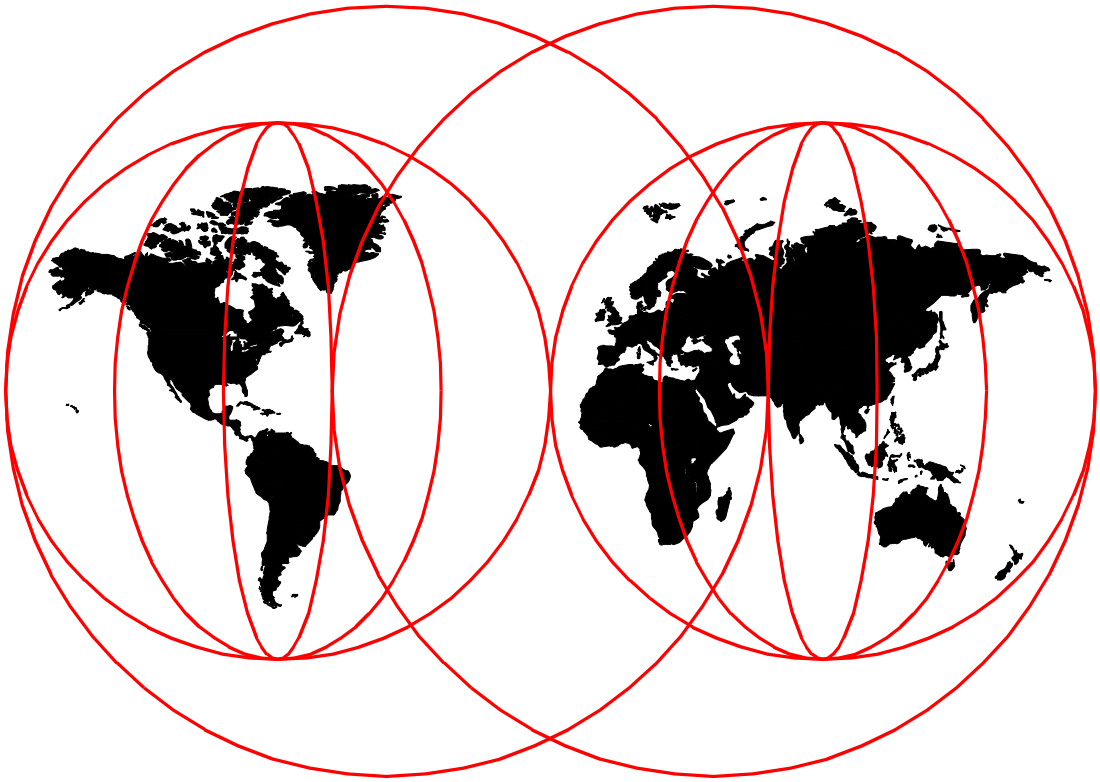
AS/400 Applications: IBM Year 2000 Tools Tips and Techniques

Brian R. Smith, Simon Ho, Michele Martiradonna



International Technical Support Organization

http://www.redbooks.ibm.com

SG24-2156-01

International Technical Support Organization

SG24-2156-01

AS/400 Applications: IBM Year 2000 Tools Tips and Techniques

June 1999

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix G, “Special Notices” on page 183.

Second Edition (June 1999)

This edition applies to Version 3 Release 1 Modification Level 2 of BYPASS2000 (Program Number 5697-D11) and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure that you are using the correct edition for the level of the product.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. JLU Building 107-2
3605 Highway 52N
Rochester, Minnesota 55901-7829

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Tables

Preface

This redbook is the third in a series of three redbooks that provides practical guidance to AS/400 application developers to enable their applications for the approach of the next century. Through the use of tutorials and a step-by-step hands-on approach, you are guided through the features and capabilities of IBM tools and products, such as BYPASS2000 V3R1M2, for a rapid increase in user knowledge and awareness.

The book begins by using basic tools, looking at simple problems, such as "Does my source code match my compiled code?". Then, it explores sophisticated products that automate much of the Year 2000 application enabling process.

This redbook is available only in softcopy. You can access this redbook on the Web at: http://www.redbooks.ibm.com/

In the Redbooks Home page, click **Redbooks Online!** On the page that appears, enter **SG24-2156-01** for the search string and then click **Submit Search**.

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world coordinated from the International Technical Support Organization Rochester Center.

Simon Ho is a Customer Support Specialist with HAL North America in Toronto, Canada. Simon has received intensive training at HAL headquarters, and has been working closely with the Search2000 and Bypass2000 development team in Milan, Italy. His main areas of interest are RPG, COBOL and SQL programming. Prior to joining HAL, Simon worked for some time as an Information Technology Specialist at the University of Toronto. Simon holds a B.S. in Computer Science from the University of Toronto.

Michele Martiradonna is a Technical Advisor and Manager of Year2000 Products with HAL North America based in Toronto, Canada. Part of the original development team for Search2000, Michele has since joined the team at HAL North America, providing education and support on Year2000 issues to the North American market. He has taught Bypass2000 and Search2000 classes in Europe and in North America, provided skill transfers to the ITSO in Rochester, and has worked on several Year2000 projects in the U.S. and Canada. Michele holds a Ph.D. in Environmental Engineering from Politecnico di Milano, Italy.

Brian R. Smith is a Senior AS/400 Specialist in the International Technical Support Organization in IBM Rochester. The first half of his career was spent in design, coding and testing on the System/38 and AS/400 systems in the area of communications. He then moved into technical marketing support in 1990 to pursue teaching and writing. You can reach Brian on the Internet at: brsmith@us.ibm.com

Thanks to the following people for their invaluable contributions to this project:

Bypass2000 Support Team, HAL corporate offices in Milan, Italy

A special thanks to:

Enrico Franchi, Year2000 Support Manager
Elena Castoldi, Customer Support Specialist
Laura Borri, Customer Support Specialist

Thanks also goes to the authors who wrote the first edition of this redbook, which was published in December 1997. Their redbook served as a basis for this project.

Veronica Chatfield, IBM UK
Susan Gantner, IBM Canada
Patricia Gianella, IBM Peru
Doug Jenkins, IBM USA
Jim Kaminski, IBM USA
Jon Paris, HAL North America
Paolo Tedone, HAL Italy
Neil Willis, IBM USA

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in “ITSO Redbook Evaluation” on page 191 to the fax number shown on the form.
- Use the online evaluation form found at `http://www.redbooks.ibm.com/`
- Send your comments in an internet note to `redbook@us.ibm.com`

Chapter 1. Introduction

This chapter gives an overview of the Year 2000 challenge for the AS/400 system and provides details about the tools available from IBM to help with the issues that can arise from a legacy of two-digit dates.

1.1 The Challenge

The Year 2000 is fast approaching. There has been a lot of speculation about what may happen, such as elevators will stop working, planes will not fly, and pacemakers will cease to tick. Whether this is correct, the challenge is real and serious. All businesses (large and small) need to consider the impact.

The "Year 2000 Challenge" is not one challenge, but several. There is the consideration of computer hardware and internal clocks, but the most important consideration is that of applications that have traditionally used only two digits to represent a year (for example, 10/08/97 in DDMMYY format represents August 10, 1997).

1.1.1 Hardware

Most computer hardware contains a system clock. Personal Computers (PCs) contain a Basic Input-Output System (BIOS) that uses the stored date and time. The hardware timer is the key component in determining the date and time for an operating system or application.

The timer is initialized with a value based on a time and date and in increments at a sub-second factor. This value is then converted to a date and time, and returned back to the operating system or application. Depending on the size of the timer and the conversion method used, errors may result once the year is 2000 or beyond. The date and time are accessible by the operating system and applications. However, when the date changes from December 31, 1999 to January 1, 2000, different results can be obtained. All models of the AS/400 system contain a hardware timer that maintains the correct date and time, including the switch between centuries. However, this is only true if the operating system installed on the AS/400 system is Year 2000 ready (OS/400 Version 3 Release 2 Modification 0, Version 3 Release 7 Modification 0, and later). This change will also occur correctly only if the system is either powered off or on at the time the year changes (that is, not in the process of being powered up, referred to as an IPL).

1.1.2 AS/400 Applications

For the AS/400 system, the main issue arises from the fact that many applications use two-digit dates. There is ambiguity when the system year passes 1999 because the computer does not know what this represents (for example, does 98 mean 1898, 1998, or even 2098). This is a real issue that cannot wait until the year 2000. In fact, it has occurred in many situations already, such as a computer mistakes a person over the age of 99 to be a baby.

The second part of this challenge is the incorrect sequencing of dates. This occurs in situations where purchase orders are sequenced by date (the orders to process that day are at the top of the list).

The other part of the challenge is the addition and subtraction of dates. For example, a simple stock check program may do a calculation by subtracting the current date from the expiration date to ensure it is valid. In a real case, the expiration date may be August 10, 2000 (some tinned products can quite easily be set several years in advance). If the current date is October 6, 1997, the calculation results in 001008 minus 971006, which is a negative number. To pass the product as still current, the result of this calculation must be positive. Therefore, this product is marked as expired.

1.1.2.1 Leap Years

The following diagram shows the correct sequence for a leap year calculation.

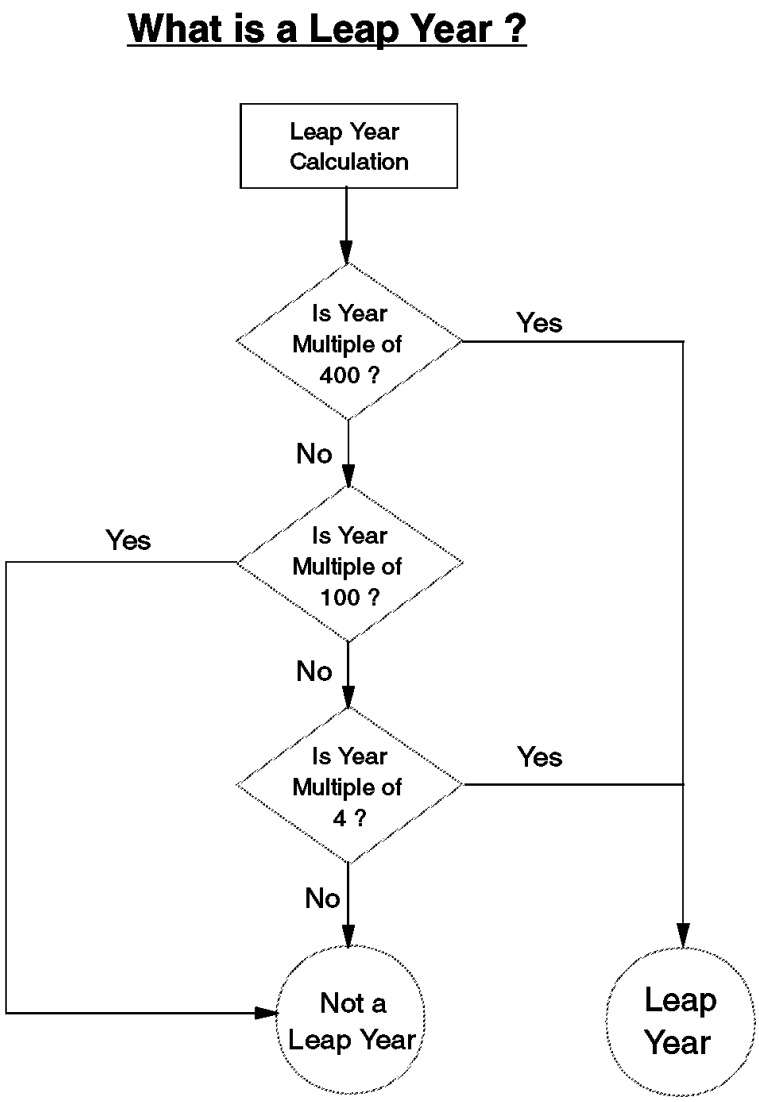


Figure 1. Leap Year Calculation

Typically, AS/400 applications test for a leap year by dividing a two-digit year by four and examining the remainder (a remainder of zero indicates it is a leap year). By following the flow chart, you can see that a year of 00 is flagged as a leap year. This is correct for the year 2000 but not for 2100. Therefore, leap year calculations should not pose a problem for the year 2000. However, it is worth checking the validity of your calculations to ensure all the checks are performed because some logic tests omit the first and second stage (multiples of 400 and 100).

1.1.2.2 Hidden Dates

Hidden dates are dates that are embedded in other fields. For example, invoice numbers sometimes use the last (or first) two digits of the date as the first two digits of the number. This poses a challenge if these orders are sorted into descending order as 2000 appears after 1997.

1.1.2.3 Invalid and Special Date Values

Many applications use certain values for an invalid date entry. This may be all zeros for an invalid year entry. Therefore, to view all orders due for delivery in the year 2000 with only a two-digit field entry, a value of 00 may flag as an invalid entry unless the logic is changed. A date entry of 99 can also have another meaning (for example, any orders with a due date of 99 are out of stock).

1.2 What Needs to Be Done

For the AS/400 system, there are two issues. The first is the operating system, OS/400 and Licensed Program Products (LPPs) and the second is the applications. OS/400 V3R2M0 (for 48-bit CISC (Complex Instruction Set Computer) AS/400 systems) and OS/400 V3R7M0 (for 64-bit RISC (Reduced Instruction Set Computer) AS/400 systems) and subsequent releases are Year 2000 ready. The details as to which versions of OS/400 and associated LPPs you must have to be Year 2000 ready can be found in *Make Your AS/400 System Year 2000 Ready,* SG24-5401, available only online at:
http://www.redbooks.ibm.com/

In the Redbooks Home page, click **Redbooks Online!** On the page that appears, enter **SG24-5401** for the search string and then click **Submit Search**.

A Year 2000 ready OS/400 release, ensures that:

- The command-to-command processing program (CPP) interface is CYYMMDD.
- Output files are date qualified with CYY or YYYY.
- Application Programming Interfaces (APIs) are date qualified with CYY or YYYY.

Note: The "C" denotes the century digit to indicate if the year is the 20th or 21st century.

IBM no longer provides support for Version 2 of OS/400. Therefore, we are unable to guarantee the results of any systems that are using Version 2 (or prior releases) when run in the Year 2000.

The majority of OS/400 licensed programs on systems at V3R2M0 or V3R7M0 and later are Year 2000 ready. A complete list of these can be found in the IBM Year 2000 Product Readiness Database found at:
http://wwwyr2k.raleigh.ibm.com/

The following flow chart shows the stages required to address this challenge.

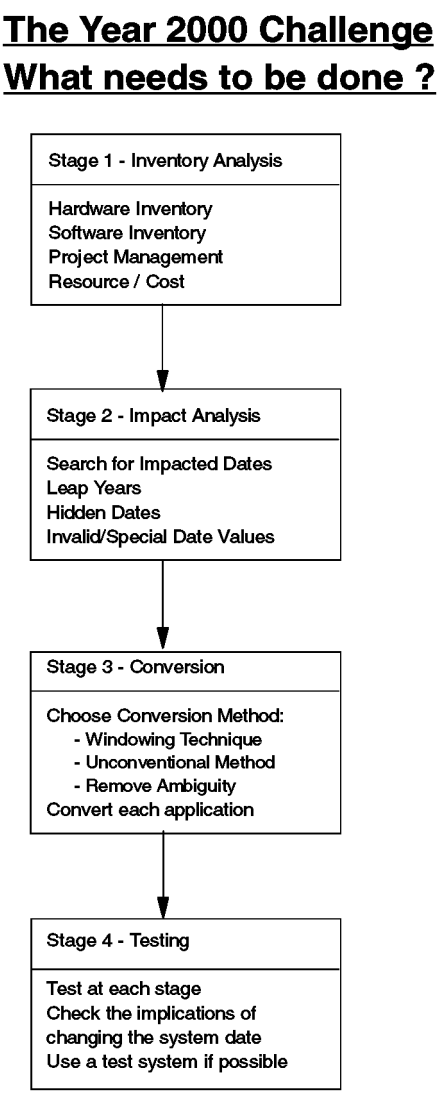


Figure 2. Stages When Addressing the Year 2000 Challenge

1.2.1 Resource

Each organization needs to develop a formal plan to resolve the year 2000 challenge. First, all resource must be allocated in terms of a project team consisting of not only the IT staff (as this affects the entire organization), but also management and staff from other departments.

1.2.2 Software Inventory

Having an idea of the scope of the project is the first stage. The Application Inventory Tool (available at no extra charge (free) through PTFs (see Chapter 2, “AS/400 Analyze User Objects Inventory Tool” on page 7) can help evaluate the software you have on your system.

After you have investigated the software on your system that may have potential date impact problems, you can then decide whether to use one of the following methods:

- **Upgrade**

  If the software has been written by another company, try to obtain a Year 2000 ready version.

- **Replace**

  Where an upgrade is not possible (for older applications), a replacement may be necessary.

- **Convert**

  Conversion is only an option if the source code is available and you have the authority to modify it. Owning the source code for an application does not necessarily mean that you have the authority to change it. Program objects without source code can be identified from the application inventory tool described in Chapter 2, “AS/400 Analyze User Objects Inventory Tool” on page 7.

1.2.3 Impact Analysis

The purpose of the impact analysis stage is to search the applications for all the affected date fields. Some tools do this by finding date-type data in the database and others look for field names that are dates. Regardless of the method, the result is a file containing all of the date fields from the application that are fed into a conversion tool to expand them to remove any ambiguity.

1.2.4 Conversion

Consider the following ways to make an application Year 2000 ready. The method you choose depends on the needs and resources of your organization:

- **Unconventional methods**

  There has been a case where a company has overcome the Year 2000 issue for one application by subtracting 28 years from all dates and using logic to bridge what is interpreted in the application to what the user displays and inputs. This is a temporary solution and should only be used in the case where an application may be replaced in the near future.

- **Windowing technique**

  This method uses a window of a fixed number of years to interpret which century the year is. For example, for a window of 100 years, this typically means that the century is the 20th if the two-digit year value is greater than 40. For a year value less than or equal to 40, the century is the 21st. This method does not change the database but changes the programs that access the dates from the database.

- **Removing ambiguity**

  By far the most popular and permanent solution is to remove all ambiguity from date fields. That is, ensure it is clear throughout the application which century the year refers to. Ambiguity can be removed in the following ways:

  – Converting a two-digit year to a four-digit year format
  – One-digit century code
  – Standard date representation
  – Windowing techniques (discussed previously)

Regardless of the method you choose, be aware that all date logic throughout the application must be modified to ensure there is no confusion when you move to the next century.

Note: For a detailed description of some of these techniques, along with their advantages and disadvantages, refer to *The Year 2000 and 2-Digit Dates: A Guide for Planning and Implementation,* GC28-1251.

1.3 Tools

There are many different tools offered to aid Year 2000 conversions. IBM offers the following tools:

- **Application Inventory Tool**

  Available through PTFs, the ANZUSROBJ command allows you to collect an inventory listing of the software on your system with potential Year 2000 issues.

- **Impact Analysis**

  There are many impact analysis tools available. Please refer to the IBM AS/400 Partners in Development Year 2000 home page for the latest information on these tools, including those tools that interface with BYPASS2000 at: http://www.as400.ibm.com/developer/

- **Code Conversion**

  BYPASS2000 converts programs from two-digit to four-digit years leaving display (MAP) and printer (PRT) files the same but bridging with extra logic.

Disclaimer - Important Please Read

Notice that it is the customer's responsibility to ensure that all applications on the AS/400 system are ready for the change in the century. All AS/400 hardware models are ready if they are running a Year 2000 certified version of the operating system (V3R2M0, V3R7M0, or later). If applications are modified, they should be rigorously tested by the customer to validate that they are operating correctly. Refer to *Make Your AS/400 System Year 2000 Ready*, SG24-5401, available only online at: http://www.redbooks.ibm.com/

In the Redbooks Home page, click **Redbooks Online!** On the page that appears, enter **SG24-5401** for the search string and then click **Submit Search**.

Chapter 2. AS/400 Analyze User Objects Inventory Tool

This chapter provides an overview of the IBM AS/400 Analyze User Objects Inventory Tool for OS/400 and the importance it has in planning for the Year 2000. It also describes the Analyze User Objects Inventory Tool options and how to use these options along with additional hints and tips.

2.1 Overview of Analyze User Objects Inventory Tool

The Analyze User Objects Inventory Tool is available as an OS/400 Program Temporary Fix (PTF) through normal PTF distribution channels. The tool searches all user AS/400 libraries for objects and verifies that the source code exists for each object (assuming that both object and source exist on the same system).

The Analyze User Objects Inventory Tool is an initial step at determining a high level awareness of which applications are without source code and which applications are out-of-sync with their source code (that is, objects with source that have a different date and time).

Both a system summary report and a library summary report are provided with information in the following categories:

- Program type:
  - OPM RPG
  - ILE RPG
  - OPM COBOL
  - ILE COBOL
  - OPM CLP
  - ILE CLP
  - C
  - DFU
  - Other
- File type:
  - Physical
  - Logical
  - Display
  - Printer
  - Other
- Other source-based objects:
  - Commands
  - Service programs
  - Modules
  - Panel groups
- Data holding objects:
  - User spaces
  - Data areas
  - User indexes
  - Other objects

A detailed exception report by library also provides the following information:

- Objects with different source date and time
- Objects without source available on the system
- Objects that often contain user data which are not created from source
- Source files that contain source members that are part of an application which are not compiled into objects (for example, REXX and FMTDTA source members).

It identifies the total number of lines of code known for those exceptions and if the object has been used in the last 13 months. A summary or detailed output file is provided for additional query reports.

The Analyze User Objects Inventory Tool does not provide a list of source files for which no object exists. Two sample Query/400 definitions are provided at the end of this chapter to meet this requirement.

The Analyze User Objects Inventory Tool does not provide a date impact analysis. Many impact analysis tools are available; some are even free.

Be sure to visit the IBM AS/400 Partners In Development Year 2000 home page for the latest information on impact analysis tools at:
http://www.as400.ibm.com/developer/

An IBM tool to provide impact analysis is SEARCH2000 (see Chapter 3, “SEARCH2000 Impact Analysis Tool” on page 29).

Source conversion can be done using AS/400 BYPASS2000 (see Chapter 4, “BYPASS2000 Product Overview” on page 47).

The Analyze User Object (ANZUSROBJ) function is distributed through a PTF (Program Temporary Fix) for the following OS/400 releases:

```
Version 2 Release 3 Modification 0 (V2R3M0)        SF41293
Version 3 Release 0 Modification 5 (V3R0M5)        SF41405
Version 3 Release 1 Modification 0 (V3R1M0)        SF41382
Version 3 Release 2 Modification 0 (V3R2M0)        SF41383
Version 3 Release 6 Modification 0 (V3R6M0)        SF41385
Version 3 Release 7 Modification 0 (V3R7M0)        SF41384
Version 4 Release 1 Modification 0 (V4R1M0)        SF42427
```

The Inventory Tool is not available for OS/400 releases prior to V2R3M0.

Alternative Tool

The SSILIB (System Summary Information Library) tool was written by Ed van Weeren from IBM Netherlands and is available at no extra charge (free). It is offered as an unsupported application. The tool runs on an AS/400 system to provide you with a quick overview (inventory) of all objects to be examined for the Year 2000 exercise. The following main functions are provided by the application:

- List object types for each library and the source relationship
- List the number of source lines
- List the potential fields which could contain a date (the selection criteria can be modified by using the Query product)

The purpose of this application is similar to the Analyze User Objects Inventory Tool. It provides information about objects and their relationship to sources from which they are built. It provides reports for both objects with or without source and source with or without objects. It also gives an indication of fields that may be date related.

You must download this tool from the ITSO Web page at:
http://www.redbooks.ibm.com/

From the navigation bar on the left, select **Additional Materials**. Then, in the list of folders on the right, find **SG242156**. Follow that link. Download the ssilbzip.exe file to your PC. The self-extracting file contains a README file with all further instructions in English.

2.2 Installing the Analyze User Objects Inventory Tool

The Analyze User Objects Inventory Tool can be installed by applying the appropriate OS/400 PTF. Use the PTF cover letter instructions to correctly install the tool.

You can access the Analyze User Objects Inventory Tool functions through the use of the Analyze User Objects (ANZUSROBJ) command.

2.3 Analyze User Objects Inventory Tool Command

When you install the inventory tool, the Analyze User Object (ANZUSROBJ) command has its public authority set to *EXCLUDE. You must have *ALLOBJ special authority of to run the ANZUSROBJ command.

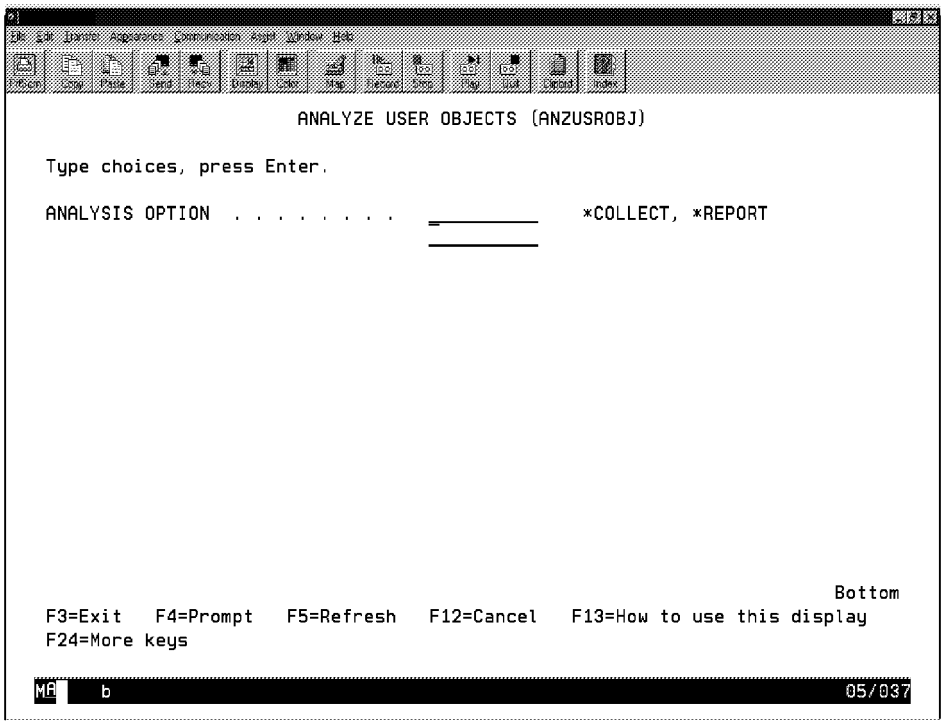


Figure 3. Command Statement for Analyze User Object (ANZUSROBJ) Command

The following parameters and their options can be used with the ANZUSROBJ command:

OPTION

Specifies whether data collection, data reporting, or both should be done. The possible values are:

***COLLECT**

User object data in all user libraries is searched and collected for later analysis.

***REPORT**

User object data is analyzed and reported for the libraries specified (LIB parameter).

When you specify the REPORT option, you are prompted for the following parameters (see Figure 4 on page 11).

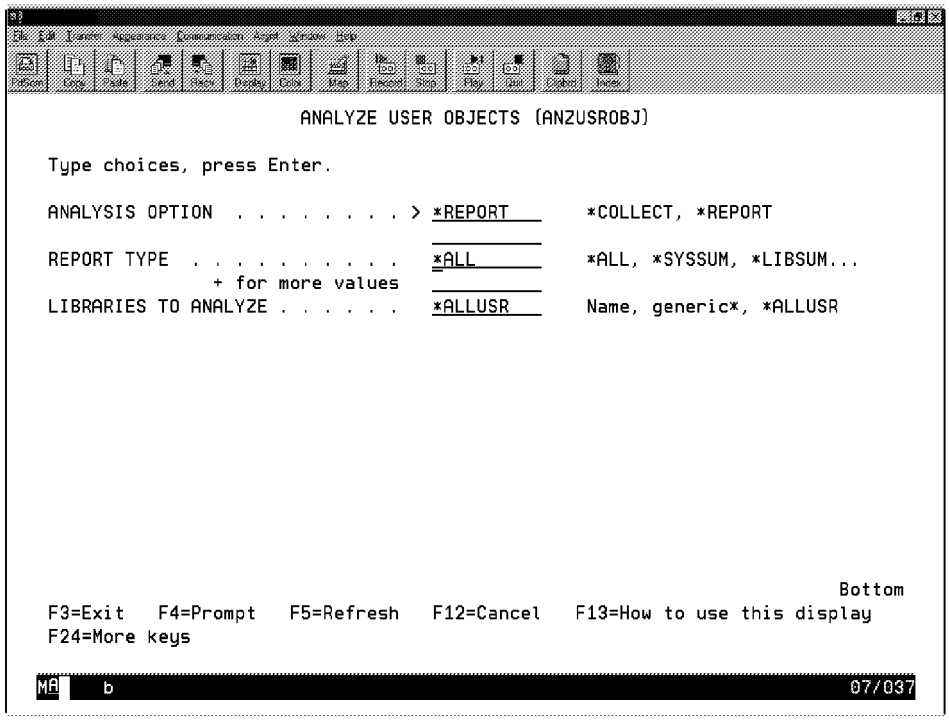


Figure 4. Command Language for Analyze User Object (ANZUSROBJ) Command

RPTTYPE

Specifies the types of user object information that are to be created. The possible values are:

***ALL** The system summary, library summary, and library detail user object analysis reports are created.

***SYSSUM** The system summary report is created.

***LIBSUM** The library summary report is created.

***LIBDTL** The library detail report is created.

LIB

Specifies the libraries to be analyzed when creating reports. The possible values are:

***ALLUSR** All user libraries are analyzed.

***generic-library-name**

Specify the generic name of the library or group of libraries to be analyzed. To specify a generic library name, add an asterisk (*) at the end of the character string that is common to the names of all libraries to analyze.

library-name

Specify the full name of the library to analyze.

2.4 Collecting Year 2000 Inventory Information

The Analyze User Object (ANZUSROBJ) command may take some time to complete, depending on your system configuration. We suggest that you run this command in batch. Testing was performed at the IBM Rochester facility using the following AS/400 systems:

- AS/400 Model D60:
  - OS/400 V3R1M0
  - 21GB of disk storage at 46% used
  - Approximately 28 libraries
  - Over 479 source members
  - 41,000 lines of source code
- AS/400 Model 510:
  - OS/400 V3R7M0
  - 45GB of disk storage at 82% used
  - Approximately 200 libraries
  - Over 2400 source members
  - 240,000 lines of source code

The ANZUSROBJ command took around five minutes to complete on these systems and used approximately 5% CPU.

The command searches all user libraries for objects created on the AS/400 system and writes the output to the QAUS2DET database file in library QUSRSYS. See Section 2.6, “Record Layout for QAUS2DET” on page 20, for additional information if you want to write your own query programs.

2.5 Reports

The three reports the ANZUSROBJ command can produce for you are object inventory exception reports. These reports warn you of possible problems with your application components.

The possible problems that are signaled are:

- Objects that appear to be created from source, which is not available on the system or has a last change date and time mismatch.
- Objects that often contain user data which are not created from source.
- Source files that contain members that are part of an application which are not compiled into objects (for example, REXX and FMTDTA source members).

The following tables provide information on the various object source types that can show up on the reports.

Table 1. Program Object Extensions

Program Object	Extension (Source Type)
OPM RPG	RPG
ILE RPG	RPGLE
OPM COBOL	CBL
ILE COBOL	CBLLE
OPM CLP	CLP
ILE CLP	CLLE
C	C
DFU	DFU DFUEXEC
Other	CBL36 CBL38 CLP38 RPG36 RPG38 PAS PLI PLI38 QRYEXC And so on

Table 2. File Object Extensions

File Objects	Extension (Source Type)
Physical Files	PF PHY
Logical Files	LF LGL
Display Files	DSPF
Printer Files	PRTF
Other	CMNF38 DDMF DDMF38 DSPF38 ICFF SAVF And so on

Table 3. Other Source-Based Object Extensions (if any)

Other Source-Based Objects	Extension (Source Type)
Commands	
Service Programs	
Modules	CLE RPGLE
Panel Groups	

Table 4. Data Holding Objects and Their Object Types

Data Holding Objects	Object Type
User Spaces	*USRSPC
Data Areas	*DTAARA
User Indexes	*USRIDX

Table 5. Other Object Extensions (if any)

Other Objects	Extension (Source Type)
*MENU	UIM
*DTAQ	
*JOBD	
*JRN	
*JRNRCV	
*OUTQ	
*QRYDFN	
And so on	

2.5.1 System Summary Report

The *System Summary - Object Inventory Exception Report* shows the count for each type of application component and the number that have problems. The only libraries considered are the ones that have possible problems.

The report contains a summary of the system. The component and possible problem counts are grouped into programs, files, data holding objects, and other objects. See Figure 5 on page 15.

```
                                OBJECT INVENTORY EXCEPTION REPORT

SYSTEM SUMMARY:                                        VERSION/RELEASE/MOD . :   V3R1M0
  DATE OF COLLECTION  . :   05/28/97                   AUXILIARY STORAGE . . :   20462
  TIME OF COLLECTION  . :   12:25:52                   SYSTEM ASP SIZE . . . :   19048
  INVENTORY FROM SYSTEM :   SYSXXXXX                   SYSTEM ASP PERCENT
  SYSTEM MODEL  . . . . :    E95                       USED  . . . . . . . :   70.7606

PROGRAMS:                                            FILES:

 OPM RPG . . . . . . . :   1587                        PHYSICAL FILES  . . . :   4174
 POSSIBLE OPM RPG                                     POSSIBLE PHYSICAL FILE
   SOURCE PROBLEMS . . :   299                          PROBLEMS  . . . . . :   410

 ILE RPG . . . . . . . :   169                         LOGICAL FILES . . . . :   1171
 POSSIBLE ILE RPG                                     POSSIBLE LOGICAL FILE
   SOURCE PROBLEMS . . :   64                           PROBLEMS  . . . . . :   182

 OPM COBOL . . . . . . :   165                         DISPLAY FILES . . . . :   2347
 POSSIBLE OPM COBOL                                   POSSIBLE DISPLAY FILE
   SOURCE PROBLEMS . . :   79                           PROBLEMS  . . . . . :   293

 ILE COBOL . . . . . . :   3                           PRINTER FILES . . . . :   353
 POSSIBLE ILE COBOL                                   POSSIBLE PRINTER FILE
   SOURCE PROBLEMS . . :   1                            PROBLEMS  . . . . . :   118

 POSSIBLE ILE CLP                                     POSSIBLE COMMAND
 C . . . . . . . . . . :   151                          PROBLEMS  . . . . . :   531
 POSSIBLE C SOURCE
   PROBLEMS  . . . . . :   124                        SERVICE PROGRAMS  . . :   35
                                                      POSSIBLE SERVICE
 DFU . . . . . . . . . :   26                           PROGRAM PROBLEMS  . :   27
 POSSIBLE DFU SOURCE
   PROBLEMS  . . . . . :   26                         MODULES . . . . . . . :   402
                                                      POSSIBLE MODULE
 OTHER . . . . . . . . :   2910                         PROBLEMS  . . . . . :   92
 POSSIBLE OTHER PROGRAM
   SOURCE PROBLEMS . . :   2234                       PANEL GROUPS  . . . . :   237
                                                        PROBLEMS  . . . . . :   382
                                                      POSSIBLE PANEL GROUP


DATA HOLDING OBJECTS:                                   PROBLEMS  . . . . . :   92
  USER SPACES . . . . . :   365                     OTHER OBJECTS:
  DATA AREAS  . . . . . :   179
  DATA QUEUES . . . . . :   46                       OTHER . . . . . . . . :   4806
  USER INDEXES  . . . . :   9                        POSSIBLE OTHER SOURCE
                                                       PROBLEMS  . . . . . :   382
```

Figure 5. Sample System Summary Report

2.5.2 Library Summary Report

The *Library Summary - Object Inventory Exception Report* lists only those libraries that have possible problems. Libraries are listed in alphabetical order. A count of the different types of application components with possible problems is provided (see Figure 6 on page 16).

OBJECT INVENTORY EXCEPTION REPORT Page 2

SYSXXXXX 05/29/97

09:47.37

SUMMARY OF OBJECTS WITH POSSIBLE SOURCE PROBLEMS:

LIBRARY	PROGRAMS	FILES	OTHER SOURCE	DATA OBJECTS	OTHER
$$TEST	0	1	0	0	0
$SPAPI32	0	0	0	1	0
$SPAPI33	0	0	0	1	0
A	10	0	0	0	0
ACCOLA	29	1	11	3	1
AFPUTEST	0	4	0	0	0
ANYIMG2I	6	0	1	0	0
ATSADMIN	2	1	0	0	0
A1	2	0	0	0	0
BERESSI	0	0	0	0	1
BILLRTEMP	1	0	0	0	0
BN	0	1	0	0	0
BRUCE	1	0	0	0	0
BSCHLOSS	5	0	0	0	0
.					
.					
.					
QGPL	22	16	8	18	1
QRCL	1	0	0	13	0
QUSRSYS	3	15	2	145	1
RADARS	4	0	2	2	0
RAMARAO	15	0	9	1	0
RAMOST	1	0	0	0	0
RAPTOR2	3	0	4	0	0
RAS	16	0	3	0	0
RAVITEST	27	5	8	1	1
RED	2	0	0	0	0
REXX	5	1	0	0	0
RI04	0	28	0	1	0
RKFCLIB	6	0	2	0	0
RODJ	20	0	4	2	0
RODJFIX	1	0	0	0	0

Figure 6. Sample Library Summary Report

2.5.3 Library Detail Report

The *Library Detail - Object Inventory Exception Report* lists detailed information for the libraries that have possible problems. The report shows (library-by-library in alphabetical order) the count of application components and possible errors for each of the components. The components are grouped into programs, files, data holding objects, and other objects (see Figure 7 on page 17 and Figure 8 on page 19).

```
                                OBJECT INVENTORY EXCEPTION REPORT                                Page   275
                                                                                      SYSXXXXX  05/29/97  09:47:37
LIBRARY:   QGPL
PROGRAMS:                                                    FILES:

  OPM RPG . . . . . . . :   1                                  PHYSICAL FILES  . . . :   268
  POSSIBLE OPM RPG                                             POSSIBLE PHYSICAL FILE
    SOURCE PROBLEMS . . :   0                                    PROBLEMS  . . . . . :    10
  ILE RPG . . . . . . . :   0                                  LOGICAL FILES . . . . :    15
  POSSIBLE ILE RPG                                                    POSSIBLE LOGICAL FILE
    SOURCE PROBLEMS . . :   0                                    PROBLEMS  . . . . . :     0
  OPM COBOL . . . . . . :   1                                   DISPLAY FILES . . . . :   1
  POSSIBLE OPM COBOL                                           POSSIBLE DISPLAY FILE
    SOURCE PROBLEMS . . :   0                                    PROBLEMS  . . . . . :     1
  ILE COBOL . . . . . . :   0                                  PRINTER FILES . . . . :   22
  POSSIBLE ILE COBOL                                           POSSIBLE PRINTER FILE
    SOURCE PROBLEMS . . :   0                                    PROBLEMS  . . . . . :     3
  OPM CLP . . . . . . . :   7                                  OTHER . . . . . . . . :  132
  POSSIBLE OPM CLP                                             POSSIBLE OTHER FILE
   SOURCE PROBLEMS . . :    5                                    SOURCE PROBLEMS . . :     2
   ILE CLP . . . . . . . :  0                                OTHER SOURCE BASED OBJECTS:
   POSSIBLE ILE CLP
    SOURCE PROBLEMS . . :   0                                  COMMANDS  . . . . . . :   13
                                                                POSSIBLE COMMAND
  C . . . . . . . . . . :   2                                    PROBLEMS  . . . . . :     6
   POSSIBLE C SOURCE
     PROBLEMS  . . . . . :  2                                  SERVICE PROGRAMS  . . :    1
                                                                POSSIBLE SERVICE
   DFU . . . . . . . . . :  0                                     PROGRAM PROBLEMS  . :    1
   POSSIBLE DFU SOURCE
    PROBLEMS  . . . . . :   0                                  MODULES . . . . . . . :    3
                                                                POSSIBLE MODULE
  OTHER . . . . . . . . :  20                                   PROBLEMS  . . . . . :     2
  POSSIBLE OTHER PROGRAM
     SOURCE PROBLEMS . . : 14                                 PANEL GROUPS  . . . . :     0
                                                                POSSIBLE PANEL GROUP
 DATA HOLDING OBJECTS:                                            PROBLEMS  . . . . . :   0
  USER SPACES . . . . . :   4
  DATA AREAS  . . . . . :  11                               OTHER OBJECTS:
  DATA QUEUES . . . . . :  2
  USER INDEXES  . . . . :  1                                 OTHER . . . . . . . . :   128
                                                                POSSIBLE OTHER SOURCE
                                                                PROBLEMS  . . . . . :     1
```

Figure 7. Sample Library Detail Report (Part 1 of 3)

Every object in each included library is listed with the following information (see Figure 8 on page 19):

Object Name
This is the name of the user object. For files containing source members, this is blank for the member entries. The members follow the file within the report.

Object Type
This is the type of the object. For files containing source members, this is the member extension for the member entries.

Object Extension
This is used for the source member name when listing REXX and FMTDTA members.

Text
This is a text description associated with the object or source member.

Source Last Changed Date/Time
This is the date/time of the last change found for the source. If blank, this indicates the source is not available on the system or the object is an ILE

program or service program that is created from modules and not source directly.

Object Creation Source Date/Time
This is the date of the last change for source used in the creation of the object. If blank, this indicates the object was not created from source or the object is an ILE program or service program that is created from modules and not source directly.

Lines
This is the number of lines of source code found.

*****
A "1" indicates that the object has not been used in the last 12 months.

LAST OBJECT	OBJECT TYPE	CREATION EXTENSION	SOURCE TEXT	SOURCE LAST CHANGED DATE/TIME	OBJECT CREATION SOURCE DATE/TIME	LINES
EDTVAR	*CMD		Edit variable command - EDTVAR		5/25/94 17:45:19	0
PUTSPLF	*CMD				4/25/91 9:39:17	0
RTVJOBD	*CMD		Retrieve JOBD parms - RTVJOBD		5/25/94 17:45:44	0
STRPRTINT	*CMD		Command for Call PRTINT in Lib PIT		3/24/92 14:26:24	0
TESTHPT	*CMD		Command for QwpzHostPrintTransform		12/08/94 10:33:51	0
TRCPSF	*CMD		PSF Trace		3/11/94 16:51:11	1
CHINOOK	*DTAARA					0
FRISCO	*DTAARA					0
PJ	*DTAARA		R/DARS-400 User defaults data area			0
PRT02	*DTAARA					0
PRT040205	*DTAARA					0
PRT050105	*DTAARA					0
PSF3130	*DTAARA					0
QFFMSTOPT1	*DTAARA					0
QSS1MRI	*DTAARA					0
QTPST	*DTAARA					0
TCP3130	*DTAARA					0
DSI00001Q	*DTAQ					0
DSI00002Q	*DTAQ					0
COMPILE	*FILE	PF			4/04/95 14:31:08	0
KPS	*FILE	PF			6/28/96 12:05:24	0
KPSNEW	*FILE	PF			6/28/96 12:09:57	0
KPSOLD	*FILE	PF			6/28/96 12:16:41	0
LQ4SAVE	*FILE	PF			7/29/96 13:49:52	0
QAEZDI0001	*FILE	PF	Old name QAEZDISK in QGPL owned by XZS25		8/21/96 10:30:37	0
QDGDCMDI	*FILE	PF			7/17/96 18:12:13	0
TSL#WSC	*FILE	PF			7/25/95 17:03:19	0
T8189DB	*FILE	PF	Database file for APPC examples		7/26/93 6:55:00	
T8189DSP	*FILE	DSPF	Display file for APPC examples		7/26/93 6:55:02	
T8189ICF	*FILE	ICFF	ICF file for APPC examples		7/26/93 6:55:01	0
USERS	*FILE	PF			6/28/96	12:07:16
WPAF198	*FILE	PRTF			6/16/93 7:17:07	0
WPAF199	*FILE	PRTF			6/16/93 7:17:07	0
WPAF200	*FILE	PRTF			6/16/93 7:17:07	0
SAMPLE	*MODULE	CLE		8/04/94 18:45:04	3/31/94 9:37:14	4
TESTHPTC	*MODULE	CLE			4/11/95 7:50:10	0
BOBFFACB6	*PGM	CLE				0
QBNBLOAD	*PGM	BND		10/18/96 11:49:35		160 1
QDCVRD	*PGM			10/18/96 11:49:35		160
QLPFRC1P	*PGM			10/18/96 11:49:35		160
QPDCFGPM	*PGM	CLP		11/15/94 16:51:06		0
QPDMSGCR	*PGM	CLP		9/21/94 22:40:04		108
QPZCPFI	*PGM			9/21/94 22:40:04		108
QQXCPI	*PGM	BND		9/21/94 22:40:04		108
QQXFASTW	*PGM	BND		9/21/94 22:40:04		108
QSPPUTF	*PGM			9/21/94 22:40:04		108
QTVENDLN	*PGM			9/21/94 22:40:04		108
QWPZATRN	*PGM	CLE				0
QYPISERV	*PGM					0
Q5ANXTTG	*PGM	PAS				0
SAPACK	*PGM	CLE				0
TAACLPCC	*PGM	CLP	Edit variable - CPP for EDTVAR		8/01/94 9:25:58	0
TAACLPCC2	*PGM	CLP	Edit variable -Simple front end that doe		8/01/94 9:33:51	0
TAAJBDBC	*PGM	CLP	Retrieve JOBD parms - CPP for RTVJOBD		3/31/97 13:43:59	0
TESTHPTO	*PGM	CLE				0
T8189ICS	*PGM	C	APPC C program example ICF - Source		7/30/93 17:05:05	0
T8189ICT	*PGM	C	APPC C program example ICF - Target		7/30/93 17:05:07	0
DOC	*QRYDFN	QRY				0
QC2ALLOC	*SRVPGM	CSYSD				0
QEZPWRCLN	*USRIDX		OA CLEANUP AND POWER SCHEDULE DATA			0
COMM_AREA_	*USRSPC	SPOOL_FILE	Spool file transform space			0
PJ	*USRSPC	QUSLOBJ	For QUSLOBJ API use			0
TRANSSPACE	*USRSPC	SPOOL_FILE	Spool file transform space			0
TSTUS	*USRSPC	TEST	Event Adapter Buffe { 3			0
TESTHP4	*WSCST			3/31/94 17:20:24	3/31/94 10:23:29	346

Figure 8. Sample Library Detail Report (Part 2 of 3)

```
WPAF198     *FILE    PRTF                                                                  6/16/93   7:17:07           0
WPAF199     *FILE    PRTF                                                                  6/16/93   7:17:07           0
WPAF200     *FILE    PRTF                                                                  6/16/93   7:17:07           0
SAMPLE      *MODULE  CLE                                                 8/04/94  18:45:04   3/31/94   9:37:14           4
TESTHPTC    *MODULE  CLE                                                                    4/11/95   7:50:10           0
BOBFFACB6   *PGM     CLE                                                                                              0
QBNBLOAD    *PGM     BND                                               10/18/96  11:49:35                         160 1
QDCVRD      *PGM                                                       10/18/96  11:49:35                           160
QLPFRC1P    *PGM                                                       10/18/96  11:49:35                           160
QPDCFGPM    *PGM     CLP                                                                 11/15/94  16:51:06           0
QPDMSGCR    *PGM     CLP                                                 9/21/94  22:40:04                           108
QPZCPFI     *PGM                                                         9/21/94  22:40:04                           108
QQXCPI      *PGM     BND                                                 9/21/94  22:40:04                           108
QQXFASTW    *PGM     BND                                                 9/21/94  22:40:04                           108
QSPPUTF     *PGM                                                         9/21/94  22:40:04                           108
QTVENDLN    *PGM                                                         9/21/94  22:40:04                           108
QWPZATRN    *PGM     CLE                                                                                              0
QYPISERV    *PGM                                                                                                      0
Q5ANXTTG    *PGM     PAS                                                                                              0
SAPACK      *PGM     CLE                                                                                              0
TAACLPCC    *PGM     CLP         Edit variable - CPP for EDTVAR                                  8/01/94   9:25:58 0
TAACLPCC2   *PGM     CLP         Edit variable -Simple front end that doe                        8/01/94   9:33:51 0
TAAJBDBC    *PGM     CLP         Retrieve JOBD parms - CPP for RTVJOBD                           3/31/97  13:43:59 0
TESTHPTO    *PGM     CLE                                                                                            0
T8189ICS    *PGM     C           APPC C program example ICF - Source                             7/30/93  17:05:05 0
T8189ICT    *PGM     C           APPC C program example ICF - Target                             7/30/93  17:05:07 0
DOC         *QRYDFN  QRY                                                                                             0
QC2ALLOC    *SRVPGM  CSYSD                                                                                           0
QEZPWRCLN   *USRIDX              OA CLEANUP AND POWER SCHEDULE DATA                                                  0
COMM_AREA_  *USRSPC  SPOOL_FILE  Spool file transform space                                                          0
PJ          *USRSPC  QUSLOBJ     For QUSLOBJ API use                                                                 0
TRANSSPACE  *USRSPC  SPOOL_FILE  Spool file transform space                                                          0
TSTUS       *USRSPC  TEST         Event Adapter Buffe       { 3                                                      0
TESTHP4      *WSCST                                                 3/31/94  17:20:24   3/31/94  10:23:29         346
```

Figure 9. Sample Library Detail Report (Part 3 of 3)

2.6 Record Layout for QAUS2DET

The following information can be used to write your own query programs to produce reports using the information collected by the Inventory Tool in the QAUS2DET database file in library Y2RPTF. This file contains the results of running the *COLLECT option of the ANZUSROBJ command and is the basis for the *Library Detail* report.

```
R QUSDET
  OBJNAM        10H         TEXT('OBJECT NAME')
  OBJLIB        10H         TEXT('OBJECT LIBRARY')
  OBJTYP        10H         TEXT('OBJECT TYPE')
  INFSTS         1H         TEXT('INFO STATUS')
  OBJEXT        10H         TEXT('EXTENDED ATTR')
  OBJTXT        50H         TEXT('OBJECT TEXT')
  USRATR        10H         TEXT('USER DEFINED ATTR')
  ASP            3  0       TEXT('ASP')
  OBJOWN        10H         TEXT('OBJECT OWNER')
  OBJDMN         2H         TEXT('OBJECT DOMAIN')
  TSCRT          8H         TEXT('TIME STAMP - CREAT')
  CRTYMD        13H         TEXT('CYMD - CREATION')
  TSCHG          8H         TEXT('TIME STAMP - CHANGE')
  CHGYMD        13H         TEXT('CYMD - CHANGE')
  OBJSTG        10H         TEXT('STORAGE')
  CMPSTS         1H         TEXT('COMPRESSION STATUS')
  ALWCHG         1H         TEXT('ALLOW CHANGE BY PGM')
  CHGED          1H         TEXT('CHANGED BY PGM')
  OBJAUD        10H         TEXT('OBJECT AUDIT VALUE')
  SRCFIL        10H         TEXT('SOURCE FILE')
  SRCLIB        10H         TEXT('SOURCE FILE LIBRARY')
  SRCMBR        10H         TEXT('SOURCE FILE MEMBER')
  SRCCHG        13H         TEXT('SRC MBR CHANGED')
  SRCUPD        13H         TEXT('SRC FILE UPDATED ON')
  SRCLIN         9B 0       TEXT('SRC RECS IN MBR')
  CRTPRF        10H         TEXT('CREATORS USRPRF')
  CRTSYS         8H         TEXT('CREATED ON SYSTEM')
  SYSLVL         9H         TEXT('SYSTEM LEVEL')
  CRTCMP        16H         TEXT('COMPILER')
  OBJLVL         8H         TEXT('OBJECT LEVEL')
  USRCHG         1H         TEXT('USER CHANGED')
  LICPGM        16H         TEXT('LICENSED PROGRAM')
  PTF           10H         TEXT('PTF')
  APAR          10H         TEXT('APAR')
  PRMGRP        10H         TEXT('PRIMARY GROUP')
  TSOBJS         8H         TEXT('TIME STAMP OF SAVE')
  DTOBJS        13H         TEXT('CYMD - SAVE')
  TSOBJR         8H         TEXT('TIME STAMP OF RST')
  DTOBJR        13H         TEXT('CYMD - RST')
  SAVSIZ         9B 0       TEXT('SAVE SIZE')
  SAVMLT         9B 0       TEXT('SAVE SIZE MULT')
  SAVSEQ         9B 0       TEXT('SAVE SEQUENCE NBR')
  SAVCMD        10H         TEXT('SAVE COMMAND')
  SAVVOL        71H         TEXT('SAVE VOLUME')
  SAVDEV        10H         TEXT('SAVE DEVICE')
  SAVF          10H         TEXT('SAVE FILE')
  SAVFL         10H         TEXT('SAVE FILE LIBRARY')
  SAVLBL        17H         TEXT('SAVE LABEL')
  TSSAVA         8H         TEXT('TIME STAMP - SAVACT')
  DTSAVA        13H         TEXT('CYMD - SAVACT')
  TSLSTU         8H         TEXT('TIME STAMP LASTUSED')
  DTLSTU        13H         TEXT('CYMD - LAST USED')
  TSRST          8H         TEXT('TIME STAMP - RESET')
  DTRST         13H         TEXT('CYMD - RESET')
  DAYUSD         9B 0       TEXT('DAYS USED COUNT')
  USGINF         1H         TEXT('USAGE INFO UPDATED')
  RPTTYP         1H         TEXT('REPORT TYPE')
K RPTTYP
K OBJLIB
K OBJTYP
K OBJNAM
```

Figure 10. Record Layout for QAUS2DET

2.7 Additional Query/400 Queries

The following section shows additional and useful Query/400 queries developed by Jim Kaminski from IBM U.S. Included are Query/400 source listings that allow some quick analysis not covered by the ANZUSROBJ command.

2.7.1 Total Source Code by Library

This query provides a count of lines of source code in each library, a count of the number of source members, and a final count of all members and all lines of source code for all libraries.

A file called DSPFDMBR was created from the following CL command:

```
DSPFD FILE(*ALLUSR/*ALL) TYPE(*MBR) OUTPUT(*OUTFILE)
   FILEATR(*PF) OUTFILE(QGPL/DSPFDMBR)
```

The DSPFDMBR file is used as input to the following query:

```
  Query . . . . . . . . . . . . . . . . SRCTOTL
    Library . . . . . . . . . . . . . . A970123J
  Query text  . . . . . . . . . . . . .
  Query CCSID . . . . . . . . . . . . .    37
  Query language id . . . . . . . . . . ENU
  Query country id  . . . . . . . . . . US
  Collating sequence  . . . . . . . . . Hexadecimal
  Processing options
    Use rounding  . . . . . . . . . . . Yes (default)
    Ignore decimal data errors  . . . . . No (default)
    Ignore substitution warnings  . . . . Yes
    Use collating for all compares  . . . Yes
 Selected files
  ID     File           Library       Member        Record Format
  T01    DSPFDMBR       A970123J      DSPFDMBR      QWHFDMBR
 Select record tests
  AND/OR   Field            Test     Value (Field, Numbers, or 'Characters')
           MBDTAT           EQ       'S'
 Ordering of selected fields
   Field           Sort      Ascending/  Break  Field
   Name            Priority  Descending  Level  Text
   MBLIB           10        A           1      Library
   MBSEU2          20        A           2      Source type
   MBFILE          30        A           3      File
   MBTXT                                        Text 'description'
   MBNAME                                       Member
   MBMTXT                                       Member text description
   MBNRCD                                       Current number of records
 Report column formatting and summary functions
   Summary functions:  1-Total, 2-Average, 3-Minimum, 4-Maximum, 5-Count          Overrides
   Field          Summary    Column                          Dec  Null       Dec  Numeric
   Name            Functions  Spacing  Column Headings        Len  Pos  Cap   Len  Pos
Editing
   MBLIB                       0        Library                 10
   MBSEU2          5           2        Source                  10
                                        Type
   MBFILE                      2        File                    10
   MBTXT                       2        TEXT                    50              30
                                        Text
                                        'Description'
   MBNAME                      2        Member                  10
   MBMTXT                      2        Member Text             50              30
                                        Description
   MBNRCD          1           2        Current                 10               0
                                        Number Of
                                        Records
```

Figure 11. Total Source Code by Library - Query Source (Part 1 of 2)

```
Report breaks
  Break  New   Suppress   Break
  Level  Page  Summaries  Text
  0      No    No         Final Total Members/Records
  1      No    No         Library Total Members/Records
  2      No    No         ' &MBSEU2 ' Source Type Total Mbr/Rcds
  3      No    Yes
Selected output attributes
  Output type . . . . . . . . . . . . . . Printer
  Form of output  . . . . . . . . . . . . Detail
  Line wrapping . . . . . . . . . . . . . No
Printer Output
  Printer device  . . . . . . . . . . . . *PRINT
  Report size
    Length  . . . . . . . . . . . . . . .  66
    Width . . . . . . . . . . . . . . . . 132
  Report start line . . . . . . . . . . .   6
  Report end line . . . . . . . . . . . .  60
  Report line spacing . . . . . . . . . . Single space
  Print definition  . . . . . . . . . . . Yes
Printer Spooled Output
  Spool the output  . . . . . . . . . . . (Defaults to value in print file, QPQUPRFIL)
  Form type . . . . . . . . . . . . . . . (Defaults to value in print file, QPQUPRFIL)
  Copies  . . . . . . . . . . . . . . . .   1
  Hold  . . . . . . . . . . . . . . . . . (Defaults to value in print file, QPQUPRFIL)
Cover Page
  Print cover page  . . . . . . . . . . . Yes
Cover page title
      Total Source Code by Library
Page headings and footings
  Print standard page heading . . . . . . Yes
    Page heading
      Total Source Code by Library
    Page footing
```

Figure 12. Total Source Code by Library - Query Source (Part 2 of 2)

```
                                   Total Source Code by Library
                                   QUERY NAME . . . . . SRCTOTL
                                   LIBRARY NAME . . . . A970123J
                                   FILE         LIBRARY      MEMBER       FORMAT
                                   DSPFDMBR     A970123J     DSPFDMBR     QWHFDMBR
                                   DATE . . . . . 05/20/97
                                   TIME . . . . . 09:58:21
05/20/97  09:58:21                                                                  Source Code by Library PAGE    1

Library    Source       File        TEXT                              MEMBER      Member Text                              Current
            Type                    Text  Description                                                                   Number Of
                                    '                                                                                 Description' Records
QGPL        CBL         QLBLSRC     FILE FOR COBOL SOURCE             VERIFY      COBOL VERIFICATION PROGRAM                     1
     ' CBL ' Source Type Total Mbr/Rcds
     TOTAL31
      COUNT 1
CBLLE                   QCBLLESRC   FILE FOR ILE COBOL SOURCE         VERIFY      ILE COBOL VERIFICATION PROGRAM                67
     ' CBLLE ' Source Type Total Mbr/Rcds
      TOTAL 67
      COUNT 1
QGPL         CBL38      QCBLSRC     FILE FOR COBOL SOURCE             VERIFY      COBOL VERIFICATION PROGRAM                    31
      ' CBL38 ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                                   31
      COUNT 1
            CL          QCLSRC      Default source data base file     QPADEV0005  Bryan O'Brien - dlt after 9/13                 6
                                    Default source data base file     TAP02       CREATED BY AUTO-CONFIGURATION                 3
      ' CL ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                                    9
      COUNT 2
            CLP         QCLSRC      Default source data base file     QSTRUP                                                   55
                                    Default source data base file     QSTRUP1                                                  59
      ' CLP ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                                  114
      COUNT 1
            RPG         QRPGSRC     FILE FOR RPG SOURCE               PROOF       CHECK OUT INSTALLATION                       23
      ' RPG ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                                   23
      COUNT 1
            RPGLE       QRPGLESRC   ILE RPG/400 Source File           PROOF       Check out Installation                       26
      ' RPGLE ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                                   26
      COUNT 1
            RPG38       QRPGSRC     FILE FOR RPG SOURCE               PROOF38     CHECK OUT INSTALLATION                       19
      ' RPG38 ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                                   19
      COUNT 1
            SQL         QM                                            J1                                                        9
      ' SQL ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                                    9
      COUNT 1
            TXT         QDDSSRC     Default source data base file     IFSSRC                                                    1
      ' TXT ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                                    1
      COUNT 1
      Library Total Members/Records
      TOTAL                                                                                                                  330
      COUNT 12

     .
     .
     .

       Final Total Members/Records
       TOTAL                                                                                                             242,577
       COUNT 2,412
```

Figure 13. Total Source Code by Library - Query Output Listing

2.7.2 Total Source Code by Library for Source Code without Objects

For each library by source type, a total count of the number of members and the number of lines of source code is provided. This includes a final count of all members and all lines of source code for all libraries, but only for members without created objects.

The DSPFDMBR file is created with the following CL command:

```
DSPFD FILE(*ALLUSR/*ALL) TYPE(*MBR) OUTPUT(*OUTFILE)
   FILEATR(*PF) OUTFILE(QGPL/DSPFDMBR)
```

The DSPOBJD file is created with the following CL command:

```
DSPOBJD OBJ(*ALLUSR/*ALL) OBJTYPE(*ALL) OUTPUT(*OUTFILE)
   DETAIL(*BASIC) OUTFILE(QGPL/DSPOBJD)
```

The DSPFDMBR and the DSPOBJD files were used as input to the following query:

```
  Query . . . . . . . . . . . . . . . . SRCTOTLNOB
    Library . . . . . . . . . . . . . . A970123J
  Processing options
    Use rounding  . . . . . . . . . . . Yes (default)
    Ignore decimal data errors  . . . . . No (default)
    Ignore substitution warnings  . . . . Yes
    Use collating for all compares  . . . Yes
Selected files
  ID     File           Library       Member        Record Format
  T01    DSPFDMBR       A970123J      DSPFDMBR      QWHFDMBR
  T02    DSPOBJD        A970123J      DSPOBJD       QLIDOBJD
Join tests
  Type of join  . . . . . . . . . . . . . Unmatched records with primary file
  Field              Test              Field
  T01.MBRES4         EQ                T02.ODOBNM
Ordering of selected fields
  Field           Sort      Ascending/  Break  Field
  Name            Priority  Descending  Level  Text
  T01.MBLIB       10        A           1      Library
  T01.MBSEU2      20        A           2      Source type
  T01.MBFILE      30        A           3      File
  T01.MBTXT                                    Text 'description'
Ordering of selected fields (continued)
  Field           Sort      Ascending/  Break  Field
  Name            Priority  Descending  Level  Text
  T01.MBNAME                                   Member
  T01.MBMTXT                                   Member text description
  T01.MBNRCD                                   Current number of records
```

Figure 14. Total Source Code (without Object) by Library - Query Source (Part 1 of 2)

```
 Report column formatting and summary functions
   Summary functions:  1-Total, 2-Average, 3-Minimum, 4-Maximum, 5-Count          Overrides
   Field          Summary    Column                              Dec  Null      Dec  Numeric
   Name            Functions  Spacing  Column Headings          Len  Pos  Cap   Len  Pos
Editing
   T01.MBLIB                  0        Library                   10
   T01.MBSEU2      5          2        Source                    10
                                       Type
   T01.MBFILE                 2        File                      10
   T01.MBTXT                  2        TEXT                      50             30
                                       Text
                                       'Description'
   T01.MBNAME                 2        Member                    10
   T01.MBMTXT                 2        Member Text               50             30
                                       Description
   T01.MBNRCD      1          2        Current                   10    0
                                       Number Of
                                       Records
 Report breaks
   Break  New   Suppress   Break
   Level  Page  Summaries  Text
   0      No    No         Final Total Members/Records
   1      No    No         Library Total Members/Records
   2      No    No         ' &MBSEU2 ' Source Type Total Mbr/Rcds
   3      No    Yes
 Selected output attributes
   Output type . . . . . . . . . . . . . . Printer
   Form of output  . . . . . . . . . . . . Detail
   Line wrapping . . . . . . . . . . . . . No
 Printer Output
   Printer device  . . . . . . . . . . . . *PRINT
   Report size
     Length  . . . . . . . . . . . . . . .  66
     Width . . . . . . . . . . . . . . . . 132
   Report start line . . . . . . . . . . .   6
   Report end line . . . . . . . . . . . .  60
   Report line spacing . . . . . . . . . . Single space
   Print definition  . . . . . . . . . . . Yes
Printer Spooled Output
  Spool the output  . . . . . . . . . . . (Defaults to value in print file, QPQUPRFIL)
  Form type . . . . . . . . . . . . . . . (Defaults to value in print file, QPQUPRFIL)
  Copies  . . . . . . . . . . . . . . . .   1
  Hold  . . . . . . . . . . . . . . . . . (Defaults to value in print file, QPQUPRFIL)
Cover Page
  Print cover page  . . . . . . . . . . . Yes
    Cover page title
      Total Source Code (without objects) by Library
Page headings and footings
  Print standard page heading . . . . . . Yes
    Page heading
      Total Source Code (without objects) by Library
    Page footing
```

Figure 15. Total Source Code (without Object) by Library - Query Source (Part 2 of 2)

```
                                  Total Source Code (without objects) by Library
                                 QUERY NAME . . . . . SRCTOTLNOB
                                 LIBRARY NAME . . . . A970123J
                                 FILE         LIBRARY      MEMBER       FORMAT
                                  DSPFDMBR     A970123J     DSPFDMBR     QWHFDMBR
                                  DSPOBJD      A970123J     DSPOBJD      QLIDOBJD
                                  DATE . . . . . . . . . 05/20/97
                                  TIME . . . . . . . . . 09:59:05
05/20/97  09:59:05                      Total Source Code (without objects) by Library                          PAGE    1

Library     Source      File        TEXT                            Member      Member Text                     Current
            Type                    Text                                        Description                     Number Of
                                    'Description'                               Records
QGPL        CBL         QLBLSRC     FILE FOR COBOL SOURCE           VERIFY      COBOL VERIFICATION PROGRAM            31
      ' CBL ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                       31
      COUNT 1
            CBLLE       QCBLLESRC   FILE FOR ILE COBOL SOURCE       VERIFY      ILE COBOL VERIFICATION PROGRAM        67
      ' CBLLE ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                       67
      COUNT 1
QGPL        CBL38       QCBLSRC     FILE FOR COBOL SOURCE           VERIFY      COBOL VERIFICATION PROGRAM            31
      ' CBL38 ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                       31
      COUNT 1
            CL          QCLSRC      Default source data base file   QPADEV0005  Bryan O'Brien - dlt after 9/13      6
                                    Default source data base file   TAP02       CREATED BY AUTO-CONFIGURATION       3
      ' CL ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                        9
      COUNT 2
            CLP         QCLSRC      Default source data base file   QSTRUP                                         55
                                    Default source data base file   QSTRUP1                                        59
      ' CLP ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                      114
      COUNT 2
            RPG         QRPGSRC     FILE FOR RPG SOURCE             PROOF       CHECK OUT INSTALLATION            23
      ' RPG ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                       23
      COUNT 1
            RPGLE       QRPGLESRC   ILE RPG/400 Source File         PROOF       Check out Installation            26
      ' RPGLE ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                       26
      COUNT 1
            RPG38       QRPGSRC     FILE FOR RPG SOURCE             PROOF38     CHECK OUT INSTALLATION            19
      ' RPG38 ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                       19
      COUNT 1
            SQL         QM                                          J1                                             9
      ' SQL ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                        9
      COUNT 1
            TXT         QDDSSRC     Default source data base file   IFSSRC                                         1
      ' TXT ' Source Type Total Mbr/Rcds
      TOTAL                                                                                                        1
      COUNT 1
      Library Total Members/Records
      TOTAL                                                                                                      330
      COUNT 12
     .
     .
     .
       Final Total Members/Records
       TOTAL                                                                                                 242,577
       COUNT 2,412
 * * *  E N D  O F  R E P O R T  * * *
```

Figure 16. Total Source Code (without Object) by Library - Query Output

Chapter 3. SEARCH2000 Impact Analysis Tool

This chapter describes SEARCH2000, which is the impact analysis tool used to discover the impact of date fields within your application.

3.1 Introduction

SEARCH2000 helps you find fields containing dates (year-sensitive fields) in your database files. It provides automated assistance in locating potential date fields in your databases. You confirm date fields and their format from the selected fields list created by SEARCH2000.

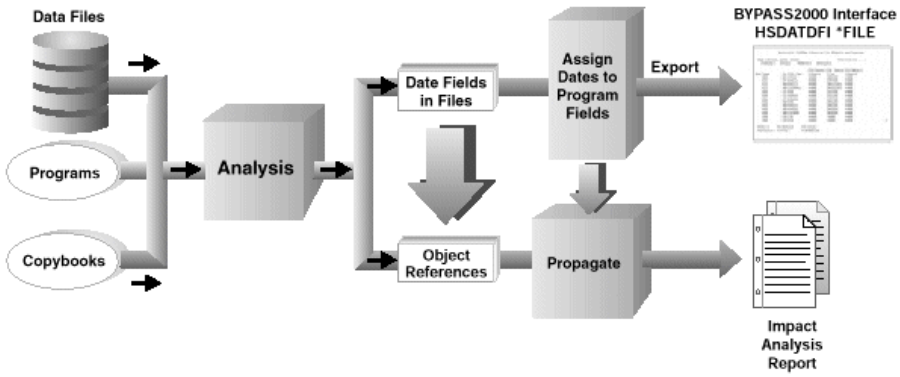


Figure 17. How Search2000 Works

SEARCH2000 produces a series of reports that enable you to more specifically size the conversion effort (see Section 3.2.2, “Reports” on page 31). These reports show files and programs impacted by changes to date layout.

SEARCH2000 produces a file called HSDATDFI that can interface with BYPASS2000, providing it with your year-sensitive database fields (see Appendix A, “Interfacing to BYPASS2000” on page 149). These files can automate a potentially time-consuming and error-prone manual process.

3.2 SEARCH2000 Process

You can see the process flow for SEARCH2000 in Figure 18 on page 30.

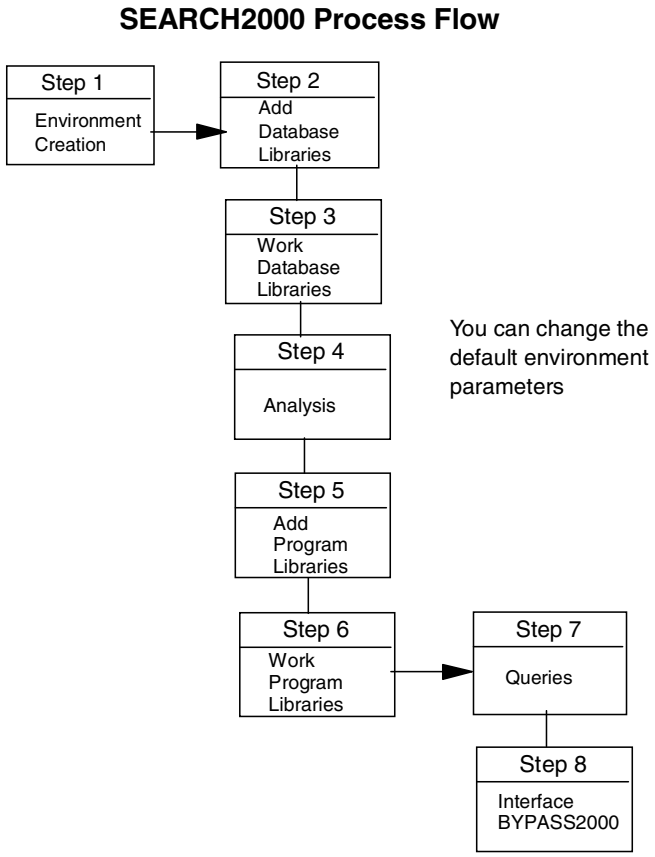


Figure 18. SEARCH2000 Process Flow

3.2.1 Analysis

SEARCH2000 analyzes the information based on the following requirements:

- You must specify the libraries that are searched.
- You may specify a subset of date formats to consider (see a list of possible date formats in Appendix A, “Interfacing to BYPASS2000” on page 149).
- You may specify a date range and some other parameters potentially affecting the date search technique.
- You may specify the number of records to include in the sample that is to be browsed.

SEARCH2000 looks for the date formats you have indicated in character, zoned, or packed fields into your data. For this search, it not only looks for dates in data fields that are explicitly defined in DDS, but also for dates inside larger fields and dates defined in multiple contiguous fields (that is, a date defined by one field for the year, one field for the month, and one field for the day). Date data type fields are also detected by the tool and reported.

After finishing the analysis, SEARCH2000 displays a list of candidate date fields. Then, you can delete any non-date fields that the tool has found or add fields the tool may have missed.

You can also run the search again with different parameters.

3.2.2 Reports

After all the date fields have been identified and confirmed, SEARCH2000 produces the following report, detailing files and programs impacted by the date data:

- Date fields found in user files and libraries
- Impacted files in user libraries
- Files with date fields referenced in user programs
- Impacted programs in user libraries
- Impacted lines of code in user programs
- Impacted program areas and fields

These reports assist you with the following tasks:

- Estimating the size of the work effort
- Deciding whether to use a conversion tool or to make changes manually
- Identifying the files and programs requiring review if you decide to work manually

For detailed information about the reports, refer to Section 3.5, “Work with Reports” on page 43.

3.2.3 Interface to BYPASS2000

If you decide to use BYPASS2000 to convert your application, SEARCH2000 prepares a file to interface with BYPASS2000 (the description of this file is in Appendix A, “Interfacing to BYPASS2000” on page 149).

In addition, by using a SEARCH2000 repository, BYPASS2000 at V3R1M2 generates its own file called HSDATDFN, which is an enhanced version of HSDATDFI. The HSDATDFN file allows seeding of packed date fields of a program described file to be exported and used in BYPASS2000. The HSDATDFI file does not possess this feature.

This greatly simplifies the use of BYPASS2000 because it minimizes the amount of manual work you must do.

3.3 SEARCH2000 Tutorial

This section provides information about setting up the SEARCH2000 environment and the reports the analysis process produces.

3.3.1 Setting Default Parameters

Before you analyze any files, you can choose to change the default parameters for analyzing files.

From the Work with Files display, press **F16**. Figure 19 on page 32 shows the default parameter values that are used for analysis. These parameters help SEARCH2000 find the date fields.

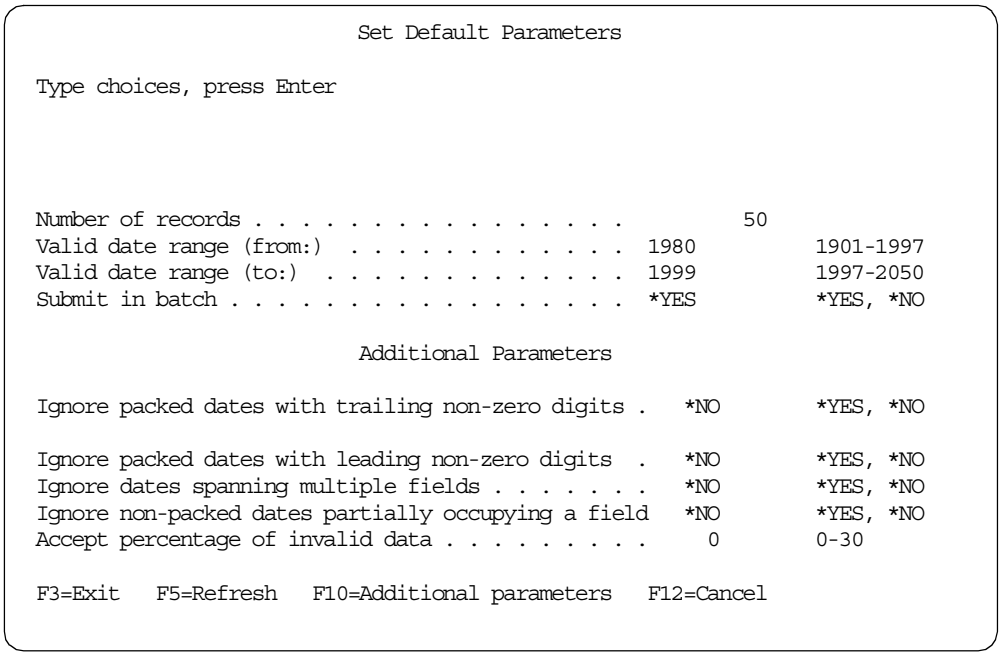


Figure 19. Set Default Parameters

You can change the default values for the following parameters:

- **Number of records** — Specifies the number of records that are sampled during the analysis. If it is greater than the number of records in a database file, all the records in that file are analyzed. Otherwise, the records to be analyzed are chosen from the file at even intervals. In most cases, approximately 500 to 1000 records are enough to find the date fields. Remember that the analysis time grows with the number of records analyzed.
- **Valid date range (from - to)** — Tells SEARCH2000 the range of dates where the majority of dates in your database files belong. This is primarily to avoid getting too many *false* selections for a two-digit year field. Because SEARCH2000 looks for dates based on their values, if it searched for all possible two-digit year values, almost any numeric field appears to be a potential year. Therefore, limiting the range of expected date values allows the tool to be more selective in looking for two-digit year values. SEARCH2000 finds many dates if they are made up of more than a year value (YMD or Julian), even if they are out of the valid date range. The default range (1980 - 1999) seems to work well with most database files.
- **Submit in batch** — Specifies whether to process the job in batch. Leave the default value of *YES. The analysis process can take a long time.

If you press **F10**, the following additional default parameters are shown:

- **Ignore packed dates with trailing nonzero digits** — If you are sure there are no dates with trailing nonzero digits embedded in packed fields in your database files, change the default value from *NO to *YES.
- **Ignore packed dates with leading nonzero digits** — If there are dates with leading nonzero digits embedded in packed fields in your database files, change the default value from *YES to *NO.

- **Ignore dates spanning multiple fields** — If you are sure that all the dates in your database files are completely within one field, change the default value from *NO to *YES.
- **Ignore dates occupying a partial field** — If you are sure that all the dates in your database fields are not inside larger fields, change the default value from *NO to *YES. Notice that if your data is not externally described, you should leave this value as *NO.
- **Accept percentage of invalid data** — If you have some fields that have some data that may not be valid date values (for example, month values greater than 12 or day values greater than 31), you can specify that a small percentage of data that is not valid be overlooked when browsing for dates. We recommend that you leave this number very small to avoid having too many false dates found.

After you finish setting the analysis parameters, press **Enter** to save the changes and then press **F3** to exit. If you inadvertently press F12, you lose any changes that you have made.

The Work with Files display is shown again.

If you want to set different analysis parameters for a particular file, you can select option **1** for that file, and then press **F4** (do not press Enter). The Set Parameters for Analysis display is shown for just that file. The available parameters and their meanings are the same as in setting the default parameters in Figure 19 on page 32.

You can change the parameters the same way you previously changed the default parameters. The changes are effective only for that file, for this specific analysis job.

After submitting the analysis for the file, the Work with Files display is shown again.

3.3.2 Analysis Process

Select option **1** for the files you want SEARCH2000 to analyze on the Work with Data display. The analysis job is submitted to batch.

To see the results of the analysis, the process flag must be set to **Y** and the *Found* count must be greater than zero (number of possible date fields found).

3.3.2.1 Work with Dates for a File

Select option **5** (Work with dates) on the Work with Files display to obtain the list of possible date fields found for a file shown in Figure 20 on page 34.

```
                              Work with Dates
                              File    ORDERHDR
                              Library BPMASTER
 Type choices, press Enter.                        Position to . . .
  4=Delete/undelete  6=Expand/not expand
    Num. Date Date        Out of Min      Max      Field      Digits        Inf
Opt Type Pos. Format      Range  Date     Date     Name       Be Aft  D Exp Pro
 _   P     11 MDY             0  052996   053096   ORHDTA     01 00       0  PGM
 _   P     15 MDY             0  052997   053097   ORHDLY     01 00       0  PGM
 _   A     26 YMD             0  960121   960121   SRNBR      01 02       0  PGM


                                                                        Bottom
 F3=Exit   F5=Refresh   F6=Create   F12=Cancel   F16=Show contents (DSPPFM)
 F17=Top   F18=Bottom   F19=Show/hide deleted    F24=More keys
```

Figure 20. Work with Dates for a File

Select option **4** (Remove) on the Work with Files display to eliminate the fields that do not contain dates. If you decide to undelete any fields, press **F19** (Show/hide deleted) before using option 5 (Undelete) for the fields you choose.

The columns on the Work with Dates display, shown in Figure 20, are explained in Table 6.

Table 6. Columns in Work with Dates Display

Column Title	Description
Opt	Option you use for the possible date field
Num Type	Numeric type of the field (P = packed, A = zoned decimal or alphanumeric)
Date Pos.	Position of the record where the possible date field begins
Date Format	Date format fo the possible date field
Out of Range	Percentage of date values found outside the range
Min Date	Minimum value found for a date in this possible date field
Max Date	Maximum value found for a date in this possible date field
Field Name	Name of the possible date field
Digits Be Aft	Number of digits in the field before the beginning of the date value and the number of digits after the end of the date value
Deleted Flag	Flag for deleted fields. Value is Y if possible date field has been deleted
Exp	Expansion type of the field. Value is 0 for expansion, 1 for no expansion
Inf Pro	Information provider: PGM means SEARCH2000 found the possible date; USR means the user inserted the date field

You can use several function keys on the Work with Dates display shown in Figure 20 on page 34. Press **F24** to show more keys.

Press **F21** (Show contents - RUNQRY) to see the Display Report display with the results of a query run over the file. A sample report is shown in Figure 21. This report can help you discard non-date fields.

```
                              Display Report
                                          Report width . . . . . :      63
Position to line  . . . . .             Shift to column  . . . . . .
Line   ....+....1....+....2....+....3....+....4....+....5....+....6...
       ORHNBR  CUSNBR   ORHDTA    ORHDLY           ORHTOT   SRNBR
000001 00001   00003    52,996    52,997             5.65   A960121C
000002 00002   00003    52,996    52,997             5.65   A960121C
000003 00003   00003    52,996    52,997             5.65   A960121C
000004 00004   00003    52,996    52,997             7.00   A960121C
000005 00005   00003    52,996    52,997             7.00   A960121C
000006 00006   00003    52,996    52,997             7.00   A960121C
000007 00007   00003    52,996    52,997             7.00   A960121C
000008 00008   00003    52,996    52,997             7.00   A960121C
000009 00009   00003    52,996    52,997             7.00   A960121C
000010 00010   00003    52,996    52,997            10.50   A960121C
000011 00011   00003    53,096    53,097             5.65   A960121C
000012 00012   00003    53,096    53,097             5.65   A960121C
000013 00013   00003    53,096    53,097             5.65   A960121C
000014 00014   00003    53,096    53,097             5.65   A960121C
****** ********  End of report  ********

                                                                  Bottom
F3=Exit      F12=Cancel      F19=Left      F20=Right      F21=Split
```

Figure 21. Display Report - File Query

Notice that, in this case, the SRNBR field that was found by SEARCH2000 does not appear to be a true date field. It contains a date value in the middle of some other characters. Within our application, we do not want to treat this field as a date.

We have some options to handle this situation. You can delete the SRNBR field from the list of possible date fields with option **4**. There is another option we can explore by using some of the search parameters.

Notice that, in this case, this field is not selected as a date field if we choose to ignore dates that do not fully occupy a complete field. There are extra characters before and after the *date* information. If we want to ask SEARCH2000 to ignore these kinds of situations in the files, we do not include this field on the list of possible date fields.

To make this work, go back to the Work with Files display and select option **1** to work with the ORDERHR file again. This time, press **F4** to prompt for search parameters. The display in Figure 22 on page 36 is shown.

```
                        Set Parameters for Analysis

Type choices, press Enter

File  . . . . . .> ORDERHDR
  Library . . . . . .>   BPMASTER
Member  . . . . . .  *FIRST       Name, *FIRST
Number of records . . . . . .          50
Valid date range (from:)  . . . . . .  1980        1901-1997
Valid date range (to:)  . . . . . .  1999          1997-2050
Submit in batch . . . . . .  *YES                 *YES, *NO

                           Additional Parameters

Ignore packed dates with trailing non-zero digits .   *NO        *YES, *NO

Ignore packed dates with leading non-zero digits  .   *NO        *YES, *NO
Ignore dates spanning multiple fields . . . . . .     *NO        *YES, *NO
Ignore non-packed dates partially occupying a field   *YES       *YES, *NO
Accept percentage of invalid data . . . . . .          0        0-30

F3=Exit   F5=Refresh   F10=Additional parameters   F12=Cancel
```

Figure 22. Set Parameters for Second Analysis of ORDERHDR

On this display, change the parameter value for *Ignore dates occupying partial fields* from *NO to ***YES**. Then, submit the analysis of this file.

When the analysis completes, take the option to Work with Dates for the ORDERHDR file. This time, you see that the SRNBR field is ignored because it occupied only part of the field. See Figure 23 for the results of this new analysis.

```
                              Work with Dates
                              File    ORDERHDR
                            Library BPMASTER
 Type choices, press Enter.                      Position to . . .
  4=Delete/undelete  6=Expand/not expand
    Num. Date Date        Out of Min      Max      Field      Digits        Inf
Opt Type Pos. Format      Range  Date     Date     Name       Be Aft  D Exp Pro
     P     11 MDY             0  052996   053096   ORHDTA     01 00       0  PGM
     P     15 MDY             0  052997   053097   ORHDLY     01 00       0  PGM




                                                                          Bottom
 F3=Exit   F5=Refresh   F6=Create   F12=Cancel   F16=Show contents (DSPPFM)
 F17=Top   F18=Bottom   F19=Show/hide deleted    F24=More keys
```

Figure 23. Work with Dates - Second Analysis

3.3.2.2 Work with Fields

To view the fields that are present in your SEARCH2000 environment, select option **19** (Work with fields) from the Work with Environment menu. Using the subsequent menu shown in Figure 24 on page 37, you can select the fields you want to filter out.

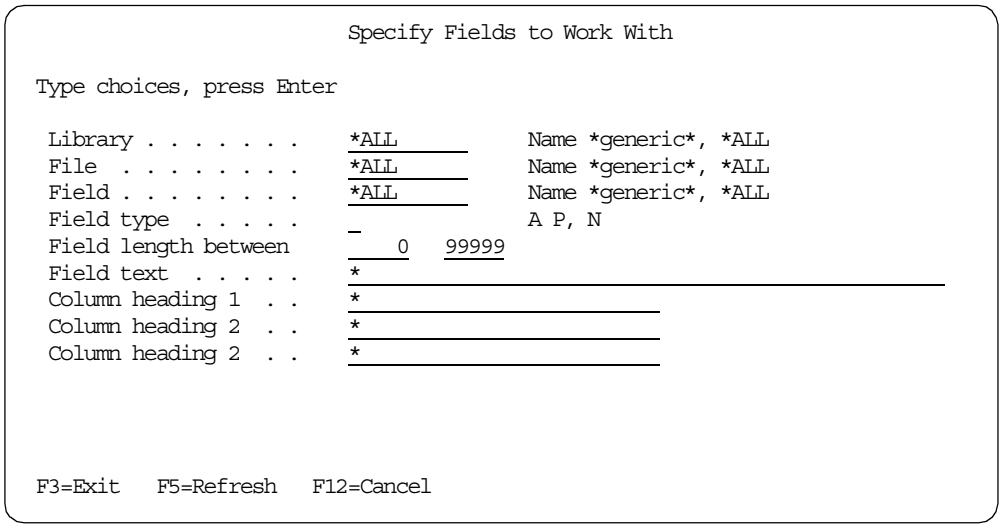


Figure 24. Working with Specific Fields

You can change the values for the following parameters:

- **Library** — This allows you to look at fields from a specific library or libraries. The default is set to *ALL.
- **File** —This allows you to look at fields from a specific file or files. The default is set to *ALL.
- **Field** — This allows you to look at a specific field or fields. The default is set to *ALL.
- **Field Type** — This allows you to look at fields according to type. The options are *A* for alphanumeric fields, *P* for packed fields and *N* for zoned decimal fields. The default is set to blank to indicate that all field types are selected.
- **Field Length** — This allows you to look at fields that have a specific length or lengths. The default is set to 0 to 99999.
- **Field Text** — This allows you to look at fields with a specific text description or text descriptions.
- **Column Heading** — This allows you to look at fields with a specific column heading.

After you finish selecting the parameters, press **Enter** to display the fields that you want to work with, as shown in Figure 25 on page 38.

```
                              Work with Fields

 Type choices, press Enter.                    Position to library .
  1=Work with field  9=Delete assignment       Position to file  . .
  Quick assignment:  2=YMD   3=MDY   4=DMY   5=YM   6=MY   7=YJ   8=Y
    Field      Field Field    Field   Date       Inf
Opt Name       Displ Type     Length  Format     Pro Exp  Library    File
    NAME           1 A            10                      BIRTH2000  CUST
    ADDRES        11 A            20                      BIRTH2000  CUST
    BDAY          31 P(8,0)        5  YY       * PGM  1   BIRTH2000  CUST
    COMP          36 A            20                      BIRTH2000  CUST
    LASTCH        56 P(14,0)       8  MDYY       PGM  1   BIRTH2000  CUST
    CUSTNO        64 P(6,0)        4                      BIRTH2000  CUST
    NAME           1 A            10                      BIRTH2000  CUST6
    ADDRES        11 A            20                      BIRTH2000  CUST6
    BDAY          31 P(6,0)        4                      BIRTH2000  CUST6
    COMP          35 A            20                      BIRTH2000  CUST6
    LASTCH        55 P(14,0)       8  MULT.               BIRTH2000  CUST6
    CUSTNO        63 P(6,0)        4                      BIRTH2000  CUST6
                                                                        More...

 F3=Exit   F5=Refresh   F10=Command line   F11=More info   F12=Cancel
 F17=Top   F18=Bottom
```

Figure 25. Result of Work with Fields

3.4 Object Reference

After all the date fields are established, SEARCH2000 browses your object program libraries and tells you the impact of the date fields on your programs. You have to tell SEARCH2000 the names of the libraries where your program objects reside.

Before you begin loading your copy sources and programs, you must let SEARCH2000 know where your source libraries and Copybook files are. You only need to do this if the source has been moved from the libraries where they were at the time of compile. To do this, select option **11** (Set default libraries) in the Work with Environments menu. As shown in Figure 26 on page 39, you can enter up to 10 source libraries in the list specifying which file in the library contains the COBOL and RPG copy source.

Important

Always remember to run option 11 (Set default libraries) before you run option 3 (Add a program library).

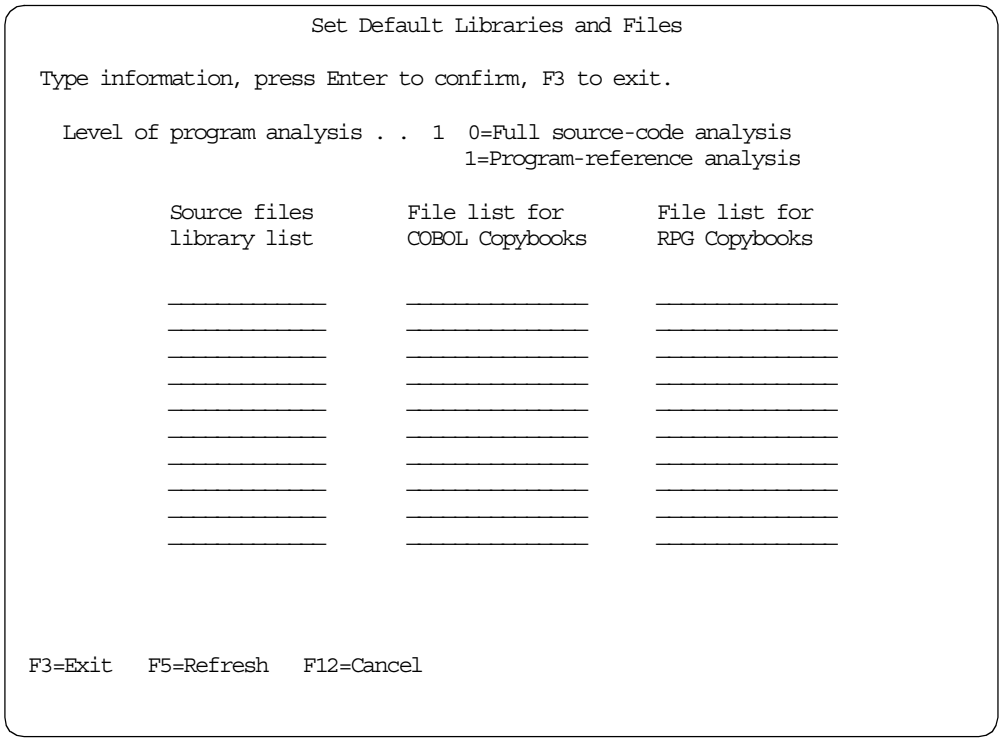


Figure 26. Set Default Libraries and Files

After you have set your default libraries and files by using option 11, select option **3** (Add a program library).

You add these libraries the same way you add your database libraries. Enter the name of your program library. Choose whether to submit the job to batch. If you choose batch, you are prompted for a job queue to use. Press **Enter**. You receive a message confirming that the library is being loaded into the files (of the analysis environment library).

Your program libraries may be the same as your database libraries. However, you must add them as program libraries. You can add as many program libraries as you need.

3.4.1 Analyzing Programs

After you have identified and verified the date fields found by SEARCH2000 through file analysis, proceed with analyzing the program source. The program analysis establishes relationships between programs and the files that it uses. In addition, all lines of code that involve date sensitive information in the program are identified.

After adding the program library (or libraries) to the environment, select option **13** to work with programs and analyze the programs one at a time, as you previously did with the files. This time, the programs are analyzed in a different way. Select option **16** on the Work with Environments display (as shown in Figure 27 on page 40) to have SEARCH2000 analyze all the programs in the environment in a single step. Press **Enter** to analyze all the programs in the environment.

```
                        Work with Environments

Type choices, press Enter.                Position to environment
 2=Add a file library    3=Add a program library   12=Work with files
 13=Work with programs   15=Analyze files          16=Analyze programs

Opt  Environment  Text
 16  TEST         Impact analysis environment library.
 __  TST          Impact analysis environment library.
 __  USENOT       Impact analysis environment library.
 __  XYZ          Impact analysis environment library.



                                                                      Bottom
F3=Exit   F4=Prompt    F5=Refresh   F6=Create   F10=Command line   F12=Cancel
F17=Top   F18=Bottom   F23=More options
```

Figure 27. Work with Environments - Analyze All Programs

Notice that to analyze all of the files in a single step, select option **15**.

To display more options, press **F24**. Select option **7** (Display dates found) to display all the date fields in your program found by SEARCH2000.

If SEARCH2000 did not recognize a field that you know is a date field, you can manually assign a field as a date field. Select option **2** (Assign date to program area) in the Work with Programs menu (shown in Figure 28) to tell SEARCH2000 that you want to make a field a date field.

```
                            Work with Programs

 Type choices, press Enter.                    Position to library
  1=Analyze   2=Assign date to program area    Position to program
  3=Propagate  5=Browse source  6=Display impacted LOC  7=Display dates found
                 Source    Source     Source     Lines     LOC     --File Inf--
Opt  Program     File      Library    Member     of Code P Impact  Used   Dates
 2   CANPGM      QRPGSRC   HODEC16    CANPGM          53 Y      6      3      4
 __  CURSET      QRPGSRC   HODEC16    CURSET          17 Y             2
 __  CUSTSET     QRPGSRC   HODEC16    CUSTSET         23 Y      2      2      2
 __  INTPGM      QRPGSRC   HODEC16    INTPGM          56 Y      6      4      4
 __  REPORT      QRPGSRC   HODEC16    REPORT          36 Y             4      4









                                                                      Bottom
 F3=Exit       F4=Prompt    F5=Refresh    F10=Command line    F11=Toggle
 F12=Cancel    F17=Top      F18=Bottom    F20=Subset list     F23=More options
```

Figure 28. Work with Programs

From the Work with List of Program-Area Fields menu (shown in Figure 29), you can set a field to be a date field. The information is stored in the SEARCH2000 repository.

```
                    Work with List of Program-Area Fields

Source name .   CANPGM          I/O area. . .  *ALL
Source type .   PGM             Field name  .  *ALL
Field type  .                   Field length between   00000  and  99999


Type choices, press Enter.
  1=Switch date assignment on/off

Opt  Field                                Displ. Length  Type  Int  Source
 _    01  BEG                                 1      2  PKD     2  CANPGM
 _    01  END                                 1      2  PKD     2  CANPGM
 _    01  TMPINT                              1      3  PKD     2  CANPGM
 _    01  TEMP                                1      6  PKD     6  CANPGM
 _    01  TMPVAR                              1      6  PKD     6  CANPGM
 _    01  CALPER                              1      2  PKD     2  CANPGM
 _    01  PER                                 1      1  PKD     1  CANPGM
 _    01  CNT                                 1      1  PKD     1  CANPGM
 _    01  CURDAT                              1      6  ZND     6  CANPGM
                                                                    More...
F3=Exit          F5=Refresh          F12=Cancel
F17=Top          F18=Bottom
```

Figure 29. Work with List of Program-Area Fields

To also display the lines of code that contain date sensitive information, select option **6** (Display impacted LOC) in the Work with Programs menu, as shown in Figure 30. This display gives you a general idea of what types of instructions are executed using date information.

```
                  Display Lines of Code Impacted by Dates

Source name . . .   CANPGM

SQNBR *...+....1....+....2....+....3....+....4....+....5....+....6....+....7
00011      C                    MOVE STRDAT    START
00012      C                    MOVE ENDDAT    FIN
00013      C                    MOVELSTRDAT    BEG     20
00014      C                    MOVELENDDAT    END     20
00022      C           END      SUB  BEG       CALPER  20
00039      C                    TIME           LASTCH









                                                                    Bottom
F3=Exit          F5=Refresh          F12=Cancel
F17=Top          F18=Bottom
```

Figure 30. Display Lines of Code Impacted by Dates

The columns on the Work with Programs display are explained in Table 7.

Table 7. Columns in Work with Programs Display

Column Title	Description
Opt	Option you use for the object
Object	Name of the program object
Source File	Name of the source file that contains the source member from which the program object was created
Source Library	Library for the source file
Source Member	Name of the source member
Lines of Code	Number of lines in the source member
P	Has the analysis been processed? Y = Yes, S = Submitted for processing, W = Running now
LOC impacted	Number of lines of code that uses date sensitive information
File Used	Number of files used by the program
Dates	Number of possible date fields found by SEARCH2000

When you press **F11** (Toggle) on the Work with Programs display, the seven rightmost columns are replaced by four columns shown in Table 8.

Table 8. Columns in Work with Object Display

Column Title	Description
Opt	Option you use for the file
File	Name of the file
Library	Library for the file
Rec. Len.	File record length in bytes
Records Available	Number of records in the file
P	Process flag: Value is blank if the file has not been analyzed, "Y" if the analysis has been run, and "W" if the analysis is running in batch at the moment. It is "S" if the analysis has been submitted but not yet processed.
Found	Number of date fields in the file (found by SEARCH2000 or included by you)
File text	Text in the file description
File Attr.	Attribute of the file (PF, LF, DSPF, PRTF,...)

Option 8 (Display description) displays the full description of the program object.

Option 12 (Work with) takes you to the Work with Referenced Files display shown in Figure 31 on page 43.

```
                          Work with Referenced Files
                            Library  HODEC16
                            Program  CANPGM
 Type choices, press Enter.                        Position to . . .
  1=Analyze  5=Work with dates  6=Work with fields
                         Rec.  Records                                 File
Opt File       Library    Len.  Available   P  Found  File text         Attr.
 _  BANKDSP    HODEC16       0          0         0                     DSPF
 _  CUST01L    HODEC16      66          6  Y      2  logical file for   LF
 _  DB         HODEC16      36          6  Y      2  dds for moneypgm1  PF
                                                                       Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F17=Top   F18=Bottom
```

Figure 31. Work with Referenced Files

The contents of the columns in the Work with Referenced Files display are explained Table 9.

Table 9. Columns in Object Referenced Files Display

Column Title	Description
LIbrary	Library for the program object
Attribute	The type of the program object: RPG, CBL, or CLP
Size	The size of the program object in bytes
Text	The text in the program object description

The following options are on this display:

- **Option 1=Analyze** — The file analysis is submitted to batch.
- **Option 5=Work with dates** — The Work with Dates display is shown in Figure 20 on page 34.
- **Option 6=Work with fields** — A list of all the fields in the file is shown.

3.5 Work with Reports

Several reports are included in SEARCH2000. From the SEARCH2000 Main Menu, select option **10** for the Reports Menu. Enter the name of your environment when prompted. The Reports Menu display in Figure 32 on page 44 is shown.

```
QCKUQMNU                  SEARCH2000 Report Menu

Select one of the following:

  1. Date fields found in user files/libraries
  2. Impacted files in user libraries
  3. Files with dates referenced in user programs
  4. Impacted programs in user libraries
  5. Impacted lines of code in user programs
  6. Impacted program areas and fields



Selection or command
===> _____________________________________________________________________

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
F13=Information Assistant  F16=AS/400 main menu
```

Figure 32. Reports Menu

Options on this menu allow you to print or display four different queries that summarize the information found by SEARCH2000. Each option is described in the following text.

Date fields found in user files/libraries
The report is ordered by library, file, and field. The report lists the following information for each field:

- Date format
- Type of field
- Field position in the record
- Offset of the date within the field
- Digits before and after the date in the field

The report gives totals of date fields per file, per library, and final.

Impacted files in user libraries
The report is ordered by library and file. The query lists the following information for each file:

- Number of possible date fields
- Impact flag (flag value is 1 if the file contains date fields, and it is 0 if there are no date fields in the file).

The report gives totals of files and of impacted files per library and final.

Files with dates referenced in user programs
The query output is ordered by library, program, and file. The query lists the following information for each impacted program:

- File name
- File library
- File attribute
- Number of date fields in the file

The report gives totals of files with dates per program and of impacted files per library.

Impacted programs in user libraries
The query output is ordered by library and program. The following information is provided for each program:

- Program attribute
- Number of files used
- Number of date fields in the files
- Lines of source code
- Impact flag (flag value is 1 if any file used by the program contains date fields, and it is 0 if there are no date fields in any file used by the program).

The report gives a total of impacted programs and programs per library.

Impacted lines of code in user programs
The report is ordered by library and program name. The following information is provided for each program:

- Number of LOC that uses date fields in each program and in total
- Lines of source code that uses date fields
- Lines of source code
- Each location of the source code that uses date fields

Impacted program areas and fields
The report is ordered by library and program name. The following information is provided for each program:

- I/O Areas containing date sensitive information
- Number of I/O Areas containing date fields in a program and in total

3.6 Removing Libraries

The following two options on the Work with Environments display allow you to remove libraries from your application analysis environment:

- **Option 4 (Remove a file library)** — Use this option to remove a database library from your application analysis environment.
- **Option 5 (Remove an program library)** — Use this option to remove a program library from your application analysis environment.

Chapter 4. BYPASS2000 Product Overview

BYPASS2000 for AS/400 is a tool to assist you in migrating AS/400 RPG and COBOL applications to properly handle the transition to the Year 2000. Unlike other Year 2000 tools, BYPASS2000 uses application understanding technology to track the affected code, significantly reducing the amount of manual work required.

The BYPASS2000 for AS/400 tool also assists AS/400 customers and business partners in addressing the Year 2000 challenge. Application developers and technical support staff can convert their AS/400 RPG and COBOL applications to properly handle four-digit years throughout their applications with a minimal amount of manual intervention.

This tool uses program-understanding technology to locate and change areas of an application that need to be changed to accommodate four-digit years. After completion, the program source and data files can be recompiled and tested to ensure the application continues to execute as required.

BYPASS2000 supports AS/400 applications written in RPG and COBOL, as well as related command language programs and AS/400 database files. It runs on Version 3 Release 1 of OS/400 and later.

4.1 Overview of BYPASS2000

The following overview shows the various features of BYPASS2000:

- Supports AS/400 RPG and COBOL conversions.
- Supports several additional national languages.
- Supports double-byte systems.
- Converts database and programs from two-digit to four-digit years or allows windowing techniques.
- Allows displays and reports to remain unchanged.
- Automatically generates code to map from four-digit database fields to two-digit display and print fields.
- Converts the majority of changes required automatically.
- Clearly highlights areas of the application that the tool cannot convert automatically.
- Clearly documents all changes made to the application.
- Generates programs to move data from old files to new database format.
- Provides date simulation logic to assist with application testing.
- Generates database triggers to ensure data integrity following conversion.
- Runs on OS/400 Version 3 Release 1 and later.

4.2 New Features in BYPASS2000, V3R1M2

The following features were added in BYPASS2000, V3R1M2.

- Performance improvements (up to 50%)
- Full windowing technique available
- Date fields with century flag supported
- Online help text and level II messages improvements
- DDS source generation possible from Copybook or file object
- BYPASS2000 browser — An SEU type browser which includes function, such as letting you display the date-field properties or propagation tree.
- Ability to analyze OCL programs to obtain //FILE, //LOAD and OVRDBF and store it in the repository

4.3 A More Detailed View of BYPASS2000

There are five main phases to a successful conversion using BYPASS2000. These phases are analysis, seeding, propagation, conversion, and testing.

The first phase, analysis, begins by identifying the conversion environment. You specify the location of the programs and files to be converted. During the second phase, known as seeding or date assignment, all of the date fields that require conversion are identified. BYPASS2000 takes this information and builds a comprehensive repository of information about the application and the data used within the application.

After the repository is built, the third phase, propagation, is started. BYPASS2000 uses program understanding technology to trace the date fields stored in the database (and in program-described files) and identifies the areas within the application that need to change.

BYPASS2000 actually analyzes the use of year data at the memory level within the application. BYPASS2000 follows the areas that are impacted by the year data (whether they are referenced or overlaid) until all areas within the application are identified.

The fourth phase, conversion, reconstructs the application from the original source, making adjustments where required. All modifications made to the source are flagged for easy identification. Changes required for command language programs and data description specifications for file layouts are also generated automatically by the tool. In addition, programs to convert the database files from their original format to the new format are generated by the tool.

The final phase, after conversion, is testing. This is a critical part of the Year 2000 conversion process and must be planned. The tests required to ensure the applications perform as required vary greatly, depending on the application, and must be carefully planned.

BYPASS2000 has facilities to assist testing. BYPASS2000 has data conversion utilities to allow you to change the date inputs to your application without actually changing fields in your database. This way, you can test specific critical dates

(January 1, 1999, December 31, 1999, January 1, 2000, and so on) without generating special test databases.

BYPASS2000 also supports the generation of database triggers that can be attached to the converted files to ensure that only valid dates are input to and output from the database.

4.3.1 Operating Environment

The following items are required in the operating environment:

- Any model of an AS/400 system running Version 3 Release 1 or later
- ADTS (5763-PW1 or 5716-PW1)

4.3.2 Restrictions

The following restrictions apply:

- COBOL is supported in Version 3 Release 1 Modification 1.
- BYPASS2000 does not support ILE Languages.
- Version 1 Release 1 is English only. National language support is provided in Version 3 Release 1.

4.3.3 Ordering Information

IBM BYPASS2000 for AS/400 can be ordered with number 5697-D11 (Version 3 Release 1 Modification 2).

IBM BYPASS2000 for AS/400 is included in several solution packs (software and hardware and software).

4.3.4 Technical Support

There are several ways to obtain support on the internet. There is the BYPASS2000 forum located at:
news://news.software.ibm.com/ibm.software.bypass2000.as400

To obtain more information, visit the IBM Web site at:
http://www.software.ibm.com/ad/as400/bypass/

You can also visit the designers of BYPASS2000 in Italy on the Hal Web site, at:
http://www.halinfo.it

To obtain more information in North America, visit the Web site at:
http://www.halna.com

For more information about APARs and PTFs, visit the Web site at:
http://www.halna.com/Products/SupportBP.html

4.4 How BYPASS2000 Works - Overview

BYPASS2000 supports the conversion of the following data:

- Field reference files (dictionaries)
- Physical and logical files
- External data structures
- Display and printer files

- RPG/400 (RPG, RPG38, RPT)
- COBOL/400 (CBL, CBL36, CBL38)
- SQL programs (SQLRPG, SQLCBL)
- Control language programs (CLP)

4.4.1 Storage Mapping Propagation

Storage mapping propagation locates all fields related to date fields within program logic. The tool analyzes program code to build a complete storage map of all data used or defined in the program. You identify all date fields only in database files. The tool uses the storage map (not field names) to locate all areas of storage in the program related to the date fields, for example, work fields, data structures and DS sub-fields, display or printer fields, and program parameters.

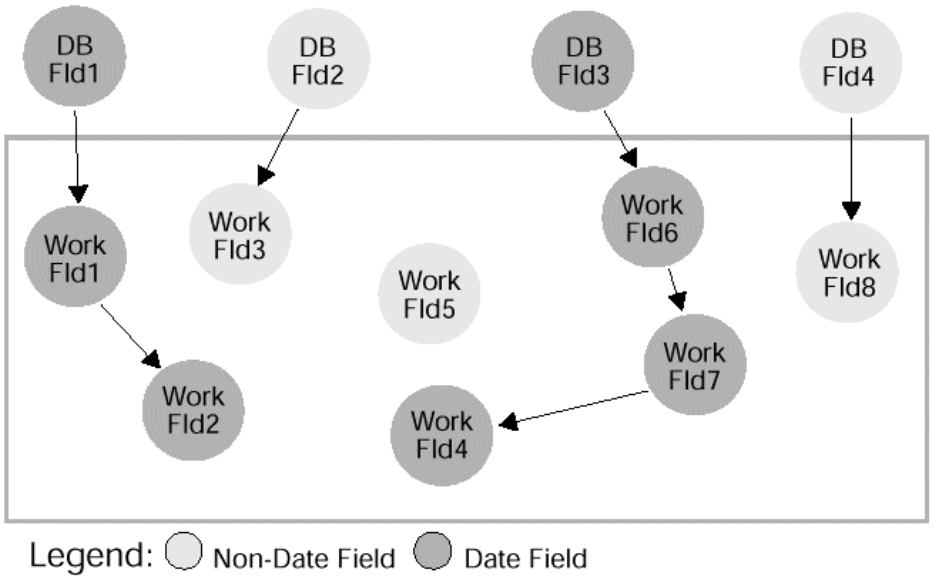


Figure 33. Propagation: Searching for Affected Fields

Figure 33 shows how BYPASS2000 propagation technology is able to find date fields defined within programs by means of understanding the relationships between fields (or storage) within a program. The arrows represent some kind of association. For example, a field is moved from one to another or is compared to another. By tracing all of the relationships among the pieces of storage, the tool identifies all of the fields that contain year data in preparation for modifying the code to support four-digit years in the database files.

Propagation does not stop at the boundary of the program. Propagation can extend to externally described files and parameters passed to other programs.

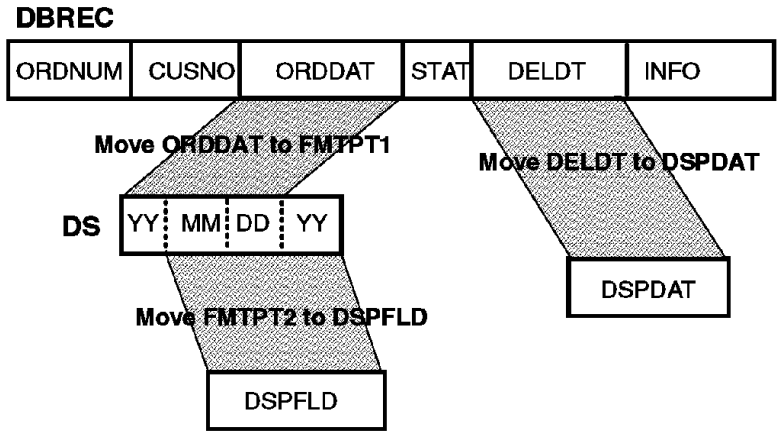


Figure 34. Propagation Can Be Difficult

Figure 34 shows how program understanding is crucial to correctly handling date propagation. In the move on the right, the propagation is quite simple. Many tools, even those that use only field names to associate fields, find this.

In the two moves on the left, the propagation is more complex. The ORDDAT field is first moved to a sub-field of the data structure that contains the current YYMMDD structure of the date. Then, the YY portion is moved to the last two positions of the data structure. Finally, the restructured MMDDYY sub-fields of the data structures are moved to a display field.

Notice that the same area of storage may be referred to by different names. In this example, the sub-fields labeled as MM and DD are referred to indirectly as part of FMTPT1 and FMTPT2.

This example further complicates the process for tools using field names to find date files because FMTPT1 and FMTPT2 do not have names that are obviously date-related. With temporary work fields created by each programmer, nonstandard naming conventions are much more likely. BYPASS2000 uses its storage map to locate these fields and finds them because they are related to the ORDDAT field from the database.

Note: Propagation works not only with moving fields as shown here, but with any association of the fields. Another common example is comparison of the fields (ORDDAT > YMD).

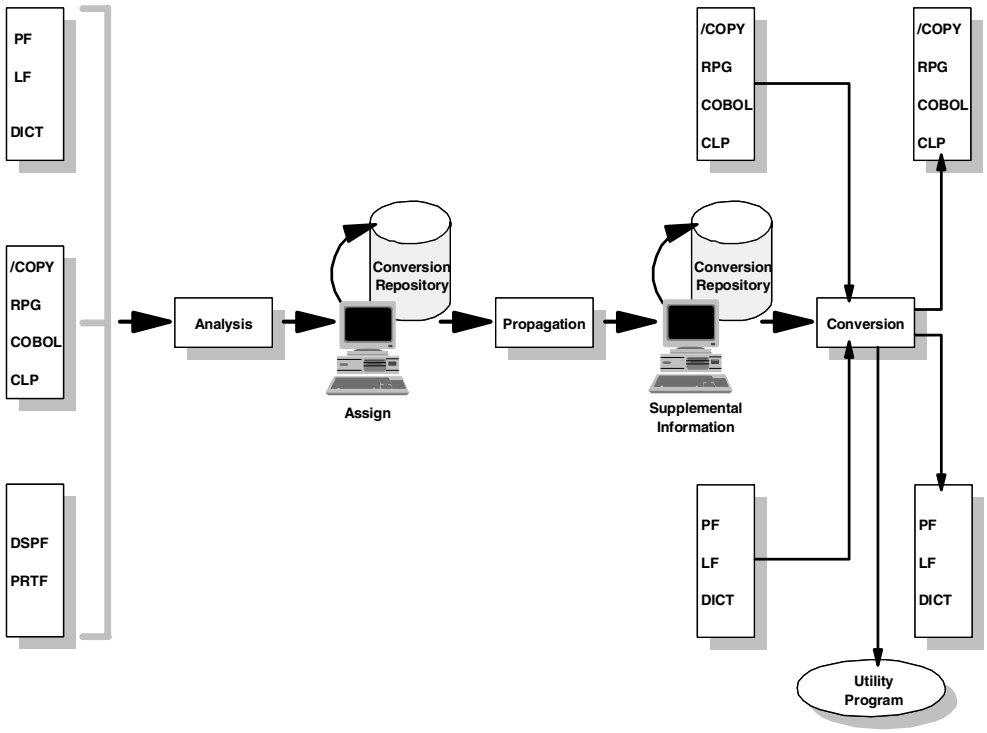


Figure 35. BYPASS2000 Process Overview

Figure 35 shows an overview of the major processes that are part of conversion. The three long vertical boxes represent the three major batch processes. Take a look in more detail at each of the major steps to converting applications.

To begin an application conversion, you set up a conversion environment to identify and classify the components of the application to be converted. This is a manual process.

The main purpose of this function is to create a repository library and to specify which libraries contain the source to be converted. BYPASS2000 starts a function that lets you specify the old-source libraries, the old-objects libraries, new-source libraries, and the new-objects libraries.

4.4.2 The Analysis Process

The analysis process is made up of the following steps within the BYPASS2000 tool:

1. Load AS/400 database information
2. Memory-level analysis

The tool creates a complete storage map of all data used in the programs and their relationships.

The first step is the analysis of the database information. You perform this step using the Load AS/400 database information menu option. The analysis starts from the list of old-object libraries.

The second step is the analysis of the source code (DDS, RPG, and so on) using the Memory-level analysis menu option. This is the analysis of the source

statements related to memory definition and management. Every field definition is stored in the repository with the information about its memory position. Every instruction is stored as a relationship between two or more portions of the memory.

Memory analysis is done in these four parts:

- DDS (Data Description Specifications)
- Copy members (containing file descriptions for program described files)
- SQL table definition source
- Program source

Attention

These procedures are CPU intensive!

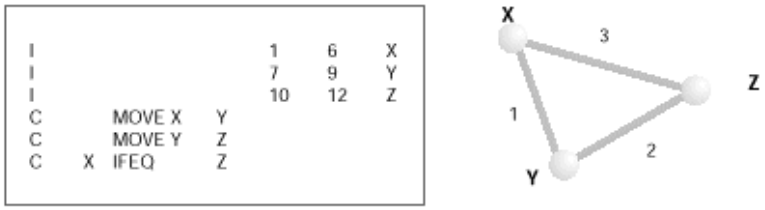


Figure 36. Nodes and Links in the Analysis Phase

The purpose of the analysis is to create a network model of the application, where each node represents a field declared in a source object and each link represents a relationship between a two memory position or nodes.

Analysis is done for all fields and instructions in the application. Date fields have not been identified to BYPASS2000 at this time. The model is built by analyzing program source, files, DDS source, and copies or includes. The model is stored in the BYPASS2000 repository database.

4.4.3 Date Field Assignment

You typically assign only fields from the database as dates. Using this information, together with the knowledge of the language constructs and syntax, BYPASS2000 can identify all of the remaining date impacted fields within the programs. The date field assignment is often called *seeding*. Seeding serves as preparation for the propagation phase that follows. Year-sensitive fields are known as *anchors*.

If you already know the names of date fields used in programs, display files, and printer files, you identify them in the assignment process as well. Doing so may improve the performance of the propagation phase and may result in fewer situations where BYPASS2000 requests information after propagation.

In addition to identifying which fields are dates, it is important to identify the format of the dates so that BYPASS2000 can track the year information.

4.4.4 Propagation

The following topics provide information about propagation and how it is handled by BYPASS2000.

4.4.4.1 Overview

Propagation is probably the strongest feature of BYPASS2000. During this batch process, the tool locates and catalogs all other date fields in the programs from the assignment information received from the previous step. The propagation engine can find date fields even if they are only indirectly referred to in the program code (for example, if they are part of a data structure) or even if they are not referred to in the code at all (for example, record format containing date information is moved to another area). The propagation analysis is also extended to called programs by analysis of passed parameters.

Attention

This procedure is CPU intensive!

Each program in the BYPASS2000 data model is made up of several networks of nodes and links. The BYPASS2000 model also includes relationships between programs, as well as within programs. This is important because some dates may only enter a program by means of a passed parameter.

It may be found that, after the first run of the propagation, the propagation analysis tree is too deep or too large to enable the propagation analysis to complete successfully. Further definition of one or more internal fields and internal program areas as anchors adds an element of certainty that can shorten and speed up the propagation process itself. After providing the information required, it is necessary to delete the old propagation analysis and to perform the propagation analysis again.

4.4.4.2 Application ID

When propagating a very large application, you may encounter many problems that can overwhelm you. Application ID is a new feature of BYPASS2000 which allows you to partition your code so that the process can run in groups, rather than in one giant lump sum. By placing programs that exclusively call each other into groups, you can propagate each group while maintaining the global propagation property. This is different from single program propagation. When you propagate a single program, you are only propagating one level down the tree. This can result in date fields being missed. Application ID solves that problem by traversing through the entire group of programs to find all the date fields. See Chapter 6, “BYPASS2000 Tips and Techniques” on page 97, for more details.

4.4.5 Conversion Output

Converted DDS source contains the new expanded year information. Converted program source not only contains expanded year fields, but also often includes calls to subroutines that perform the necessary *windowing* technique for date data that appears on displays or reports. Remember that BYPASS2000 does not expand display or print files. However, it does generate the extra logic required to logically interpret two-digit years.

BYPASS2000 highlights the following information:

- Changes made in the modified sources
- Lines of code added (for example, the removal of the century information before the output of the display file)
- Requests to manually verify and solve any logical incongruences (for example, the case of an area used both for dates and for non-date related data)
- Instructions whose correctness must be checked (for example, the comparison between a date and a constant)

BYPASS2000 creates the following utility programs for testing the converted application:

Migration Module

Copies the content of the original file to the converted format, adding the century to the widened-year fields.

Test Module

Performs a time shift of the database year-sensitive fields to allow simulating tests in the future.

Data Integrity Verification Module

Tests the logical congruence of the value assumed by the date fields following addition and modification operations of records to the file. BYPASS2000 provides a trigger program that can be used in the test activity.

Chapter 5. BYPASS2000 Tutorial

The purpose of this chapter is to guide you through the basic steps for setting up and using BYPASS2000.

5.1 BYPASS2000 Basic Steps

In Figure 37, you can see the eight basic steps that you must follow when using BYPASS2000.

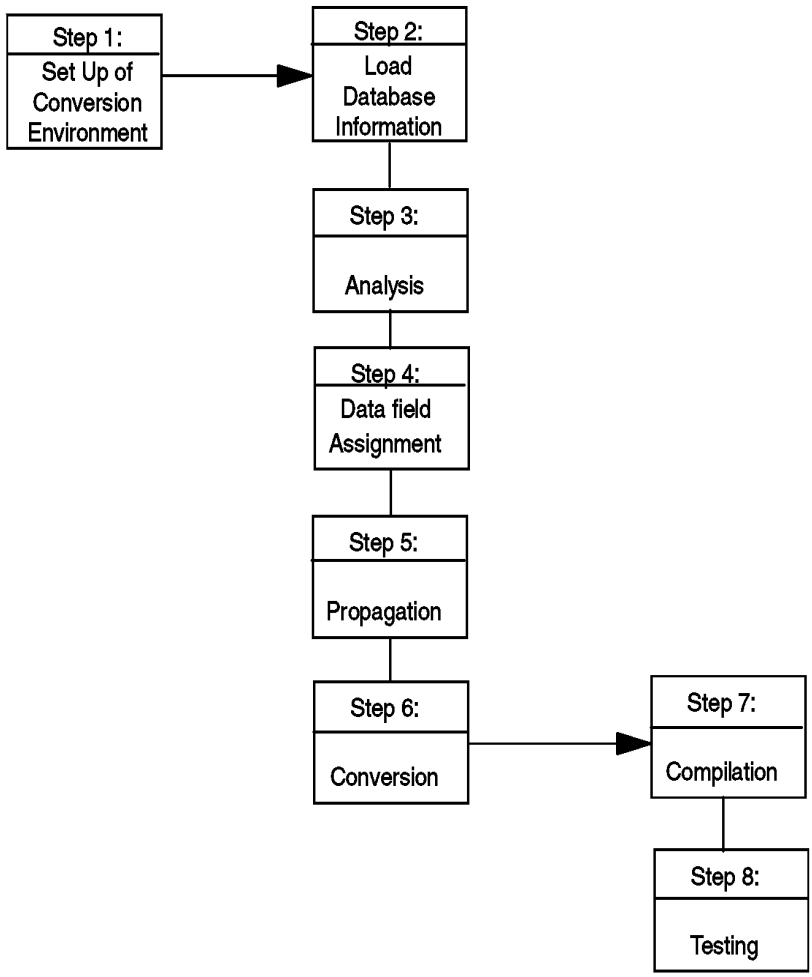


Figure 37. Basic Steps for Setting Up and Using BYPASS2000

Important

BYPASS2000 does not support duplicate names even in different libraries.

In Step 1 through Step 6, you work with the BYPASS2000 tool directly. In Step 7 and Step 8, you work with output produced by BYPASS2000.

Step 1 (Environment Setup) is generally performed only once. Follow the sequence for Step 2 through Step 8. You may need to repeat these steps more than once. When you repeat any of these steps, you must discard the results of any of the following steps you have already run.

5.2 Step 1. Setting Up BYPASS2000 Conversion Environment

For each application conversion, perform the steps in this process only once.

1. At the AS/400 Main menu, issue the following command and press **Enter**:

   ```
   ADDLIBLE QBP2000
   ```

2. Start BP2000 to set up your conversion environment using the following command and press **Enter**:

   ```
   BP2000 *GEN
   ```

   The BYPASS2000 Environment Setup menu appears as shown in Figure 38.

3. Select option **1** and press **Enter**.

```
BP4AMBI                 BYPASS2000 Environment Setup
                                                        System:   S1031GKM
Select one of the following:

     1. Create conversion environment

     3. Customize conversion parameters

     6. Work with object and source libraries
     7. Work with relationships between objects and sources
     8. Customize environment COPYs
     9. Work with field type
    10. Work with additional parameters
    11. Work with system fields

    13. Apply BYPASS2000 software key
```

Figure 38. BYPASS2000 Environment Setup

4. On the next display, shown in Figure 39 on page 59, enter the chosen name for your conversion environment and specify ***NO** for the second parameter (Create default library). Press **Enter**.

```
                    Create Conversion Environment (BPCRTLIB)

 Type choices, press Enter.

 Conversion identifier . . . . .   bp01____         Name
 Create default environment . . .  *no__            *YES, *NO





                                                                       Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 39. Create Conversion Libraries (Part 1 of 3)

Note: If you specify *YES for Create default library parameter, BYPASS2000 creates libraries with names based on the conversion environment name. You are responsible for moving your original source files and database files to these libraries.

The name of the Conversion database library is set to your conversion environment name, in addition to the suffix DB.

Important

Each time you want to work with an existing BYPASS2000 environment, you must type in BP2000 xxxxDB.

```
                    Create Conversion Environment (BPCRTLIB0)

 Type choices, press Enter.

 Create default environment . . . > *NO            *YES, *NO
 Original source library  . . . . > *NONE          Name, *NONE
 Converted source library . . . . > *NONE          Name, *NONE
 Converted object library . . . . > *NONE          Name, *NONE
 Conversion-database library  . . > BP01DB____     Name
 User-database library  . . . . . > *NONE          Name, *NONE
 User-SQL-database library  . . . > *NONE_____     Name, *NONE
 Run in batch . . . . . . . . . .   *YES_          *YES, *NO
```

Figure 40. Create Conversion Libraries (Part 2 of 3)

5. Press **Enter**. A last parameter is added to the display (Job queue). Specify the job queue you want to use, or accept the default of the job queue in your user profile, and press **Enter** (see Figure 41 on page 60). Because many of the batch processes in BYPASS2000 should be completed before proceeding to the next step, you may find it convenient to use a single-threaded job queue for your BYPASS2000 jobs.

```
                Create Conversion Environment (BPCRTLIB0)

Type choices, press Enter.

Create default environment . . . > *NO          *YES, *NO
Original source library  . . . . > *NONE        Name, *NONE
Converted source library . . . . > *NONE        Name, *NONE
Converted object library . . . . > *NONE        Name, *NONE
Conversion-database library  . . > BP01DB       Name
User-database library  . . . . . > *NONE        Name, *NONE
User-SQL-database library  . . . > *NONE        Name, *NONE
Run in batch . . . . . .          *YES          *YES, *NO
Job queue  . . . . . .            *JOBD         Name, *JOBD
  Library  . . . . . .                          Name, *LIBL


                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 41. Create Conversion Libraries (Part 3 of 3)

After a while, the Work with Object and Source Libraries display appears, as shown in Figure 42.

6. When using a customized setup, the entries for the Old Source Library, Old Object Library, and New Source Library are set to *NONE. You must change the corresponding values. For each type of information that you are converting (such as *FILE, CLP, DDS, and RPG), you must specify the location of the old source files and old object libraries, as well as the location for new source files and new objects. To change these values, place your cursor on the value you want to change and then, enter the new name.

```
                Work with Object and Source Libraries

 Type choices, press Enter.                      Position to . . :
   3=Copy    4=Delete     5=Display    *=Reset old information

                                        Old Src Lib  Old Src File  Old Object Lib
Opt Type       Src SEU-Type   Status    New Src Lib  New Src File  New Object Lib
 _  *FILE      DAT PF                   *NONE        *NONE         *NONE
                                        *NONE        *NONE         *NONE
 _  CBL        CPY CBL                  *NONE        QLBLSRC       *NONE
                                        *NONE        QLBLSRC       *NONE
 _  CBL        CPY CBL38                *NONE        QLBLSRC       *NONE
                                        *NONE        QLBLSRC       *NONE
 _  CBL        CPY UNS36                *NONE        QS36SRC       *NONE
                                        *NONE        QS36SRC       *NONE
 _  CBL        PGM CBL                  *NONE        QCBLSRC       *NONE
                                        *NONE        QCBLSRC       *NONE
 _  CBL        PGM CBL36                *NONE        QS36SRC       *NONE
                                        *NONE        QS36SRC       *NONE
                                                                          More...

 F3=Exit     F5=Refresh     F6=Create       Enter=Validate      F9=Confirm
 F12=Cancel  F17=Top        F18=Bottom      F20=One/Two Rows
```

Figure 42. Work with Old and New Libraries

You can see the information for the new libraries and files using option 5 (Display).

Notice that if you have display files and printer files that are in other libraries, you must specify the location of those files. You must create an entry to specify that physical files exist in the subsequent library. The reason is that BYPASS2000 looks in an object library if, and only if, you specify that there are physical files in a library. By adding another entry for physical files, you are letting BYPASS2000 know that you have other objects in another library.This is shown in Figure 43.

7. Press **F6** to add a new entry, which tells BYPASS2000 the location of the other file objects or program sources. If you have type DSPF or PRTF located in another library, type in ***FILE** as shown in Figure 43.

```
                          Create New Entry

Type choices, press Enter.

Type  . . . . . . . . . . . . . : *FILE

F3=Exit     F5=Refresh     F12=Cancel
```

Figure 43. Create New Entry (Part 1 of 2)

8. Press **Enter.** You are prompted to enter your *Old object library* and your *New object library,* as shown in Figure 44.

```
                          Create New Entry

Type choices, press Enter.

Type  . . . . . . . . . . . . . : *FILE

Source type . . . . . . . . . . : DAT

SEU type  . . . . . . . . . . . : PF

Old source library  . . . . . . : *NONE
Old source file . . . . . . . . : *NONE
Old object library  . . . . . . : HALNABP

New source library  . . . . . . : *NONE
New source file . . . . . . . . : *NONE
New object library  . . . . . . : HALY2K



F3=Exit     F5=Refresh     F12=Cancel
```

Figure 44. Create New Entry (Part 2 of 2)

9. Press **Enter** when finished. The display shown in Figure 45 on page 62 appears.

```
                    Work with Object and Source Libraries

 Type choices, press Enter.                          Position to . . :
   3=Copy    4=Delete     5=Display    *=Reset old information

                                                 Old Src Lib  Old Src File  Old Object Lib
Opt Type        Src SEU-Type   Status Old Src Lib  Old Src File  Old Object Lib
 _  *FILE       DAT PF                *NONE        *NONE         *NONE
                                      *NONE        *NONE         *NONE
 _  *FILE       DAT PF         NEW    *NONE        *NONE         HALNABP2
                                      *NONE        *NONE         HALY2K
 _  CBL         CPY CBL               *NONE        QLBLSRC       *NONE
                                      *NONE        QLBLSRC       *NONE
 _  CBL         CPY CBL38             *NONE        QLBLSRC       *NONE
                                      *NONE        QLBLSRC       *NONE
 _  CBL         CPY UNS36             *NONE        QS36SRC       *NONE
                                      *NONE        QS36SRC       *NONE
 _  CBL         PGM CBL               *NONE        QCBLSRC       *NONE
                                      *NONE        QCBLSRC       *NONE
                                                                          More...

 F3=Exit     F5=Refresh     F6=Create       Enter=Validate      F9=Confirm
F12=Cancel F17=Top    F18=Bottom  F20=One/Two Rows
```

Figure 45. Work with Object and Source Libraries

Notice that you still must specify where your first set of data files are.

If you plan to use the test migration feature, BYPASS2000 needs to use its migration programs to correctly migrate your data, as well as alter the dates in your files. Some modification may be needed to the sources for the test migration routine to work properly. One option is to change the member source located in QBP2000/QRPGSRC and recompile it. However, this is not safe because you are changing BYPASS2000 sources globally. If you have another project using BYPASS2000, then the changes made to the test migration routine will be used inadvertently by that project.

A better way is to have duplicate BYPASS2000 sources in your user library. As shown in Figure 46, you can have the BYPASS2000 sources for test migration in your environment. You can alter the copy sources without the risking accidental change to the BYPASS2000 original files, which may affect another project using BYPASS2000. The best way to do this, besides making direct changes to the copy sources, is to select option **8** (Customize Environment COPYs) in the Environment Setup Menu. For example, this option, lets you change the time window your programs use to *window* dates after conversion.

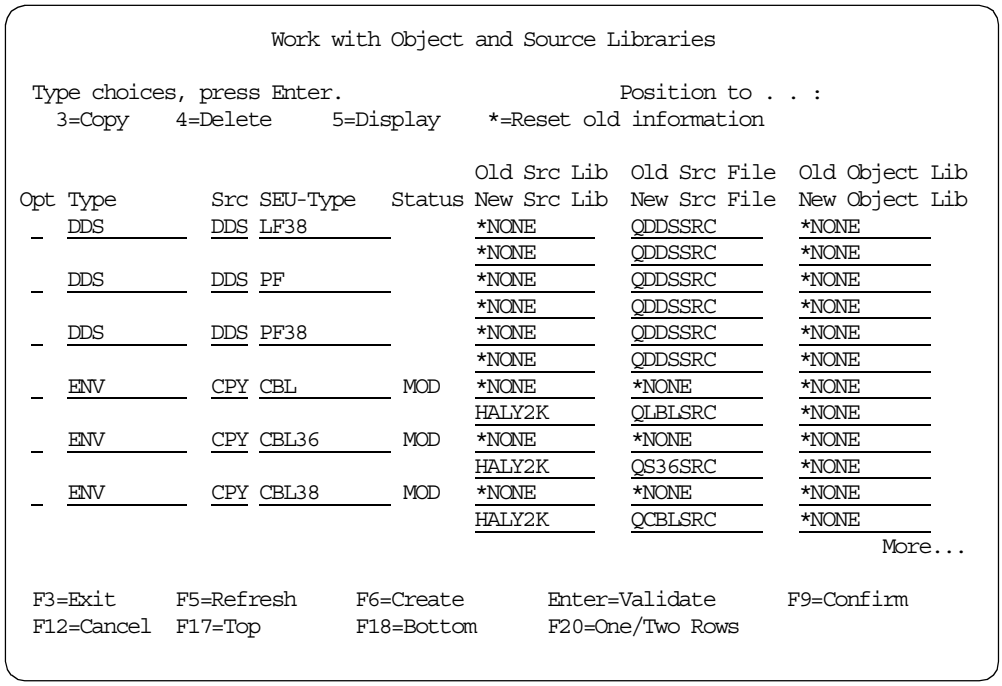


Figure 46. Work with Object and Source Libraries

Important

If you do decide to have BYPASS2000 source members copied into your new library, you must fill in all the entries for TYPE ENV in your enviroment setup.

10. When you finish adding the entries for all of your file objects, file sources and program sources, press **F9**. The Customize Conversion Parameters display appears, as shown in Figure 47 on page 64.

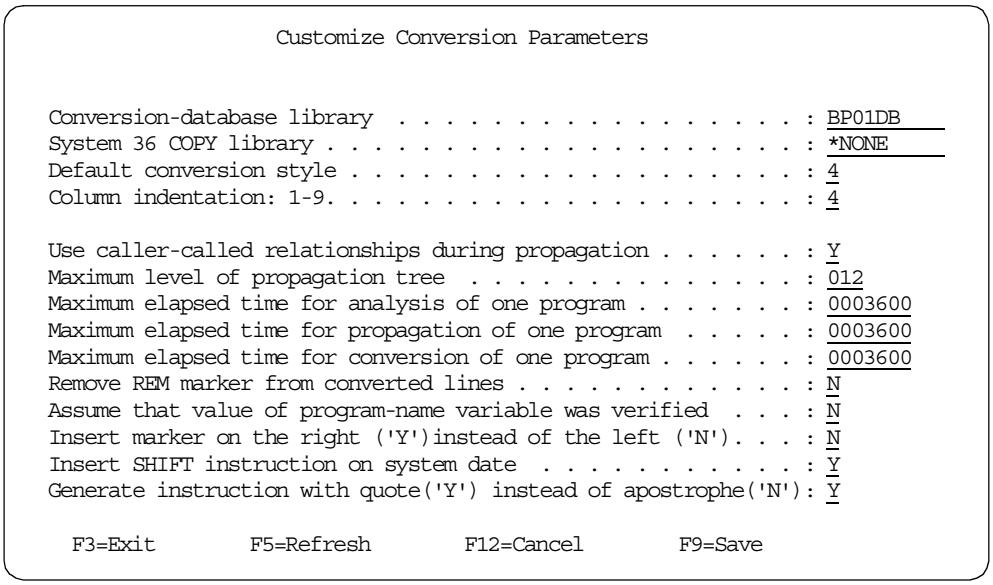


Figure 47. Customize Conversion Parameters

After you have made any required changes, press **F9**.

11. You are asked to confirm the conversion environment creation as shown in Figure 48. Press **F9** to confirm.

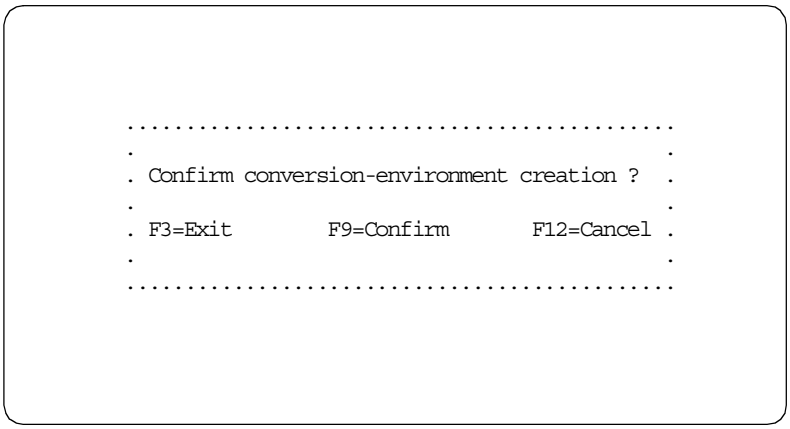


Figure 48. Confirm Creation

12. The display in Figure 49 on page 65 is shown, informing you that the creation has been submitted. Press **F9** to continue.

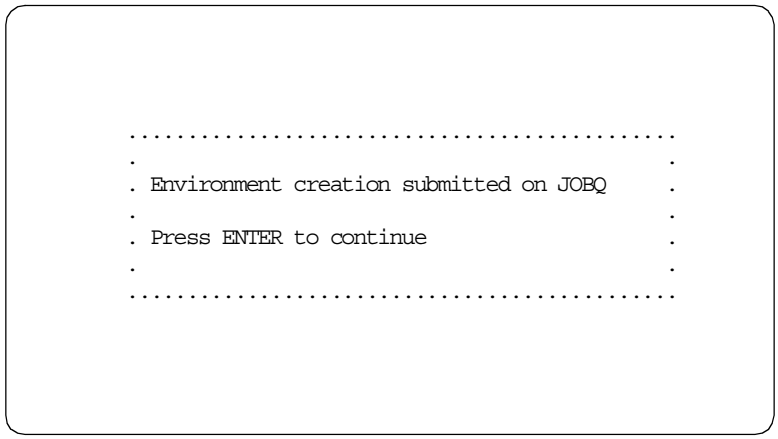


Figure 49. Creation Submitted

After your environment is created, if you want to change any of the information you specified here, select option **22** from the BYPASS2000 Environment Setup menu shown in Figure 38 on page 58. For example, if you must change or add some source information or change some of the conversion parameters, you can change them even after the conversion environment is created.

5.3 Step 2. Loading AS/400 Database Information

In this step, BYPASS2000 analyzes all the database files found in the old object library specified for your data files in the previous step.

BYPASS2000 finds the following information and loads it into the repository:

- Structure of the record format of the files
- Relationships between physical and logical files
- Structure of access paths
- Relationships between files and source members

Important

Make sure your file objects match your file sources and that you are using the most current version of them (especially in relation with field reference files).

At this point (and before running any BYPASS2000 program), you must ensure QBP2000 is in your library list. Perform the following steps:

1. Type the following command and press **Enter**:

   ```
   ADDLIBLE QBP2000
   ```

2. Type the following command and press **Enter**:

   ```
   BP2000 xxxxxDB
   ```

   Where `xxxxx` is your conversion environment name (For more information, refer to Section 5.2, “Step 1. Setting Up BYPASS2000 Conversion Environment” on page 58).

3. Select option **2** (Load database information) from the BYPASS2000 main menu, as shown in Figure 50 on page 66.

```
BP2000              BYPASS2000 for AS/400 - Main Menu
                                                        System:   S1031GKM
Select one of the following:

     1. Environment setup
     2. Load database information
     3. Memory-level analysis
     4. Date-field assignment
     5. Propagation analysis
     6. Application conversion

    70. Work with submitted jobs
    71. Work with all spooled files

    90. Sign off




                                                                    Bottom
Selection or command
===>
F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 50. BYPASS2000 Main Menu

4. The display in Figure 51 is shown. Press **Enter** to accept the defaults.

```
                    User-database information (BPLODASDB)

Type choices, press Enter.

Environment library  . . . . . *LIBL        Name, *LIBL, *CURLIB
Run in batch . . . . . . . . . . *YES        *YES, *NO



                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 51. Load AS/400 Database Information (Part 1 of 2)

5. The display in Figure 52 on page 67 allows you to change the name of the job queue to be used.

   Remember, you may want to use a single-threaded job queue.

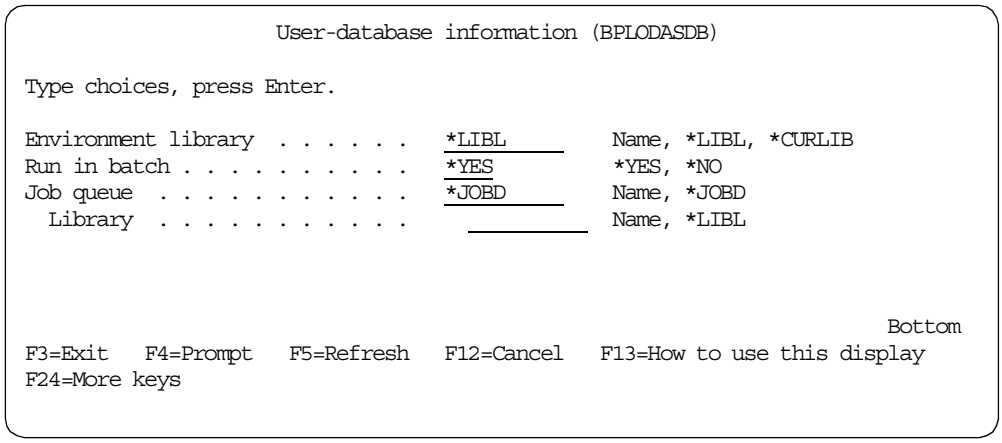


Figure 52. Load AS/400 Database Information (Part 2 of 2)

6. Press **Enter**. The load database information job is submitted to batch.

   You must wait for this step to complete before continuing with Section 5.4, “Step 3. Memory-Level Analysis” on page 67.

5.4 Step 3. Memory-Level Analysis

In this step, BYPASS2000 analyzes all the source statements for files, programs, copy members, and SQL table definitions. BYPASS2000 uses information from the repository that was loaded in the previous step.

It finds and stores the following information in the repository:

- The memory position of every field definition
- The relationship between two or more memory positions for each instruction of every program

For the memory-level analysis process, perform the following steps:

1. Select option **3** (Memory-level analysis) on the BYPASS2000 Main Menu shown in Figure 50.

   The BYPASS2000 Memory-Level Analysis menu is shown in Figure 53 on page 68.

```
BP4SANL              BYPASS2000 Memory-Level Analysis
                                                        System:   S1031GKM
Select one of the following:

     1. Work with database information
     2. Analyze database information
     3. Work with COPY sources
     4. Analyze COPY sources
     5. Work with SQL sources
     6. Analyze SQL sources
     7. Work with program sources
     8. Analyze program sources

    10. Work with file overrides
    11. Work with dynamic calls

    13. Work with logical REDEFINES

                                                                    More...
Selection or command
===> _____________________________________________________________________

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
(C) COPYRIGHT HAL S.p.A. 1994, 1998.
```

Figure 53. BYPASS2000 Memory-Level Analysis

2. If the old object library for database files contains source physical files, select option **1** (Work with database information) . This allows you to delete your source files (a sample of the display is shown in Figure 57 on page 70). You do not want source files converted. Deleting them takes them out of the analysis process. This saves time and space on the conversion environment repository.
3. If step 2 is not valid, select option **2** (Work with database information) to run the global analysis of database files in batch mode.
4. Select option **4** (Analyze Copy) if you have copy members.
5. Select option **6** (Analyze SQL table-definition sources) if you have tables defined through SQL DDL.
6. Select option **8** (Analyze programs).

Options 1, 3, 5, and 7 (Work with...) are usually more appropriate when tasks have been performed more than once.

5.4.1 Batch Options

There is batch support on the BYPASS2000 Memory-Level Analysis menu shown in Figure 53, for the following options:

- Option 2, Analyze database information
- Option 4, Analyze Copy
- Option 6, Analyze SQL table-definition sources
- Option 8, Analyze program

These batch options leads you to a corresponding Analyze... display.

See Figure 54 for an for an example of this display (option 2, Analyze database information).

```
                  Analyze Database Information (BPANLDDS)

Type choices, press Enter.

From DDS . . . . . .  *FIRST       Name, *FIRST
To DDS . . . . . .    *LAST        Name, *ONLY, *LAST
Application code . . . . . .  *ALL Name, *ALL, *BLANK
Environment library  . . . . . .  *LIBL  Name, *LIBL, *CURLIB
Run in batch . . . . . .  *YES     *YES, *NO


                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 54. Analyze DB Information (Part 1 of 2)

Notice an added parameter for the job queue as shown in Figure 55.

```
                  Analyze Database Information (BPANLDDS)

Type choices, press Enter.

From DDS . . . . . .  *FIRST       Name, *FIRST
To DDS . . . . . .    *LAST        Name, *ONLY, *LAST
Application code . . . . . .  *ALL Name, *ALL, *BLANK
Environment library  . . . . . .  *LIBL  Name, *LIBL, *CURLIB
Run in batch . . . . . .  *YES     *YES, *NO
Job queue  . . . . . .    *JOBD    Name, *JOBD
  Library  . . . . . .              Name, *LIBL


                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 55. Analyze DB Information (Part 2 of 2)

The analysis job is submitted to batch. You can see the status of the analysis using the interactive option for the same type of source.

5.4.2 Interactive Options

The following interactive options on the BYPASS2000 Memory-Level Analysis menu are shown in Figure 53 on page 68:

- Option 1, Work with database information
- Option 3, Work with Copy
- Option 5, Work with SQL table-definition sources
- Option 7, Work with program

Select option **1** (Work with data base information). The Select Parameters display appears as shown in Figure 56 on page 70.

```
                        Work with Database Information
                              Select Parameters


DDS  name . . . . : *ALL          *ALL, name, generic*


Application code  : *ALL          *ALL, name, generic*


Processing status          Analysis              Conversion

Not started   (0) :           Y                     Y
In progress (1-8) :           Y                     Y
Completed     (9) :           _                     _


F3=Exit   F12=Cancel
```

Figure 56. Work with Database Information - Select Parameters

You can select specific names or remove any of the prompted selections (as shown in Figure 56) if you want to reduce the list shown after you press Enter (a sample list is shown in Figure 57).

```
                        Work with Database Information
                                                Position to . . :
 Type choices, press Enter.
   2=Change    4=Delete    5=Display    8=Hold    9=Release
   I=Input source   O=Output source   V=View log  E=Reset cnv. flag K=Reset chk
   A=Analyze  D=Delete analysis  C=Convert   F=Override files    J=Dynamic call
Opt DDS  Name                              Ana     Cnv Gen Log Req Chk
 __ CUSTOMER                               9       0
 __ DATETEST                               9       0
 __ ORDERDTL                               9       0
 __ ORDERHDR                               9       0
 __ PROD                                   9       0
 __ SALESCUS                               9       0
 __ STOCK                                  9       0


                                                                      Bottom
 F3=Exit F4=Prompt F5=Refresh F6=Create  F10=Reset flag F11=Toggle F13=Repeat
 F12=Cancel F15=Include new member       F21=Command entry      F24=More keys
```

Figure 57. Work with Database Information - Type Choices

With this display, you gain control of the status of your application conversion. There is also a variety of choices available for each element included in the list shown.

You can get online information by pressing **F1**.

5.4.3 After Running Any Memory Analysis Option

Every time you run a memory-level analysis option, review the conversion log. Select option **22** on the BYPASS2000 Memory-Level Analysis menu shown in Figure 53 on page 68.

The Display Conversion Log display is shown. You can select the type of messages you want to look at or accept the default value for seeing all the messages and press **Enter** (see Figure 58).

```
                          Display Conversion Log

                                             From date (YYMMDD) :
 Type choices, press Enter.
   4=Delete    5=Display

 Opt Program    Source     Type Message description
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...
  _  PRPDAT155  OE         PGM  Field not propagated: type CALL, caller OE,  ...

                                                                       Bottom
 F3=Exit    F5=Refresh    F11=Display date,time,object      F12=Cancel
 F13=Repeat option        F17=Top        F18=Bottom         F23=Clear log
```

Figure 58. Display Conversion Log

The list of messages is shown in the conversion log. To display any of them for more information, **select** option 5.

If any messages direct you to check requested information, select option **21** on the BYPASS2000 Memory-Level Analysis menu shown in Figure 53 on page 68.

After you have taken action on the messages, select option **4** to delete them one at a time. Press **F23** to clear the entire log. If you clear the entire log, you are given the option to create a backup copy and store it in your environment for future reference.

Important

If you use F23 to clear the log, remember that you are clearing *all* the messages in the log, and not only those you are viewing.

Conversion logs occupy space on your disk. Therefore, delete them as soon as you no longer need them.

5.5 Step 4. Field Assignment (Seeding)

BYPASS2000 converts fields in source programs according to the results of the propagation analysis. Propagation analysis starts from the seeded database fields to find all the fields affected by them in the internal areas of the programs.

5.5.1 Database Field Assignment

In this field assignment step, all database fields containing dates must be identified with their correct format.

Field assignment divides all database fields into two groups:

- Year-sensitive fields
- Non-year sensitive fields

This is the phase of the conversion process that requires the most intervention from you.

You must enter the information interactively, unless you use an impact analysis tool that interfaces with BYPASS2000 (more information can be found in Appendix A, “Interfacing to BYPASS2000” on page 149). Perform these steps:

1. Select option **4** from the BYPASS2000 Main Menu as shown in Figure 50 on page 66. The BYPASS2000 Date-Field Assignment menu appears, as shown in Figure 59.

```
BPSRCASS              BYPASS2000 Date-Field Assignment
                                                          System:   S1031GKM
Select one of the following:

     1. Assign date field for I/O area related to file
     2. Assign date field
     3. Assign I/O area to related file
     4. Assign record type to related I/O area
     5. Work with migration utility programs
     6. Print list of assigned date fields
     7. Work with user-default date
     8. Work with date fields not to expand or propagate
     9. Assign dates from files to program and COPY areas
    10. Work with dates in files
    11. Import external field assignment (HSDATDFI)
    12. Import external field assignment (HSDATDFN)
    13. Load field assignment into HSDATDFN for export
    14. Create interface from IBM SEARCH2000 repository
                                                                      More...
Selection or command
===>
F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 59. BYPASS2000 Date - Field Assignment

2. Select option **1** (Assign date field for I/O area related to file) and press **Enter**. The display shown in Figure 60 on page 73 appears.

```
                Assign Date Field for I/O Area Related to File
                      Select Source Name or I/O Area

 Type choices, press Enter.

   Source:
     Name  . . . . .  *ALL                          *All, name, generic*

   I/O area:
     Name  . . . . .  *ALL                          *All, name, generic*
     Sequence. . . .   0                            Number

   File:
     Type  . . . . .  *ALL                          *All, name
     Name  . . . . .  *ALL                          *All, name


 F3=Exit  F12=Cancel
```

Figure 60. Assign Date Field for I/O Area Related to File

3. Because this is the first time that you are seeding fields, accept the defaults and press **Enter**.

   The Assign Date Field display in Figure 61 shows the list of database files BYPASS2000 previously stored in the repository.

```
                              Assign Date Field
                                I/O-Area List
                                                    Position to . . :
 Type choices, press Enter.
   1=Select    6=Print assigned date fields   A=Assign from dict. to database
   2=Set/Reset file as dictionary                    ---- File attributes -----
Opt Src. Name  I/O-Area Name                  Seq A  Type  Name               D
 _  CUSTOMER   *CUSTOMER                      000    PF    CUSTOMER
 _  DATETEST   *DATES                         000    PF    DATETEST
 _  ORDERDTL   *ORDERDTL                      000    PF    ORDERDTL
 1  ORDERHDR   *ORDERHDR                      000    PF    ORDERHDR
 _  PROD       *PROD                          000    PF    PROD
 _  SALESCUS   *SALESCUS                      000    PF    SALESCUS
 1  STOCK      *STOCK                         000    PF    STOCK









                                                                        Bottom
 F3=Exit     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom     F21=Print assigned date fields of listed sources
```

Figure 61. Assign Date Field (I/O Area List)

4. Select the database files that contain date fields using option **1** and press **Enter**. You see the field list for each selected file.
5. Use option **D** for the date fields shown in Figure 62 on page 74.

```
                        Assign Date Field - Field List
 DDS name  ORDERHDR    I/O area *ORDERHDR                    Seq   0
 Search field name:                                  Position to displ:

Type choices, press Enter.
 D=Assign date field    1-7=Quick-assign date field   W=Work with assigned date
 *=Annul assignment   R=Reusable field                   V=View all attributes
    Data  Field                                F       F Field Fld  Field Int. D
Opt Type  Name                                 R  Occ. G Displ Type Len.  Dec. A
 _        01  ORDERHDR                                 Y     1 CHAR    34  0 0

 _          02  ORHNBR                                      1 CHAR     5  0 0
              NUMBER
 _          02  CUSNBR                                      6 CHAR     5  0 0
              CUSTOMER CODE
 _          02  ORHDTA                                    11 PKD      4  6 0
              ORDER DATE !!!
 _          02  ORHDLY                                    15 PKD      4  6 0
              DELIVERY DATE !!!
                                                                       More...

 F3=Exit F5=Refresh  F11=View colhdg F14=Set HI rules  F15=View only highlight
 F17=Top F18=Bottom  F19=Field type  F20=Default type  F21=PF contents F12=Canc.
```

Figure 62. Assign Date Field - Field List

F21 causes a file query to run. This query shows default-edited fields. If you are not sure which fields are year-sensitive, it can help you find them or determine what format the dateis in.

For a sample of the query output, see Figure 63.

```
                              Display Report
                                          Report width . . . . . :      63
Position to line  . . . . .             Shift to column  . . . . . .
Line   ....+....1....+....2....+....3....+....4....+....5....+....6...
       ORHNBR  CUSNBR   ORHDTA    ORHDLY           ORHTOT   SRNBR
000001 00001   00003    52,996    52,997             5.65   A960121C
000002 00002   00003    52,996    52,997             5.65   A960121C
000003 00003   00003    52,996    52,997             5.65   A960121C
000004 00004   00003    52,996    52,997             7.00   A960121C
000005 00005   00003    52,996    52,997             7.00   A960121C
000006 00006   00003    52,996    52,997             7.00   A960121C
000007 00007   00003    52,996    52,997             7.00   A960121C
000008 00008   00003    52,996    52,997             7.00   A960121C
000009 00009   00003    52,996    52,997             7.00   A960121C
000010 00010   00003    52,996    52,997            10.50   A960121C
000011 00011   00003    53,096    53,097             5.65   A960121C
000012 00012   00003    53,096    53,097             5.65   A960121C
000013 00013   00003    53,096    53,097             5.65   A960121C
000014 00014   00003    53,096    53,097             5.65   A960121C
****** ********  End of report  ********

                                                                    Bottom
F3=Exit      F12=Cancel      F19=Left      F20=Right      F21=Split
```

Figure 63. Display Report

When you select a field as a date field, you must specify the date format in the Create New Date-Field entry display, as shown in Figure 64 on page 75.

BYPASS2000 provides you with a default date format for your field based on the length of the field. If this is not the correct date format, press **F4** to select from the list of possible date formats.

You can also see a complete list in Appendix A, “Interfacing to BYPASS2000” on page 149.

6. Leave the default values for the remaining entries.

```
                    Create New Date-Field Entry

Type choices, press Enter.

Source name . . . . . :  ORDERHDR
I/O-area name . . . . . :  *ORDERHDR
I/O-area sequence number. . . :  000
Field name. . . . . :  ORHDTA
Displacement. . . . . :  00011
Type. . . . . :  PKD
Length. . . . . :  00004
Integer / Decimal . . . . . :   6   0

Date type . . . . . :  003  DAY,MONTH,YEAR,/MONTH,DAY,YEAR
Position-field substring. . . :  00001
Length-field substring. . . . :  00004
Field type. . . . . :  P
Year length . . . . . :  2
Expansion type. . . . . :  0   (0=Default /1=Not expand /2=Expand + one
                               reusing /3=Expand + one /4=Expand  + two)
Propagation type. . . . . :  0   (0=Propag./1=Not propag./2=Upon confirm.)
F3=Exit    F4=List date type    F5=Refresh    F12=Cancel
```

Figure 64. Create New Date-Field Entry

7. Press **Enter**. You return to the field list shown in Figure 65 on page 76. The information on the date format has been added to the left of the date fields and they are marked as Date Assigned fields.

```
                        Assign Date Field - Field List
 DDS name  ORDERHDR    I/O area *ORDERHDR                    Seq   0
 Search field name:                                  Position to displ:

Type choices, press Enter.
 D=Assign date field    1-7=Quick-assign date field   W=Work with assigned date
 *=Annul assignment   R=Reusable field                   V=View all attributes
    Data  Field                                F       F Field Fld  Field Int. D
Opt Type  Name                                 R  Occ. G Displ Type Len.  Dec. A
 _        01  ORDERHDR                                 Y     1 CHAR    34  0 0

 _          02  ORHNBR                                      1 CHAR     5  0 0
              NUMBER
 _          02  CUSNBR                                      6 CHAR     5  0 0
              CUSTOMER CODE
 _  DMY     02  ORHDTA                                    11 PKD      4  6 0 Y
              ORDER DATE !!!
 _  DMY     02  ORHDLY                                    15 PKD      4  6 0 Y
              DELIVERY DATE !!!
                                                                       More...

 F3=Exit F5=Refresh  F11=View colhdg F14=Set HI rules  F15=View only highlight
 F17=Top F18=Bottom  F19=Field type  F20=Default type  F21=PF contents F12=Canc.
```

Figure 65. Assign Date Field - Field List

A new feature in BYPASS2000 V3R1M2 is the ability to convert date fields with a century bit to a full four-digit year date field. There are several scenarios that can occur.

If you have a one-digit century field by itself that you want to expand, you can simply assign type 20 (Century Flag) to that field. BYPASS2000 expands that to a full four-digit year field as shown in Figure 66.

```
                        Assign Date Field - Field List
 DDS name  DATA        I/O area *DATES                       Seq   0
 Search field name:                                  Position to displ:

Type choices, press Enter.
 D=Assign date field    1-7=Quick-assign date field   W=Work with assigned date
 *=Annul assignment   R=Reusable field                   V=View all attributes
    Data  Field                                F       F Field Fld  Field Int. D
Opt Type  Name                                 R  Occ. G Displ Type Len.  Dec. A
 _        01  DATES                                    Y     1 CHAR     5  0 0

 D          02  CENTRY                                      1 PKD      1  1 0
              CENTURY
 _          02  CENDTE                                      2 PKD      4  7 0
              DATE W/CENTURY


                                                                       Bottom

 F3=Exit F5=Refresh  F11=View colhdg F14=Set HI rules  F15=View only highlight
F17=Top F18=Bottom F19=Field type F20=Default type F21=PF contents F12=Canc.
```

Figure 66. Assign Date Field - Field List

8. To continue, press **Enter**. Choose Date type **20** for the century flag, as shown in Figure 67.

```
                    Create New Date-Field Entry

Type choices, press Enter.

Source name . . . . . :  DATA
I/O-area name . . . . . :  *DATES
I/O-area sequence number. . . :  000
Field name. . . . . :  CENTRY
Displacement. . . . . :  00001
Type. . . . . :  PKD
Length. . . . . :  00001
Integer / Decimal . . . . . :   1   0

Date type . . . . . :  020  CENTURY FLAG
Position-field substring. . . :  00001
Length-field substring. . . . :  00001
Field type. . . . . :  P
Year length . . . . . :  1
Expansion type. . . . . :  1   (0=Default /1=Not expand /2=Expand + one
                                reusing /3=Expand + one /4=Expand  + two)
Propagation type. . . . . :  0   (0=Propag./1=Not propag./2=Upon confirm.)
F3=Exit    F4=List date type    F5=Refresh    F12=Cancel
```

Figure 67. Create New Date-Field Entry - Before

9. Press **Enter** to confirm. Notice that when you have a century flag by itself, BYPASS2000 forces the Expansion type to be 1.

   If your century flag is part of a larger date field, you must seed the century and year-sensitive part of it, as shown in Figure 68 on page 78. Assuming that you have a seven-digit date field with the first bit being the century flag and the second and third bits being the year bits, you must seed the field as type **1** (Year). Be sure that you start at position 1 and have your Year length set to ***3***. By doing this, BYPASS2000 automatically knows that your field contains a century flag followed by year bits.

```
                    Create New Date-Field Entry

Type choices, press Enter.

Source name . . . . . :  DATA
I/O-area name . . . . . :  *DATES
I/O-area sequence number. . . :  000
Field name. . . . . :  CENDTE
Displacement. . . . . :  00002
Type. . . . . :  PKD
Length. . . . . :  00004
Integer / Decimal . . . . . :   7   0

Date type . . . . . :  001  YEAR
Position-field substring. . . :  00001
Length-field substring. . . . :  00004   Date position . . . :   00001
Field type. . . . . :  P
Year length . . . . . :  3
Expansion type. . . . . :  0   (0=Default /1=Not expand /2=Expand + one
                                reusing /3=Expand + one /4=Expand  + two)
Propagation type. . . . . :  0   (0=Propag./1=Not propag./2=Upon confirm.)
F3=Exit    F4=List date type    F5=Refresh    F12=Cancel
```

Figure 68. Create New Date-Field Entry - After

10. Press **Enter** to continue. Select option **W** beside the field that you have just seeded as shown in Figure 69.

```
                        Assign Date Field - Field List
 DDS name  DATA       I/O area *DATES                        Seq   0
 Search field name:                                 Position to displ:

Type choices, press Enter.
 D=Assign date field    1-7=Quick-assign date field   W=Work with assigned date
 *=Annul assignment   R=Reusable field                   V=View all attributes
    Data  Field                              F       F Field Fld  Field Int. D
Opt Type  Name                               R  Occ. G Displ Type Len.  Dec. A
 _        01  DATES                                  Y     1 CHAR     5  0 0

 _  CF      02  CENTRY                                      1 PKD      1  1 0 Y
               CENTURY
 W  Mult.   02  CENDTE                                      2 PKD      4  7 0 Y
               DATE W/CENTURY




                                                                        Bottom

 F3=Exit F5=Refresh  F11=View colhdg F14=Set HI rules  F15=View only highlight
 F17=Top F18=Bottom  F19=Field type  F20=Default type  F21=PF contents F12=Canc.
```

Figure 69. Assign Date Field - Field List

11. Press **Enter**. In Figure 70 on page 79, you can see that BYPASS2000 automatically sees that there is a century bit followed by a two-digit year bit in your seeding.

```
                        Work with Assigned Date Field
PGM name  DATA        I/O area *DATES                          Seq   0
Field name : CENDTE                              Displacement . . :     2
Field type : PKD         Length . . :     4      Integer / Decimal:  7  0

Type choices, press F3 to confirm your update.
2=Change       *=Annul assignment

Opt Dtf  Description---------  Date Pos  Position  Length  Type Year Sty  Pro
 _  020  CENTURY FLAG          00001     00001     00004    P    1    1    0
 _  001  YEAR                  00002     00001     00004    P    2    0    0


F3=Exit with Confirm        F5=Refresh       F6=Create new date field
F12=Cancel     F17=Top      F18=Bottom
```

Figure 70. Work with Assigned Date Field

5.5.2 If You Use a Dictionary (Field Reference File)

If you use a field reference or data dictionary file for your database files, it is possible to seed the date fields only in the dictionary. BYPASS2000 moves them to all the files that refer to the dictionary.

Do this seeding first, then proceed with the rest of the files (the ones that do not use a field reference file). Select option **2** on the Assign Date Field display to specify the field reference file as a dictionary. This display is shown in Figure 61 on page 73. If you have multiple copies of your data dictionary, make sure you use the most updated one. Also, make sure that the files you are using match the field reference file definition.

After assigning the date fields in this dictionary file, assign them from the dictionary to the database. Select option **A** in the Assign Date Field I/O Area List display for the dictionary and for the database files using it, as shown in Figure 61 on page 73.

If you specify Create New dictionary in the Customize Conversion Parameters (option 3 in the Setup Environment menu), BYPASS2000 creates the new default source (HBPDIZ, unless you have chosen another name in option 3 (Customize conversion parameters) in the Environment Setup Menu). After conversion of the dictionary, the new source is placed in the first library containing your converted DDS-PF source file.

5.5.3 Assigning Date Fields Not in Database

You can assign fields that are not defined in databases but in other I/O areas in programs or in COPY source members. Follow these steps:

1. If you must seed these types of fields, select option **2** from the BYPAS2000 Date-Field Assignment menu shown in Figure 59 on page 72. Option 2 shows the Assign Date Field - Select Source - I/O Area display as shown in Figure 71 on page 80.

```
                          Assign Date Field
                      Select Source - I/O Area

Type choices, press Enter.

  Source:
    Name  . . . . .  *ALL                      *All, name, generic*
    Type. . . . . .  PGM                       PGM, CPY

  I/O area:
    Name  . . . . .  *ALL                      *All, name, generic*
    Sequence. . . .  0                         Number



F3=Exit  F12=Cancel
```

Figure 71. Assign Date Field - Select Source - I/O Area

2. Specify if you want to select fields from COPY source members or from non-database related I/O areas in source programs. You can also change the default from *ALL (for all the members) to just a name or a generic name to reduce your selection. Press **Enter**.

   You see a list of the selected CPY or PGM members.Select option **1** for the source members where you seed date fields (see Figure 72).

```
                              Work with Programs
                                                  Position to . . :
 Type choices, press Enter.
   2=Change    4=Delete    5=Display    8=Hold    9=Release
   I=Input source      O=Output source  V=View log             A=Analyze
   D=Delete analysis   P=Propagate      Q=Delete propagation   C=Convert
Opt Pgm Name   Type                      Ana Prp Cnv Gen Log Req Chk
    CINS       RPG                       9   9   0
    CORD       RPG                       9   9   0
    DATRTN     RPG                       9   9   0
    FNLO       RPG                       9   9   0
    GETJOB     CLP                       9   9   0
    OE         RPG                       9   9   0
    ORDSTK     RPG                       9   9   0
    RIDT       RPG                       9   9   0
    RSTO       RPG                       9   9   0
    RSTR       RPG                       9   9   0
    STROE      CLP                       9   9   0


                                                                      Bottom
 F3=Exit F4=Prompt F5=Refresh F6=Create  F10=Reset flag F11=Toggle F13=Repeat
 F12=Cancel  F21=Command entry  F23=More option      F24=More keys
```

Figure 72. Assign Date Field - Select Source

3. After you select a source member, press **Enter**. The Assign Date Field - Program I/O-Area List display is shown.

   On this display, you can select the I/O areas (fields) you assign as *seeds*, as shown in Figure 73 on page 81.

```
                  Assign Date Field - Program I/O-Area List
 Pgm name  CINS
                                Position to . . :
 Type choices, press Enter.
  1=Select   2=Select with lock    *=Annul all assignments and locks
  6=Print assigned dates                        -Assigned-
Opt I/O-Area Name                   Seq       Date  Lock
  _  CUSTOT                         000
  _  CUSZIP                         000
  _  DATE1                          000
  _  DATE2                          000
  _  DRDATA                         000
  _  DRHDLY                         000
  _  DRHDTE                         000
  _  INSOK                          000
  1  JAREA                          000              Y
  _  JB400                          000
  _  NBR                            000
  _  NEWORD                         000
  _  ORDNO                          000
                                                                      More...
 F3=Exit     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom     F21=Print assigned date fields of listed sources
```

Figure 73. Assign Date Field - Program I/O-Area List

4. After you have selected an I/O area, pressed **Enter**. The Assign Date Field - Field List display for the selected I/O area appears, as shown in Figure 74.

```
                        Assign Date Field - Field List
 PGM name  CINS        I/O area JAREA                          Seq   0
 Search field name:                                  Position to displ:

Type choices, press Enter.
 L=Lock area   A=Assign date fields, locking selected area level   5=Displ.attr.
 *=Annul lock  O=Assign date fields, locking only assigned fields
    Lock  Field                                R       G Field Fld  Field Int. D
Opt Type  Name                                 F  Occ. F Displ Type Len.  Dec  A
  _       01  JAREA                                          1 CHAR    34  0 0
  _         02  BRTN                                        1 BIN      4  9 0
  _         02  BAVA                                        5 BIN      4  9 0
  _         02  JBNAME                                      9 CHAR    10  0 0
  _         02  UNAME                                      19 CHAR    10  0 0
  _         02  JBNBR                                      29 CHAR     6  0 0
                                                                      Bottom
 F3=Exit  F5=Refresh  F12=Cancel  F14=Set highlight rules F15=Only highlight
 F17=Top  F18=Bottom  F19=Work with field type  F20=Work with default type
```

Figure 74. Assign Date Field - Field List (for the I/O Area)

When you assign database fields, all non-seeded fields are automatically locked by BYPASS2000 as non-year sensitive fields and seeded fields are locked as year-sensitive fields. However, when you seed non-database I/O areas, you may want to ensure that all fields are properly locked, either as year-sensitive or as

non-year sensitive fields. In Figure 74 on page 81, you have different options for locking the non-database I/O area fields.

5.5.4 Display and Printer Fields

BYPASS2000 does not, by default, expand any display (MAP) or printer (PRT) fields. However, these fields can be assigned as year-sensitive fields.

Printer fields are never propagated but you can choose to expand them. If so, you must be aware of not overflowing the number of characters in the print line and make the appropriate title alignments. Display fields may be expanded and propagated.

5.5.4.1 Expanding MAP/PRT Fields

To expand the MAP/PRT fields, perform these step:

1. Select option **8** (Work with field not to be expanded (MAP, PRT)) on the BYPASS2000 Field Assignment menu shown in Figure 59 on page 72. The Select Programs display in Figure 82 on page 89 is shown.
2. Select the program where the MAP/PRT field is used.

   The Work with Not to be Expanded or Propagated Field display in Figure 75 is shown.

```
                Work with Not Expanded or Propagated Fields
Program name : CINS
                                          Position to date:
 Type choices, press Enter.
 4=Delete  *=Undelete  5=Display full data
              Inf I/O Area    I/O                              Field Field Usr
Opt COPY Name  Typ Name        Seq.    Field Name               Displ Len.  Act
 _             MAP CUSDES        0     CUSDES                       1    18
 _             MAP DRHDLY        0     DRHDLY                       1     6
 _             MAP DRHDTE        0     DRHDTE                       1     6


                                                                          Bottom
 F3=Exit          F6=Insert new field         F12=Cancel
 F17=Top          F18=Bottom                  F23=Delete all user fields
```

Figure 75. Work with Field Not-to-Be Expanded or Propagated

3. Select option **4** (Delete) for the display or printer fields you want to be expanded.

5.5.4.2 Century Window Routine

For dealing with non-expanded MAP/PRT fields in calculations, BYPASS2000 inserts routines that add the century according to Section 5.7.1.1, “Add Century Routine” on page 94.

5.5.5 Additional Options in Field Assignment

On the BYPASS2000 - Date Field Assignment menu shown in Figure 59 on page 72, there are additional options that are briefly described here:

- **Assign I/O area to related file (option 3):** This option is used to relate I/O areas to files to get the field description for non-externally described files or to use a different field description for a file.

- **Assign record type to related I/O area (option 4)***:* This option is used for multiple format files. You see every record format assigned in option 3 for files to which you have assigned more than one record format. Assign different record types to each of the record formats used in multiple format files.
- **Work with migration utility programs (option 5):** This option is used to create migration, test, and DIM routines for selected files.
- **Print list of assigned date field (option 6):** This option is used to print the list of assigned database date fields.
- **Work with user-default date (option 7):** This option is used to specify one or several field names as dates with their corresponding date format. These field names are assigned as year-sensitive fields in all your application programs.
- **Work with date fields not to expand or propagate (option 8):** This option is used to work with date fields found on display and printer files. You choose to make some of these date fields expandable (you can find more information in Chapter 6.6, “Hints and Tips” on page 132).
- **Assign dates from files to program and COPY areas (option 9):** This option is used to allow BYPASS2000 to automatically assign date fields in your program and copy source that references single or multiple format files.
- **Work with dates in file (option 10):** This option is used to display a list of files with dates assigned. There you can delete any of them (this just removes the date field assignments for that file) or you can select one file. When you select a file, you see a list of the assigned date fields with their characteristics (not field names). You can delete any of the fields (this will just remove the seeding).
- **Import external field assignment (HSDATDFI) (option 11):** This option is used if you have used an impact analysis tool that can interface to BYPASS2000, for example, SEARCH2000 (for information, see Appendix A, “Interfacing to BYPASS2000” on page 149). If you use this option, you do not need to use Assign Date Field for I/O Area Related to File" (option 1); see 5.5.1, “Database Field Assignment” on page 72.
- **Import external field assignment (HSDATDFN) (option 12):** This option is used if you have used SEARCH2000 to interface with BYPASS2000. However, unlike HSDATDFI which is generated by SEARCH2000, HSDATDFN does not exist until you run option 14 (Create interface from IBM SEARCH2000 repository).
- **Load field assignment into HSDATDFN for export (option 13):** This option is used to export the current field assignment to the file HSDATDFN to be used in another conversion environment.
- **Create interface from IBM SEARCH2000 repository (option 14):** This option is used if you have completed your task using SEARCH2000. to create the file HSDATDFN. When this option is selected, BYPASS2000 looks in your SEARCH2000 environment and generates the file HSDATDFN which BYPASS2000 can use.

5.5.6 Most Common Problems

This section provides a list of some common problems that can occur when using BYPASS2000:

- **If you must modify the definition of a field as a seed** — You must delete the assignment of that field using option "*" in the Assign Date Field - Field List display shown in Figure 62 on page 74.

  You must reassign it using option D in the same display. If the date field definition that you must modify belongs to a data dictionary field and it has already been assigned to the physical files, (see Section 5.5.2, “If You Use a Dictionary (Field Reference File)” on page 79), you must follow these steps:

  1. Delete the previous definition from all the files where it has been assigned from the dictionary.
  2. Delete and re-create the assignment in the data dictionary.
  3. Perform the assignment of the data dictionary seeded fields to the physical files again.

  You must take the previous steps because the automatic assignment of seeded data dictionary fields to physical files does not replace any definitions that already exist.

- **If you must seed other data dictionary fields** — You just perform the assignment to the physical files after having seeded the fields.

- **If a work field (a field declared in a CPY or in a program I/O area) uses two different date formats** — It must be assigned using option M. If you must delete the assignment, you must use option W in the Assign Date Field - Field List display shown in Figure 74 on page 81. Before using option M or W, you must first select option A (Assign date fields).

- **If a field is used in many programs as a date field** — It is possible to assign it using option 7, Assign general user default date (see Figure 60 on page 73). If this field is not used as a date in a specific program, you can lock it in the program as a non-year sensitive field. Program seeding overrides general user-default date seeding. For more information, see Section 5.5.3, “Assigning Date Fields Not in Database” on page 79.

Note

The correct identification and assignment of year-sensitive fields is the basis for the propagation process. This step requires the careful attention.

5.6 Step 5. Propagation-Level Analysis

This is the main step. In this step, BYPASS2000 propagates year-sensitive fields through the internal program areas. The propagation is also extended to called programs by the analysis of passed parameters.

The propagation process is iterative. The maximum number of iterations is established in the Customize Conversion environment step assigning a value to the Maximum level of propagation tree parameter shown in Figure 47 on page 64.

Propagation works with the following input stored in the repository tables:

- The results of the memory-level analysis step
- The assignment of the year-sensitive and non-year sensitive fields done in the field assignment step (see Section 5.5, “Step 4. Field Assignment (Seeding)” on page 72)

Based on this input, BYPASS2000 builds a complex tree of relationships along the branches of which the year field attribute is propagated from one area to another.

The iterative propagation process ends when the following conditions occur:

- All the branches of the propagation tree have been searched until finding a field defined as a seed or a field already *infected* (that is, a field to which the year attribute has been passed to). Therefore, the job ends normally.
- A higher level of iteration than the threshold defined during the conversion environment setup needs to be performed. Therefore, the job ends abnormally.

Because the relationships between memory areas include even the association of the parameters passed by the calling programs with the parameters received by the called programs, propagation analysis can work within a single program or between different programs (inter-program propagation analysis).

Therefore, all programs that are related must be included in the BYPASS2000 process. To start the propagation-level analysis, follow these steps:

1. Select option **5** on the BYPASS2000 Main Menu, as shown in Figure 50.

```
BP4SPRP                BYPASS2000 Propagation Analysis
                                                          System:   S1031GKM
Select one of the following:

     1. Assign date fields

     5. Work with programs
     6. Analyze date-field propagation

    22. Display conversion log
    23. Acknowledge requested information

    30. Work with propagation-analysis result
    31. Check propagation trace
    32. Delete propagation-analysis results

    70. Work with submitted jobs
    71. Work with all spooled files
                                                                      Bottom
Selection or command
===>
F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 76. BYPASS2000 Propagation-Level Analysis

2. Select option **6** (Analyze date-field propagation) on the BYPASS2000 Propagation Analysis menu as shown in Figure 76.

The Analyze Date-Field Propagation display in Figure 77 appears.

```
                Analyze Date-Field Propagation (BPPRPPGM)

Type choices, press Enter.

From program . . . . . .   *FIRST       Name, *FIRST
To program . . . . . .     *LAST        Name, *ONLY, *LAST
Application code . . . . . .  *ALL      Name, *ALL, *BLANK
Environment library  . . . . . .  *LIBL  Name, *LIBL, *CURLIB
Run in batch . . . . . .   *YES         *YES, *NO


                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 77. BYPASS2000 Analyze Date-Field Propagation (Part 1 of 2)

3. Accept the defaults to submit the job to batch and press **Enter**. A parameter for job queue is added to the previous display shown in Figure 78.

   Remember that you must use a single-threaded job queue.

```
                Analyze Date-Field Propagation (BPPRPPGM)

Type choices, press Enter.

From program . . . . . .   *FIRST       Name, *FIRST
To program . . . . . .     *LAST        Name, *ONLY, *LAST
Application code . . . . . .  *ALL      Name, *ALL, *BLANK
Environment library  . . . . . .  *LIBL  Name, *LIBL, *CURLIB
Run in batch . . . . . .   *YES         *YES, *NO
Job queue  . . . . . .     *JOBD        Name, *JOBD
  Library  . . . . . .                  Name, *LIBL


                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 78. BYPASS2000 Analyze Date-Field Propagation (Part 2 of 2)

4. To check the progress of the job, select option **5** (Work with program) from the BYPASS2000 Propagation-Level Analysis menu shown in Figure 76 on page 85.

```
                              Work with Programs
                              Select Parameters


Source name . . . : *ALL          *ALL, name, generic*

Source file . . . : *ALL          *ALL, name, generic*

Application code  : *ALL          *ALL, name, generic*


Processing status           Analysis  Propagation  Conversion

Not started   (0) :             Y           Y           Y
In progress (1-8) :             Y           Y           Y
Completed     (9) :             Y           Y           Y




F3=Exit   F12=Cancel
```

Figure 79. Work with Program - Select Parameters

The Work with Program - Select Parameters display appears, as shown in Figure 79. You can accept the default values or you can reduce the list of programs to receive by changing the default values.

5. Press **Enter**. The Work with Program list display in Figure 80 appears.

```
                                Work with Programs
                                                      Position to . . :
 Type choices, press Enter.
   2=Change    4=Delete    5=Display    8=Hold    9=Release
    F=Override files   L=Logical REDEFINES   J=Dynamic call E=Reset cnv.flag

Opt Pgm Name   Src File   Library        Ana Prp Cnv      History    Appl. Code
 _  CINS       QRPGSRC    BPSRC           9   9   0       PA
 _  CORD       QRPGSRC    BPSRC           9   9   0       PA
 _  DATRTN     QRPGSRC    BPSRC           9   9   0       PA
 _  FNLO       QRPGSRC    BPSRC           9   9   0       PA
 _  GETJOB     QCLPSRC    BPSRC           9   9   0       PA
 _  OE         QRPGSRC    BPSRC           9   9   0       PA
 _  ORDSTK     QRPGSRC    BPSRC           9   9   0       PA
 _  RIDT       QRPGSRC    BPSRC           9   9   0       PA
 _  RSTO       QRPGSRC    BPSRC           9   9   0       PA
 _  RSTR       QRPGSRC    BPSRC           9   9   0       PA
 _  STROE      QCLPSRC    BPSRC           9   9   0       PA


                                                                      Bottom
 F3=Exit F4=Prompt F5=Refresh F6=Create  F10=Reset flag F11=Toggle F13=Repeat
 F12=Cancel  F21=Command entry  F23=More option      F24=More keys
```

Figure 80. Work with Program - List

6. You can use different options on this display according to the issues presented within the BYPASS2000 conversion process (refer to Section 6.6, “Hints and Tips” on page 132 for more information).

At the end of the propagation step, you can check the output using the Work with propagation result option (for information, refer to Section 5.6.1, “Work with Propagation Results” on page 88) and the Check propagation trace option (refer to Section 5.6.2, “Check Propagation Trace” on page 90).

Remember to review the conversion log and check the required information.

5.6.1 Work with Propagation Results

A list of the analyzed programs is shown. You can see (for each of them) all the date fields found with *certainty* (flag = Y) and the areas defined as *generic* (flag = G). Generic fields are fields linked both with seeded date fields and with fields that have been locked as non-year sensitive. Resolve the generic fields before continuing. For each field, you can see which date field has propagated the year-sensitive condition to it and you can check the source member for the instruction that has generated the propagation.

To work with the propagation-level results, follow these steps:

1. Select option **30** on the BYPASS2000 Propagation-Level Analysis menu shown in Figure 76 on page 85, or, on the Work with Program list display shown in Figure 80 on page 87, press **F23** (More option) and then, use option **W** for any file.

   If you select option 30 on the BYPASS2000 Propagation-Level Analysis menu, the Work with Propagation Result - Select Parameters display shown in Figure 81 appears.

```
                        Work with Propagation Result
                             Select Parameters


Source name . . . . . . . . . . . . . : *ALL       *ALL, name, generic*


Only sources with

     Propagation incongruences found  : *   (Y/N/*)     (Y=At least one found)
                                                        (N=None found)
     Propagation status forced by user: *   (Y/N/*)     (*=All programs)









F3=Exit  F12=Cancel
```

Figure 81. Work with Propagation Result - Select Parameters

2. You can accept the defaults for the parameters or change them to reduce the list you see after you press **Enter**.

   The Select Programs display in Figure 82 appears.

3. Options available at this point are option 1 (Select) and option I (Input source). Option O (Output source) is not available because programs have not yet been converted. Therefore, no output source is created.

```
                              Select Programs

                                                  Position to . . :
Type choices, press Enter.
 1=Select   I=Input source  O=Output source
                                                ---------- Propagation ----------
 Opt Program Name Source File  Library      Incongruence  Changed      Status
  _  CINS         QRPGSRC      BPSRC                                      9
  _  CORD         QRPGSRC      BPSRC                                      9
  _  DATRTN       QRPGSRC      BPSRC                                      9
  _  FNLO         QRPGSRC      BPSRC                                      9
  _  GETJOB       QCLPSRC      BPSRC                                      9
  _  OE           QRPGSRC      BPSRC                                      9
  _  ORDSTK       QRPGSRC      BPSRC                                      9
  _  RSTO         QRPGSRC      BPSRC                                      9
  _  RSTR         QRPGSRC      BPSRC                                      9
  _  STROE        QCLPSRC      BPSRC                                      9

                                                                        Bottom
F3=Exit     F12=Cancel     F17=Top     F18=Bottom
```

Figure 82. Select Program

If you select any programs, you receive the propagation results for each of them on the Select Date Field display as shown in Figure 83.

```
                            Select Date Field
Program name: CINS
                                    Position to date field:
 Type choices, press Enter.
  1=Date origin           5=View statement        F=File origin
  D=Date field assigned   I=I/O-area field list   U=Update flag

Opt Src Name   Field Name------------ Type  Len. Date Type-------------- Proc.
  _ CINS       APP                   PKD      2     1 YEAR               OK
  _ CINS       DATE1                 PKD      6     3 DAY,MONTH,YEAR,/M OK
  _ CINS       DATE2                 PKD      6     3 DAY,MONTH,YEAR,/M OK
  _ CINS       PARM2                 ZND      6     3 DAY,MONTH,YEAR,/M OK
  _ CINS       DRHDLY                ZND      6     3 DAY,MONTH,YEAR,/M OK
  _ CINS       DRHDTE                ZND      6     3 DAY,MONTH,YEAR,/M OK









                                                                        Bottom
 F3=Exit  F4=Propag.trace  F5=Refresh   F8=Converted source   F17=Top F18=Bottom
 F12=Cancel  F7=Original source   F19=List assigned date     F20=One/Two Rows
```

Figure 83. Select Date Field

4. Use the options and function keys according to your requirements.

   Option 1 permits you to know which date field seeded by you allowed BYPASS2000 to discover this date field by propagation.

   You see this same display if you use F23 and then, option W from the Work with Programs list display shown in Figure 80 on page 87.

5.6.2 Check Propagation Trace

This function allows you to check the results of the propagation analysis to discover and solve ambiguous transitions between fields.

Check Propagation trace can only be run if the propagation analysis has ended normally (propagation analysis status flag = 9).

For each program, it is possible to view every *infected* area and the possible incongruences with the direction of the transition (which area *infected* it and which areas have been *infected* by it), that is, the propagation steps.

To start this function, select option **31** from the BYPASS2000 Propagation-Level Analysis menu shown in Figure 76 on page 85. This function shows the propagation steps followed by BYPASS2000. The Display Propagation-Trace entry display is shown in Figure 84.

```
                        Display Propagation Trace

Type choices, press Enter.                        Position to . . :
  1=Select

 Opt Program Name Source File  Library      Ana.   Prop.  Conv. Incongruence

  _  CINS         QRPGSRC      BPSRC          9      9      0       N
  _  CORD         QRPGSRC      BPSRC          9      9      0       N
  _  DATRIN       QRPGSRC      BPSRC          9      9      0       N
  _  FNLO         QRPGSRC      BPSRC          9      9      0       N
  _  GETJOB       QCLPSRC      BPSRC          9      9      0       N
  _  OE           QRPGSRC      BPSRC          9      9      0       N
  _  ORDSTK       QRPGSRC      BPSRC          9      9      0       N
  _  RSTO         QRPGSRC      BPSRC          9      9      0       N
  _  RSTR         QRPGSRC      BPSRC          9      9      0       N
  _  STROE        QCLPSRC      BPSRC          9      9      0       N



                                                                    Bottom
F3=Exit      F12=Cancel        F17=Top      F18=Bottom
```

Figure 84. Display Propagation-Trace Entry

When you select any program and press **Enter**, the propagation-trace entry for the selected program appears, as shown in Figure 85 on page 91.

```
                    Source name : CINS                              Page 01
    I/O Area                Field                   Date Field
    ____________________    ____________________    ____________________

--->ORHDLY                  ORHDLY                  ORHDLY
                                                    Displ:     1 Pos:     1  Y
--->ORHDTA                  ORHDTA                  ORHDTA
                                                    Displ:     1 Pos:     1  Y
--->UDATE                   UDATE                   UDATE
                                                    Displ:     1 Pos:     1  Y


                                                                       Bottom
F3=Exit      F4=Expand input/Display entry    F5=Initial list
F12=Cancel   F10=Statement list               F15=Display only incongruences
```

Figure 85. Propagation Trace for Selected Entry

If you select option T on the Work with Programs list display, as shown in Figure 80 on page 87, you get the same result.

5.6.3 Most Common Problems

This section provides a list of some problems that can occur during propagation analysis:

- When a field is found where the year is defined both on the right and on the left, you must verify the origin and probably define the field as a multi-format field. For more information, refer to Section 5.5.6, “Most Common Problems” on page 84.
- When areas in a program are found that receive both date fields and non-date fields. This situation occurs for the following reasons:
  - Insufficient or incorrect seeding

    A year sensitive field in the database, that has not been seeded, has been related by the propagation analysis to a field that has either been defined as a seed or that has been reached by propagation.

    In this case, define the field as a date, delete the propagation analysis, and run it again.
  - The program uses the same area to contain different types of information.

    A double seed definition is recommended. Use the expansion type option with a value of 1 (not to expand), and the propagation type option with a value of 2 (upon confirmation) (see the Create New Date-Field Entry display in Figure 64 on page 75). Any subsequent uncertainty on a field that should receive the *infection* through the propagation analysis is highlighted by the function `Check requested information` with messages, such as `Field related with a not propagated field...`.
- When dynamic program calls are found (name of the program being called is a variable), then you must define the real name of the called program for each occurrence. Select option **11** (Work with dynamic calls) in theBYPASS2000 Memory-Level Analysis menu shown in Figure 53 on page 68.
- When arithmetic operations using years are found in calculation, you must verify in the source program the correct logic of the highlighted operations and probably assign, as a seed, any variable not recognized as a date.

- If the propagation analysis does not terminate normally, that is, the maximum number of iterations is exceeded, you must perform the following tasks:
  1. Define as seeds the I/O areas suggested by the function Check requested information function when messages, such as `Propagation tree too deep` or `Propagation tree too big`, are found in the conversion log.
  2. Delete the results of the propagation analysis. Select option **42** on the BYPASS2000 Propagation-Level Analysis menu shown in Figure 77 on page 86.
  3. Perform the propagation analysis again.

5.7 Step 6. Conversion

This is the final step in the BYPASS2000 process of your application.

Based on the results of the propagation analysis, BYPASS2000 creates source members in your new libraries. These source members are modifications of your physical file definitions (DDS), CPYs. CLP, RPG, CBL, SQLRPG, and SQLCBL input source members (the ones in your old libraries).

All the seeded dates and all the year-sensitive fields found by the propagation analysis are expanded from a two-digit year value to a four-digit year value, except those belonging to displays or to printer files. You can optionally expand display, printer, or logical file fields (more information is available in Section 5.5.4.1, “Expanding MAP/PRT Fields” on page 82).

Regarding display and printer files, BYPASS2000 adds to the converted source programs instructions for managing the two-digit year fields on the basis of the *sliding window* concept (see Section 5.5.4.2, “Century Window Routine” on page 82).

BYPASS2000 clearly identifies all the modifications made to the source members by means of *markers* put in the first positions of the statement being added, changed, or replaced (see Appendix C, “BYPASS2000 Markers” on page 165).

BYPASS2000 also identifies the following items:

- Any requests to manually verify and solve any logical incongruences (as is the case of an area used both for date and non-date related data)
- Any instructions that must be verified to ensure they are correct
- Any instructions for which automatic handling is not possible

To begin conversion after the propagation analysis is complete, perform these steps:

1. Select option **6** (Application conversion) on the BYPASS2000 Main Menu shown in Figure 50 on page 66.

   The BYPASS2000 Application Conversion menu is shown in Figure 86 on page 93.

```
BP4SCNV              BYPASS2000 Application Conversion
                                                        System:   S1031GKM
Select one of the following:

     1. Work with database information
     2. Convert DDS sources
     3. Work with COPY sources
     4. Convert COPY sources
     5. Work with programs
     6. Convert program sources
     7. Convert DDS,COPY, and program sources
     8. Convert logical files
     9. Work with new dictionary names
    10. Work with files

    12. Create migration programs
    13. Create migration-dispatcher program

                                                                  More...
Selection or command
===> _____________________________________________________________________

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 86. BYPASS2000 Application Conversion Menu

2. Take the menu options in the order that they are listed on the menu (that is, Convert DDS first, then CPYs, then programs and so on).
3. Wait for each conversion to complete before you continue.

   Use the batch options (Convert...) the first time you run the conversion (options 2, 4 and 6 or only option 7). Afterwards, you can use the interactive options (1, 3, and 5). Interactive options allow you to select one or more members you want to convert from a list. Interactive options (Work with...) should be used when you repeat the steps shown in Figure 86.
4. Always select options **22** and **23** (Display conversion log and Check requested information), shown in Figure 87 on page 94, at the end of each conversion option.

There can be problems even if the conversion flags are all set to "9". If there are any, resolve them before continuing.

5.7.1 Post Conversion Utility Programs

After you finish converting all your sources and have compiling all your files, you can create different kinds of utility programs as shown in Figure 86. If you scroll down, you can see options to create two more utility programs as shown in Figure 87 on page 94.

```
BP4SCNV              BYPASS2000 Application Conversion
                                                        System:   S1031GKM
Select one of the following:

    15. Create test-migration programs
    16. Create test-migration-dispatcher program

    18. Create DIM programs
    19. Create DIM dispatcher program

    22. Display conversion log
    23. Acknowledge requested information

    70. Work with submitted jobs
    71. Work with all spooled files



                                                                    Bottom
Selection or command
===> _____________________________________________________________________

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 87. BYPASS2000 Database Conversion Menu

Select these options after the source members conversion is completed and you are sure you do not have to repeat any of the BYPASS2000 process steps.

The options on this menu are used to create different types of programs. These programs are useful tools for tasks subsequent to the conversion process.

They are briefly explained here:

- **Migration programs (MGR***)* — Convert database files to the expanded format from two-digit year fields to four-digit year fields. The two digits added for the century are established, as explained in Section 5.7.1.1, “Add Century Routine” on page 94.
- **Migration-dispatcher programs** — CL programs that run the migration programs.
- **Test migration programs (TSR)** — Perform a time shift of the database year-sensitive fields to allow you to simulate tests in the future. You can test your application as if you were at the end of the century or at the beginning of the new century.
- **Test migration-dispatcher programs** — CL programs that run the test migration programs.
- **DIM programs** — Data integrity verification modules that can be used as trigger programs that logically check the validity of date fields after addition/modification of records in a file.

Refer to *BYPASS2000 User's Guide*, SC09-2591, for more details.

5.7.1.1 Add Century Routine

Year fields (or year portion of a field) are expanded from two digits to four digits. The century two digits are added according to the value for the Minimum YEAR

value for 20th century conversion environment parameter. This value is set in the Customize Conversion Parameters display, as shown in Figure 47 on page 64.

For example, if the parameter value is set to 50, the century added to all two-digit years that contain any value between 51 and 99 is 19. For years between 00 and 50, the century used is 20.

5.8 Step 7. Compilation

After you have completed the conversion process, you must compile your programs from your new libraries into your test libraries. Ensure all file and program objects are created before proceeding to test your application.

5.9 Step 8. Test

Test your application to be sure that all date related functions are Year 2000 ready.

You may use the test migration programs that BYPASS2000 lets you create (more information is available in Section 5.7, “Step 6. Conversion” on page 92).

Important

You are responsible for ensuring that your applications are Year 2000 ready. Using BYPASS2000 does not ensure that all the Year 2000 related concerns in your applications have been 100% resolved. The only way to be sure that your applications and data are ready for working with Year 2000 and beyond is through a well-planned test of them.

Chapter 6. BYPASS2000 Tips and Techniques

This chapter describes the process details for BYPASS2000 and offers some helpful hints to overcome common problems.

6.1 Setup

By taking the time to set up BYPASS200, you can reduce the amount of time it takes to convert you applications.

6.1.1 Default Conversion Style

Before you begin BYPASS2000, you must create a BYPASS2000 environment. Creating a default environment allows BYPASS2000 to create several different libraries that you must populate. After BYPASS2000 generates these libraries, you must manually copy your sources, along with your file objects, to the BYPASS2000 created libraries. The following series of displays show you a step-by-step process to create a default environment.

```
BP4AMBI                 BYPASS2000 Environment Setup
                                                          System:   S1031GKM
Select one of the following:

     1. Create conversion environment

     3. Customize conversion parameters

     6. Work with object and source libraries
     7. Work with relationships between objects and sources
     8. Customize environment COPYs
     9. Work with field type
    10. Work with additional parameters
    11. Work with system fields

    13. Apply BYPASS2000 software key
    14. Create DDS from COPY
    15. Create COPY from File Object
                                                                    More...
Selection or command
===> _____________________________________________________________________

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
(C) COPYRIGHT HAL S.p.A. 1994, 1998.
```

Figure 88. BYPASS2000 Environment Setup

Complete the following series of steps:

1. Select option **1** (Create conversion environment) as shown in Figure 88.
2. Type your environment name and specify ***YES** for Create default environment value as shown in Figure 89 on page 98.

```
                Create Conversion Environment (BPCRTLIB)

Type choices, press Enter.

Conversion identifier  . . . . .   BP02          Name
Create default environment . . .   *YES          *YES, *NO


                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 89. Create Conversion Environment (BPCRTLIB)

A list of libraries that BYPASS2000 creates appears as shown in Figure 90.

```
                Create Conversion Environment (BPCRTLIB0)

 Type choices, press Enter.

 Create default environment . . . > *YES          *YES, *NO
 Original source library  . . . . > BP02OLD       Name, *NONE
 Converted source library . . . . > BP02Y2K       Name, *NONE
 Converted object library . . . . > BP02OBJ       Name, *NONE
 Conversion-database library  . . > BP02DB        Name
 User-database library  . . . . . > BP02DAT       Name, *NONE
 User-SQL-database library  . . . > *NONE         Name, *NONE
 Run in batch . . . . . .          *YES           *YES, *NO


                                                                    Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 90. Create Conversion Environment (BPCRTLIB0)

Table 10 provides a description of the libraries generated by BYPASS2000.

Table 10. BYPASS2000 Generated Libraries

Library Names	Description
xxxxOLD	This library should contain the sources of your files, Copybooks and programs.
xxxxY2K	This library will contain the newly converted sources of your files, Copybooks and programs.
xxxxOBJ	This library should contain the new converted objects of your files.
xxxxDB	This library used by BYPASS2000, contains information about your environment.
xxxxDAT	This library should contain the old objects of your files (*FILE objects, with attribute PF, LF, DSPF and PRTF).

3. Press **Enter** to continue. A detailed list of languages and source types is shown in Figure 91 on page 99.

```
                    Work with Object and Source Libraries

 Type choices, press Enter.                          Position to . . :
   3=Copy    4=Delete     5=Display    *=Reset old information

                                                 Old Src Lib  Old Src File  Old Object Lib
Opt Type        Src SEU-Type    Status           New Src Lib  New Src File  New Object Lib
 _  *FILE       DAT PF                           *NONE        *NONE         BP02DAT
                                                 *NONE        *NONE         BP02Y2K
 _  CBL         CPY CBL                          BP02OLD      QLBLSRC       *NONE
                                                 BP02Y2K      QLBLSRC       *NONE
 _  CBL         CPY CBL38                        BP02OLD      QLBLSRC       *NONE
                                                 BP02Y2K      QLBLSRC       *NONE
 _  CBL         CPY UNS36                        BP02OLD      QS36SRC       *NONE
                                                 BP02Y2K      QS36SRC       *NONE
 _  CBL         PGM CBL                          BP02OLD      QCBLSRC       *NONE
                                                 BP02Y2K      QCBLSRC       *NONE
 _  CBL         PGM CBL36                        BP02OLD      QS36SRC       *NONE
                                                 BP02Y2K      QS36SRC       *NONE
                                                                                More...

 F3=Exit     F5=Refresh     F6=Create       Enter=Validate      F9=Confirm
F12=Cancel F17=Top   F18=Bottom  F20=One/Two Rows
```

Figure 91. Work with Object and Source Libraries

By selecting a default environment, BYPASS2000 has generated libraries and files for you to populate.

6.2 Analysis

Analysis may be one of the easier stages when using BYPASS2000. However, it is very crucial that you do it properly. Analysis of programs and files allows BYPASS2000 to gather all the necessary information to complete the project correctly. See Section 5.4, “Step 3. Memory-Level Analysis” on page 67, for more information.

Important

When changing the source of any programs, you must run analysis again. Failure to do so yields improper results in the later stages. If the source of a file is changed, you must analyze that file again and then select option **2** in the main menu to reload database information.

6.3 Date Field Assignment

Date field assignment is also known as *seeding*. The following sections describe seeding and how it is used.

6.3.1 Multiple Format Files

If you have a file with different record formats and each record format is defined as a redefine of one of the previous ones (with the date field in different positions in each), it is necessary to define each record format. Each record format in a

multi-format file must have its own record type. The application database is the starting point for the propagation. A seeding error can occur due to the different positions of the date fields in each record format (because they are considered as only one record format). Furthermore, if you assign the same record type to different record formats, this can result in data that is not valid, error messages, or even the loss of data during migration.

Obviously, these problems occur only if at least one of the record formats contains year-sensitive fields. If not, you can ignore the problem.

Assume we have assigned a file to more than one logical area (multi-format file). If one of these layouts contains a date field, this information does not have to be propagated to the fields that are in the same position for different layouts. After all, during the generation of the migration routine for the files, it is necessary to know which layouts require century information and which do not.

To resolve this, a record type assignment function is provided to assign a different record type value to each record format for the file. This allows BYPASS2000 to determine which of the layouts require century information. This is shown in the example in Figure 92 on page 101.

```
0001.00      H             J
0002.00       ************************************************************
0003.00       * Verify file multiformat (S36)
0004.00       ************************************************************
0005.00      FMLT002  IF  F      50  4AI     1 DISK
0006.00       *
0007.00      IMLT002  NS       1 CA   2 CB   3 CC
0008.00      I                                      5  100DTYMD1
0009.00      I                                   P  11  140DTYMD2
0010.00       *
0011.00      I        NS       1 C1   2 C2   3 C3
0012.00      I                                      5   60DTYEAR
0013.00      I                                      7  100ORCD01
0014.00      I                                   P  11  140DTYMD2
0015.00       *
0016.00      I        NS       1 CM   2 CI   3 CO
0017.00      I                                      5  100DTDMY1
0018.00      I                                   P  15  180DTYMD2
0019.00       *
0020.00      I        NS
0021.00       *
0022.00      I            DS
0023.00      I                                      1   60DS006
0024.00      I                                      7  140DS008
0025.00       *
0026.00      C           'ABC '    CHAINMLT002              99
0027.00      C  N99                 DO
0028.00      C                      MOVE DTYMD1    DTYMD2
0029.00      C                      MOVE DTYMD2    NSD006  60
0030.00      C                      MOVE DTYMD2    DS006
0031.00      C                      ENDDO
0032.00       *
0033.00      C           '123 '    CHAINMLT002              98
0034.00      C  N98                 DO
0035.00      C                      MOVELDTYEAR    NSD00A  60
0036.00      C                      MOVE ORCD01    NSD00A
0037.00      C                      MOVE DTYEAR    DS008
0038.00      C                      ENDDO
0039.00       *
0040.00      C           'MIO '    CHAINMLT002              97
0041.00      C  N97                 DO
0042.00      C                      MOVE DTYMD2    CHAR7   7
0043.00      C                      MOVE DTDMY1    CHAR7A  7
0044.00      C                      ENDDO
0045.00       *
0046.00      C                      SETON                    LR
```

Figure 92. Assign a Different Record Type Value to Each Record Format

This example consist of the following elements:

- An RPG III program called DOC_MULTF1 (described above)
- A multi-format file named MLT002

The file MLT002 has the following record types:

- The first has ABC in the first three positions
- The second has 123 in the first three positions
- The third has MIO in the first three positions

The following six date fields are in the file:

- The following two fields are in the first format:
  - DTYMD1 data type 002, year length 2 (to be expanded)
  - DTYMD2 data type 002, year length 2 (to be expanded)
- The following two fields are in the second format:
  - DTYEAR data type 001 year length 4
  - DTYMD2 data type 002 year length 2 (to be expanded)
- The following two fields are in the third format:
  - DTDMY1 data type 003 year length 2 (to be expanded)
  - DTYMD2 data type 002 year length 2 (to be expanded)

Follow these steps to handle multiple-record format files in BYPASS2000:

1. To analyze the file, select option **3** (Memory-level analysis) in the BYPASS2000 main menu.
2. Select option **1**(Work with database information).
3. Select option **A** (Analyze) as shown in Figure 93.

```
                          Work with Database Information
                                                 Position to . . :
 Type choices, press Enter.
   2=Change    4=Delete    5=Display    8=Hold    9=Release
   I=Input source   O=Output source   V=View log  E=Reset cnv. flag K=Reset chk
   A=Analyze  D=Delete analysis  C=Convert   F=Override files    J=Dynamic call
Opt DDS  Name                              Ana     Cnv Gen Log Req Chk
 _   ITALFIL2                              9       9
 _   KNDSTP                                9       9
 _   KNDSTP1                               9       9
 _   KONG1                                 9       9
 _   MF.EF1                                9       9
 _   MF.REF                                9       9
 _   MF.SCX                                9       9
 _   MFREF                                 9       9
 _   MLT001                                9       9
 A   MLT002                                9       9
 _   MULTT1                                9       9
 _   MUL001                                9       9
 _   MUL001A                               9       9

                                                                        Bottom
 F3=Exit F4=Prompt F5=Refresh F6=Create  F10=Reset flag F11=Toggle F13=Repeat
 F12=Cancel F15=Include new member       F21=Command entry      F24=More keys
```

Figure 93. Work with Database Information

4. To find the multiple record format file, select option **4** (Date-field assignment) in the main menu.
5. Select option **1** (Assign date field for I/O area related to file) as shown in Figure 94 on page 103.

```
                              Assign Date Field
                                I/O-Area List
                                                    Position to . . :
 Type choices, press Enter.
   1=Select    6=Print assigned date fields   A=Assign from dict. to database
   2=Set/Reset file as dictionary                    ---- File attributes -----
Opt Src. Name  I/O-Area Name                  Seq A  Type  Name               D
 _  KONG1      *RECKO1                       000 Y  PF    KONG1
 _  MF.EF1     *MFREFR                       000 Y  PF    MF.EF1
 _  MF.REF     *MF.REF                       000    PF    MF.REF
 _  MF.SCX     *MF.SCX                       000    PF    MF.SCX
 1  MLT002     *MLTRE1                       000    PF    MLT002
 _  MUL001     *RMUL01                       000 Y  PF    MUL001
 _  MUL001C    MUL001A                       000 Y  PF    MUL001A
 _  MUL002     *RMUL02                       000 Y  PF    MUL002
 _  MUL002C    MUL002A                       000 Y  PF    MUL002A
 _  PERCSPF    *PERCSR                       000 Y  PF    PERCSPF
 _  PERSOPF    *PERSOR                       000 Y  PF    PERSOPF
 _  PF         *PFMODR2                      000 Y  PF    PF
 _  PFL001     *RECPF                        000 Y  PF    PFL001             +

 F3=Exit     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom     F21=Print assigned date fields of listed sources
```

Figure 94. Assigning Date Fields - Part 1

If you select option 1, you see only the key field and a unique field which contains all the data.

```
                        Assign Date Field - Field list
 DDS name  MLT002      I/O area *MLTRE1                        Seq   0
 Search field name:                                  Position to displ:

Type choices, press Enter.
 D=Assign date field    1-7=Quick-assign date field   W=Work with assigned date
 *=Annul assignment   R=Reusable field                    V=View all attributes
    Data  Field                                F       F Field Fld  Field Int. D
Opt Type  Name                                 R  Occ. G Displ Type Len.  Dec. A
 _        01  MLTRE1                                   Y     1 CHAR    50  0 0

 _          02  NKEY                                        1 CHAR     4  0 0

 _          02  FILL01                                      5 CHAR    46  0 0









 F3=Exit F5=Refresh  F11=View colhdg F14=Set HI rules  F15=View only highlight
 F17=Top F18=Bottom  F19=Field type  F20=Default type  F21=PF contents F12=Canc.
```

Figure 95. Assigning Date Fields - Part 2

6. Create a COPY that contains all the formats of the file, as well as every field. BYPASS2000 uses the COPY only in the migration phase.

   Using the example program shown in Figure 92 on page 101, we can create a COPY source COPY_MULTF1 using the following specification:

```
0001.00      IMLT002  NS       1 CA   2 CB   3 CC
0002.00      I                                      1   4 KEY
0003.00      I                                      5  100DTYMD1
0004.00      I                                   P  11  140DTYMD2
0005.00      I                                      15  50 FLNEW1
0006.00       *
0007.00      I        NS       1 C1   2 C2   3 C3
0008.00      I                                      1   4 KEY
0009.00      I                                      5   60DTYEAR
0010.00      I                                      7  100ORCD01
0011.00      I                                   P  11  140DTYMD2
0012.00      I                                      15  50 FLNEW1
0013.00       *
0014.00      I        NS       1 CM   2 CI   3 CO
0015.00      I                                      1   4 KEY
0016.00      I                                      5  100DTDMY1
0017.00      I                                   P  15  180DTYMD2
0018.00      I                                      19  50 FLNEW2
```

7. To analyze it, select option **3** (Memory-level analysis) in the main menu.
8. Select option **3** (Work with COPY sources). The following display appears.

```
                              Work with COPYs
                                                 Position to . . :
 Type choices, press Enter.
   2=Change    4=Delete    5=Display    8=Hold    9=Release
   I=Input source   O=Output source   V=View log  E=Reset cnv. flag K=Reset ch
   A=Analyze  D=Delete analysis  C=Convert   F=Override files    J=Dynamic cal
Opt COPY Name  Type                      Ana     Cnv Gen Log Req Chk
  _  ARTLSC     RPG                       9       0
  _  ARTLSR     RPG                       9       0
  _  ARTLXCG    RPG                       9       0
  _  ARTLXC1    RPG                       9       0
  _  ARTLYCG    RPG                       9       0
  _  ARTLSR     RPG                       9       0
  _  ARTLXCG    RPG                       9       0
  _  ARTLYC1    RPG                       9       0
  _  CAMPICPY   RPG                       9       0
  _  CDISIN1    RPG                       9       0
  A  CPY_MULTF1 RPG                       9       0
  _  CPYFDUE    RPG                       9       0
  _  CPYMUL01   RPG                       9       0
  _  CPYMUL02   RPG                       9       0
                                                                      More..
 F3=Exit F4=Prompt F5=Refresh F6=Create  F10=Reset flag F11=Toggle F13=Repeat
 F12=Cancel F15=Include new member       F21=Command entry      F24=More keys
```

Figure 96. Working with Copies

Important

After you have created your COPY source, you should see it in the list in Figure 96. If you do not see it in the list, press F15. By pressing F15, you will include any new members that were not present before.

9. To link the COPY to the physical file, select option **3** (Assign I/O area to related file) in the Date-Field Assignment menu.

10. Select option **2** (Change), as shown in Figure 97.

```
                    Assign I/O Area to Related File

 File to assign . :                              Type/Name, F4 (List)

                                                 Position to . . :
 Type choices, press Enter.
   1=Select    2=Change    5=Display

Opt Src Name    I/O-Area Name                    File Type   Filename
 _   CEMASAPF    *REGCEMASA                       PF          CEMASAPF
 _   CPY_MULTF   MLT001                          PF          MLT001
 2   CPY_MULTF1  MLT002
 _   CPYFDUE     RECFUNO                         PF          FDUE
 _   CPYMUL01    MULTT1                          PF          MULTT1
 _   CPYRMF      MSFILEMF                        PF          FILEMF
 _   CPYRMFN     MSFILEMF                        PF          FILEMFN
 _   CURRENCY    *TENDER                         PF          CURRENCY
 _   CUST        *CUSTOMER                       PF          CUST
 _   CUST_1      *CUSOTMER                       PF          CUST_1
 _   C77380C     INTST1                          PF          INTST1
                                                                      More...
 F3=Exit     F4=File list     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom
```

Figure 97. Assign I/O Area to Related File (Part 1 of 4)

11. Press **Enter** to continue. As shown in Figure 98, enter the name of the file you want to associate with the member.

```
                          Change entry

 Type choices, press Enter.


 Source name . . . . . :  CPY_MULTF1

 I/O-area name . . . . . :  MLT002

 File type . . . . . :  PF              Type, F4 (List)

 Filename  . . . . . :  MLT002          Name, F4 (List)


 F3=Exit     F4=File list     F5=Refresh     F12=Cancel
```

Figure 98. Assign I/O Area to Related File (Part 2 of 4)

12. You must disassociate the file from itself. Select option **2** (Change) for the MLT002 file as shown in Figure 99 on page 106.

```
                    Assign I/O Area to Related File

 File to assign . :                              Type/Name, F4 (List)

                                                 Position to . . :
 Type choices, press Enter.
   1=Select    2=Change    5=Display

Opt Src Name    I/O-Area Name                  File Type   Filename
 _  MINEIRO01   COPY-AUT-AT
 _  MLT001      *MLTREC
 2  MLT002      *MLTRE1                        PF          MLT002
 _  MULTT1      *MULTT1
 _  MUL001      *RMUL01                        PF          MUL001
 _  MUL001A     *RMUL01A
 _  MUL001C     MUL001A                        PF          MUL001A
 _  MUL002      *RMUL02                        PF          MUL002
 _  MUL002A     *RMUL02A
 _  MUL002C     MUL002A                        PF          MUL002A
 _  PERCSPF     *PERCSR                        PF          PERCSPF
                                                                    More...
 F3=Exit     F4=File list     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom
```

Figure 99. Assign I/O Area to Related File (Part 3 of 4)

13. You must delete the entry for the file type and filename, as shown in Figure 100.

```
                              Change entry

 Type choices, press Enter.


 Source name . . . . . :  MLT002

 I/O-area name . . . . . :  *MLTRE1

 File type . . . . . :  _____________       Type, F4 (List)

 Filename  . . . . . :  _____________       Name, F4 (List)


 F3=Exit     F4=File list     F5=Refresh     F12=Cancel
```

Figure 100. Assign I/O Area to Related File (Part 4 of 4)

Figure 101 on page 107 shows the result of this action.

```
                    Assign I/O Area to Related File

 File to assign . :                              Type/Name, F4 (List)

                                                 Position to . . :
 Type choices, press Enter.
   1=Select    2=Change    5=Display

Opt Src Name    I/Area Name                  File Type   Filename
 _ MINEIRO01   COPY-AUT-AT
 _ MLT001      *MLTREC
 _ MLT002      *MLTRE1
 _ MULTT1      *MULTT1
 _ MUL001      *RMUL01                       PF          MUL001
 _ MUL001A     *RMUL01A
 _ MUL001C     MUL001A                       PF          MUL001A
 _ MUL002      *RMUL02                       PF          MUL002
 _ MUL002A     *RMUL02A
 _ MUL002C     MUL002A                       PF          MUL002A
 _ PERCSPF     *PERCSR                       PF          PERCSPF          +

 F3=Exit     F4=File list     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom
```

Figure 101. File Is Now Associated with Copy

14. Assign the record type to each format. Select option **4** (Assign record type to related I/O area) in the Date-Field Assignment menu.

    A line for each multiple format file is shown. In addition, the list of the different formats in each file is shown. In this example, there are three record formats shown for file MLT002.

```
                  Assign Record Type to Related I/O Area
                                File list
                                                 Position to . . :
 Type choices, press Enter.
   1=Select

                                  Record Fmt  Record Type  Record Type
Opt File Type   Filename          Number      Position     Length
 _   PF         FILEMF               004         00001        03
 _   PF         FILEMFN              003         00001        03
 _   PF         INTSTG               002         00082        02
 _   PF         INTSTP               002         00081        02
 _   PF         INTST1               002         00081        02
 _   PF         MLT001               003         00001        03
 _   PF         MLT002               003         00000        00
 _   PF         MULTT1               003         00081        02




 F3=Exit     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom
```

Figure 102. Assign Record Type to Related I/O Area - File List

15. Select option **1** to assign date fields for a record format, as shown in Figure 103.

```
                  Assign Record Type to Related I/O Area
                                File list
                                                    Position to . . :
 Type choices, press Enter.
   1=Select
                                      Record Fmt  Record Type  Record Type
 Opt File Type   Filename             Number      Position     Length
  _  PF          FILEMF                  004         00001        03
  _  PF          FILEMFN                 003         00001        03
  _  PF          INTSTG                  002         00082        02
  _  PF          INTSTP                  002         00081        02
  _  PF          INTST1                  002         00081        02
  _  PF          MLT001                  003         00001        03
  1  PF          MLT002                  003         00000        00
  _  PF          MULTT1                  003         00081        02
 F3=Exit      F5=Refresh     F12=Cancel
 F17=Top      F18=Bottom
```

Figure 103. Assign Record Type to Related I/O Area - Select MLT002

The Assign Record Type to Related I/O Area display appears.

```
                  Assign Record Type to Related I/O Area
                              I/O-area list
 File type/name: PF         / FILEIN             Position to . . :
 Record fmt nbr: 002                  Record type position/length: 00001 / 01

    Press F3 to confirm your update.
    Copy Name   I/O-Area Name                 Seq  Record Type
    CPY_MULTF1  MLT002                        001  _____________________
    CPY_MULTF1  MLT002                        002  _____________________
    CPY_MULTF1  MLT002                        003  _____________________

                                                                    Bottom
 F3=Exit with Confirm      F17=Top          F12=Cancel
 F5=Refresh                F18=Bottom
```

Figure 104. Assign Record Type to Related I/O Area

16. In our example, each record type begins at position 1 and has a length of 3. Enter these values.

    The result is shown in Figure 105 on page 109.

```
                  Assign Record Type to Related I/O Area
                                I/O-area list
File type/name: PF         / FILEIN             Position to . . :
Record fmt nbr: 002                  Record type position/length: 00001 / 01

   Press F3 to confirm your update.
   Copy Name   I/O-Area Name                  Seq  Record Type
   CPY_MULTF1  MLT002                         001  ABC
   CPY_MULTF1  MLT002                         002  MIO
   CPY_MULTF1  MLT002                         003  123


                                                                      Bottom
F3=Exit with Confirm      F17=Top          F12=Cancel
F5=Refresh                F18=Bottom
```

Figure 105. Assign Record Type to Related I/O Area - Insert Record Types

17. Press **F3** to confirm. This returns you to the previous display as shown in Figure 106.

```
                  Assign Record Type to Related I/O Area
                                File list
                                                    Position to . . :
 Type choices, press Enter.
   1=Select
                                      Record Fmt  Record Type  Record Type
Opt File Type   Filename              Number      Position     Length
 _  PF          FILEMF                   004         00001         03
 _  PF          FILEMFN                  003         00001         03
 _  PF          INTSTG                   002         00082         02
 _  PF          INTSTP                   002         00081         02
 _  PF          INTST1                   002         00081         02
 _  PF          MLT001                   003         00001         03
 _  PF          MLT002                   003         00001         03
 _  PF          MULTT1                   003         00081         02


F3=Exit     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom
```

Figure 106. Assign Record Type to Related I/O Area - Updated Record Information

18. Select option **1** (Assign date field for I/O area related to file) in the Date-Field Assignment menu to seed the date fields of the COPY related to file. See Figure 107 on page 110.

```
                              Assign Date Field
                                I/O-Area List
                                                    Position to . . :
 Type choices, press Enter.
   1=Select    6=Print assigned date fields   A=Assign from dict. to database
   2=Set/Reset file as dictionary                    ---- File attributes -----
Opt Src. Name  I/O-Area Name                  Seq A  Type  Name              D
 1  CPY_MULTF1 MLT002                         001    PF    MLT002
 _  CPY_MULTF1 MLT002                         002    PF    MLT002
 _  CPY_MULTF1 MLT002                         003    PF    MLT002
 _  CPYFDUE    RECFUNO                        000    PF    FDUE
 _  CPYMUL01   MULTT1                         001 Y  PF    MULTT1
 _  CPYMUL01   MULTT1                         002 Y  PF    MULTT1
 _  CPYMUL01   MULTT1                         003    PF    MULTT1
 _  CPYRMF     MSFILEMF                       001 Y  PF    FILEMF
 _  CPYRMF     MSFILEMF                       002 Y  PF    FILEMF
 _  CPYRMF     MSFILEMF                       003 Y  PF    FILEMF
 _  CPYRMF     MSFILEMF                       004    PF    FILEMF
 _  CPYRMFN    MSFILEMF                       001 Y  PF    FILEMFN
 _  CPYRMFN    MSFILEMF                       002    PF    FILEMFN           +

 F3=Exit     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom     F21=Print assigned date fields of listed sources
```

Figure 107. Assign Date Field - Select with Option 1

The seeding of the copy is necessary for the migration of the file.

19. You must seed the date fields in the COPY because seedings performed in the file fields are not recognized from the association between file and programs. Select option **2** (Assign date field) and seed each COPY and program that uses the file.

    Alternatively, you can choose option 9 (Assign dates from files to program and COPY areas) to automatically seed all programs and copies that uses the file. See Figure 108 on page 111.

```
BPSRCASS              BYPASS2000 Date-Field Assignment
                                                          System:   S1031GK
Select one of the following:

     1. Assign date field for I/O area related to file
     2. Assign date field
     3. Assign I/O area to related file
     4. Assign record type to related I/O area
     5. Work with migration utility programs
     6. Print list of assigned date fields
     7. Work with user-default date
     8. Work with date fields not to expand or propagate
     9. Assign dates from files to program and COPY areas
    10. Work with dates in files
    11. Import external field assignment (HSDATDFI)
    12. Import external field assignment (HSDATDFN)
    13. Load field assignment into HSDATDFN for export
    14. Create interface from IBM SEARCH2000 repository
                                                                      More..
Selection or command
===>  9

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 108. BYPASS2000 Date-Field Assignment

20.Press **Enter**. The display shown in Figure 109 appears.

```
                                Work with Files
                                                Position to . . :
Type choices, press Enter.
  1=Select  C=Create DDS


Opt File        Lib         Typ
 _  MF.EF1      APAREXDAT   PF
 _  MF.REF      APAREXDAT   PF
 _  MF.SCX      APAREXDAT   PF
 _  MFREF       APAREXDAT   PF
 _  MLT001      APAREXDAT   PF
 1  MLT002      APAREXDAT   PF
 _  MULTT1      APAREXDAT   PF
 _  MUL001      APAREXDAT   PF
 _  MUL001A     APAREXDAT   PF
 _  MUL002      APAREXDAT   PF
 _  MUL002A     APAREXDAT   PF
 _  MUL1LF      APAREXDAT   LF
 _  MUL2LF      APAREXDAT   LF                                            +

F3=Exit F5=Refresh F12=Cancel F17=Top F18=Bottom F21=Command entry
```

Figure 109. Work with Files - Select MLT002 in Library APAREXDAT

21.Select option **1**. The display shown in Figure 110 on page 112 appears.

```
                              Work with Files
                                              Position to . . :
Type choices, press Enter.
  1= ..........................................................................
     :                    Work with Physical File MLT002                      :
     : Type choices, press Enter.                 Position to.                :
Opt  : 1=Select A=Assign date field I=Input source V=View fields              :
 _   :                                                                        :
 _   : Opt Program    LF         I/O-Area Name                  Seq  Size     :
 _   :     Name       Name                                                    :
 _   :  _  DOC_MULTF1            MLT002                         003  00050    :
 1   :  _  DOC_MULTF1            MLT002                         002  00050    :
 _   :  _  DOC_MULTF1            MLT002                         001  00050    :
 _   :                                                                        :
 _   :                                                                        :
 _   :                                                                        :
 _   :                                                                        :
 _   : F3=Exit F4=Prompt F5=Refresh F12=Cancel F17=Top F18=Bottom             :
 _   :                                                                        :
     :........................................................................:
                                                                       More...
F3=Exit F5=Refresh F12=Cancel F17=Top F18=Bottom F21=Command entry
```

Figure 110. Work with Physical File MLT002

22. Type **1** next the first I/O Area Name (Seq. 003) and press **Enter**.

```
                              Work with Files
                                              Position to . . :
Type choices, press Enter.
  1= ..........................................................................
     : ..................................................................     :
     : :               Work with Program DOC_MULTF1                   :     :
Opt  : : Type choices, press Enter.       Position to.                 :     :
 _   : : 1=Select                                                       :     :
 _   : :                                                                : e   :
 _   : : Opt Srcname    Type I/O-Area Name                  Seq         :     :
 _   : :  _  CPY_MULTF1 CPY  MLT002                         001         : 50  :
 1   : :  _  CPY_MULTF1 CPY  MLT002                         002         : 50  :
 _   : :  _  CPY_MULTF1 CPY  MLT002                         003         : 50  :
 _   : :                                                                :     :
 _   : :                                                                :     :
 _   : :                                                                :     :
 _   : :                                                     Bottom : ttom :
 _   : : F3=Exit F5=Refresh F12=Cancel F17=Top F18=Bottom               :     :
 _   : :                                                                :     :
 _   : :                                                                : ..... :
     :.....................................................................: ore...
F3=Exit F5=Refresh F12=Cancel F17=Top F18=Bottom F21=Command entry
```

Figure 111. Work with Program DOC_MULTF1

23. Associate the corresponding I/O Area in the associated copy. In this example, select the third (Seq=003) as shown in Figure 112 on page 113.

    The seeding done to the I/O Area in CPY_MULTF1 is assigned, as well as to the selected I/O Area in program DOC_MULTF1.

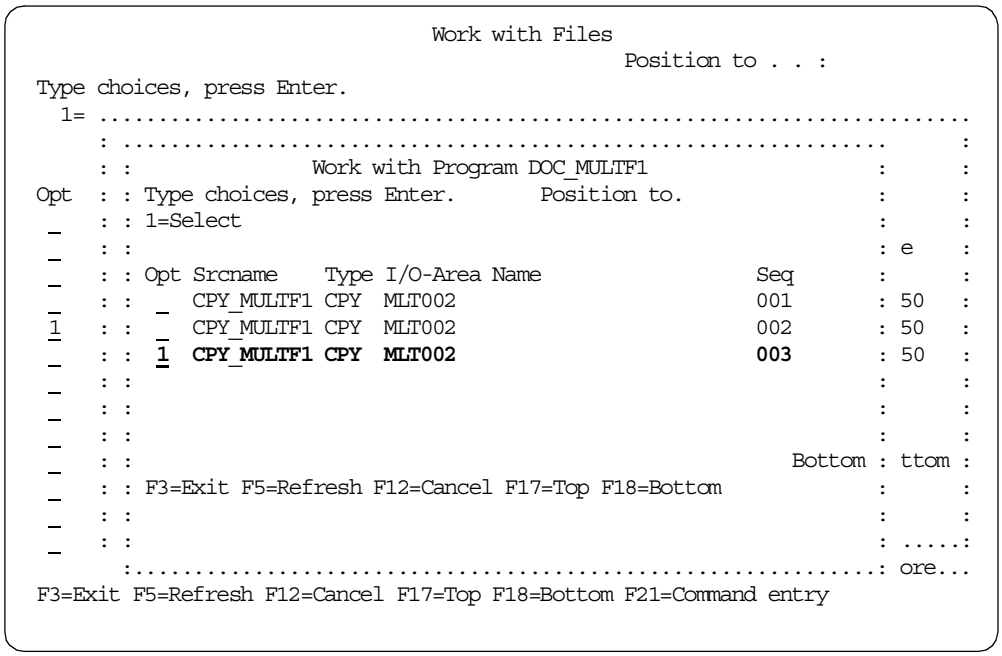


Figure 112. Work with Program DOC_MULTF1 - Option 1=Select

24. Press **Enter.** Then press **F12** to continue. A message appears, as shown in Figure 113.

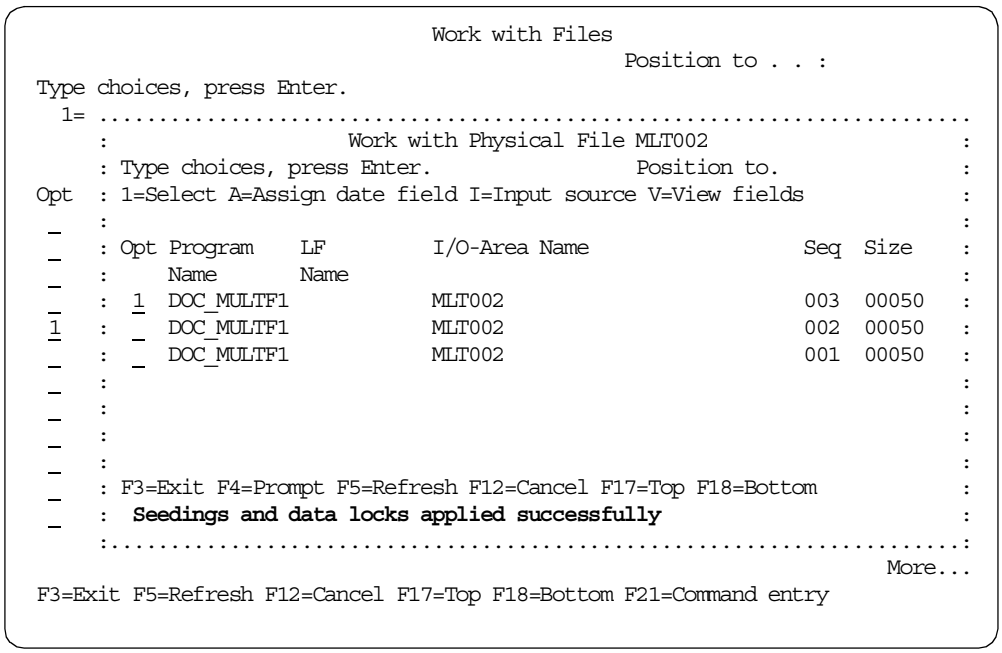


Figure 113. Work with Physical File MLT002 - Repeat for Other I/O-Area Name

25. Repeat the same instructions to assign the second I/O Area (Seq. 002) and first I/O Area (Seq 001).

26. Verify that BYPASS2000 has seeded the programs properly. Select option **2** (Assign date field) in the Date-Field Assignment menu as shown in Figure 108 on page 111.

```
                  Assign Date Field - Program I/O-Area List
 Pgm name  DOC_MULTF1
                                Position to . . :
 Type choices, press Enter.
  1=Select   2=Select with lock    *=Annul all assignments and locks
  6=Print assigned dates                        -Assigned-
Opt I/O-Area Name                   Seq       Date  Lock
 _  CHAR7                           000
 _  CHAR7A                          000
 _  DS-DOC_MULTF1-00001             000
 _  DTDMY1                          000
 _  DTYEAR                          000
 _  DTYMD1                          000
 _  DTYMD2                          000
 _  MLT002                          001         Y    Y
 _  MLT002                          002         Y    Y
 _  MLT002                          003         Y    Y
 _  NSD00A                          000
 _  NSD006                          000
 _  ORCD01                          000

 F3=Exit     F5=Refresh     F12=Cancel
 F17=Top     F18=Bottom     F21=Print assigned date fields of listed sources
```

Figure 114. Assign Date Field - Program I/O-Area List

27.From this point, propagate and convert as usual.

If the record type for each format is not sequential or correctly defined, the migration routine cannot be built. However, you can define your own record type, build the migration routine, and then modify it, if required. You must also be aware that the propagation will fail if there are multi-format files with unassigned record types.

6.3.2 General Seeding

After the memory-level analysis is complete, BYPASS2000 requires the following information. This can be provided manually or from an impact analysis tool. Refer to Chapter 3, “SEARCH2000 Impact Analysis Tool” on page 29, for more information.

- All the year-sensitive fields identified from the database
- The format of these fields (for example, YYMMDD, DDMMYY, and so on)
- Whether the fields require expansion

After all this information is provided, there are two distinct groups for the variables:

- Year-sensitive fields and their formats
- Non-year sensitive fields - these are fields that have not been seeded but they belong to an I/O area to which year-sensitive fields are related.

Each field in the database must be identified as either year or non-year sensitive. If a field is not seeded, BYPASS2000 assumes that this field is not a date. See the following example:

```
A      R DICT              TEXT('Data Definition.')
A        DTEY     2  0     COLHDG('Year LEN 2 Packed.')
A        DTEYMD   6  0     COLHDG('YMD LEN 6 Packed.')
A        DTEDMY   6s 0     COLHDG('DMY LEN 6 Zoned.')
         ....
A        CDCLI    6        COLHDG('Client')
         ....
```

By defining DTEY, DTEYMD, and DTEDMY as year-sensitive fields in this file, this implies that CDCLI is not year-sensitive. Any fields that are used in the programs, but not previously identified in the database, are discovered during propagation.

Seeding is the most important stage for the following reasons:

- The seeded variables are propagated to find any relationships between them and other variables.
- An incongruence occurs if a relationship exists between a year-sensitive and a non-year sensitive field.
- An undefined year-sensitive variable results in the propagation being incorrect.

The overall conversion process is greatly improved by investing sufficient time and resource into the seeding phase.

Refer to Chapter 5, “BYPASS2000 Tutorial” on page 57, for more information about seeding the Century Flag.

6.3.3 Assigning I/O Area to Related File

As discussed earlier, BYPASS2000 looks at the database to find year-sensitive fields. Most database files on the AS/400 system have an external description (DDS or DDL). However, if your database file is not externally described, a definition of the file (written in the same language as the programs you are going to convert) is required. Usually this definition already exists for the files and databases in the form of a copybook.

If you have a file that contains year-sensitive fields in different positions, you can identify the date fields using their relative positions (if you have this information). However, it is simpler to relate the layout of the file and identify the date fields within it. BYPASS2000 does not allow you to assign date fields for a file if the file does not have external definitions. To convey the detailed layout of the file, use the relevant options in the Field Assignment (seeding) menu or use a copybook source member for the I/O area.

6.3.3.1 Externalizing Internally Described Files

The main programming concepts are record format and layout.

Record Format

A file can have different kinds of records (for example, header record, detail record, and so on). We define each of these as a *record format*.

Layout
This is the definition of the fields in a record format. There can be different layouts for the same record (both in the copybook/include and inside the program).

If there is no external description for your files, there must be a copybook for each file being seeded. If you need to create one, the layout must be copied from the program in which the layout is defined. The copied layout must not be removed from the program because BYPASS2000 updates the existing program layout, as well as the newly created copybook.

BYPASS2000 works with the displacement of the seeded field rather than the name. Therefore, the dates are always in the same position, even if the field layout is different. Therefore, there must be enough information to uniquely identify the displacements of all seeded variables for each record format (see Section 6.3.1, “Multiple Format Files” on page 99, for more information on multiple format files).

After the copies in QCPYSRC are created and analyzed, they must be associated with the I/O area for the related file. If there are any old relationships, they must be removed. The copies can then be seeded.

6.3.4 Pre-Assigned Dates

Seeding provides the majority of information for the propagation phase. The propagation input is called the s*et of preloaded dates*. It is composed of the following dates:

- Seeded dates (dates defined from the database)
- System dates: UDATE/UYEAR (two-digit), *YEAR/*DATE (four-digit)
- Dates defined in copybooks

  Assume you have a copybook that is not seeded, but it contains a year-sensitive field that has been discovered by the propagation. All other programs using this copybook have this year-sensitive field identified as part of their own *set of preloaded dates.*
- Dates loaded by inter-program communication

  Any year-sensitive fields found in the parameter list for the program being called belong to the *set of preloaded dates* for the calling program.
- Dates loaded by inter-file propagation

  If a file is used in a program with an area that is different from the one seeded for the file, the area used in the program contains the same year-sensitive field for the seeded area.

6.3.5 Seeding Multiple Date Formats

Sometimes, a field can change its format type from database (YMD) to display (DMY). For this purpose, a work field is defined in the program, as shown in the following example:

```
          ....
 I     DS
 I                              1  60YDTE
          ....
 C          MOVE YDTE   WRKFL     60
 C          MOVELWRKFL  YY        20
 C          MOVE WRKFL  DD        20
 C          MOVELDD     WRKFL
 C          MOVE YY     WRKFL
          ....
```

If YDTE is in the YMD format, `WRKFL` receives the same format by propagation. Using the propagation result, `WRKFL` also receives data in the DMY format. BYPASS2000 considers this field as generic because the year is not always in the same position. Therefore, it is not possible to propagate a specific date type to DD and YY. As a result, for `MOVE YDTE WRKFL`, BYPASS2000 finds an incongruence because YDTE passes WRKFL a format that is not valid. Without resolving this incongruence, the conversion looks similar to the following example:

```
     ....
     I      DS
     I                            1  60YDTE
               ....
B2???C         MOVE YDTE    WRKFL    60
B2INF *CVR5011 sev.50 Field WRKFL is generic.
B2INF *The propagation engine found a
B2INF *relationship between a seeded date and
B2INF *non-date so the tool cannot decide
B2INF *whether it is a date field or not.
B2INF *The cause of this problem may be
B2INF *incorrect or insufficient seeding.
B2INF *Please check.
     C         MOVELWRKFL   YY       20
     C         MOVE WRKFL   DD       20
     C         MOVELDD      WRKFL
     C         MOVE YY      WRKFL
               ....
```

In this case, the cause is insufficient seeding. The field WRKFL must be seeded in the following ways:

- YMD (No expansion / Propagate with confirm)
- DMY (No expansion / Propagate with confirm)

When a field is seeded with more than one format, the year extension and propagation are difficult (due to the ambiguous information). Propagation with confirm ensures we find the date fields that may be loose in the propagation tree. Review the *Check Requested Information* option. If you find messages, such as `Field related with a non-propagated field`, these relate to DD and YY. If they are year-sensitive fields, they must be seeded. Selecting option **M** against the field that must be seeded in the Assign Date Field window allows you to seed the field

with multiple formats. When seeding is complete, the new conversion looks similar to the following example:

```
              ....
      I   DS
      I                              1   60YDTE
              ....
  B2TRCC        MOVE YDTE   WRKFL         60
  B2000C        MOVE WRKFL  DD            20
  B2REM ************************
  B2REM *** CONVERSION START ***
  B2REM ************************
  B2OLDC*       MOVELWRKFL  YY            20
  B2CHKC        MOVE '006P2'HB2&FA
  B2CHKC        MOVE WRKFL  HB2&F
  B2CHKC        MOVE '004P1'HB2&TA
  B2CHKC        EXSR HB&ADD                    ADD CENTURY
  B2CHKC        MOVE HB2&T  YY            40
  B2REM ************************
  B2REM *** CONVERSION  END  ***
  B2REM ************************
  B200C         MOVELDD     WRKFL
  B2TRCC        MOVE YY     WRKFL
              ....
  B2REM ************************
  B2REM *** CONVERSION START ***
  B2REM ************************
  B2NEWC/COPY HBP2CVTR
  B2REM ************************
  B2REM *** CONVERSION  END  ***
  B2REM ************************
```

Notice the B2CHK in the ADD CENTURY. This is inserted because BYPASS2000 must switch between two different date type formats. See Appendix C, “BYPASS2000 Markers” on page 165, for details of markers.

6.4 Propagation

The propagation engine uses the seeded fields from the database to discover all impacted fields throughout the application. Figure 115 on page 119 shows the propagation flow through BYPASS2000.

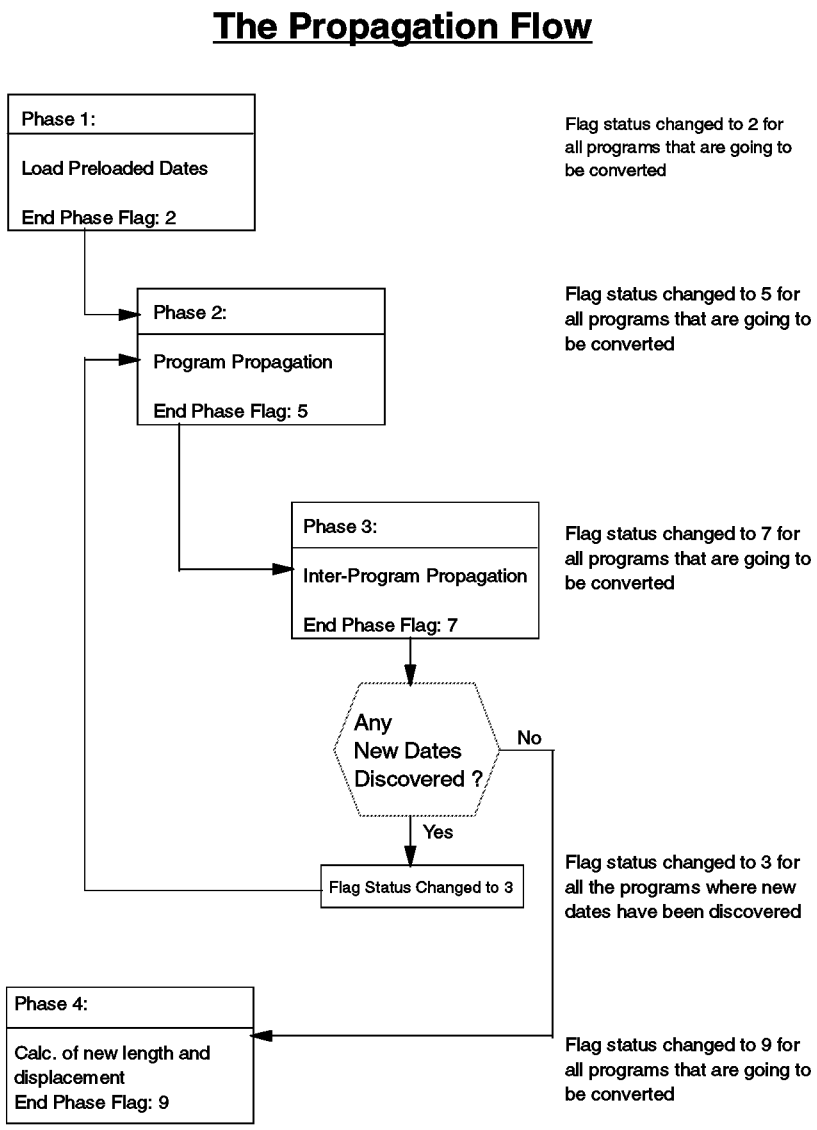


Figure 115. Propagation Flow through BYPASS2000

Table 11 on page 120 shows the status and frequency of phases 1 through 4 of the propagation flow through BYPASS2000.

Table 11. Phases 1 through 4 of the Propagation Flow through BYPASS2000

Phase	Status	Frequency
1	0 —> 2	Once only
2	2 —> 5	Repeated each time new seeds are assigned [1]
3	5 —> 7	Reprocessed only if phase 2 is repeated [2]
4	7 —> 3 or 7 —> 9	Each time new dates are discovered Only once
Notes: [1] Phase 2 is repeated if new year-sensitive fields are discovered after phase 3 is complete. [2] Phase 3 is repeated after phase 2 when new fields are discovered.		

During phase 2 and phase 3, it is also possible to see a value of 4 and 6 respectively. This is normal and simply means that an intermediate step is being processed. However, it may look as if the flag value changes straight from 7 to 5. In fact, it is changed back to 3. The step corresponding to value 4 is processed quickly.

6.4.1 Global and Single Propagation

The propagation stage is predominantly a global process. It is only by analyzing all the programs that it is possible to recognize all the inter-program relationships and propagate year-sensitive fields throughout.

Sometimes it may be necessary to run the propagation for a single program. This may be required if you make a seeding change that has no effect on other programs, or simply due to the duration of the global propagation. We recommend that the first propagation is run globally. If a subsequent single propagation is then performed, it is necessary to query the file ANDATOLR (see Appendix B, “Conversion Repository” on page 153) to understand if any new year-sensitive fields have been discovered but not processed (FLELAB = 'N'). If this is the case, you must decide whether to process again using the following methods:

- Global propagation
- Single propagation for programs that have the FLELAB = 'N' and then check table ANDATOLR again at the end of this process.

To process an inter-program propagation for specific programs, use one of the following methods:

- Set the propagation flags for the programs to 0, delete the propagation (on the Work with Programs display, select option **Q**), and rerun the global propagation, which is done only for programs with a flag set to 0.
- Put the programs that you do not want to propagate into hold status (option 8) and rerun the global propagation.

Notes:

1. A single P against the program only permits single propagation for that program alone.
2. It is imperative that a global propagation is run at the end of the investigative work.

6.5 Conversion

At this point, BYPASS2000 has enough information to take your converted code and modify it to make it Year 2000 ready. Refer to Section 5.7, “Step 6. Conversion” on page 92, for more details.

6.5.1 Understanding BYPASS2000 Code Inserted into Your New Source

When conversion of your programs is complete, BYPASS2000 inserts code into your new source. It does this so that your programs can run properly. The following topics provide the definitions of subroutines and parameters that BYPASS2000 may insert into your source code.

6.5.1.1 ADD and REMOVE CENTURY

This section defines the subroutines and parameters that BYPASS2000 may insert into your source code.

- **HB@68L ADD CENTURY** to a field type YMD — Routine optimized to reduce the lines of code inserted in customer program and to increase performance. Fast ADD century 6->8. This routine is present in the copy HBPSIM (QBP2000/QRPGSRC).
- **HB2@F6 FROM** — Input field
- **HB2@T8 TO** — Output field
- **HB@24Y ADD CENTURY** to a date field type Year — Routine optimized to reduce the lines of code inserted in customer program and to increase performance. Fast ADD century 2->4. This routine is present in the copy HBPSIM (QBP2000/QRPGSRC).
- **HB2@YY FROM** — Input field
- **HB2@Y4 TO** — Output field
- **HB@68R ADD CENTURY** to a date field type DMY — Routine optimized to reduce the lines of code inserted in customer program and to increase performance. Fast ADD century 6->8. This routine is present in the copy HBPSIM (QBP2000/QRPGSRC).
- **HB2@F6 FROM** — Input field
- **HB2@T8 TO** — Output field
- **HB@ADD ADD CENTURY** (standard data type) — This routine is present in the copy HBP24C (QBP2000/QRPGSRC).
- **HB2@FA FROM** — Input field attribute information:
  - First three digits (Field length)
  - Fourth digit (Field type):
    P = Packed
    X = Char

N = Zoned
B = Binary
1 = System Date

– Fifth digit (Data type):
1 Year
2 Year,Month,Day
3 Day,Month,Year / Month,Day,Year
4 Day-Month-Year / Month-Day-Year
5 Year and Julian day
6 Month,Year
7 Year,Month

- **HB2@F FROM** — Input field
- **HB2@TA TO** — Output field attribute information:
  – First three digits (Field length)
  – Fourth digit (Field type):
  P = Packed
  X = Char
  N = Zoned
  B = Binary
  1 = System Date
  – Fifth digit (Data type):
  1 Year
  2 Year,Month,Day
  3 Day,Month,Year / Month,Day,Year
  4 Day-Month-Year / Month-Day-Year
  5 Year and Julian day
  6 Month,Year
  7 Year,Month
- **HB2@T TO** — Output field
- **ADD@HB ADD CENTURY** (nonstandard data type) — This routine is present in the copy HBPNSC (QBP2000/QRPGSRC).
- **FA@HB2 FROM** — Input field attribute information:
  – First two digits (Field length)
  – Third digit (Field type):
  P = Packed
  X = Char
  N = Zoned
  B = Binary
  1 = System Date
  – Fourth and fifth digits (Year initial position)
  – Sixth digit (Year length)
- **F@HB2 FROM** — Input field
- **T@HB2 TO** — Output field
- **HB@RMV REMOVE CENTURY** (standard data type) — This routine is present in the copy HBP24C (QBP2000/QRPGSRC).

- **HB2@FA FROM** — Input field attribute information:
  - First three digits (Field length)
  - Fourth digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
  - Fifth digit (Data type):
    1 Year
    2 Year,Month,Day
    3 Day,Month,Year / Month,Day,Year
    4 Day-Month-Year / Month-Day-Year
    5 Year and Julian day
    6 Month,Year
    7 Year,Month
- **HB2@F FROM** — Input field
- **HB2@TA TO** — Output field attribute information:
  - First three digits (Field length)
  - Fourth digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
    1 = System Date
  - Fifth digit (Data type):
    1 Year
    2 Year,Month,Day
    3 Day,Month,Year / Month,Day,Year
    4 Day-Month-Year / Month-Day-Year
    5 Year and Julian day
    6 Month,Year
    7 Year,Month
- **HB2@T TO** — Output field
- **RMV@HB REMOVE CENTURY** (nonstandard data type) — This routine is present in the copy HBPNSC (QBP2000/QRPGSRC).
- **FA@HB2 FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
  - Fourth and fifth digits (Year initial position)
  - Sixth digit (Year length)
- **F@HB2 FROM** — Input field
- **T@HB2 TO** — Output field

- **HB23 ADD** and **REMOVE CENTURY** flag — This routine is present in the copy HBP23C (QBP2000/QRPGSRC). It refers to relationship that transform year (YY) in year with century flag (FYY) and back.
- **HB23FA FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
    1 = System Date (only for ADD)
  - Fourth and fifth digits (Year initial position)
  - Sixth digit (ADD or RMV century):
    A = Add century
    R = Remove century
- **HB23F FROM** — Input field
- **HB23T TO** — Output field
- **HB34 ADD** and **REMOVE CENTURY** flag — This routine is present in the copy HBP34C (QBP2000/QRPGSRC). It refers to relationship that transform year with century flag (FYY) in year with century flag completely expanded (CCYY) and back.
- **HB34FA FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
  - Fourth and fifth digits (Year initial position)
  - Sixth digit (ADD or RMV century):
    A = Add century
    R = Remove century
- **HB34F FROM** — Input field
- **HB34T TO** — Output field

  Note: All the routines described previously are present not only in source file QRPGSRC in library QBP2000, but also in source files QS36SRC, QS38SRC, and QRPGLESRC.

Figure 116 and Figure 117 on page 125 show the subroutines that were inserted into the new source code.

```
Customer Code
DBYMD     IFLE PRTYMD

V3R1M2 Conversion
MOVE PRTYMD    HB2@F6  6
EXSR HB@68L                  + CENTURY 6->8 LEFT
MOVE HB2@T8    H@8N01  80
DBYMD     IFLE H@8N01
```

Figure 116. Code Conversion - Add and Remove Century - Example 1

```
Customer Code
MOVE DSPY      DBY     2

V3R1M2 Conversion
MOVE DSPY      HB2@YY  2
EXSR HB@24Y                    + CENTURY 2->4 YEAR
MOVE HB2@Y4    DBY     4
```

Figure 117. Code Conversion - Add and Remove Century - Example 2

Legend:

DBY Database field with date type 1 (Year), Exp = 4, Prop = 0
DBYMD Database field with date type 2 (YMD), Exp = 4, Prop = 0
PRTYMD Printer field with date type 2 (YMD)

6.5.1.2 SHIFT Enabled

During set up environment, if you set the parameter Insert SHIFT instruction on system date parameter to Y, BYPASS2000 inserts date-shifting logic for system dates in the converted code. This means that BYPASSS2000 does not replace system date keywords.

The definitions of BYPASS2000 SHIFT subroutines are described here:

- **HB@SHF SHIFT** year (standard data type) —This routine is present in the copy HBP24C (QBP2000/QRPGSRC).
- **HB2@FA FROM** — Input field attribute information:
  - First three digits (Field length)
  - Fourth digit (Field type):
    1 = System Date
  - Fifth digit (Data type):
    1 Year
    2 Year,Month,Day
    3 Day,Month,Year / Month,Day,Year
    4 Day-Month-Year / Month-Day-Year
    5 Year and Julian day
    6 Month,Year
    7 Year,Month
- **HB2@FFROM** — Input field

- **HB2@TA TO** — Output field attribute information:
  - First three digits (Field length)
  - Fourth digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
  - Fifth digit (Data type):
    1 Year
    2 Year,Month,Day
    3 Day,Month,Year / Month,Day,Year
    4 Day-Month-Year / Month-Day-Year
    5 Year and Julian day
    6 Month,Year
    7 Year,Month
- **HB2@T TO** — Output field
- **SHF@HB SHIFT** year (nonstandard date type) — This routine is present in the copy HBPNSC (QBP2000/QRPGSRC).
- **FA@HB2 FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
    1 = System Date
  - Fourth and fifth digits (Year initial position)
  - Sixth digit (Year length)
- **F@HB2 FROM** — Input field
- **T@HB2 TO** — Output field

  Note: All the routines described previously are present, not only in source file QRPGSRC of library QBP2000, but also in source files QS36SRC, QS38SRC, and QRPGLESRC.

The examples in Figure 118 are SHIFT subroutines inserted into the converted code:

```
Customer Code
MOVE UDATE     DBYMD

V3R1M2 Conversion
MOVE '00613'   HB2@FA
MOVE UDATE     HB2@F
MOVE '008P3'   HB2@TA
EXSR HB@ADD                      ADD CENTURY
MOVE HB2@T     DBYMD
```

Figure 118. Code Conversion - Shift Enabled

Legend:

DBDMY Database field with date type 2 (YMD, Exp = 4, Prop = 0
DISPY Display field zoned (2,0) with date type 1 (YEAR)

6.5.2 Understanding Migration Routines

After the files are converted and recompiled, it is time to generate migration routines. These routines migrate your data so that it is Year 2000 compliant. This requires data mapping, which is done by BYPASS2000. However, the you should verify that their data has been migrated properly. This section contains the definition of the BYPASS2000 migration subroutines.

Important

Make sure that you have completed conversion of your files before you try to create any migration program. Otherwise, migration programs will not be created correctly.

- **DIXXXX** — Date field - Input
- **DOXXXX** — Date field - Output
- **NDXXXX** — Part of the record that contain consecutive areas that are non-year sensitive fields or date fields not to expand
- **ADD@HB ADD CENTURY** generic routine — This routine is present in the copy HBP2RPGMTC (QBP2000/QRPGSRC).
- **FA@HB2 FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
  - Fourth and fifth digit (Year initial position)
  - Sixth digit (Year length)
- **F@HB2 FROM** — Input field
- **T@HB2 TO** — Output field
- **HB23ADD and REMOVE CENTURY** flag — This routine is present in the copy HBP2RPGMTC (QBP2000/QRPGSRC). It refers to relationship that transform year (YY) in year with century flag (FYY) and back.
- **HB23FA FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
  - Fourth and fifth digits (Year position)

– Sixth digit(ADD or RMV century):
A = add century
R = remove century

- **HB23F FROM** — Input fieldB23T TO - Output field

Only date fields with two-digit year lengths seeded to expand are defined in the program shown in Figure 119. The other fields (date fields with four-digit year lengths or seeded not to expand and non-date fields) are defined in a single field whether they are consecutive.

```
* BYPASS2000 - MIGRATION PROGRAM FOR FILE FILEX1
     FFILEIN  IP  F      87            DISK
     FFILEOUT O   F      94            DISK
     E/COPY HBP2RPGMTE
     IFILEIN  AA
     I                                      1   1 ND0001
     I                                      2   70DI0001
     I                                      8   8 ND0002
     I                                    P   9  120DI0002
     I                                    P  13  160DI0003
     I                                       17  31 ND0003
     I                                    P  32  330DI0004
     I                                       34  78 ND0004
     I                                       79  82 DI0005
     I                                       83  87 ND0005
     I/COPY HBP2RPGMTI
     C                    MOVE '06N012'  FA@HB2
     C                    MOVE DI0001    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0001  80
     C*
     C                    MOVE '07P062'  FA@HB2
     C                    MOVE DI0002    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0002  90
     C*
     C                    MOVE '07P062'  FA@HB2
     C                    MOVE DI0003    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0003  90
     C*
     C                    MOVE '02P012'  FA@HB2
     C                    MOVE DI0004    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0004  50
     C*
     C                    MOVE '04X012'  FA@HB2
     C                    MOVE DI0005    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0005  6
     C*
     C                    EXCPTFM0001
     C/COPY HBP2RPGMTC
     OFILEOUT E               FM0001
     O                        ND0001     1
     O                        DO0001     9
     O                        ND0002    10
     O                        DO0002    15P
     O                        DO0003    20P
     O                        ND0003    35
     O                        DO0004    38P
     O                        ND0004    83
     O                        DO0005    89
     O                        ND0005    94
```

Figure 119. Example Migration Routines

For more information on COBOL migration code, see the Web site at: http://www.software.ibm.com/ad/as400/bypass/

6.5.3 Understanding Test Migration Routines

BYPASS2000 offers test migration routines to move your date fields in your files into the future by any amount. This section describes the BYPASS2000 subroutines.

Important

Make sure that you have completed conversion of your files and have compiled them before you try create any test migration program. Otherwise, the test migration programs will not be created.

- **DIXXXX** — Date field - Input
- **DOXXXX** — Date field - Output
- **NDXXXX** — Part of the record without any year-sensitive information
- **ADD@HB ADD CENTURY** generic routine — This routine is present in the copy HBP2RPGMTC (QBP2000/QRPGSRC).
- **SHF@HB SHIFT YEAR** generic routine — This routine is present in the copy HBP2RPGMTC (QBP2000/QRPGSRC).
- **FA@HB2 FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    Q = Packed
    X = Char
    N = Zoned
    B = Binary
  - Fourth and fifth digits (Year initial position)
  - Sixth digit (Year length)
- **F@HB2 FROM** — Input field
- **T@HB2 TO** — Output field
- **HB23FA FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    P = Packed
    X = Char
    N = Zoned
    B = Binary
  - Fourth and fifth digits (Year position)
  - Sixth digit (ADD or RMV century):
    A = Add century
    R = Remove century
- **HB23F FROM** — Input field
- **HB23T TO** — Output field

Program Example

All the date fields (both date fields with two-digit year lengths seeded to expand or not to expand and date fields with four-digit year lengths) are defined in this program. For date fields with two-digit year lengths seeded to expand, BYPASS2000 calls both the *add century* routine and the *shift* routine. For date fields with four-digit year lengths or seeded not to expand, it calls *shift* routine. The non-date fields are defined in a single field whether they are consecutive.

```
* BYPASS2000 - "TEST" MIGRATION PROGRAM FOR FILE FILEX1
      *
      * USE ONLY FOR TEST PURPOSES
      * DO NOT USE FOR PRODUCTION
      *
     FFILEIN  IP  F      87            DISK
     FFILEOUT O   F      94            DISK
     E/COPY HBP2RPGMTE
     IFILEIN  AA
     I                                      1   1 ND0001
     I                                      2   70DI0001
     I                                      8   8 ND0002
     I                                    P   9  120DI0002
     I                                    P  13  160DI0003
     I                                       17  31 ND0003
     I                                    P  32  330DI0004
     I                                       34  78 ND0004
     I                                       79  82 DI0005
     I                                       83  87 ND0005
     I/COPY HBP2RPGMTI
     C                    MOVE '061012'  FA@HB2
     C                    MOVE DI0001    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0001  80
     C*
     C                    MOVE '07Q062'  FA@HB2
     C                    MOVE DI0002    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0002  90
     C*
     C                    MOVE '07Q062'  FA@HB2
     C                    MOVE DI0003    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0003  90
     C*
     C                    MOVE '02Q012'  FA@HB2
     C                    MOVE DI0004    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0004  50
     C*
     C                    MOVE DI0005    F@HB2
     C                    EXSR ADD@HB
     C                    MOVE T@HB2     DO0005  6
     C*
     C                    EXCPTFM0001
     C/COPY HBP2RPGMTC
     OFILEOUT E               FM0001
     O                        ND0001     1
     O                        DO0001     9
     O                        ND0002    10
     O                        DO0002    15P
     O                        DO0003    20P
     O                        ND0003    35
     O                        DO0004    38P
     O                        ND0004    83
     O                        DO0005    89
     O                        ND0005    94
```

Figure 120. Test Migration Routines

For more information on COBOL migration code, see the Web site at: http://www.software.ibm.com/ad/as400/bypass/

6.5.4 Understanding Data Integrity Module (DIM)

BYPASS2000 also creates a module which tests if data being written into a file is of a certain format. Date formats that are not valid are detected by the module. This section shows the subroutines and variables used in the BYPASS2000 DIM.

Important

Make sure that you have completed conversion of your files and have compiled them before you try create any Data Integrity Module. Otherwise, no DIM is created.

- **DTXXXX** — Year or Date field - Input for DIM
- **WFMT FROM** — Input field attribute information:
  - First two digits (Field length)
  - Third digit (Field type):
    P= Packed
    X = Char
    N = Zoned
    B = Binary
  - Fourth and fifth digits (Year initial position)
- **WDATE FROM** — Input field
- **WNAME** — Field name
- **HB@CHK YEAR VALIDITY CHECK** generic routine — This routine is present in the copy HBP2DIMC (QBP2000/QRPGSRC).
- **FYYCHK YEAR VALIDITY CHECK** generic routine — This routine is present in the copy HBP2DIMC (QBP2000/QRPGSRC).
- **WATBL** — Internal field used by the tool
- **PGM** — Internal field used by the tool
- **PGR** — Internal field used by the tool
- **BUFFER** — Internal field used by the tool

The following example shows a data integrity module for a file.

```
* BYPASS2000 - DATA INTEGRITY MODULE FOR FILE FILEX1
      /COPY HBP2DIMEI
     IBUFFER      DS
     I                                        2   50DT0001
     I                                    P  11  150DT0002
     I                                    P  16  200DT0003
     I                                    P  36  380DT0004
     I                                       84  87 DT0005
     C           *ENTRY    PLIST
     C                     PARM           PGM    10
     C                     PARM           PRG     3
     C                     PARM           WATBL
     C                     PARM           BUFFER
     C                     MOVEL'FILEX1  'FILE   10
     C                     MOVE *BLANK    WATBL
     C                     MOVE *ZERO     I
     C                     MOVE DT0001    WDATE     P
     C                     MOVE '04N01'   WFMT
     C                     MOVEL'XSYMD'   WNAME
     C                     EXSR HB@CHK
     C*
     C                     MOVE DT0002    WDATE     P
     C                     MOVE '08P05'   WFMT
     C                     MOVEL'XSDMY'   WNAME
     C                     EXSR HB@CHK
     C*
     C                     MOVE DT0003    WDATE     P
     C                     MOVE '09P06'   WFMT
     C                     MOVEL'XSDMY2'  WNAME
     C                     EXSR HB@CHK
     C*
     C                     MOVE DT0004    WDATE     P
     C                     MOVE '04P01'   WFMT
     C                     MOVEL'XSYY'    WNAME
     C                     EXSR HB@CHK
     C*
     C                     MOVE DT0005    WDATE     P
     C                     MOVE '04X01'   WFMT
     C                     MOVEL'XSYM'    WNAME
     C                     EXSR HB@CHK
     C*
     C                     RETRN
     C/COPY HBP2DIMC
```

Figure 121. Example of a Data Integrity Module (DIM)

For more information about COBOL DIM, see the Web site at:
http://www.software.ibm.com/ad/as400/bypass/

6.6 Hints and Tips

Converting your applications using BYPASS2000 is not straightforward. You need a good understanding of the application to ensure the correct seeding is fed in for the propagation and conversion to be successful. You must also detect the reason for any generic fields and incongruences by understanding the program logic.

This section gives some hints on how to determine the cause of common problems and how to prevent them.

6.6.1 The Structural Solution

BYPASS2000 applies the *structural solution* to help resolve your application Year 2000 issues. In other words, every year-sensitive field discovered during propagation is extended to a four-digit year. The only exceptions to this rule are display (MAP) and report (PRT) year-sensitive fields. These keep the same

length and BYPASS2000 applies logic to add and remove the century (see Section 5.7.1.1, “Add Century Routine” on page 94, for more details).

If required, you can choose not to expand certain files, fields in files, or internal program areas. In this case, BYPASS2000 applies the same logic as for the display and report year-sensitive fields. If you choose not to expand a field, it is important to be aware of incongruences that may arise if this field is related to other date fields.

Consider the following example:

```
    IDT1     DS
    I                                     1 10 MM1
    I                                    11 12 AA1
    I                                    13 22 DD1
    IDT2     DS
    I                                     1 10 MM2
    I                                    11 12 AA2
    I                                    13 22 DD2
    C            MOVE DT1     DT2
```

Suppose we want to expand the year AA1 but do not want to expand AA2. The result after the conversion is shown in the following example:

```
     IDT1     DS
     I                                     1 10 MM1
B2MODI                                    11 14 AA1 1
B2MODI                                    15 24 DD1
     IDT2     DS
     I                                     1 10 MM2
     I                                    11 12 AA2 2
     I                                    13 22 DD2
B2???C            MOVE DT1     DT2 3
B2INF *CVR3013 sev.30 Field DT1 contains a year inside but
B2INF *        does not have a standard data-type
B2INF *CVR3013 sev.30 Field DT2 contains a year inside but
B2INF *        does not have a standard data-type4
```

Notes

1. AA1 is expanded.
2. AA2 is not expanded.
3. The group level relationship between one expanded area and a non-expanded area produces an incongruence that must be dealt with manually (BYPASS2000 does not apply logic for adding/removing the century).
4. This error shows how the position of the year within the field is important.

6.6.2 Fields Shared between PF and PRT/MAP Files

Suppose we have a field DDATE in both the database and the printer file. BYPASS2000 chooses to expand this field and informs the user that the printer field has been expanded as follows:

```
FFILE01 UP E             K      DISK
FPRTF01 O  E                    PRINTER
* Database Field
C                  MOVE UDATE      DDATE
* Non-Database Field
C                  MOVE UDATE      PRTDTE  60
The result after the conversion is shown in the following example:
         FFILE01 UP E              K      DISK
B2CHK    FPRTF01 O  E                     PRINTER
B2INF    *CVR3010 sev.30 Printer file PRTF01 must be converted because
B2INF    *    field DDATE (and others) is shared by name with other
B2INF    *    database fields which require expansion. Please convert
B2INF    *    PRTF01 enlarging all related fields prior to compilation,
B2INF    *    or manually insert the correct RENAME and MOVE statements
B2INF    *    in this source to avoid name-sharing.
         * Database Field
B2REM    ************************
B2REM    *** CONVERSION START ***
B2REM    ************************
B2OLDC*              MOVE UDATE    DDATE
B2NEWC               MOVE '00613'  HB2&FA
B2NEWC               MOVE UDATE    HB2&F
B2NEWC               MOVE '008N3'  HB2&TA
B2NEWC               EXSR HB&ADD             ADD CENTURY[1]
B2NEWC               MOVE HB2&T    DDATE
B2REM ************************
B2REM ***  CONVERSION END ***
B2REM ************************
      * Non-Database Field
B2REM ************************
B2REM *** CONVERSION START **
B2REM ************************
B2OLDC*              MOVE UDATE    PRTDTE  60
B2NEWC               MOVE '00613'  HB2&FA
B2NEWC               MOVE UDATE    HB2&F
B2NEWC               MOVE '006P3'  HB2&TA
B2NEWC               EXSR HB&SHF             SHIFT YEAR[2]
B2NEWC               MOVE HB2&T    PRTDTE  60
B2REM ************************
B2REM ***  CONVERSION END ***
B2REM ************************
```

Notes

1. Add Century routine is explained in Section 5.7.1.1, “Add Century Routine” on page 94.
2. Shift Year routine is explained in Section 5.5.4.2, “Century Window Routine” on page 82.

The same result is obtained when a display field is shared with a database field. BYPASS2000 works by expanding the database fields. When relationships exist between database fields and display or printer fields (and not between data structures containing fields to be and not to be expanded), BYPASS2000 inserts the routine for the *Add Century*. If these printer or display fields are shared with other database fields and are related to system fields, BYPASS2000 only inserts the routine for the *Shift Year* (see Section 5.7.1.1, “Add Century Routine” on page 94, and Section 5.5.4.2, “Century Window Routine” on page 82, for more details on the *Add Century* and *Shift Year* routines).

6.6.3 Seeding and Propagation

BYPASS2000 can recognize 15 different date types (see Appendix A, “Interfacing to BYPASS2000” on page 149, for this list) and allows seeding with a substring in packed or zoned fields.

Always ensure that the seeding information you have obtained (whether through a tool or manually) includes all the possible date fields. If this information is provided from an impact analysis tool, it is vital that you check the result.

If the propagation runs with insufficient input information, it is possible that generic areas (fields containing data of more than one type) are discovered, causing the propagation to end.

Propagation uses the set of predefined dates from the seeding to build its tree. However, each connection is terminated for the following reasons:

- There are no relationships between the last variable of a branch and other variables.
- There is a relationship between a variable (defined as a date) and a locked (but not seeded) date.

See the following example:

```
IREC1       DS
I                     1  20 FIL1
I                    21  26 DTE1
I                    27  35 FIL2
IREC2       DS
I                     1  20 FIL3
I                    21  26 NUM2
I                    27  35 FIL4
IREC3       DS
I                     1  20 FIL5
I                    21  26 QTY3
I                    27  35 FIL6
INODTE      DS
I                     1  15 FIL7
I                    16  21 WRKFL
                      ....
```

REC1 is locked and DTE1 is seeded. The other records do not contain dates. However, the following example shows how more information is required:

```
                ....
C               MOVE DTE1  WRKFL
C               MOVE WRKFL NUM2
C               MOVE WRKFL QTY3
                ....
```

During the propagation, `WRKFL` becomes a date. In turn, both NUM2 and QTY3 also have dates passed. `WRKFL` is a generic field. By seeding only REC1, you do not see any incongruences but both NUM2 and QTY3 are expanded. To prevent this from happening, you must lock REC2 and REC3. This produces incongruences, but the fields are not expanded.

To avoid incongruences, you must seed DTE1 using the *propagate with confirm* option. Do not forget that non-year sensitive fields (when related) can also stop the propagation chain. You can avoid this by locking them as non-year fields.

In addition to the database, other kinds of date fields may need to be seeded. Here is the recommended list:

- Seed the COPYs that are used by more than one program.

  If a COPY is used by several programs, the set of preloaded dates is duplicated. You can retrieve this information easily by querying the file, ANDATRPC (see Appendix B, “Conversion Repository” on page 153).

- Seed the parameter list of the most common routines (when COPYBOOK defined).

  You can retrieve this information by querying the file ANDATCCR (see Appendix B, “Conversion Repository” on page 153 for more information).

- Seed the parameters of routines with no source (when a COPYBOOK is defined).

  There are some situations where source code is not available for the program objects. In this case, inter-program-communication is not effective so dates within these parameters are not discovered.

6.6.3.1 Implicit Definitions for Year-Sensitive Fields

Consider a multi-format file defined as follows:

```
IREC1        DS
I                          1   2 TPE1
I                          3  18 KEY1
I                          3   8 KDTE
I                          9  18 FIL1
             ....
IREC2        DS
I                          1   2 FIL3
I                          3  18 KEY2
             ....
```

KEY2 may contain some date information even though the name does not imply this. Do not rely solely on the names of the fields because it is not always obvious from the field names whether they contain dates.

6.6.4 Date Handling Routines

We do not advise using BYPASS2000 to convert date handling routines. It is preferable to invest time to rewrite them with new logic to resolve the ambiguity.

The problem is that date handling routines generally use the same working area for different purposes and consequently using different date formats. This can affect the calling program during the propagation process, moving incorrect date information through the linkage area of the date handling routine.

The following list shows date handling routines that can cause propagation problems:

- A numeric algorithm date handling routine that changes the date format.
- A routine that calculates the difference between two dates.
- When the date format is received in the routine parameters and, therefore, it is impossible to know the date format until the program is called.

- If the date routine uses the same redefined area to contain different date information.
- A date routine that uses the same date field for input and output.

6.6.5 Incongruences

Incongruences are usually a result of a date and non-date field coming into contact with each other. The example in the following section describes how this can happen.

6.6.5.1 Link between a Year and Non-Year Sensitive Field

Consider the following examples:

```
R  REC01              (Locked)
   FLD01     5
   FLD021    2s 0
   FLD022    2s 0
   FLD023    2s 0
   FLD03     4
   YEARFL    2s 0     (Seeded)
The program is:
             ....
    C  FLD021 IFGT  YEARFL
```

YEARFL is a year-sensitive field (as it is part of the preloaded dates). However, FLD021 is not year-sensitive (it belongs to a locked area and is not seeded). The same thing happens for a copy or structure not related to a file but seeded with a lock in areas containing the two fields and the same seeding information.

The results after the conversion are shown in the following example:

```
          R  REC01            (Locked)
             FLD01    5
             FLD021   2s 0
             FLD022   2s 0
             FLD023   2s 0
             FLD03    4
   B2MOD     YEARFL   4s 0    (Seeded)

The program is:
             ....
   B2???C    FLD021  IFGT YEARFL
   B2INF *CVR0001 sev.00 Field FLD021 is not a date or a year
```

The preceding situation is symptomatic of incorrect seeding.

6.6.5.2 Generic Variables (or Work Fields)

Consider the following example:

```
    FILE01
           R REC01            (Locked)
             FLD01   5
             FLD021  2s 0
             FLD022  2s 0
             FLD023  2s 0
             FLD03   4
             YEARFL  2s 0     (Seeded)
   PROGRAM
```

```
          ....
C              MOVE  YEARFL   WCHKNU 18
C         CALL 'CHKNUM'
C              PARM     WCHKNU
C              PARM     WRCODE
          ....
C              MOVE  FLD03    WCHKNU
C         CALL 'CHKNUM'  WCHKNU
C              PARM     WCHKNU
C              PARM     WRCODE
```

WCHKNU is a generic field because it receives both year and non-year sensitive fields. To resolve this, WCHKNU must be defined as a year-sensitive field, setting it not to expand or propagate. Alternatively, you can choose Propagation with Confirm. This ensures that all fields propagating to WCHKNU are identified (see Chapter 5, “BYPASS2000 Tutorial” on page 57, for more details).

6.6.5.3 Fields Used to Change the Date Format

Consider the following example:

```
C          READ FILE1                   30
C               MOVE DBDATE WDATE
C          EXSR SWAP                    DAY / YEAR SWAP
C               MOVE WDATE  DSPFDT
          ....
C               MOVE DSPFDT WDATE
C          EXSR SWAP                    DAY / YEAR SWAP
C               MOVE WDATE  DBDATE
C               UPDATREC01
```

The WDATE field receives the date in YYMMDD format (from the database) and changes it to DDMMYY for display purposes. After you change and validate this, WDATE field receives the date in DDMMYY format from the display field and changes it back to YYMMDD for database storage.

The result is shown in the following example:

```
     C          READ FILE1                   30
B2???C               MOVE DBDATE WDATE
B2INF  *CVR5011 sev.50 Field WDATE is generic...
     C          EXSR SWAP                    DAY / YEAR SWAP
B2???C               MOVE WDATE  DSPFDT
B2INF  *CVR5011 sev.50 Field WDATE is generic...
B2INF  *CVR5011 sev.50 Field DSPFDT is generic...
               ....
B2???C               MOVE DSPFDT WDATE
B2INF  *CVR5011 sev.50 Field DSPFDT is generic...
B2INF  *CVR5011 sev.50 Field WDATE is generic...
     C          EXSR SWAP                    DAY / YEAR SWAP
B2???C               MOVE WDATE  DBDATE
B2INF  *CVR5011 sev.50 Field WDATE is generic...
     C                 UPDATREC01
```

To resolve this, WDATE must be seeded with both date formats (using option M from the Assign Date Field window) and set it not to expand but *Propagate with Confirm*. This reduces the number of incongruences.

The propagation engine can determine the correct date type for DSPFDT if it has definite contacts with other year-sensitive fields. Otherwise, BYPASS2000 requests further seeding for field DSPFDT from the user.

6.6.5.4 Adding a Variable to a Year-Sensitive Field

If a variable is added to a year-sensitive field, the result is dependent on the format of the receiving year. For example:

```
ADD WSINT WSYEAR
```

WSYEAR may be in CCYY (Century Year) instead of YY format. This is a true conceptual problem because, even if the field is expanded, the result may not be correct.Therefore, a manual check is required. After the propagation, there are messages, such as `Unknown propagation effect...`, in the Check Request Information, and in the converted code, there is a B2INF label to indicate where the problem lies.

6.6.5.5 Arithmetic Operations on Year-Sensitive Fields

There may be a calculation involving a year-sensitive field that is not a leap year calculation. This causes a conceptual problem and a manual check is required.

However, there are the following exceptions:

- Addition and subtraction of a constant value to or from the year-sensitive field.

```
   YY ADD 1 = YY1          YYYY ADD 1 = YYYY1
   96 ADD 1 = 97           1996 ADD 1 = 1997
(20)00 ADD 1 = 01 (?)      2000 ADD 1 = 2001
```

- Division on certain values.

```
YY DIV 10 = YY1          YYYY DIV 10 = YYYY1
YY DIV 10000 = YY1       YYYY DIV 100000 = YYYY1
YY DIV 4 = YY1           YYYY DIV 4 = YYYY1
```

- Multiplication of certain values.

```
YY MULT 100 = YY1        YYYY MULT 100 = YYYY1
YY MULT 10000 = YY1      YYYY MULT 10000 = YYYY1
YY MULT 100.0001 = YY1   YYYY MULT 100.0001 = YYYY1
YY MULT 10000.01 = YY1   YYYY MULT 10000.01 = YYYY1
```

The preceding results are all correct. Therefore, the B2000 marker is added instead of B2??? (see Appendix C, “BYPASS2000 Markers” on page 165, for a list of all the markers and their meanings). BYPASS2000 is unable to handle arithmetic operations between a field that is to be expanded and one that is not. This results in error message CVR5008, which says that the converter cannot handle arithmetic operations between expanded and not-expanded fields. Therefore, this must be handled manually.

6.6.5.6 Instructions Relating an Expanded and Non-Expanded I/O Area

Consider the following example:

```
IREC01    DS                                (display Input Record)
I                      1      5 FLD01
I                      6     70FLD021    (Year)
I                      8     90FLD022
I                     10    110FLD023
I                     12    150FLD03
I                     16    170YEAR01    (Year)
```

```
     IREC02    DS                                    (Program I/O Area)
     I                      1       5FLD21
     I                      6      70FLD221    (Year)
     I                      8      90FLD222
     I                     10     110FLD223
     I                     12     150FLD23
     I                     16     170YEAR02    (Year)
          ....
     C            MOVE REC01    REC02
```

Fields within REC01 must not be expanded because they are display fields. The result is shown in the following example:

```
     IREC01    DS
     I                      1      5 FLD01
     I                      6      70FLD021     (Year)
     I                      8      90FLD022
     I                     10     110FLD023
     I                     12     150FLD03
     I                     16     170YEAR01     (Year)
     IREC02    DS
     I                      1       5FLD21
B2MODI                      6      90FLD221     (Year)
B2MODI                     10     110FLD222
B2MODI                     12     130FLD223
B2MODI                     14     170FLD23
B2MODI                     18     210YEAR02     (Year)
          ....
B2???C            MOVE REC01    REC02
B2INF  *CVR3012 sev.30 Field REC01 has more than one year inside
```

This happens because instructions that add or remove centuries can be inserted only for the standard year-sensitive field types.

6.6.6 Repeating Certain Steps in the Process

Figure 122 on page 141 shows the BYPASS2000 flow and how and when certain steps are repeated.

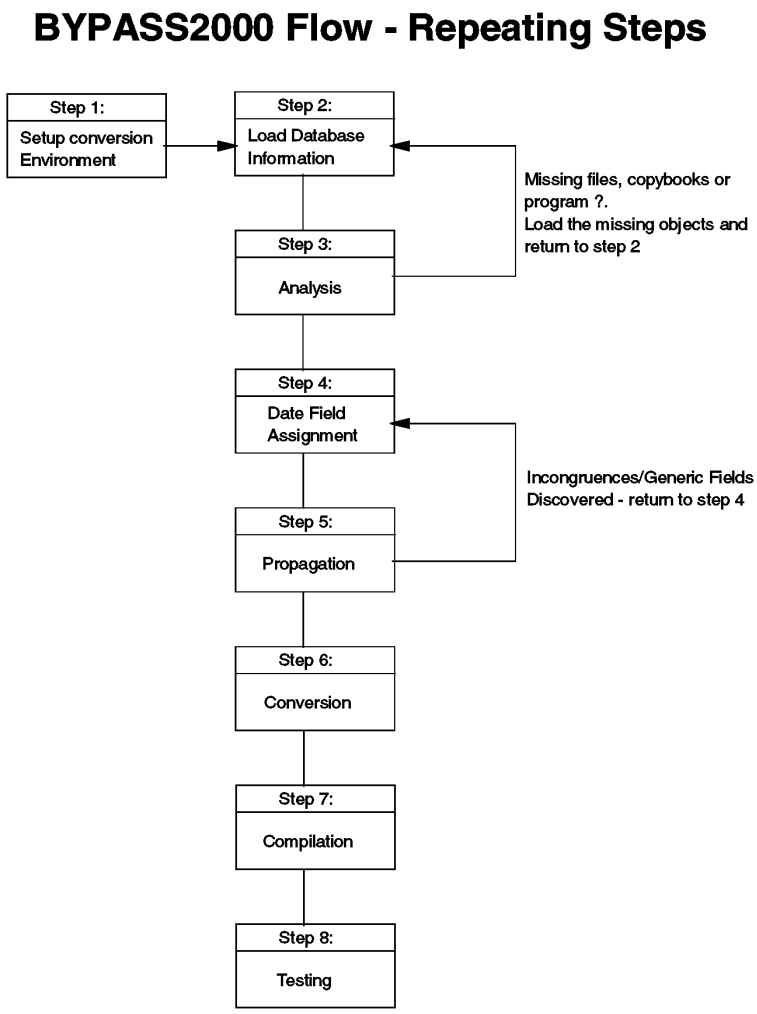


Figure 122. The BYPASS2000 Steps to a Successful Migration

The conversion process can be divided into the following phases:

- Analysis (memory-level and propagation) phase — Updates the repository.
- Conversion phase — Produces output, repository not updated.

It is possible that certain steps of the analysis process may need to be repeated. If stages of the first phase (see preceding example) must be rerun, then all information previously recorded must be deleted. If the DDS must be analyzed again, then all the old DDS analysis and all programs using this DDS must be deleted. However, if certain stages of the conversion phase must be repeated, it is sufficient to set the corresponding flag to 0.

If the option to delete a phase is selected, this function automatically sets the corresponding status flag to 0, deletes the relevant messages in the Display Conversion Log, and also deletes those messages that have not been checked in the Check Request Information. When a propagation phase is deleted, the corresponding propagation trace and generic areas are also deleted. In this case, the status flag cannot be set to 0 manually because there are corresponding

interactive functions that are required. In this case, selecting option Q from the Work with Programs display is necessary.

Information that has been entered manually or retrieved by BYPASS2000 is stored in the repository. It is never deleted unless explicitly requested by you. This is done by either removing seeding, deleting the analysis, or propagation.

The following repository files are not updated by the delete steps:

- HSDATFLD — Assigned fields (seeding file)
- HSDATINQ — User inquiry
- HSDATLCK — Locked areas
- HSDATPAR — Parameter table
- USxxxxxx — User entry files

Table 12 shows whether the repository is updated for a given function.

Table 12. BYPASS2000 - Given a Function, Is the Repository Updated

Function	**Repository Updated?**	**Deletion Steps**
Load AS/400 Database Information	Y	Automatically restarted - no deletion required Memory Level
Analyze copy	Y	Delete copy analysis Delete copy and program analysis
Analyze SQL table definition source	Y	Delete SQL table analysis
Analyze program	Y	Delete program analysis Delete copy and program analysis
Perform program propagation	Y	Delete propagation level analysis
Convert Copy *	N	"0" for CONV.FLAG in the source listing
Convert Program *	N	"0" for CONV.FLAG in the source listing
Create migration program *	N	"0" for MIGR.FLAG in the source listing
Create test migration program *	N	"0" for TXT FLAG in the source listing
Create DIM program *	N	"0" for DIM FLAG in the source listing
* - A manual interactive process		

The Table 13 shows the impact of each deletion step.

Table 13. BYPASS2000 - The Impact of a Deletion Step

Deletion Step	Impact
Delete program analysis	Remove program data field propagation analysis information. Remove the program analysis information (memory level). Set the PROP flag to "0". Set the CONV flag to 0.
Delete copy and program analysis	Remove the program date field propagation analysis information. Remove the program analysis information (memory level). Remove the copy analysis information (memory level). Set the ANA flag to 0 for the programs and copies. Set the PROP flag to 0 for programs and copies. Set the CONV flag to 0 for programs and copies.
Delete propagation level analysis	Remove the program date field propagation analysis information. Remove the CLP date field propagation analysis information (global). Set the PROP flag to 0. Set the CONV flag to 0.

6.6.7 Making Changes

During the conversion process, changes may need to be made to the application. The next section describes what must be done on the BYPASS2000 side to continue with the process.

6.6.7.1 Source Copy Is Changed

The BYPASS2000 side must perform these steps before the conversion process can continue:

1. If seeded, remove the seeding.
2. Remove the assignment to the file.
3. Delete the program memory-level analysis for the programs using that copy (this also deletes the propagation).
4. Delete the copy memory-level analysis.
5. Analyze the programs (memory-level).
6. Assign the new copy to the file.
7. Seed the copy.
8. To see the new result, run program date field propagation for copy and programs. You can check file ANDATCCR to ensure the new copy is associated.
9. Convert the copy.
10. Convert the programs.

6.6.7.2 Physical File Is Changed

The BYPASS2000 side must perform these steps before the conversion process can continue:

1. If seeded, remove the seeding.
2. Delete the program memory level analysis for the programs, copy, and CLP using that PF (this, in turn, deletes the propagation).
3. Delete the program, CLP, and copy memory-level analysis.
4. Analyze the PF (memory-level).
5. Analyze the programs and CLP (memory-level).
6. Seed the PF.
7. Run program date field propagation for copy, CLP, and programs. Analyze ANDATCCR to ensure the new copy is associated.
8. Convert the PF.
9. Convert the copy.
10. Convert the programs and the CLP.

6.6.7.3 Changing a Source Program

The BYPASS2000 side must perform these steps before the conversion process can continue:

1. Remove any seeding done on the fields in the programs working storage.
2. Delete program memory-level analysis.
3. Analyze the program (memory-level).
4. Reseed the working storage fields.
5. Run program date field propagation.
6. Convert the program.

6.6.7.4 Changing the Seeding (PF, RPG, CLP, COPY)

The BYPASS2000 side must perform these steps before the conversion process can continue:

1. Change the seeding of the field in either PF, RPG, CLP, or COPY.
2. Delete program date field analysis for programs, copy, and CLP using the PF (this also deletes the propagation), run program date field propagation, and convert again.

6.6.7.5 Inserting a New Physical File

The BYPASS2000 side must perform these steps before the conversion process can continue:

1. Insert the physical file in your old data file library (xxxxDAT if you took the default).
2. Take the option to Load AS/400 Database Information.
3. Include the new member (F15) in Work with File (memory-level).
4. Analyze the PF (memory-level).

5. Seed the PF.
6. Convert the PF.

6.6.7.6 Inserting a New Program

The BYPASS2000 side must perform these steps before the conversion process can continue:

1. Add the PGM to xxxxOLD (your old source programs library) in QCLPSRC (for CLP), QRPGSRC (for RPG), and QCBLSRC (for COBOL).
2. Include the new member (F15) in "Work with Program" (memory-level).
3. Analyze the PGM (memory-level).
4. Seed the program if required.
5. Run program date field propagation. Analyze file ANDATCCR to ensure this program has been associated.
6. Convert the PGM and the other programs involved.

6.6.7.7 Modifying a Source Program

The BYPASS2000 side must perform these steps before the conversion process can continue:

1. If seeded, remove the seeding.
2. After modifying the source, delete the propagation memory-level analysis for the program.
3. Analyze the PGM (memory-level).
4. Seed the program if required.
5. Run program date field propagation. Analyze file ANDATCCR to ensure the source listing has been updated.
6. Convert the PGM and the other programs involved.

6.6.8 Performance Strategies

This topic discusses performance considerations when converting your applications using BYPASS2000 and general AS/400 performance considerations when moving to the Year 2000.

6.6.8.1 Inter-Program Communication

The propagation process flows through many different cycles (see Figure 115 on page 119, for details of the flow). Table 14 shows the main difference between analysis and propagation. For analysis, the entire execution is completed program-by-program. Propagation runs through each program one step at a time.

Table 14. BYPASS2000 - Inter-Program Communication

Analysis		Propagation	
Step	**Program**	**Step**	**Program**
1	1	1	1
2	1	1	2
3	1	1	3
...		...	

Analysis		Propagation	
Step	**Program**	**Step**	**Program**
n	1	1	n
1	2	2	1
2	2	2	2
3	2	2	3

The reason for the difference is because propagation needs to consider all the links between programs. Therefore, any date field found in a program must be propagated to all other programs linked to it.

6.6.8.2 Access Paths

Because of the numerous factors that affect processing time, it is almost impossible to produce a formula to calculate the length of time it takes to rebuild access paths for the Year 2000. Some of the factors that must be taken into consideration are:

- Type of processor
- Workload during access path build
- Number of key fields
- Number of records in the physical files
- Programming development environment

For optimum performance, consider the ACCPTHSIZ field in Create Physical File (CRTPF) command. Systems at V3R6M0 and later have the value of *MAX1TB for the access path size. This is essential if there is high contention for keys and also gives better performance results.

6.6.9 Expanding PRT/MAP Fields

As previously discussed, BYPASS2000 does not expand display or printer reports. However, it may be that some applications require display and report fields to be expanded, which is possible by removing the MAP/PRT fields that you want to expand from the list of fields not to be expanded. See Section 5.5.4, “Display and Printer Fields” on page 82, for more details.

6.6.10 Authorization Violation

If you receive an `Authorization Violation` message when attempting certain stages of the process, this means that you have not entered your software license key. You can obtain your license key from HAL (as per the instructions in the installation guide). Then, follow these steps:

1. Type the following command on a command line and press **Enter**.
   `ADDLIBLE QBP2000.`
2. Enter `BPPR` to work with BYPASS enabling.
3. Enter your software key.

6.6.11 Seeding Unsupported Date Types

BYPASS2000 can handle 12 different date types (see Appendix A, “Interfacing to BYPASS2000” on page 149, for the detailed list). If your database contains dates in an unsupported format, it highlights this instruction and requires manual intervention.

BYPASS2000 does not support data areas as objects. However, it does recognize a data area as an input/output area in the program. However, the data area is not updated or flagged in any way by BYPASS2000. Therefore, it is important that you check your application for use of data areas to ensure your results are correct. For example, the following sample data area contains a date field of six digits.

```
                              Display Data Area

Data area . . . . . :   BYPASS
  Library . . . . . :     BYPASS
Type  . . . . . :   *DEC
Length  . . . . . :   6 0
Text  . . . . . :   BYPASS2000 DATE DATA AREA
Value . . . . . :   061097
```

Figure 123. Display Data Area

This information may have been stored in this way to reduce disk space. When this data is passed to a data structure in a program already converted by BYPASS2000, it is expanded to eight digits to be 06101997. However, if the value is then returned to the data area, the two extra digits are lost as the data area can only hold six digits.

6.6.12 Migrating from V3R1M1 to V3R1M2

If you have been converting your applications using V3R1M1 (the previous version of BYPASS2000) and you want to upgrade to V3R1M2 and continue with further conversions, run the following command:

```
BPMGRR2R3
```

This command is for BYPASS2000 migration from V3R1M1 TO V3R1M2. This command should be run when you are in the environment you want to convert, for example, from a command line on the BYPASS2000 menu. This allows you to take advantage of the new functions in the new release.

6.6.13 SST and Concat Functions

The SST and Concat functions found in logical file definitions are not supported in the current version of BYPASS2000. To ensure that BYPASS2000 converts date fields that use these functions, it is necessary to perform these steps:

1. Use the following command to retrieve all of the members that use these functions.

   ```
   FNDSTRPDM
   ```

2. Insert them manually using the Work with DDS display:
    a. Press **F6** to create.
    b. Enter the name of the logical file in the DDS name.
    c. Enter **DDS** for Language.
    d. Enter **DDS** for Category.
    e. Press **Enter** two times.
    f. Press **F3** and your logical file is shown.
3. Seed the logical files.
4. Propagate and convert.

SST (substring) is not handled if it is a variable because the value for this variable is not known until run time.

Appendix A. Interfacing to BYPASS2000

Impact analysis tools can use the HSDATDFI file to *seed* (assign) date fields into BYPASS2000. At the end of this appendix, it shows how to feed the information in HSDATDFI to BYPASS2000.

A.1 File Description

Table 15 provides a description of the HSDATDFI file.

Table 15. HSDATFI Description (Record Length: 135)

Field name	Type	Length	Field description
FLDNAME	CHAR	30	Field name
PARENTNAME	CHAR	30	Must be blank
IOAREANAME	CHAR	30	I/O Area name. Name of Record format or Data Structure of which this field is a part. See Note 1.
SRCNAME	CHAR	10	Source member name. See Note 2.
SRCTYPE	CHAR	3	Source type. DDS, CPY or PGM. See Note 3.
DTFTYPE	ZONED	3,0	Date field type. See Note 4.
DTYLEN	ZONED	1,0	Length of year portion of date. Only valid values are 2 and 4.
EXPTYPE	CHAR	1	Expansion type. 0 = Expand field. 1 = Do not expand field. See Note 5.
FILTYPE	CHAR	4	File type. See Note 6.
FILNAME	CHAR	10	File name
FLGLOCK	CHAR	1	Area to be locked. Y = YES. N = NO. See Note 7.
PRPTYPE	CHAR	1	Propagation type. 0 = Propagate. 1 = Do not propagate. See Note 8.
FLDSSTPOS	ZONED	5,0	Field substring start position. See Note 9.
FLDSSTLEN	ZONED	5,0	Field substring length. If not using substring, it should be the field length.
FLGELAB	CHAR	1	Internal use only. Set to ″N″.

Notes:

1. **I/O area name**

   For file fields, the I/O Area Name is the record format name preceded by an asterisk (*). If the field is contained in a data structure, the I/O Area Name is the name of that data structure. If the field does not relate to any data structure or record format, the I/O Area Name is the same as the field name.

2. **Source member name**

   Libraries and source files to be used are specified to BYPASS2000 by you as part of creating the conversion environment. Although multiple source files may be specified in each category, BYPASS2000 assumes that the combination of source member name and source type is unique. If the application to be converted contains multiple members with the same name and type, this situation must be corrected before BYPASS2000 analyzes the source.

   The Impact Analysis tool should warn the you of this situation.

3. **Source member type**

   Do not confuse this with the regular SEU type. This simply identifies to BYPASS2000 whether the member represents a program (PGM), file (DDS),

or copy member (CPY). Notice that copy members include those used to provide a file description for files that are internally described.

4. **Date field type**

   If you encounter date types that are not in the following list, identify them as type 001 (Year) and make the appropriate entries in the substring position and substring length fields (FLDSSTPOS & FLDSSTLEN).

Table 16. Code Description

Code	Short	Description
001	Year	Year
002	YMD	Year, month, day
003	DMY	Day, month, year or month, day, year
004	D-M-Y	Day, month, year or month, day, year with separators
005	Y-JUL	Year and Julian day
006	MY	Month, year
007	YM	Year, month
010	Cent.	Century (for example, the century portion of a CCYYMMDD date)
011	C-Yea	Complement of year
012	C-YMD	Complement of year, month, day
015	C-JUL	Complement of year, Julian day
017	C-YM	Complement of year, month

5. **Expansion type**

   The normal action of BYPASS2000 is not to extend any dates encountered on display files, printer files, or O-specs (they normally have an EXPTYPE of 1). You can override this on a field-by-field basis. It is up to the Impact Tool vendor to determine if they provide some means for you to control this value or if they simply follow BYPASS2000 conventions and allow you to make changes to specific fields through the regular BYPASS2000 interface.

6. **File type and file name**

   Only physical files, program fields and copy fields can be seeded. For a program or copy, you must put SPACES in these fields. For physical files, the file type field must be set to PF and the file name set to the name of the file. Any logical files using the physical file automatically receives their seeding.

7. **Area to be locked**

   It is unlikely that you will ever want to set this to anything other than N. If set to Y, BYPASS2000 takes it as true that this field can never contain a date. We strongly recommend that the lock option not be used by any tools building this file.

8. **Propagation type**

   This option determines whether any field that this field touches is automatically considered a date. Normal operation is to propagate the field. Therefore, the value should be zero (0). Again, it is unlikely that an impact analysis tool would ever need to set any other value.

9. **Substring start position and length**

   If the whole of the field is a date, then the position (FLDSSTPOS) should be set to 1 and the length (FLDSSTLEN) set to the length of the field. If the date is contained within a larger field, the position should be set to the offset within the field and the length to the length of the date.

   For example, if an invoice number is on the database as a single field but contains a year and month (for example, format nnnnnYYMM), the following two options apply:

   – **Seed the full date**

     In this case, DTFTYPE is 007, FLDSSTPOS is 6, and FLDSSTLEN is 4.

   – **Seed only the year**

     In this case, DTFTYPE is 001, FLDSSTPOS is 6, and FLDSSTLEN is 2.

   If the format of the date portion is nnnnnMMYY, then the following two options apply:

   – **Seed the full date**

     In this case, DTFTYPE is 006, FLDSSTPOS is 6, and FLDSSTLEN is 4.

   – **Seed only the year**

     In this case, DTFTYPE is 001, FLDSSTPOS is 8, and FLDSSTLEN is 2.

A.2 Loading the Interface File into BYPASS2000

After BYPASS2000 completes the analysis phase, the interface file can be applied. To apply the interface file, follow these steps:

1. Copy the interface file into the xxxxDB library (where xxxx is your environment name).
2. While in the environment (you have already run the command BP2000 xxxxDB), run the command `BPLOADDFI`. BYPASS2000 processes the interface file.

   You can select option **11** (Import external seeding) from the BYPASS2000 - Date-Field Assignment menu as shown in Figure 59 on page 72. Option 11 runs the BPLOADDFI command.
3. Now check the conversion log. BYPASS2000 reports any errors encountered while loading the file in this log.
4. Use the field-level assignment options to review the date assignments.

Appendix B. Conversion Repository

The conversion repository is made up of about 75 relational tables that can be analyzed using native AS/400 query or SQL.

Each table contains information relating to certain steps of the conversion process. For example, file ANDATFLD contains the information related to the fields defined in copybooks and programs for the entire application.

B.1 Naming Convention

The naming convention for the files is shown here:

```
              XX YYY ZZZ
where,
XX is the process identifier:       AN  Analysis
                                    CV  Conversion
                                    HS  List of Sources
                                    US  Work Tables
                                    XW  Work Tables

YYY is the type of activity:        DAT Copy and program analysis
                                        for BYPASS2000
                                    SQL SQL/DB2 Databases
                                    ERR Generic Use
                                    TAB Parameter Table

ZZZ is the content identifier:
                                    FLD Fields
                                    PAR Parameters
                                    LIB Libraries
```

B.2 Conversion Repository Files

If you want to view the contents of a particular file, use Query or SQL for each file in library BPxxDB. The tables in the following sections give the description of each file and in which phase they are updated or populated.

B.2.1 Conversion Environment

Table 17. Create Conversion Environment

File	Description
HSDATLIB	List of old/new libraries
HSDATPAR	List of all BP parameters
HSDATSDT	List of system date fields
HSDATTYF	List of data types

B.2.2 Load AS/400 Database Information

Table 18. Load AS/400 Database Information

File	Description
ANDATFFD	Field file descriptions
ANDATDBR	Database relationships
ANDATDBK	Database keys
HSDATSRC	List of source files
ANDATFMT	Database record format
HSDATROS	Object/source relationship
CVERRLOG	Analysis/conversion error messages

B.2.3 Memory Level Analysis

Table 19. Analyze DDS (Flag Status = 9)

File	Description
ANDATFLD	Memory level field definitions
ANDATVAL	List of source files with their status
HSDATFLS	DIM/MGR/TXT routines generated for each source file
HSDATFIL	File/COPYBOOK relationship
HSDATLCK	Locked I/O areas
HSDATFLD	List of date fields
CVERRLOG	Analysis/conversion error messages

Table 20. Analyze Copy (Flag Status = 9

File	Description
ANDATFLD	Memory level field definitions
ANDATVAL	List of source files with their status
HSDATFLS	DIM/MGR/TXT routines generated for each source file
HSDATFIL	File/COPYBOOK relationship
HSDATLCK	Locked I/O areas
HSDATFLD	List of date fields
CVERRLOG	Analysis/conversion error messages

B.2.4 Seeding

Table 21. Analyze SQL Table Definition Source (Flag Status = 9)

File	Description
ANDATTDF	Table definitions
ANDATRVT	Relationship between view and table
ANDATFLD	Field definitions
ANDATFIL	Date field types
HSDATFIL	File listing
HSDATLCK	Locked areas
HSDATSQL	List of SQL files and flag status
CVERRLOG	Analysis/conversion error log

Table 22. Analyze Program (Flag Status = 9)

File	Description
HSDATINQ	Inquiry messages
HSDATSRC	List of source files
ANDATRPC	Source/COPYBOOK relationships
ANDATFLD	Field definitions
ANDATVAL	Default for variables
ANDATSQI	List of SQL statements
ANDATCST	List of code statements
ANDATRPF	Program/file relationships
ANDATPRO	Basic relationship between variables
ANDATQST	List of SQL statements
USDATCCR	Caller-Called relationship
ANDATRNM	List of field renames in programs
ANDATRFF	Internal/external field name relationships
ANDATRFA	Relationships between files and IO areas
ANDATNOE	Non-expandable areas
USDATPNE	Details on non-expanded areas
ANDATNOP	Areas which will not be propagated
ANDATCCR	Relationship between caller and called program
ANDATVRC	Physical/virtual copy relationships
CVERRLOG	Analysis/conversion error log

B.2.5 Seeding

Table 23. Seeding Repository (Flag Status = 9)

File	Description
HSDATFLD	List of date fields
HSDATDFI	Seeding from external interface
HSDATSDT	General seeding
ANDATFIL	List of date types
HSDATFIL	List of files
HSDATLCK	Locked areas
USDATNOE	Details on areas which are not expanded
HSDATTYD	Default date type from HSDATTYF

B.2.6 Propagation Level Analysis

Table 24. Analyze Program Date Field Propagation (Flag Status = 9)

File	Description
ANDATOLR	List of all dates found in the programs
ANDATIPT	Inter-program trace
ANDATOLC	List of all dates found in COPYBOOKs
ANDATPTR	Propagation trace
ANDATNOE	Areas which are not to be expanded
ANDATNOP	Areas which are not to be propagated
ANDATDNA	Images of areas related to date fields
ANDATDFR	Origins of all year-sensitive fields
ANDATFLD	Memory field definition
ANDATFIL	List of date fields
HSDATINQ	Inquiry messages

B.2.7 Conversion

Table 25. Files from Conversion Phase (Flag Status = 9)

File	Description
CVDATRPT	Report definition
CVDATRPL	Report log
ANDATFLD	Memory field definition
CVDATCNV	Report log

B.2.8 Files that Contain List of Source

Table 26. Files Containing Source

File Name	Type	SRCTYPE
HSDATSRC	Copy	CPY
HSDATSRC	Programs	PGM
HSDATSQL	DDL SQL	DDS
HSDATSRC	DDS	DDS

B.2.9 Additional Files

Table 27. Conversion Repository

File Name	Description
ANDATDCL	DCLGEN definitions
ANDATDIZ	New long-date ref-fields in dictionaries
ANDATRPD	Relationship between program and DDS by means of COPY DDS
ANDATTRE	Date origins
CVDBGLOG	Debug log
CVSQLPFM	SQL Performance monitor
CVTABPAR	Parameter Table
HSCCPYFIL	List of source copy files
HSDATAPP	Identification application code
HSDATFRN	File routine names
HSDATOVR	Preloaded file overrides (DB, DSP, PRT)
HSDATRRA	Relations between record format and program I/O area
HSDATUEX	User exit
HSDBGLOG	Guide for messages that have been sent
USDATFLD	User's field information
USDATOLR	User's choice about propagated fields
USDATPRG	User defined multiple I/O area assignment
USDATRPF	Variables file name inventory list
USDATPRM	Called information

B.3 Detailed Description of Certain Files

This section gives more details for some of the files listed in the previous tables.

B.3.1 ANDATCCR

BP2000 -Relationship between caller and called program. Here, it is possible to find all of the relationships between the connected programs. This table is important to run an inter-program between a small part of the programs.

Table 28 is populated in the analyze program phase.

Table 28. ANDATCCR

Field	Type	Length	Dec.	Domain
CALLTYPE	CHARACTER	10	0	Call Type
CALLERNAME	CHARACTER	10	0	Caller Name
CALLEDNAME	CHARACTER	10	0	Called Name
PARMPRG	PACKED	2	0	Parameter Program
IOAREANAME	CHARACTER	30	0	IO Area Name
IOAREAPRG	PACKED	2	0	IO Area Program
IOAREASCP	PACKED	2	0	IO Area Scope
FLDDISPL	PACKED	3	0	Field Displacement
FLDNAME	CHARACTER	30	0	Field Name
FLDSCP	PACKED	2	0	Field Scope

B.3.2 ANDATCST

Table 29 tracks the program code-statement. It is populated in analyze program phase.

Table 29. ANDATCST

Field	Type	Length	Dec.	Domain
PGMNAME	CHARACTER	10	0	Program Name
CPYNAME	CHARACTER	10	0	Copy Name
STMTNBR	PACKED	4	0	Statement Number
STMTPRG	PACKED	2	0	Statement Program
STMTSCP	PACKED	2	0	Statement Scope
STMTNAME	CHARACTER	16	0	Statement Name
STMTOPER1	CHARACTER	30	0	Statement Operand
STMTOPER2	CHARACTER	30	0	Statement Operand
STMTKWD	CHARACTER	10	0	Statement Keyword
FLELAB	CHARACTER	1	0	Elab.Flag, P=Tree of Propagation N=No Tree of Propagation Y=Comput.Instructions
FLSIGNIF	CHARACTER	1	0	Signif.Flag N=No comput.instruct. C=Instruc.Compl. U=Instr.USA P=Other Data type
FLCOMPLEX	CHARACTER	1	0	Complex flag, Y=Multiple Instruction
FLSENSIT	CHARACTER	1	0	Sensitive Flag, Y=Year Sensitive (only for COBOL/400)
NSTMTSTR	PACKED	4	0	Start Statement Number
NSTMTEND	PACKED	4	0	End Statement Number
STMTOPERAZ	CHARACTER	10	0	Conversion Statement Operation

B.3.3 ANDATDBK

Table 30 tracks database keys. It is populated in load AS/400 database information phase.

Table 30. ANDATDBK (Part 1 of 2)

Field	Type	Length	Dec.	Domain
APRCEN	CHARACTER	1	0	Century Retrieval: 0=20th, 1=21st
APRDAT	CHARACTER	6	0	Date Retrieval: YY/MM/DD
APRTIM	CHARACTER	6	0	Time Retrieval: hour/minutes/seconds
APFILE	CHARACTER	10	0	File
APLIB	CHARACTER	10	0	Library
APFTYP	CHARACTER	1	0	Type of File: P=PF, L=LF, R=DDM PF, S=DDM LF
APFILA	CHARACTER	4	0	File Attribute: *PHY or *LGL
APMXD	CHARACTER	3	0	Reserved
APFATR	CHARACTER	6	0	File Attribute: PF, LF, PF38 or LF38
APSYSN	CHARACTER	8	0	System Name (Original System if File=DDM)
APASP	PACKED	2	0	ASP Auxiliary Storage Pool ID: 1=System ASP
APRES	CHARACTER	4	0	Reserved
APMANT	CHARACTER	1	0	Maintenance: I=*IMMED, R=*REBLD, D=*DLY
APUNIQ	CHARACTER	1	0	UNIQUE Keys must be Unique: N=No, Y=Yes
APKEYO	CHARACTER	1	0	Key Order: L=LIFO, F=FIFO, C=FCFO, N=None
APSELO	CHARACTER	1	0	S/O Select/Omit file: N=No, Y=Yes
APACCP	CHARACTER	1	0	Access Path: A=Arrival, K=Keyed, S=Shared
APNSCO	PACKED	2	0	Number of Files Accessed by Logical Files
APBOF	CHARACTER	10	0	Physical File
APBOL	CHARACTER	10	0	Library
APBOLF	CHARACTER	10	0	Logical File Format through which data is accessed

Table 31. ANDATDBK (Part 2 of 2)

Field	Type	Length	Dec.	Domain
APNKYF	PACKED	2	0	Number of Key Fields per Format
APKEYF	CHARACTER	10	0	Key Field Name
APKSEQ	PACKED	1	0	DESCEND Key Sequence: D=Descending, A=Ascending
APKSIN	PACKED	1	0	Key Sign Specified: N=Unsigned, S=Signed, A=AbsVal
APKZD	CHARACTER	1	0	Zone/digit specified: N=None, Z=Zone, D=Integer
APKASQ	CHARACTER	1	0	ALTSEQ Alternative Collating Sequence: N=No, Y=Yes
APKEYN	PACKED	2	0	Key field number: 1= First key in format
APJOIN	CHARACTER	1	0	JFILE Join logical file: N=No, Y=Yes
APACPJ	CHARACTER	1	0	Access path journaled: N=No, Y=Yes
APRIKY	CHARACTER	1		Record type: P=PRIMARY U=UNIQUE N=NONE

B.3.4 ANDATDBR

Database relationships. This file contains all the information about the relationships between files. It is populated in load AS/400 database information phase.

Table 32. ANDATDBR

Field	Type	Length	Dec.	Domain
WHRTYP	CHARACTER	1	0	Type of File: P=PF, L=LF
WHRFI	CHARACTER	10	0	File
WHRLI	CHARACTER	10	0	Library
WHRMB	CHARACTER	10	0	Member
WHRRD	CHARACTER	10	0	Record Format
WHNO	ZONED	5	0	Number of Dependencies
WHDTM	CHARACTER	13	0	Retrieval date and time, Century/Date/Time
WHREFI	CHARACTER	10	0	Dependant File
WHRELI	CHARACTER	10	0	Dependant Library
WHREMB	CHARACTER	10	0	Dependant Members
WHTYPE	CHARACTER	1	0	Type of Dependency: D=Data, I=Access Path, O=Access Path Owner, V=SQL View, C= ???
WHJDIL	ZONED	2	0	Ordinal number of position in the JFILE
WHJREF	ZONED	2	0	JREF reference number
WHSYSN	CHARACTER	8	0	System Name (Original System if File=DDM)
WHCTLN	CHARACTER	10	0	Constraint Library
WHCSTN	CHARACTER	260	0	Constraint Name

B.3.5 ANDATDFR

Origin of all the year-sensitive fields that have been discovered. In this file, you can find out where the seeding of year-sensitive fields originate. It is populated in analyze program date field propagation phase.

Table 33. ANDATDFR (Part 1 of 2)

Field	Type	Length	Dec.	Domain
PGMNAME	CHARACTER	10	0	Program Name
SRCTYPE	CHARACTER	3	0	Source Type
SRCNAME	CHARACTER	10	0	Source Name
ISADATE	CHARACTER	1	0	Is it a date? Y=Yes, N=No
IOAREA	CHARACTER	30	0	IO Area Name
IOPRG	PACKED	2	0	IO Area Program
IOSCP	PACKED	2	0	IO Area Scope
DTFNAME	CHARACTER	30	0	Propagated Date Field
DTFSCP	PACKED	2	0	Propagated Date Scope
DTFDISP	PACKED	3	0	Propagated Date Displacement
DTFPOS	PACKED	3	0	Propagated Date Position
DTFTYPE	PACKED	2	0	Propagated Date Type
DTYLEN	ZONED	1	0	Propagated Year Length
DTYOFF	PACKED	3	0	Propagated Year Offset
WINDISP	PACKED	3	0	Propagated Memory Window Displacement
WINLEN	PACKED	3	0	Propagated Memory Window Length
WINDISPBAT	PACKED	3	0	Seed Memory Window Displacement
SRCTYPEBAT	CHARACTER	3	0	Seed Source Type
SRCNAMEBAT	CHARACTER	10	0	Seed Source Name
IOAREABAT	CHARACTER	30	0	Seed IO Area Name
IOPRGBAT	PACKED	2	0	Seed IO Area Progressive
IOSCPBAT	PACKED	2	0	Seed IO Area Scope
DTFNAMEBAT	CHARACTER	30	0	Seed Date Field
DTFSCPBAT	PACKED	2	0	Seed Date Scope
DTFDISPBAT	PACKED	3	0	Seed Date Displacement
DTFPOSBAT	PACKED	3	0	Seed Date Position
DTFTYPEBAT	PACKED	2	0	Seed Date Type (how the field was discovered)

Table 34. ANDATDFR (Part 2 of 2)

Field	Type	Length	Dec.	Domain
DTYLENBAT	ZONED	1	0	Seed Year Length
DTYOFFBAT	PACKED	3	0	Seed Year Offset
STEPNBR	PACKED	3	0	Distance from Seed
ORITYPE	CHARACTER	30	0	Seed Type
FLPROPAG	CHARACTER	1	0	Propagation Flag: W/Y/M/E=Yes, C/D=No

B.3.6 ANDATDNA

Table 35 tracks the I/O areas related to date fields. It is populated in analyze program date field propagation phase.

Table 35. ANDATDNA

Field	Type	Length	Dec.	Domain
PGMNAME	CHARACTER	10	0	Program Name
IOAREANAME	CHARACTER	30	0	IO Area Name
IOAREAPRG	PACKED	2	0	IO Area Program
IOAREASCP	PACKED	2	0	IO Area Scope
IMAGESTART	PACKED	3	0	Image Start
IMAGELEN	PACKED	3	0	Image Length
IMAGEDNA	CHARACTER	256	0	Image DNA (not visible)
IMAGEPRG	PACKED	2	0	Image Program
EMPTYREC	PACKED	2	0	Empty Records

B.3.7 ANDATPTR

Table 36 tracks the BYPASS2000 propagation trace. This file lists all the from and to fields affected by propagation. It is populated in analyze program date field propagation phase.

Table 36. ANDATPTR

Field	Type	Length	Dec.	Domain
PGMNAME	CHARACTER	10	0	Program Name
FROMIOAREA	CHARACTER	30	0	From IO Area Name
FROMIOPRG	PACKED	2	0	From IO Area Program
FROMIOSCP	PACKED	2	0	From IO Area Scope
FROMNAME	CHARACTER	30	0	From Field Name
FROMFLDSCP	PACKED	2	0	From Field Scope
FROMDISPL	PACKED	3	0	From Field Displacement
FROMFLDLEN	PACKED	3	0	From Field Length
FROMDTFLD	CHARACTER	30	0	From Data Field
FROMDTSCP	PACKED	2	0	From Data Scope
FROMDTDISP	PACKED	3	0	From Data Displacement
FROMDTPOS	PACKED	3	0	From Data Position
FROMDTLEN	PACKED	3	0	From Data Length
FROMDTTYPE	PACKED	2	0	From Data Type
FROMDTYLEN	ZONED	1	0	From Year Length
FROMDTYOFF	PACKED	3	0	From Year Offset
TOIOAREA	CHARACTER	30	0	To IO Area Name
TOIOPRG	PACKED	2	0	To IO Area Program
TOIOSCP	PACKED	2	0	To IO Area Scope
TONAME	CHARACTER	30	0	To Field Name
TOFLDSCP	PACKED	2	0	To Field Scope
TODISPL	PACKED	3	0	To Field Displacement
TOFLDLEN	PACKED	3	0	To Field Length
TODTFLD	CHARACTER	30	0	To Data Field
TODTSCP	PACKED	2	0	To Data Scope
TODTDISP	PACKED	3	0	To Data Displacement
TODTPOS	PACKED	3	0	To Data Position
TODTLEN	PACKED	3	0	To Data Length
TODTTYPE	PACKED	2	0	To Data type
TODTYLEN	ZONED	1	0	To Data Year Length
TODTYOFF	PACKED	3	0	To Year Offset
FLPROPAG CHARACTER 1 0 Propag. Flag; Y=yes, I=incongruences, B=seeded	CHARACTER	1	0	Propagation Flag: Y=Yes, I=Incongruences, B=Seeded

For problem determination purposes, you may require more detailed information on some other files from the BYPASS2000 conversion repository. You can obtain this file from the Web site at: http://www.redbooks.ibm.com/

From the navigation bar on the left, click **Additional Materials**. Follow the instructions on the page that appears to access the list of available down loads. In the list of folders on the right, click **SG242156**.

From there, you can download the following three files:

- **README.TXT** — This file contains the instructions for loading the save file on to the AS/400 system.
- **BPASSQRY.SAV** — This is a save file containing some BYPASS2000 queries to help you view the information in your conversion repository tables.
- **REPFILES.LWP** — This Lotus WordPro document contains the details of all the BYPASS2000 repository files.

After you have restored the library from the save file, add xxxxDB (your conversion database library) to your library list. You can access the queries through menus using the following commands:

- Type `GO BPENVI` for the Environment Reports.
- Type `GO BPANAL` for the Analysis Reports.
- Type `GO BPSEED` for the Seeding Reports.
- Type `GO BPPROP` for the Propagation Reports.
- Type `GO BPCONV` for the Conversion Reports.
- Type `GO BPREPORT` for access to all the preceding menus.

Appendix C. BYPASS2000 Markers

The following markers can be found in file HSDATPAR in library xxxxDB and are inserted in the converted source code to indicate where BYPASS2000 has made modifications.

Table 37. BYPASS2000 Markers

RPG Marker	COBOL Marker	Description
B2???	BP2???	Identifies instructions that cannot be converted by BYPASS2000 and marks the existence of an incongruence between the two areas involved in the instruction. For example, one instruction can contain a date field whereas another does not contain a date field due to previously discovered limitations.
B2000	BP2000	Marks an instruction that involves areas or date fields but does not need to be changed. The lengthening of the year-related information in both data fields involved is sufficient to handle century data. This marker can also refer to instructions that involve figurative constants.
B2CHK	BP2CHK	Marks potential conceptual problems in the conversion of the instruction. BYPASS2000 has determined that the instruction involves date fields. You must check the logic of the converted code.
B2CPY	BP2CPY	Marks the point where copies have been inserted in the program. The copybooks contain the instructions for adding/removing the century and the relevant work areas.
B2MOD	BP2MOD	Marks instructions that modify the data definition.
B2NEW	BP2NEW	Marks lines of codes that have been added for adding/removing the century information.
B2OLD	BP2OLD	Marks the lines of code that have been commented out and replaced by new code to add/remove century information.
B2REM	BP2REM	Marks comments that signal the beginning and end of new lines of code and information for highlighted conversion problems.
B2TRC	BP2TRC	Marks instructions in which normal truncation of the numeric field type is used to remove century information.
B2VFY	BP2VFY	Marks instructions that BYPASS2000 has changed based on parameters shown in BPPARM or based on the type of operation. You must verify these instructions.
B2INF	BP2INF	Marks information for the user (for example, after a B2CHK).
B2LEA	BP2LEA	Marks an instruction that involves a leap year calculation.
	BP2SHF	Marks lines of code that have been added for shifting century information.
	BP2OUT	Marks lines of code in which a year-sensitive field is right-truncated in a non-year sensitive field.

Appendix D. BYPASS2000 Messages

The following messages can be found in either Check Request Information or the conversion log.

Table 38. BYPASS2000 Error Messages (APG0002 - BAP2004)

Message ID	Sev.	Message Text	Definition
APG0002	20	Copy &1 not found in the scheduling file.	A program calls a copy not included in the scheduling file; the user has to insert the copy source in the file QCPYSRC.
APG0026	20	The requested program &1 has not been scheduled for analysis.	The analysis of a program has been attempted (more than once) without deleting the previous analysis.
BAP2001	20	File & record not found for internal name &1 on key-field &2.	This message is displayed after the analysis when BYPASS2000 finds a statement containing a file and a key but it cannot find the corresponding file and record in its libraries.
BAP2002	20	External key not found for file &1 rec &2 seq &3 on key-field &4.	This message is displayed after analysis. BYPASS2000 is unable to find the correct link between file key fields and external keys, for example, a difference between file key number and external key number. The user should check these keys.
BAP2003	20	Internal field-name not found for key-field &1 file &2 rec &3.	It is displayed after analysis phase when a logical file contains a statement CONCAT and the key field does not belong to the list of fields of the logical file but is included in the CONCAT operation code. BYPASS2000 is unable to create a relationship between the program key fields and the file key fields. The solution to avoid loss of propagation is to seed the program's date key fields referred to KLIST.
BAP2004	20	Missing internal key-field name assumed as external one &1 in file &2 rec &3. Possible dates will not be propagated.	It is displayed after analysis phase when a logical file contains a CONCAT statement and the key field does not belong to the list of fields of the logical file but is included in the CONCAT operation code. BYPASS2000 is unable to create a link between the program key field and file key field. The solution (to prevent any propagation loss) is to seed the programs date key field referred to FACTOR-1.

Table 39. BYPASS2000 Error Messages (BAP2005 - BPA0001)

Message ID	Sev.	Message Text	Definition
BAP2005	20	Substring in CVTDAT parameters not handled.	This message is found in the display conversion log after the analysis of a CLP program which contains a statement with a sub-string in CVTDAT. BYPASS2000 is unable to handle this kind of statement.
BAP3001	20	Warning! Found complex &1 expression at statement &2. Handling ended after first element.	This message is displayed after the analysis phase. It warns the user that BYPASS2000 has found a complex expression, for example, a CHGVAR in a CLP program. BYPASS2000 is unable to handle this type of complex expression.
BCA0001	20	The requested copy &1 has not been scheduled for analysis.	The user has performed (more than once) the analysis of a copy without deleting the previous analysis.
BFC0002	20	File &1 already scheduled for routines generation.	The user has performed (more than once) the creation of migration routine of a file without setting the Migration Flag to <0>.
BFP0001	20	Field not defined: program &1, I/O area name &2, I/O area prg. &3, field disp. &4.	Check the definition of that field; it may be in a file or copy that is not present or not analyzed.
BFP0002	20	More than 1024 date fields found for &1 file &2. Not handled.	BYPASS2000 has found a file where the user has seeded more than 1024 date fields. BYPASS2000 cannot handle this situation.
BFP0004	20	I/O area name not found for file &2 in program &1.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BFP0005	20	No fields found containing the following date field: source &1, I/O area name &2, prg. &3, field displ. &4, length &5.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BFP0007	20	More than one record format found for &4 file &3. Unable to load date fields for I/O area &2 of program &1.	This is a multi-format file. If some fields inside one format are year-sensitive, they need to be seeded.
BFP0009	20	More than one logical area for I/O area &2 of program &1. Unable to load date fields for &3 file &4.	This program contains one or more multi-format files. Declare it in a copy, assign I/O area to related file, and assign record type for related I/O area.
BPA0001	20	More than 32767 statements found for source &1. Not handled.	This message is displayed when the user tries to call a source program (with option I or O in the inventory file) that has more than 32767 statements.

Table 40. BYPASS2000 Error Messages (BPA0002 - BPA0037)

Message ID	Sev.	Message Text	Definition
BPA0002	20	Field not defined: source type &1, source name &2, I/O area name &3, field name &4.	Check the definition of that field; it may be in a file or in a copy that is not present or analyzed.
BPA0012	20	Unknown structure instruction on statement &1, instruction &2.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0015	20	More than one field &1 found in program &2.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0020	20	Some problem found with the displacement of the field &1.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0021	20	More than 256 different prg. found for field &2 in I/O area &1.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0022	10	Unexpected SQL ERROR in program: &1.	System error
BPA0023	20	Maximum number of IO-Area progressive reached for IO-Area: &1. Splitting skipped.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0024	20	Field not defined: source type &1, source name &2, I/O area number &3, field number &4.	Check the definition of that field; it may be that the field is in a file or copy that is not present or analyzed.
BPA0032	20	No field definition found in &1 &2.	In this program, there are no field definitions. This message is for information only.
BPA0036	20	Dynamic linking found. Use "Work with dynamic call" to assign values.	For "Additional user information" at memory level, you need to specify which are the called programs and their parameters.
BPA0037	20	Program analysis ended because of some message on user inquiry. Use the "Check Requested Information" function and run analysis again.	There are messages in "Check Requested Information" for this program. These messages need to be checked and the phase deleted and re-run to continue.

Table 41. BYPASS2000 Error Messages (BPA0039 - BPC0007)

Message ID	Sev.	Message Text	Definition
BPA0039	20	Duplicated key for I/O area nbr &1 field nbr &2.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0040	20	Unknown propagation effects for: Source Type: &1, Source Name: &2, Statement &3, Instruction &4.	The converted code could have some problems. If there are date fields involved in this instruction, seed them.
BPA0047	20	Source &2 (&1) not found in the scheduling file.	This is a message that appears after the analysis phase. It is a warning that BYPASS2000 has found a COPY that has not been loaded in the inventory file. The user needs to load it in the inventory file and analyze it.
BPA0048	20	Duplicate key: source name &2 (&1), I/O area nbr &3, I/O area prg &4, field nbr &5.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0049	20	Field not defined: source name &2 (&1), I/O area name &3, prg &4, field name &5, displ &6.	Check the definition for the field; it could be contained in a file or copy that is not present or analyzed.
BPA0051	20	Duplicate key: source &1 (&2), I/O area nbr &3, field nbr &4.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0052	20	Duplicate key: source name &2 (&1), I/O area name &3, I/O area scope &4.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPA0450	20	File &1 in library &2 ignored. Previously analyzed in library &3 with the same name.	The same file is found in different libraries. BYPASS2000 will not allow this situation. Put the file in only one library.
BPC0005	10	An error occurs in retrieve routine name for file &1 &2.	This message appears when BYPASS2000 tries to create DIM, MIGR, or TST routines but it is unable to find the routine. The user should check the file HSDATFRN.
BPC0007	20	No copy relation for FILE &1 &2.	This message appears when BYPASS2000 tries to create DIM, MIGR, or TST routines but is unable to find a link between the file and copy. The user should check the file HSDATFIL.

Table 42. BYPASS2000 Error Messages (BPC0010 - BPP0002)

Message ID	Sev.	Message Text	Definition
BPC0010	20	Found problems processing statement &1, instruction &2. Conversion might be unreliable.	This message appears when BYPASS2000 finds an SQL statement, for example, a statement in a program where a cursor does not correspond to a FETCH statement.
BPC0410	20	Unable to retrieve source output information for routine &1, type &2, language &3.	This is a conversion problem. It could be that a DDS is not present in the file HSDATROS - this needs to be checked.
BPC0411	20	Unable to retrieve source member for file &1.	It could be there is no source member for this file, so it needs to be loaded and analyzed. Or, it could be that the name of the object is different from the source so you need to assign object/source relation in 'Set conversion environment'.
BPC0412	20	Unable to retrieve source input library for file &1.	It could be that there is an error in compiling source input libraries in "Set Conversion Environment" - this needs checking.
BPC0413	0	Unexpected error copying statement &1.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPC0414	0	Variable &1 not found.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPC0415	0	Enlarged temporary field not found.	This appears when BYPASS2000 finds a lot of IF, OR statements, and it does not have enough temporary fields to use. The user should check the program.
BPM2001	20	Source Library not found file's OVERRIDE executed in *LIBL. Check the library list.	This message appears after the creation of MIGR or TST dispatcher. BYPASS2000 cannot find the name of the source library in HSDATLIB. Add the name of the source library.
BPM2002	20	New Object Library not found. File's OVERRIDE executed in *LIBL. Check the library list.	This message appears after the creation of MIGR or TST dispatcher. BYPASS2000 cannot find the name of the New Objects library in HSDATLIB. Add the name of the source library.
BPP0001	20	Field not defined: pgm &1, I/O area &2, I/O area prg. &3, field &4, field displ. &5, field scope &6.	Check the definition of that field. It could be that it is defined in a file or in a copy that is not present or analyzed.
BPP0002	20	The requested program &1 has not been analyzed.	It is necessary to run the analysis before the propagation.

Table 43. BYPASS2000 Error Messages (BPP0003 - BPP0015)

Message ID	Sev.	Message Text	Definition
BPP0003	20	The requested program &1 has not been scheduled for date field propagation analysis.	It is necessary to run the analysis before the propagation.
BPP0004	20	Field found more than once: pgm &1, I/O area &2, I/O area prg. &3, field &4, field displ. &5, field scope &6.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPP0005	30	Duplicate key: pgm &1, I/O area &2, I/O area prg. &3, field &4, field displ. &5, field scope &6.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPP0007	20	Invalid internal or external parameter: type &1, caller &2, called &3, parmprg &4.	This message appears after the propagation when BYPASS2000 has found an incongruence between parameters passed from caller and called programs. The user should check the program code.
BPP0008	20	Conversion problem between &1 and &2 variables.	This message is displayed after a program conversion when BYPASS2000 finds two variables with attributes that are not compatible.
BPP0010	20	Field not propagated: type &1, caller &2, called &3, parmprg &4, I/O area &5, field &6.	The caller program called more than one program and so the date field information on parameters passed is not propagated. If one or more parameters are date fields, it is necessary to seed them in called programs.
BPP0011	20	The requested program &1 has not been scheduled for conversion. Date field propagation not found.	It is compulsory to run the propagation throughout to status <9> to do the conversion.
BPP0012	20	Field related with a non-propagated field IOAREA: &1, IOAREAPRG: &2, FLDNAME &3.	The field is related with a field seeded with "Propagation with Confirm". If it is a year-sensitive field, it needs to be seeded.
BPP0013	20	Propagation tree too deep.	The propagation tree has too many levels, the default is 10. After the 10th level, there are other dates. More seeding is required. The message shows the last field in the tree.
BPP0014	20	Program &1 ran for &2 sec.	This message is displayed after the propagation of a program. It is a warning that the propagation of this program has reached the maximum elapsed time for a single program propagation.
BPP0015	20	Propagation tree too big.	The propagation tree is too large. More seeding is required. The message shows the last field used.

Table 44. BYPASS2000 Error Messages (BPP0016 - BPP0034)

Message ID	Sev.	Message Text	Definition
BPP0016	20	Elementary field not found in I-O area &2 prg. &3 scope &4 in position &5-&6 (Caller pgm &1).	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPP0017	20	Program ended because of some unanswered message on user inquiry.	There are messages in "Check Requested Information" for this program. These messages need to be checked, the phase deleted, and rerun to continue.
BPP0019	20	Field not defined: Pgm. &1, Field &2.	This appears after the propagation or conversion of pgm &1 when BYPASS2000 cannot find where &2 field is defined. The most likely cause of this message is a file not available in the old object library, or a file is not analyzed.
BPP0020	20	Field found more than once: Pgm. &1, Field &2.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPP0021	20	Copy not propagated because of too many date fields.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.
BPP0022	20	Field not defined: Copy &1, Field &2.	Check the definition of the field.
BPP0023	20	Record type for &1 file &2 not assigned. Program date field propagation analysis cannot be run.	When a multi-format file is defined, it is compulsory to assign the record type before running the propagation.
BPP0028	20	Elapsed time of &1 phase was &2 sec. The processing was stopped.	The available elapsed time is ended before the phase is terminated. Delete the phase and rerun.
BPP0031	20	Data type &1 invalid for field &2 of I/O area &3.	Find the right data type and remember that it is not possible to sub-string packed fields.
BPP0032	20	Year length &1 invalid for field &2 of I/O area &3.	Set the correct year length.
BPP0033	20	Field not found: I/O area &2, parent name &3, field &1.	No definition for fields are found; this could be a problem so it needs to be checked.
BPP0034	20	Field found more than once: I/O area &2, parent name &3, field &1.	This message should be removed from the CVRLOG because it is an internal check on BYPASS2000 programs. It does not require any user intervention and can be ignored.

Table 45. BYPASS2000 Error Messages (BPU0001 - CVR2016)

Message ID	Sev.	Message Text	Definition
BPU0001	20	File &1 not available.	This message is displayed after the analysis when a program contains a file or a DS externally described that is not available in the old object library.
CCY0001	20	The requested copy &1 has not been scheduled for conversion.	The user has performed (more than once) the conversion of a file or a copy when the conversion flag is not < 0 > .
CCY0004	20	The requested copy &1 has not been scheduled for conversion. Analysis not found.	The user tried to convert a copy that has not been analyzed yet. Analyze the copy; then run conversion again.
CPG0001	20	The requested program &1 has not been scheduled for conversion.	The propagation phase must reach <9> before running the conversion. It is likely that the user has run the conversion of this program before the end of the propagation phase.
CVR0001	0	Field &1 is not a date or year.	This is a warning message. During the conversion, BYPASSP2000 has not modified the previous statement because this field is not year-sensitive. The user should check the converted code.
CVR0003	0	Assumed as extraction of part (not containing year) of a date-field.	This is a warning message. BYPASS2000 has discovered, during the propagation, that the field involved in the statement is not a year so it has not modified the code during the conversion.
CVR0004	0	Assumed as update of a date-field in its not-containing-year zone.	This message appears in the converted code of a program. It is a warning that the field in the statement does not contain a year.
CVR2001	20	Converted code may contain conceptual errors, check code.	BYPASS2000 has done the conversion of the statement but the user is required to check if the statement was correctly converted due to the presence of a line of code that can cause conceptual errors (IF, COMP, ext).
CVR2015	20	Field &1 has been enlarged. Check output for possible and incorrect overlapping or shifting of fields.	A field that is present in a specific O has been enlarged. It is possible that this field overlaps another field with known effects. The user should check the output carefully.
CVR2016	20	This statement involves year-sensitive fields already enlarged or preceded by century. The century usage could not be determined.	This is a warning message; BYPASS2000 has converted the preceding statement but it is not sure that this type of conversion is correct because it does not follow the logic of the program.

Table 46. BYPASS2000 Error Messages (CVR3005 - CVR5007)

Message ID	Sev.	Message Text	Definition
CVR3005	30	Converted code may not work properly; check code.	This is a warning message displayed in the converted code usually after a ADD/REM routine or a SHIFT routine when the date type of the fields involved in the statement are different. BYPASS2000 warns the user to check the converted code.
CVR3009	30	&1 file &2 must be converted because field &3 is shared by name with other database fields that need to be expanded.	Field &3 will be enlarged because it is in DB. If this field does not need to be enlarged, change its name with a name that is not shared with DB.
CVR3010	30	&1 file &2 must be converted because field &3 (and others) are shared by name with other database fields that need to be expanded.	Field &3 (and others) will be enlarged because they are in DB. If they do not need to be enlarged, change their names with other names not shared with DB.
CVR3012	30	Field &1 has more than one year inside.	BYPASS2000 is not able to convert this statement because the field or the data structure is too large (more than 10 digit) or because during the propagation, it has discovered that the field contains a year in more than one position.
CVR3013	30	Field &1 contains a year but does not have a standard date-type.	BYPASS2000 converter inserts this message in the converted code of a program when it finds a field that contains a non-standard date type (the user can find a standard date type in "Set conversion environment", then "Work with Field Type").
CVR3016	30	Found more than one format with a field originally ending at position &2 in file &1. Please check.	This message is displayed in the converted code when BYPASS2000 has found more than one field that ends at a particular position. Generally, this happens because BYPASS2000 has enlarged some fields included in an O-Spec shifting all the others. The user must check that the conversion works.
CVR5002	50	Mismatch between date-field type of operand.	This is inserted in the converted code when an incongruence has not been solved. The user should find the cause of this incongruence (possibly wrong seeding or propagation problems) to establish if the converted code is correct.
CVR5006	50	The converter tool cannot handle add/remove century between fields &1 and &2 for key-field usage.	This is a BYPASS2000 conversion limitation; manual handling is required for this line of code.
CVR5007	50	The converter tool cannot handle add/remove century to adjust comparison of &1 with &2 into a DO loop.	This is a BYPASS2000 conversion limitation; manual handling is required for this line of code.

Table 47. BYPASS2000 Error Messages (CVR5008 - HMI0034)

Message ID	Sev.	Message Text	Definition
CVR5008	50	The converter tool cannot handle op-code &1 involving not expanded and expanded date fields requiring add/remove century.	This is a BYPASS2000 conversion limitation; manual handling is required for this line of code.
CVR5011	50	Field &1 is generic.	Field &1 receives more than one date format. Please check and seed.
CVR5014	50	Field &1 has been enlarged so EDTCDE and EDTWRD are no longer valid.	Informational Message
CVR5015	50	Field &1 has been enlarged while file area &2 remains unchanged.	A field seeded or propagated as year-sensitive to be expanded is present in an O-spec in a position corresponding to an area not expanded by BYPASS2000 because on the file of output in that position, there is a field seeded as date not to be expanded. The user must check the file because BYPASS2000 cannot remove the century from an output statement.
CVR5016	50	File area &1 has been enlarged while field &2 remains unchanged.	A field seeded or propagated as year-sensitive not to be expanded is present in an O-spec in a position corresponding to an area expanded by BYPASS2000 because on the file of output in that position, there is a field seeded as date to be expanded. The user must check the file because BYPASS2000 cannot remove the century from an output statement.
CVR5017	50	The conversion tool cannot handle constant values being added to or subtracted from an expanded date field.	This is a BYPASS2000 conversion limitation; manual handling is necessary for this line of code.
CVR5018	50	The array has been enlarged but the converter tool cannot change values of compile-time arrays at the end of the program.	The converter has found a compile-time array seeded by the user or propagated as standard date type and has enlarged it but advises the user to manually modify the values that the array assumes during compile time.
HDS0001	20	Internal error in routine &1.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
HDS0002	20	Error during creation of USER INDEX &1.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
HDS0003	20	Wrong LOG logging parameters: name of affected program not specified.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
HMI0034	20	The 1st word of string &3 of source &2 is longer than expected.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.

Table 48. BYPASS2000 Error Messages (HMI0038 - SEC0002)

Message ID	Sev.	Message Text	Definition
HMI0038	20	No word could be found in string &3 of source &2.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
HMI0040	20	Name of copy &2 not found within string &3 of program &1.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
HMI0051	20	Extension &4 not applicable to string &3 in program &2.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
HMI0073	20	Due to a lack of space, string &2 in source &1 could not be modified.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
HMI0074	20	Word &3 could not be found in string &2 of source &1.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
NPG0004	0	Line of pgm &1 has been truncated in wrong type: &2.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
OBJ0001	20	Library &1 not found.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
OBJ0007	20	Library &1 specified more times.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
OBJ0012	20	Library &1 is not valid.	This is a rare error message. It implies unforeseen problems. You should contact IBM support.
SEC0002	30	Authorization violation.	This message appears when BYPASS2000 finds a discrepancy between the AS/400 serial number where it has been installed and the AS/400 serial number that the user has conveyed to IBM when the enabling code has been required. You also see this message if you have failed to enter your software key (BPPR). You can obtain your license key by faxing HAL your AS/400 serial number.

Appendix E. IBM SmoothStart Service for BYPASS2000

IBM SmoothStart services are installation services designed to help accelerate the productive use of your IBM solution. You can use these services for IBM or non-IBM hardware and software, or just software. When you leverage SmoothStart for your business, you are getting a more complete solution and freeing up your valuable time to spend on revenue-generating activities.

SmoothStart delivers:

- Project management
- Installation planning
- Software installation
- Software configuration
- SmoothStart installation record
- Skills transfer
- Operational customization (optional)

Each time you engage IBM to perform SmoothStart Services, these services can be delivered to you at a very competitive price. If you have special needs, we can work with you to address those requirements in a customized solution.

Our SmoothStart Services advance the commitment by IBM to provide you with need-driven, flexible support for your information systems to help you reach your business goals.

E.1 Customized Installation Service

IBM SmoothStart Service for BYPASS2000 for AS/400 is your quick start in preparing your applications for the next millennium. An IBM services specialist assists with the installation of BYPASS2000. This is a tool that converts AS/400 application dates to handle the year 2000, requiring a minimal amount of manual intervention. When you contract for this on-site service, you designate one or two staff representatives to work approximately forty hours with a highly trained IBM services specialist. This specialist assists and prepares your staff to start the BYPASS2000 conversion on your AS/400 system.

After installation, the BYPASS2000 tool analyzes and converts your specific application that uses dates in its programs, updating all two-digit year fields to handle a four-digit format.

Note: BYPASS2000 for AS/400 (5697-C20) is a prerequisite for this service.

E.2 For More Information

For more information about the IBM SmoothStart Service for BYPASS2000 for AS/400, please visit the Web site at:
http://www.as.ibm.com/

Then, select your country.

Appendix F. AS/400 BYPASS2000 Education

The following course is available in the U.S.A. It may not be available in all IBM centers worldwide. Contact your local IBM Sales Office for more details on IBM Education offerings.

Or, you can start on the Web site at: http://www-3.ibm.com/services/learning/

Choose your country, and then search for BYPASS2000 to find AS/400 BYPASS2000 Workshop [S6107].

Appendix G. Special Notices

This publication is intended to help AS/400 applications use IBM Year 2000 products. The information in this publication is not intended as the specification of any programming interfaces that are provided by BYPASS2000 V3R1M2 (Program 5697-D11). See the PUBLICATIONS section of the IBM Programming Announcement for BYPASS2000 V3R1M2 (Program 5697-D11) for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AFP	AS/400
AT	COBOL/400
CT	DB2
DRDA	IBM ®
OS/400	RMF
RPG/400	RS/6000
SmoothStart	SP
System/38	System/390
XT	400

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and/or other countries licensed exclusively through X/Open Company Limited.

SET and SET logo are trademarks owned by SET Secure Electronics Transaction LLC

Other company, product, and service names may be trademarks or service marks of others.

Appendix H. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

H.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see “How to Get ITSO Redbooks” on page 187.

- *AS/400 Applications: Moving to the 21st Century*, SG24-4790
- *AS/400 Applications: Year 2000 Enablement & Services Considerations,* SG24-4829
- *Make Your AS/400 System Year 2000 Ready*, SG24-5401

H.2 Redbooks on CD-ROMs

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at `http://www.redbooks.ibm.com/` for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	Collection Kit Number
System/390 Redbooks Collection	SK2T-2177
Networking and Systems Management Redbooks Collection	SK2T-6022
Transaction Processing and Data Management Redbooks Collection	SK2T-8038
Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
AS/400 Redbooks Collection	SK2T-2849
Netfinity Hardware and Software Redbooks Collection	SK2T-8046
RS/6000 Redbooks Collection (BkMgr Format)	SK2T-8040
RS/6000 Redbooks Collection (PDF Format)	SK2T-8043
Application Development Redbooks Collection	SK2T-8037

H.3 Other Publications

These publications are also relevant as further information sources:

- *AS/400 Magazine - A User Publication, March 1997 Issue*, G325-6310
- *BYPASS2000 Licensed Program Specifications Version 3 Release 1 Modification Level 2*, GC09-2592
- *BYPASS2000 User's Guide Version 3 Release 1 Modification Level 1*, SC09-2591
- *Program Directory for IBM BYPASS2000 for AS/400 Version 3 Release 1 Modification Level 2*, GI10-0406 (ftp://service.software.ibm.com/as400/products/ad/bypass/bppd2.pdf)
- *The Year 2000 and 2-Digit Dates: A Guide for Planning and Implementation*, GC28-1251
- *Year 2000 General Information Brochure*, GC26-9002
- *Year 2000-Ready Solutions from AS/400*, G221-8005

H.4 Web Sites

- **HAL's Site**

  http://www.halinfo.it

- **AS/400 Year 2000 Support**

  http://www.as400.ibm.com/developer/year2000/index.html

- **IBM Year 2000 Site**

  http://www.software.ibm.com/year2000

- **IBM Bypass2000 Site**

  http://www.software.ibm.com/ad/as400/bypass/

- **IBM AS/400 Partners in Development (PID) Site**

  http://www.as400.ibm.com/developer

- **IBM ITSO Redbooks Site**

  http://www.ibm.com/redbooks

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** http://www.redbooks.ibm.com/

  Search for, view, download or order hardcopy/CD-ROM redbooks from the redbooks web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this redbooks site.

  Redpieces are redbooks in progress; not all redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

  Send orders via e-mail including information from the redbooks fax order form to:

	e-mail address
In United States	usib6fpl@ibmmail.com
Outside North America	Contact information is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl/

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	Country coordinator phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl/

- **Fax Orders**

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	Fax phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl/

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and redbooks by accessing the IBM Intranet Web site at http://w3.itso.ibm.com/ and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at http://inews.ibm.com/ to view redbook. residency, and workshop announcements.

IBM Redbook Fax Order Form

Please send me the following:

Title	Order Number	Quantity

First name	Last name	
Company		
Address		
City	Postal code	Country
Telephone number	Telefax number	VAT number

☐ Invoice to customer number

☐ Credit card number

Credit card expiration date	Card issued to	Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Index

M

N

O

P

Q

R

S

T

U

V

W

Y

ITSO Redbook Evaluation

AS/400 Applications: IBM Year 2000 Tools Tips and Techniques
SG24-2156-01

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at the address: redbook@us.ibm.com
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Which of the following best describes you?
_ Customer _ Business Partner _ Solution Developer _ IBM employee
_ None of the above

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes___ No___

If no, please explain:

What other redbooks would you like to see published?

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)

SG24-2156-01

Printed in the U.S.A.

AS/400 Applications: IBM Year 2000 Tools Tips and Techniques

SG24-2156-01