

[YOUR ORGANIZATION]
Year 2000 System-of-Systems Test Plan

[YOUR ORGANIZATION]

insert author(s) here

insert date here

Abstract

This Test Plan describes the procedures to be followed to verify that the [YOUR ORGANIZATION] systems identified in the System Inventory in Appendix F can successfully interoperate as a system-of-systems during and after the Year 2000 (Y2K) century date change.

Table of Contents

1	YEAR 2000 SYSTEM-OF-SYSTEMS TEST	5
1.1	Y2K OVERVIEW	5
1.2	Y2K SOST – OBJECTIVE.....	6
1.2.1	Y2K – Definitions.....	6
1.2.2	Y2K Activity.....	6
1.2.3	Year 2000 (Y2K) Compliant.....	6
1.2.4	Certification.....	7
1.2.5	Interoperability.....	7
1.2.6	Contingency Planning.....	7
1.3	TEST SCOPE.....	8
1.4	TEST APPROACH.....	8
1.4.1	Y2K Data Re-use.....	8
1.4.2	Test Environment.....	8
1.4.3	Test Constraints and Limitations.....	8
1.4.4	Risk Management.....	9
1.4.5	Configuration Management.....	9
1.5	TEST ASSUMPTIONS.....	9
1.5.1	Y2K Compliance.....	9
1.5.2	Contingency Plans.....	10
1.5.3	Test Environment.....	10
1.5.4	Personnel.....	10
1.5.5	Funding.....	11
1.6	TEST TEAM.....	11
1.6.1	Participating Organizations.....	11
1.6.2	Roles and Responsibilities.....	11
1.6.3	Test Team.....	12
1.7	TEST PROCESS.....	12
1.7.1	Test Suspension Criteria.....	13
2	INFORMATION REQUIREMENTS	14
2.1	SYSTEM PARTICIPATION AND PRIORITIZATION	14
2.2	OPERATIONAL THREAD IDENTIFICATION.....	14
2.3	CONTINGENCY PLANNING.....	14
2.3.1	Bridge Programs.....	14
3	TEST PLANNING	15
3.1	TEST SCHEDULE.....	15
3.1.1	Milestones.....	15
3.1.2	Road Map.....	15
3.1.3	Test Progression.....	15
3.2	RESOURCE REQUIREMENTS	16
3.2.1	Personnel Requirements.....	16
3.2.2	Test Environment Requirements.....	16
3.2.3	Logistics Requirements.....	17
3.2.4	Documentation Requirements.....	18
3.3	TEST CASE DEVELOPMENT.....	18
3.3.1	Prerequisite Conditions.....	18

3.3.2	<i>Assumptions and Constraints</i>	19
3.3.3	<i>Criteria for Evaluating Results</i>	19
3.3.4	<i>Data and Configuration Standards</i>	20
4	TEST EXECUTION	22
4.1	PRE-TEST PREPARATION	22
4.1.1	<i>Hardware</i>	22
4.1.2	<i>Software</i>	22
4.2	TEST PROCEDURES.....	22
4.3	POST-TEST ANALYSIS.....	23
4.3.1	<i>Data Reduction</i>	23
4.3.2	<i>Final Report Generation</i>	23
4.4	REGRESSION TESTING	23
A.	INFORMATION REQUIREMENTS CHECKLISTS	24
A.1	SYSTEM INVENTORY CHECKLIST	24
A.2	SYSTEM INTERFACE CHECKLIST	24
A.3	SYSTEM INVENTORY	24
B	TEST PLANNING CHECKLISTS	25
B.1	TEST RESOURCES CHECKLIST.....	25
C.	TEST EXECUTION, DATA REDUCTION, AND FINAL REPORT CHECKLISTS	26
C.1	PRE-TEST EXECUTION CHECKLIST	26
C.2	TEST EXECUTION CHECKLIST	26
C.3	DATA REDUCTION CHECKLIST.....	26
C.4	FINAL REPORT CHECKLIST.....	26
C.5	REGRESSION TEST CHECKLIST	27
D	TEST INCIDENT REPORT	28
E	BASE Y2K TEST CASES	30
F	SYSTEMS PARTICIPATING IN SOST	31
	GLOSSARY	32

1 Year 2000 System-of-Systems Test

This Test Plan describes the procedures to be followed to verify that the [YOUR ORGANIZATION] systems identified in the System Inventory in Appendix F can successfully interoperate as a system-of-systems during and after the Y2K century date change.

1.1 Y2K Overview

The Y2K problem results from the storage of year data as two numeric digits instead of four (for example, using the MMDDYY Gregorian date format instead of MMDDYYYY; or using a YYDDD Julian date instead of YYYYDDD). Historically, this approach worked because systems could correctly assume that the two year digits were always preceded by “19”. This practice fails, however, as the year 2000 is approached and dates on either side of the century crossing are encountered by information systems. Now, systems must have some way of determining the correct century.

While a strict definition of the Y2K problem limits consideration to the use of two digits to represent a year, other potential date-related anomalies are generally grouped into the broad definition of Y2K. Specifically, the year 2000 is a leap year even though it is evenly divisible by 100. The rule for determining leap years is

- If a year is evenly divisible by 4 it is a leap year, unless
- It is also evenly divisible by 100 in which case it is not, unless
- It is also evenly divisible by 400 in which case it is.

Instances exist where systems have been developed implementing only two of the three rules.¹

Additionally, many systems use date information to implement “non-date” functions. Examples of these functions are:

1. Using the date (usually in conjunction with the time of day) as a seed for random number generation;
2. Using the date field in data records to indicate special processing (e.g., setting an expiration date to 9/9/99 to indicate an indefinite retention);

Such uses and others are usually included in the category of potential Y2K problems.

At risk are applications software, system software, or hardware platforms that use dates. The possibilities for interaction problems as dates and date-related calculations pass among applications and files are unlimited.

¹ It has been pointed out that if these systems had limited themselves to the first rule only, they would correctly recognize 2000 as a leap year (although they would also believe that 1900 was a leap year, which it was not).

1.2 Y2K SOST – Objective

The object of the Y2K System-of-Systems Test (SOST) is to ensure that interconnected systems can properly transmit and receive date information. Because the exchange of data among several systems is the primary subject of SOST, emphasis will be on interfaces external to individual systems. SOST augments previous Y2K test phases performed as part of a structured, layered approach to achieve Y2K compliance among [YOUR ORGANIZATION]'s inter-operating systems.

Failure to assure proper exchange of date data among [YOUR ORGANIZATION]'s systems can allow the promulgation of corrupt data. Any such failure can significantly affect [YOUR ORGANIZATION]'s core missions. This plan describes a comprehensive SOST that will ensure uninterrupted access to [YOUR ORGANIZATION] system resources, untainted from Y2K-related data corruption, throughout the Y2K transition period.

SOST will encounter several challenges, among them interoperability coordination, analysis and testing among systems that may be geographically separated.

1.2.1 Y2K – Definitions

This document contains terminology commonly used by the Y2K community. Definitions of these are provided in the following sections.

1.2.2 Y2K Activity

Y2K activity is that activity intended to ensure that a system is Year 2000 compliant (see 1.2.3). This activity takes place in five phases:

- **Awareness.** The awareness phase focuses on promoting cognizance of the Y2K situation.
- **Assessment.** The assessment phase consists of compiling a system inventory and evaluating the severity of the Y2K problem.
- **Renovation.** The renovation phase involves replacing non-compliant systems, retiring them, or repairing them.
- **Validation.** The validation phase involves testing renovated systems for Y2K compliance and for continued compliance with functional and interface requirements.
- **Implementation.** The implementation phase is the final one and involves deployment of validated systems.

1.2.3 Year 2000 (Y2K) Compliant

To be Year 2000 compliant,² a system must:

² As defined in the Federal Acquisition Regulation, released August 22, 1997.

- Correctly process dates before and after the year 2000,
- Recognize the year 2000 as a leap year,
- Accept and display dates unambiguously, and
- Correctly process logic dates that are used for “non-date functions.”

In addition, a system must operate properly and provide correct and unambiguous date information as it encounters other “critical date transitions.” Critical date transitions are those for which evidence, either hard or circumstantial, exists that date-related processes might encounter error. Some examples of critical date transitions are:

- 31 December 1999 to 1 January 2000—the Y2K transition date;
- 28 February 2000 to 29 February 2000—the Y2K leap-year transitions;
- 29 February 2000 to 1 March 2000; and
- 31 December 2000 to 1 January 2001.

The above are the set of critical date transitions for which tests will be required. Other critical date transitions may be exercised (e.g., Government or corporate fiscal year change-over).

The basic set of test cases SOST will use to determine compliance is contained in Appendix E.

1.2.4 Certification

Certification is the culmination of the validation process, where a trained certifier and [YOUR ORGANIZATION]’s Y2K Program Manager jointly and formally attest that a system is Y2K compliant.

1.2.5 Interoperability

Interoperability is the ability of two or more systems to exchange information and to mutually use the information that has been exchanged.³

Interoperability testing ensures that the external interfaces to a given system are functional. This provides confidence that the information required to maintain operations of each interfacing system will be provided in an acceptable format with acceptable content.

1.2.6 Contingency Planning

Contingency planning is the process of identifying potential failures of a system or systems and then developing procedures to minimize or eliminate the effects of these failures.

Y2K contingency planning is the process of identifying proposed courses of action to take if the Y2K and associated risk control efforts do not achieve their desired outcomes. As part of SOST,

³ ANSI/IEEE Std 100 –1988, *Standard Dictionary of Electrical and Electronics Term*, 1988.

system contingency plans will be reviewed, and risk areas will be identified. Events that cause contingency conditions will be invoked as part of each test.

1.3 Test Scope

Y2K SOST will include only those systems identified by [YOUR ORGANIZATION] (see Appendix F).

The test process will focus on inter-system operational threads identified by the Y2K Program Manager. To ensure operational integrity, all operational threads will be tested, including those not obviously involving date data. The systems on the receiving side of each interface will be monitored to ensure date data is accurate and appropriately processed.

All functions, operations, and system processing methods that provide date data to an external interface of a system under test will be exercised.

1.4 Test Approach

1.4.1 Y2K Data Re-use

The SOST effort will utilize existing test scenarios, simulations, and test input data from the Awareness, Assessment, and Renovation Y2K phases and from normal system test activities. Reusing this information will significantly minimize the time and effort required to coordinate and perform SOST.

1.4.2 Test Environment

To accommodate SOST, a test environment(s) that duplicates each inter-system operational component (i.e., hardware and software), and supports the Y2K transition time frame will be developed. All the individual components in this environment will be Y2K safe. There may be cases where some systems involved in the test may not yet be Y2K compliant. To accommodate this possibility, data bridges (see 1.4.3 and 2.3.1) or other techniques will be used.

[YOUR ORGANIZATION]'s Local Area Network (LAN) will provide the communications backbone for SOST. In the event a system cannot be tested via the LAN, an appropriate alternate test facility will be used.

1.4.3 Test Constraints and Limitations

Test constraints include the unavailability and inaccessibility of resources needed during the Y2K renovation phase, incomplete knowledge of *all* the test areas, and incomplete Y2K test certification for participating systems.

Technology limitations specific to Y2K testing may impede SOST in simulating the near- or post-transition time frame. For example, abruptly advancing the system date may cause products or underlying infrastructure (e.g., the operating system) to irreversibly "expire." In addition, other system resources may be affected:

- User IDs and access privilege(s) may expire;
- COTS licenses may expire;
- Data files may be deleted;
- System authorizations and protections schemes may expire.

1.4.4 Risk Management

Risk Management is the practice of applying discipline to the Y2K correction process to allow for the identification and mitigation of risks before they negatively affect a system's ability to perform its mission. Y2K risk management activities include the identification and assessment of risks, and the planning of contingent or alternative solutions and activities.

Risk identification and assessment include determining how a system or device may fail and what the effect will be: will it stop working; will it work but display dates incorrectly; will it appear to work correctly but pass incorrect data to interfacing systems. The effect must be assessed and the appropriate countermeasures applied.

1.4.5 Configuration Management

Configuration Management (CM) will play a vital role in the Y2K SOST process. CM will ensure that

- Software versions are adequately controlled;
- Software modifications are accurately tracked; and
- Data used for test cases are identified and controlled.

CM of system source code modifications will be performed according to standard procedures for each system or application. When a defect in the application code is detected, the standard modification procedures of the affected system will be followed. The test team will send a Test Incident Report (see Appendix D) to the Test Manager who will determine the appropriate course of action.

1.5 Test Assumptions

System-of-systems testing is not intended as a comprehensive test of all functionality of each participating system. Rather, it is a test to minimize the risk that inter-system interfaces cannot accommodate the needs of the connected systems during and after the century roll-over and the transition of other key dates. Consequently, SOST activities are predicated on certain assumptions. These assumptions are discussed below.

1.5.1 Y2K Compliance

Y2K SOST presupposes applications, COTS/GOTS products, and individual systems are Y2K compliant and that participating systems have certificates of compliance available.

It is also assumed that any infrastructure provided, including test-specific infrastructure (e.g., simulators), is Y2K safe.⁴

1.5.2 Contingency Plans

Many systems or sites have developed Continuity of Operations Plans (COOPS) describing procedures and actions to be taken in the event of infrastructure failure such as:

- Loss of power or air conditions;
- Flood;
- Loss of critical systems or interfaces.

Y2K SOST assumes that these COOPs have been reviewed and updated to accommodate failures associated with critical date transitions (Y2K issues). Specifically, it is important that

1. Critical operations have disaster recovery plans and facilities;
2. Y2K failure recovery has been tested during the system testing phase;
3. Operational work-arounds for interface failures and “likely” infrastructure failures are ready.

Because the Y2K problem is endemic, failures may occur in several systems or infrastructure components simultaneously. In particular, back-up systems may not be available to take over for failed systems because they, too, may have failed.

Each system’s contingency plan will be reviewed and a set of risk contingency actions will be identified for inclusion the SOST test procedures. At an appropriate point in the SOST, the contingency conditions will be invoked in order to assess the effectiveness of the mitigation procedures.

1.5.3 Test Environment

The SOST process assumes the availability of and accessibility to the test facilities. Any operational test bed(s) that duplicate(s) the field environment will have the appropriate equipment, run the proper version of the baseline software, and contain the appropriate network resources.

1.5.4 Personnel

The proper system support personnel, familiar with the operation of the systems under test (or qualified to operate them) must be available for the SOST. Test personnel must have the appropriate security clearances to enter the test facility and perform their required functions. Their responsibilities will include:

⁴ We use the term “safe” rather than compliant because the test infrastructure components may be procured and assembled in an environment that is somewhat less formal than that in which deployable systems are acquired. Nevertheless, it is important that evidence exist that these infrastructure components will not contribute to Y2K failures during testing.

- Conduct of SOST and documentation of the results;
- Determination and approval of Y2K SOST compliance; and
- Performance of Configuration Management activities.

1.5.5 Funding

To prepare for and conduct SOST, the proper funding must be in place. Limitations in this area can hamper the test process and/or reduce the number of systems that can be certified Y2K compliant during the system-of-systems test phase. Cost estimates for SOST will be provided as a separate document.

1.6 Test Team

This section discusses the make-up and activities of the SOST test team.

1.6.1 Participating Organizations

[YOUR ORGANIZATION] will have the lead role in the planning and execution of the [YOUR ORGANIZATION] Y2K test effort.

Other organizations participating are: TBD.

To ensure a strong, cohesive, flexible, and effective SOST team, the services of other organizations and/or employees will be enlisted as required.

1.6.2 Roles and Responsibilities

The [YOUR ORGANIZATION] SOST team will have the lead role in the planning and execution of the [YOUR ORGANIZATION] test effort.

Specifically, [YOUR ORGANIZATION] Y2K SOST team will:

1. Develop and maintain this Y2K SOST plan and coordinate it with other cognizant organizations;
2. Determine test strategies;
3. Support Y2K validation and certification activities, if required, to help ensure that systems can participate in the SOST;
4. Define entrance and exit criteria to validate that multiple systems are Y2K-compliant and interoperable;
5. Support Y2K interoperability testing and associated data analysis;
6. Interface with applicable organizations with systems being tested;
7. Establish and co-host a Y2K Test Working Group (TWG) which, in turn, will establish committees to work specific test issues (e.g., test asset arrangements, test strategy definition, etc.);

8. Provide a distributed support environment to permit testing of selected Y2K-compliant systems for interoperability;
9. Provide both secure and unclassified connectivity;
10. Provide a development/integration environment for risk reduction and technology assessment;
11. Provide individual system test teams test team an early opportunity and test bed to identify Y2K issues and assess alternatives;
12. Establish communications links to various facilities and geographically separated systems under test.

1.6.3 Test Team

System-of-systems testing requires the execution of a number of functions. Ideally, each function is the responsibility of a single individual or group of individuals. However, it is acceptable for several functions to be executed by a single individual. The SOST test team will be composed of the following members:

- **Test Manager**—The Test Manager is in charge of the total the testing activity and oversees all test planning, preparation of test specifications, coordination (with project, development and remediation, interface and third party products, configuration, quality assurance, and redeployment managers), test/retest scheduling, test monitoring and tracking, and reporting.
- **Test Environment Leader**—The Test Environment Leader ensures proper test bed preparation and maintenance.
- **Test System Technical Leader**—The Test System Technical Leader provides system support services for the system under test (e.g., platforms, system software support, networks).
- **Test Planners**—Test Planners develop test plans, test cases, and scenarios. In addition, they coordinate inter-system interoperability testing; they develop test data; and they define the required test environments.
- **Testers**—Testers actually execute the planned test steps and record the results. The test team can include end-users and subject matter experts as well as technical staff and professional testers.
- **Observers**—During the course of the SOST process, observers may be present to oversee testing activities and to witness results.

1.7 Test Process

The Y2K SOST process will consist of three phases:

- Information gathering;

- Test Planning; and
- Test execution and analysis

This approach is an iterative, evolutionary model of testing. Each phase will proceed linearly to the next unless an update is required in the current phase, which will necessitate a return to the previous phase. Each phase will have required input items, associated processing and other activities, and products.

Where appropriate, as resources and other test constraints permit, identified test items will be performed in parallel to expedite the test process. The components of the Y2K SOST process are described in the following section.

1.7.1 Test Suspension Criteria

In the event a system does not respond to test stimuli as expected or cannot support a significant number of scheduled activities, the Test Manager will determine if testing should be suspended. Upon corrective action, the Test Manager will also determine if and when testing can resume.

2 Information Requirements

2.1 System Participation and Prioritization

SOST will be limited to systems identified by [YOUR ORGANIZATION]' s Program Manager for Y2K and listed in Appendix F.

2.2 Operational Thread Identification

Initial SOST efforts will concentrate on the identification of interconnected systems and their corresponding operational interfaces. Many systems may be connected, however, SOST will only test the operational interfaces among the identified systems. That is, systems A, B, C, and D may interconnect, but if only the interface between systems A and C provides operational functionality; that interface will be the only one tested.

Test plan design will focus on information that characterizes the operational interfaces (e.g., type and volume of data passed across the interface) to determine the granularity of SOST testing. Additionally, information regarding each system's internal components (e.g. date horizons) and date formats (e.g., Julian) will be utilized.

2.3 Contingency Planning

SOST will rely on the existence of up-to-date, complete and accurate contingency plans for each system under test. The disaster recovery plans and facilities (including all operational "work-arounds") for each system will also be utilized as appropriate. This presupposes that appropriate Y2K failure recovery testing has occurred during the Y2K renovation and testing phases.

System mechanisms (e.g., bridge programs, see 2.3.1) are also expected to be in place among inter-system interfaces to handle known differences in Y2K renovation.

2.3.1 Bridge Programs

When two systems expect the exchanged data to be in different formats, bridge programs are created to translate between them. These bridge programs can be used as "firewalls" to help prevent data corruption. Industry has defined two types of bridges: filters, and wrappers.

A filter converts date formats between two files or databases. A wrapper converts date formats between two actively interacting systems.

A forward bridge converts dates from legacy format (two digits) to century format (four digits). A backward bridge converts a date from century format to legacy format.

Error handling is a primary consideration when designing bridges. In an error-end design, errors are not passed through the bridge, but must be handled by the sending system. In an error-through design, errors are passed through the bridge and are handled by the receiving system. Each method can and should provide for the logging of detected errors.

3 Test Planning

3.1 Test Schedule

A master schedule will be developed for the systems and will include milestones for the conduct of SOST. This master schedule will include information for the systems, all required assets, the planning of the test, its conduct, data reduction, analyses, and publication of the test report.

3.1.1 Milestones

In order to track and report progress for the Y2K SOST effort, milestones will be established. For each system, the following milestones will be scheduled and reported:

1. Development of baseline tests and test data;
2. Execution and review of the baseline tests;
3. Execution and review of simulated Y2K date testing;
4. Execution and review of actual Y2K date testing;
5. Regression testing (as necessary);
6. Documentation review; and
7. Test reporting.

Periodically, a master report of test progress will be produced by the Test Manager and forwarded to the appropriate Y2K SOST management personnel.

3.1.2 Road Map

TBD. Upon identification of the systems under test, a roadmap will be developed.

3.1.3 Test Progression

Testing will occur in three phases:

1. **Baseline testing** which will determine that the systems under test provide required functionality while working in a current date environment;
2. **Simulated Date testing** which will test system interfaces in a current date environment, but with advanced date information; and
3. **Actual Date testing** which will exercise system interfaces in an advanced date environment, with past (if appropriate) and advanced date information.

3.2 Resource Requirements

3.2.1 Personnel Requirements

The number, type, and skill level of personnel needed during the test period at the test site(s) and the dates and times they will be needed will vary depending on the test case. The categories of personnel for SOST have been described in 1.6. For each test, the following minimum personnel are required:

- Test Manager;
- Test Environment Manager;
- Test Systems Engineer;
- Tester.

3.2.2 Test Environment Requirements

The specific requirements of each facility will vary according to the specific requirements of each of the systems being tested and their interrelationship to the other systems under test. A database will be developed to catalog the existing and required assets at each of the facilities.

Part of the test planning process will verify that each test facility is certified as Y2K compliant. Ensuring certification will help isolate Y2K issues to particular interfaces and/or individual systems.

3.2.2.1 Hardware

As stated earlier, the specific needs of each test facility with respect to the systems under test will vary. For each test, a list of required hardware will be maintained. Prior to the test team arrival, the test environment personnel at the test facility will ensure that all hardware items are available and operational and have been integrated into the test configuration. This equipment may include, but is not limited to, the following hardware:

- Teleprocessing monitors;
- Workstations
- Telecommunications hardware;
- Network timestamps (LAN/WAN network, clock time);
- Storage/RAM for all equipment;⁵

3.2.2.2 Software

A list of system software will be maintained for each test. Prior to test team arrival, the appropriate systems support personnel will ensure that all software items are available and

⁵ Y2K SOST may increase storage requirements because of internal monitoring, etc. In some instances, the SOST may require a more powerful CPU for systems under test than would normally be deployed.

operational and have been integrated into the test configuration. This software may include, but is not limited to, the following:

- Operating system;
- Systems under test;
- Major subsystems;
- Database management system(s);
- Compilers and assemblers;
- Locally developed software that is used in conjunction with the system(s) under test;
- Workstation software;
- Embedded software;
- Telecommunications software.

3.2.2.3 Tools

The use of computer-aided software testing tools and test scripts has the potential to significantly reduce the testing and validation effort. Such tools include

- Clock simulators that can transparently change the system clock;
- Test case analyzers that assist in determining test data coverage;
- Specification-driven test data generators;
- Test libraries to organize test data;
- Test drivers that trigger test case events;
- Test management tools that help in preparation and management of test data, comparison of results, scheduling, incident tracking, and documentation management.

3.2.2.4 Target Configuration

The target configuration for the SOST will allow the systems under test to interoperate in an environment where the system date can be advanced to the Year 2000 situation.

3.2.3 Logistics Requirements

The coordination of interface testing among geographically disparate systems will require the appropriate acquisition and delivery of materials (see 3.2.2.1 and 3.2.2.2) for SOST, and the participation of the appropriate test personnel at particular times before, during, and after SOST. The proper coordination of software and hardware assets will minimize SOST resource usage. The logistics activities required to ensure a coordinated test are:

- Required hardware and software assets delivered to the test facility;

- Operations personnel set up and check equipment;
- All personnel conduct test;
- All personnel gather material necessary for test report;
- Operations personnel break down equipment and return assets.

3.2.4 Documentation Requirements

Documentation requirements fall into two categories: documents needed for testing; and documents required for the test report.

The test documents required for testing must be available at the test site and be accessible by the test team during all aspects of SOST. These test documents include:

- Test systems Interface Control Documents (ICDs) or Memoranda of Agreement (MOAs);
- Test systems Operators' Manuals;
- Test System Test Plan.

Documents required for preparation of the test report include:

- Test logs;
- Test setup checklist;
- Reports of individual Test Team members;
- Test deviation reports, as necessary.

3.3 Test Case Development

This section identifies approaches to be used in Y2K System-of-Systems testing. This includes defining common approaches to test case development and the identification of common data sets. The overall intent is to minimize the amount of test data needed the number of tests to be run.

For each test case, test environment requirements will be used to support the coordination of testing. This will include hardware,

software versions, communications equipment, mode of usage (e.g., "stand alone"), test tools, and other facilities.

3.3.1 Prerequisite Conditions

For each test case developed for [YOUR ORGANIZATION] SOST, the test case developer will assume:

- Each system participating in the test (including contractor supported components of the communications infrastructure) has been Y2K certified;

- For interfacing systems, all aspects of Y2K failure conditions have been tested according to the DOD Checklist
- A system failure that results in the invoking of a contingency action that permits continued mission support will not count as a Y2K failure.

3.3.2 Assumptions and Constraints

Because each participating system will have been previously Y2K certified, the risk of a hard failure of any component due to internal causes is minimal. Consequently, test cases developed for SOST will emphasize normal operation and the successful transmission of data across interfaces.

Test will be conducted by the test team in a controlled environment defined and established for SOST.

Members of the test team will ensure that test assets and facilities have been Y2K certified and will document any deviations from this assumption as well as any deficiencies noted as a result of testing.

All observations and anomalies during testing will be logged and documented in the test report.

If changes to the systems undergoing testing are required, the test team will identify the amount of retesting necessary to verify the change and to ensure the continued operation of other components.

3.3.2.1 Test Case Descriptions

Three kinds of test cases will be utilized for Y2K SOST:

1. **Baseline Test.** Exercise the systems under test in their normal configuration with a current date to demonstrate basic functionality;
2. **Simulated Future Test (optional).** Exercise the systems under test in their normal configuration but with data containing advanced dates.
3. **Actual Future Date Test.** Exercise the systems under test with the system date advanced to critical Y2K transition dates.

3.3.3 Criteria for Evaluating Results

System-of-systems testing for Year 2000 compliance is a risk determination and management activity. At the completion of each SOST test case, the ability of each system and of the ensemble will be reviewed to determine the degree to which each demonstrated an ability to accomplish its assigned mission tasks.

Additionally, the ability of contingency plans to direct activities during and after the failure of interfacing systems and infrastructure will be assessed. Evaluation criteria will be assigned to the categories as shown in Table 1.

Table 1. Success/Fail Criteria

Non-Contingency Testing	Contingency Testing
Success: Mission tasks accomplished	Success: Critical mission tasks accomplished
Limited Success: Critical mission tasks accomplished, but one or more non-critical tasks not accomplished	N/A
Failure: One or more critical mission tasks not accomplished	Failure: One or more critical mission tasks not accomplished

If, during non-contingency testing, mission tasks (critical or not) are not accomplished, the failure will be documented in a Deficiency Report (DR), and a preliminary determination of the cause will be attempted. If the problem appears to be related to a critical date transition, the failure will be noted as a Y2K deficiency. Note, however, that the categorization of a problem as either a Y2K deficiency or as some other type will not affect the final SOST evaluation: the inability to accomplish a critical mission tasks, whether because of Y2K errors or others, will result in a “Failure” evaluation.

3.3.4 Data and Configuration Standards

To ensure the integrity of test procedure data processing, data will be entered and recorded in a standard fashion. Any deviations from this standard will be presented to the Test Manager who will determine if the test procedure has been sufficiently compromised to warrant the suspension of testing. To ensure the proper use and recording of data during the SOST process, the following data standard will be adhered to:

- Unless approved by the Test Manager, only data from the test plan will be entered into the systems under test. All test data deviations will be appropriately documented.
- All observations, notes, or other notable items relevant to the conduct of data entry will be treated as data and will be included in the test procedure documentation.
- Entry of test data items will be marked (i.e., checked off) in the appropriate test procedure step. The tester will sign-off each page of the test procedure, attesting to test data entry and respective test data entry verification, and the correctness of all other relevant observations and notes.
- The Test Manager will review each page of the test procedure and sign the test procedure document as a reviewer. This signature attests that the test procedure was performed properly and that all relevant documentation is included.

The proper configuration of test assets is important to ensure the integrity of the test environment. The configuration will be verified before the start of testing and will be maintained throughout the test process. Any deviations will be reported to the Test Manager who will determine if the integrity of the test environment has been sufficiently compromised to warrant the suspension of testing. To ensure the proper use and recording of data during the SOST process, the following configuration standard will be adhered to:

- The test environment will be set up in accordance with the specifications by the Test Environment support staff;
- At the completion of set up, the Test Team operations and support staff will perform and document check out testing;
- At the completion of check-out, and with the approval of the Test Manager, the gathering and recording of test data will begin;
- After Baseline and Future date testing (see 3.1.3 and 3.3.2.1) is complete, the operations and support staff will update the test environment to accommodate Actual Date Testing.
- Upon Test Manager review and approval of the modifications made for Actual Date Testing, Actual Date Testing will begin.
- Any deviation from the test configuration standard will be reported to the Test Manager who will determine if the integrity of the test environment has been sufficiently compromised to warrant the suspension of testing.

4 Test Execution

4.1 Pre-Test Preparation

Prior to the start of test case execution, the test environment hardware and software will be configured. Pre-test activities will be documented, reviewed, and approved prior to any testing activity.

4.1.1 Hardware

The hardware requirements for each test case will vary by system. The specification of these requirements will be included in the individual test case procedures.

4.1.2 Software

The software requirements for each test case will vary by system. The specification of these requirements will be included in the individual test case procedures.

4.2 Test Procedures

During the course of testing, the following procedural steps will occur:

1. Initiate the test log to record the session;
2. Initiate set-up steps;
3. Start test case
 - a. Perform Baseline Test—Test with the current system date. Demonstrate system operability.
 - b. Perform Simulated Future Date Test (optional)—Simulated future date testing using a date simulation tool.
 - c. Perform Actual Future Date Test—Test with clocks set to specified future system date(s).
4. Perform retesting as required;
5. Shut system down;
6. Stop log;
7. Perform test wrap-up duties;
8. Produce test status reports.
 - a. Include appropriate test metrics;
 - b. Include test deviation report.

4.3 Post-Test Analysis

4.3.1 Data Reduction

Post test analysis involves the enumeration and characterization (success, limited success, or failure) of incidents that occurred by test case and the development of overall SOST statistics.

4.3.2 Final Report Generation

The final report will be prepared by the Test Manager with the assistance of all personnel involved in the SOST effort. The Program Manager for Y2K will review and approve the report before its release.

4.4 Regression Testing

Regression testing will be performed as deemed necessary by the Test Manager. Standard system modification procedures will be adhered to for modification of source code. The Test Manager will provide detailed descriptions of the purpose and results of regression testing in the test status report(s).

A. Information Requirements Checklists

A.1 System Inventory Checklist

1. System name, location, and statement of purpose;
2. POC;
3. Y2K compliance certification date and level;
4. System environment;
5. Availability of system for testing;
6. External interface(s) description including interface strategy for sending and receiving messages;
7. Infrastructure components (e.g., date horizons, date-dependent functions, non-date functions);
8. Current test plan and contingency plan status.

A.2 System Interface Checklist

1. Name of interface;
2. POC for reporting progress and coordinating schedule;
3. Kind of interface and data transmitted;
 - a. Kind of connection;
 - b. Nature of data;
 - c. Reference to ICD or MOA documenting the interface.
4. Nature of modifications (if any);
 - a. General category (field expansion, windowing, etc.)
 - b. Details of the date-handling algorithm
5. Testing dates

A.3 System Inventory

TBD

B Test Planning Checklists

B.1 Test Resources Checklist

1. Test facility;
 - a. Date and time access is required;
 - b. Site clearance required;
 - c. Room access clearance required.
2. On-site test support
 - a. Attempt(s) to repeat
 - b. Tester(s)
 - c. Observer(s)

C. Test Execution, Data Reduction, and Final Report Checklists

C.1 Pre-test Execution Checklist

1. Name of system
2. Test account
 - a. Password
 - b. Privileges
 - c. Network access
 - d. Expiration date
3. Test facility access
4. System backup completed

C.2 Test Execution Checklist

1. Test ID
2. Test date and time
3. Test operator
4. Completed test plan and test log delivered to Test Manager

C.3 Data Reduction Checklist

1. Date
2. Personnel
3. Analysis performed

C.4 Final Report Checklist

1. Author(s)
2. Reviewer(s)
3. Recipient(s)
4. Delivery
5. Presentation

C.5 Regression Test Checklist

1. Modification installed in target system, tested, documented, and approved (Y2K compliant) in accordance with standard system modification procedures;
2. Proper configuration management procedures followed;
3. Re-run procedure causing initial failure and document results;
4. Review results with Test Manager for approval to proceed with original test plan, or re-test.

D Test Incident Report

The purpose of the test incident report is the documentation of any event that occurs during testing that requires investigation.

1. Identifier;
2. Summary
 - a. Test procedure reference;
 - b. Problem description and symptom;
 - c. Severity;
 - d. Function tested;
 - e. Test case;
 - f. Test items involved;
 - g. Test log reference;
 - h. Originator of incident report;
 - i. Description of repair and time estimate;
 - j. Related components;
 - k. Original observer of problem or symptom;
 - l. OPR for repair;
 - m. Resolution status (by whom, how long, description);
 - n. Resolution reviewer;
 - o. Test plans and procedures changed;
 - p. Scheduled retest date;
 - q. Date closed.
3. Incident description
 - a. Inputs;
 - b. Expected results;
 - c. Anomalies;
 - d. and time of incident;
 - e. Procedure step;
 - f. Environment;
 - g. Attempts to repeat;

- h. Testers;
 - i. Observers;
 - j. Action taken;
 - k. Attachments;
 - l. Recommendations.
4. Impact --describe the incident's effect on the testing process.

E Base Y2K Test Cases

1. Test setting and display of dates including, as appropriate;
 - a. 1900/2/29 --should fail, 1900 was not a leap year
 - b. 1996/2/29 --should succeed, 1996 was a leap year
 - c. 1999/12/31 --should succeed
 - d. 2000/01/01 --should be unambiguously represented
 - e. 2000/01/10 --first 7-digit date
 - f. 2000/10/10 --first 8-digit date
 - g. 2000/2/29 --should succeed, 2000 is a leap year
 - h. 2000/12/31 --should succeed and be identified as day 366
 - i. 2000/13/01 --should fail
2. Test processing of time-data with different data and time periods.
 - a. Using the current system clock, process key dates before and after 2000/01/01;
 - b. Set the system clock to 2000/01/01 or later, and process key dates before and after 2000/01/01.

F Systems Participating in SOST

Glossary

ICD	Interface Control Document
MOA	Memorandum of Agreement
SOS	System of Systems
SOST	System of Systems Test or System of Systems Testing