources Management Policy Manual*	EPA IRM policies, including software management (Chapter 4) and system lifecycle management (Chapter 17).

EPA Year 2000 Resources

Agency Documents, *cont.*	EPA *Operations and Maintenance Manual* (Draft)	Issued by EPA as the primary guidance for directing system operations and maintenance efforts.
	EPA *System Design and Development Guidance*	EPA standards for software design, development, operation, and maintenance.

Other Year 2000 Resources

Federal Resources	CIO Council Sub-Committee on Year 2000 (formerly known as the Year 2000 Interagency Workgroup). Office of Management and Budget (OMB).
Non-Federal Resources	Industry journals and trade magazines.

Developing Year 2000 Cost Estimates
(As of February 1997)

Professional/ Commercial Information	*Cost Estimation for Year 2000 Efforts.* MITRE. Available on the Internet: **http://www.mitre.org/research/y2k**
Federal Information	*Building a Y2K Conversion Cost Estimate.* Background Documents from the EPA Year 2000 Workgroup. Not dated. *Guidance on Preparing FY-1998 Budget Estimate for the Year 2000 Date Problem.* Memo from Alvin M. Pesachowitz, Acting Chief Information Officer, EPA. Summer 1996.

Developing Year 2000 Cost Estimates
(As of February 1997)

Federal Information, *cont.*

Air Force Year 2000 Web Site
http://infosphere.safb.af.mil/~jwid/fadl/world/asses.htm#estimate

Health Care Financing Administration
http://www.itpolicy.gsa.gov/mks/yr2000/hcfa.htm

Housing and Urban Development
http://www.itpolicy.gsa.gov/mks/yr2000/hudcost.htm

Developing Contingency Plans

Federal Guidance

Federal Information Processing Standards Publications (FIPS PUBS) 31: *Guidelines for ADP Physical Security and Risk Management*, June 1974. Available from:

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
(703) 487-4650

National Institute of Standards and Technology (NIST), Computer Security Resource Center.

- NIST Computer Systems Laboratory (CSL) Bulletin. *Preparing for Contingencies and Disasters*. September 12, 1995. Available on the Internet at **http://csrc.nist.gov/nistbul/**

- NIST Special Publication 800-12: *An Introduction to Computer Security: The NIST Handbook*. February 6, 1996. Available on the Internet at **http://csrc.nist.gov/nistpubs/**

Office of Management and Budget (OMB).

- OMB Circular A-130, *Management of Federal Information Resources*. February 8, 1996.

EPA Guidance

EPA Directive 2195, *Information Security Manual (ISM)*.

APPENDIX F

ACRONYM LIST

ACRONYM LIST

BIOS	Basic Input/Output System
CIO	Chief Information Officer
CPU	Central Processing Unit
COTS	Commercial-off-the-Shelf
EPA	Environmental Protection Agency
ESD	Enterprise Systems Division
ETSD	Enterprise Technology Services Division
GAO	General Accounting Office
GSA	General Services Administration
IRM	Information Resources Management
LOC	Lines of Code
OMB	Office of Management and Budget
SBO	Senior Budget Officer
SIRMO	Senior Information Resources Management Official
WCF	Working Capital Fund
Y2K	Year 2000

APPENDIX G

GLOSSARY

GLOSSARY

Application - "A software program that carries out some useful task. Database managers, spreadsheets, communications packages, graphics programs, and work processors are all applications." (From IRM Policy Manual, Chapter 4 - Software Management.)

Applications System - "Refers to an information system composed of one or more units of software supported by automated data processing equipment (ADPE) and automating the work methods and procedures to collect, store, process and disseminate information to support specific agency missions." (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

Application Software - "Software specifically produced for the functional use of a computer system, e.g., payroll, inventory control, environmental monitoring and scientific modeling." (From IRM Policy Manual, Chapter 4 - Software Management.)

Database Management System (DBMS) - "Computer software used to create, store, retrieve, change, manipulate, sort, format, and print the information in a database." (From IRM Policy Manual, Chapter 4 - Software Management.)

Disk Operating System (PC/MS DOS) - "MS-DOS or PC-DOS, which stands for Microsoft Disk Operating System and Personal Computer Disk Operating System (IBM) respectively, is the software that organizes how a personal computer reads, writes, and reacts with its various input/output devices, including keyboards, screens, disks, serial and parallel ports, printers, modems, etc." (From IRM Policy Manual, Chapter 4 - Software Management.)

External Data Partners - EPA's external data partners include other federal agencies, state and local governments, and the regulated community.

Hardware - "The actual physical computing machinery, as opposed to Software which is the list of instructions to operate the hardware." (From IRM Policy Manual, Chapter 4 - Software Management.)

Information System Category - "Refers to the manner in which systems are classified according to a combination of factors including the system's type, cost, and organizational scope in terms of use and funding. All systems are categorized in one of the following four categories:

(1) Major Agency Systems;
(2) Major AAship/Regional Systems;

(3) Significant Program Office Systems;
(4) Local Office or Individual Use Systems."

(From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

Interface - "The point of meeting between a computer and an external entity, whether an operator, a peripheral device, or a communications medium. An interface may be physical involving a connector, or logical involving software." (From *Webster's New World Dictionary of Computer Terms*, Third Edition, 1988.)

Knowledge-Based Systems - "A class of systems that employ decision rules." (From IRM Policy Manual, Chapter 4 - Software Management.)

Local Office of Individual Use System - Also referred to as a Level 4 system. A Local Office or Individual Use System is an EPA system classified below a Level 3 system. The system cost is greater than $100,000 annually for one project. (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

Major AAship or Regional System - Also referred to as a Level 2 system. A Major AAship or Regional System is an EPA system classified as mission critical for one AAship or regional office. The system cost is greater than $10 million throughout the lifecycle or $1 million annually. (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

Major Agency System - Also referred to as a Level 1 system. An EPA system classified as mission critical for multiple AAships or regions; or an Agency core financial system. The system cost is greater than $25 million throughout the lifecycle or $5 million annually. (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

Operating System - "A software program which manages the basic operations of a computer system. It figures how the computer main memory will be apportioned, how and in what order it will handle tasks assigned to it, how it will manage the flow of information into and out of the main processor, how it will get material to the printer for printing or to the screen for viewing, how it will receive information from the keyboard, etc. In short, the operating system handles the computer's basic housekeeping." (From IRM Policy Manual, Chapter 4 - Software Management.)

Platform - "Hardware architecture of a particular model or computer family. It is the standard to which software developers write their programs. The term may also include the operating system." (From IRM Policy Manual, Chapter 4 - Software Management.)

Significant Program Office System - Also referred to as a Level 3 system. A Significant Program Office System is an EPA system classified as mission critical in a program office. The system cost is greater than $2 million throughout the lifecycle or $100,000 annually. (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

Software - "Computer programs, procedures, rules and possibly associated documentation and data pertaining to the operation of a computer system." (From IRM Policy Manual, Chapter 4 - Software Management.)

Software Tools - "Packaged, often commercial, computer program(s) used to help develop, test, analyze or maintain computer programs, data and information systems. Examples include statistical software such as SAS, SPSS, sort systems, etc." (From IRM Policy Manual, Chapter 4 - Software Management.)

Systems - "Refers to an organized set of functions, data, procedures, hardware, software, communications and/or documentation which enables an organization to solve a specific information management problem. A system need not be automated, but most instances of life cycle management apply to automated systems." (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

System Category, Level 1 - Also referred to as a Major Agency System. A Level 1 system is an EPA system classified as mission critical for multiple AAships or regions; or an Agency core financial system. The system cost is greater than $25 million throughout the lifecycle or $5 million annually. (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

System Category, Level 2 - Also referred to as a Major AAship or Regional System. A Level 2 system is an EPA system classified as mission critical for one AAship or regional office. The system cost is greater than $10 million throughout the lifecycle or $1 million annually. (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

System Category, Level 3 - Also referred to as Significant Program Office System. A Level 3 system is an EPA system classified as mission critical in a program office. The system cost is greater than $2 million throughout the lifecycle or $100,000 annually. (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

System Category, Level 4 - Also referred to as a Local Office or Individual Use System. A Level 4 system is an EPA system classified below a Level 3 system. The system cost is greater than $100,000 annually for one project. (From IRM Policy Manual, Chapter 17 - System Life Cycle Management.)

Triage - In a broad sense, triage means: *the assigning of priority order to projects on the basis of where funds and resources can be best used or are most needed.*[1] However, in a medical sense, triage also has another meaning: *the sorting and allocation of treatment to patients and especially battle and disaster victims according to a system of priorities designed to maximize the number of survivors.*[2]

Endnotes

1. *Merriam-Webster's Collegiate Dictionary*, 10th ed., Merriam-Webster, Inc. (Springfield, MA, 1995).

2. Ibid.