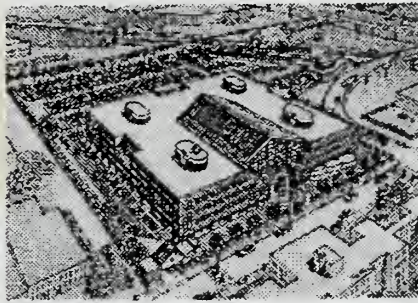


MASS. ITD 1.2: IT 2

UMASS/AMHERST



312066016588701



GOVERNMENT DOCUMENTS  
COLLECTION

SEP 01 1998

University of Massachusetts  
Depository Copy

# ITD Data Center

## AGENCY YEAR 2000

### TEST SUPPORT MANUAL

December 1997

Prepared for:  
Information Technology Division  
Data Center  
Massachusetts Information Technology Center  
Chelsea, MA

Prepared by:  
Information Technology Division  
Strategic Planning Group  
Year 2000 Program Management Office



Digitized by the Internet Archive  
in 2014

<https://archive.org/details/itddatacenterage00mass>

## TABLE OF CONTENTS

|          |   |            |
|----------|---|------------|
| <b>1</b> | <b><i>Introduction</i></b> _____  | <b>1</b>   |
| 1.1      | <b>Intended Audience</b> _____  | 1          |
| 1.2      | <b>Data Center Responsibilities</b> _____   | 1          |
| 1.3      | <b>Customer Responsibilities</b> _____  | 1          |
| <b>2</b> | <b><i>Standard Year 2000 Test Guidelines</i></b> _____                            | <b>3</b>   |
| 2.1      | <b>Types of Testing</b> _____   | 3          |
| 2.2      | <b>Test Planning</b> _____  | 4          |
| 2.3      | <b>Test Cases</b> _____   | 5          |
| <b>3</b> | <b><i>Data Center Year 2000 Testing</i></b> _____                                 | <b>6</b>   |
| 3.1      | <b>Environment</b> _____  | 6          |
| 3.2      | <b>Testing Protocols</b> _____  | 9          |
| 3.2.1    | Scheduling Process _____  | 9          |
| 3.2.2    | Notifications _____   | 10         |
| 3.3      | <b>Agency Activities</b> _____  | <b>10</b>  |
| 3.3.1    | Identify Source Code _____  | 10         |
| 3.3.2    | Verify Supported Software Versions _____  | 11         |
| 3.3.3    | Verify Interface Compatibility _____  | 11         |
| 3.3.4    | Modify Data Structures _____  | 11         |
| 3.3.5    | Verify Third Party Compliance _____   | 12         |
| 3.3.6    | Remediate Code _____  | 12         |
| 3.3.7    | Determine Access to History Files _____   | 12         |
| 3.3.8    | Plan for External Dates _____   | 12         |
| 3.4      | <b>Year 2000 Tools</b> _____  | <b>12</b>  |
| 3.4.1    | Date Simulation Tools _____   | 12         |
| 3.4.2    | Other Tools _____   | 13         |
| <b>4</b> | <b><i>Summary</i></b> _____   | <b>14</b>  |
|          | <b><i>Appendix A FY99 DASD Storage Requirements Letter</i></b> _____              | <b>A-1</b> |
|          | <b><i>Appendix B Compliance Status of Selected Data Center Software</i></b> _____ | <b>B-1</b> |

## TABLE OF FIGURES

|   |   |
|---|---|
| FIGURE 3-1 YEAR 2000 BUILDING BLOCKS          | 7 |
| FIGURE 3-2 PROCESSOR DRAFT SCHEDULE OF EVENTS | 8 |

# 1 INTRODUCTION

The Commonwealth of Massachusetts Information Technology Division (ITD) Data Center, located at the Massachusetts Information Technology Center in Chelsea, is currently preparing for Year 2000 testing. Agencies that have *mainframe* applications and are *current* customers of the ITD Data Center will need to coordinate their Year 2000 testing with the Data Center staff. The purpose of this document is to provide agencies with the information they will need to ensure a successful test program. This includes not only a description of the resources that the Data Center can provide for testing, but also guidance for the agencies in requesting and scheduling their testing and in developing and running their test programs. This Test Manual is intended to help both the Data Center and agencies to understand the test process and allow the Commonwealth to transition its major mainframe systems smoothly into the 21<sup>st</sup> century.

## 1.1 Intended Audience

This document is intended as a guide for current Data Center customers who:

- Anticipate the need for Data Center manual intervention during their Year 2000 testing (*e.g.*, mounting tapes, transferring files); and/or
- Need a means of changing the system date for future date testing.

## 1.2 Data Center Responsibilities

The role and responsibilities of the ITD Data Center at MITC for Year 2000 testing are:

- To guarantee that the operating system and system-related, third party supported software are Year 2000 compliant;
- To provide the physical test environment for Data Center customers;
- To coordinate scheduling of Year 2000 testing among all Data Center customers;
- To play an advisory role in customers' implementation, conversion, testing strategy, and selection of Year 2000 tools; and
- To provide a date simulation tool.

## 1.3 Customer Responsibilities

The Data Center customers are responsible for conversion activities related to their own applications, including: all code changes, clists, panels, screens, and copybooks; production data conversion; versioning of their programs; data structure needs; testing; and migration from test to production. For testing, Data Center customers are responsible for all testing from unit testing through full system testing, including supplying the test data and ensuring the integrity of their data. For non-database test users, customers should follow standard procedures for requesting back ups of their test data. If additional Year 2000 tools are needed, it is the customer's responsibility to contact the Data Center to request purchase and installation of the tools.

The Data Center staff is available to provide the same level of technical support which they currently provide, but it is the responsibility of the customers themselves to do the actual remediation and testing and to build any bridge software which may be required. It is also the responsibility of the customers to provide the Data Center staff with their anticipated schedules for testing and any specific requirements for the test environment so that the Data Center can develop a master schedule. Coordination of testing for all users, migration to the production environment, and upgrades to Data Center hardware and software will be necessary in order to ensure adequate test time for everyone. Test schedule and requirements information should be submitted to the Data Center staff soon, preferably by the end of February, 1998.

By now, agencies should have received a letter from MITC requesting current and future disk space requirements for their systems, including Y2K initiatives, for FY99, the period July 1998 through June 1999. Agencies should return these forms as soon as possible so that an acquisition for additional disk space, if necessary, can be processed quickly. See Appendix A for a copy of the letter.

Finally, when an agency requires Data Center testing services, they must be prepared to provide the answers to the following questions:

- ◆ What application is being tested?
- ◆ What is the system environment? (*e.g.*, database, VSAM, CICS, COMPLETE)
- ◆ Is disk space needed? If so, how much?
- ◆ Are tapes needed?
- ◆ What languages are being used?
- ◆ Are there any data security issues?
- ◆ Are there any data storage issues?
- ◆ What is the schedule for testing?
- ◆ Do any tests require changing the system clock?
- ◆ What, if anything, should the operator do in the event of an ABEND?

## 2 STANDARD YEAR 2000 TEST GUIDELINES

Year 2000 experts agree that testing for Year 2000 compliance is resource intensive and considerably more complex than what is required for a typical application development project. Estimates of the portion of a project's total resources (time and dollars) that should be devoted to testing in a Year 2000 project range from 35 to 60%. In fact, in Massachusetts both the Department of Revenue and the Department of Employment and Training have estimated that over 50% of their Year 2000 project resources are related to testing. In their pilot project, the Massachusetts Housing and Finance Agency actually devoted 69% of their project to testing although they had anticipated 48%. The Department of Revenue noted that the actual amount of resources needed for testing was double the initial estimates.

Unlike standard application development projects in which only the application itself is being tested, Year 2000 projects involve testing multiple layers of components. In Year 2000 projects, everything is potentially impacted by the rollover to 2000: operating systems, compilers, network software, database engines, and application code. This means that care must be taken in scheduling testing. Components must be individually subjected to rigorous testing to ensure Year 2000 compliance before they can be tested together as a system. This alone can add considerable time and expense to a Year 2000 project.

Unfortunately, most of the Year 2000 literature emphasizes the steps leading up to testing: awareness, inventory, remediation strategies, and tools. While these are important activities in a Year 2000 project, it can be argued that testing is the most important activity. Consider this statistic cited by Paul Gerrard in his paper, *Test Strategies for Year 2000 Projects*: "We estimate that programmers will, on average, introduce 100 errors per million lines of code in a migration. Studies tell us that poor testing detects only 30-40% of errors so we could expect to see between 60-70 errors per million lines of code. Just what will your customers or business users think if you introduced 600-700 errors in their system which consists of 10 million lines of Cobol?" Obviously, a sound test program is an important part of a Year 2000 project.

The remainder of this chapter describes best practices for Year 2000 testing.

### 2.1 Types of Testing

There are four types of testing required for Year 2000 testing: baseline testing, current date testing, and 2 types of future date testing. The first two types of testing are fairly standard and typically used for normal system maintenance. However, the third and fourth types of testing are more specific to Year 2000 remediation and typically require special software and/or a stand-alone processor so that the CPU dates may be set to 2000 and beyond.

#### ➤ Baseline Testing

Baseline testing requires the development of a test suite of functional tests for an application, running all test cases on the current system, and recording both intermediate and final results. This provides a reference point for the other types of testing.

#### ➤ Current Date Testing

After software remediation, testing of the application should be done using the current date and the test suite developed for Baseline Testing. This will verify that the changes made did not introduce other functional errors and that the application can handle current dates seamlessly. This testing should also include testing the date, September 9, 1999, since that date (9/9/99) was often used as a flag for other functions.

### ➤ Simulated Future Date Testing

This type of testing uses a Date Simulator tool to test applications using future dates. Dates to be tested should include:

- ◆ the rollover from December 31, 1999 to January 1, 2000;
- ◆ the rollover from February 28, 2000 to February 29, 2000,;
- ◆ the rollover from February 29, 2000 to March 1, 2000;
- ◆ the rollover from December 31, 2000, to January 1, 2001; and
- ◆ the rollover from February 28, 2001 to March 1, 2001.

Some agencies may have other significant dates which require testing. For example, accounting systems tests should include fiscal year rollovers:

- ◆ June 30, 1999 to July 1, 1999; and
- ◆ June 30, 2000 to July 1, 2000.

### ➤ Actual Future Date Testing

- ◆ This type of testing requires changing the system clock to test applications using future dates. Dates to be tested should include those listed above for Simulated Future Date Testing. Future date testing requires special handling by Data Center staff. Because the system clock is changed, only one group can test at a time, or multiple agencies must coordinate their system testing to use the same system date. This type of testing is further complicated by the fact that certain system files are populated with the system date. Therefore, after testing, it is not sufficient to just roll back data; the environment itself must be restored in order to set up for testing other systems or other future dates.

## 2.2 Test Planning

It is assumed that, at this point, agencies are aware of the need for test planning and documenting the test plan. As a reminder, the following is a partial list of activities that should occur prior to the start of formal testing:

- Identify criticality of each system;
- Identify functional and performance requirements for each system;
- Identify interfaces;
- Assess the current test environment for systems being remediated;
- Perform baseline testing; and
- Perform unit testing.

The test plan should include system testing, integration testing, performance testing, and stress testing. It is very important to develop a test schedule since Year 2000 testing involves a multitude of components being converted along different timelines. Furthermore, because all of the systems at the Data Center will have to undergo rigorous testing within a limited time, it is critical that agencies submit a documented schedule of testing that involves future date changes. The Data Center then can coordinate the testing among all the agencies and effectively manage the availability of the test environment.

A well documented test plan will include not only a schedule for testing but also descriptions of test cases (see Section 2.3 below), bridge software required, test environment requirements, assumptions and constraints, test procedures, and a problem reporting and correction system. If the test team includes a large percentage of vendors or new hires, it is also helpful to include the organization of the test team and a discussions of roles and responsibilities

## 2.3 Test Cases

Identification of test cases is potentially a very time consuming task. It is important to recognize the effort involved. The following example demonstrates how a Cobol application with a million lines of code may require a minimum of 45,966 test cases. This estimate was presented at a recent Year 2000 Day by EDS.

Assume a 1,000,000 Cobol application and further assume:

- ➔ 1500 L.O.C. per program
- ➔ 150 screens
- ➔ 50 reports
- ➔ 4% or 40 impacts
- ➔ 666 programs (85% affected)
- ➔ 200 batch transactions
- ➔ 500 files

Then:

| Base Cases   | Year 2000 Test Cases   |  |   |
|--|--|--|---|
| 400  | 566  | 5,566  | 45,566  |
| <ul style="list-style-type: none"> <li>• screens</li> <li>• reports</li> <li>• batch transactions</li> </ul> | <ul style="list-style-type: none"> <li>• programs</li> </ul> | <ul style="list-style-type: none"> <li>• programs</li> <li>• 10 I/Os per file</li> </ul> | <ul style="list-style-type: none"> <li>• programs</li> <li>• 10 I/Os per file</li> <li>• affected L.O.C.</li> </ul> |

Keep in mind that one of the purposes of testing is to minimize risk. Measurable, repeatable test cases will help to ensure an efficient and effective test program. In addition, tests of the most important or mission critical functions should be planned and performed first so that less important tests can be deferred if time becomes an issue.

For a very good description of test case design for Year 2000, see Paul Gerrard's article at <http://www.fttech.net/~evollutif/Y2K/techs.html>

## 3 DATA CENTER YEAR 2000 TESTING

Testing is a process that starts with activities performed early in a Year 2000 project and continues throughout the life of the project. The ITD Data Center at MITC has started their portion of the testing process by:

- compiling and continually updating a list of vendors and the Year 2000 compliance status of their products,
- installing and testing new releases of software as Year 2000 compliant versions become available, and
- creating a test environment for use by their customers.

The purpose of this chapter is to describe the new environment that has been established for testing, activities that agencies should be focussing on now prior to testing, and important information that Data Center customers will need for testing.

### 3.1 Environment

The Data Center at MITC has been preparing a Year 2000 test environment which is currently scheduled for completion by the end of FY 1998. This date depends upon the Year 2000 compliance status of the hardware and software in the new environment. Appendix A provides a partial list of software in use at the Data Center and their Year 2000 compliance status as of November 1997. This list will continue to be updated as new information is received.

Figure 3-1 shows the Building Blocks for the Data Center Test Environment and the compliance status of the blocks. Note that the first seven levels are the responsibility of the Data Center staff and that the top level is the responsibility of the Data Center customers. A more detailed description of the top level is provided in Section 3.2 below.

The Data Center has a CMOS processor that is Year 2000 certified. Figure 3-2 shows the draft schedule of events for the configuration and readiness preparations for this processor. The events are slated to be completed in four phases with everything running under OS/390 no later than June 30, 1998.

|          |  |  |
|----------|--|--|
| <b>H</b> | <b>APPLICATIONS</b>  | Users responsible for converting programs to support new dates or upgrading to higher level software. Data structures may have to be changed.<br><br>DCSB is responsible for IBM and 3rd Party Vendors Software Upgrades |
| <b>G</b> | <b>DATABASES</b><br>ADABAS                      DB2  |  |
| <b>F</b> | <b>PROGRAMMING LANGUAGES</b><br><br>COBOL/VS (Not Supported as of July 1,1994)<br>COBOL II                      NATURAL<br>PLI                                  FORTRAN<br>ASSEMBLER |  |
| <b>E</b> | <b>TELEPROCESSING MONITORS</b><br>CICS                      COMPLETE<br><br>TSO  |  |
| <b>D</b> | <b>COMMUNICATIONS SOFTWARE</b><br><br>VTAM    NCP    TCP/IP<br><br>CANDLE Gateway  |  |
| <b>C</b> | <b>OPERATING SYSTEMS</b><br>MVS/ESA  |  |
| <b>B</b> | <b>MICROCODE</b>   |  |
| <b>A</b> | <b>Hardware</b><br>AMDAHL    HDS    IBM<br>EMC    STK  |  |

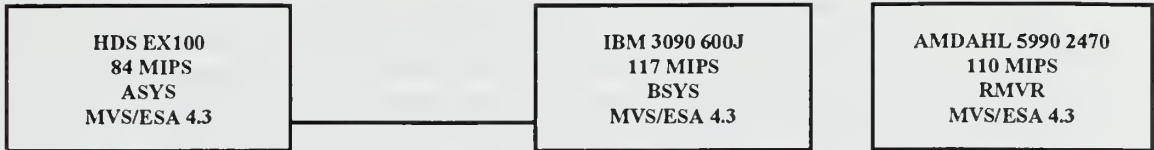
**A & B Year 2000 Ready**

- C** Operating System has to be upgraded to MVS/ESA 5.2
- D** VTAM, NCP & TCP/IP are Ready. CANDLE Gateways will be ready next release.
- E** TSO & COM-LETE are Ready. CICS needs to be upgraded to CICS V4.1.
- F** PLI & ASSEMBLER are READY. COBOL II will have to be converted to COBOL for MVS. FORTRAN will have to upgrade to Version 2.5. NATURAL will have to be converted to Version 2.3.11.
- G** DB2 is Ready. ADABAS needs to be converted to Version 6.2.
- H** The User Departments will have to evaluate there own situation. Anyone that has planned ahead will only have to wait until all software products have become 2000 ready.

**FIGURE 3-1 YEAR 2000 BUILDING BLOCKS**

**Current Images**

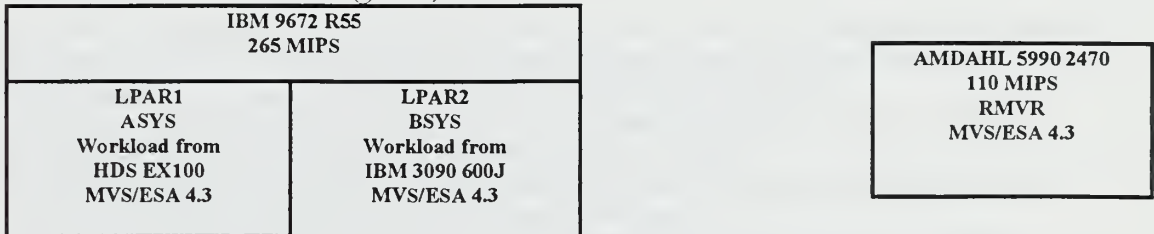
Apply PTF's for new CMOS Processor



**Phase One**

Install @ Nov 1,1997

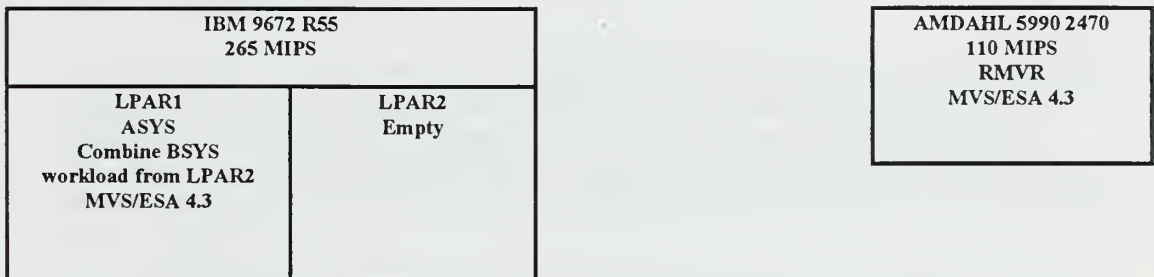
Apply any PTF's needed for



Complete connection of systems by Nov 29,1997

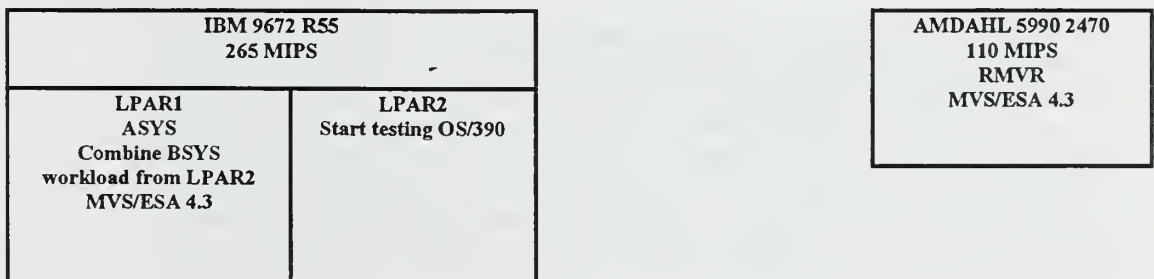
**Phase Two**

Migrate two Images into one by Dec 28,1997



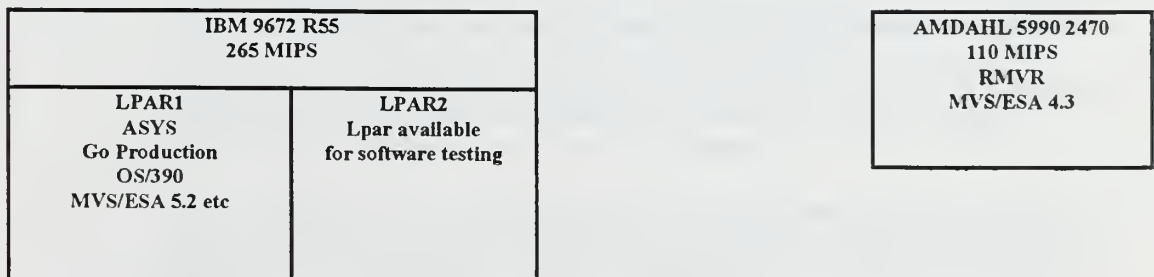
**Phase Three**

Jan 1,1998 Start testing applications under OS/390



**Phase Four**

Everything running under OS/390 by June 30,1998



**FIGURE 3-2 PROCESSOR DRAFT SCHEDULE OF EVENTS**

## 3.2 Testing Protocols

The importance of and effort required for Year 2000 testing cannot be emphasized enough. As noted previously, testing consumes approximately 50% or more of a Year 2000 project's resources. Agencies can compile programs, do unit testing, and current date testing without Data Center involvement, much as they do today when modifying programs or developing new systems. However, to perform their system testing with future dates, agencies will need to coordinate with the Data Center staff. This means providing Data Center Operations staff with a test schedule and test requirements, *as soon as possible*. Other important elements that Data Center staff wish to pass along to their customers:

- There will most likely be a lot of free time on non-prime time shifts; since there are limited cycles and disk space, customers should plan to do some testing in non-prime time (balance the load!)
- Customers should be prepared to work nights and weekends during testing!
- Customers should be prepared to reiterate testing. As underlying software reaches compliance, customers may experience glitches that could require some retesting.
- When agencies are ready to migrate a system from test to production, they should keep in mind that existing Data Center standard policies and procedures will apply.
- Agencies with large applications should not plan to migrate from test to production **in a single weekend**. A phased approach should be used (and documented) to move applications into production.

### 3.2.1 Scheduling Process

A major concern of the Data Center staff is scheduling of future date testing. There are obviously limited resources in regard to disk space and time for testing. In addition, because this type of testing involves date changes, Data Center staff will have to support a significant amount of stand-alone testing, *i.e.*, the testing of only one system at a time. Furthermore, all types of testing will need to be coordinated with the installation of Year 2000 compliant software and hardware in the test environment itself.

The Data Center will use a process similar to their existing change control process for Year 2000 testing. The process is described below:

Each agency that develops or maintains applications on the Data Center mainframe should currently have a System Administrator that acts as a liaison between the agency and the Data Center. The System Administrator is a single point of contact within the agency for all functions related to system development including, but not limited to, databases, application code, file conversions, control language, and testing. If an IS staff person needs support from the Data Center, for example, to make a change to the database structure, the request must be coordinated with the System Administrator. The System Administrator will then contact the appropriate person at the Data Center with the request.

Coordination of testing will be handled the same way. When a developer at an agency needs to test code, for example, the developer will contact the System Administrator, who, in turn, will contact the appropriate person at the Data Center. At the weekly Change Control meeting, the key Data Center staff will review all change requests and schedule the changes. If there are conflicts, Data Center staff will work with the agencies' System Administrators to negotiate resolution.

System Administrators will request changes via the existing on-line Change Request Form. The Change Request Form includes information such as department, change type, system, type and description of change, operator instructions, requirements, requestor, and comments.

For Year 2000 testing, agencies that are performing baseline or current date testing can do their testing without the Change Control process if no operator intervention (i.e., no manual tasks) is involved. The change control process is required for the final steps of testing prior to migrating Year 2000 compliant applications to production.

### 3.2.2 Notifications

Only current mainframe customers of the Data Center will be authorized for Year 2000 testing there. It will be necessary for agencies to notify the Data Center of their Year 2000 testing plans if the agency will require:

- a system IPL date to test with;
- file transfers from production to the test environment;
- operator intervention in the event of an ABEND;
- modifications to the database structure;
- special DASD;
- tapes mounted/dismounted; and/or
- other operator intervention not listed above.

Agency System Administrators should submit their schedules and requirements for testing to the appropriate Data Center point of contact as soon as possible. This information should include:

- Timeline for testing – This is absolutely critical since there will be a number of agencies requiring the use of the test environment and the services of the Data Center staff. In order to accommodate everyone, coordinate the testing with software upgrades, and coordinate testing of interfaces, the Data Center staff will need to know planned test schedules for all agencies so that a master test schedule can be drawn up.
- Disk space – It is expected that some agencies will test only small systems or use small amounts of data. For these agencies, disk space will not be an issue. However, other agencies will need to test their whole system which will require mirroring their entire database. For these agencies, this may be a problem. Therefore, the Data Center needs to be notified as soon as possible.
- Special data storage requirements – If users are going to need additional data storage devices, such as tapes, the Data Center should be notified in advance.
- Other requirements – other requirements noted above should also be included in an agency's request for testing.

## 3.3 Agency Activities

Agency testing activities should begin long before the actual testing is scheduled to take place. There are a number of activities that agencies can and should be doing *right now*. The following is a list of activities that agencies should do before testing begins. Note that it is assumed that agencies are aware of all of the steps involved in a Year 2000 project and that the activities listed below are those that directly impact the Data Center and its customers (Chapter 3 of *Year 2000, Meeting the Challenge*, Second Edition, contains a detailed description of the steps involved in a Year 2000 project.) It is also assumed that all agencies testing at the Data Center will develop a schedule and set of requirements for testing.

### 3.3.1 Identify Source Code

Pre-testing activities obviously include an inventory of all applications, associated software units, and information about the units. An important part of this inventory is to make sure that the source code for all software modules is available and that it matches the current executable code. To test this, agencies should compile and run their existing source code, run selected test cases with both the compiled source code and the existing executable and compare the results of the

tests. If the source code is missing or does not match the executable code, then the first step toward testing is to recreate the source code to match the executable.

### 3.3.2 Verify Supported Software Versions

As a related step, agencies should verify that their software builds use system software, (e.g., compilers, etc.) at versions which are supported by the Data Center. This is important because most users are running down-level releases and do not realize it. Users should look at their compile JCL to determine the level of the software being used. This should be compared to the list of software in Appendix A. If the level of software an agency is using is not on the list, then the applications must be recompiled with a compliant version. For example, if an agency is using Natural for DB2, then they must verify that they are using Version 2.4.1. If not, they must upgrade to that version. The version of the software must also support Year 2000 dates, as indicated in Appendix A. But, Data Center customers must also keep in mind that there are external and internal changes for the new releases that may require customer action. Some users may be able to upgrade to a supportable version of the software as part of their remediation effort.

### 3.3.3 Verify Interface Compatibility

In addition to ensuring that software versions are supported by the Data Center, it is also important for agencies to verify that the software versions used for creating and receiving interface data is compatible with the receiving or sending application. This consists of two checks: first, it is necessary to know when either a sending or receiving application has changed the format of any date data which is transferred. The most common example is a system which sends data to receives data from MMARS. Both applications, MMARS and the sending/receiving application must be using the same interface structures, in particular, the same size year field and the same format. Another, less common, check is for applications which call modules written by another organization or written in another language. For example, if Agency 1's application calls a calculation module from MMARS, then Agency 1 must ensure that the module has been compiled in the same version of the software as MMARS. Otherwise, there will be an incompatibility and the compilation will fail.

### 3.3.4 Modify Data Structures

For applications using expansion for their remediation strategy, agencies should coordinate their data structure modifications with their Data Base Administrator (DBA) at the Data Center. Agencies should also compute any DASD that may be needed to accommodate the expanded date fields. Requests for additional DASD should be submitted to Paul Carrick as soon as possible.

Note that ITD has specified a standard format for dates, as follows:

The Commonwealth of Massachusetts is adopting the international standard that uses a four-digit year where a two-digit century precedes, and is contiguous with, a two-digit year-of-century (e.g. 1999, 2000, etc.) for the purpose of electronic data interchange among its agencies. The international standard date notation is CCYYMMDD where CC=century, YY=year, MM=month and DD=day. For example, August 4, 1997 would be "19970804"; February 29, 2000 would be "20000229". Commonwealth agencies are strongly advised to use this notation for date representation.

This standard is also published at ITD's web site <http://www.state.ma.us/Y2K> and is also available in their resource document, *Year 2000, Meeting the Challenge*.

### 3.3.5 Verify Third Party Compliance

Some agencies that use the Data Center run applications which have been created and are maintained by third party vendors, *e.g.*, one agency uses software created and maintained by John Hopkins University. Agencies in this situation should contact each third party vendor *as soon as possible* and request, in writing, the status of the application's Year 2000 compliance and the vendor's schedule for releasing any required upgrades.

### 3.3.6 Remediate Code

It is the responsibility of each agency to remediate its application source code. This should include moving test libraries, compiling the code, and performing unit tests.

### 3.3.7 Determine Access to History Files

An area of concern to the Data Center staff is history, or archive, files. Agencies must determine how their converted systems will access historical data. For agencies which seldom or never access this data, it may be sufficient to do nothing. However, a number of agencies have requirements to report data to the Federal Government spanning 5- 7 years. These agencies will have to decide how to handle any historical data that is non-compliant. There are two options: all historical data can be converted so that it will be compatible with the Year 2000 compliant application; or, a 'bridge' program or subroutine can be developed to correct the data when it is needed for reporting. Since the Data Center operates on a payback system and users are charged based on, among other things, data storage, cost should be a consideration in determining how to handle history files. If an agency expects frequent requests for old data, it will most likely be more efficient to convert the data. However, if an agency expects few, if any, requests for history data, writing a bridge program when it is needed may be more appropriate. In either case, Data Center customers should include history data in their test plans.

### 3.3.8 Plan for External Dates

While agencies can monitor and control their own interfaces, they will not have control over external data coming from other organizations. This can be a serious problem since it has the potential for corrupting existing, compliant data and systems. Agencies must be aware of this and plan for it. If necessary, agencies will need to develop and test bridge programs that will validate the data coming into an application. If the interface data has not been made Year 2000 compliant, then the agency must develop a bridge program to convert the data to a Year 2000 compliant format.

## 3.4 Year 2000 Tools

A number of tools are available commercially to assist organizations with their Year 2000 project. The Data Center has no plans at this time to purchase any tools, (except date simulation tools). Agencies wanting additional Year 2000 tools must coordinate the purchase with the Data Center.

### 3.4.1 Date Simulation Tools


Two methods of testing applications using future dates (*i.e.*, the years 1999, 2000 , and beyond) are being considered by the Data Center for their test environment.

One method is to change the date via IPL statement. If this approach is used, agencies will need to notify the Data Center in advance so that Data Center staff can make the requested date change. This will provide a test environment where the system date is already a future date. A drawback to this approach is that only one agency at a time will be able to test.

A second method of testing applications using future dates involves the use of a date simulator tool. Typically a date simulator tool intercepts calls to the system date. The tool allows users to specify new dates and times to be used as the system date. More than one user at a time may test using one of these tools, even if the users are testing with different dates.

The Data Center is in the process of evaluating date simulation tools. Unfortunately, most, if not all, date simulators are not all-inclusive in terms of languages, operating systems, and platforms with which they interface. It is important for Data Center customers to understand that if a date simulator tool is selected, it may not be all-inclusive. No date simulator tool has been selected yet and, in fact, it may be that more than one tool will need to be purchased. The Data Center will purchase as many tools as it determines is necessary.

Additional information regarding the future date issue and the Data Center decision will be published as it becomes available.

 **NOTE:** If the Data Center supplies date simulation tool(s), it may require that its customers pass their Year 2000 tests using a date simulator before moving to testing with an actual internal clock change.

### 3.4.2 Other Tools

In addition to date simulation tools, a number of other tools for Year 2000 projects have been developed and are on the market. Types of tools which may be purchased include:

- Inventory Tools
- Scanners and Assessment Tools
- Remediation Tools
- Testing Tools, for record and playback, data aging, and file comparisons
- Configuration Management and Test Deficit Tracking Tools

Any agency considering the use of these tools must request that the Data Center purchase and install the tools. The request should include the purpose of the tool and justification of the need for it. The Data Center will review all requests and will work with agencies by either purchasing and installing requested tools or suggesting alternate ways to accomplish what the tool is meant to do.

## 4 SUMMARY

The ITD Data Center at MITC is preparing an environment where agencies can test their remediated code for the Year 2000. In preparation for such support, a review of date simulation options is in process and third party packages are being upgraded and/or replaced to achieve Year 2000 compliance.

From now through the end of the fiscal year (when the Data Center will be ready for agencies to begin their testing), there are a number of activities that agencies can and should be doing now to prepare for the testing. These include:

- ◆ Development of a test plan, including schedule, test cases, and testing requirements;
- ◆ Submitting test schedules and requirements for disk space, storage requirements, and data center operations support to the data center by February 1998;
- ◆ Identification of current source code and data files;
- ◆ Verification that system support software used for conversion are versions supportable by the data center;
- ◆ Determination of bridges required for testing/remediation;
- ◆ Modification of data structures, remediation of code, and conversion of data;
- ◆ Contacting third party vendors to determine Year 2000 compliance of their product(s); and
- ◆ Determination of bridges required for testing/remediation.

The Data Center asks that agencies submit their test schedules and requirements as soon as possible, preferably by February 28, 1998, so that the Data Center may begin to develop its master plan. Agencies should try to include some testing in non-prime time, which means planning to work some nights and weekends. In addition, agencies should plan on performing initial future date testing using a date simulator; testing with the internal clock physically changed will not occur until tests with a date simulator have been successfully completed. Once testing is complete, agencies should plan to follow the Data Center standard procedures for moving applications from test to production. Agencies with large systems should not plan to migrate to production in a single weekend, but should plan a phased migration.

The Data Center staff is there to help – agencies should not hesitate to contact their Data Center representative with any questions regarding Year 2000 testing and the ITD Data Center at MITC.

## **Appendix A FY99 DASD STORAGE REQUIREMENTS LETTER**



**The Commonwealth of Massachusetts  
Executive Office for Administration and Finance  
Information Technology Division**

200 Arlington Street Suite 2100 • Chelsea • Massachusetts • 02150

ARGEO PAUL CELLUCCI  
ACTING GOVERNOR  
CHARLES D. BAKER  
SECRETARY  
T. LOUIS GUTIERREZ  
CHIEF INFORMATION OFFICER

Telephone: (617)660-4441  
Facsimile: (617)660-4407

**TO: Chief Information Officer**

**FROM: Paul E. Carrick, Technical Services Unit**

**SUBJECT: FY99 MAINFRAME DASD STORAGE REQUIREMENTS**

Mainframe DASD equipment and related storage management services are a significant expenditure for the ITD data center as well as a considerable chargeback cost component for customer agencies. The procurement of DASD equipment is closely related to customer feedback information about their agency's new and on-going projects and to historical demand (allocated DASD space) data collected by the data center. It is essential that ITD have sufficient DASD storage available to meet the needs of its customers' mission critical applications. To meet this objective, ITD must be aware of customers' plans for the next fiscal year.

The ITD data center is collecting the FY99 mainframe DASD storage requirements for the July 1, 1998 to June 30, 1999 time frame, for new DASD space. New DASD space is defined as additional space that is required to support the agency and does not include space that is currently allocated (being used). The DASD space requirements should include estimates for new and planned development projects, maintenance activities, testing, and production growth. Remember to include test and production DASD space requirements for the year 2000 projects. The DASD space requirements may be reported in tracks, cylinders, megabytes, or gigabytes.

Do not include requirements for ADABAS and DB2 DASD space. The data base administrator, who supports your agency, will contact your agency to discuss future plans and objectives. The Database Technology Group will submit the ADABAS and DB2 DASD space requirements for your agency.

**If your agency will require additional DASD space, please submit your estimates NO LATER than January 23, 1998.** If your agency will not require additional DASD space, no reply is required. The estimates can be mailed, faxed, or e-mailed. Please note

that the estimates can be adjusted anytime during the year. Requirements that involve substantial increases in DASD storage should be submitted to the data center as soon as possible.

Dasdfy99.doc

**Appendix B   COMPLIANCE STATUS OF SELECTED DATA CENTER  
SOFTWARE**

## Data Center Software

| IBM Products            |                      |        | Year2000                    | Year2000                | New       | Product Name  | Sched   |
|-------------------------|----------------------|--------|-----------------------------|-------------------------|-----------|---------------|---------|
| Product #               | Software             | Level  | Ready?                      | Release Needed          | Product # |               | Upgrade |
| 5688-216                | AD/Cycle C/370       | 1.1.0  | NO                          | V1R2                    |           |               | 2QTR98  |
| 5688-197                | COBOL for MVS        | 1.1.0  | NO                          | V1R2                    |           |               | 2QTR98  |
| 5688-194                | AD/Cycle<br>CODE/370 | 1.1.0  |                             | V1R2                    |           |               | 2QTR98  |
| 5688-198                | LANG ENV             | 1.1.0  |                             | V2R2                    | 5688-188  | C/370 Library | 2QTR98  |
| 5655-121                | C/C++                | 3      |                             | V3R2                    |           |               | 2QTR98  |
| 5665-403                | CICS V2.1            | 2.1.2  | NO                          | Convert to CICS<br>V4   |           |               |         |
| 5668-958                | COBOL II             | 4      | NO                          | Convert to COBOL fo MVS |           |               |         |
| 5740-XC5                | DMS/CICS             | 4      |                             | V1R5                    |           |               | 2QTR98  |
| 5740-CB1                | Cobol                | 2.4.1  | Not Supported as of July 97 |                         |           |               |         |
| 5655-018                | CICS/ESA V4          | 4.1    | Yes                         | V4R1 OK                 |           |               |         |
| 5748-FO3                | Fortran              | 1.4.1  |                             | V2R5                    | 5668-806  | Fortran       | 2QTR98  |
| 5668-909                | OS PL/I V2           | 2.3.0  | Yes                         | V2.3.0                  |           |               |         |
| <b>Non IBM Products</b> |                      |        |                             |                         |           |               |         |
|                         | Natural              | V2.2.8 | No                          | V2.3                    |           |               | 4QTR98  |
|                         | ADABAS               | V5.3.3 | No                          | V6.2                    |           |               | 1QTR99  |
|                         | COM-LETE             | V4.6.3 | No                          | V5.1                    |           |               | 4QTR98  |

