

Year 2000 (Y2K) Compliance Test Plan for [Subject]

Introduction

This document summarizes the minimum set of tests to be run on the [Subject] to determine its current level of Year 2000 (Y2K) compliance. Additional testing and test situations will be determined by an analysis of the [Subject] operational requirements and of its design.

Definition of Y2K Compliance

In order to evaluate any system for Y2K compliance, we require a definition of compliance. That definition is composed of four parts.

Correctly process dates before and after 2000-01-01

For a product to be Y2K compliant, it must correctly process dates before and after 1 January 2000. This requires (to the extent that the [Subject] missions require it) that

From the time the product is declared Y2K compliant until 1 January 2000, it must correctly process dates that are on either side of the millennium point.

After 1 January 2000, the product must continue to correctly process dates on both sides of the millennium point.

The product must correctly process 2000-01-01 itself.

Recognize Year 2000 as leap year

To be Y2K compliant, the product being evaluated must recognize the year 2000 as a leap year. This requires (to the extent that the [Subject] missions require it) that

Recognizing the date 2000-02-29 as valid.

Recognizing the (short) Julian date 00060 (sixty days from the start of the year) as 2000-02-29.

Recognizing the (short) Julian date 00366 as 2000-12-31.

Calculating the number of days between 2000-03-01 and 2000-02-28 as 2.

Accept and display dates unambiguously

To be Y2K compliant, the product being evaluated must receive, display, print, and store dates unambiguously.

Correctly process logic dates that are used for non-date functions

To be Y2K compliant, the product must properly perform those non-date functions that have been implemented using date information. A non-date function is defined as a function performed by a system or component that, by its nature, does not require information to satisfy its requirements. Many products, however, utilize dates and time to assist or control the

execution of non-date functions. Some examples of non-date functions that may use date information are

- Information archiving functions
- Naming conventions
- Passwords
- Random number generation

As the list above indicates, most of these non-date functions are operating system related. Hence, few, if any such problems are anticipated with [Subject] software since the processor operator sessions are assumed to be Y2K compliant. However, the tests will be run.

Critical Midnight Crossings

To properly evaluate the [Subject] system for Year 2000, its behavior prior to, through, and after four critical midnight crossings must be examined:

- **December 31, 1999 to January 1, 2000** — This is the basic millennium transition.
- **February 28, 2000 to February 29, 2000** – This is the first of the leap year midnight transitions.
- **February 29, 2000 to March 1, 2000** — This is the second of the leap year midnight transitions.
- **December 31, 2000 to January 1, 2001** — This is the midnight transition that takes us beyond the Year 2000 situation.

Functional Operation

Initial Conditions

Before any testing, a complete system backup is required. This will include operational software, system and application files, and mission-related data.

This testing activity further assumes that all COTS/GOTS products, including operating systems, are current and are considered Y2K compliant by their vendors.

Operation

Routine operation tests will consist of a set of exercises or [Subject] system operations that require the [Subject] software to perform mission support functions exactly as would occur in normal operation. Each of these test situations will be run for each of the critical midnight crossings described above. As part of the test observations will be made of data presentations and displays as well as mission-related system functioning.

Operation from N minutes prior to M minutes after Midnight

The object of this test case is verification that the [Subject] software can survive the critical midnight transition and to verify that, once the transition has occurred, the remainder of [Subject] date processing. This test case can be combined with the data entry verification case (below).

Enter Time/Date-Dependent Data into system prior to Midnight and observe operation through Midnight crossing

The objective of this test case is verification that time- or date-dependent data entered into the [Subject] survive the transition and, after the transition, are processed correctly. This test case can be combined with the routine operation test case (above).

Additional Test Cases

The test cases defined for evaluating the [Subject] when operating through a critical midnight crossing and for ensuring that time- or date-dependent data are handled properly through the crossing are sufficient for many systems.

There are some systems, however, that process dates in ways that require additional testing. For example, systems that make projections into the future or that analyze past data will require tests associated with those functions. Each such test will necessarily consist of four parts, one for each of the critical midnight crossings described above.

Further, if [Subject] has a data recording and playback capability, test cases must be developed that play back information recorded during each of the critical midnight crossings. These playback tests should consist of two parts: first, initiating a playback beginning sometime prior midnight and extending through the crossing; and, initiating a playback beginning after midnight.

Failover Operation

A failover is defined as an event or situation that causes the master computer to a system to fail and the assumption of application processing control by a standby computer. For systems that employ a “hot standby” computer for use in case of the failure of the primary computer system, failover tests are necessary to ensure Y2K compliance of the total system. Systems that do not employ standby computers may omit failover test cases.

Failover test cases are not executed because one expects a failover to occur at the critical midnight crossings (although they are possible), but because the takeover operations performed by the standby computer frequently expose shortcomings in software design and implementation that routine test cases do not. In each case, the failover should be initiated as close to the midnight crossing as possible. The goal of the test is to force the standby computer to begin operation after the midnight crossing with data that was accumulated prior to the crossing.

Test 1:

Midnight Crossing: December 31, 1999 to January 1, 2000

Situation: Routine Operation from N minutes prior to M minutes after Midnight

Enter Time- or date-dependent data item(s) into system prior to Midnight and observe operation through Midnight crossing

The object of the first part of this test case is verification that the [Subject] software can survive the critical midnight transition and to verify that, once the transition has occurred, the remainder of [Subject] date processing.

The objective of the second part of this test case is verification that time- or date-dependent data entered into the [Subject] survive the transition and, after the transition, are processed correctly.

Test Steps	√
1. Restart [Subject] system.	
2. Set system date/time to December 31, 1999, 23:50:00.	
3. Observe general system operation.	
4. Initiate data recording (if capability exists).	
5. Identify "key parameter" or event and note time of occurrence Event:_____ Time:_____	
6. Generate time- or date-dependent data item Time- or date-dependent data item parameters: _____ _____ _____	
7. Observe midnight crossing OK_____ Not OK_____	
8. Identify "key parameter" or event and note time of occurrence Event:_____ Time:_____	
9. Perform additional tests of date processing as required by system characteristics.	
10. Terminate test.	

Was this test Y2K satisfactory? _____ **If other than "Yes," explain:**

Test 2:

Midnight Crossing: February 28, 2000 to February 29, 2000

Situation: Routine Operation from N minutes prior to M minutes after Midnight

Enter Time- or date-dependent data item(s) into system prior to Midnight and observe operation through Midnight crossing

The object of the first part of this test case is verification that the [Subject] software can survive the critical midnight transition and to verify that, once the transition has occurred, the remainder of [Subject] date processing.

The objective of the second part of this test case is verification that time- or date-dependent data entered into the [Subject] survive the transition and, after the transition, are processed correctly.

Test Steps	√
1. Restart [Subject] system.	
2. Set system date/time to February 28, 2000, 23:50:00.	
3. Observe general system operation.	
4. Initiate data recording (if capability exists).	
5. Identify “key parameter” or event and note time of occurrence Event:_____ Time:_____	
6. Generate time- or date-dependent data item Time- or date-dependent data item parameters: _____ _____ _____	
7. Observe midnight crossing OK_____ Not OK_____	
8. Identify “key parameter” or event and note time of occurrence Event:_____ Time:_____	
9. Perform additional tests of date processing as required by system characteristics.	
10. Terminate test.	

Was this test Y2K satisfactory? _____ If other than “Yes,” explain:

Test 3:

Midnight Crossing: February 29, 2000 to March 1, 2000

Situation: Routine Operation from N minutes prior to M minutes after Midnight

Enter Time- or date-dependent data item(s) into system prior to Midnight and observe operation through Midnight crossing

The object of the first part of this test case is verification that the [Subject] software can survive the critical midnight transition and to verify that, once the transition has occurred, the remainder of [Subject] date processing.

The objective of the second part of this test case is verification that time- or date-dependent data entered into the [Subject] survive the transition and, after the transition, are processed correctly.

Test Steps	√
1. Restart [Subject] system.	
2. Set system date/time to February 29, 2000, 23:50:00.	
3. Observe general system operation.	
4. Initiate data recording (if capability exists).	
5. Identify “key parameter” or event and note time of occurrence Event:_____ Time:_____	
6. Generate time- or date-dependent data item Time- or date-dependent data item parameters: _____ _____ _____	
7. Observe midnight crossing OK_____ Not OK_____	
8. Identify “key parameter” or event and note time of occurrence Event:_____ Time:_____	
9. Perform additional tests of date processing as required by system characteristics.	
10. Terminate test.	

Was this test Y2K satisfactory? _____ **If other than “Yes,” explain:**

Test 4:

Midnight Crossing: December 31, 2000 to January 1, 2001

Situation: Routine Operation from N minutes prior to M minutes after Midnight

Enter Time- or date-dependent data item(s) into system prior to Midnight and observe operation through Midnight crossing

The object of the first part of this test case is verification that the [Subject] software can survive the critical midnight transition and to verify that, once the transition has occurred, the remainder of [Subject] date processing.

The objective of the second part of this test case is verification that time- or date-dependent data entered into the [Subject] survive the transition and, after the transition, are processed correctly.

Test Steps	√
1. Restart [Subject] system.	
2. Set system date/time to December 31, 2000, 23:50:00.	
3. Observe general system operation.	
4. Initiate data recording (if capability exists).	
5. Identify "key parameter" or event and note time of occurrence Event:_____ Time:_____	
6. Generate time- or date-dependent data item Time- or date-dependent data item parameters: _____ _____ _____	
7. Observe midnight crossing OK_____ Not OK_____	
8. Identify "key parameter" or event and note time of occurrence Event:_____ Time:_____	
9. Perform additional tests of date processing as required by system characteristics.	
10. Terminate test.	

Was this test Y2K satisfactory? _____ **If other than "Yes," explain:**

Summary of Y2K Evaluation