Communicator 3000 MPE/iX Release 6.0 (Platform Software Release C.60.00)

HP 3000 MPE/iX Computer Systems

Volume 10 Issue 1

Part Number 32650-90269

HEWLETT ®
PACKARD

Notice

The information contained in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for direct, indirect, special, incidental or consequential damages in connection with the furnishing or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

This document contains proprietary information which is protected by copyright. All rights reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

Trademark Notice

UNIX is a registered trademark of The Open Group.

Contents

Contents

Contents

12. Catalog of User Documentation

Contents

Tables

Tables

Obtaining Software Security Patches for your HP Computer System

Hewlett-Packard would like to make you aware of a special free service provided for all customers of HP 3000 and HP 9000 computer systems. This service gives customers a direct route to Hewlett-Packard for obtaining information relating to the security of their Hewlett-Packard Computer System(s).

Hewlett-Packard issues information on the availability of software security patches via Security Bulletins to subscribers of the HP Security Bulletin Digest e-mail service, a part of the HP Electronic Support Center. A Hewlett-Packard support contract is NOT required to subscribe to this service to obtain information or security patches. Any purchaser of an HP 3000 or HP 9000 Computer System can make use of the HP Security Bulletin services at no charge.

Customers may also obtain information and Security Bulletin services via the World Wide Web.

A security problem is a software defect that allows unauthorized personnel to gain access to a Computer System or to circumvent any of the mechanisms that protect the confidentiality, integrity or availability of the information stored on the system. When such problems in Hewlett-Packard software are brought to the attention of the company, their resolution is given a very high priority. This resolution is usually in the form of a Security Bulletin which may explain how to correct the problem or describe how to obtain a software security patch that will correct the problem.

Hewlett-Packard has introduced this service as the primary mechanism to alert subscribers to security problems and provide corrections. Hewlett-Packard will not analyze the relevance of any security patch to any individual customer site within the scope of the HP Security Bulletin service. The responsibility for obtaining and applying security patches resides with the customer.

The remainder of this letter outlines the various security related services offered by HP Electronic Support Center and the methods for subscribing to and retrieving information from it. It also outlines how you can inform Hewlett-Packard of potential security concerns you may have with your Hewlett-Packard Computer System.

HP Electronic Support Center Security-Related Services

HP Electronic Support Center offers subscribers the following benefits:

• Receive Security Bulletins automatically when they are published.

• Retrieve the archive list of bulletins issued prior to subscription.

• Download security patches if the subscriber configuration supports it.

Remember, an HP support contract is not required to subscribe to HP Security Bulletin services.

Subscribing to HP Electronic Support Center Security Bulletin Services

Once you have placed your name on the subscriber list for future Security Bulletins (see instructions below), you will receive them via e-mail on the day they are issued by HP.

As referenced below, you can also view a list of past Security Bulletins issued in the "HP Security Bulletins Archive".

Instructions

To subscribe to automatically receive future NEW HP Security Bulletins from the HP Electronic Support Center via electronic mail, do the following (instructions subject to change without notice):

1. Use your browser to access the HP Electronic Support Center web page at:

   http://us-support.external.hp.com   US, Canada, Asia-Pacific, and Latin-America

   http://europe-support.external.hp.com   Europe

2. Logon with your User ID and password (or register for one). Remember to save the User ID assigned to you, and your password.

3. Once you are on the HP Electronic Support Center home page, click on "Support Information Digests". On this page, you can subscribe to many different digest services, including the Security Bulletin Digests.

To review Security Bulletins that have already been released, click on "Search Technical Knowledge Base (Security Bulletins only)" on the HP Electronic Support Center home page. Near the bottom of the next page, click on "Browse the HP Security Bulletins Archive".

Once in the archive, click on "HP-UX Security Patch Matrix" to get a patch matrix of current HP-UX and BLS security patches. Updated daily, this matrix categorizes security patches by platform/OS release, and by Security Bulletin topic.

If You Discover a Security Problem

To report new security vulnerabilities, send e-mail to

security-alert@hp.com

Please encrypt any exploit information using the security-alert PGP key, available from your local key server, or by sending a message with a -subject- (not body) of 'get key' (no quotes) to security-alert@hp.com.

1   Overview—MPE/iX Release 6.0 (C.60.00)

This *Communicator 3000* provides general and detailed information on the new and enhanced functionality for the MPE/iX 6.0 Release (C.60.00), as well as information on release strategy and installation prerequisites.

MPE/iX 6.0 will accumulate all of the functionality previously released on MPE/iX 5.5 Express releases such as Year 2000 enhancements, ODBCLink/SE—a 32-bit driver from M.B. Foster Associates, and B-Tree functionality with IMAGE/SQL. In addition, it will also offer support for user-defined job queues—for improved system management, SAMBA—for NT integration, Java, FTP enhancements, and many more enhancements.

Starting with Release 5.5, Hewlett-Packard distributes HP 3000 Release tapes in Native Mode STORE format. Your installation personnel do not need to be retrained—the installation steps are unchanged. However, since Native Mode STORE generally performs better than Compatibility Mode STORE, you will probably see some significant performance improvements during your installation.

Communicator Summary

Following are brief descriptions of the articles and chapters.

MPE/iX Patches on HP Electronic Support Center

MPE patches are now available on HP Electronic Support Center via the Internet to all customers. For more details, refer to the article, "MPE/iX Patches Now on HP Electronic Support Center," in this chapter.

HP Instant Information CD-Rom

HP Instant Information is a replacement for the HP LaserRom on-line document delivery system. For more details, refer to the article, "HP Instant Information for Delivery of Learning Products," in this chapter.

Chapter 2, System Information/Before You Install

- "Updating To and Backdating From MPE/iX 6.0" briefly describes the requirements for updating to MPE/iX 6.0 and backdating from MPE/iX 6.0.

- "Planning Your Move to MPE/iX Release 6.0" describes the requirements and procedures for updating to MPE/iX Release 6.0, including the disk space requirements.

- "Upgrading a Pre-997 Corporate Business Server to a 997" explains some configuration considerations when upgrading to a 997 server.

- "Special Note on SCSI Device Configuration for MPE/iX 6.0" explains some considerations in configuring SCSI devices.

- "New MPE/iX Obsolescence Strategy" explains Hewlett-Packard's revised plan for obsoleting operating systems.

- "HP System Account and Directory Naming Structure" adds information on reserved directory names and a list of these names.

Chapter 3, System Management

- "User Defined Job Queues" explains that MPE/iX can now have more than one job queue. It also explains the HIPRI option of ALTJOB.

- "Overview of HP Patch/iX" gives an overview of the Patch Management tool, HP Patch/iX. For a technical overview, see the technical article, "HP Patch/iX Technical Overview," in Chapter 10, "Technical Articles."

- "Overview of HP Stage/iX" describes HP Stage/iX, which is an operating system facility for applying and managing MPE/iX patches on your systems.

- "Transaction Manager Overview" gives an overview of System Log Expansion, Checkpoint Improvement, and Copy-Forward.

- "Increase in Number of TCP Connections" explains how to configure your system to allow up to 5600 connections.

- "Overview of CI Enhancements" describes the new functionality added to the CI software to improve the ease of use and overall functionality for C.55.03 and 6.0. Refer also to "CI Enhancements for the Year 2000," in the chapter, "Year 2000 Enhancements," and for more detailed information refer to "CI Enhancements in C.55.03" and "CI Enhancements in 6.0" in Chapter 10, "Technical Articles."

- "MPE/iX POSIX Installation Changes" explains the installation changes for MPE/iX POSIX.

- "DNS BIND/iX and Syslog/iX" gives an overview of DNS BIND/iX and Syslog/iX.

Chapter 4, System Support Tools

- "Predictive Support" describes the new peripherals and SPUs supported by Predictive.

- "On-Line Diagnostics System (Sherlock)" explains that the Sherlock system will add support for new systems.

- "GLANCE has a New Name and a New Home!" reports that HP GlancePlus/XL has been renamed to HP GlancePlus/iX and the program file has also been renamed. There is a Customer Trial Demonstration available to provide a one-time-only trial period for the use of the Glance performance tool.

Chapter 5, Application Development

NOTE   Additional technical information on many of these topics is in Chapter 10, "Technical Articles."

- "Transact/V and Transact/iX Enhancements" describes the new enhancements to Transact/V and Transact/iX.

- "Dictionary/3000" describes the capability of copying entities with the new COPY command, and the ability to enter elements multiple times within an HP Inform group.

- "Inform/V" describes enhancements to support a new version of Dictionary/3000.

- "HP ALLBASE/BRW" explains that this version of ALLBASE/BRW focuses on a TPI defect repair.

- "COBOL II/iX Enhancements" describes the COBOL II/iX enhancements.

- "Query/iX Enhancements" describes the enhancements made to QUERY to utilize the new B-tree index features in TurboIMAGE/XL.

- "IMAGE/SQL with TurboIMAGE/XL Enhancements" gives an overview to the highlights of this release of IMAGE/SQL and TurboIMAGE/XL including IMAGE/SQL Dynamic Dataset Expansion for Master Datasets (MDX) and reviews the previous release including B-Trees. Reviews a problem with IMAGE/SQL Dynamic Dataset Expansion for Detail Datasets (DDX).

- "Features in ALLBASE/SQL Version G3" describes the new string functions in Version G3. Reviews these major enhancements in version G2: new authorities, CAST function, ODBCLink/SE, and composite key.

Chapter 6, Data/Application Integration

- "DCE RPC Clients for FOS through MPE/iX" gives an overview of the DCE RPC Clients modules for FOS.

Chapter 7, Interoperability and Internet Solutions

NOTE   Additional technical information on many of these topics is in Chapter 10, "Technical Articles."

- "Introducing 100VG-AnyLAN Network Adapter" describes the high-speed network link that provides HP 3000 systems with a connection to a 100VG local area network.

- "Introducing 100Base-T Network Adapter" describes the high-speed network link that provides HP 3000 systems with a connection to a 100Base-T local area network.

- "Introducing the REMSH Client Functionality" describes the REMSH Client that allows a user to connect to the remote system, execute a command on that remote system, and see the results of that command.

- "Hewlett-Packard NetWorker ClientPak for MPE/iX" explains that MPE/iX Express 3 for 5.5 (or later version) or a patch is required for the Legato NetWorker ClientPak.

- "Introducing ODBCLink/SE" gives an overview of the product replacing HP PC/API for ALLBASE/SQL links.

- "Overview of the DTS/TIO Subsystem" describes the new features of the Datacommunications and Terminal Subsystem (DTS).

- "Telnet/iX Server—Full Functionality Release" describes the Telnet/iX Server enhancement on MPE/iX Release 5.5, which strengthens the HP 3000 as an open system by providing access between the HP 3000 and computers that support Telnet, such as UNIX-based systems and PCs. Describes the functionality added with Release 5.5 PowerPatch (C.55.01).

- "FTP Enhancements" gives an overview of the new FTP features.

- "Samba/iX Overview" gives an overview of Samba/iX.

- "Introducing Java/iX " gives an overview of Java.

Chapter 8, Peripherals

- "9 GB Disk Drive" is an announcement that the 9 Gigabyte (GB) disk drive is available for use with 5.5 (or later) systems; however, it cannot be used for LDEV 1, the boot disk.

- "New Disk Devices Supported" describes the new 4 GB and 9 GB disk drives that are now supported.

- "High Availability Disk Array (Nike)" describes the support for 9 GB and 18 GB disk drives.

- "A3716A 144 GB DDS-3 Autoloader for MPE/iX" gives an overview of the A3716A high-speed, high-capacity tape product combined with an autoloader.

- "EMC Symmetrix Disk Arrays" overviews the SYM4 version of the EMC Symmetrix Disk Array supported starting with Express 3 for 5.5.

Chapter 9, Year 2000 Enhancements

NOTE   This is a new chapter to feature the enhancements to products to handle the year 2000 issues.

Additional technical information is in Chapter 10, "Technical Articles."

- "Year 2000 Web Page" gives the URL for the Hewlett-Packard Y2K web page.

- "Year 2000 Enhancements for MPE/iX" gives an overview of the Y2K enhancements for MPE/iX.

- "New Date Intrinsics for MPE/iX" gives an overview of the new date intrinsics for MPE/iX.

- "VPLUS Enhancements for Turn-of-Century" describes enhancements in B.06.07 version of VPLUS for the Turn-of-century enhancement. Refer to the technical article in Chapter 10 for other VPLUS enhancements.

- "Predictive" announces that Predictive is year 2000 safe.

- "HP MPE/iX Databases" describes the readiness of ALLBASE/SQL, IMAGE/SQL, and TurboIMAGE/XL to handle the year 2000.

- "HP Stage/iX" announces that HP Stage/iX is Y2K compliant.

- "Documentation Changes for MPE/iX" provides changes to existing documents.

- "RPG/iX 4-Digit Year Enhancements" describes the support for predefined fields that return a 4-digit year.

Chapter 10, Technical Articles

- "QUERY/iX Enhancements" describes the enhancements made to QUERY to utilize the new B-tree index features in TurboIMAGE/XL.

- "COBOL II/iX Enhancements" describes the new COBOL II/iX enhancements.

- "IMAGE/SQL with TurboIMAGE/XL Enhancements" explains the highlights of this release of IMAGE/SQL and reviews these features of the previous release: B-Tree indices, and ODBCLink/SE. Additional manuals are now bundled with IMAGE/SQL. Describes the new IMAGE/SQL dynamic dataset expansion for master datasets (MDX). Describes the highlights of this release of TurboIMAGE/XL, B-Tree indices and Scalability. Reviews a problem with IMAGE/SQL dynamic dataset expansion for detail datasets (DDX).

- "New Features in ALLBASE/SQL Version G3" describes the new string functions in version G3 and reviews the major enhancements in version G2: new authorities for security and year 2000 compatibility. It gives an overview of the CAST function that was added in the G1.15 release of ALLBASE/SQL. Describes the process of moving pseudo-mapped files to mapped files before upgrading ALLBASE/SQL to G3.

- "Year 2000 and Other Enhancements in VPLUS" describes enhancements in B.06.07 version of VPLUS for the following functionality: turn-of-century enhancement, ability to define default Field Type and Data Type for a form, ability to define more than 52 single character fields per Form, new intrinsics to get and set Save Field values programmatically, and the ability to renumber a Form with interactive FORMSPEC.

- "CI Enhancements" describes POSIX filenames, `COPY` command enhancements, and new Evaluator functions. It also reviews the CI Enhancements in C.55.03. It describes the new functionality added to the CI software to improve the ease of use and overall functionality of these commands: `LISTF,LISTFILE, PAUSE, PRINT, FLABELINFO` and `FINFO()`, and `INPUT`. There are two new variables: `HPLASTSPID` and `HPSPOOLID`.

- "The REMSH Client Functionality" gives more technical information on the REMSH Client that allows a user to connect to the remote system, execute a command on that remote system, and see the results of that command.

- "Transaction Manager" discusses these new features: System Log Expansion, Transaction Copy-Forward, and Checkpoint Improvement.

- "New Features of DTS/TIO Subsystem" describes the new functionality that was added in release 6.0 and reviews the functionality in release C.55.02. These enhancements significantly increase the number of DTCs, supported modem types, and parity setting for HP 3000 machines.

- "Introducing ODBCLink/SE" describes the product replacing HP PC/API for ALLBASE/SQL links.

- "Introducing Samba/iX" provides more technical detail on Samba/iX.

- "Developer's Kit for Java/iX" provides more information on Java for MPE/iX.

- "Hewlett-Packard NetWorker ClientPak for MPE/iX" gives more technical information the Hewlett-Packard Legato NetWorker ClientPak.

- "DNS BIND/iX and Syslog/iX" describes the functionality of DNS BIND/iX and Syslog/iX. This article includes the available utilities and descriptions of how to run both.

Chapter 11, Product Release History

This chapter adds product information for MPE/iX Release 6.0 and updates the termination dates in the Supported System Release Matrix table.

Chapter 12, Catalog of User Documentation

This chapter provides two types of manual listings:

- A listing of all new or updated manuals by the time of the MPE/iX 6.0 Release.

- A current listing of the MPE/iX manuals grouped by collections.

MPE/iX Patches on HP Electronic Support Center

by Patch Support Team
Commercial Systems Division

MPE/iX patches for MPE/iX Release 5.0 and beyond, are available on the HP Electronic Support Center to all customers.

Features and Benefits

The new patch access and delivery system benefits all MPE/iX customers with:

- Improved overall communication between HP and customers.

- Provision of useful and timely information for patch justification and decision making.

- Reduced system downtime for known problems.

- Reduction of the turnaround time for patch availability and delivery.

- Close to 24*7 access time.

- Unification of the MPE/iX and HP-UX patch delivery process.

Electronic access to patch information and delivery of patches provide three basic services:

1. Access to patch information in an automated, timely and accurate manner.

2. Electronic downloading of patch information and binaries.

3. Proactive notification of new patches via email.

Access Method to the HP Electronic Support Center

To serve customers the HP Electronic Support Center provides World Wide Web access for downloading patches.

Access to World Wide Web Server (www)

HP Electronic Support Center is available through the World Wide Web.World Wide Web access is the easiest, fastest, and most popular method of browsing for patch information and downloading patches. It is more reliable, especially for large patches.

- **U.S. Web accessing address:**
  http://us-support.external.hp.com

- **European Web accessing address:**
  http://europe-support.external.hp.com

Electronic Digests

If you want to keep yourself up-to-date on the latest development of MPE/iX patches, you can sign up for the daily Security Bulletin and weekly mpeix_patch Bulletin. Once you have subscribed to these two bulletins, you will receive these digests on a periodic basis via electronic mail. HP Electronic Support Center will inform you proactively about newly developed security and GR patches. For more information, refer to the instructions on the Electronic Support Center website.

AUTOPAT Installation Document

AUTOPATINST is a document with instructions to assist you in installing one or more patches needed by your MPE/iX system using the AUTOPAT installation tool.

To retrieve the AUTOPAT patch installation instructions:

1. Access the "HP Electronic Support Center" website using the appropriate web address for your country.

2. Enter login user ID and password.

3. Click on Technical Knowledge Base.

4. Click on Retrieve a document.

5. Enter `AUTOPATINST` in the Enter Document ID field.

HP Instant Information for Delivery of Learning Products

Hewlett-Packard recently introduced HP Instant Information, a new way of delivering learning products on CD-ROM that gives you easier access to multiple HP support documents. Instant Information on CD-Rom replaces HP LaserROM as the viewing mechanism for documentation. With Instant Information, Hewlett-Packard has created a delivery system that better meets the current needs of customers. At the same time, we're establishing an infrastructure to meet future demand.

Improved Searches, Printing and Collection

Starting with MPE/iX 6.0, documentation will be written in FrameMaker+SGML, then published and stored on Instant Information. This uses SGML instead of HP's proprietary HP Tag format. Instant Information reduces costs and employs off-the-shelf, standards-based software for viewing on PCs.

Using HP Instant Information, you can look at a book's table of contents or index and, with a mouse click, jump to a specific topic. You can search multiple manuals and documents quickly for matching key words. Users no longer will have to rely on paper manuals. Access to HP support information will be easier and more efficient.

With the new browser, you can view both document text and the table of contents in the same window. You will find an improved collection structure. Collections now will be based on specific products (such as NS3000/iX) rather than on hardware platforms, which was the base used by HP LaserROM bookshelves.

Another significant improvement is the advanced printing functionality of HP Instant Information. One or more chapters, or entire books can be printed with page numbers. Also, proportional fonts, graphics, and tables can be printed as postscript-quality.

Documents Available on the World Wide Web

The release of HP Instant Information represents the critical first step of converting all necessary documents. With 6.0 many documents will also be available on the world wide web: *http:/www.docs.hp.com.* These new delivery media will dramatically change the way our learning community accesses technical information.

The first release of HP Instant Information for MPE/iX will include the most critical MPE/iX learning products. Books for which no revisions are planned will remain on the HP LaserROM (archive copy provided with Release MPE/iX 6.0) and will not be converted to HP Instant Information.

You can subscribe to HP Instant Information through your support contact or by purchasing a stand-alone subscription. Customers who currently receive documentation updates on HP LaserROM for MPE/iX automatically, will receive HP's new Instant Information when they upgrade to MPE/iX 6.0.

2   System Information and Preparation

This chapter includes the following articles on system information:

- Updating to and Backdating from MPE/iX 6.0
- Planning Your Move to MPE/iX Release 6.0
- Upgrading a Pre-997 Corporate Business Server to a 997
- Special Note on SCSI Device Configuration for MPE/iX 6.0
- New MPE/iX Release Obsolescence Strategy
- HP System Account and Directory Naming Structure

Updating to and Backdating from MPE/iX 6.0

by MPE/iX System Release Team

Updating to MPE/iX 6.0

To update or install MPE/iX Release 6.0, your system must already be on the MPE/iX 5.5 (C.55.xx) or the MPE/iX 5.0 General Release (C.50.xx).

If your system is not on MPE/iX 5.5 or MPE/iX 5.0 General Release, you must first update your system to MPE/iX 5.0 before you can update to MPE/iX 6.0 (C.60.00).

If your system is on the MPE/iX 5.0 Limited Release (X.50.20), you must first update your system to MPE/iX 5.0 General Release (C.50.00) before you can update to MPE/iX 6.0.

Backdating from MPE/iX 6.0

If after installing MPE/iX 6.0 (C.60.00), you need to backdate to an earlier release, the only supported releases that you may backdate to are MPE/iX 5.5 (C.55.xx) and MPE/iX 5.0 General Release (C.50.xx). Contact your HP Support Representative prior to performing the backdate to ensure that all options have been discussed.

Refer to the Supported System Release Matrix in Chapter 11, "Product Release History," in this *Communicator* for the expiration dates for support of MPE/iX 5.5 and MPE/iX 5.0 General Releases.

Planning Your Move to MPE/iX Release 6.0

by Rich Webber
Commercial Systems Division

This article lists requirements for installation of or updating to Release 6.0 system software. This includes requirements for adding-on subsystem purchased products (SUBSYS) and updating your system software version level, and installing your system software. If your system does not meet the requirements listed here, make the appropriate corrections prior to installing or modifying your system.

The requirements described in this article are:

- System Software Version Compatibility
- Third-Party Software Compatibility
- CD-ROM Disk Drive Compatibility
- LDEV 1 Disk Drive Minimum Capacity
- LDEV 1 Disk Drive Maximum Usage
- Estimating Disk Space Requirements

System Software Version Compatibility

Verify you are starting with a compatible version of the system software. Only systems on General Release 5.0 or Release 5.5 are supported for the update to 6.0. If you are running a system software version that is older than 5.0 (General Release), you must perform two updates:

1. Update to 5.0 (General Release) or Release 5.5.

   To update to 5.0, the media options and corresponding documentation are as follows:

   - Tapes—*HP 3000 MPE/iX Installation, Update, and Add-On Manual, MPE/iX Release 5.0 (General)*

   - CD-ROM—*Using CD-ROM to Update Your HP 3000 System Software, MPE/iX Release 5.0 (General)*

   To update to Release 5.5 using either Tape or CD-ROM system software release media, use the following manual:

   - *HP 3000 MPE/iX System Software Maintenance Manual*

2. Then update to version 6.0 system software using tape or CD-ROM system software release media and the following manual:

- *HP 3000 MPE/iX System Software Maintenance Manual*

Refer to the appropriate manuals for further information on what is required for updating to 5.0 or 5.5 including any hardware requirements. Contact your HP support representative if you need more information.

Third-Party Software Compatibility

Verify that any third-party software product you are running is compatible with the latest version of the operating system software. Do this *before* you modify the system.

CD-ROM Disk Drive Compatibility

To update or add-on to your system software using a CD-ROM, you must have a CD-ROM drive installed and configured. Verify that the CD-ROM drive you are using is one of the following:

- HP Series 6100 Model 600/A HP-IB (C1707A)

- HP Series 6100 Model 700/S SCSI (A1999A)

- Toshiba XM-3401TA

- Toshiba XM-4101TA

An Upgrade Kit (C2293U) for the Series 6000 peripheral package allows you to use internal SCSI drives on some HP 3000 computer systems.

If you do not have one of these compatible drives, contact your HP representative to order one.

LDEV 1 Disk Drive Minimum Capacity

LDEV 1 (the system disk) requires a minimum capacity of 500 MBytes. Therefore, you cannot use an older HP7933 or HP7935 disk drive as LDEV 1. When you update to MPE/iX Release 6.0, you may still use HP7933 and HP7935 drives elsewhere on your system, but these drives do not have a large enough capacity to serve as the system disk.

To identify your LDEV 1 device, use either the `SHOWDEV` 1 or `DSTAT ALL` command to determine the logical device numbers of configured devices.

- If the device configured as your LDEV 1 is one of the supported devices, proceed with your system modification (update or add-on).

- If the device configured as your LDEV 1 is not one of the supported devices, you **must** replace your LDEV 1 device with a supported device.

LDEV 1 Disk Drive Maximum Usage

If you have a CIO system and you are using a disk drive with over 2 GBytes capacity as your LDEV 1, any disk space over the 2 GBytes is not available for use. It cannot and will not be used for system or user files.

If you have an NIO system and you are using a disk drive with over 4 GBytes capacity as LDEV 1, any disk space over the 4 GBytes is not available for use. It cannot and will not be used for system or user files.

The 9 and 18 GByte disk drive may be used as LDEV 1, but this not recommended because a maximum of 4 Gb is usable by LDEV1.

The predefined variable *hpcpuname* contains the name of your computer model as shown in this example:

```
:SHOWVAR hpcpuname
HPCPUNAME = Series 957
```

The following table lists NIO/CIO systems.

Table 2-1   **NIO/CIO Systems**

NIO Systems	CIO Systems (Support CIO and NIO Cards)	CIO Systems
Series 9x7RX, 9x7LX, 9x7SX, 9x8LX, 9x8RX, 9x9KS, 99x Update this	Series 920, 922, 932, 948, 958	Series 925, 935, 949, 950, 955, 960, 980

Estimating Disk Space Requirements

Before you begin to modify your system software make sure that you have enough disk space to complete the modification. Modifying system software includes: updating or reinstalling your system software and adding on purchased products from the SUBSYS tape. There are three types of disk space requirements that are referenced during the modification process:

- The permanent (net) amount of non-contiguous disk space required by the system software after it is modified.

- The maximum (peak) amount of non-contiguous disk space required during the system software modification process.

- The amount of contiguous disk space the update tool (either `AUTOINST` or `HPINSTAL`) requires to modify the system software.

NOTE
The disk space values listed in this section are estimated values only. The actual amount of disk space used on your system will vary.

Non-Contiguous Disk Space Estimates

The following table lists the amount of non-contiguous disk space sectors required for the three operating system components (SLT, FOS, and SUBSYS) of the currently supported versions of the operating system software. During the modification process, some files are duplicated temporarily, sometimes older versions of files are retained temporarily. This causes the peak amount of disk space usage to be greater than the net amount when the modification is complete.

You must have the peak amount of disk space available on your system during the process to successfully modify your system.

The combined System Load Tape (SLT) and Fundamental Operating System (FOS) are the minimum net requirements for any system software version level. Ensure that you have enough room on LDEV1 for all the SLT files. The FOS files do not have to go on LDEV 1.

Table 2-2   **Disk Space Requirements in K Sectors for Supported Releases**

System Software Components	Disk Space in K Sectors for Version Level				
	5.0	5.5		6.0	
		Peak	Net	Peak	Net
SLT only *	1,318	n/a	1,268	n/a	1,274
SLT and FOS	2,027	2,378	2,092	2,821	2,533
SLT, FOS, SUBSYS	3,226	3,637	3,170	3,926	3,489

* Provided for reference only. An operational system requires the SLT and FOS files at a minimum.

To estimate the amount of additional permanent (net), non-contiguous disk space required to update your system software, subtract the value of your current version from the value of the version to which you are updating.

The values listed for the SUBSYS (purchasable subsystem products) are the maximum possible. This value is the total disk space sectors required for all possible purchasable products. You may have ordered only selected SUBSYS products.

Estimate additional disk space requirements based on the number and kind of software subsystem products purchased. If you are not adding-on any new purchased products, the disk space sectors currently being used by your existing products will probably not change significantly.

Converting Between Disk Sectors and MBytes

If your products are listed in MBytes, particularly third-party products, estimate the disk space sectors by performing the following conversion:

- n MBytes x 1,000,000 sectors/256 bytes = m sectors

where:

- n = the number of MBytes
- m = the number sectors

Example 1. MBytes to sectors:

- If `VALIDATE` reports 20 MBytes total on a tape:

  - 20 x 1,000,000/256 = 78,125 sectors

Example 2. Sectors to MBytes:

- If a disk has a device size of 7,824,336 sectors:

  - 7,824,336 x 256/1,000,000 = 2,003 MBytes

  - Therefore the disk is a 2000 MByte disk, (also known as 2 Gbyte disk).

For a closer estimate, use the value 1,048,576 instead of the value 1,000,000.

For a quick estimate, use the value 4,000 instead of the conversion value of 1,000,000/256.

Contiguous Disk Space Requirements

The maximum amount of contiguous disk space sectors required to complete a system modification on any system is:

- 120,000 sectors

Use `CHECKSLT` to estimate the amount of contiguous disk space sectors required to complete a system modification on **your** system.

Refer to the *HP 3000 MPE/iX System Software Maintenance Manual* for directions on using `CHECKSLT`.

Reserving Disk Space

The update and add-on process requires a minimum number of contiguous and non-contiguous disk space sectors. Ensure that you have enough contiguous disk space on LDEV 1 to complete your task. The total amount of non-contiguous disk space does not need be available on LDEV 1.

Reserve enough contiguous and non-contiguous disk space on your system **before** you perform an install, update, or add-on. Refer to Appendix C, "Managing Disk Space," in the *HP 3000 MPE/iX System Software Maintenance Manual* for information on reserving disk space.

Upgrading a Pre-997 Corporate Business Server to a 997

by Hank Cureton
Commercial Systems Division

Introduction

The I/O configuration on a 997 server has one additional hardware layer (the HP-PB I/O bus converter) compared to the earlier Corporate Business Servers (990, 991, 992, 995, and 996). As a result, the system I/O configuration on a 997 needs to be updated to reflect this additional hardware layer.

Example:

A typical SCSI disk device I/O configuration on a 995 would be described by the following I/O path:

`2/52.3.0`

The same SCSI disk device I/O configuration on a 997 would be described by the following I/O path:

`2/28/52.3.0`

NOTE   The HP-PB bus converter always has the fixed slot address 28 (decimal).

The utility packages `IOCNVRT` and `JADETOOL` are used to convert the OS software I/O configuration of a pre-997 Corporate Business Server to the new 997 I/O configuration format.

IOCNVRT.PUB.SYS

The command file `IOCNVRT.PUB.SYS` is used to convert the system `SYSGEN` I/O configuration from a pre-997 Corporate Business Server configuration to a 997 configuration, or vice versa.

Syntax:

`:IOCNVRT {`*source group*`}, {`*target group*`}`

source group   is the name of a valid `SYSGEN` configuration group in the `SYS` account, containing the "source system" I/O configuration. This is a required parameter. The files in the *source group* will remain untouched unless the *target group* name is the same as the *source group*. The group must exist, and its files must be valid for `IOCNVRT` to work.

target group   is the name of the SYSGEN configuration group in the
                     SYS account into which the "target system"
                     configuration files will be stored. This is a required
                     parameter. If the group does not exist, it will be created
                     by the program.

Examples:

```
:IOCNVRT CONFIG, CONFIG
```

This command converts a pre-997 configuration in the group
CONFIG.SYS into a 997 configuration in the same group CONFIG.SYS.

```
:IOCNVRT CONFIG, NEWCONFG
```

This command converts a pre-997 configuration in the group
CONFIG.SYS into a 997 configuration and places it into the group
NEWCONFG.SYS.

JADETOOL

The JADETOOL utility is used to convert a pre-997 Corporate Business
Server data communications configuration to a 997 data
communications configuration.

Example:

```
:JADETOOL
Jade Tool      Version A0000006

  W A R N I N G ! ! ! ! !
This program is only to be used after authorization from HP

In a failure situation, use of this program could corrupt
the configuration file which could result in a system abort

It is recommended that this program be only used on a copy
of the NMCONFIG configuration file

Do you understand and agree to the above warning? (Y/N) y

Enter file name to convert: NMCONFIG.PUB.SYS
```

Restrictions

IOCNVRT requires that its user has SM (System Manager) capability, as
a safety measure to prevent unauthorized or inadvertent changes to the
system configuration file groups.

JADETOOL does not automatically convert RJE link I/O paths. This
means that any users of RJE must reconfigure those links manually in
NMMGR.

Recommended Sequence of Steps for Upgrading from a Pre-997 to a 997

The recommended order for upgrading a pre-997 Corporate Business Server (990, 991,992, 995 or 996) to a 997 server follows:

1. Update the system with C.55.05 or C.6.0.

2. Reboot the system.

3. Back up the S995 `CONFIG.SYS` group and the file `NMCONFIG.PUB.SYS`.

NOTE   In the event that you need to convert back to the pre-997 configuration, these files will be needed.

4. Convert the `SYSGEN` configuration files using the command file `IOCNVRT`.

   `:IOCNVRT CONFG995,CONFG997`

   where your current 995 `SYSGEN` configuration files are located in the group `CONFG995.SYS` and the new 997 `SYSGEN` configuration files will be placed in the group `CONFG997.SYS`.

5. Convert the file `NMCONFIG.PUB.SYS` using the tool `JADETOOL`.

   `:RUN JADETOOL`

6. If RJE links are used on the system, run `NMMGR` to update any RJE links.

   `:RUN NMMGR.PUB.SYS`

7. Shut down the system.

8. Upgrade the hardware to a 997.

9. Reboot the system using the new configuration group.

   `:START NO RECOVERY GROUP=NEWCONFIG`

10. If the 997 `SYSGEN` configuration files are not in the group `CONFIG.SYS`, copy them into the `CONFIG.SYS` group.

11. Validate DTS.

   `:RUN NMMGR.PUB.SYS`

12. Shut down and reboot the system from `CONFIG.SYS` configuration group.

Special Note on SCSI Device Configuration for MPE/iX 6.0

by Ruben Gallegos
Commercial Systems Division

Overview

With the release of MPE/iX 6.0, *all* SCSI devices *must* have as part of their device configuration a pseudo device with a "transparent_mgr" that was created with SYSGEN. This requirement is for *all* SCSI devices on a *system* running MPE/iX 6.0.

Future I/O technology requires the consistent usage of "transparent_mgr" in the base MPE/iX I/O configuration.

The default configuration files that are part of the factory System Load Tape (SLT) all have the "TRANSPARENT_MGR" requirement built into the device configuration for all SCSI devices.

Today, all systems with SCSI devices should already be configured with the "transparent_mgr" requirement except for Magneto-Optical(MO) Devices. Previous MO installation instructions disallowed "transparent_mgr" in their configuration.

If you are performing an UPDATE to MPE/iX 6.0 or are using your archived configuration files when performing an INSTALL, you *must* check and ensure that each SCSI device has a "TRANSPARENT_MGR" in its device path, prior to performing the UPDATE or INSTALL. Failure to perform this step will cause problems that could lead to inability to access data on the SCSI device.

The remainder of this article provides some examples for configuration of SCSI devices on HP3000 systems. As with most devices on the HP3000, SCSI device configuration is done via SYSGEN. The actual entries made in SYSGEN are dependent upon the *device* rather than the I/O architecture (CIO, hybrid or Precision Bus).

On the HP3000 the requirement for configuration of SCSI devices is to actually make *two path* entries per device. The first entry is the pseudo device entry for the "TRANSPARENT_MGR." This entry is *not* a logical device itself; rather, it is a place holder required by the SCSI Device Manager Software. The second entry is for the actual device itself. This is true for CIO, "hybrid" or Precision Bus systems even though these systems use different SCSI controllers.

In the past there was *only* one set of devices that did not follow this convention, Magneto-Optical Devices. With the release of MPE/iX 6.0, Magneto-Optical devices *must* also make use of pseudo devices in their configuration. You must add *pseudo devices* for the MO Autochanger

and MO disk drives. Not adding *pseudo devices* for an MO device on MPE/iX 6.0, will cause problems that could lead to the inability to access data from this device.

Examples:

Pre-MPE/iX 6.0 Configuration

All SCSI devices, *except* Magneto-Optical devices should have a "transparent_mgr" configured in systems running MPE/iX versions *prior* to MPE/iX 6.0.

This example shows the standard MPE/iX 5.5 SCSI device configuration, for a C1700A 20GB Magneto-Optical Jukebox. This library is actually made up of an Autochanger and two disk drives, resulting in three LDEVs associated with this library. Notice that the 'TRANSPARENT MANAGER' is *not* used with any of these devices. Notice also that this happens to be a Precision Bus system.

```
PATH:   24                                    LDEV:
  ID:   A1703-60003-SCSI                      TYPE: DA
PMGR:   SCSI_DAM                              PMGRPRI:    6
LMGR:                                         MAXIOS:     0

PATH:   24.3                                  LDEV:    31     ^
  ID:   HPC1700A                              TYPE: AC        |
PMGR:   SCSI_AC_DM                            PMGRPRI:   10   |
LMGR:   LOGICAL_DEVICE_MANAGER                MAXIOS:     0   |  These three
                                                             |  Logical
PATH:   24.4                                  LDEV:    32     |  Devices
  ID:   HPC1701A                              TYPE: DISC      |  represent
PMGR:   SCSI_DISC_DM                          PMGRPRI:    8   |  ONE C1700A
LMGR:   LOGICAL_DEVICE_MANAGER                MAXIOS:     0   |  20GBtye
                                                             |  Device
PATH:   24.5                                  LDEV:    33     |
  ID:   HPC1701A                              TYPE: DISC      |
PMGR:   SCSI_DISC_DM                          PMGRPRI:    8   |
LMGR:   LOGICAL_DEVICE_MANAGER                MAXIOS:     0   v
```

MPE/iX 6.0 Magneto-Optical Device Configuration

This example is the new/proper configuration for the C1700A 20GB Magneto-Optical Jukebox on MPE/iX 6.0. This device is made up of the Autochanger and two disk drives resulting in three LDEVs associated with this library. Notice that the 'TRANSPARENT MANAGER' *must* now be used with both the Autochanger and disk drives. Notice also that this happens to be a Precision Bus system.

```
PATH:   24                                    LDEV:
  ID:   A1703-60003-SCSI                      TYPE: DA
PMGR:   SCSI_DAM                              PMGRPRI:    6
LMGR:                                         MAXIOS:     0

PATH:   24.3                                  LDEV:
  ID:   PSEUDO                                TYPE: DA
PMGR:   TRANSPARENT_MGR                       PMGRPRI:    6
LMGR:                                         MAXIOS:     0

PATH:   24.3.0                                LDEV:    31     ^
```

```
   ID:   HPC1700A                            TYPE: AC
PMGR:   SCSI_AC_DM                        PMGRPRI:    10
LMGR:   LOGICAL_DEVICE_MANAGER             MAXIOS:     0     These three
                                                            Logical
PATH:   24.4                                 LDEV:          Devices
   ID:   PSEUDO                              TYPE: DA        represent
PMGR:   TRANSPARENT_MGR                   PMGRPRI:     6     ONE C1700A
LMGR:                                      MAXIOS:     0     20GBtye
                                                            Device
PATH:   24.4.0                               LDEV:   32
   ID:   HPC1701A                            TYPE: DISC
PMGR:   SCSI_DISC_DM                      PMGRPRI:     8
LMGR:   LOGICAL_DEVICE_MANAGER             MAXIOS:     0

PATH:   24.5                                 LDEV:
   ID:   PSEUDO                              TYPE: DA
PMGR:   TRANSPARENT_MGR                   PMGRPRI:     6
LMGR:                                      MAXIOS:     0

PATH:   24.5.0                               LDEV:   33
   ID:   HPC1701A                            TYPE: DISC
PMGR:   SCSI_DISC_DM                      PMGRPRI:     8
LMGR:   LOGICAL_DEVICE_MANAGER             MAXIOS:     0     v
```

MPE/iX 6.0 Tape Configuration

This example shows a standard SCSI tape device configuration. This form should be followed for all SCSI devices including Magneto-Optical Devices on MPE/iX 6.0. This particular example shows the configuration for the HPC1503B tape drive using the ID= HPC1503B. Notice that this happens to be a Precision Bus system.

```
PATH:   52                                   LDEV:
   ID:   A1703-60003-SCSI                    TYPE: DA
PMGR:   SCSI_DAM                          PMGRPRI:     6
LMGR:                                      MAXIOS:     0

PATH:   52.0                                 LDEV:          ^
   ID:   PSEUDO                              TYPE: DA        Repeat for
PMGR:   TRANSPARENT_MGR                   PMGRPRI:     6     each
LMGR:                                      MAXIOS:     0     Physical
                                                            Device
PATH:   52.0.0                               LDEV:    7
   ID:   HPC1503B                            TYPE: TAPE
PMGR:   SCSI_TAPE_DM                      PMGRPRI:    10
LMGR:   LOGICAL_DEVICE_MANAGER             MAXIOS:     0     v
```

MPE/iX 6.0 Disk Device Configuration

This particular example shows the configuration for the 2.0 GB version of the C302x disk using the ID= HPC3010M1 (the 1.0 GB version is HPC2247M1). Notice that this happens to be a CIO system.

```
PATH:  6/4.4                              LDEV:
  ID:  HP27251A                           TYPE: DA
PMGR:  CIO_SCSI_DAM                       PMGRPRI:    6
LMGR:                                     MAXIOS:     0

PATH:  6/4.4.6                            LDEV:             ^
  ID:  PSEUDO                             TYPE: DA          |   Repeat for
PMGR:  TRANSPARENT_MGR                    PMGRPRI:    6     |   each
LMGR:                                     MAXIOS:     0     |   Physical
                                                           |   Device
PATH:  6/4.4.6.0                          LDEV:    16       |
  ID:  HPC3010M1                          TYPE: DISC        |
PMGR:  SCSI_DISC_DM                       PMGRPRI:    8     |
LMGR:  LOGICAL_DEVICE_MANAGER             MAXIOS:     0     v
```

New MPE/iX Release Obsolescence Strategy

by Jon Cohen
Commercial Systems Division

The "MPE/iX Release Obsolescence Strategy" is a collection of guidelines that determines how long a particular MPE/iX Release is supported.

In recent years, our Obsolescence Strategy has grown to be very complex, ambiguous, and confusing. Even HP people had difficulty interpreting these guidelines to determine when a particular release would no longer be supported.

In response, we have adopted a new, simpler Obsolescence Strategy:

- In general, HP will support the two most recent mainline releases.

- In order to facilitate customers moving to new releases, we will support the oldest release for at least six months after the new release ships. In other words, Release "N-2" will still be supported for at least six months after Release "N" ships.

- The release of an Express Release or a Powerpatch on any particular release does not extend its support life.

Given the rate with which we ship releases, this new strategy implies that any particular release will be supported for at least three years, and in most cases, most releases will be supported for a significantly longer period of time.

This new strategy is effective now. This implies that Release 5.0, which first shipped February 17, 1995, will become obsolete as early as six months after Release 6.0 ships.

HP System Account and Directory Naming Structure

by The Release Delivery Team
Commercial Systems Division

Since the MPE operating system originated, Hewlett-Packard has developed and maintained a number of system accounts that are considered reserved. With the addition of POSIX, this idea is extended to include a number of Hierarchical File System (HFS) directories as well. Each time you perform a system installation or update, the new information added to your system is placed in these reserved structures.

To avoid overwriting your user accounts and directories, Hewlett-Packard recommends that you do not use reserved names. Rename all non-Hewlett-Packard accounts and directories that use any of the reserved names.

System Accounts

The following accounts are currently in use:

BIND
CONV
CLL
HPLANMGR
HPNCS
HPOFFICE
HPOPTMGT
HPPL85
HPPL87
HPPL89
HPSKTS
HPSPOOL
ITF3000
INDHPE
RJE
SNADS
SOFTREP
SUPPORT
SYSLOG
SYS
SYSMGR
TELESUP
WWW

System Directories

The following directories are currently in use:

Table 2-3   **System Directories**

/bin	/usr/lib/terminfo/m
/etc	/usr/lib/terminfo/n
/hpshell-examples	/usr/lib/terminfo/o
/hpshell-examples/lexyacc	/usr/lib/terminfo/p
/lib	/usr/lib/terminfo/q
/tmp	/usr/lib/terminfo/r
/usr	/usr/lib/terminfo/s
/usr/curses	/usr/lib/terminfo/t
/usr/include	/usr/lib/terminfo/u
/usr/include/sys	/usr/lib/terminfo/v
/usr/lib	/usr/lib/terminfo/w
/usr/lib/curses	/usr/lib/terminfo/x
/usr/lib/tabset	/usr/lib/terminfo/y
/usr/lib/terminfo	/usr/lib/terminfo/z
/usr/lib/terminfo/a	/usr/lib/terminfo/0
/usr/lib/terminfo/b	/usr/lib/terminfo/1
/usr/lib/terminfo/c	/usr/lib/terminfo/2
/usr/lib/terminfo/d	/usr/lib/terminfo/3
/usr/lib/terminfo/e	/usr/lib/terminfo/4
/usr/lib/terminfo/f	/usr/lib/terminfo/5
/usr/lib/terminfo/g	/usr/lib/terminfo/6
/usr/lib/terminfo/h	/usr/lib/terminfo/7
/usr/lib/terminfo/	/usr/lib/terminfo/8
/usr/lib/terminfo/j	/usr/lib/terminfo/9
/usr/lib/terminfo/k	/usr/mail
/usr/lib/terminfo/l	/usr/man

3   System Management

This chapter contains the following articles related to networking and client-server applications:

- User-Defined Job Queues and `ALTJOB HIPRI` Enhancements

- Overview of HP Patch/iX

- Overview of HP Stage/iX

- Transaction Manager

- Increase in Number of TCP Connections

- Overview of CI Enhancements

- MPE/iX POSIX Installation Changes

- DNS BIND/iX and Syslog/iX

- Easytime/XL Bundled with MPE/XL 6.0

User-Defined Job Queues and ALTJOB HIPRI Enhancements

by Mohan Das Konanoor and Jayaram

Enhancements Summary

Previously MPE/iX had one job queue into which all the submitted jobs went before getting launched by the dispatcher. A few long jobs could prevent many short jobs from running.

MPE/iX can now have up to 255 user-defined job queues, each having a separate limit on the number of jobs that can be run. Users can specify the queue to which their jobs should log on by using the newly added `;JOBQ=` parameter of `!JOB` command or `!STREAM` command. There will be one default job queue named `HPSYSJQ`, which will be created during the system bootup.

- Three new commands have been added: `NEWJOBQ`, `PURGEJOBQ`, and `LISTJOBQ`

- `LIMIT, ALTJOB, JOB` and `STREAM` commands now have a new parameter, `;JOBQ=`.

- `LIMIT` command can now be used to increment or decrement the limit values.

- `SHOWJOB` command now has a new parameter `;JOBQ` to display the queue name to which each job belongs.

- `ALTJOB` command now has a new option `;HIPRI` to change the priority of the job as HIPRI. This option can be used to override the job limit. SM or OP capability is required for this option.

New Commands

NEWJOBQ

Syntax:

```
NEWJOBQ qname  [;limit=n]
```

The `NEWJOBQ` command creates a new job queue. `SM/OP` capability is required to execute this command.

Limit is the only queue controlling property. The jobs in the queue are sorted by their `INPRI`. In case of a tie for `INPRI`, jobs are sorted by their `INTRO` time.

The global limit takes precedence over individual queue limits. That is, even if a JOBQ has a slot available, if the overall limit has been reached, jobs have to wait until one of the jobs finishes or the global

limit is increased. When a global slot becomes available, the next job is picked from among the eligible job queues (those which haven't yet reached their individual limits) using the following algorithm:

- Across all job queues, the highest INPRI job is selected. In case of a tie for INPRI, the one which was introduced earliest is selected. There cannot be a tie in INTRO time.

The job queues persist across reboots, provided a START RECOVERY is done. Any other system starts will cause the job queues to be deleted and they will have to be created again.

This command is available in a session, job, or in BREAK. Pressing [Break] has no effect on this command. This command is not allowed in SYSSTART.

Parameters:

qname is the name of the queue to be created. Queue names can contain any characters. Queue names can be up to 8 characters in length and longer names will be truncated. If a queue of this name already exists, an error is indicated.

limit is the maximum number of jobs that can be allowed in this queue. The limit value can be changed using the :LIMIT [+-]*n*;JOBQ= command. If omitted, a value of zero is assumed.

Example:

```
:NEWJOBQ MYJOBQ;LIMIT=100
```

PURGEJOBQ

Syntax:

```
PURGEJOBQ    qname
```

The PURGEJOBQ command deletes a job queue. The queue will be deleted only if it is empty, that is if no jobs are waiting or executing in the queue. The default system job queue can not be purged. The user must have SM/OP capability to execute the command.

This command is available in a session, job, or in BREAK. Pressing [Break] aborts the execution of this command. This command is not allowed in SYSSTART.

Parameters:

qname is the name of the queue to be deleted.

Example:

```
:PURGEJOBQ myjobq
```

LISTJOBQ

Syntax:

```
LISTJOBQ
```

The `LISTJOBQ` command allows the user to list all the existing job queues in the system. It displays the queue name, limit, number of jobs in the queue that are in the `EXEC` state and the total number jobs in the queue, (Number of jobs in the `EXEC` state + Number of jobs in the `WAIT` state). This command is not allowed in `SYSSTART`.

Example:

```
:LISTJOBQ

JOBQ      LIMIT     EXEC   TOTAL

HPSYSJQ   3500      1      1
MYJOBQ    100       1      1
MJQ       10        1      2
```

Modified Commands

LIMIT

Syntax:

```
LIMIT [[+/-] numjobs],
      [[+/-] numsessions
      [[;JOBQ=] qname]
```

The `LIMIT` command allows users with OP capability to change the job and session limits. As a result of this enhancement, it can be used to change the limit value of individual job queues. The `LIMIT` command now accepts a new parameter `;JOBQ=`. It also accepts + or – before the numeric values to indicate increment or decrement.

Parameters:

qname is the name of the job queue whose limit is to be changed. If the queue name is not specified, `LIMIT` command will alter the global limit values.

+/-   if specified before the numeric value, the corresponding limit value will be incremented/decremented.

Examples:

```
:LIMIT 60;JOBQ=myjobq
```
Set the limit of MYJOBQ to 60

```
:LIMIT –1;JOBQ=jobq
```
Decrement the limit by 1

```
:LIMIT +1
```
Increment the global job limit by 1

JOB

Syntax:

```
JOB [jobname,]username[/userpass].acctname[/acctpass]
                      [,groupname[/grouppass] ]

[;TIME=cpusecs] [;PRI={BS
                       CS
                       DS
                       ES}]
```

```
[;INPRI=inputpriority
 ;HIPRI ][;RESTART]
```

```
[;OUTCLASS= [ [device] [,[outputpriority][,numcopies] ] ] ]
```

```
[;TERM={termtype}] [;PRIVATE] [;SPSAVE] [;JOBQ= qname]
```

`JOB` command now accepts a new parameter `;JOBQ=`. Users can specify the job queue name into which a particular job should log on. If a job queue is specified in the `STREAM` command then it overrides the name specified in the `JOB` command. If no queue name is specified default system job queue is used.

Parameters:

qname is the name of the queue into which the job should log on.

Example:

```
:JOB foo,manager.sys;JOBQ=myjobq
```

ALTJOB

Syntax:

```
ALTJOB [JOB=] {#Jnnn
               #Snnn}
[ ;INPRI=inputpriority |
  ;HIPRI] [ ;OUTDEV={ldev
                     devclass}]
[;JOBQ=qname]
```

`ALTJOB` alters the attributes of waiting or scheduling jobs. `ALTJOB` now accepts a new parameter `;JOBQ=`. Jobs waiting or executing in a queue can moved to other queues. `;HIPRI` is used to change the priority of the job as HIPRI. This option can be used to override the job limit. SM or OP capability is required for both `JOBQ= and ;HIPRI`.

When an executing job is moved, the limit of the target queue is ignored and the job continues to execute in the new queue. A waiting job continues to wait in the new queue if the queue has already reached its limit.

Parameters:

qname is the name of the job queue into which the job is to be moved.

hipri is the name of the option to change the priority of a job.

Example:

```
:ALTJOB #j324;JOBQ=mjq
```

STREAM

Syntax:

```
STREAM [filename] [,char]
[;AT = timespec][;DAY = {day-of-week
                         day-of-month
                         days-until-month}]
[;DATE = datespec]
[;IN = [days[,[hours] [,minutes] ] ] ]
[;JOBQ = qname]
```

STREAM command now accepts a new parameter ;JOBQ= User can specify the queue name into which a particular job should go. The name specified overrides the queue name specified in the JOB command.

Parameters:

qname is the name of the queue into which the job must log on. If no queue name is specified the default system job queue will be used.

Example:

```
:STREAM FOO; JOBQ= myjobq
```

SHOWJOB

Syntax:

```
SHOWJOB [ [#]Snnn
           [#]Jnnn
           STATUS
           SCHED
           item[;item[;...] ] ] [;*listfile]
           [;JOBQ]
```

SHOWJOB command displays status information about jobs/sessions. It now accepts a new format parameter ;JOBQ which will indicate the queue name to which the job belongs. A new field JOBQ is added into the SHOWJOB output format. If the ;JOBQ option is not specified the SHOWJOB output will be the same as before.

Example:

```
:SHOWJOB;JOBQ
                          New field
                          ~~~~~~~~~
JOBNUM   STATE IPRI JLIST   JOBQ       INTRODUCED  JOB NAME

#J3      EXEC       LP      HPSYSJQ    WED 11:46A  FTPMON,FTP.SYS
#J7      EXEC       LP      SYSMGRQ    WED  5:47P  EMG,MGR.SYSMGR
#S81     EXEC       34                 THU 12:17P  MGR.GOPI
```

Overview of HP Patch/iX

by Venugopal
Commercial Systems Division

Overview

The MPE/iX Patch Management tool, HP Patch/iX, was introduced with PowerPatch C.50.05 based on MPE/iX General Release 5.0, and is available with the MPE/iX Release 6.0 Fundamental Operating System (FOS) software.

HP Patch/iX provides a user-friendly environment for customization and installation of Hewlett-Packard supported patches on an MPE system. The new tool provides the functionality of simultaneous installation of a PowerPatch tape with Reactive patches, thus reducing system unavailability associated with patch installations by as much as three hours per system. This downtime eliminates one of the system UPDATE/REBOOTs.

HP Patch/iX also provides qualification of all patches being applied to the system (both PowerPatch tapes and Reactive patches), thus reducing the likelihood of an older patch backing out a previously installed newer patch.

Detailed information on supported processes and comprehensive usage instructions are included in the *HP 3000 MPE/iX System Software Maintenance Manual*.

Benefits and Features

HP Patch/iX allows the following processes:

- Qualification of all patches (Reactive and PowerPatch)
- Simultaneous installation of a PowerPatch tape with Reactive patches
- Customization of patches
- Creation of the patch installation tape while users remain on the system
- Creation of a CSLT only when necessary

HP Patch/iX Supported Processes

HP Patch/iX supports the following installation processes:

- Single Reactive patch
- Multiple Reactive patches from various sources

- PowerPatch tape

- PowerPatch tape with Reactive patches

- SUBSYS with a PowerPatch tape (referred to as an Express)

- SUBSYS with a PowerPatch tape along with Reactive patches

NOTE   HP Patch/iX does not support the UPDATE of a system from one MPE release to another. A system installed with MPE/iX 5.5 cannot be UPDATEd to MPE/iX 6.0 via HP Patch/iX. The system must already be installed with MPE/iX 6.0 if patches are to be applied via Patch/iX. Additionally, HP Patch/iX does not support the addition of a SUBSYS product without a PowerPatch tape (SUBSYS Add-On only), because there are required information files on the PowerPatch tape about the SUBSYS components.

Reflection Users

Reflection is a separately purchasable third-party product that is a PC-based terminal emulator program.

If you want to use Patch/iX from a PC using Reflection software, you must install Reflection with the MPE/iX POSIX option (if available). Otherwise, Patch/iX will not work from your PC.

For more information on the Reflection MPE/iX POSIX option, see the installation instructions for Reflection.

Overview of HP Stage/iX

by Venugopal
Commercial Systems Division

HP Stage/iX is an operating system facility for applying and managing
MPE/iX patches on your system. Using HP Stage/iX reduces system
downtime and provides an easy and reliable method for backing out
patches.Use HP Stage/iX to place PowerPatch and/or reactive patches
into staging areas on disk while the system is up, then choose a staging
area to use at boot time to apply the patches. After the patches are
applied, they can be backed out at any time through a reboot to the
Base (the version applied by the last tape update). Once you are
satisfied with the patches on the running system, you can commit the
staging area to form a new Base while the system is running (no reboot
is needed).

Requirements

HP Stage/iX is targeted for larger HP 3000 systems with adequate
extra disk space. Since putting patches in staging areas requires that
additional files be placed on disk, using HP Stage/iX uses a lot of disk
space. Adequate disk space on LDEV1 is especially critical, since a
large percentage of patches require changes to large files that must be
staged on LDEV1 (`NL.PUB.SYS` is an example).

NOTE   Not all patches are stageable. At the time of release 6.0, Stage/iX
cannot handle patches which involve System configuration files,
MMSAVE.MPEXL.SYS, non-system volume set files, and patches with
files that contain jobs that are streamed during patch installation.
Patches will be identified as stageable or not stageable during the
installation process.

HP Stage/iX Concepts

HP Stage/iX has three interfaces:

1. **HP Patch/iX menu**—allows you to stage patches to staging areas
   once HP Stage/iX is initialized (as well as create CSLT/`STORE` tapes
   as usual). Refer to "Overview of HP Patch/iX" in this chapter for
   information about HP Patch/iX.

2. **STAGEMAN utility**—allows you to manage your HP Stage/iX
   environment, and obtain information about the environment and
   individual staging areas.

3. **STAGEISL utility**—is an ISL utility available when the system is
   down. It contains a subset of the `STAGEMAN` functionality, and allows
   you to recover from most errors or mistakes.

Your operating system resides in what HP Stage/iX refers to as the **Base**. This is the set of files laid down by the last system installation or update (from tape). HP Stage/iX creates and manages **staging areas**, which are file containers on disk that hold versions of files that are different from the Base. (A staging area is actually an HFS directory which holds all the files associated with that staging area.) More than one staging area can exist at a time. Each staging area contains the difference, or delta, between the Base Operating System and a patched OS.

When a staging area is **activated** on the next boot, the files in the staging area's directory are moved (renamed) into their **natural locations**. (For example, the staged version of the NL is moved into `NL.PUB.SYS`.) At the same time, the Base versions of the files are saved into an HP Stage/iX archive directory. When the staging area is backed out (when the system is booted back to the Base), the converse takes place, and the system is restored to its original state.

When an active staging area is **committed** to the Base, the staging area's directory is deleted, and all archived Base files are purged. The files that were switched into their natural locations when the staging area was activated remain there as part of the new Base. This releases any disk space that was used by the staging area.

HP Stage/iX (with the help of HP Patch/iX) allows new patches to be staged and applied in a cumulative fashion. If you create a new staging area while a staging area is active, the new staging area will contain all the changes between the Base and the active staging area, *plus* the new patches applied to the new staging area.

Installing and Initializing HP Stage/iX

HP Stage/iX is part of the Fundamental Operating System (FOS), and is therefore automatically installed when you update to OS version 6.0 or later. To be able to use HP Stage/iX you must:

1. Update or install your system software to version 6.0.

2. Install HP Patch/iX.

    This step is required to manage your qualified patches.

3. Initialize HP Stage/iX using the `STAGEMAN` utility.

    Refer to the *HP 3000 MPE/iX System Software Maintenance Manual Release 6.0* and follow the directions in the manual for updating or installing your system software, and managing patches by staging areas.

STAGEMAN Utility

Using the STAGEMAN utility, you can control and obtain information about your HP Stage/iX environment.

Uninstalling HP Stage/iX

The HP Stage/iX environment can be uninstalled any time there is no active staging area. This action deletes all staging areas and any files and directories that HP Stage/iX has built. This activity does not just turn off HP Stage/iX; it completely removes any and all files associated with the environment.

Safety Features

There are two safety nets available for HP Stage/iX users:

STARTBASE   is an ISL utility which always points to the Base version of START. If your system ever gets into a state where you cannot boot from the last staging area you have SET to be activated, you can enter the STARTBASE command at the ISL prompt to boot your system back to the Base.

STAGEISL   is an ISL utility that provides users with a subset of the STAGEMAN functionality when the system is down. Specifically, it allows you to LIST the available staging areas, get the STATUS of the Stage/iX environment, and SET to either a staging area or the Base.

HP Stage/iX HELP

Help is available for all HP Stage/iX functions from within the HP Patch/iX, STAGEMAN, and STAGEISL facilities. With STAGEMAN or STAGEISL, use the HELP command. With HP Patch/iX, help is available in HP Stage/iX context with the [F1] function key.

Transaction Manager

Overview

Transaction Manager (XM) is an integral part of the MPE/iX file system. It provides a facility to log all the transactions that are done on a file that is attached to XM via a system intrinsic. It ensures data integrity by protecting such files across system crashes. Three new enhancements are added to XM with 6.0 to provide improved performance and reliability. The enhancements are the following:

- System Log Expansion

- Transaction Copy-forward

- Checkpoint Improvement

For more technical information, refer to "Transaction Manager" in Chapter 10, "Technical Articles."

Increase in Number of TCP Connections

by Ganesh HS and Chandrashekar MN
Commercial Systems Division

Introduction

Prior to the MPE/iX 6.0 release, TCP on MPE/iX allowed a maximum of 4096 connections. An enhancement to the TCP module allows up to 5600 connections. The version ID of TCP (NET_TCP_VERS in nmmaint,3) which supports this feature will start with B060.

Configuration through NMMGR

You need to configure the machine to allow it to go up to 5600 connections. The NMMGR screen for configuring the maximum connections looks like the following:

```
NMMGR/3000 (B.06.00) Transmission Control Protocol (TCP) Config  Data:
Y
When Data Flag is "N", press "Save Data" to create the data record.
Command:

Path:  NETXPORT.GPROT.TCP

[N]       Checksum Enabled (Y For Yes, N For No)

[256 ]    Maximum Number of Connections

[2  ]     Retransmission Interval Lower Bound (Secs)
[180  ]   Maximum Time to Wait For Remote Response (Sec)
[10 ]     Initial Retransmission Interval (Secs)
[4  ]     Maximum Retransmissions per Packet

[600  ]   Connection Assurance Interval (Secs)
[3  ]     Maximum Connection Assurance Retransmissions

File:  NMCONFIG.PUB.SYS


                                    Save      Help     Prior
                                    Data      Screen
```

The field, "Maximum number of connections" can now handle up to 5600 connections. Before this enhancement the range was 0 to 4096. To reach this screen, "Transmission Control Protocol (TCP) Config" in NMMGR, do the following:

1. Go to [Open Config]

2. Go to [NS]

3. Go to [Unguided]

4. Go to [Netxport]

5. Go to [Gprot]

6. Go to [Tcp]

After the re-configuration is validated, stop, then restart, the network to implement the increased TCP connection limit. It is not necessary to reboot the machine.Now TCP is ready to accept up to 5600 TCP connections.

Overview of CI Enhancements

by Jeff Vance
Commercial Systems Division

Overview

CI enhancements introduced in MPE/iX Release 6.0 are the following:

- POSIX filenames support additional special characters.

- `COPY` command was enhanced.

- `SHOWVAR` command was enhanced.

- New CI evaluator functions were added: `ANYPARM()`, `BASENAME()`, `DECIMAL()`, `DIRNAME()`, `FQUALIFY()`, `FSYNTAX()`, and `XWORD()`.

Details for these enhancements are found in the article, "6.0 CI Enhancements," in chapter 10, "Technical Articles."

CI enhancements introduced in MPE/iX 5.5 Express 3 (C.55.03) are the following:

- The `LISTF` and `LISTFILE` commands allow users to discover who is accessing files on their system, including remote accessors. `LISTFILE` output can now be filtered by selecting files based on access type or file code.

- Two new CI variables were added: `HPLASTSPID` and `HPSPOOLID`. Both contain the spoolfile ID in the form: *Onnnnnnn*.

- The `FLABELINFO` intrinsic and `FINFO`() function are able to return the number of sectors and extents a file occupies, file creation time, and the number of accessors to a file.

- The `PAUSE` command can now wait for one or more jobs to complete, to change state from waiting to executing, or to begin execution.

- The `INPUT` command allows the number of characters to read from `$STDIN` to be specified.

- The `PRINT` command is now able to display all data in a file, even if this data appears to be line numbers. The new `NONUM` option enables this feature.

Details for these enhancements are found in the article, "CI Enhancements in 6.0," in chapter 10, "Technical Articles."

MPE/iX POSIX Installation Changes

by Jeff Vance
Commercial Systems Division

MPE/iX Release 6.0 has fixed several POSIX install problems as well as adding a few POSIX enhancements. On 6.0, the System Manager should notice that:

- POSIX files have the correct permissions so that all users can run the shell.

- Several new symbolic links are created.

- The original /etc/profile is saved.

- A new file, /etc/profile.local, is defined.

- The "SH" UDC runs the shell without changing the current working directory.

Three files were changed to accomplish these improvements:

- `I0036431.USL.SYS`—the install job for most POSIX files.

- /etc/profile—the shell's environment initialization script.

- `HPPXUDC.PUB.SYS`—the system POSIX UDC file.

For the POSIX programmer, all archive libraries are copied and converted to shared libraries.

File Permissions

The `POSIX` installation job establishes the permissions on many files in the `POSIX` namespace. Prior to Release 6.0, this job caused file permissions to be set such that only users with SM capability could access the POSIX- named files. For example, the shell tries to execute the /etc/profile file; however, in earlier MPE releases, /etc/profile was installed with security that denied execute access to all non-privileged users. This caused the initialization environments to be skipped when the shell was run, and forced system managers to alter the security on many POSIX-related files.

In 6.0, I0036431 installs files with the correct permissions:

- All users have read access to all text files, like /etc/motd, and /usr/lib/terminfo/hp2392a

- All users have execute access to initialization files, like /etc/profile

- All users have traverse permission (execute access in UNIX terms) to directories commonly used by the shell

Of course the system manager can change the file permissions but the default is now correct.

New Symbolic Links:

UNIX users are accustomed to invoking FTP and other networking services and files by their common UNIX filenames. To this end, Release 6.0 has added many new symbolic links. Twelve networking related links are defined:

POSIX Filename	MPE Filename
/bin/ftp	FTP.ARPA.SYS
/bin/telnet	TELNET.ARPA.SYS
/etc/protocols	PROTOCOL.NET.SYS
/etc/inetd	INETD.NET.SYS
/etc/inetd.conf	INETDCNF.NET.SYS
/usr/adm/inetd.sec	INETDSEC.NET.SYS
/etc/bootpd	BOOTPD.SYS
/etc/bootptab	BOOTPTAB.NET.SYS
/etc/resolv.conf	RESLVCNF.NET.SYS
/etc/services	SERVICES.NET.SYS
/etc/hosts	HOSTS.NET.SYS
/usr/bin/remsh	REMSH.NET.SYS

And, several miscellaneous links:

```
/bin/ksh
/bin/cc
/bin/ld
```

Three new device files have been created:

```
/dev/null
/dev/tape
/dev/tty
```

There are traditional MPE ways of reaching all of the above files and services, but these new links make it more familiar to UNIX-trained system managers, without any changes visible to the MPE system administrator.

These new links and device files are defined in the I0036431.HP36431.SUPPORT job file. If a file by the same name already exists it will not be overwritten in this job.

Shell initialization:

/etc/profile is invoked by the POSIX shell whenever the -L info string is supplied, assuming the user running the shell has execute permission to this file. /etc/profile contains global initialization for the shell; whereas, user-specific initialization is accomplished via the user's own .profile file.

The main enhancement is that /etc/profile supports the execution of a custom initialization file named /etc/profile.local. If this file exists it is executed as the final step in /etc/profile. This allows sites to override the behavior of ./etc/profile, thus impacting all POSIX shell users, but managed in a central location. /etc/profile.local will never be modified by POSIX installs or updates. Also, if /etc/profile exists at POSIX install time, meaning at the time that I0036431.HP36431.SUPPORT is streamed, it will be renamed to /etc/profile.bak prior to the creation of this HP supplied version of /etc/profile. This allows you one on-line backup copy of this important shell initialization file.

Additional /etc/profile enhancements:

- All shell variables defined in /etc/profile will default to a CI variable of the same name if one exists. This lets you initialize certain shell variables via LOGON UDCs. The one exception is the PATH shell variable since it is created within the shell program itself prior to the execution of /etc/profile. This prevents detection of a previous PATH variable.

- The MANPATH variable is defined as "/usr/local/man:/usr/man:" if it has not been previously defined.

- The PATH variable is set to "/usr/local/bin:/bin". Specifically, dot (".") has been removed from PATH. This implies that users will need to prefix their local scripts and programs filenames with "./" to execute them.

- The HISTFILE variable, which defines the location of the shell's history file, is defined to be "/your account/your group/.sh_history" if it is not previously defined. If the user does not have a HOME group then their current group name is used; otherwise their HOME group name is used.

- The TERM variable is defined to be "hp2392a" unless it has been defined previously in the CI. If you have defined TERM in the CI as a qualified file name this will not likely work in the shell. You may need to re-define TERM in your own .profile file or in the system's /etc/profile.local file.

- The HOME variable is defined as "/your account/your group" as long as it has not been defined previously in the CI. If the user does not have a home group then their logon group name is used; otherwise their home group name is used.

- The popular "ll" alias is defined as "ls -l".

- The cursor is not homed and the screen is no longer cleared.

- The current working directory (CWD) is not altered (see also the SH UDC changes).

The above changes in /etc/profile should make the shell more user-friendly.

"SH" UDC:

The HPPXUDC.PUB.SYS file contains the "SH" UDC, among other useful POSIX-related UDCs. This UDC file has been in FOS since Release 5.0 but is not cataloged by default. The "SH" UDC has been modified in Release 6.0.

Assuming that an SH command from the CI invokes the SH UDC described here, the shell can be invoked from the CI in a more UNIX-like manner as:

```
:sh -c makefile_6.0
```

which runs the shell (SH.HPBIN.SYS) passing it the name (in POSIX syntax) of a file to execute. In this case the filename passed to the shell is "makefile_6.0" in the current working directory (CWD) and in lowercase.

This is because the "SH" UDC has been simplified to accept any number of parameters and to no longer change the CWD. In prior MPE releases, the SH UDC changed the CWD to the user's home group in order to keep the shell's history file in a single location for individual users. Now the history file is defined by the HISTFILE shell variable, which is defined in /etc/profile.

Users that depended on the SH UDC to change the CWD to their home group will need to issue the shell's "cd" command to establish the CWD in the desired location. For example, to change the CWD to the users' home group the following command can be executed directly from the shell or via the .profile file:

```
cd /$HPACCOUNT/$HPHGROUP
```

This assumes that the user has a home group assigned. See the contents of /etc/profile on how to handle cases where a home group is not defined.

UNIX Porting

Existing archive library (".a") files residing in /lib or /usr/lib are converted to shared library (".sl") files, leaving the original ".a" files intact. This conversion is done in the I0036431.HP36431.SUPPORT job

which is streamed during the update process. The shared library concept is familiar to MPE developers but relatively new in UNIX. The existence of shared libraries helps facilitate most modern UNIX ports.

DNS BIND/iX and Syslog/iX

DNS BIND, which stands for Berkeley Internet Name Domain, is an implementation of the Domain Name System (DNS) and is the most common use of the Domain Name System (DNS). Now, the complete implementation of DNS BIND/iX runs on an MPE/iX shell operation system on MPE/iX release 6.0. Initially, DNS BIND/iX was written for UNIX; however, the latest version of DNS BIND/iX 8.1.1 runs on MPE/iX systems.

Overview of DNS BIND/iX

BIND DNS/iX is a domain name system which consists of a client-server mechanism. The name servers comprise the server half of the DNS's client-server mechanism. The name server is the NM program `NAMED` which maintains information about some part of the DNS called a zone and has capabilities to retrieve information regarding other zones. The clients are resolver routines provided as NMRL libraries. The resolvers are clients that query the name servers, interpret the responses and send the answers to the requester.

BIND DNS/iX makes your domain names visible to the internet as well as handling client requests to resolve domain names within your domain and external domains. Prior to DNS BIND/iX, the HP3000 users had to rely on other machines or other operating system to host their organization DNS information. Now, you can host it on HP3000 systems.

Overview of Syslog/iX

Syslog is the standard event logging system for Syslog/iX UNIX, now the Syslog/iX can run on the MPE/iX shell operating system on MPE/iX release 6.0. With the features of Syslog/iX available on MPE/iX systems, the event messages can be logged to files, terminal devices, or even forward to other syslog systems. Syslog/iX can accept data from the local system via an AF_UNIX socket or from any system on the network via an AF_INET UDP socket on port 514. DNS BIND/iX uses Syslog/iX as the event logging subsystem.

Easytime/XL Bundled with MPE/iX 6.0

by Gopalakrishnan & Kumaran,
Commercial Systems Division

Easytime/XL is a software tool that allows you to manage your system from easy-to-use screens. You will be able to perform most of the tasks for managing your system from the HP Easytime/XL screens.

With this release, HP Easytime/XL is bundled with Fundamental Operating System and it is supported.

What Is HP Easytime/XL

HP Easytime/XL is an easy-to-use program through which you can communicate with your HP 3000 computer. Based on a system of pull-down menus, HP Easytime/XL guides you with descriptive phrases through the day-to-day tasks that you need to do your job, whether that means managing system resources, starting a job, printing a report, or sending a message to another user on the system.

Whether you are the person who manages the system or someone who only uses applications on it, HP Easytime/XL provides easy methods or several tasks. The system administrator manages the system in many ways such as creating accounts and users, setting passwords, establishing schedules for backups, and replying to requests. Some of these tasks can be done through HP Easytime/XL.

For example, HP Easytime/XL screens provide ways to inform users of backup schedules, a system administration task. The screens also provide ways for all users to send messages to each other, or to the person operating the console. HP Easytime/XL helps in performing both system administration tasks and end-user tasks.

Here are some of the tasks that you can perform with HP Easytime/XL:

- print, copy, rename, delete, and store files

- review and manage files waiting to be printed

- start and delete jobs

- show and delete sessions

- send messages to other users

- review messages and reply to requests from the system

- perform full and partial backups

- access an extensive on-line Help Facility

For more information, refer to *HP Easytime/XL Quick Reference* and *HP Easytime/XL User's Guide*.

4   System Support Tools

This chapter contains the following articles related to MPE/iX system support tools:

- Predictive Support
- On-Line Diagnostic System (Sherlock)
- GLANCE has a New Name and a New Home!

Predictive Support

Product Overview

HP Predictive Support provides hardware support and helps increase the uptime of your systems by monitoring system memory and disk/tape drives.

When the HP Predictive Support software detects a potential problem, it sends a message to the HP Response Center. The Response Center portion of the system screens the data and forwards problems requiring further analysis to a Response Center Engineer. If action is needed at your site, the Response Center Engineer and the account Customer Engineer will work with you until the problem is resolved.

This proactive hardware support is provided as part of the HP Hardware Support Agreement.

Predictive Support Understands the Year 2000

The Predictive Support software has been modified to correctly handle dates in the Year 2000 including correctly displaying the dates as appropriate and being able to understand code words that expire in the year 2000 or beyond.

Upgrading Predictive Support Software

The HP Predictive Support software is upgraded each time you install a new version of the MPE Operating System. The Predictive Support Monitor (PSMON) runs as a system process so this program file and its message catalog file is on the SLT. The rest of the files needed to make up the Predictive Support software is on the FOS tape and is automatically updated by using the `AUTOINST` process of updating the operating system software.

Support for New Peripherals

HP Predictive Support now supports the following new peripherals:

- HPC2257A Cascade 2 Way Disk
- HPC2257B Cascade 2 Way Stripped Disk
- HPC2257HA High Availability Cascade 2 Way Disk
- HPA2958A 1 GB 3.5 in. SCSI Disk
- HPA3304A 2 GB 3.5 in. SCSI Disk
- HPA3351A 2 GB 3.5 in. SCSI Disk

- HPA3352A 4 GB 3.5 in. SCSI Disk

- HPA3353A 4 GB 3.5 in. Fast Wide SCSI Disk

- HPA3306A 2 GB 3.5 in. Fast Wide SCSI Disk

- HPA3628A 9 GB 3.5 in. SCSI Disk

- HPA3629A 9 GB 3.5 in. Fast Wide SCSI Disk

- Quantum DLT4000 DAT Drive

- HPC1533B DAT Drive

- HPC1537A DAT Drive

Before You Update HP Predictive Support

Before you install the update, you should run HP Predictive Support to ensure that any data indicating potential problems is transmitted to the HP Response Center.

If you have an HP Hardware Support Agreement, and wish to have HP Predictive Support activated, please contact Hewlett-Packard and a Customer Engineer will perform the initial configuration and activation.

On-Line Diagnostic System (Sherlock)

by Mariano Nunez
Commercial Systems Division

The Sherlock diagnostic system is made up of a user interface program that provides access to the diagnostic programs, a diagnostic system monitor that provides control and logging functions, and a diagnostic system library (XL.DIAG.SYS) that provides a standard interface for all diagnostic program developers.

The Sherlock system will add support for the 997 system. This will also include support for the 4X memory for all 9x9 systems.

The following diagnostic or utility programs will support the products listed below for the MPE/iX 6.0 release:

```
  PRODUCT              DIAGNOSTIC NAME         DIAG VERSION

997                 LOGTOOL's LAYMSG            N/A

997                 Sherlock Library           N/A

XWING SCSI Disks    SCSIDSK2                  B.40.00

JEDI SCSI Disks     SCSIDSK2 (pre-enabled)    B.40.00

100 VG Lan          VGPBA                     A.55.01

Terminals           TERMDSM                   C.02.00
```

GLANCE has a New Name and a New Home!

by James Overman
CSBU R&D

Overview

As of the MPE/iX 6.0 Release, the performance tool, HP GlancePlus/XL, has been renamed HP GlancePlus/iX. With version HPB1787B.09.06, the program file has been changed from `GLANCEXL.PUB.SYS` to `GLANCEIX.PUB.SYS`. For most users, these changes should have no impact, as the `GLANCE.PUB.SYS` command execute file has been changed to run the new program name. Users can continue to employ the GLANCE command to access the tool as before. If users have used the GLANCEXL program name in UDCs, they will need to either change the references, use a logon file equate, or copy the GLANCEIX program to GLANCEXL.

Several areas in Glance were changed to be more aware of the Hierarchical File System (HFS or POSIX) naming convention. Most noticeably, the Use command prompt now reflects the alternative file names allowed (MPE or HFS). Also, the File display portion of the Process Display screen has been rearranged to display more of the longer HFS file names before truncation occurs. Other changes to Glance include the display of the duration of the last data collection interval, some general performance optimizations, and changes necessary for MPE/iX 6.0 internal structural modifications.

Trial Program

The Customer Trial Demonstration capability will continue to be available and to provide a one-time-only trial period for the use of the Glance performance tool. Thus, all MPE systems come with a GLANCEIX program that may be activated for the trial use. The `GLREADME.PUB.SYS` file describes how to activate the trial period. The trial software no longer resides in a separate group as in previous releases. The trial program now resides at the same location as the purchased product so that two differing versions of the program may not be present on the system at the same time. Customers who have purchased Glance will automatically have the trial version overlaid by the full capability product. Those customers who have not purchased Glance should leave the product files in PUB.SYS for use by the HP Response Center is resolving problem calls.

The features described above are also available as patch GLAKX69 for MPE/iX 5.0 and 5.5 but without the program renaming. Note that there are unique versions of the Glance program for each operating system version and they may only be utilized on the appropriate platform.

5   Application Development

This chapter contains the following overview articles related to new applications and enhancements to existing applications:

- Transact V and Transact/iX Enhancements

- Dictionary /3000

- Inform/V

- HP ALLBASE/BRW

- COBOL II/iX Enhancements

- QUERY/iX Enhancements Overview

- IMAGE/SQL and TurboIMAGE/XL Overview

- Features in ALLBASE/SQL Version G3

Additional technical information on many of these topics is in Chapter 10, "Technical Articles."

Transact V and Transact/iX Enhancements

by Kelly Sznaider
Support Technology Lab

Overview

Two new enhancements have been added to Transact/V as of version
A.11.00 and also to Transact/iX as of version A.06.00:

- Literal strings and numeric constants can now be passed as
  parameters in the PROC verb.

- Transact now provides a method for increased decimal precision in
  arithmetic expressions.

New Parameter Types

The parameter list for the PROC statement has two additional types of
parameters. A programmer-defined character string can be passed by
placing the string within quotes, similar to other languages. The
default is to pass this literal as a 16-bit address. To change the literal to
a byte address, precede the string with a "%", the same as with
variables. The other parameter type is to pass a numeric value. The
numeric constant is passed by value.

Examples:

```
PROC ASCII (65, 10, %"abc", &(numchar));
PROC FWRITE (#FILEID(TRANOUT), "Hello", -5, 0);
```

Increased Decimal Precision

The second Transact enhancement increases decimal precision in
arithmetic expressions (`IF`, `LET`, `REPEAT`, and `WHILE` verbs) via the
new `!PRECISION` compiler command. A specified minimum number of
decimal places is maintained for all packed-decimal intermediate
results when calculating the value of an arithmetic expression. This
minimum number is used only if the expression's default evaluation
method would have maintained fewer decimal places.

Example:

```
!PRECISION(5) maintains at least five decimal places.
```

Documentation

The detailed information for these enhancements has been included in
the *HP Transact Documentation Update Notice*, which is also included
in the latest edition of the *HP Transact Reference Manual*. A revised *HP
Transact Quick Reference Guide* is also available.

Dictionary/3000

by Kelly Sznaider
Support Technology Lab

Overview

As of version A.03.00, Dictionary/3000 offers two new enhancements in the DICTDBM utility:

- The capability of copying entities with the new COPY command.

- The ability to enter elements multiple times within an HP Inform group.

COPY Command for Entities

For the COPY command, a category, class, element, file, group, location, or procedure is copied to a new like entity with a different name. The entity's associated child entities and/or its related entities can also be copied if desired. This enhancement adds productivity improvements for the user.

Multiple Entries for HP Inform

The same element can now be added to an HP Inform group more than once, as long as the physical file designation or the parent file is different making the record unique. This makes elements in HP Inform groups easier to use and more straightforward. (Inform/V version A.11.00 or later is needed with this enhancement.)

Documentation

The detailed information for these enhancements has been included in the *Dictionary/3000 Documentation Update Notice* which is also included in the latest edition of the *Dictionary/3000 Reference Manual*.

Inform/V

by Kelly Sznaider
Support Technology Lab

Inform/V has been enhanced to support the Dictionary/3000 version A.03.00 enhancements. As of Version A.11.01, Inform/V is capable of handling multiple items with the same primary name in an HP Inform group. The alias names for the same primary item must be unique.

The Dictionary/3000 enhancement allows the definition of duplicate primary items in an HP Inform group within the dictionary. Whereas the Inform/V enhancement allows these group items to be selected and reported on as defined by the dictionary. Recompilation of Inform reports is not needed for this Inform/V release.

HP ALLBASE/BRW

by Kelly Sznaider
Support Technology Lab

The last few revisions of HP ALLBASE/BRW have concentrated on the integration of TurboIMAGE Third Party Indexing (TPI) with BRW. BRW version A.01.54 introduced the support of TPI. In BRW version A.01.55, more TPI enhancements were provided, as well as several defect repairs. In the latest version of BRW, A.01.56, the main focus is a TPI defect repair.

This version of HP ALLBASE/BRW, A.01.56, fixes a TPI problem that was introduced in version A.01.55. When using wildcard characters for the Selection Set, the result shows that no records are found when data actually exists. The TPI tracing mechanism shows the incorrect `DBFIND` mode being used.

The other repair for version A.01.56, not related to TPI, corrects the continuation spoolfile number when a report is split between multiple spoolfiles.

In the previous version of HP ALLBASE/BRW, A.01.55, three enhancements and miscellaneous defect repairs were included. Among these enhancements were two new JCWs: `BRWSQLTIMEOU`T and `BRWTPITRACE`. `BRWSQLTIMEOUT` is given a value that is the number of minutes the BRW SELECT statement waits before timing out. A special value of 65535 causes BRW to wait indefinitely. `BRWTPITRACE` displays the TPI tracing information to $STDLIST. The third enhancement in A.01.55 was to enhance the TPI selection algorithm. If more than one BRW condition matches a TPI key, the BRW condition that appears first in the compiler listing is used. If two TPI keys start with the same BRW condition, the longer key is used.

COBOL II/iX Enhancements

by Walter Murray
Support Technology Lab

Overview

COBOL II/iX on MPE/iX 6.0 provides several enhancements that were requested by Interex SIGCOBOL. Some of these enhancements have already been made available in various patches to MPE/iX 5.5.

This article gives an overview. For details, refer to "COBOL II/iX Enhancements" in chapter 10, "Technical Articles."

1. COBOL now provides the ability to specify secondary entry points in a main program. The ENTRY statement, which was previously allowed only in subprograms, may now be used in the main program. The nonnumeric literal specified in the ENTRY statement becomes the name of a secondary entry point for the program, and it can be specified in the MPE/iX RUN command.

2. Support is now provided for procedure call by plabel. In a CALL statement, the identifier designating the subprogram to be called may now be the name of a numeric data item. It will be interpreted as a procedure label of a previously loaded procedure, and the compiler will generate a dynamic call to that procedure.

3. Six new procedures have been added to the run-time library to simplify bit manipulation in COBOL.

4. Compiler limits were increased to permit compiling much larger programs, such as those created by automated code generation tools. COBOL II/iX should now be able to compile programs in excess of 200,000 lines of source code.

5. An index-name may now be used as the operand of a DISPLAY statement.

6. Attempting to compile a source file in Qedit format, on a system where Qedit is not properly installed, will now result in a specific error message suggesting a problem related to Qedit. Qedit is a full-screen text editor from Robelle Consulting Ltd.

QUERY/iX Enhancements Overview

by James Overman
SSG Group R&D

Overview

The following are the new enhancements for QUERY/iX:

- B-Tree Enhancement

- Dynamic Dataset Capacities

- Wildcard Retrievals

- Wildcard Character Limitations

- B-Tree Limitations

- "NO MATCH" Enhancement

With the HP32216D.03.11 and N.03.11 releases of QUERY, the new B-Tree access capabilities will be supported within QUERY on systems with the B-Tree software release. QUERY is fully backward compatible with the prior releases of TurboIMAGE/XL and QUERY.

When a dataset has utilized the Dynamic Expansion feature, QUERY will display the `CURRENT, MAXIMUM,` and `INITIAL` capacities and the `INCREMENT` amount. The `FORM SETS` output has been slightly modified such that the heading reflects that the Current capacity is being reported rather than the Maximum capacity.

QUERY will automatically invoke the B-Tree retrieval feature for the `FIND, LIST, SUBSET,` and `MULTIFIND` commands when a wildcard character is used in the search value.

Some wildcard characters are used by QUERY for syntax scanning and are limited in their usage. These special characters may be utilized when the data values are delimited by double quotes (for example, tilde, carat, comma, and parenthesis).

QUERY will utilize the B-Tree functionality when possible for both character and binary data types. The wildcard feature is only implemented by TurboIMAGE/XL for character data types (X and U) and not for binary data types. QUERY will continue to use the older access methods (normally serial reads) for Not Equal retrievals. Note that the LIST command always uses a serial read but will apply the wildcard search character for relations.

When a database has the `BTREEMODE1` flag set to `OFF`, QUERY will not attempt to use the B-Tree functionality for Wildcard retrievals. This condition is reported by all forms of the `FORM` command.

The `FIND`, `MULTIFIND`, and `SUBSET` commands have been enhanced to accept a `NO MATCH` alternative such that all data records `NOT MATCHING` the pattern provided are returned by the commands.

For more information, refer to "QUERY/iX Enhancements" in chapter 10, "Technical Articles."

IMAGE/SQL and TurboIMAGE/XL Overview

by Bharati Desai
Commercial Systems Division

Overview

This is an overview of the enhancements to IMAGE/SQL and TurboIMAGE/XL that have been implemented since the release MPE/iX 5.5 (C.55.00). For additional information, refer to the following articles in Chapter 10, "Technical Articles":

- IMAGE/SQL with TurboIMAGE/XL Enhancements
- TurboIMAGE/XL Dynamic Detail Dataset Expansion (DDX) Problem
- IMAGE/SQL Dynamic Dataset Expansion for Master Datasets (MDX)
- New Features in ALLBASE/SQL Version G3
- Introducing ODBCLink/SE

The highlights of this release of IMAGE/SQL, version B.G2.07, and TurboIMAGE/XL, version C.07.14 are the following:

- B-Tree indices enhancement
- ODBCLink/SE, 16-bit and 32-bit ODBC driver, bundled with IMAGE/SQL
- Support for third-party composite indices
- Scalability
- IMAGE/SQL Dynamic Dataset Expansion for Master Datasets (MDX)
- Additional manuals bundled with IMAGE/SQL
- Manuals available on HP Instant Information CD-ROM:
  - *HP IMAGE/SQL Administration Guide*
  - *TurboIMAGE/XL Database Management System Reference Manual*

Updating from an Earlier Release

If you are updating from an earlier release of IMAGE/SQL, you must `DETACH` databases which are `ATTACH`ed to one or more DBEnvironments and `ATTACH` again in order to benefit from the enhancements. For performance improvement, issue the `UPDATE`

STATISTICS command of ISQL for each of the tables you access frequently. If you have created SQL data using an earlier release of IMAGE/SQL, perform the ALLBASE/SQL migration to migrate your DBEnvironment to the current G3 format. For more information, refer to the *ALLBASE/SQL Database Administration Guide*.

TurboIMAGE/XL is enhanced to increase the concurrency of modification intrinsics, DBPUT, DBDELETE, and DBUPDATE (Critical Item Update feature ON). To use this enhancement, your database must be activated to use the Dependency Semaphore. To achieve this, the ENABLE and DISABLE commands of DBUTIL introduce a new option, DSEM. The default for DSEM is DISABLED.

Important Information

1. The default for Critical Item Update option is now changed from DISALLOWED to ALLOWED.

2. A jumbo data set cannot be enabled for dynamic expansion.

3. If you have third-party indexing (TPI) enabled for the database and want to create B-Tree indices, obtain the correct version of the third-party software which recognizes this coexistence. Otherwise, you will get run-time errors returned by third-party software for new modes of DBFIND, DBINFO, and other utility commands.

ODBCLink/SE

IMAGE/SQL is bundled with ODBCLink/SE, which is a replacement for HP PC API. ODBCLink/SE is an implementation of Microsoft's Open Database Connectivity (ODBC) interface that enables Microsoft Windows based applications and tools to access TurboIMAGE/XL data on the HP 3000 in a client/server environment.

HP PC API will no longer be distributed with IMAGE/SQL software. It can co-exist with ODBCLink/SE but will no longer be enhanced. For more information, refer to the article, "Introducing ODBCLink/SE," in this *Communicator 3000* and the manual, *ODBCLink/SE Reference Manual*, which is bundled with IMAGE/SQL.

Features in ALLBASE/SQL Version G3

by Doug Myers
Database Lab, Commercial System Division

Overview

This is an overview of the features of ALLBASE/SQL version G3 on MPE/iX 6.0 release (C.60.00):

- ALLBASE/SQL manual set on is on Instant Information CD-ROM

- String functions, a new feature for ALLBASE/SQL G3

- Security features, introduced with ALLBASE/SQL G2

- Year 2000 compatibility, introduced with ALLBASE/SQL G2

- Manuals updated

- CAST function, introduced with G1.15

Instant Information CD-ROM

The entire manual set for ALLBASE/SQL is now available on the new CD-ROM documentation system, "Instant Information." You can now look up information quickly in the on-line documentation.

String Functions

The supported SQL syntax has been enhanced to include the following string manipulation functions: UPPER, LOWER, POSITION, INSTR, TRIM, LTRIM and RTRIM. These string functions allow you to manipulate or examine the CHAR and VARCHAR values within the SQL syntax, allowing for more sophisticated queries and data manipulation commands to be formed. The specifications for each of these functions is described in "New Features in ALLBASE/SQL Version G3" in Chapter 10, "Technical Articles."

MONITOR and INSTALL Authorities

Two new authorities were added in the G2 release of ALLBASE/SQL: MONITOR and INSTALL. These authorities can be granted or revoked to users. MONITOR authority gives the ability to run SQLMON, an on-line diagnostic tool that monitors the activity of an ALLBASE/SQL DBEnvironment.

INSTALL authority gives users the ability to use the SQL preprocessors to create an SQL module owned by someone other than themselves, or to use ISQL to install a module owned by someone other than

themselves. This is an important capability for those needing to move modules from a development DBEnvironment to a production DBEnvironment.

Year 2000 Compatibility

ALLBASE/SQL has always stored dates in a four-digit year format which largely eliminates the year 2000 compatibility problem. ALLBASE/SQL applications can be coded using four-digit year dates. However, for existing applications that may allow dates to be entered using a two-digit year format, there is still an issue as to how these dates get converted to the ALLBASE/SQL internal format using the `TO_DATE` function or `TO_DATETIME`. For an explanation on how these are handled, refer to the article, "New Features in ALLBASE/SQL Version G3," in Chapter 10, "Technical Articles."

CAST Function

The `CAST` function enhancement for ALLBASE/SQL and IMAGE/SQL became available to customers with the G1.15 release. The `CAST` function is used to explicitly convert data from one data type to another.

6 Data/Application Integration

This chapter contains the following article related to Data/Application Integration:

• DCS RPC Clients for FOS through MPE/iX

DCE RPC Clients for FOS through MPE/iX

by Sridhar Murthy T.R.
Commercial Systems Division

Overview

For the MPE/iX 6.0 release and later, the DCE RPC Client (Version A.01.12) modules, will be included in the FOS. These DCE RPC modules are based on OSF DCE 1.0.2. These changes will support customers who were continuing to be dependent on NCS. NCS has been obsoleted from 6.0 release.

A DCE/NCS application is developed by first building an idl source file, for example foo.idl for DCE,and foo.nidl for NCS. When compiled with the idl compiler, the idl source file will create Server Client Stubs and an .h file (DCE Stubs in the case of DCE idl and NSStubs for NCS idl.) The DCE/NCS application must be linked with these stubs.

MPE/iX 6.0 release contains the utility NIDL2IDL which converts an existing template file (for example, foo.nidl) to a DCE template(foo.idl.) This application using the NCS calls must be ported to use the appropriate DCE calls. Refer to NCS 1.5.1 in the *DCE RPC Transition Guide* for more information.

NOTE
CDS client on 6.0 will be available in a later release.

The following two products have been discontinued from 6.0.

B3824AA DCE/3000 CDS Server License
B3825AA CDE/3000 Security Server License

Technical Details

In order for the RPC library calls to succeed, the RPC daemon RPCD must be running. This daemon can be started in two ways, from the Command Interpreter or from the POSIX Shell.

To start the daemon from the MPE CI, enter the command:

```
:RPCD.PUB.SYS
```

This command file will build a job file and stream it to create the RPCD daemon.

To start the daemon from the POSIX Shell, enter the following commands:

```
$ export PATH=$PATH:/usr/bin
```

```
$ rpcd
```

These commands will execute the script /usr/bin/rpcd which will invoke the command file `'RPCD.PUB.SYS'`.

The customer needs to run his applications as:

```
:Run foo;xl="dcexl.hpdce.sys othdxl.threads.sys"
```

OR

Link this program as follows:

```
:linkedit altprog foo.pub.sys;xl='dcexl.hpdce.sys,othdxl.threads.sys'
```

Then run foo as a executable.

<table>
<tr><td>NOTE</td><td>For more information on application development, refer to the section, "Building DCE Programs," in the document, DCE for the HP 3000 (Second Edition.)</td></tr>
</table>

The following binaries are available as part of this submittal:

GETIP   Executable which gets a host's IP address.

HOSTNAME   Gets the host name.

IDL   The IDL compiler.

IPCS   Utility to get the status of IPC (Inter Process Communication) resource usage status.

LIBDCE   Archive library of DCE modules.

NSLOOKUP   Utility to get the name and IP address of a host. It also gives name and IP adress of DNS Server.

NIDL2IDL   Cross Compiler which will convert the existing NIDL source file to IDL compatible source.

OTHDXL   Contains the PASCAL implementation of the pthread intrinsic calls.

PS   Process/Session Status

RPCCP   Control Program executable to manage end point mapping.

RPCD   Script create a job file and streams it to run rpcd daemon

WHOAMI   Command to get user name.

7   Interoperability and Internet Solutions

This chapter contains the following articles on Interoperability and Internet solutions:

- Introducing 100VG-AnyLAN Network Adapter
- Introducing 100Base-T Network Adapter
- Introducing the REMSH Client functionality
- Hewlett-Packard NetWorker ClientPak for MPE/iX
- Introducing ODBCLink/SE
- DTS/TIO Subsystem
- Telnet/iX Server--Full Functionality Release
- FTP Enhancements
- Samba/iX Overview
- Introducing Java/iX

Introducing 100VG-AnyLAN Network Adapter

by Randy Pawell
Commercial Systems Division

Overview

100VG-AnyLAN Network Adapter (B5426AA) is a high-speed network link providing HP 3000 Series 900 systems that have HP-PB (HP Precision Bus, also known as NIO) backplanes, with a connection to a 100VG local area network. The 100VG adapter connects to the 100 Mb/s (Megabits per second) network through any 100VG hub or switch meeting the IEEE 802.12 standards.

100VG-AnyLAN takes advantage of collision-avoidance and noise-immunity characteristics of the 100VG protocol to provide high network reliability. It provides an evolutionary path for HP 3000 customers needing increased throughput over their existing 10 Mb LAN links.

100VG-AnyLAN Network Adapter is configured and managed the same way as current ThinLAN, Token Ring, and FDDI 3000/iX link products.

100VG-AnyLAN Network Link includes the 100VG network adapter, 100 Mb driver components, and hardware product manuals. TCP/IP Transport is already included with the HP 3000/iX fundamental operating system (FOS).

Product Features

The product features include:

- Supports ARPA, NS, NetIPC, and a subset of 4.3 BSD Sockets over TCP/UDP/IP.

- Supports current Netware and Appletalk products.

- Supports DTC-based terminals, printers, and X-25, when the 10 Mb DTC is attached to the 100 Mb network via suitable 10:100 speed-matching bridge hardware.

- Integrated with the same system tools as ThinLAN, Token Ring, and FDDI 3000/iX.

- VGPBA on-line diagnostics provides status information and limited hardware testing.

- Software configuration, speed, duplex, and autonegotation settings selectable with NMMGR.

- Is compatible with the IEEE 802.12 standard.

- Provides two traffic priorities: normal and demand.

- Provides 100 Mb high-speed connectivity over CAT-5 or CAT-3 voice-grade unshielded twisted pair (UTP), RJ-45 terminated cabling.

- Provides 10 Mb low-speed connectivity compatible with existing 10Base-T UTP networks.

- Supports a maximum cable length of 100 meters in 100 Mb mode.

Introducing 100Base-T Network Adapter

by Randy Pawell
Commercial Systems Division

Overview

100Base-T Network Adapter (B5427AA) is a high-speed network link
providing HP3000 Series 900 systems that have HP-PB (HP Precision
Bus, also known as NIO) backplanes, with a connection to a
100Base-TX local area network. The 100Base-TX adapter attaches to a
100 Mb/s (Megabits per second) network through any 100Base-TX hub
or switch meeting the IEEE 802.3u standards.

100Base-T Network Adapter provides an evolutionary path for HP 3000
customers needing increased throughput over their existing 10 Mb
LAN links.

100VG-AnyLAN Network Adapter is configured and managed the same
way as current ThinLAN, Token Ring, and FDDI 3000/iX link products

100Base-T Network Adapter includes the 100Base-TX network adapter,
100 Mb driver components, and hardware product manuals. TCP/IP
Transport is already included with the HP 3000/iX fundamental
operating system (FOS).

Product Features

The product features include:

- Supports ARPA, NS, NetIPC, and a subset of 4.3 BSD Sockets over
  TCP/UDP/IP.

- Supports current Netware and Appletalk products.

- Support DTC-based terminals, printers, and X.25, when the 10 Mb
  DTC is attached to the 100 Mb network via suitable 10:100
  speed-matching bridge hardware.

- Integrated with the same system tools as ThinLAN, Token Ring, and
  FDDI 3000/iX.

- VGPBA on-line diagnostics provides status information and limited
  hardware testing.

- Software configuration, speed, duplex, and autonegotation settings
  selectable with NMMGR.

- Compatible with the IEEE 803.2u standard.

- Provides 100 Mb high-speed connectivity over CAT-5 unshielded
  twisted pair (UTP), RJ-45 terminated cabling.

- Provides 10 Mb low-speed connectivity compatible with existing 10Base-T UTP networks, also with CAT-5 cabling.

- Supports a maximum cable length of 100 meters in 100 Mb mode.

Introducing the REMSH Client Functionality

by Cas Caswell
Commercial Systems Division

Overview

Remsh allows a user to connect to the remote system and execute a command on that remote system. Output from the remote command is sent to remsh's standard output so the user can see the results of their command.

Remsh is the same service as rsh on BSD Unix systems. The name was changed due to a conflict with the existing command rsh (restricted shell) on System V Unix systems.

Using the REMSH Client

The remsh client allows an HP3000 "client" system connect to a remote "server" system and execute a single command on that system. The server can be any system connected to the network that has implemented the remshd daemon using TCP/IP. The user may run the remsh client from either the POSIX shell or from the MPE/iX prompt.

REMSH Client Requirements

The REMSH Client requires version C.60.00 or C.55.00 Power Patch 3 or later of the MPE/iX Fundamental Operating System and a supported TCP/IP network link.

To install the REMSH Client, the system manager must update the file `SERVICES.NET.SYS`. `SERVICES.NET.SYS` is the Internet Services file that associates an official service name and alias with the port number and protocol the service uses.

Documentation

Documentation for the REMSH Client can be found in the *Configuring and Managing MPE/iX Internet Services* that is shipped with the MPE/iX FOS. Functionality details of the REMSH Client are in this communicator.

Hewlett-Packard NetWorker ClientPak for MPE/iX

by Lalitha Pejavar and Alex Early
Commercial Systems Division

Overview

Available now for HP 3000 users is NetWorker from Legato, a client/server application that provides advanced storage management capabilities to a heterogeneous environment of servers and desktop computers.

Benefits

These are the key benefits of using NetWorker:

- Complete Solution for HP Environments

  NetWorker is the most widely-implemented backup utility in open system environments. It provides HP 3000 users with better integration in a multiplatform environment, allowing them to back up data to an UNIX, NT, or Netware server.

- Support for HP's DLT Library System

  NetWorker is fully compatible with HP's Advanced DLT Library System enabling all server and desktop data to be automatically backed up, indexed, and policy-managed for either user-based file recovery or complete disaster recovery.

- Centralized Enterprise Management

  NetWorker's client/server architecture and centralized storage model allow integration into network management frameworks. Legato NetWorker integration modules allows SNMP-based administration of NetWorker servers using HP OpenView, or complete enterprise administration using HP OpenView IT/Operations.

- Superior Protection for On-Line Databases

  NetWorker supports on-line backup and restore operations for Oracle, Informix, and application data of SAP R/3, Oracle Financials and PeopleSoft HRMS.

Features

These are the key features of the NetWorker ClientPak for MPE/iX:

- Provides an easy-to-operate graphical user interface and maintains a file index and media index to track data backed up and recovered.

- Supports a wide range of industry tape (DAT) and DLT libraries and furnishes a sophisticated media manager which allows for unattended backup and restore using autochange library.

- Provides for concurrent device support for backup and recovery operations using up to 16 backup devices simultaneously.

- Utilizes most of the features of 7x24 True-Online Backup since NetWorker uses TurboSTORE/iX 7x24 for backup. However, it will also work with FOS STORE and TurboSTORE/iX.

- Supports automatic backups of MPE volume sets. Also provides the ability to run two or more scheduled backups in parallel.

- Provides an optional Legato NetWorker Archive Application which gives the ability to take a snapshot of finished project files, directories, or MPE volume sets residing on primary media and safely stored on removable media.

Required Hardware and Software

Required hardware:

- HP 3000 any Series 9xx as a client

- HP 9000 server or Net Server

- DLT auto-library or DAT auto-changer

  - HP DLT 4/48, 2/28

  - STK 9714, 9710

- Networking connection (that is, FDDI, LAN, or 100VG-ANY LAN; 100VG ANY-LAN or FDDI LAN recommended)

Required software:

- NetWorker ClientPak for MPE/iX

- NetWorker Server software (`LEGNETWR`), HP-UX Network Edition

- Auto-library license for the server (`LEGAUT0C`)

NetWorker ClientPak for MPE/iX is supported on MPE/iX 5.5 with patch MPEKX21 (or supersede) and subsequent releases of all HP MPE/iX platforms.

HP Response Center Support

The NetWorker ClientPak for MPE/iX currently has software supported through Legato, and NetWorker server software (`LEGNETWR`) has software support available through HP or Legato.

Legato NetWorker is the only third-party storage management solution supported by HP Worldwide Customer Support Operations. HP Response Center engineers have been fully trained on NetWorker server operations. HP support contracts are available for NetWorker from your HP sales representative.

Documentation

The Hewlett-Packard Network ClientPak Kit (B5475AA) contains the Legato Network ClientPak software, *Legato NetWorker ClientPak for MPE/iX Installation Guide*, and *Legato NetWorker ClientPak for MPE/iX User's Guide*.

Introducing ODBCLink/SE

by Dawna Whitehead
M. B. Foster Associates Limited

Overview

ODBCLink/SE is a replacement for HP PC/API. It is an implementation of Microsoft's Open Database Connectivity (ODBC) interface that enables Microsoft Windows based applications and tools to access IMAGE/SQL on the HP 3000 and ALLBASE/SQL on the HP 3000 and HP 9000, in a client/server environment. ODBCLink/SE is a part of IMAGE/SQL and ALLBASE/SQL.

Application developers and end-users can take advantage of the PC's graphical user interface (GUI) and processing power, while relying on the security, integrity, and database management capabilities of ALLBASE/SQL and IMAGE/SQL.

ODBCLink/SE is a "Special Edition" of M. B. Foster Associates Limited ODBCLink option of DataExpress.

ODBCLink/SE is 95 percent ODBC Level 2 compliant.

Requirements

The networked HP 3000 Series 9xx with ALLBASE/SQL and IMAGE/SQL (or the HP 9000 Series 7xx or 8xx with ALLBASE/SQL) provide the relational database environment on the server.

ODBCLink/SE server runs under MPE/iX 5.0 or higher on the HP 3000 server, or under HP-UX Versions 9.x and 10.x on the HP 9000 Series 7xx and 8xx servers. The client side runs under Microsoft Windows 3.1 and 3.11, Windows95, or WindowsNT V3.51 and V4.0. Connection via Winsock is available in both the 16-bit and the new 32-bit version.

The driver can be used in two ways: by direct calls from a Windows program, such as Foxbase, or through an ODBC-compliant application, such as Microsoft Access or Visual Basic.

Utilities

A data migration tool is provided to migrate data sources created for HP PC API to ODBCLink/SE. Once the data sources are identified, the translation is performed automatically. HP PC/API with Gupta can co-exist with ODBCLink/SE, but it will no longer be supported, and will not be distributed in MPE 6.0 and subsequent releases.

Utilities are provided with ODBCLink/SE to verify that the driver is functioning correctly and to delete unwanted drivers or translation DLL's.

For more detailed information, refer to "Introducing ODBCLink/SE" in Chapter 10, "Technical Articles."

DTS/TIO Subsystem

by Wendy Cheng
Commercial Systems Division

Overview

The Datacommunications and Terminal Subsystem (DTS) consists of host (HP 3000) based software and the DTC (Datacommunications and Terminal Controller) terminal server.

The DTC family of products supports X.25 communication and Telnet access to HP 3000 systems, and TCP/IP communication to HP 9000 and non-HP systems.

The DTS allows terminal, PCs in terminal emulation mode, serial printer, and other asynchronous devices to communicate with LAN-based hosts in HP-only and multivendor networking environments.

New Features

The DTS/TIO subsystem provides the following new functionality for MPE/iX 6.0 release:

- Support of 400 DTCs.

- Support of 2000 device classes.

- Configuration of the maximum number of sessions for a terminal switching port on the Host-Based DTC management.

- Configuration of the number of hosts sharing the printer for a printer switching port on the Host-Based DTC management.

The following enhancements were made to MPE/iX release 6.0 for the DTS/TIO subsystem to provide support for all DTC port configuration parameters of DTC version 14.4 for Host-Based DTC management. Prior to release 6.0, these features were only available on the OpenView DTC Manager in a PC-Based management configuration.

- Three new modem types for attached device: US modem In Out, European modem In Out, and Five wires modem.

- Two new data length/parity settings.

- Configuration to enable or disable flow control between the DTC and the attached device.

- Configuration of the number of stop bits on Host-Based DTC ports.

- "Keep Alive" mechanism can be activated.

- Port configuration parameters (data length and parity) can be specified as forced when binary mode is selected.

- Dynamic Configuration of new port configuration parameters.

For more information refer to "New Features for DTS/TIO Subsystems" in Chapter 10, "Technical Articles," and to the manual, *Configuring System for Terminals, Printers, and Other Serial Devices.*

Telnet/iX Server—Full Functionality Release

by Cas Caswell
Commercial Systems Division

The version of the Telnet/iX Server released with MPE/iX Release 5.5 had several functionality restrictions that were documented in the MPE/iX 5.5 *Communicator* article, "Telnet/iX Server Functionality Details." The Telnet/iX Server included with this MPE/iX release is now fully functional. The purpose of this article is to update the information from the MPE/iX 5.5 *Communicator*.

Changes from the MPE/iX 5.5 Telnet/iX Version

- FControl and FDeviceControl behavior for this version will function as documented in the *Asynchronous Serial Communications Programmer's Reference Manual*.

  - All Block Modes and Binary Mode are now supported.

  - Type-ahead is now supported.

- The maximum number of concurrent Telnet Sessions has been raised from 250 to 2000.

- The TLNETDOC.ARPA.SYS file has been updated with information for the Telnet/iX Server, and is included in this patch.

- The Telnet/iX Server now supports all of the Telnet commands that require interaction with the Server.

  In addition to the commands `open`, `send break`, `send AYT`, `send IP`, `send EC`, `send EL`, `quit`, and `close` that were supported with the MPE/iX 5.5 version, this version of the Telnet/iX Server also supports the `toggle` and `mode` commands.

  You should be aware of the following points when using the `toggle` and `mode` options:

  - `toggle`

    The Server was written to assume the user is always right. This means that if you do a `toggle echo` to allow the client to do all the echoing, the Server stops echoing and leaves all echoing to the client. However, if you then run an application on the HP 3000 that enables echoing, there will be a double echo on your display.

  - `mode`

1. Line mode for the HP 3000 server was implemented to be consistent with the HPUX 9.0 Telnet product. Neither of these line modes comply with the RFC-1116 line mode standard.

2. We recommend that you do not use line mode when running block mode applications on the HP 3000. This may cause the applications to operate poorly.

FTP Enhancements

by Raghuram B.
Commercial Systems Division

The following is an overview of new features available with FTP/iX.

Implementation of FTPSRVR under INETD

Until now, FTP/iX server service, FTPSRVR, was initiated by the
FTPMON process which was started by the `JFTPSTRT.ARPA.SYS` job.
The INETD/iX process will now replace the FTPMON process and the
`JFTPSTRT.ARPA.SYS` job will be replaced by `JINETD.NET.SYS job.`

Anonymous FTP Functionality

FTP/iX anonymous logon allows a restricted access to the MPE/iX file
system via the FTP/iX server, without the need for a logon account. The
remote user can access the FTP/iX anonymous environment by
specifying either one of the accepted user names 'anonymous' or 'ftp'
(these login id's are currently defacto standards).

Support for Non-Standard MPE/iX File Types

The following types of MPE/iX files can be transferred to a remote host
using FTP.

- Standard MPE files with fixed, variable or bytestream format.

- Files with either binary or ASCII data types.

The Configurable Server Time-Out

Previous patch releases of FTP/iX 5.0 and 5.5 have a hard-coded infinite
time-out value applied to the data connection timer of the FTP/iX
server. The FTP/iX client, which shares this data connection, has a
configurable time-out which is user-controlled through the `TIMEOUT`
command.

Hash Command

While transferring files, if the user desires to receive progress status
during file transmission, the hash command provides this capability.

REPLACE Command

This FTP/iX client controlled option allows the user to override the
default file placement strategy of FTP/iX (unconditional replacement of
the target file.)

MOVE Option

This FTP/iX user-controller option allows the user to selectively specify on the `GET, PUT, MGET,` or `MPUT` command whether to delete the source file after confirmation of a successful file transfer to the target system.

Details and example of these options for FTP/iX are provided in the "Technical Articles" chapter of this book.

The Implementation of CI Environment Variables for Additional FTP/iX Status Information

The following status information will be provided through these CI environment variables:

`FTPXFERFILES` number of files successfully transferred during the last FTP/iX execution.

`FTPREQFILES` total number of files requested for transfer during the last FTP/iX execution.

`FTPREPLACE` reflects the state of the file placement policy of FTP/iX: either unconditionally (TRUE) or conditionally (FALSE) replace the target file(s) specified in the user commands GET, PUT, MPUT, and MGET (see the REPLACE option for additional details).

The following environment variables are currently supplied with FTP/iX:

`FTPLASTERR` identifies the last FTP error encountered.

`FTPLASTREPLY` identifies the last FTP protocol reply message string.

Samba/iX Overview

by Wendy Cheng
Commercial Systems Division

Overview

Samba/iX is a suite of programs which work together to allow clients to access a server's file space and printers via the Server Message Block (SMB) protocol. Samba is the SMB file server that runs on MPE/iX release 6.0. It allows the MPE/iX shell operating system to act as a file and printer server for SMB clients which are mainly for Microsoft LAN Manager and compatible products. These compatible clients can be Windows for Workgroups, Windows 95, Windows NT, or other clients.

Initially Samba/iX was written for UNIX; now it also runs on MPE/iX, NT, and other operating systems. Samba/iX provides the file and printer sharing services between NT and HP3000 systems. The user can redirect disks and printers to MPE disks and printers from Windows NT, Windows 95, LAN Manager, or other clients.

A generic UNIX program that is part of the Samba/iX suite has also been ported to MPE/iX Shell operating system. This program allows UNIX or MPE users to use an FTP-like interface to access file space and printers on any other SMB servers. This capability enables these operating systems to act like a LAN server or Windows NT server.

Major Components

Samba/iX contains programs like `SMBD` and `NMBD` to provide server functionality. It also contains utilities like `SMBCLIENT` or `NMBLOOKUP` to provide client functionality.

For more information, refer to "Introducing Samba/iX" in chapter 10, "Technical Articles."

Introducing Java/iX

by Mike Yawn

Overview

MPE/iX 6.0 is the first MPE release to provide a bundled version of Java. Java is a platform-independent language and runtime environment with a wealth of information available. Prospective users of Java are encouraged to observe the caveats regarding this initial release of Java/iX described in the detailed technical article found in Chapter 10Java sources include on-line reference materials, books and magazines devoted to Java, and widely available classes and instructional materials.

The Java programming language is an object-oriented language based on C. It first gained wide acceptance as an internet programming language because of its built-in support for handling distribution of code over the Internet, as well as prepackaged class libraries to handle sockets, URLs, and graphical content. In the few years since its introduction, it has grown beyond the internet market to become a popular language for the development of many kinds of programs, including traditional applications, especially when client/server or web-based access is required.

Programmers will find that Java facilitates the rapid development of error-free code, allowing programmers to be more productive than when writing in most other languages. The object-oriented nature of Java and the large number of readily available classes in the standard Java packages allow a high degree of code reuse.

Java includes the following major components:

- Java Virtual Machine, for running java applications

- Java compiler (javac), for compiling Java programs

- Java Packages, or Class Libraries, providing Application Programming Interfaces (APIs) that cover a large number of functional areas, including I/O, networking, and user interface.

Detailed instructions on installing and using Java can be found in the article, "Developer's Kit for Java/iX" in Chapter 10, "Technical Articles."

NOTE   In late breaking news, Sun has announced their intention to deliver a 'Server only' Java platform in 1999. Because the Java Server platform is a better fit for the needs of MPE/iX users, we expect to use it as the basis for future ports of Java to MPE/iX. Accordingly, we discourage the use of the Abstract Windowing Toolkit (AWT) that is included with this

version of Java/iX The AWT comprises the Java packages in the
java.awt hierarchy. These interfaces are unsupported on the current
release and will likely not be a part of future Java/iX releases.

8   Peripherals

This chapter contains the following articles on peripherals:

- 9 GB Disk Drive
- New Disk Devices Supported
- High Availability Disk Array (Nike)
- A3716A 144 GB DDS-3 Autoloader for MPE/iX
- EMC Symmetrix Disk Arrays

9 GB Disk Drive

by Alex Early
Commercial Systems Division

Overview

The 9 Gigabyte (GB) disk drive has been available since March 1997 as an embedded disk and as of April 1997 in the mass storage enclosure on the HP 3000 computer systems. Due to the performance of these disks, there are special requirements for their use on MPE/iX HP 3000 computer systems. Before installing this disk drive, consider the following:

1. While the 9 GB disks are supported as an embedded disk, they cannot be used as LDEV 1 (the boot disk). Therefore, customers ordering the 9 GB disk as an internal disk must also order a smaller boot disk such as a 2 GB or 4 GB for the LDEV 1 disk. The 9 GB disk can then be just another disk in the system domain.

2. Due to a number of differences in technology, the 9 GB disk drive is only supported on MPE/iX 5.5 and later releases.

Documentation shipped with the disk to help with installation has information which incorrectly implies that the 9 GB is supported on MPE/iX 5.0. The patches referenced in the documentation are for support of the EMC disk array which also uses 9 GB disk drives. The 9 GB disk drive documentation is being changed to state that the current 9 GB disk drive is only supported on MPE/iX 5.5 and later releases.

New Disk Devices Supported

by Rakesh Saha and Sam Boles
Commercial Systems Division

Overview

The HP 3000 supports the following disk devices on MPE/iX 6.0.

Table 8-1   **Supported Disk Devices**

Product ID	New Disk Device Supported
A3304A	ST32272N (2 GB single-ended narrow)
A3517A	ST32272WC (2 GB single-ended narrow)
A3351A	ST32272WC (2 GB differential wide)
A3318A	ST32272WC (2 GB differential wide)
A3352A	ST34572N (4 GB single-ended narrow) ST34573N (4 GB SE)
A3646A	ST34572WC (4 GB single-ended narrow) ST34573WC (4GB SE)
A3353A	ST34572WC (4 GB differential wide) XP34361W (4 GB differential wide) ST34573WC (4GB DF)
A3647A	ST34572WC (4 GB differential wide), XP34361W (4 GB differential wide) DGHS04Y (4GB DF) ST34573WC (4GB DF)
C6390A C6391A	ST34573WC (4 GB DF)
C6386A C6387A	ST34573N (4 GB SE)
A3628A	ST39173N (9 GB SE)
C6392A C6393A	ST39173N (9 GB SE)
A3629A	ST39173WC (9 GB DF)

Product ID	New Disk Device Supported
C6396A C6397A	ST39173WC (9 GB DF)
A5238A	ST39173WC (9 GB DF) DGHS09Y (9 GB DF)
A4909A	ST118273WC (18 GB DF),
A5239A	ST118273WC (18 GB DF)

Ambiguous Product ID

In the continuing effort to provide the price-performance advantage of the latest disk technology and, at the same, timeprotect existing customer investment, HP uses the same disk mechanism for both Single-Ended and Differential disks. However, there is a layer of transparent hardware that masks the electrical differences. This enables customers to continue to use both their existing Single-Ended and their existing Differential interface cards, but it requires some caution in connecting the disk device to the right interface.

Since an Inquiry by a utility such as `ODE MAPPER` returns the same Product ID (for example, ST34573WC) for both the Single-Ended and the Differential version of the disk, you must use the HP Model ID to distinguish the Single-Ended (for example, A3646A) from the Differential (for example, A3647A).

As a work-around for SYSGEN, use the "WC"-suffix ID (for example, ST34573WC) to bind to the Differential drivers; use the "N"-suffix ID (for example, ST34573N) to bind to the Single-Ended drivers.

CAUTION   It is critical that you connect the disk drive to the right bus: Single-Ended to Single-Ended; Differential to Differential. Failure to do so will result in unexpected system behavior.

Table 8-2   **HP Model Disk Drives**

HP Model	
Single-Ended	Differential
A3517A	A3318A
A3646A	A3647A

The following table details the Product IDs and the SCSI interface IDs.

Table 8-3   **Product ID for Configuring Disk Drives**

Drive Cap.	Drive Type	Product ID			Product ID for Corresponding Device Adapter
		Raw Mech	**Hot-Swap Module**	**SYSGEN**	
2 GB	SE	ST32272N	ST32272WC	ST32272N	HP27251A(CIO) HP28642A-SCSI(NIO) A1703-60003-SCSI(NIO) A2372-60003-SCSI(NIO) A2372-60004-SCSI(NIO)
2 GB	DF	ST32272WC	ST32272WC	ST32272WC	HP28696A(NIO) A2372-60003-FWSCSI(NIO) A2372-60004-FWSCSI(NIO)
4 GB	SE	ST34572N ST34573N	ST34572WC ST34573WC	ST34572N ST34573N	HP27251A(CIO) HP28642A-SCSI(NIO) A1703-60003-SCSI(NIO) A2372-60003-SCSI(NIO) A2372-60004-SCSI(NIO)
4 GB	DF	ST34572WC ST34573WC NA	ST34572WC ST34573WC DGHS04Y	ST34572WC ST34573WC DGHS04Y	HP28696A(NIO) A2372-60003-FWSCSI(NIO) A2372-60004-FWSCSI(NIO)
9 GB	SE	ST39173N	ST39173N	ST39173N	HP28642A-SCSI(NIO) A1703-60003-SCSI(NIO) A2372-60003-SCSI(NIO) A2372-60004-SCSI(NIO)
9 GB	DF	ST39173WC NA	ST39173WC DGHS09Y	ST39173WC DGHS09Y	HP28696A(NIO) A2372-60003-FWSCSI(NIO) A2372-60004-FWSCSI(NIO)
18 GB	DF	ST118273WC	ST118273WC	ST118273WC	HP28696A(NIO) A2372-60003-FWSCSI(NIO) A2372-60004-FWSCSI(NIO)

High Availability Disk Array (Nike)

With MPE/iX 6.0 there is support for 9 GB and 18 GB disk drives.
Please note the following restrictions.

1. The 9 GB and 18 GB drives are supported as RAID 1 only (that is,
   NOT as RAID 5, since this would exceed the 18 GB volume limit
   currently in effect.)

2. The 9 GB and 18 GB drives may not be used as the boot device;
   however, that is not recommended.

NOTE   This is NOT recommended because only 4 GB is allowed by IODC in the
current implementation. The 5 GB and 14 GB difference would be
unusable capacity. The release marks the space beyond 4 GB as
unusable so the OS updates can occur without violating the 4 GB
limitation.

3. Flare code 9.55.01 (or later) and PROM code 0173 (or later) are
   required for the array controller.

4. Support is for Model 20 Differential only.

5. HPC2300WD is the SYSGEN ID.

The following table shows the supported configurations:

Table 8-4   **Supported Configuration of Disk Arrays**

Disk Capacity	RAID 1	RAID 5	Boot
4 GB	X	X*	X
9 GB	X	NS	X**
18 GB	X	NS	X**
* Maximum 18 GB usable capacity per volume ** NOT recommended because 5 GB or 14 GB unusable			

The following table shows the product numbers and descriptions of available disk drives.

Table 8-5 **Product Numbers and Descriptions for Disk Drives**

HP Product Number		Description
A3549A A3550A A3550AZ	#341	5 ea 9.1 GB DD
A3549A A3550A A3550AZ	#342	10 ea 9.1 GB DD
A3549A A3550A A3550AZ	#343	15 ea 9.1 GB DD
A3549A A3550A A3550AZ	#344	20 ea 9.1 GB DD

A3549A A3550A A3550AZ	#351	5 ea 18.2 GB HH DD
A3549A A3550A A3550AZ	#352	10 ea 18.2 GB HH DD
A3549A A3550A A3550AZ	#353	15 ea 18.2 GB HH DD
A3549A A3550A A3550AZ	#354	20 ea 18.2 GB HH DD

A3549A A3550A A3550AZ	#361	5 ea 4 GB LP DD
A3549A A3550A A3550AZ	#362	10 ea 4 GB LP DD
A3549A A3550A A3550AZ	#363	15 ea 4 GB LP DD

HP Product Number		Description
A3549A A3550A A3550AZ	#364	20 ea 4 GB LP DD

A5132A		9.1 GB single disk
A5132AM		9.1 GB multi-pack disk

A5133A		18.2 GB single disk
A5133AM		18.2 GB multi-pack disk

A5169A		4 GB single disk
A5169AM		4 GB multi-pack disk

A3716A 144 GB DDS-3 Autoloader for MPE/iX

by Aida Silva
Commercial Systems Division

Product Overview

The A3716A is a high-speed high-capacity tape product combined with an autoloader. The tape drive conforms to the DDS-3 standard. The autoloader houses six cartridges in the magazine. DDS-3 tape allows 12 GB of uncompressed data to be stored on 125m DDS-3 Tapes. It ensures backward compatibility in reading and writing existing DDS tapes. The sustained data rate on uncompressed data tape is 1 MBPS.

Support Information

The A3716A is a Single-Ended SCSI-2 Device. The A3716A is supported on MPE/iX Release 6.0.

NOTE   The A3716A is NOT supported on MPE/iX 5.0, and there are currently no plans for 5.0 Support of this device.

Configuration

A3716A is supported only on MPE/iX NIO platform. It is not supported on MPE/iX CIO platform. A3716A can be configured on single-ended NIO SCSI card. The autoloader option switch settings should be set to 6 (Stacker mode). Other mode settings are NOT supported.

Configuration Example

On an NIO machine, Path 52 is configured as follows:

```
io>lp 52

PATH:  52                          LDEV:
ID:  A1703-60003-SCSI              TYPE:  DA
PMGR: SCSI_DAM                     PMGRPRI:  6
LMGR:                              MAXIOS:   0
```

If the SCSI ID of the device is 3, and path 52.3 is not configured:

```
io>addpath path=52.3 id=pseudo

io>ad ldev=31 path=52.3.0 id=hpc1557A
```

Verify the configuration using the `LDEV` and `LP` commands as shown in the following example:

Figure 8-1

```
io>ldev 31
LDEV:  31     DEVNAME:              OUTDEV:    0   MODE:
ID: HPC1557A                        RSIZE:   128   DEVTYPE:  TAPE
PATH:  52.3.0                       MPETYPE:  24   MPESUBTYPE:  7
CLASS: TAPE

io>lp 52.3.0
PATH:  52.3.0                       LDEV:     31
ID:  HPC1557A                       TYPE:  TAPE
PMGR: SCSI_TAPE_DM                  PMGRPRI:  10
LMGR:  LOGICAL_DEVICE_MANAGER       MAXIOS:    0

io>lp 52.3

PATH:    52.3                       LDEV:
ID:   PSEUDO                        TYPE:   DA
PMGR: TRANSPARENT_MGR               PMGRPRI:    6
LMGR:                               MAXIOS:     0
```

Operating Instructions

If you need to store more than one tape per day, you should purchase multiple devices or consider other technology.

WARNING   **This device is intended for customers who need to store one tape per day (maximum 12 GB on DDS-3 tape). Writing more than one tape per day will exceed the duty cycle of this technology. This could result in premature failure of the device, and more importantly, you may face DATA LOSS due to unrecoverable errors.**

The A3716A device magazine holds up to 6 DDS tapes.

With this usage, it is recommended that you load the magazine with five DDS-3 cartridges in slots one to five and a cleaning cassette in the sixth slot. The device should then be configured with the auto tape eject feature. With this configuration and the magazine loaded with the recommended cartridges, you could then perform backups Monday through Friday, without any intervention. After backup on the fifth cartridge, the drive will be cleaned automatically by the cleaning cassette from the sixth slot.

You can then load new tapes into the magazine (or a new magazine) and start the next week's cycle.

To set the tape eject option, the autoloader switch setting must be six and the following command should be issued interactively, before backup is started (assuming your A3716A is LDEV #31):

```
:run devctrl.mpexl.telesup 31 eject=enable
```

EMC Symmetrix Disk Arrays

The new SYM4 version of the EMC Symmetrix Disk Array is supported on MPE/iX 5.5 starting with Express 3. The SYM4 has the CUDA-9 9 GB disk mechanism that spins at 7200 rpm, a 20 MBs fast-wide differential bus to replace the 5MBs back-end single-ended bus, larger integrated cache, and other features to boost capacity and performance.

NOTE   This is NOT an HP or customer installable device.

The following is a summary of software requirements for support of the EMC Symmetrix FW/SCSI Disk Arrays. Device installation and support is provided by EMC. For this reason, the following guidelines are provided only for general information purposes. For detailed information on the support and configuration of these devices, refer to the chapter, "HP 3000 Server Environment," in the *EMC Product Guide—Symmetrix Open Systems Host Environment* (200-999-563).

This is the minimum microcode level supported:

- 5062-61-40 (3/26/97) with Busy and Environmental bits set on the Symmetrix

- 5063-39-28 (5/27/97) with Environmental bits set on the Symmetrix

- 5063-39-29 (5/27/97) with Environmental bits set on the Symmetrix

9   Year 2000 Enhancements

This chapter contains the following articles related to the enhancements for the compliance for Year 2000 readiness.

- Year 2000 Web Page

- Year 2000 Enhancements for MPE/iX

- New Date Intrinsics for MPE/iX

- VPLUS Enhancements for Turn-of-Century

- Predictive

- HP MPE/iX Databases

- HP Stage/iX

- Documentation Changes for MPE/iX Languages

- RPG/iX 4-Digit Year Enhancement

Year 2000 Web Page

For up-to-date information on the state of year 2000 readiness for specific HP products, refer to the web page, "HP Cure2000 Solution," at the following URL:

http://`www.hp.com/go/year2000`

Year 2000 Enhancements for MPE/iX

by M Gopalakrishnan
Commercial System Division

Overview

This version of MPE/iX provides enhancements to the operating system for the year 2000 and beyond. The enhancements include the enhancements to the operating system commands, utilities, VPLUS and databases. This article explains the enhancements.

Background

The year 2000 issues stem from the ability to handle correctly the year 2000 and beyond. This is particularly an issue for systems and applications that are using a two-digit year to express dates. There are a few commands, utilities, and databases in MPE/iX which were designed to handle two-digit years and these had to be enhanced to support year 2000 and beyond.

The year 2000 is a leap year, and this also had to be addressed. The MPE/iX operating system uses the Gregorian calendar and its definition of leap year: any year that is exactly divisible by 400, or that is exactly divisible by 4 and not exactly divisible by 100.

Two-Digit Years Interpretation

One of the ways for handling two-digit year input is to interpret the years in the method called fixed window method. In this method, the two-digit years map as follows:

- 00 .. 49 maps to 2000 .. 2049

- 50 .. 99 maps to 1950 .. 1999

For example, both of the following STREAM commands will schedule the JOBFILE as January 21, 2001:

```
:STREAM JOBFILE;DATE=01/21/2001
:STREAM JOBFILE;DATE=01/21/01
```

Intrinsics

The CALENDAR intrinsic and CALENDAR date representation format are widely used in MPE/iX. There are many other intrinsics that accept or return dates in calendar format. The following discussion clarifies the interpretation of calendar format year value.

CALENDAR Intrinsic

The CALENDAR intrinsic returns the CALENDAR date, including the day of year and the year of century. The existing CALENDAR intrinsic documentation refers to the year as "year of century." This should be interpreted as "year since 1900."

The new definition of the CALENDAR intrinsic is as follows:

Syntax

```
U16
date:=CALENDAR;
```

where *date* is a 16-bit unsigned integer (assigned functional return).

This returns the CALENDAR date in the following format:

Table 9-1   **CALENDAR Date Formats**

Bits	Value/Meaning
7:9	day of year
0:7	year since 1900

Description

For the year 2000, the CALENDAR intrinsic will return the year as 100, for 2001 as 101, and so on. With this current interpretation, CALENDAR format can handle years through 2027.

All intrinsics accepting or returning dates in CALENDAR format will behave the same way as the CALENDAR intrinsic.

NOTE   If your application uses the CALENDAR intrinsic or CALENDAR format, it may be required to verify the source for conformance to the above interpretation.

HPFOPEN/FOPEN Intrinsic

HPFOPEN item 31 (labeled tape expiration parameter) and FOPEN (*formmsg* label tape parameter) will accept both two- and four-digit years as the labeled tape expiration date. This is the same as the FILE command expiration date parameter. For an interpretation of two-digit years, refer to the section, "Two-Digit Years Interpretation," in this article.

New Date Intrinsics

This version of MPE/iX has new intrinsics added to manipulate dates in various formats. For more information, refer to the article, "New Date Intrinsics for MPE/iX," in this *Communicator.*

Commands

The following commands are enhanced to support both two- and four-digit years. For an interpretation of two-digit years, refer to the section, "Two-Digit Years Interpretation." New syntax for the commands with parameters accepting years follows:

- `STREAM` **command**

  `:STREAM` *jobname*`;DATE=`*datespec*

  where *datespec* is the date for streaming the job in the format *mm*/*dd*/[*yy*]*yy*. If omitted, the current date is used.

- `FILE` **Command**

  ```
  :FILE   [;LABEL=[ [volid][,[IBM][,[expdate][,seq] ] ] ] ]
          [          [ [ANS]                      ] ] ]
  ```

  where *expdate* is the expiration date in the format *mm*/*dd*/[*yy*]*yy*.

- `SETCLOCK` **command**

  `:SETCLOCK ;DATE=`*datespec*

  where *datespec* is the local date in the form *mm*/*dd*/[*yy*]*yy*.

- `LISTSPF/SPOOLF` **command**

  `:LISTSPF/SPOOLF ;SELEQ=[DATE=`*mm*/*dd*/`[yy]`*yy*`]`

- `STORE` **Command**

  ```
  :STORE          [{;DATE<=accdate}]
                  [{;DATE>=moddate}]
  ```

  where *accdate and moddate* are the accessed and modified dates in the form *mm*/*dd*/[*yy*]*yy*.

CI Variables

The available `HPYEAR` variable is corrected to return the last two digits of the current system year, and a new CI variable is added to have the four-digit current system year.

Existing variable:

`HPYEAR`   is a global variable used by the CI that shows the last two digits of the current year; the initial value is the year at logon. Type= R I

For example, `HPYEAR` will have 0 for year 2000 and 5 for year 2005.

New four-digit year variable:

HPYYYY   is a global variable used by the CI that shows the current year; the initial value is the year at logon. Type= R I

                 For example, for year 2000, HPYYYY will have 2000.

HPSPLITYEAR   is a global variable used by the CI that shows the current century split year; the initial value is 50. Type=W I

                 This variable is used in the new date intrinsics to manipulate dates in various formats.

Utilities

CLKUTIL and START utilities/commands available in ISL (Initial System Loader) are enhanced to accept dates both in two- and four-digit years.

New Date Intrinsics for MPE/iX

by M Gopalakrishnan
Commercial Systems Division

This release of the MPE/iX operating system has enhanced intrinsics for handling date formats. These intrinsics are targeted to support widely used existing date formats in MPE/iX and three new date formats.

The date intrinsics are broadly categorized as follows:

- HPDATECONVERT—Converting dates from one supported format to another.

- HPDATEFORMAT—Converting the supported format dates to the display formats desired by the user.

- HPDATEDIFF—Determining the number of days that separate two given dates.

- HPDATEOFFSET—Adding/subtracting an offset (days) to/from the given date.

- HPDATEVALIDATE—Validating the given date for conformance to a supported date format.

- HPCALENDAR, HPFMTCALENDAR—Using the new 32-bit HPCALENDAR format.

The following sections will discuss various date formats and the syntax and semantics of new date intrinsics. COBOL and Pascal programs using these intrinsics are provided as examples.

Existing Date Formats

Existing date formats that are widely used are summarized in the following table, "Existing Date Formats." The column "Sortable?" indicates whether the dates can be sorted (either numerically or lexicographically depending on how they are stored). The last column "Y2K Ready?" indicates whether dates in the format under consideration can be used to represent dates beyond 1999-12-31. The column "#Bytes" represents the number of bytes required to store the date format.

Table 9-2 **Existing Date Formats**

Storage Type	# Bytes	Explanation	Sortable?	Y2K Ready?
longint [a]	8	Microseconds since 1970-01-01 (MPE time-stamp)	yes	yes
integer	4	Upper 2 bytes: year next byte: month of year bottom byte: day of month	yes	yes
integer	4	Upper 2 bytes: year bottom 2 bytes: day of year	yes	yes
integer	4	Seconds since 1970-01-01 (POSIX.1 time() format; valid through 2038-01-18)	yes	yes
shortint	2	Upper 7 bits: #years since 1900 Lower 9 bits: day of the year (CALENDAR format; valid up to 2027-12-31)	yes	yes
integer	4	YYMMDD date	yes	no
integer	4	MMDDYY date	no	no
integer	4	DDMMYY date	no	no
ASCII [b]	6	YYMMDD date	yes	no
ASCII	6	MMDDYY date	no	no
ASCII	6	DDMMYY date	no	no
ASCII	6	YYMMDD date YY:MM3000 date [c]	yes	yes
ASCII	6	MMDDYY date YY:MM3000 date	no	yes
ASCII	6	DDMMYY date YY:MM3000 date	no	yes

a. "integer" and "longint" are binary values.
b. "ASCII" means ASCII character code.
c. MM3000 dates are represented as in the MM3000 product which used the ASCII letters "A" - "Z" for decades starting with the year 2000.

New Date Formats

Three new date formats are listed in the table, "HP Standard Formats." The first format is an integer representation of the ISO 8601 date format. The second is an ASCII representation of the same. The last date format is an extension of the existing 16-bit CALENDAR date format to a 32-bit format. These date formats will be referred to as "HP Standard Formats."

Table 9-3   **HP Standard Formats**

Storage Type	# Bytes	Explanation	Sortable ?	Y2K Ready?
integer	4	YYYYMMDD date	yes	yes
ASCII	8	YYYYMMDD date	yes	yes
integer	4	Upper 23 bits: #years since 1900 Bottom 9 bits: day of the year. (Extension of the existing CALENDAR format.)	yes	yes

Supported Date Formats

The "HP Standard Formats" and "Existing Date Formats" are combined in the following table, "Supported Date Formats." Each date format is assigned a date type code. The date intrinsics support these date formats and date type codes.

Table 9-4   **Supported Date Formats**

Date Type Code	Storage Type	# Bytes	Explanation	Sortable?	Y2K Ready?
1	longint	8	MPE time-stamp (microseconds since 1970-01-01)	yes	yes
2	integer	4	Upper 2 bytes: year next byte: month of year bottom byte: day of month	yes	yes
3	integer	4	Upper 2 bytes: year bottom 2 bytes: day of year	yes	yes
4	integer	4	Upper 23 bits: #years since 1900 bottom 9 bits: day of the year. (analogous to the existing `CALENDAR` format.)	yes	yes
10	integer	4	Seconds since 1970-01-01 (POSIX.1 time() format; valid through 2038-01-18)	yes	yes
14	shortint	2	Upper 7 bits: #years since 1900 Lower 9 bits: day of the year (`CALENDAR` format; valid up to 2027-12-31)	yes	yes
15	integer	4	`YYMMDD` date	yes	no
16	integer	4	`MMDDYY` date	no	no
17	integer	4	`DDMMYY` date	no	no
18	integer	4	`YYYYMMDD` date	yes	yes
25	ASCII	6	`YYMMDD` date	yes	no
26	ASCII	6	`MMDDYY` date	no	no
27	ASCII	6	`DDMMYY` date	no	no
35	ASCII	6	`YYMMDD date YY:MM3000` date[a]	yes	yes
36	ASCII	6	`MMDDYY date YY:MM3000` date	no	yes
37	ASCII	6	`DDMMYY date YY:MM3000` date	no	yes
38	ASCII	8	`YYYYMMDD` date	yes	yes

a. MM3000 dates are represented as in the MM3000 product which used the ASCII letters "A" - "Z" for decades starting with the year 2000.

Special Date Values

Special date values to represent UNKNOWN, INVALID, NEVER, NEEDED, EXPIRED, and ILLEGAL dates are defined for dates in "HP standard formats." The following table captures this date value information. The other date formats have been in existence for some time and hence the special date values are not defined for them. These special dates, when passed to the HPDATEFORMAT intrinsic, will result in a corresponding output string. For example, passing "00000103" to HPDATEFORMAT will result in the output string: "NEEDED".

When special dates are passed to HPDATECONVERT, HPDATEOFFSET, and HPDATEDIFF intrinsics as input dates, an error status is returned and the output date is initialized to a binary zero value or a blank string, depending on the date type of the output date.

Table 9-5   **Special Date Values**

Data Type	Format	Un-known	Invalid	Never	Needed	Expired	Illegal
4	YYYYDDD	0	0000367	0000368	0000369	0000370	0000371
18	YYYYMMDD	0	00000101	00000102	00000103	00000104	0000105
38	YYYYMMDD	"blank"	"0000010"	"00000102"	"00000103"	"00000104"	"0000105"

New Date Intrinsics

The date intrinsics support dates in the range 0001-01-01 through 9999-12-31. They use the Gregorian calendar for all calculations beyond 1753. The calendar followed by the intrinsics ignores the fact that calendars in different countries changed at different times and some dates were dropped from the calendar (around the year 1753). The validity of dates prior to 1753 cannot be guaranteed. All intrinsics accept byte-aligned input/output date parameters, and all these intrinsics are NM callable.

On an error, the intrinsics initialize the output parameters to either a binary zero or a blank string depending on the type of the parameter.

Though the date type "4" can represent years beyond 9999, a year beyond 9999 (which needs five digits/characters) is considered an error.

It is recommended that character array based output date parameters are initialized to blanks before passing them as arguments to these intrinsics.

New errors and warnings are documented in the System Message Catalog, SYSCAT.PUB.SYS, under the date intrinsics subsystem number 529.

HPDATECONVERT

This intrinsic converts the dates from one supported format to another.

Syntax

```
                    I32V    *    I32V    *    I32 I32V
HPDATECONVERT(inputcode , inputdate , outputcode , outputdate , status , cutoff)
```

Parameters

inputcode	is a 32-bit signed integer by value.
	The value should be one of the date type codes listed in the table, "Supported Date Formats."
inputdate	varies for type by reference.
	The interpretation depends upon the value of *inputcode*. See the table, "Supported Date Formats," for the supported date codes and their layouts.
outputcode	is a 32-bit signed integer by value.
	The value should be one of the date type codes listed in the table, "Supported Date Formats."
outputdate	returns the date as per the format chosen by the *outputcode* parameter.
	See the table, "Supported Date Formats," for the supported datecodes and their layouts.
status	is the HPE_STATUS parameter through which the error codes are returned. A value of 0 indicates no error and no warnings.
cutoff	is a 32-bit signed integer by value (optional).
	This is used in validating the input parameter when the input date has two-digit year. This is a required parameter for dates with two-digit years. In all other cases, this parameter is ignored.
	If the *cutoff* parameter is given as -1, the value of the CI environment variable HPSPLITYEAR is used as the cutoff year.
	This parameter's value should be in the range 0 .. 100. If the value of the parameter is 50, two digit years in the range 0 .. 49 will translate to 200 .. 2049 and those in the range 50 .. 99 will be translated to 1950..1999. If you specify the *cutoff* year as 70, the mapping will be 0 .. 69 as 2000 .. 2069 and 70 .. 99 as 1970 .. 1999.

HPDATEFORMAT

You can use this routine to format the dates that can be combinations of display formats as explained below. Many of these elements are taken from ALLBASE/SQL date formats.

You can convert dates in the "Supported Date Formats" to a display string of your choice (with restrictions). The HPDATEFORMAT intrinsic will accept these format strings. The format specification strings can have the following syntax:

Syntax

[{*FormatElement*} {*Punctuation*}]

Table 9-6   **Valid Parameters for** *FormatElement*

CC	Century (01 to 99)
YYYY	Year (0001 to 9999)
YY	Year of century (00 to 99) with leading zeros suppressed (0 to 99)
Q	Calendar quarter of the year (1 to 4)
MM	Month of the year (01 to 12)
ZMM	Month of the year with leading zeros suppressed (1 to 12)
DD	Day of the month (01 to 31)
ZDD	Day of the month with leading zero suppressed (1 to 31)
DDD	Day of the year (001 to 366).
ZDDD	DDD with leading zeros suppressed (1 to 366)
D	Day of the week (1 to 7 where Sunday is 1, Monday is 2,…)
WW	Week of the year (01 to 53)
ZWW	Week of the year with leading zero suppressed (1 to 53)
Mon	Month of the year in ASCII format (Jan, Feb,…)
Day	Day of the week in ASCII format (Sun, Mon,…)
MON	Month of the year in ASCII (uppercase) format (JAN, FEB,…)
DAY	Day of the week in ASCII (uppercase) format (SUN, MON,…)

Table 9-7   **Valid Characters for** *Punctuation*

-	Hyphen
/	Slash
.	Dot
	Blank
,	Comma
	Null (or no delimiter)

Thus, `YYYY.MON.DAY`, `YY/MM/DD`, `DDMONYY`, and `DD-ZMM-YYYY` are valid date formats. For example, "31 Jan 1997" when formatted through `DD-ZMM-YYYY` results in "31-1-1997," formatted through `YYYY.MON.DAY` results in "1997.JAN.FRI," while `YYYYMMDD` results in "19970131."

NOTE   Mixing the NULL punctuation character with other punctuation characters is *not* allowed. Thus, `YYYY/MM/DD` is a valid format, while `YYYYMM/DD` is not. *fmtdatelen* parameter is length of *formatspec* parameter including the null termination character.

Syntax

```
                I32V    *    CA      CA    I32   I32  I32V
HPDATEFORMAT(datecode,inputdate,formatspec,fmtdate,fmtdatelen,status,cutoff)
```

Parameters

datecode   is a 32-bit signed integer by value.

This value should be one of the date type codes listed in the table, "Supported Date Formats."

inputdate   is the input date.

The interpretation depends upon the value of *datecode*. See the table, "Supported Date Formats," for the supported datecodes and their layouts.

formatspec   is a character array (required).

This should be a NULL terminated string as per the syntax explained above in the table, "Format Specification Strings."

fmtdate   is a character array (required).

This array size should be at least that of *formatspec*. On return, it will contain the date formatted as per the *formatspec*.

If an invalid date is passed, on return from the intrinsic its contents will be UNKNOWN. For the "HP Standard Formats," if special values in the table, "Special Date Values," are passed for date parameter, on return from the intrinsic, the value of the string will be appropriately initialized. For example, for the date type 18, the initialized values for different special date values are as follows:

Date Value	Returned Contents
00000000	"UNKNOWN"
00000101	"INVALID"
00000102	"NEVER"
00000103	"NEEDED"
00000104	"EXPIRED"
00000105	"ILLEGAL"

If the character array passed does not have enough space to hold the special values or the formatted date, the behavior is undefined.

fmtdatelen   is a 32 bit integer by reference (required).

On input, it is the length of the *formatspec* parameter.

On return, it represents the number of characters HPDATEFORMAT placed into *fmtdate*.

status   is the HPE_STATUS parameter through which the error codes are returned. A value of 0 indicates no error and no warnings.

cutoff   is a 32-bit signed integer by value (optional).

This is used in validating and converting from two-digit to four-digit years. (See HPDATECONVERT documentation for more information on the *cutoff* parameter).

HPDATEDIFF

This intrinsic determines the number of days that separate two given dates.

Syntax

```
          I32V      *       *       I32    I32 I32V
HPDATEDIFF(datecode,  firstdate,seconddate,diffindays,status,cutoff)
```

Parameters

datecode is a 32-bit signed integer by value.

     This value should be one of the date type codes listed in the table, "Supported Date Formats."

firstdate is the first input date. The interpretation depends upon the value of *datecode*. See the table, "Supported Date Formats," for the supported datecodes and their layouts.

seconddate is the second input date. The interpretation depends upon the value of *datecode*. See the table, "Supported Date Formats," for the supported datecodes and their layouts.

diffindays is the number of days difference between the two dates, computed as: *seconddate* minus *firstdate*. Thus, if *seconddate* is earlier than *firstdate*, *diffindays* will be negative.

status is the `HPE_STATUS` parameter through which the error codes are returned. A value of 0 indicates no error and no warnings.

cutoff is a 32-bit signed integer by value (optional).

     This is used in validating and converting the two-digit years to four-digit years before computing the difference. (See `HPDATECONVERT` documentation for more information on *cutoff* parameter.)

HPDATEOFFSET

This intrinsic adds or subtracts a specified offset to or from the given date.

Syntax

```
            I32V    *    I32V    *      I32  I32V
HPDATEOFFSET(datecode,inputdate,offset,outputdate,status,cutoff)
```

Parameters

datecode is a 32-bit signed integer by value

     This value should be one of the date type codes listed in the table, "Supported Date Formats."

inputdate is the input date. The interpretation depends upon the value of *datecode*. See the table, "Supported Date Formats," for the supported datecodes and their layouts.

offset is a 32-bit signed integer by value.

The number of days to be added to the input date. A negative value will result in a subtraction.

outputdate   is the output date. The result of the date offset operation. The interpretation depends upon the value of *datecode*. See the table, "Supported Date Formats," for the supported datecodes and their layouts.

status   is the HPE_STATUS parameter through which the error codes are returned. A value of 0 indicates no error and no warnings.

cutoff   is a 32-bit signed integer by value (optional).

This is used in validating and converting the two-digit years to four digit ones before computing the difference. (See HPDATECONVERT documentation for more information.)

HPDATEVALIDATE

This intrinsic checks the validity of the given date with respect to the supported formats given in the table, "Supported Date Formats."

Syntax

```
 I32                I32V    *    I32V
result := HPDATEVALIDATE(datecode,inputdate,cutoff)
```

Parameters

datecode   is a 32-bit signed integer by value.

This value should be one of the date type codes listed in the table, "Supported Date Formats."

inputdate   is the input date.

The interpretation depends upon the value of *datecode*. See the table, "Supported Date Formats," for the supported datecodes and their layouts.

cutoff   is a 32-bit signed integer by value (optional).

This is used in validating dates with two digit years. (See HPDATECONVERT documentation for more information on *cutoff* parameter).

result   is a 32-bit signed integer (assigned functional return).

This value will be 0 if the *inputdate* conforms to the date format represented by *datecode*. If it is not so, its value will be positive. If an error has occurred in evaluating the conformance, its value will be negative. This return value ranges from -999 to 1.

HPCALENDAR

The new `HPCALENDAR` intrinsic returns the date in the supported date type code 4 listed in the table, "Supported Date Formats." (This is also a HP standard format.)

Syntax

```
I32
date := HPCALENDAR ;
```

where *date* is the 32-bit unsigned integer (assigned functional return).

This returns the calendar date in the following format:

Table 9-8   **HPCALENDAR Date Format**

Bits	Value/Meaning
23:9	Day of year
0:23	Year since 1900

HPFMTCALENDAR

This a new routine to handle `HPCALENDAR` format. It does the same job as `FMTCALENDAR` except that it accepts the 32-bit integer returned by `HPCALENDAR` intrinsic.

Syntax

```
         I32V  CA
HPFMTCALENDAR(date,formatdate)
```

Parameters

date   is a 32-bit signed integer by value

                This holds the calendar date, in the same format as the `HPCALENDAR` intrinsic (that is, date type 4).

formatdate   returns the formatted calendar date in a 17-character array. If the day of the month is less than 10, a blank precedes it. For example,

```
FRI, JAN  6, 1989
```

Examples

Following are two examples:

- The Pascal program example uses most of the new date intrinsics.

- The COBOL program example uses the `HPDATECONVERT` intrinsic.

Pascal Example of New Date Intrinsics.

```
$standard_level 'hp_modcal'$

Program dateintr(input,output);

     {Constants for the different date types.}
const
     hp_dt_mpe_time_stamp_fmt   = 1;
     hp_dt_packed_yymmdd_fmt    = 2;
     hp_dt_packed_yyddd_fmt     = 3;
     hp_dt_new_calendar_fmt     = 4;

     hp_dt_posix_time_stamp_fmt = 10;

     hp_dt_calendar_fmt         = 14;
     hp_dt_int_yymmdd_fmt       = 15;
     hp_dt_int_mmddyy_fmt       = 16;
     hp_dt_int_ddmmyy_fmt       = 17;
     hp_dt_int_yyyymmdd_fmt     = 18;

     hp_dt_ascii_yymmdd_fmt     = 25;
     hp_dt_ascii_mmddyy_fmt     = 26;
     hp_dt_ascii_ddmmyy_fmt     = 27;

     hp_dt_mm3000_yymmdd_fmt    = 35;
     hp_dt_mm3000_mmddyy_fmt    = 36;
     hp_dt_mm3000_ddmmyy_fmt    = 37;
     hp_dt_ascii_yyyymmdd_fmt   = 38;

     {Type definitions for the program}
type
    pac_20      = packed array [1..20] of char;
    iptr_type   = ^integer;

VAR
    date2_pac        : pac_20;       {Dates in packed character arrays.}
    fmt_pac          : pac_20;
    print_pac        : pac_20;

    cutoff           : integer;      {To represent cutoff date.}
    date1            : integer;      {Working dates.}
    date1_18         : integer;
    date2_18         : integer;
    date3_18         : integer;
    i                : integer;      {temp variables.}
    inptype          : integer;
    j                : integer;
    outtype          : integer;
    print_len        : integer;      {Length of formatted date.}
    temp_date        : integer;      {Another temp variable}

    status           : integer;

function HPCALENDAR : integer; intrinsic;
procedure HPFMTCALENDAR; intrinsic;
function HPDATEVALIDATE : integer; intrinsic;
procedure HPDATEFORMAT ; intrinsic;
procedure HPDATECONVERT ; intrinsic;
procedure HPDATEOFFSET ; intrinsic;
procedure HPDATEDIFF ; intrinsic;

begin
     {Initialize the variables.}
     cutoff  := 50;
     inptype := hp_dt_int_ddmmyy_fmt;

     date2_pac := '960121'; {The YYMMDD date in Supported format '25'.}
     date1     := 230196;   {The DDMMYY date in Supported format '17'.}

          {conversion from one non standard formatto another}
     HPDATECONVERT(inptype,date1,hp_dt_ascii_yymmdd_fmt,fmt_pac,
               status,cutoff);
     if( status <> 0) then
```

```
                writeln('HPDATECONVERT intrinsic returned error',status);

        {Convert 'date1' to the YYYYMMDD integer format. With 50}
        {as the cutoff year.                                    }
     outtype := hp_dt_int_yyyymmdd_fmt;
     HPDATECONVERT(inptype,date1,outtype,date1_18,status,cutoff);
     if( status <> 0) then
                writeln('HPDATECONVERT intrinsic returned error',status);


        {Convert the 'date2_pac' to a YYMMDD integer format.}
        {Default cutoff is assumed (i.e., 50)               }
     HPDATECONVERT(hp_dt_ascii_yymmdd_fmt,date2_pac,
      outtype,date2_18,status);
     if( status <> 0) then
        writeln('HPDATECONVERT intrinsic returned error',status);

        {compute the difference between the two dates}
     HPDATEDIFF(outtype, date1_18,date2_18,temp_date,status,cutoff);

     if( temp_date <  0) then
            writeln(date1,' is #',temp_date,
      'days later compared to ',date2_pac)
     else if (temp_date > 0) then
         writeln(date2_pac,' is #',temp_date,
         'days later compared to ',date1)
     else writeln(date1,' is same as ',date2_pac);


        {offset date1_18 by the difference to get the      }
        {value of date2_18 through HPDATEOFFSET intrinsic.}
     HPDATEOFFSET(outtype,date1_18,temp_date,date3_18,status,cutoff);
     if( (date3_18 <> date2_18 ) and (status = 0)) then
            writeln('Error Malfunction of HPDATEOFFSET intrinsic');

        {Use the flexibility in converting the date to a display }
        {string. Use '#0' to NULL terminate the format           }
        {specification string.                                   }
     fmt_pac := 'YY.ZMM.ZDD'#0;
     print_len := 11; { 11 characters including #0 in fmt_pac }
     HPDATEFORMAT(outtype,date1_18,fmt_pac,print_pac,print_len,status);

     if( status <> 0) then
            writeln('HPDATEFORMAT intrinsic returned error',status);

     writeln('The converted date is: ',print_pac);

        {The following demonstrates how a byte array can be      }
        {passed in place of an integer. We are storing the       }
        {integer 230196 in a packed array of characters, which is}
        {an equivalent of a byte array. We subsequently cast the }
        {address of this array to an integer pointer to read the }
        {integer value out of it.                                }
     HPDATECONVERT(hp_dt_int_yyyymmdd_fmt,date1_18,
         hp_dt_int_ddmmyy_fmt,fmt_pac,status);

     $push, type_coercion 'storage'$
     if( iptr_type (addr(fmt_pac))^ = date1 ) then
            writeln( 'Program worked fine! ')
     else    writeln( 'Error, in using byte array parameters !');
     $pop$

end.
```

The expected output from the program is:

```
230196 is #    2days later compared to 960121
The converted date is : 96.1.23
Program worked fine!
```

COBOL Example of `HPDATECONVERT`.

```
       IDENTIFICATION DIVISION.
       PROGRAM-ID.    DATEINTR.
       REMARKS. SAMPLE PROGRAM USING DATE INTRINSICS
       DATA DIVISION.
       WORKING-STORAGE SECTION.
      *Date code for YYMMDD is 25, for YYYYMMDD is 38
       01 INDATE-CODE                    PIC S9(9) VALUE 25.
       01 OUTDATE-CODE                   PIC S9(9) VALUE 38.
       01 SPLITYEAR                      PIC S9(9) VALUE 70.
       01 OUTDATE-YYYYMMDD               PIC X(8) VALUE SPACES.
       01 INDATE-YYMMDD                  PIC X(6) VALUE SPACES.
       01 STATUS-VAR.
          05 S-INFO                      PIC S9(4) COMP VALUE 0.
          05 S-SUBSYS                    PIC S9(4) COMP VALUE 0.

       PROCEDURE DIVISION.
       CONVERT-DATE-PARA.
          DISPLAY 'Enter the date in YYMMDD format:'
          ACCEPT INDATE-YYMMDD
          CALL INTRINSIC "HPDATECONVERT" USING
                                         INDATE-CODE
                                         INDATE-YYMMDD
                                         OUTDATE-CODE
                                         OUTDATE-YYYYMMDD
                                         STATUS-VAR
                                         SPLITYEAR
          IF S-INFO NOT = 0
             PERFORM DISPLAY-ERROR
          ELSE
             DISPLAY "Convert Date in YYYYMMDD = " OUTDATE-YYYYMMDD
          END-IF
          STOP RUN.

       DISPLAY-ERROR.
          DISPLAY "HPDATECONVERT FAILED. ERROR = " S-INFO.
```

VPLUS Enhancements for Turn-of-Century

VPLUS version B.06.07 (and later) contains enhancements to handle the year 2000 and beyond in applications. For more detailed information including syntax and parameters, refer to the article, "Year 2000 and Other Enhancements in VPLUS," in the chapter, "Technical Articles."

Turn-of-century Enhancement

VPLUS version B.06.07 contains enhancements to handle Year 2000 and beyond in applications. These enhancements can be grouped into three segments as follows:

- Intrinsic enhancements

- FORMSPEC enhancements

- Processing enhancements

Intrinsic Enhancements

Two new intrinsics have been added to VPLUS to handle dates in the next century. These are `VGETYYYYMMDD` and `VPUTYYYYMMDD`. These intrinsics are similar in structure and function to the existing intrinsics `VGETYYMMDD` and `VPUTYYMMDD` respectively. The difference is that the date values handled by the new intrinsics contain 4-digit year components.

FORMSPEC Enhancements

FORMSPEC Application Ready Buffer (ARB) processing has been enhanced with the addition of a new ARB data type. This data type can be specified on the DTC Menu for Data Type Conversions from Screen Type to Application Type. The new ARB Type is `YYYYMD` and is defined as a 8-byte ASCII field containing a date value in `YMD` order with no separators. Also the year component in the date value has 4 digits. Note that FORMSPEC will ensure a length of 8 for the data type `YYYYMD` in the ARB. In essence, the value is of the form `YYYYMMDD` (8 bytes) even though the Type designator on the DTC Menu has been abbreviated to `YYYYMD`.

Processing Enhancements

Date processing in VPLUS has been enhanced to handle date values with 4-digit year components. Specifically, the following additions have been made.

Date output will be formatted with 4-digit years if the target field is wide enough to receive the date, and the user requests this action. Examples of such output include the SET statement in FORMSPEC. The target field should be at least 10 characters in length to have a 4-digit year output.

VPLUS, by default, will format date output with 2-digit years. To change the behavior as described above, define a JCW called VSETNEXTCENTURY and set bit 15 to 1 (bits are numbered from 0 through 15). It should be noted that this option will be active for fields wide enough to hold the 4-digit year date. For other fields, 2-digit years will be used.

Existing 2-digit years can be interpreted differently to handle dates in the next century. Currently VPLUS handles year components 00-99 as 1900-1999. This scheme of interpretation can be changed. In the new scheme, 00-49 is interpreted as 2000-2049 and 50-99 is interpreted as 1950-1999. This interpretation will be valid through all processing done by VPLUS. For example, this scheme will allow the user to enter 02/29/00 (MDY order) in an existing application and have it processed correctly.

To invoke this method of date interpretation, define a JCW called VSETNEXTCENTURY and set bit 14 to 1. This method is used for processing only 2-digit year dates and not 4-digit year dates. The display of 2-digit year dates is not affected by this enhancement. This means that 12/11/10 (MDY order) can refer to 1910 in one application and 2010 in another application.

NOTE   When defining the JCW VSETNEXTCENTURY, set all unused bits to 0.

Other Products

Predictive

Hardware predictive (HP50779) has been modified to be year 2000 safe by modifications to internal date comparison routines and database access methods. Various other defect fixes which have been made since the release of MPE release 5.5 are also rolled into the 6.0 version of hardware predictive.

HP MPE/iX Databases

ALLBASE/SQL

The ALLBASE/SQL functions TO_DATE and TO_DATETIME are enhanced to handle two-digit years appropriately. For more information, refer to the article, "New Features in ALLBASE/SQL Version G3," in the chapter, "Technical Articles."

In SQLUtil, the SETUPRECOVERY command is used for roll-forward recovery. SETUPRECOVERY accepts two-digit years as part of the date specification. These two-digit years are interpreted as discussed in the section, "Two-Digit Years Interpretation," in the article, "Year 2000 Enhancements for MPE/iX."

IMAGE/SQL and TurboIMAGE/XL

IMAGE/SQL and TurboIMAGE/XL use the ALLBASE/SQL TO_DATE and TO_DATETIME functions to handle the two-digit years appropriately.

HP Stage/iX

HP Stage/iX can now handle dates with year greater than or equal to 2000 correctly (when the LIST command is used). It is now Y2K compliant.

Documentation Changes for MPE/iX Languages

by Allan Hertling
SSG Group R&D

FORTRAN 77/iX

Please make the following corrections to the *FORTRAN 77/iX Reference Manual*.

This manual incorrectly states that a four-digit calendar year may be computed by adding 1900 to the YEAR parameter resulting from a `IDATE` subroutine call. As the year parameter is always in the range of 0 to 99, that calculation will not be correct after 1999.

The last sentence describing the four-digit calculation under the `IDATE` Subroutine discussion on page B-21 should be deleted.

Business BASIC

Please make the following corrections to the *Business BASIC Reference Manual*.

The TIME function documentation on page 18-8 should be changed as follows:

Under Parameters, num_expr, Three or greater:
change "Current year" to "Years since 1900".

Under Examples, add an example of getting the current year:

```
160 Year2 = TIME (3) MOD 100 ! Get the current year of the century
170 Year4 = TIME (3) + 1900   ! Get the current 4-digit year
```

Business BASIC/iX

Please make the following corrections to the *Business BASIC/iX Reference Manual* (Part # 32715-90001).

The TIME function documentation on page 5-98 should be changed as follows:

Under Parameters, num_expr, Three or greater:
change "Current year" to "Years since 1900".

Under Examples, lines 140 and 150:
change "the current year" to "years since 1900".

Add an example to get the current year:

```
160 Year2 = TIME(3) MOD 100 ! Get the current year of the century
170 Year4 = TIME(3) + 1900  ! Get the current 4-digit year
```

BASIC

Please make the following corrections to the *BASIC INTERPRETER Reference Manual.*

The TIM function documentation on page E-3 should be changed as follows:

For x >= 3, change "current year (0-99)" to "years since 1900".

Add an example of getting the current year:

```
150 REM -- Get the current year of the century
160 Y2 = TIM(3) MOD 100
170 REM -- Get 4-digit year
180 Y4 = TIM(3) + 1900
```

RPG/iX

Please make the following changes to the *RPG/iX Reference Manual*.

The following changes to the *RPG/iX Reference Manual* describe the new functionality:

In the Header Specifications chapter, make the following changes to UDATE Source (Column 17):

- Add *DATE and *YEAR to the list of predefined fields whose initial value is set by the RPGUDATE file.

- Change the second paragraph under F (Date Record) to:

  You can enter the date (in the date record) in a flexible manner. The first number begins the date (the number does not have to start in column 1). Enter the date as six or eight consecutive digits (mmddyy or mmddyyyy) or in three sets of digits (mm, dd, and yy or yyyy). Enter the month (mm),the day (dd), and then the year (yy or yyyy).

- Add the following:

  If the RPGUDATE file contains a 2-digit year, *DATE and *YEAR will contain a 4-digit year based on a date window of 1940-2039. Thus, a yy of 39 will be taken as 2039, while a yy of 40 will be taken as 1940.

- Add the following examples using 4-digit years:

  ```
  04101988
  FORMAT OF DATE IS MMDDYYYY = 04101988
  4/10/1988
  MONTH IS 4; DAY IS 10; YEAR IS 1988
  ```

- Remove the invalid date example for 4/10/1988.

In the Output Specifications chapter, make the following changes to Field Description Fields (Columns 32-70):

- Add *DATE and *YEAR to the section on UDATE, UDAY, UMONTH, UYEAR:

Table 9-9   **DATE Field Descriptions**

Field Name	Contents	Unedited Example	Edited Example	Description
*DATE	Current date with 4-digit year	10111988	10/11/1988	October 11, 1988 (Domestic Format)
		11101988	11/10/1988	October 11, 1998 (United Kingdom Format)
		11101988	11.10.1988	October 11, 1988 (European Format)
*YEAR	Current 4-digit year	1988	19/88	1988

NOTE   At the point that editing is done, there is no way for RPG to tell that the original field was a 4-digit year and not a truncated date. Therefore, the edited output for *YEAR will be of the form nn/nn, as described in the documentation for Edit Code (column 38).

In the Output Specifications chapter, make the following changes to Edit Code (Column 38):

- In the Description for the Y edit code field, change the maximum field length to 8, and add the specification for field lengths 7 and 8:

  Slash marks can be used with fields having three to eight digits as follows:

Table 9-10   **Edit Code Fields**

Field length	Edited output
7	nn/nn/nnn
8	nn/nn/nnnn

RPG/iX 4-Digit Year Enhancement

by Sue Meloy
Software and Services Group

*DATE and *YEAR

RPG/iX version A.00.18 contains support for new predefined fields that return a 4-digit year. *DATE is analogous to UDATE, except that the year is 4 digits instead of 2 digits. In the same way, *YEAR is analogous to UYEAR.

RPGUDATE

The RPGUDATE file format has also been changed to allow a 4-digit year.

Documentation Changes

There are a number of changes to the *HP RPG/iX Reference Manual* that describe the new functionality. See the article, "Documentation Changes for MPE/iX Languages" in Chapter 9 for more information.

Compatibility with Previous Versions

If 4-digit years are used in the RPGUDATE file, RPG/iX programs must be run using the new RPG/iX library.

Programs that reference the new *DATE and *YEAR fields must be compiled with the new RPG/iX compiler and run with the new RPG/iX library.

Programs that were compiled with previous versions of the RPG/iX compiler can be run using the new RPG/iX library without recompilation.

10   Technical Articles

This chapter contains the following articles with additional technical information:

- QUERY/iX Enhancements
- COBOL II/iX Enhancements
- IMAGE/SQL with TurboIMAGE/XL Enhancements
- New Features in ALLBASE/SQL Version G3
- Year 2000 and Other Enhancements in VPLUS
- CI Enhancements
- The REMSH Client Functionality
- Transaction Manager
- New Features for DTS/TIO Subsystem
- Introducing ODBCLink/SE
- Introducing Samba/iX
- Developer's Kit for Java/iX
- Hewlett-Packard NetWorker Client Pak for MPE/iX
- FTP Enhancements
- DNS BIND/iX and Syslog/iX

QUERY/iX Enhancements

by James Overman
SSG Group R&D

Overview

The following topics in this article review the new enhancements for QUERY/iX and some limitations on their use:

- B-Tree Enhancement
- Dynamic Dataset Capacities
- Wildcard Retrievals
- Wildcard Character Limitations
- B-Tree Limitations
- "NO MATCH" Enhancement

B-Tree Enhancement

The latest B-Tree enhancement to TurboImage/XL has required some enhancements to the QUERY product to utilize the new features. With the HP32216D.03.11 and N.03.11 releases of QUERY, the new B-Tree access capabilities will be supported within QUERY on systems with the B-Tree software release. QUERY is fully backward compatible with the prior releases of TurboIMAGE/XL and QUERY.

New messages will be seen when B-trees are utilized by a database and the `FORM` command is entered. The number of indexed sets, the wildcard character, the status of the `BTREEMODE1` flag when OFF, and the individual sets and items that have been indexed will be reported as shown in the following example:

Figure 10-1   **FORM Example**

```
>FORM

DATA BASE: TOYDB                         MON, MAY 30, 1997, 10:48 AM
DATA BASE LANGUAGE ATTRIBUTE: NATIVE-3000
DATA BASE HAS 2 INDEXED SETS WITH WILDCARD CHARACTER= "~"   NEW
WARNING! DATA BASE BTREEMODE1 IS OFF.  WILDCARD DISABLED.   NEW

SET NAME:
   CUSTOMERS,MANUAL/INDEXED                                 NEW

       ITEMS:
          CUSTOMER-NO,          X6    <<KEY,INDEX ITEM>>    NEW
          CUSTOMER-NAME,        X30

       CAPACITY: 50              ENTRIES: 8

SET NAME:
   ORDER-MASTER,AUTOMATIC/INDEXED                           NEW

       ITEMS:
          ORDER-NO,             X8    <<KEY,INDEX ITEM>>    NEW

          CAPACITY: 503             ENTRIES: 16

SET NAME:
   INVOICES,DETAIL

       ITEMS:
          INVOICE-NO,           X6
          CUSTOMER-NO,          X6    <<SEARCH,INDEX ITEM>>   NEW
          ORDER-NO,             X8    <<SEARCH,INDEX ITEM>>   NEW
          AMOUNT,               P12

        CAPACITY: 5016          ENTRIES: 5000
   MAXIMUM CAPACITY: 5016    INITIAL CAP.: 38    INCREMENT: 38  NEW
```

Dynamic Dataset Capacities

The last line above shows another enhancement to QUERY that reflects
the Dynamic Dataset Capacity Expansion capability. When a dataset
has utilized the Dynamic Expansion feature, QUERY will display the
CURRENT, MAXIMUM, and INITIAL capacities and the
INCREMENT amount. The FORM SETS output has been slightly
modified such that the heading reflects that the Current capacity is
being reported rather than the Maximum capacity.

Wildcard Retrievals

QUERY will automatically invoke the B-Tree retrieval feature for the
FIND, LIST, SUBSET, and MULTIFIND commands when a wildcard
character is used in the search value. As QUERY may have up to ten
databases open and each could have a different wildcard, the wildcard
for the DEFINED database will be used when specified, else the first
wildcard specified by an open database will be used. QUERY uses
modes 21 or 24 for B-Tree DBFIND for best performance.

Wildcard Character Limitations

Certain possible wildcard characters are used by QUERY for syntax scanning and are limited in their usage. These special characters may be utilized when the data values are delimited by double quotes (for example, tilde, carat, comma, and parenthesis).

Figure 10-2

```
>FIND CUSTOMERS.CUSTOMER-NO=01540~
```
Tilde is a limited wildcard

```
EXPECTED A CONNECTOR OR 'END'

INVALID CONNECTOR OR TERMINATOR!
```

```
>FIND CUSTOMERS.CUSTOMER-NO="01540~"
```
Tilde enclosed in quotes works

```
8  ENTRIES QUALIFIED
```

```
>FIND CUSTOMERS.CUSTOMER-NO=01540"
```
Double-quote will not work…
```
EXPECTED A CONNECTOR OR 'END'


INVALID CONNECTOR OR TERMINATOR
```

```
>FIND CUSTOMERS.CUSTOMER-NO="01540"""
```
unless doubled in quotes and double-quote is wildcard
```
8  ENTRIES QUALIFIED
```

```
>FIND CUSTOMERS.CUSTOMER-NO=01540*
```
If asterisk is NOT the wildcard for   this database, then no match unless the value 01540 is in dataset.*
```
0  ENTRIES QUALIFIED
```

B-Tree Limitations

QUERY will utilize the B-Tree functionality when possible for both character and binary data types. The wildcard feature is only implemented by TurboIMAGE/XL for character data types (X and U) and not for binary data types. Neither has the Not Equal operator been implemented for B-trees and so QUERY will continue to use the older access methods (normally serial reads) for Not Equal retrievals. Note that the LIST command always uses a serial read but will apply the wildcard search character for relations.

When a database has the BTREEMODE1 flag set to OFF, QUERY will not attempt to use the B-Tree functionality for Wildcard retrievals. This condition is reported by all forms of the FORM command.

"NO MATCH" Enhancement

The FIND, MULTIFIND, and SUBSET commands have been enhanced to accept a NO MATCH alternative such that all data records NOT MATCHING the pattern provided are returned by the commands. The new form of the FIND command is:

```
FIND [#LIMIT=I;] RELATION/ITEM IDENTIFIER [NO] M[ATCHING] "PATTERN"
[AND/OR RELATION/ITEM IDENTIFIER [NO] M[ATCHING] "PATTERN"]... [END]
```

The MULTIFIND and SUBSET command syntax are similarly enhanced.

The NO may be followed by additional characters such that all of the following examples are accepted:

```
FIND PRODUCTS.NAME NO MATCH "ABC?*"
FIND PRODUCTS.NAME NOT MATCHING "ABC?*"
FIND PRODUCTS.NAME NONE M "ABC?*"
```

This NO Matching enhancement is NOT related to the B-Tree functionality and does not utilize the B-Tree retrieval mechanisms.

COBOL II/iX Enhancements

by Walter Murray
Support Technology Lab

Overview

This release of COBOL II/iX on MPE/iX 6.0 provides several
enhancements that were requested by Interex SIGCOBOL. Some of
these enhancements have already been made available in various
patches to MPE/iX 5.5.

Secondary Entry Points

COBOL II/iX now supports secondary entry points in a main program.
To specify a secondary entry point, use the ENTRY statement, and give
a nonnumeric literal with the desired entry point name. Specifying that
entry point on the MPE/iX RUN command will cause execution to begin
at that ENTRY statement. Any number of ENTRY statements may
appear, as long as they specify different entry point names. The ENTRY
statement is a nonstandard HP extension to the COBOL language.

The following sample program demonstrates the use of a secondary
entry point.

```
      IDENTIFICATION DIVISION.
      PROGRAM-ID. UPDATE-MASTER.
      DATA DIVISION.
      WORKING-STORAGE SECTION.
      01  SW-CLEAR-YTD-TOTALS     PIC X     VALUE "F".
      PROCEDURE DIVISION.
      0000-BEGIN.
*      Enter here for routine processing.
          GO TO 1000-MAIN.
      ENTRY "YEAREND".
*      Enter here for year-end processing.
          MOVE "T" TO SW-CLEAR-YTD-TOTALS.
      1000-MAIN.
*      Update master file, optionally resetting
*      year-to-date totals, etc.
          DISPLAY "Begin UPDATE-MASTER:  SW-CLEAR-YTD-TOTALS = ",
                  QUOTE, SW-CLEAR-YTD-TOTALS, QUOTE
*      ...
          STOP RUN.
      END PROGRAM UPDATE-MASTER.
```

Suppose this program is compiled and linked into an executable
program named UPDATE. The usual way to invoke it would be with
this command:

```
      RUN UPDATE
```

Alternately, it may be invoked through its alternate entry point, using
this command:

```
      RUN UPDATE,YEAREND
```

CALL by plabel

This enhancement provides an alternate way to achieve execution-time binding with the CALL statement. In standard COBOL, the CALL statement can use an identifier that specifies a nonnumeric data item containing the name of the desired subprogram. COBOL II/iX generates a call to the HPGETPROCPLABEL intrinsic to dynamically load the specified procedure, and that procedure is then executed. The XLs searched in this case are those in the binding sequence of the calling process.

There are situations where it may be desirable for the programmer to call HPGETPROCPLABEL explicitly, for example, when the XL to be searched is not known at load time or is not in the calling program's binding sequence. It is now possible to code your own call to HPGETPROCPLABEL, save the returned procedure label in a numeric data item, and then use that plabel in a CALL statement.

As an example, suppose that the following subprogram has been compiled and placed in an executable library named TESTXL.

```
        IDENTIFICATION DIVISION.
        PROGRAM-ID. COBOLSUB INITIAL.
        PROCEDURE DIVISION.
        BEGIN.
            DISPLAY "In COBOLSUB"
            EXIT PROGRAM.
        END PROGRAM COBOLSUB.
```

Using the functionality of call by plabel, this subprogram can now be dynamically loaded and invoked as follows.

```
        IDENTIFICATION DIVISION.
        PROGRAM-ID. COBOL-MAIN.
        DATA DIVISION.
        WORKING-STORAGE SECTION.
        01  PROCNAME        PIC X(10)    VALUE "-COBOLSUB-".
        01  PLABEL          PIC S9(9)    COMP.
        01  FIRSTFILE       PIC X(8)     VALUE "-TESTXL-".
        PROCEDURE DIVISION.
        BEGIN.
            CALL INTRINSIC "HPGETPROCPLABEL" USING
                    PROCNAME, PLABEL, \\, FIRSTFILE
            CALL PLABEL
            STOP RUN.
        END PROGRAM COBOL-MAIN.
```

The ability to use a numeric data item in a CALL statement is a nonstandard HP extension to the COBOL language.

New Routines for Boolean Operations

There are six new routines for performing boolean operations:

- HP_BYTE_AND

- HP_BYTE_OR

- HP_BYTE_XOR

- HP_BYTE_NOT

- `HP_BYTE_UNPACK`

- `HP_BYTE_PACK`

These procedures reside in the COBOL II run-time library in
`XL.PUB.SYS`, but may be called from any program running in Native
Mode.

The routines `HP_BYTE_AND`, `HP_BYTE_OR`, and `HP_BYTE_XOR` perform
bitwise `AND`, bitwise inclusive `OR`, and bitwise exclusive `OR`. The two
operands and the result may be any length, but must be the same
length, and must be an integral number of bytes. The three routines
have identical calling sequences. The first two parameters are the two
operands, passed by reference. The third parameter is the result, also
passed by reference. The final parameter is the length, in bytes, of the
operands, and is passed by value. The first three parameters may not
overlap, except in the case where two of them, or all three, are the same
data item.

Examples

```
CALL "HP_BYTE_AND" USING OPERAND-1, OPERAND-2, RESULT, \4\.
CALL "HP_BYTE_OR" USING DATA-ITEM, MY-BIT-MASK, RESULT, \2\.
CALL "HP_BYTE_XOR" USING INPUT-BUFFER (J:1), RUNNING-XOR,
                        RUNNING-XOR, \1\.
```

Note that in COBOL II/iX, backslashes ("\") are used to indicate that a
parameter is passed by value. If the parameter is a literal, the
backslashes are optional.

The routine `HP_BYTE_NOT` has the same calling sequence, except that
there is only one operand rather than two. The result is computed as
the bitwise complement of the operand. The operand and the result
must be the same length, and must be an integral number of bytes.
They may not overlap, except that the same data item may be used for
both.

Example

```
CALL "HP_BYTE_NOT" USING OPERAND, RESULT, \4\.
```

The routine `HP_BYTE_UNPACK` takes three parameters: an operand, a
result, and the length of the operand in bytes. The bits of the operand
are unpacked into the result, left to right. Each "zero" bit of the operand
becomes an ASCII "0" byte in the result; each "one" bit becomes an
ASCII "1" byte. The length specified is the byte length of the operand.
The byte length of the result must be 8 times the byte length of the
operand.

Example

```
01  FIELD-A    PIC S9(4)    COMP.
01  RESULT     PIC X(16).
...
MOVE 5 TO FIELD-A.
CALL "HP_BYTE_UNPACK" USING FIELD-A, RESULT, \2\.
DISPLAY RESULT.
* Results in "0000000000000101".
```

The routine `HP_BYTE_PACK` is similar. The first parameter, the operand, is a sequence of ASCII bytes. Each byte of the operand is converted to a bit in the result, left to right. An ASCII "0" becomes a "zero" bit; anything other than an ASCII "0" becomes a "one" bit. The length specified is the length of the result in bytes. The byte length of the operand must be 8 times the byte length of the result.

Example

```
    01  BYTE-STRING    PIC X(16).
    01  RESULT-N       PIC S9(4)    COMP.
    ...
    MOVE "0000000000001111" TO BYTE-STRING.
    CALL "HP_BYTE_PACK" USING BYTE-STRING, RESULT-N, \2\.
    DISPLAY RESULT-N.
  * Results in +15.
```

Compiling Larger Programs

Internal data structures of the compiler have been expanded to permit compiling significantly larger programs. While there is no specific limit on the number of lines permitted in a source program, this version of the compiler should be capable of processing programs well in excess of 200,000 lines.

DISPLAY Index-name

The compiler has been enhanced to permit an index-name to be used as an operand of a `DISPLAY` statement. This is an ANSI extension, and if this feature is used and `$CONTROL STDWARN` is specified, the compiler will display warning 517, `DISPLAY of index-name is nonconforming nonstandard HP extension.`

New Qedit Diagnostic

The compiler is now able to detect when a source file is in Qedit format and the Qedit product has not been properly installed on the system. Qedit is a full-screen text editor from Robelle Consulting Ltd. In the past, Qedit users have sometimes encountered problems when updating to a new release of MPE/iX and forgetting to reinstall Qedit. COBOL II/iX now detects this condition and produces error 472, `QEDIT FORMAT ENCOUNTERED FOR FILE.`

IMAGE/SQL with TurboIMAGE/XL Enhancements

by IMAGE/SQL Team
Commercial Systems Division

Overview

This article discusses enhancements as well as other important information commencing with the Express 3 release of MPE/iX 5.5. Express 3 (and later releases) also includes the revised editions of some selected manuals.

Highlights

Highlights of this release of IMAGE/SQL, version B.G3.00 or later, and TurboIMAGE/XL, version C.07.14 or later, are the following:

- Dynamic data set expansion for master sets (MDX or DDXM)

- Performance enhancement for databases with third-party indices

- 80 GB data sets

- Predicate locks when updating manual masters

- DBSCHEMA and P type with odd sub-item length

- B-Tree indices enhancement

- New 16-bit and 32-bit ODBC driver, ODBCLink/SE, bundled with IMAGE/SQL

- Support for third-party composite indices in IMAGE/SQL

- Scalability for DBPUT, DBDELETE, and DBUPDATE (CIU on)

- Dynamic detail data set expansion (DDX) fix

Manuals Bundled with IMAGE/SQL

The following manuals are included with IMAGE/SQL:

- *HP IMAGE/SQL Administration Guide*

- *Getting Started With HP IMAGE/SQL*

- *TurboIMAGE/XL Database Management System Reference Manual* (Revised August 1997)

- *ALLBASE/SQL Database Administration Guide*

- *ALLBASE/SQL Reference Manual* (36216-90001)

- *ALLBASE/SQL Message Manual*

- *ALLBASE/SQL Performance and Monitoring Guidelines*

- *ISQL Reference Manual for ALLBASE/SQL and IMAGE/SQL*

- *ODBCLink/SE Reference Manual*

Updating IMAGE/SQL

If you are updating from an earlier release of IMAGE/SQL and you have databases which are already `ATTACH`ed to one or more DBEnvironments, you must `DETACH` and `ATTACH` again in order to benefit from the enhancements. Also, it is recommended that you issue the `UPDATE STATISTICS` command of ISQL for each of the tables you access frequently. This may enable you to attain performance improvement.

Furthermore, if you have created SQL data using an earlier release of IMAGE/SQL and are now updating to a latter release, you must perform the ALLBASE/SQL migration using SQLINSTL or SQLMigrate. For more information, refer to the *ALLBASE/SQL Database Administration Guide* and the *Read Before Installing* for this release.

TurboIMAGE/XL Enhancements

The following are TurboIMAGE/XL enhancements, independent of SQL access:

- Dynamic data set expansion for masters (MDX or DDXM)

- 80 GB data sets

- Performance Enhancement for databases with third-party indices

- DBSCHEMA and P type with odd sub-item length

- B-Tree indices

- Scalability for DBPUT, DBDELETE, and DBUPDATE (CIU on)

Dynamic Data Set Expansion for Masters

Dynamic data set expansion feature for non-jumbo (file size within 4 GB limit) detail data sets (DDX) was first released in MPE/iX 5.0. With this release, this feature is available for non-jumbo master data sets (MDX or DDXM) as well. The feature allows dynamic expansion of a data set during DBPUT when the data set has approximated its current capacity and DBPUT would fail unless the data set is expanded. As in DDX, the capacity parameters which are maximum capacity, initial capacity, and increment (optional), used for dynamic expansion, must be set prior to the actual expansion. For new databases, these parameters can be specified in the CAPACITY statement of the schema definition to be processed by DBSCHEMA. For existing databases, third-party tools which support MDX need to be employed.

The capacity parameters are:

- maximum capacity

  is a required parameter and is a maximum number of entries the data set can contain. It must be less than or equal to $2^{31}-1$ (2,147,483,647). The maximum capacity is adjusted by TurboIMAGE/XL to represent a multiple of the blocking factor.

- initial capacity

  is a required parameter and is the initial, primary, or hashing capacity for the data set. It is the number of entries for which space will be allocated and initialized when the data set is created. More importantly, the initial capacity, and not the current capacity (initial capacity with expansions included), will be used in the hashing algorithm to calculate the primary address of the new entry being added. If initial capacity is very low, there can be frequent expansions which could result into both long synonym chains which degrade the performance of some TurboIMAGE intrinsics and disk fragmentation. Therefore, this number should be derived after giving consideration to the expected volume of current data,

anticipated growth, and synonym chain lengths. It should be such that a minimum number of expansions are necessitated. The initial capacity must be between 1 and 2^{31} −1 inclusive but must be less than or equal to the maximum capacity. If it is not specified, or if it is either zero or equal to the maximum capacity, dynamic capacity expansion is not enabled for the data set, and maximum capacity is used for the data set file creation and primary address calculation. The initial capacity is adjusted to represent a multiple of the blocking factor.

- increment

    is an optional parameter and is either the number of entries or the percentage of the initial capacity by which the data set needs to be expanded each time. If a percentage is used, the percent sign (%) must follow the incremental amount. The increment is adjusted (reduced) for the last expansion so that it does not exceed the maximum capacity. This increment parameter can only be used if a valid initial capacity parameter is also specified. This number must be 1 to 100 inclusive for percent, or 1 to 2^{31} −1 (2,147,483,647) inclusive for number of entries. If it is very low, there can be frequent expansions leading to severe disk fragmentation and performance degradation. If it is very high, DBPUT that triggers the expansion may take long time to complete which could impact other database users.

    If the increment is not specified for the data set, or is zero, but the initial capacity is greater than zero and not equal to the maximum capacity, the increment for each expansion defaults to ten percent (10%) of the initial capacity for the data set. If the initial capacity is equal to the maximum capacity, or the initial capacity is zero, it indicates that the data set cannot be expanded and increment is ignored.

The enhanced syntax for the CAPACITY statement of DBSCHEMA is same for both DDX and MDX, and is as follows:

{CAPACITY: | C:} *maximum capacity* [(*blocking factor*)]
[,*initial capacity* [,*increment*]];

Dynamic Expansion and Placement of a New Entry

The instant when the expansion is triggered and how record address is assigned to the new entry in a master data set varies greatly from detail data set. In a detail data set, expansion takes place during DBPUT when the free entry count for the set is zero. In case of a master data set, expansion is triggered during DBPUT when there is no free (unoccupied) entry in the close proximity (within an internally-maintained number of blocks) of the calculated primary address of the new entry and the set is expandable. This means that an expansion takes place even when the free entry count is greater than zero.

Following the first expansion, the master data set can be perceived as having two areas, original area and expanded area, managed differently. The original area is managed in the same way as in releases prior to MDX, that is, the primary address for the new entry is calculated using the hashing algorithm and the secondary address is assigned according to the availability of the free entry, but with modifications in searching for a free entry. The use of the expanded area is controlled using the pointer to a delete chain head and the last entry used in the file (high-water mark) as in detail data sets. Subsequent expansions belong to the expanded area. The original area can have primary entries and secondary entries, while the expanded area can only have the secondary entries. Note that the primary capacity (also hashing or initial capacity) used to calculate the primary address does not change. In essence, the dynamic expansion results in allocating additional space for secondary entries which implicitly makes room for additional primary entries in the original area. The primary address of a new entry is calculated using the value of the entry's key item and primary capacity in the hashing algorithm. The new entry can reside either at its primary address in the original area, or in the close vicinity of the primary address in the original area, or in the expanded area.

To elaborate further, when a DBPUT for a manual master, or implied DBPUT to an automatic master, is processed, the address where the new entry resides is based on the following:

1. If the primary address in the original area is not occupied, the new entry resides there. The new entry also becomes a synonym chain head with a count of one.

2. If the primary address is already occupied by a primary entry which hashed to the same address, TurboIMAGE/XL scans a few blocks (predetermined based on internal values, also known as quick search) in the close proximity of the primary address to find a free entry (empty entry). If found, the new entry resides at this secondary address in the original area. Otherwise, search for a free entry is directed towards the expansion area. At this point, if the set has not been expanded at all and can be expanded, expansion will take place and the new entry will reside at the first entry address in the new expanded area.

   If prior expansion has already taken place, the pointer to the delete chain head and the last entry used in the file for the expansion area are interrogated to determine the secondary address in the expansion area for the new entry. When there is room, the new entry is assigned a secondary address in the expanded area, and the new entry becomes the last entry in the synonym chain. If the expanded area is full and can be expanded, it is further expanded to accommodate the new entry.

When there is no room in the expanded area and cannot be expanded further, however, there is room in the original area based on the free entry count, the original area is scanned once again (long search) to find a secondary address for the new entry. If a free entry is found, the new entry resides there, and it becomes the last entry in the synonym chain. Otherwise, the set is full and DBPUT will fail.

3. If the primary address is occupied by a secondary entry, the secondary entry is relocated to another secondary address in either the original area or the expansion area. A free entry is searched as described in item 2 above. The new entry (primary) becomes a synonym chain head with a count of one. TurboIMAGE/XL performs synonym chain maintenance for the secondary entry which relocates.

Following the expansion, TurboIMAGE/XL updates various fields related to the expansion such as current capacity, including the expansion, in the root file. Also, the data set user label is updated for the last entry used in the file (high-water mark), free entry count, and pointer to the put-delete chain head.

Intrinsics and utilities. The following intrinsics and utilities are enhanced to support MDX:

- DBINFO modes 205 and 208 (new for MDX) give detailed information pertaining to dynamic expansion. The buffer layout for mode 205 remains the same as that of DDX except that it is relevant to MDX. The buffer layout for new DBINFO mode 208 is in the following table.

    Buffer   must be at least a 64-byte record and returns the following (each element is a 32-bit word):

Element	Contents
1	Primary (hashing) capacity for masters, 0 for details
2	Current capacity, including expansions
3	Maximum capacity
4	Expansion threshold: -1 ... 100 percentage -1 this value means expansion is not triggered by percentage, or it is a non-expandable set.
5	Delete chain free head (0 for non-expanded masters)

Element	Contents
6	High-water mark (0 for non-expanded masters)
7	Expansion threshold: -1 ... 2 billion blocks -1 this value means expansion not triggered by traversing 3 blocks without success, or it is a non-expandable set.
8…16	Reserved; 0 is returned

DBINFO mode 208 returns information about internals which will be meaningful to only a few customers.

- DBUTIL SHOW *database* ALL command shows the presence of MDX. The SHOW database CAPACITY command gives detailed information pertaining to capacity parameters and identifies sets enabled for dynamic expansion.

   For more information, refer to the sixth edition of the *TurboIMAGE/XL Database Management System Reference Manual.*

NOTE   Select master data sets which need dynamic expansion based on anticipated growth and specify the capacity parameters (in particular the initial capacity) very carefully. Specifying low initial (primary/hashing) capacity and increment leading to frequent expansions may result into long synonym chains, which degrade the performance of some TurboIMAGE intrinsics, and disk fragmentation. Select initial capacity such that expansions will be minimal.

80 GB Data Sets

TurboIMAGE/XL version C.07.14 includes an enhancement which will allow jumbo data sets to be up to 80 GB in size, twice as big as the old limit. There are no new externals/options added for this enhancement. For new databases, DBSCHEMA will allow data sets to be up to 80 GB in size. As for jumbo data sets, the control option '$CONTROL JUMBO' is required prior to specifying a data set greater than the default of 4 GB. For existing databases, use third-party tools which support this feature. If you use third-party indexing software, you may need the version of the software which supports this feature.

When creating a jumbo data set of this magnitude using DBUTIL, you will experience some delay.

Performance Enhancement for Databases with Third-party Indices (TPI)

TurboIMAGE/XL intrinsics DBGET and DBFIND are enhanced for improved performance in the following circumstances when the database is enabled for TPI:

- DBFIND and DBGET when the set does not have TPI and the trace facility, if applicable, is turned off.

- Serial DBGET (modes 2 and 3) for a set with TPI when the mode need not be promoted to other mode (possibly a chained mode) by the third-party software.

- Chained DBGET (modes 5 and 6) for a DBFIND of an IMAGE search item or a B-Tree item for a set with TPI.

In order to attain the performance gain, you must obtain the version of the third-party software which supports this enhancement. In the absence of either one of the needed software products (TurboIMAGE/XL or third-party product), DBFIND and DBGET will continue to work as they do in the absence of this enhancement. That is, TurboIMAGE/XL and the third-party products are backward compatible for this enhancement.

DBSCHEMA and P Type with Odd Sub-Item Length

DBSCHEMA is enhanced to generate an error when the sub-item length of the datatype P is odd, even if the product of sub-item count and sub-item length is divisible by 4. For example, DBSCHEMA will flag the data items 4P1 or 8P3 as invalid. Although syntactically correct, none of the major languages can use them. This change will not affect existing databases. That is, if you currently have databases with P type defined using odd sub-item length and sub-item count such that their product is divisible by 4, you may continue to use it without any problem when creating a new database. The default for DBSCHEMA is to generate an error in this situation. If you want to create a new database with P data type having odd sub-item length, you will need to use the new CONTROL option, $CONTROL ODDPALLOWED.

B-Tree Indices

You can create a B-Tree index on the master data set's key item and perform B-Tree index searches using the key item as well as all of its corresponding detail data set search items. The B-Tree index searches are done using DBFIND with a master or detail data set and its key or search item respectively. The qualified entries can be retrieved using DBGET mode 5 or 6.

Key Features of B-Tree Indices

Following are key features of B-Tree indices:

1. B-Tree index can be created only on the key item of the master data set.

2. B-Tree searches using the search items of all of its corresponding detail paths can be done.

3. You may create B-Tree indices for zero, one, or more master data sets.

4. `DBSCHEMA` has a new option, `INDEXED`, for the `SET` specification.

   New syntax

   ```
   NAME:setname,{M[ANUAL] | A[UTOMATIC]}[/INDEXED]
   [(read class list)/(write class list)][,device class];
   ```

5. `DBUTIL` has new commands and options.

   • New `ADDINDEX`, `DROPINDEX`, and `REBUILDINDEX` commands (to be used for one, more than one, or all masters).

   Syntax

   ```
   {ADDI[NDEX] | DROPI[NDEX] | REBUILDI[NDEX]}
   database name[/maintword] FOR {ALL | setnamelist | setnumlist}
   ```

   Parameters

   setnamelist   is the list *setname*[,…]

   setnumlist   is the list *setnum*[,…]

   ALL   means all master data sets for the database.

   • The `SET` command has a new `BTREEMODE1` option to set `DBFIND` mode 1 access `ON` or `OFF` for a B-Tree wildcard search for X and U types. The default is `OFF`. The `ON` setting allows you to use B-Tree indices for generic search without making application changes. Also, it allows you to define your own database-wide wildcard character.

   New Syntax

   ```
   SET database name[/maintword]
   BTREEMODE1={ON | OFF}[,[WILDCARD=]c]
   ```

   where *c* is any printable ASCII character, and the default character is @.

   • The `SHOW` command has a new `INDEX`, `INDEXES`, or `INDICES` option.

   New Syntax

   ```
   SHOW database name[/maintword]
   {INDEX | INDEXES| INDICES}
   ```

- CREATE, ERASE, PURGE, SECURE, RELEASE, and SHOW commands include B-Tree index files in their operation.

- The MOVE command does not allow moving the index file.

6. DBCONTROL has these modes pertaining to B-Tree indices:

13	is for B-Tree index file control. That is, to ADD, DROP, REBUILD, ATTACH, or DETACH a B-Tree index file. This mode requires PRIVILEGED mode, and hence, must be used carefully as improper use may damage your system.
14	is used to obtain and control database-wide B-Tree information. This mode requires PRIVILEGED mode, and hence, must be used carefully as improper use may damage your system.
15	sets BTREEMODE1 ON, and optionally allows the wildcard character to be set for the current DBOPEN.
16	sets BTREEMODE1 OFF for the current DBOPEN.

7. DBINFO has these modes pertaining to B-Tree indices:

209	informs whether or not a B-Tree index exists for a master.
113	gives BTREEMODE1 setting and the wildcard character for the database as well as current DBOPEN.

8. DBFIND has these features pertaining to B-Tree indices:

- Can be used for details as well as masters to specify B-Tree index searches.

- Introduces a new fixed-format structured argument.

- DBFIND modes with added functionality pertaining to B-Tree indices:

1	can continue to work as it does in releases prior to the release of B-Tree index feature (C.07.00), despite the presence of a B-Tree index, or it can be used for a B-Tree search if BTREEMODE1 is ON. Uses the classic mode 1 argument format.
4	used for B-Tree index searches on numeric as well as ASCII types and, for details, it returns accurate chain (super-chain, that is, multiple qualified chains) counts. Requires a new structured argument.

10   allows you to simulate `DBFIND` mode 1 as if there were no B-Tree index. This is the same as TPI mode 10. Requires the classic mode 1 argument format.

21   is the same as B-Tree index search using `DBFIND` mode 1, except it is a faster version and does not return accurate chain counts. Requires the classic mode 1 argument format.

24   is the same as B-Tree index search using `DBFIND` mode 4, except it is a faster version and does not return accurate chain counts. Requires a new structured argument.

Allows wildcard search, as well as range search (`<`, `<=`, `>`, `>=`, `"PK"` for partial key, or `[ ]` for between).

9. `DBGET` modes 5 and 6 can be used for B-Tree index retrieval for masters or details. Super-chains are traversed for detail data sets.

10. KSAM/iX files are used for B-Tree index files. These are KSAM related key points:

   - There is one KSAM/iX privileged file with a filecode of -412 for each B-Tree index file. The size limit for this B-Tree index file is 4 GB. A jumbo master (larger than 4 GB) can have a B-Tree index file provided the B-Tree index file remains within its 4 GB limit.

   - KSAM/iX B-Tree index file is named using the POSIX file format with the "idx" extension (lowercase). For example:

     `/ACTSALES/GRPSALES/ORDERS03.idx`

   - The KSAM file has the Native Language Support language specified to match the language of the database, if the key is an ASCII type.

11. Third-party index can coexist with B-Tree index, that is, on the same item of the data set.

Quick Start for Using B-Tree Indices

If you are interested in generic trailing-@ searches only and want to get started quickly without making any application changes, you may use the following steps.

1. Identify the masters with ASCII key item that can benefit from B-Tree indices.

2. Create B-Tree indices using either of these methods:

   a. Use the `INDEXED` option of `DBSCHEMA` for new databases.

   b. Use the `ADDINDEX` command of `DBUTIL` for existing databases.

3. Set the `BTREEMODE1` option `ON` using `DBUTIL` as follows:

```
:Run DBUTIL.PUB.SYS

>SET database name [/maintword] BTREEMODE1=ON
```

If your database is new, you will need to add data. You are now ready to perform B-Tree index searches. You can include the wildcard character in your DBFIND argument and observe the results.

Scalability

Prior to this enhancement, the modification intrinsics, DBPUT, DBDELETE, and DBUPDATE (Critical Item Update feature ON) were serialized for the database. That is, only one such intrinsic could execute at a time for the database. In order to scale with the high-end machines, TurboIMAGE/XL is enhanced to increase the concurrency of these modification intrinsics.

The increased concurrency is based strictly on the database design. The database is internally grouped into independent SUB-DATABASES based on the physical relationships of master and detail data sets. Subsequently, the modification intrinsics can execute concurrently for these independent subdatabases. The throughput is dependent on the number of subdatabases, the more the better. The worst case is a database with only one subdatabase, one detail linked to 16 masters. The best case is when there are numerous stand-alone masters or details.

To use this enhancement, your database must be activated to use the Dependency Semaphore. To achieve this, the ENABLE and DISABLE commands of DBUTIL introduce a new option, DSEM. The default for DSEM is DISABLED.

New syntax

EN[ABLE] *database name*[/*maintword*] FOR DSEM

DI[SABLE] *database name*[/*maintword*] FOR DSEM

After enabling your database for DSEM, if you feel that your database design cannot attain expected concurrency, you may DISABLE the feature as there is some overhead when the database is enabled for DSEM.
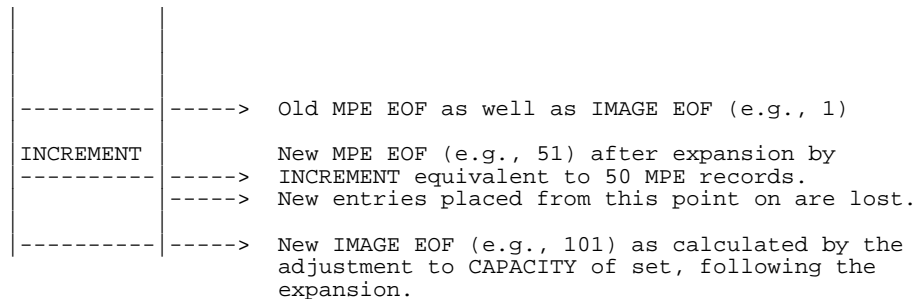
Dynamic Detail Dataset Expansion (DDX) Fix

(SR# 5003-367607)

The DDX feature of TurboIMAGE/XL has a known problem which *may* cause you to lose *some* of your new data entries added following the dynamic detail dataset expansion. This defect has been in the feature since its introduction, but has surfaced recently and has been reported by a few customers. The problem has been fixed both for MPE/iX 5.5 and MPE/iX 6.0. The patch ID for MPE/iX 6.0 is TIXKX62 (or superseded patch id) and the TurboIMAGE/XL version is C.07.10. This patch is contained in this release.

Problem Description

The problem is that, following a dynamic expansion, the addition of the new data entries (DBPUTs), which *exceed* the INCREMENT in number, are written to the data set beyond the physical end-of-file (MPE EOF). When the database is closed and reopened, those new entries exceeding the INCREMENT are inaccessible. If a program attempts to read that data or add more data entries in that area, it gets an error -212, `Database is Corrupt`. The following diagram illustrates the problem.

```
    |          |
    |          |
    |          |
    |          |
    |----------|-----> Old MPE EOF as well as IMAGE EOF (e.g., 1)
    |          |
 INCREMENT     |       New MPE EOF (e.g., 51) after expansion by
    |----------|-----> INCREMENT equivalent to 50 MPE records.
    |          |-----> New entries placed from this point on are lost.
    |          |
    |----------|-----> New IMAGE EOF (e.g., 101) as calculated by the
                       adjustment to CAPACITY of set, following the
                       expansion.
```

The real problem is that, for certain expansions, the new CAPACITY is inadvertently adjusted by adding the INCREMENT *more than once* instead of adjusting only *once*. This results in incorrect calculation of IMAGE EOF and subsequently allows new data entries to be placed in the area which really does not belong to the data set file. Therefore, when the data set is closed, the new entries added beyond the MPE EOF are not retained as part of the data set file by the MPE file system and are subsequently lost.

The circumstances in which such an erroneous adjustment could take place are as follows:

- While an actual DDX is being performed by DBPUT for one process, a second process (user) accessing the same data set for the first time causes execution of IMAGE's open data set module. In a narrow timing window, the result will be incorrect extraneous adjustment of the CAPACITY by adding the INCREMENT! Specifically, the CAPACITY has been incremented *twice*, while the MPE EOF has been correctly incremented—only *once*.

How to tell if you already have this problem. If you use DDX, check each detail data set enabled for DDX. Remember, the problem is associated only with the detail data sets which have undergone dynamic expansion.

You can check for this condition as follows:

1. Use the FORM SETS command of QUERY, which gives the Current Capacity (CC) and Blocking Factor (BF) of each data set.

```
:query
HP32216D.03.11  QUERY/3000  TUE, NOV  4, 1997,  3:20 PM
COPYRIGHT HEWLETT-PACKARD CO. 1976

>b=dbusa
PASSWORD = >>
MODE = >>1
>form sets

DATA BASE: DBUSA                             TUE, NOV  4, 1997,  3:20 PM

DATA BASE LANGUAGE ATTRIBUTE: NATIVE-3000

                 ITEM    CURRENT   ENTRY   ENTRY   BLOCKING
SETS:      TYPE  COUNT   CAPACITY  COUNT   LENGTH  FACTOR

   PEPU01  D     3       1010      11      12      10
```

2. Calculate the IMAGE EOF of a DDX data set using the following formula:

```
(CC + (BF-1))/BF
```

For example:

```
IMAGE EOF = (1010 + (10-1))/10 = 101
```

3. Verify this calculated EOF with the MPE EOF given by the command :LISTF *dbnamenn*,2 where *dbnamenn* is the file name of that DDX data set.

```
:listf dbusa01,2

ACCOUNT=  TESTACCT    GROUP=  DDX

FILENAME   CODE   -----------LOGICAL RECORD----------  ----SPACE----
                  SIZE  TYP        EOF     LIMIT R/B  SECTORS #X MX

DBUSA01    PRIV   256W  FB         51       200   1    112   2  4
```

4. If the two EOFs do not match, there is a problem.

   In the example, MPE EOF of 51 and IMAGE EOF of 101 do not match—the problem exists.

   The SHOW DBUSA CAPACITY command of DBUTIL gives capacity parms:

```
                No. of  %Max -----------Capacity------------- Dyn
Data Set Name Type Entries Cap Maximum Current Initial Increment Exp:

PEPU01        D    11      1   2000    1010    10      500       YES
```

Based on the above formula for calculating MPE records, the initial capacity was 1 MPE record ((10+9)/10), and the increment was 50 MPE records ((500+9)/10). The correct physical EOF should be 51 (1+50). However, IMAGE EOF as shown by QUERY is 101 (1+50+50). This establishes that the IMAGE EOF was adjusted twice.

If you use one of the IMAGE/SQL structure maintenance tools (such as Adager, DBCPLUS, or DBGENERAL), it may have the capability to locate instances of this problem, and if you discover that one of your data sets is so afflicted, it may be possible to correct the problem using the same tool. (See the documentation for your tool.)

How to avoid experiencing this problem on your IMAGE database. The simplest answer is to install and use the TurboIMAGE/XL patch mentioned above. This is the option that HP recommends.

However, if you are unable to acquire and install the patch immediately, consider the following alternative:

• Until you install the patch, avoid opening a data set from one process while another is expanding it!

Here are some ways you might accomplish this; each way is independent of the others:

1. Preallocate and pre-expand DDX data sets which are *nearly-full* by adding, in exclusive mode, many new dummy entries. Add more than your actual processing will add. Then delete them before your real processing begins.

2. Open all DDX-enabled data sets needed from each process which accesses them before allowing any process to add data to any of these sets.

3. Access the database exclusively, if possible, until the expansion has completed.

4. Disable DDX on any nearly-full data set, using an HP or third-party tool.

Important Information

1. The default for Critical Item Update option was changed from `DISALLOWED` to `ALLOWED` starting with Express 3 for 5.5.

2. A jumbo data set cannot be enabled for dynamic expansion.

3. `PURGE` and `ERASE` commands of `DBUTIL` are enhanced to prompt the user in session mode only (not batch) for confirmation (starting with version C.07.07).

4. If you want the TPI performance enhancement, you also need to get the third-party indexing software which supports this enhancement.

IMAGE/SQL Enhancements

Information regarding enhancements to IMAGE/SQL for SQL access is described in this section:

- Predicate level lock when updating masters
- Support for B-Tree indices
- Support for third-party composite indices
- New ODBC driver, ODBCLink/SE

Predicate Locks when Updating Manual Masters

IMAGE/SQL version B.G2.07 is enhanced to apply predicate level locks (row level locks), instead of a set level lock as in prior versions, when updating selected entries in the manual master data set.

When IMAGE/SQL is used as the SQL interface to TurboIMAGE/XL database, the responsibility of enforcing appropriate locks rests with IMAGE/SQL. These locks should conform to the locking schemes present in TurboIMAGE/XL. Accordingly, locks are set by IMAGE/SQL depending on the DBOPEN modes and isolation level requirements. At present, for a master dataset, for all modify intrinsics, that is, insert, delete and updates, and for DBOPEN modes 1 to 4, IMAGE/SQL puts a set level lock (or a pseudo set level lock covering all the rows in the set). But for update function in DBOPEN mode 1, setting a set level lock (or pseudo set lock) is not mandatory, a row level lock would be sufficient. Therefore, IMAGE/SQL will now lock at the predicate level if a 'where' clause is specified, otherwise a set lock will be given. This feature is actually transparent to the user.

Benefits

When multiple users are operating on the same data set, it is difficult to update different entries with set level lock in place. But when a predicate level lock is in effect, the unnecessary wait time is avoided, thus allowing multiple updates of different entries on the same data set.

For an example, User A has an update statement:

```
UPDATE Music.Albums set RecordingCo='ABC Music Ltd.' where Medium='ca';
```

The kind of lock that is placed can be seen by invoking DBUTIL as shown:

```
:DBUTIL

HP30391C.07.04 TurboIMAGE/XL:  DBUTIL (C) COPYRIGHT HEWLETT-PACKARD COM
PANY 1987

>>show Music locks
   For database MUSIC
                                               PIN/      PROGRAM
      LOCKED ENTITY  /  ( - waiting process )   PATH     NAME    JOBNUM

ALBUMS: MEDIUM = ca . . . . . . . . . . . . . . . . 41/1    ISQL    #S18
```

Now, User B wants to update row(s) in the same data set via an update statement:

```
UPDATE Music.Albums set RecordingCo='California Music' where
Medium='cd';
```

The kind of lock that is applied by User B (JOBNUM #S21) is shown in the following example:

```
:dbutil

HP30391C.07.04 TurboIMAGE/XL:  DBUTIL (C) COPYRIGHT HEWLETT-PACKARD
COMPANY 1987

>> show Music locks
   For database MUSIC
                                               PIN/      PROGRAM
      LOCKED ENTITY  /  ( - waiting process )   PATH     NAME    JOBNUM

ALBUMS: MEDIUM = cd . . . . . . . . . . . . .   56/1    ISQL    #S21
ALBUMS: MEDIUM = ca . . . . . . . . . . . . .   41/1    ISQL    #S18
```

In this example, User A updates 2 entries and User B updates 5 entries simultaneously.

Impact on current IMAGE/SQL applications

With this enhancement, the locking scheme in IMAGE/SQL is exactly as required by TurboIMAGE/XL (as described in TurboIMAGE/XL Database Management System Reference Manual, Aug 1997). All applications conforming to this will not have any problems, and may actually see improvements in performance, when updating manual masters.

Support for B-Tree Indices

The B-Tree indices can be created using DBUTIL for existing databases and DBSCHEMA for new databases. More details about creating B-Trees indices is given under the heading, "TurboIMAGE/XL Enhancements," earlier in this article.

In order to realize the benefit of the B-Tree indices for SQL access, you must first DETACH the database, if already attached, and ATTACH the database for the DBEnvironment. IMAGE/SQL enters definitions for the B-Tree indices in the system catalog of the DBEnvironment. The index on the key item of the master, except for P and Z types, is entered as 'unique' index. Other definitions, by default, are non-unique.

Although the B-Tree index on the related search item of the detail set is not explicitly created using `DBUTIL` or `DBSCHEMA`, its definition is entered in the SQL catalog. For example, if a B-Tree index is created on a key item with paths to 16 detail data sets, a definition for a B-Tree index for all 16 data sets will be entered as well. That is, this will result into 17 (1+16) definitions, one for each data set.

If your key item or its related search item is split using the `SPLIT` command of IMAGESQL, the definition for a B-Tree index on the split item will not be entered. For an example, if a key item is split but the related search item is not, the definition for the key item of the master data set will not be entered, but the one for the search item will be.

The B-Tree indices can be viewed in the views, `SYSTEM.INDEX` and `CATALOG.INDEX`, of the system catalog of the DBEnvironment.

Multiple index definitions on the same column can coexist and the SQL optimizer derives the access plan based on the statistics present in the system catalog. In other words, the key or search item of the set can have a maximum of three index definitions. One will be a hash index (only "=" operator permitted) automatically done by `IMAGESQL` at `ATTACH` time, another can be a B-Tree index, and the third can be a third-party index. It is recommended that both B-Tree index and third-party index be not created on the same item as it will unnecessary impact the performance (Optimizer calculates cost for each index).

The Optimizer derives an access plan and decides which index to use and the proper order of operations.

The version of third-party software that supports the new B-Tree index modes for `DBFIND`, `DBCONTROL`, and `DBINFO`, is required from both third parties.

Third-Party Composite Indices

IMAGES/QL is enhanced to enter definitions for third-party composite indices in the system catalog of the DBEnvironment at `ATTACH` time. These composite indices can be on mixed data types, but must be on FULL items. Information about all third-party indices, including composite, is obtained from the third-party product, which is subsequently used during `ATTACH`. Both SUPERDEX and OMNIDEX do not provide information on the composite indices on substrings (partial item) of items to IMAGE/SQL.

These indices can be viewed in `SYSTEM.TPINDEX` along with other third-party indices.

At run-time, IMAGE/SQL may employ `DBFIND` mode 1 with "@;" appended to the argument, or mode 11 with start and stop values, as deemed appropriate.

Note that if you have multiple indices for the table, the SQL optimizer has the control on the type of access, and the specific index to be used.

For SQL access, the only step necessary to use the above enhancements pertaining to relational access, is to perform `DETACH`, if attached, and `ATTACH`.

Important Information

1. If you have third-party indexing (TPI) enabled for the database and want to create B-Tree indices, obtain the correct version of the third-party software which recognizes this coexistence. Otherwise, you will get run-time errors returned by third-party software for new modes of `DBFIND`, `DBINFO`, and other utility commands.

2. The Optimizer derives an index scan based on the current statistics for the data set. Hence, it may select an index on the column not used in the SQL statement.

ODBCLink/SE

In this release, IMAGE/SQL is bundled with a new 16-bit and 32-bit ODBC driver, ODBCLink/SE, which is a replacement for the ODBC driver HP PC API. ODBCLink/SE is an implementation of Microsoft's Open Database Connectivity (ODBC) interface that enables Microsoft Windows based applications and tools to access TurboIMAGE/XL data on the HP3000 in a client/server environment. In such an environment, the application developers and end-users can take advantage of the PC's graphical user interface (GUI) and processing power, while relying on the security, integrity, and database management capabilities of IMAGE/SQL. The client can run under Microsoft Windows 3.1 or 3.11, Windows95, or Windows NT V3.51 or V4.0. Connection via Winsock is available in 16-bit and the new 32-bit version. The driver can be used in two ways: either by direct calls from a Windows program or through an ODBC-compliant application such as Microsoft Access and Visual Basic.

A data migration tool is provided to migrate data sources created for the current HP PC API to ODBCLink/SE. Once the data sources have been identified, the translation is performed automatically.

The ODBCLink/SE server runs on MPE/iX 5.0 or later releases. ODBCLink/SE is ODBC Level 2 compliant with a few exceptions. For more information, refer to the article, "Introducing ODBCLink/SE," in this *Communicator 3000*. The *ODBCLink/SE Reference Manual* is bundled with IMAGE/SQL.

New Features in ALLBASE/SQL Version G3

by Doug Myers
Database Lab, Commercial System Division

Overview

This article includes information on the following features of
ALLBASE/SQL included with this version G3 on MPE/iX 6.0 release
(C.60.00):

- ALLBASE/SQL manual set is on Instant Information CD-ROM

- String functions are a new feature for ALLBASE/SQL G3

- Security features introduced with ALLBASE/SQL G2

- Year 2000 compatibility introduced with ALLBASE/SQL G2

- Manuals updated

- CAST Function introduced with G1.15

Instant Information CD-ROM

The entire manual set for ALLBASE/SQL is now available on the new
CD-ROM documentation system, "Instant Information." You can now
look up information quickly in the on-line documentation.

String Functions

With the G3 release of ALLBASE/SQL and IMAGE/SQL, the supported
SQL syntax has been enhanced to include the following string
manipulation functions: `UPPER`, `LOWER`, `POSITION`, `INSTR`, `TRIM`,
`LTRIM` and `RTRIM`. These string functions allow you to manipulate or
examine the `CHAR` and `VARCHAR` values within the SQL syntax,
allowing for more sophisticated queries and data manipulation
commands to be formed. These string functions were designed to be
compatible with functions specified in the ANSI SQL '92 standard and
functions used in ORACLE. In cases where the ANSI SQL '92 standard
and the ORACLE functions were not compatible (such as the `LTRIM`
and `RTRIM` in ORACLE versus `TRIM` in the ANSI standard), both
versions where implemented. The specifications for each of these
functions follows.

Function Specification

Lower

Converts all the characters in *stringexpr* to lower case

Syntax . [LOWER (*stringexpr*)]

Upper

Converts all the characters in *stringexpr* to upper case

Syntax . [UPPER (*stringexpr*)]

Position

Searches for the presence of the string *stringexpr1* in the string *stringexpr2* and returns a numeric value that indicates the position at which *stringexpr1* is found in *stringexpr2*

Syntax . [POSITION (*stringexpr1*, *stringexpr2*)]

Instr

Searches *char1* beginning with its *n*th character for the *m*th occurrence of *char2* and returns the position of the character in *char1* that is the first character of this occurrence. If *n* is negative, Instr counts and searches backward from the end of *char1*. The value of *m* must be positive. The default values of both *n* and *m* are 1, meaning Instr begins searching at the first character of *char1* for the first occurrence of *char2*. The return value is relative to the beginning of *char1* regardless of the value of *n*, and is expressed in characters. If the search is unsuccessful (if *char2* does not appear *m* times after the *n*th character of *char1*) the return value is 0.

If *n* and *m* are not specified the function is equivalent to the ANSI SQL-92 POSITION function, except that the syntax is slightly different.

Syntax . [INSTR (*char1*, *char2* [,*n* [,*m*]])]

Ltrim

LTRIM function trims the characters specified in *charset* from the beginning of the string *stringexpr*.

Syntax . [LTRIM (*charset*, *stringexpr*)]

Rtrim

RTRIM function trims the characters specified in *charset* from the end of the string *stringexpr*.

Syntax . [RTRIM (*charset*, *stringexpr*)]

Trim

TRIM function allows you to strip the characters specified in *charset* from the beginning and/or the end of the string *stringexpr*. If *charset* is not specified, then blank characters would be stripped from *stringexpr*.

Syntax.
```
[ TRIM ({ LEADING | TRAILING | BOTH} (,charset ,stringexpr)]
```

Examples:

Example 1

```
SELECT LOWER (OWNER) || '.' || LOWER (NAME)
FROM SYSTEM.TABLE
WHERE NAME = UPPER ('vendors');
```

Returns "purchdb .vendors "

Example 2

```
SELECT POSITION ('world', 'hello world')
FROM SYSTEM.TABLE
WHERE NAME = UPPER('vendors');
```

Returns the numeric value 7

Example 3

```
SELECT INSTR ('hello world hello world', 'world', 5, 2)
FROM SYSTEM.TABLE
WHERE NAME = UPPER('vendors');
```

Returns the numeric value 18 (starting position of the second occurrence of the string 'world')

Example 4

```
SELECT * FROM SYSTEM.TABLE
WHERE NAME = LTRIM ('?*', 'VENDORS?*???***')
AND OWNER = 'PURCHDB';
```

Returns the system table entry for PURCHDB.VENDORS

Example 5

```
SELECT TRIM (BOTH '?*' FROM '??**?*hello ?* world???*')
FROM SYSTEM.TABLE
WHERE NAME = 'VENDORS';
```

Returns 'hello?* world'.

Features in ALLBASE/SQL Version G2

ALLBASE/SQL, version G2, contains several major enhancements providing significant benefits in the following areas:

- Security

- Year 2000 compatibility

- CAST function (introduced in G1.15)

In addition, the following manuals were updated to fully document G1 and G2 features:

- *ALLBASE/SQL Reference Manual*

- *ALLBASE/SQL Database Administration Guide*

- *ALLBASE/SQL Message Manual*

Security

Two new authorities have been added to the G2 release of ALLBASE/SQL: MONITOR and INSTALL. These new authorities can be granted to users using the GRANT command, or can be revoked from users using the REVOKE command.

MONITOR Authority

MONITOR authority gives a user the ability to run SQLMON, an on-line diagnostic tool that monitors the activity of an ALLBASE/SQL DBEnvironment. Before this new authority, SQLMON could only be run by the creator of the DBEnvironment (also known as the DBECreator), someone who knows the maintenance word (maintword), or someone who has superuser capability. See the *ALLBASE/SQL Reference Manual* for more details.

INSTALL Authority

INSTALL authority gives a user the ability to use the SQL preprocessors to create an SQL module owned by someone other than themselves, or to use ISQL to install a module owned by someone other than themselves. This is an important capability for those needing to move modules from a development DBEnvironment to a production DBEnvironment. Before this new authority, only a DBA had this ability. See the *ALLBASE/SQL Reference Manual* for more details.

Year 2000 Compatibility

As the end of the century quickly approaches, there has been much concern about dates being handled correctly in the twenty-first century. This is particularly an issue for systems and applications that are using a two-digit year to express dates. ALLBASE/SQL has always stored dates in a four-digit year format which largely eliminates the year 2000 compatibility problem. ALLBASE/SQL applications can be coded using four-digit year dates, thus avoiding any ambiguity. However, for existing applications that may allow dates to be entered using a two-digit year format, there is still an issue as to how these dates get converted to the ALLBASE/SQL internal format using the TO_DATE function or TO_DATETIME. For example, what date value would the following functions return?

```
TO_DATE('12/31/35','MM/DD/YY')
TO_DATETIME('351231235959','YYMMDDHHMISS')
```

Would these functions evaluate the year as 1935 or 2035?

Solution

When YY is specified in the format specification of either `TO_DATE` or `TO_DATETIME`, and if its value of the year specified is less than 50, then the century part of the `DATE` or `DATETIME` defaults to 20, otherwise it is set to 19. This behavior can be overridden by setting the JCW `HPSQLsplitcentury` to a value between 0 and 100. If the year specified is less than the value of the JCW `HPSQLsplitcentury` then the century part is set to 20, otherwise it is set to 19.

For the above example,

Case 1: HPSQLsplitcentury is not set

```
TO_DATE('12/31/35','MM/DD/YY') = 2035-12-31
TO_DATETIME('351231235959','YYMMDDHHMISS') = 2035-12-31 23:59:59
```

Case 2: HPSQLsplitcentury is set to 0

```
TO_DATE('12/31/35','MM/DD/YY') = 1935-12-31
TO_DATETIME('351231235959','YYMMDDHHMISS') = 1935-12-31 23:59:59
```

Case 3: HPSQLsplitcentury is set to 70

```
TO_DATE('12/31/35','MM/DD/YY') = 2035-12-31
TO_DATETIME('351231235959','YYMMDDHHMISS') = 2035-12-31 23:59:59
```

CAST Function

With the G1.15 release, the `CAST` function enhancement for ALLBASE/SQL and IMAGE/SQL became available to customers. The `CAST` function is used to explicitly convert data from one data type to another. The `CAST` function not only allows conversion between compatible data types, such as between `CHAR` and `BINARY` or between `INTEGER` and `DECIMAL`, but it will also allow conversion between certain normally incompatible types, such as between `CHAR` and `INTEGER`.

The `CAST` function is defined in the ANSI SQL2 standard. `CAST` in ALLBASE/SQL and IMAGE/SQL complies with that standard. In addition, several extensions to the standard specification have been added to make `CAST` even more powerful.

The `CAST` function can be used anywhere a general expression is allowed. Also, as a part of this enhancement, the SQL parser has been enhanced to allow general expressions in more of the SQL syntax. For example, general expressions including nested functions are now allowed in all the date/time functions and string functions. Therefore, `CAST` will be supported inside functions that support expressions including aggregate functions. `CAST` will also take general expressions including nested functions as input.

CAST Syntax

`{CAST ({`*Expression* `NULL}{AS,}`*DataType*`[,`*FormatSpec*`])}`

Parameters

Expression   Column, USER function, host variable, local variable, AddMonths function, aggregate function, date/time conversion function, dynamic parameter, or procedure parameter, constant, current function, long column function, string function, or any combination of these in an arithmetic or concatenation expression.

DataType   ALLBASE/SQL data type: `CHAR(n), VARCHAR(n), DECIMAL(p[,s]), FLOAT, REAL, INTEGER, SMALLINT, DATE, TIME, DATETIME, INTERVAL, BINARY(n), VARBINARY(n).`

    The `LONG BINARY(n)` and `LONG VARBINARY(n)` cannot be used in the `CAST` operations.

FormatSpec   Format specification used for `DATE, TIME, DATETIME, INTERVAL` conversions. *FormatSpec* is the same as that used in the date/time conversion functions. See the *ALLBASE/SQL Reference Manual* for more details.

Pseudo-Mapped Files Removed from ALLBASE/SQL

by Poongodi R and Doug Myers
Commercial Systems Division

Introduction

This article provides information regarding the handling of pseudo-mapped files when installing ALLBASE/SQL. Pseudo-mapped files in MPE XL are disk files that are accessed without the usual file mapping by the operating system.

Pseudo-mapped files are no longer supported by ALLBASE/SQL. If the pseudo-mapped files already exist in the system, they will not be supported on the G.0 (or later) versions. Serious system problems may occur because of the use of pseudo-mapped files.

NOTE   The conversion from pseudo-mapped to mapped MUST be done on an F.0 version before updating to G.x. You cannot do the conversion after the system has been updated.

Converting Pseudo-Mapped Files to Mapped Files

ALLBASE/SQL files in pseudo-mapped mode must be converted into mapped mode using the MOVEFILE command in SQLUtil. Use the SHOWACCESS command in SQLUtil to show which files are pseudo-mapped. Use the following procedures:

1. Check for pseudo-mapped files as shown in this example:

   ```
   :sqlutil

   >> showaccess
   DBEnvironment Name: mydbe
   Maintenance Word:
   GGF0        mapped
   MYF1        pseudo-mapped
   MYF2        pseudo-mapped
   >>
   ```

2. Disable the pseudo-mapped feature using the SQLUtil MOVEFILE command as shown in this example:

   ```
   >> movefile
   DBEnvironment Name: mydbe
   File Name: MYF1
   Current Device: LDEV#2
   New Device (opt):LDEV#
   Access Mode (Mapped) (opt): mapped

   File moved.
   ```

   Repeat for other pseudo-mapped files listed by SHOWACCESS. The pseudo-mapped files are converted into mapped mode. Hence, there will be no problem involved with the converted files during the upgrade.

Year 2000 and Other Enhancements in VPLUS

by Kumar KN and Rajesh Channabasavaiah
Commercial Systems Division

Introduction

This article provides information on the following VPLUS enhancements available in version B.06.07.

- Turn-of-century enhancement

- Ability to define default Field Type and Data Type for a Form

- Ability to define more than 52 single character fields per Form

- New intrinsics to get and set Save Field values programmatically

- Ability to Renumber a Form with Interactive FORMSPEC

A description of these features follows.

Turn-of-Century Enhancement

VPLUS version B.06.07 contains enhancements to handle Year 2000 and beyond in applications. These enhancements can be grouped into three segments as follows:

- Intrinsic enhancements

- FORMSPEC enhancements

- Processing enhancements

Intrinsic Enhancements

Two new intrinsics have been added to VPLUS to handle dates in the next century. These are VGETYYYYMMDD and VPUTYYYYMMDD. These intrinsics are similar in structure and function to the existing intrinsics VGETYYMMDD and VPUTYYMMDD respectively. The difference is that the date values handled by the new intrinsics contain 4-digit year components.

Intrinsic VGETYYYYMMDD

Converts data in the data buffer to a eight-byte character value and copies it to an application.

Syntax

VGETYYYYMMDD {*comarea,fieldnum,variable*}

Chapter 10   **181**

Parameters

comarea   must be *comarea* name specified when the forms file was opened with `VOPENFORMF`. If not already set, the following *comarea* items must be set before calling `VGETYYYYMMDD`:

  cstatus   set to zero.

  comarealen   set to total number of two-byte words in *comarea*. Must be at least 70 words in length.

  `VGETYYYYMMDD` may set the following *comarea* item:

  cstatus   set to nonzero value if call is unsuccessful or requested field has an error.

fieldnum   is a two-byte integer variable which specifies the number assigned to the field by `FORMSPEC`

variable   is a character string in the application into which the converted value is placed

Discussion

`VGETYYYYMMDD` transfers the contents of the field specified by *fieldnum* to a variable in an application. The contents of the field are stored in a data buffer from which the value is taken. This value is converted to `YYYYMMDD` format and the converted value is placed in the application variable. The `YYYYMMDD` format is an 8-byte ASCII value, for example, `19961225`.

If errors occur during conversion, *cstatus* is set to an error code. If the requested field has an error, its value is moved to the variable but *cstatus* is set to an error code.

Refer to the `VGETtype` intrinsic description in the *Data Entry and Forms Management System VPLUS Reference Manual* for related information.

Example

```
COBOL
CALL "VGETYYYYMMDD" USING COMAREA FIELDNUM VARIABLE.

SPL
VGETYYYYMMDD(COMAREA,FIELDNUM,VARIABLE);
```

These calls will convert the contents of the data buffer corresponding to the field specified by `FIELDNUM` and place the converted value into `VARIABLE`.

Intrinsic VPUTYYYYMMDD

Converts a numeric value representing a date from an application and copies the converted value to a field in the data buffer in memory.

Syntax

VPUTYYYYMMDD {*comarea,fieldnum,variable*}

Parameters

comarea	must be *comarea* name specified when the forms file was opened with VOPENFORMF. If not already set, the following *comarea* items must be set before calling VPUTYYYYMMDD:

	cstatus	set to zero.
	comarealen	set to total number of two-byte words in *comarea*. Must be at least 70 words in length.

VGETYYYYMMDD may set the following *comarea* item:

	cstatus	set to nonzero value if call is unsuccessful or requested field has an error.
	numerrs	may be decremented if the new value replaces the value of a field with an error.

fieldnum	is a two-byte integer variable which specifies the number assigned to the field by FORMSPEC.
variable	is a character string in the application which contains the value to be converted.

Discussion

VPUTYYYYMMDD converts the contents of the application variable to the date order of the field specified by *fieldnum* and copies the converted value into the corresponding field in the data buffer, right justified. The application variable should contain a numeric value in YYYYMMDD format. The YYYYMMDD format is an 8-byte ASCII value, for example, 19961225. If errors occur during conversion, *cstatus* is set to an error code.

Refer to the VPUTtype intrinsic description in the *Data Entry and Forms Management System VPLUS Reference Manual* for related information.

Example

```
COBOL
CALL "VPUTYYYYMMDD" USING COMAREA FIELDNUM VARIABLE.

SPL
VPUTYYYYMMDD(COMAREA,FIELDNUM,VARIABLE);
```

These calls will convert the contents of `VARIABLE` to a date value and place the converted value in the data buffer corresponding to the field specified by `FIELDNUM`.

FORMSPEC Enhancements

`FORMSPEC` Application Ready Buffer (ARB) processing has been enhanced with the addition of a new ARB data type. This data type can be specified on the DTC Menu for Data Type Conversions from Screen Type to Application Type. The new ARB Type is `YYYYMD` and is defined as a 8-byte ASCII field containing a date value in `YMD` order with no separators. Also the year component in the date value has 4 digits. Note that `FORMSPEC` will ensure a length of 8 for the data type `YYYYMD` in the ARB. In essence, the value is of the form `YYYYMMDD` (8 bytes) even though the Type designator on the DTC Menu has been abbreviated to `YYYYMD`.

For more details on ARB Types and how to use them, refer to the "ARB Data Types" section in the *Data Entry and Forms Management System VPLUS Reference Manual*.

Processing Enhancements

Date processing in VPLUS has been enhanced to handle date values with 4-digit year components. Specifically, the following additions have been made.

Date output will be formatted with 4-digit years if the target field is wide enough to receive the date, and the user requests this action. Examples of such output include the `SET` statement in `FORMSPEC`. The target field should be at least 10 characters in length to have a 4-digit year output.

VPLUS, by default, will format date output with 2-digit years. To change the behavior as described above, define a JCW called `VSETNEXTCENTURY` and set bit 15 to 1 (bits are numbered from 0 through 15). It should be noted that this option will be active for fields wide enough to hold the 4-digit year date. For other fields, 2-digit years will be used.

Existing 2-digit years can be interpreted differently to handle dates in the next century. Currently VPLUS handles year components 00-99 as 1900-1999. This scheme of interpretation can be changed. In the new scheme, 00-49 is interpreted as 2000-2049 and 50-99 is interpreted as 1950-1999. This interpretation will be valid through all processing done

by VPLUS. For example, this scheme will allow the user to enter 02/29/00 (MDY order) in an existing application and have it processed correctly.

To invoke the above method of date interpretation, define a JCW called `VSETNEXTCENTURY` and set bit 14 to 1. Note that this method is used for processing only 2-digit year dates and not 4-digit year dates. Note also that the display of 2-digit year dates is not affected by this enhancement. This means that 12/11/10 (MDY order) can refer to 1910 in one application and 2010 in another application.

NOTE	When defining the JCW `VSETNEXTCENTURY`, set all unused bits to 0.

Programmable Defaults for Field and Data Types in FORMSPEC

`FORMSPEC` version B.06.07 has been enhanced to provide the capability to set the default Field Type and Data Type of a form. This capability can be used to reduce the time taken to complete the design of a new form.

Two new fields for user input have been added to the Form Menu screen of `FORMSPEC`. One is for the default Field Type, and the other is for the default Data Type.

On the Form Menu, the user typically specifies the name of the form and other characteristics of the form. In addition now, the user can specify the default Field Type and Data Type for the form. For a list of the allowable values for the Field Type and Data Type, refer to the *Data Entry and Forms Management System VPLUS Reference Manual*. All fields created newly for the form will have the Field Type and Data Type set to the default values supplied on the Form Menu.

Notes

- The Field Type and Data Type values supplied on the Form Menu are applicable only to fields created for the form thereafter. This means that if these default values are changed for an existing form, the types of the existing fields are not changed. For a new form, the default values apply to all the fields created for the form.

- Cloned forms have their screen already designed; hence, the values on the Form Menu do not apply to such forms.

- The Field Type and Data Type can be specified independently. This means that either, or both, fields on the Form Menu can be blank.

  - If the Field Type is blank, all newly created fields for the form will have a Field Type of "O."

- If the Data Type is blank, all newly created fields for the form will have a Data Type of "CHAR."

Ability to Define More than 52 Single Character Fields per Form

This enhancement allows more than 52 single character fields to be defined for a Form. Currently the field tag which is defined for a field at screen design (creation) time should begin with an alphabetic character. This enhancement will allow several other characters to be used for the field tag, thus increasing the number of single character fields that can be defined for a Form.

The following are the other characters which can be used for the field tag (defined at screen creation time):

Digits   0,1,2,3,4,5,6,7,8,9

Specials   @,#,$,%,&,*,-,+,<,>,/,\,!,|,=,?,:,;,_

Field Menu Initialization

In the case where one of the characters listed above (a Digit or a Special) is used in a single character field, the Field Name will be blanked out when the Field Menu is invoked for the field the first time. The user should input a new name for the field in this case. With this enhancement, a maximum of 80 single character fields can be defined for a Form.

Compatibility issues

Forms created with previous versions of FORMSPEC can be used with this version without any modifications.

Forms created with this version of FORMSPEC can be used with existing applications without any modifications.

Forms created with this version of FORMSPEC can be used with previous versions of FORMSPEC. However, some extra characters may appear near the bottom of the Form Menu screen which should be ignored. These extraneous characters appear only once, the first time the Form Menu is invoked for the form, and they do not affect the functionality in any way.

New Intrinsics to Get and Set Save Field Values

Two new intrinsics have been added to VPLUS to enable applications to access Save Fields. They are VGETSAVEFIELD and VPUTSAVEFIELD. The values of Save Fields can be accessed/changed using the new intrinsics. It is important to note that the values of the Save Fields cannot be modified in the forms file by these intrinsics.

Intrinsic VGETSAVEFIELD

Copies contents of the specified save field from save field buffer in memory to an application.

Syntax

VGETSAVEFIELD {*comarea,sfname,sfbuf,buflen,actualen*}

Parameters

comarea	must be *comarea* name specified when the forms file was opened with VOPENFORMF. If not already set, the following *comarea* items must be set before calling VGETSAVEFIELD:

	cstatus	set to zero.
	language	set to the code identifying the programming language of the calling program.
	comarealen	set to total number of two-byte words in *comarea*. Must be at least 70 words in length.

VGETSAVEFIELD may set the following *comarea* item:

	cstatus	set to nonzero value if call is unsuccessful.

sfname	is a character string in the application which contains the name of the save field.
sfbuf	is a character string in the application to which the contents of the save field is copied.
buflen	is a two-byte integer variable which specifies the number of bytes in *sfbuf*.
actualen	is a two-byte integer variable which specifies the number of bytes actually moved to *sfbuf*.

Discussion

VGETSAVEFIELD transfers the contents of the save field specified by *sfname* to a variable in an application.

If the number of bytes specified by *buflen* is less than the length of the save field, the bytes to the right are truncated. Following a successful transfer, *actualen* is set to the actual number of bytes transferred to *sfbuf*.

VGETSAVEFIELD does not convert the data moved to the application variable.

Example

```
COBOL
CALL "VGETSAVEFIELD" USING COMAREA SFNAME SFBUF BUFLEN ACTLEN.

SPL
VGETSAVEFIELD(COMAREA,SFNAME,SFBUF,BUFLEN,ACTLEN);
```

These calls will transfer the contents of the save field specified by `SFNAME` to `SFBUF`. If the call is successful, `ACTLEN` will contain the exact number of bytes transferred.

Intrinsic VPUTSAVEFIELD

Copies data from an application to the specified save field in memory.

Syntax

VPUTSAVEFIELD {*comarea,sfname,sfbuf,buflen,actualen*}

Parameters

comarea Must be *comarea* name specified when the forms file was opened with `VOPENFORMF`. If not already set, the following *comarea* items must be set before calling VPUTSAVEFIELD:

cstatus	set to zero.
language	set to the code identifying the programming language of the calling program.
comarealen	set to total number of two-byte words in *comarea*. Must be at least 70 words in length.

VPUTSAVEFIELD may set the following *comarea* item:

cstatus	set to nonzero value if call is unsuccessful.

sfname is a character string in the application which contains the name of the save field.

sfbuf is a character string in the application whose contents are copied to the save field in memory.

buflen is a two-byte integer variable which specifies the number of bytes in *sfbuf*.

actualen is a two-byte integer variable which specifies the number of bytes actually moved to the save field in memory.

Discussion

VPUTSAVEFIELD transfers the contents of the application variable specified by *sfbuf* to the save field in memory specified by *sfname*.

If the length of the save field is less than the length specified by *buflen*, the bytes to the right are truncated. Following a successful transfer, *actualen* is set to the actual number of bytes transferred to the save field in memory.

Note that the contents of the save field as recorded in the forms file is not changed by a call to VPUTSAVEFIELD. Also VPUTSAVEFIELD does not convert the data moved to the save field in memory.

Example

```
COBOL
CALL "VPUTSAVEFIELD" USING COMAREA SFNAME SFBUF BUFLEN ACTLEN.
```

```
SPL
VPUTSAVEFIELD(COMAREA,SFNAME,SFBUF,BUFLEN,ACTLEN);
```

These calls will transfer the contents of the application buffer SFBUF to the save field in memory specified by SFNAME. If the call is successful ACTLEN will contain the exact number of bytes transferred.

Renumbering a Form with Interactive FORMSPEC

FORMSPEC provides an option to renumber forms interactively. A new command has been added to the Interactive FORMSPEC to renumber forms. This command is invoked by selecting option "N" on the Main Menu. The form to be renumbered should be specified in the field next to the keyword "N". (This field is the same as where you would specify a form when also selecting the "D" option.) This option is equivalent to the RENUMBER command in batch mode FORMSPEC.

CI Enhancements

by M Gopalakrishnan and Jeff Vance
Commercial Systems Division

Introduction

This article describes the CI enhancements introduced in MPE/iX
Release 5.5 Express 5 and 6.0. The most notable enhancement is that
POSIX filenames support additional special characters. The COPY
command has been enhanced and several new CI evaluator functions
have been added.

POSIX Filenames

Prior to this enhancement, characters in POSIX filenames were
restricted to upper and lower case letters, numbers, dash, underbar and
dot. Now, a POSIX filename can contain the following additional
characters: `~\$%^*+|{}:`

A POSIX filename may begin with any supported character except a
dash. There are no changes to MPE filename rules.

Example

```
BUILD ./ab$cd:ef
LISTFILE ./ab$cb:ef,6
/SYS/PUB/ab$cd:ef

COPY ./ab$cd:ef  ./~anew%
LISTFILE ./[a~]@
ab$cd:ef  ~anew%
```

Enhancements to the COPY Command

The COPY command has been enhanced to accept MPE-syntax and
POSIX-syntax directory names for the TO= parameter. Additionally, a
space is an accepted delimiter between the FROM= and TO=
arguments.

For example (assume the CWD is /SYS/PUB):

```
NEWDIR mydir
NEWDIR ./mydir
BUILD ./foo

COPY ci,  boo        creates BOO.PUB.SYS, as always
COPY ci   mydir      creates /SYS/PUB/MYDIR/CI
COPY ci   ./mydir    creates /SYS/PUB/mydir/CI
COPY ci,  /SYS       creates /SYS/CI
COPY ./foo mydir     creates /SYS/PUB/MYDIR/foo
COPY ./foo ./mydir/  creates /SYS/PUB/mydir/foo
```

File equations are supported for both the FROM= and TO= parameters. However, the TO= name cannot be a back reference to a file equation who's actual filename is a directory. As before, symbolic links are accepted as the source and/or target filenames.

Enhancements to the SHOWVAR Command

The SHOWVAR command now allows users with SM capability to see user-defined CI variables for a job or session other than their own.

Syntax

```
SHOWVAR [var1 [,var2 [varN]]] [;JOB=[#]J|Snnnn]
        [;USER | ;HP | ;ANY]
```

JOB=   causes the user-defined variables associated with another job or session to be displayed. The HP option is not supported with JOB=, and no HP predefined variables are visible for another job/session.

USER   specifies that only user-defined variables are shown. this is the default when no variable name is supplied. USER is implied when JOB= is used.

HP   specifies that only HP predefined variables are shown.

ANY   specifies that both user and predefined variables are shown. This is the default when one or more variable names are supplied.

Operation

This command displays the variables specified and their values. It displays information in the format 'VARIABLE NAME = value string'. By default, the user's own variables are shown. System Manager (SM) capability is required to see the variables for another job or session.

If the variable names are omitted '@' and USER are assumed. The HP predefined variables are listed in alphabetical order; whereas, the user-defined variables are displayed in temporal order, that is the order in which they were created. The specification of JOB= forces the USER option too.

This command may be issued from a session, job, program or in BREAK. Pressing **Break** aborts the execution of this command.

Example

To see all user-defined variables for job #J123, enter

```
SHOWVAR @;job=J123; USER
```

New Evaluator Functions

ANYPARM

`ANYPARM()` returns any combination of characters as a string. String function.

Syntax

`ANYPARM(`*anything*`)`

This new function accepts all characters, including, commas, quotes, right parentheses, etc. and treats them as a single string value. This is handy when used in conjunction with the `ANYPARM` parameter type. The one exception is that the "!" character, which introduces a CI variable.

The `ANYPARM()` function has two important restrictions:

- It must be the last (right-most) or only function on a command line

- It may not be nested inside other function arguments. For example, `lft(ANYPARM(…),1)` is not supported, nor is `word(ANYPARM(…))`.

These constraints are necessary since `ANYPARM()` accepts all characters as its argument, including right parentheses.

Examples

Assume the script is named "doit":

```
ANYPARM p1
option list
setvar _infostr ANYPARM (!p1)
run prog; info="![repl(_infostr,'"','"""')]"
…
:doit abd"def'fhi)klm,p,q)r
setvar _infostr anyparm(abc"def'fhi)klm,p,q)r)
run prog; info="abc""def'fhi)klm,p,q)r"
```

NOTE
In the last example, if one of the characters is a "!" then the CI will attempt to reference a variable that follows the "!". For example, if the "f" is replaced by a "!" then the CI will try to reference a variable named `HI`. If this variable is not found a CI warning is reported.

BASENAME

`BASENAME()`—returns the filename component of a POSIX or MPE filename. String function.

Syntax

`BASENAME(`*str*`[,`*suffix*`])`

This new function returns the base component (file part only) for the passed filename. *str* can be any filename in MPE or POSIX syntax. *suffix* is an optional string, that if supplied and the filename base ends in *suffix*

then the *suffix* is removed from the BASENAME. *suffix* applies to MPE and POSIX names. The *suffix* portion of the BASENAME will not be deleted if this results in deleting the entire basename.

Examples

```
CALC basename('a.b.c')
A

CALC basename('/a/b/c')
c

CALC basename('./a/b')
b

CALC basename("./a.sl",".sl")
a

CALC basename('/')
/

CALC basename("*feq")
*FEQ

CALC basename('$null')
$NULL

CALC basename('abc.g','c')
AB

CALC basename('/usr/lib/liby.a','.a')
liby

CALC basename('/usr/lib/liby.a','liby.a')
liby.a
```

DECIMAL

DECIMAL( )returns a string value of an integer. String function.

Syntax

```
DECIMAL(int)
```

This new function behaves analogously to the existing hex() and octal() functions. It accepts a single integer argument and functionally returns the decimal string equivalent.

Examples

```
CALC decimal(255)
255                     < - - as a string

CALC len(decimal($ff))
3, $3, %3

setvar i 0
while setvar(i,i+1) < 10 and      finfo("FILE"+DECIMAL(i), 'exists') d
o
...
```

The last example is significant since this is the only case where explicit variable substitution ('!*varname*') does not work as an alternative to decimal. If the finfo() line was written as:

```
finfo("FILE"+"!i", 'exists') do
```

then the first iteration would call FINFO on a file named 'FILE0' rather than a file named 'FILE1'. This is because explicit variable dereferencing is performed early by the CI, well before the `setvar()` function is called.

DIRNAME

`DIRNAME()`—returns the directory components of a POSIX or MPE filename. String function.

Syntax

`DIRNAME(`*str*`)`

This new function returns the directory components for the passed filename. *str* can be any filename in MPE or POSIX syntax. MPE filenames are converted to POSIX names before extracting the directory portion of the name.

Examples

```
CALC dirname('a.b.c')
/C/B

CALC dirname('/a/b/c')
/a/b

CALC dirname('./a/b')
./a

CALC dirname("./a.sl")
.

CALC dirname('/')
/

CALC dirname("*feq")
[empty string]

CALC dirname('$null')
[empty string]

CALC dirname('abc.g')
/SYS/G

CALC dirname(fqualify('./a'))
/ACCOUNT/GROUP     # when the CWD is your logon group   or
/CWD               # when the CWD is a directory
```

FQUALIFY

`FQUALIFY()` returns a fully qualified filename. String function.

Syntax

`FQUALIFY(`*str*`)`

This new function returns the qualified filename for the passed filename in *str*. *str* can be any filename in MPE or POSIX syntax. *str* is first attempted to be translated into an MPE name. If that fails it is converted into a POSIX name. POSIX names are prepended with the CWD for relative pathnames.

Examples

```
CALC fqualify('a')
A.GROUP.ACCOUNT     # when the CWD is your logon group   or
/CWD/A              # when the CWD is a directory

CALC fqualify('a.b')
A.B.ACCOUNT

CALC fqualify('a.b.c')
A.B.C

CALC fqualify('./a')
/ACCOUNT/GROUP/a

CALC fqualify('./A')
A.GROUP.ACCOUNT     # when the CWD is your logon group   or
/CWD/A              # when the CWD is a directory

CALC fqualify('/a/b/c')
/a/b/c

CALC fqualify('*a')
*A

CALC fqualify('$null')
$NULL

CALC dirname(fqualify('./a'))
/ACCOUNT/GROUP      # when the CWD is your logon group   or
/CWD                # when the CWD is a directory
```

FSYNTAX

FSYNTAX() returns the syntax of the passed filename argument. String function.

Syntax

FSYNTAX(*filename_str*)

This new function accepts an MPE or POSIX style filename and evaluates the filename's syntax, if wildcard characters are present, lockwords, back reference to a file equation, remote environment IDs and system defined files. If an error occurs the error number, following the word "ERROR" is returned. Possible return values are:

- "MPE" "MPE;WILD" "MPE;LOCK" "MPE;FEQ" "MPE;$FILE" "MPE;REMOTE" and combinations where appropriate, for example, "MPE;LOCK,REMOTE"

- "POSIX" "POSIX;WILD"

- "ERROR=nnn" for syntax error nnn

Examples

```
fsyntax('a.b.c')
MPE
fsyntax('/a/b/c')
POSIX
fsyntax('./ab@/c')
POSIX;WILD
fsyntax('$null')
MPE;$FILE
fsyntax('a.b.c.d')
ERROR=426

if word(setvar(_error, FSYNTAX(myfile))) = "ERROR" then
    echo Syntax error: ![word(_error,,-1)]
```

```
endif

if word(FSYNTAX(a_file)) = "MPE" then
   # MPE filename
elseif word(FSYNTAX(a_file)) = "POSIX" then
   # POSIX filename
else
   # error

if pos("WILD", FSYNTAX(another_file)) > 0 then
   # wildcarded MPE or POSIX filename
```

XWORD

XWORD()—returns a string less 'word'. String function.

Syntax

XWORD(*str*[,delims][,*nth*][,*end_var*][,*start*])

This new function locates the *nth* word from *str* beginning at *start*, delimited by one of the characters in *delims*, and places the index of the delimiter that terminated the word in a CI variable named by the *end_var* argument. It returns *str* minus the located word and minus the word delimiter. It is syntactically equivalent to word(), but opposite in semantics.

The arguments for xword() are identical to the word() parameters. Please refer to word() for details. HELP word and HELP xword are available.

xword() is useful for removing a token from a command line string since the token and its delimiter are not returned.

The same examples used for the word() function in an earlier article are repeated below for xword():

```
xword('file a=bb,old;rec=40,,f,ascii')             returns:
                              'a=bb,old;rec=40,,f,ascii'

xword('file a=bb,old;rec=40,,f,ascii',,2)          returns:
                              'file bb,old;rec=40,,f,ascii'

xword('file a=bb,old;rec=40,,f,ascii',";",,j,8) returns:
                              'file a=old;rec=40,,f,ascii' and J=10

xword('file a=bb,old;rec=40,,f,ascii',,-4,j)       returns:
                              'file a=bb,old;rec=,f,ascii' and J=18
```

It may be illustrative to execute the above examples substituting word in place of xword(), or refer the word() examples referenced above.

Summary of CI Enhancements in C.55.03

- The LISTF and LISTFILE commands allow users to discover who is accessing files on their system, including remote accessors. LISTFILE output can now be filtered by selecting files based on access type or file code.

- Two new CI variables were added: HPLASTSPID and HPSPOOLID. Both contain the spoolfile ID in the form: O*nnnnnnn*.

- The `FLABELINFO` intrinsic and `FINFO()` function are able to return the number of sectors and extents a file occupies, file creation time, and the number of accessors to a file.

- The `PAUSE` command can now wait for one or more jobs to complete, to change state from waiting to executing, or to begin execution.

- The `INPUT` command allows the number of characters to read from `$STDIN` to be specified.

- The `PRINT` command is now able to display all data in a file, even if this data appears to be line numbers. The new `NONUM` option enables this feature.

LISTF/LISTFILE Enhancements

Syntax

```
LISTF [fileset][,listlevel][;listfile]

LISTFILE [fileset[,fileset[,...]]]
[[;FORMAT=]format_opt]
[[;SELEQ=]select_eq indir]
[[;NAME=] pattern]
[;PASS]
[;{PERM}{TEMP}{PERMTEMP}]
[;USENAME][;TREE][;NOTREE]
```

The `LISTF` and `LISTFILE` commands allow users to discover who is accessing files on their system, including remote accessors. The existing syntaxes remain the same; however, new format numbers, format mnemonics, and selection equations can be specified as a result of this enhancement.

New Formats

Two new formats have been added: format 8 or "access," and format 9 or "locks." Both the `LISTFILE` and `LISTF` commands support these formats, although only `LISTFILE` is able to use the format mnemonic. Format 8 shows greater access-related details compared to the existing format 3, as well as displaying individual file accessors. Format 9 is a superset of format 8. All of the format 8 data is shown with specific details about each process accessing the file, including information about which locks a process owns or is waiting for. A detailed description of the items displayed for the two new formats and examples are shown below.

The selection equation parameter of the `LISTFILE` command is also enhanced. The new `SELEQ=` choices allow selection of only files that are currently opened, in use, locked, or opened exclusively. If a file is not opened as specified in the selection equation, it is skipped. Also, files can be selected based on their file code number or mnemonic. The file code mnemonic `PRIV` includes all privileged files, that is, those files with a negative file code. More details on the access `SELEQ` choices appear later in this article.

Format 8 . Both formats 8 and 9 show the fully qualified filename or absolute pathname followed by the accessor summary line (described below). After the accessor summary line, there is one line of output per file accessor, consisting of:

- job/session number

- user name

- number of processes accessing, locking, writing and reading the file

- IP address for remote sessions, or the $STDLIST spoolfile ID for job accessors, or the LDEV number for local session accessors

Following is a format 8 example:

```
:listfile hppxudc.pub.sys,8
********************
FILE: HPPXUDC.PUB.SYS
15 Accessors(O:15,P:15,L:0,W:0,R:15),Share
#S265   MIKEP.HPE            P:2,L:0,W:0,R:2      LDEV: 49
#S263   JEFFV,MGR.JVNM       P:3,L:0,W:0,R:3      LDEV: 47
#S261   KROGERS.MPENT        P:2,L:0,W:0,R:2      LDEV: 50
#S231   SUSANC.MPENT         P:2,L:0,W:0,R:2      LDEV: 46
#S219   FAIRCHLD.MPENT       P:2,L:0,W:0,R:2      LDEV: 39
#S214   CATHY,MGR.BOSS       P:2,L:0,W:0,R:2      REM : 15.14.16.19
8
#J434   FTPMON,FTP.SYS       P:2,L:0,W:0,R:2      SPID: #O21905
```

Analyzing the above format 8 example:

1. The filename shown is always fully qualified as it appears in format 6.

2. The line beginning with "15 Accessors…" is the accessor summary line, which shows the total number of accessors of the file across all jobs and sessions. This total includes all processes that have formally opened the file and all processes that have pseudo-opened the file. This is the same number seen in the FLAGS field when executing LISTF,3. The total number of accessors may not necessarily be derived by adding up the individual accessors displayed. This is because certain file accesses are done without using the file system and these accesses are transient and difficult to detect. An example of this kind of access is when STORE pseudo-opens a file to read its contents to backup.

   The "O:" field is the actual number of processes that formally opened the file (for example, called FOPEN, HPFOPEN, OPEN, or others). The O: number can be lower than the total number when the file has been pseudo-opened, as done by STORE. Another factor is that the system protects certain files at boot time by pseudo-opening them, as is done on NL.PUB.SYS. Also, a LISTF,8 done at the same time on the same file as a LISTF,2 (for instance) will show the LISTF,2 users as an accessor that has not formally opened the file. Typically, however, the O: count and total number of accessors match.

   The total number of processes (P:) accessing the file should generally match the O: count.

The number of processes that have the file locked (L:) is seen next. The number of lockers includes all file system locks (semaphores) held by all detectable processes that are accessing the file. A single process can obtain multiple locks, and each time the count is incremented. When a process releases a lock, the count is decremented. The locks that are considered are the FLOCK semaphore (used by the FLOCK intrinsic), the OPEN semaphore (used by the file system), and the GUFD semaphore (used to control I/O).

The number of writers (W:) and readers (R:) are displayed next. The writer and reader counts can be greater than the number of accessors since a single process can open the same file multiple times. Also, when a process opens a file for append, update, or save (read/write) access both the writer and reader counts are incremented.

After all of the counts are shown, the file-sharing mode is displayed. This is either "Share" for files opened shared, or "Exclusive" for files opened exclusively. Typically when a file is opened exclusively there is only a single accessor; however, the POSIX `fork()` and `dup()` functions are exceptions.

The last item displayed in the access summary is the backup state. If the file is being stored, then "Storing" is shown. If the file is being restored, then "Restoring" is seen. If the file is being concurrently backed up, then "ConcurrBackup" is displayed. Otherwise, nothing additional is shown.

3. The lines after the accessor summary line reflect each job and session accessing the file. There is one line displayed for each job and session accessing the file, even if there are multiple processes within the job accessing the same file.

The first field shown is the job/session number. If the accessor is a system process then "system" is seen instead of a job ID. Following the job ID is the user name associated with the job. This consists of the `USER.ACCOUNT` preceded with an optional job name. The logon group name is not seen.

The same counts displayed in the accessor summary line are shown for each individual job accessor. The definitions are identical. Unlike the summary line, the reader and writer fields are truncated if the process and locker counts are extremely high. In this event the right-most field is the first to be truncated and an "$" indicates that truncation occurred.

The last field displayed depends on whether the accessor is a job, a remote session or a local session. For jobs, the spoolfile ID is shown. For local sessions, the LDEV number is displayed. If the accessor is remote, the IP address is shown.

Format 9 . In addition to the format 8 data, format 9 shows more details such as:

- PIN number

- program name

- access method (for example, read, execute, or others)

- share mode (for example, `multi`, `gmulti`)

- current record number being accessed

- file number

- locks the accessing process owns and is waiting on

Following is a format 9 example:

```
:listfile hppxudc.pub.sys,9
********************
FILE: HPPXUDC.PUB.SYS
5 Accessors(O:5,P:5,L:5,W:0,R:5),Share
#S263   JEFFV,MGR.JVNM             P:3,L:3,W:0,R:3      LDEV: 47
 #P147   (LFCI.PUB.SYS)
   ACCESS: R-excl           REC#: 0                 FNUM: 13
    LOCKSOwner   Waiter
         FLOCK
         OPEN
 #P154   (CI.PUB.SYS)
   ACCESS: R-excl           REC#: 0                 FNUM: 13
    LOCKS: none
 #P86    (JSMAIN.PUB.SYS)
   ACCESS: R-excl           REC#: 336               FNUM: 16
    LOCKSOwner   Waiter
                  FLOCK

#J434   FTPMON,FTP.SYS            P:2,L:2,W:0,R:2      SPID: #O21905
 #P79    (CI.PUB.SYS)
   ACCESS: R-excl           REC#: 0                 FNUM: 14
    LOCKS: none
 #P47    (JSMAIN.PUB.SYS)
   ACCESS: R-excl           REC#: 336               FNUM: 15
    LOCKSOwner   Waiter
         OPEN        FLOCK
```

Analyzing the above format 9 example:

1. All of the format 8 data is seen.

2. The PIN (Process ID Number) begins the next line, followed by the process filename.

3. The next line shows the type of access which includes: "R" for read, "W" for write, "X" for execute, "A" for append, "L" for lock, "S" for save, "U" for update, "RD" for directory read access. The file-sharing mode is next. Values include: "sysexc" for system exclusive (rarely seen), "excl" for exclusive, "ear" for semi-exclusive (exclusive-allow-read), "shr" for shared, "multi" for shared job, or "gmulti" for system shared. Next the current record number is seen, and last is the file number.

4. If the process does not have any locks then "NONE" is displayed. If the PIN owns or is waiting on one or more of the relevant semaphores, then the semaphore name is shown under either the "Owner" or "Waiter" column. There are three semaphores (locks) considered.

- The FLOCK semaphore is obtained when a processes calls the FLOCK intrinsic. This lock is exclusive; meaning that only one process at a time can own the FLOCK semaphore for a file. Occasionally the FLOCK semaphore is used by the file system to synchronize directory access.

- The OPEN semaphore is locked by the file system when a file is being formally opened. In this usage the OPEN semaphore is locked exclusively. It is also used by the operating system to protect certain file system data structures when they are being read. This usage allows the OPEN semaphore to be locked in a shared fashion. The OPEN semaphore can be locked shared by multiple processes, but once it is locked exclusively all other PINs trying to lock it will wait.

- The GUFD semaphore is locked by the file system when it does I/O on the target file. It is locked exclusively.

Phantom Accessors . It was mentioned earlier about processes that pseudo-open (call sm_open) a file. STORE and PROGEN are examples of processes that bypass the file system by not calling FOPEN or HPFOPEN to open a file. This type of access is usually only evident by noticing that the total accessors count is greater than the open count (O:), and format 9 output typically will not show these phantom accessors.

There are also cases where a process does not open the file in any way at all, yet this process can lock one or more of the semaphores, and thus needs to be accounted for. LISTFILE and MPEX (when calling AIFSYSWIDEGET item 2065 to obtain a file's accessors) are examples of these kinds of phantom accessors. In this case the total accessor count is lower than what may be revealed by counting individual accessors shown in format 9.

If any of these phantom accessors lock one of the three important semaphores, then LISTFILE can detect the process and report reduced information about the access. If the phantom accessor process does not lock either the FLOCK, OPEN, or GUFD semaphores, then its access is undetected. Following is a format 9 example of a file that another user is also executing a LISTF,3 against. Notice that the total accessor count is one, yet we see two accessors. PIN 58 is the process that is doing the LISTF,3 and thus is seen as a phantom accessor.

```
FILE: GOPI.PUB.SYS
1 Accessor(O:1,P:2,L:2,W:0,R:1),Share
#S3     JVREM,MANAGER.SYS           P:1,L:1,W:0,R:1    REM : 15.28.88.18
 #P62    (LOCKP.PUB.SYS)
   ACCESS: R-excl             REC#: 0               FNUM: 11
    LOCKSOwner    Waiter
```

```
        FLOCK

#S2     JVREM,MANAGER.SYS          P:1,L:1,W:0,R:0    REM : 15.28.88.18
 #P58    (CI.PUB.SYS)
   ACCESS: n/a                 REC#: n/a               FNUM: n/a
     LOCKSOwner    Waiter
          OPEN
```

Restrictions . Formats 8 and 9 are available to all users, however only more privileged users get to see individual accessor information. The user having SM or OP capabilities can see all accessors of a file—assuming the user has TD access to the file itself. If the user does not have SM or OP capabilities, but does possess AM capability, and the file's GID matches the user's GID, then all accessors are displayed. If the user lacks SM, OP and AM capabilities, but is the owner of the file being listed, then the individual accessors are displayed. Otherwise only the accessor summary line is visible.

Even if the user is able to see individual accessors, certain format 9 fields have additional restrictions for security concerns: the program name and IP address are protected.

To see the program name the user must meet the same rules that apply to the SHOWPROC command. Specifically, one or more of the following applies:

- process is within the user's logon job/session.

- process' user and account names match the user's user and account names and the system's JOBSECURITY is set to LOW.

- user has OP or SM capability.

To see the IP address the user must meet the SHOWCONN utility rules, namely: the user must have SM, OP, PM, NA, or NM capabilities.

Selection Equations for LISTFILE

Selection equations for the LISTFILE command have been expanded. It is now possible to select files by whether and how a file is accessed, and by file code. The ACCESS= criteria supports 4 types of access, as shown below:

```
SELEQ=[ ACCESS = INUSE | OPEN | LOCK[ed] | EXCL[usive] ]
```

INUSE   is true if any processes are accessing the file. This includes normal FOPEN-like accessors and sm_open accessors. Phantom accessors (like LISTF,2) that don't open the file at all are not noticed and thus INUSE evaluates to false. INUSE examines the total number of accessors field described above.

OPEN   is true if there are any processes that have formally
opened the file by calling FOPEN, HPFOPEN, RESET,
OPEN, or others. OPEN uses the same field that the "O:"
count is derived from.

LOCK   is true if any process has locked (even if they are
waiting) the FLOCK, OPEN or GUFD semaphores, or has
called the HPFLOCK pseudo intrinsic (for POSIX record
level locking).

EXCL   is true if the file was opened for exclusive access. EXCL
is true when LISTF,3 shows EXCL or LISTFILE,8
shows "Exclusive."

The CODE= criteria supports file code numbers and mnemonics as
shown below:

```
SELEQ=[ CODE = number | mnemonic | PRIV ]
```

The number can be positive or negative. PRIV will match all files with
negative file codes.

NOTE   A new FINFO item 61 ("accessors") returns the number of accessors for
a file. This represents the total number of accessors, including all
processes that open the file in any manner. This is the first value seen
in the accessor summary line, which is described under the Format 8
description. It is also the same value shown in the FLAGS field of a
LISTF,3.

If a listed file is being accessed LISTF and LISTFILE now get the EOF
for the file from runtime data structures, rather than from a possibly
out-of-date file label as done in the past.

New Variables

Two new variables were added to MPE/iX CI:

- HPLASTSPID

- HPSPOOLID

HPLASTSPID is a read-only string variable. It contains the $STDLIST
spoolfile ID for the job defined by the HPLASTJOB variable, typically the
job most recently streamed in the session or job. Since HPLASTJOB
allows write access, it can be set to any job number, and then
HPLASTSPID will contain that job's $STDLIST spoolfile ID.

HPSPOOLID is a read-only string variable. It contains the $STDLIST
spool ID of the currently streamed job. This variable is useful only in
jobs—in a session it is an empty string.

FLABELINFO and FINFO Enhancements

The FLABELINFO intrinsic returns information from the file label of a disk file and the FINFO function is a CI interface to FLABELINFO. These APIs have been enhanced to return four new items:

- number of sectors occupied by a file
- number of extents occupied by a file
- file creation time
- number of accessors to a file

FLABELINFO

Table 10-1 **FLABEL Item Numbers**

Item Number	Type	Item Description
58	I16	Number of sectors occupied by the file.
59	I16	Number of extents occupied by the file.
60	I32	File creation time (CLOCK format).
61	I32	Number of accessor of the file.

FINFO accepts the following numbers and aliases:

Table 10-2 **FINFO Numbers**

FINFO Number	Mnemonic	Return Type
58	SECTORS NUM SECTORS	Integer
59	EXTENTS NUM EXTENTS	Integer
60	CREATETIME FMTCREATETIME CREATION TIME	String
-60	INTCREATETIME CREATION TIME INTEGER	Integer
61	ACCESSORS NUM ACCESSORS	Integer

Footnote: this is the same value as the total number of accessors in the LISTF format 8 summary line.

Examples

```
finfo('CI.PUB.SYS',"sectors")    = 192
finfo('CI.PUB.SYS',"extents")    = 2
finfo('CI.PUB.SYS',"createtime") = 11:08 AM
finfo('CI.PUB.SYS', -60)         = 110808
finfo('CI.PUB.SYS', "accessors") = 124
```

PAUSE Enhancement

Syntax

```
PAUSE [num_seconds]
[[;JOB=]jobid]
[[;INTERVAL=]interval_secs]
[;EXIST | WAIT | NOTEXIST]
```

The PAUSE command allows the current task to be suspended or "sleep" for a specified number of seconds. PAUSE now supports sleeping until one or more jobs reach a certain state. For example, a script can pause while selected jobs or sessions are executing (EXIST). Or, a job can sleep while another job is suspended or waiting (WAIT), and as soon as the job starts executing or terminates the pause completes. A session can pause while no jobs exist (NOTEXIST) and wake up as soon as the first job is launched.

In its simplest form, the PAUSE command sleeps for *num_seconds*, or less if **Break** is pressed. In this simple case no *jobid* is specified and all other command arguments are ignored. If the *jobid* parameter is specified, then *interval_secs* and the remaining command parameters have meaning. When *jobid* is supplied, PAUSE typically sleeps until the jobs or sessions matching *jobid* have terminated.

Options

EXIST   (default) means to pause while all jobs and sessions matching *jobid* exist. These jobs can be scheduled, waiting, executing, and other states; but, as long as the system's global job table (JMAT) contains an entry for any of the selected jobs and sessions, the PAUSE command will continue to sleep.

WAIT   means to pause while the selected job or jobs are waiting. As soon as all the matching jobs are no longer waiting (meaning all the job states are no longer "introduced," "waiting," or "scheduled") the pause ends. The life cycle of a job is typically:

```
[sched or waiting->] intro-> initializing->
exec-> [susp-> exec->] terminate
```

Waiting jobs are considered all job states left of and excluding "initializing." Non-waiting jobs are all jobs right of and including "initializing."

NOTEXIST   means to pause while the matching job or jobs do not exist. As soon as any jobs matching *jobid* exist (in any state) the pause completes. PAUSE might miss finding short-lived jobs. This is particularly true for a match on a single job/session number. A more practical use might be:

```
PAUSE job=@J;notexist
```

which means to sleep while no jobs exist. As soon as the first job is streamed the above pause stops.

Collectively EXIST, WAIT, and NOTEXIST are referred to as the "while_state," since PAUSE sleeps "while" the specified state condition is true.

Parameters

num_ seconds   if *num_seconds* is specified without *jobid*, PAUSE sleeps for that many seconds, or until the process issuing the PAUSE is interrupted by the break signal.

If *jobid* is also supplied then *num_seconds* has a different meaning. In this case it indicates the maximum duration for the PAUSE command, such that PAUSE should continue while the selected jobs are in their "while_state" or when *num_seconds* has expired, WHICHEVER IS SHORTEST. Thus, *num_seconds* represents the maximum length of the pause. If PAUSE completes but the one or more jobs are still in their "while state" a CIWARN is reported. Note: to pause while a job is in its "while_state" or until *num_seconds* has expired, whichever is LONGEST, execute the following two commands:

```
PAUSE x
PAUSE job=y ;z
```

If after X seconds job Y is still in state Z, then the second PAUSE continues while state Z applies. On the other hand, if after X seconds job Y is not in state Z then the pause is complete.

jobid   can be one of: [#]Jnnn, [#]Snnn, [*jobname,*]*user.acct*, @, @J, @S. Note if *jobname* is included then the *jobid* must be quoted since the comma is a command token delimiter.

If the JOB= parameter is specified then PAUSE sleeps while *jobid* is in its "while_state." *jobid* can be an executing, waiting, scheduled job, or a session. *jobid* can also name many jobs or sessions. Wildcarding is supported, and a non-wildcarded [*jname,*]*user.acct* can match several jobs or sessions. The job name can be "," or "@," to match all jobs or sessions without a job name.

When more than one job or session matches *jobid*, PAUSE sleeps while all matching jobs are in their "while_state." If the job executing PAUSE matches *jobid* it will not be selected.

interval_ secs   if specified, PAUSE sleeps for this many seconds between attempts to see if *jobid* is still in its "while_state." Otherwise, PAUSE sleeps a variable amount of seconds depending on the job state and the number of previous times a particular job has been polled. This computed method favors executing jobs that terminate quickly.

Examples

The following pauses while job #J20 is in any of the JOB states such as INTRO, SCHED, and EXEC until the job terminates:

```
:PAUSE JOB=#J20;EXIST
```

The following pauses while job #J24 is in INTRO, WAIT, or SCHED state. PAUSE ends when #J24 is no longer in any of these states.

```
:PAUSE JOB=#J24;WAIT
```

The following example pauses while job #J24 exists in the system job table, (JMAT). That is, it is visible in SHOWJOB output.

```
:PAUSE job=#j24
```

The next example sleeps as long as MANAGER.SYS has any jobs or sessions running or waiting.

```
:PAUSE job=manager.sys exists
```

The next example pauses until the job streamed starts executing

```
:STREAM myjob
:PAUSE job=!hplastjob; wait
```

Or sleeps until the job you just streamed completes.

```
:PAUSE , !HPLASTJOB
```

The following example sleeps until all jobs have logged off or 5 minutes, whichever occurs first.

```
:PAUSE 300, @J
:IF hpcierr = -9032 then
    # pause terminiated but one or more jobs are still running
```

The next example pauses while all jobs (by naming convention only) in the PROD account are running.

```
:PAUSE job="J@,@.PROD"          #note the quotes are required
```

The next example sleeps while the backup JOB ("JBACKUP,OP.SYS") has not been streamed. PAUSE reports CIWARN 9032 if the job is not streamed within 30 minutes.

```
:PAUSE 1800, job="jbackup,op.sys"; notexist
```

The final example polls the `JMAT` every 3 minutes looking for any job or session matching a user name that includes the letters "`MGR`", and waits for all such job/sessions to terminate before the pause ends.

```
:PAUSE , @mgr@.@ , 180
```

INPUT Enhancement

The `INPUT` command was enhanced to accept the number of characters to read from `$STDIN`. `INPUT` optionally writes *prompt* to `$STDLIST` then reads *numchars* from `$STDIN` into *varname*, optionally waiting *seconds* for the user to enter the data.

Syntax

```
INPUT [NAME=]varname
[[;PROMPT=]prompt]
[[;WAIT=]seconds ]
[[;READCNT=]numchars]
```

The `READCNT` parameter is new. If *numchars* characters are specified then up to *numchars* bytes will be read from `$STDIN`. Of course the user can press the **Return** key at any time to complete the read. *numchars* is most useful for reads of only a few characters and exempts the user from being required to press the **Return** key. The default *numchars* is the maximum size of the CI's command buffer, currently 511 bytes. If *numchars* is specified and less characters are supplied as input, the user must still press the **Return** key to send the data.

Example

```
INPUT myvar,'Do you want to continue (Y/N)?', 10, 1
```

prompts to `$STDLIST`, and does a 10 second timed read on `$STDIN` of one character. When the user types the first character the read will complete and the read byte will be stored in the variable named MYVAR.

PRINT Enhancement

The `PRINT` command prints truncated records when an UNNUMbered file contains the trailing eight characters as digits. The `PRINT` command is not displaying the trailing eight characters as it assumes those digits as line numbers.

NOTE

This is the existing rule as of MPE/iX 5.5 for determining whether a file is a numbered file:

If the trailing 8 characters of at least 10 records of a file or all records (in the case of <10 records) are digits, then that file is considered as a numbered file, provided the record line numbers (trailing 8 character digits) are in ASCENDING order.

With this new option `NONUM`, the contents of the file would be displayed as it is, without assuming the trailing 8 characters as line numbers.

With this enhancement, the `PRINT` command syntax is

```
PRINT  [ [FILE=] filename]
[[;OUT=]outfile]
[[;START=] m]
[[;END=] n]
[[;PAGE=] p]
[;UNN | NUM ]
[;NONUM]
```

- With this new option, to print the line numbers as in the case of UNNUMbered files (that is, line numbers starting from 1 for the first record and so on), use this command:

  ```
  :PRINT infile;NUM;NONUM
  ```

- To consider the file as UNNUMbered file even when the file is a NUMbered file and then print the contents as it is in the file, use this command:

  ```
  :PRINT infile;UNN;NONUM
  :PRINT infile;NONUM
  ```

Examples

In this example, UFILEYES is an unnumbered file with trailing 8 characters as digits.

```
:PRINT UFILEYES
aaaaaaaaaaaa
bbbbbbbbbbbb
cccccccccccc
dddddddddddd
eeeeeeeeeeee
ffffffffffff
gggggggggggg
hhhhhhhhhhhh
iiiiiiiiiiii
jjjjjjjjjjjj
kkkkkkkkkkkk
llllllllllll
```

NOTE   The above file was considered by `PRINT` to be a numbered file and thus the trailing 8 bytes are truncated.

```
:PRINT UFILEYES;NONUM
aaaaaaaaaaaa00010001
bbbbbbbbbbbb00010002
cccccccccccc00010003
dddddddddddd00010004
eeeeeeeeeeee00020001
ffffffffffff00020002
gggggggggggg00020003
hhhhhhhhhhhh00020004
iiiiiiiiiiii00030001
jjjjjjjjjjjj00030002
kkkkkkkkkkkk00030003
llllllllllll00030004
```

In the above example, `NONUM` option displays all the data in the file.

The REMSH Client Functionality

by Cas Caswell
Commercial Systems Division

An Overview of the REMSH Service

The remote shell, or REMSH, service is used to connect to a specified host and execute a command on that remote host. This article describes:

- Verifying installation

- Configuring the REMSH client

- Using REMSH

- Troubleshooting REMSH

- Implementation differences between REMSH on MPE/iX and REMSH for HP-UX

- Available documentation

REMSH is the same service as rsh on BSD UNIX systems. The name was changed due to a conflict with the existing command rsh (restricted shell) on System V UNIX systems.

REMSH allows a user to connect to the remote system and execute a command on that remote system. Output from the remote command is sent to the REMSH standard output so the user can see the results of the command.

Verifying Installation

The REMSH client is part of the Internet Services product with release C.55.03. To verify that REMSH is available on your system, use NMMAINT,73 to list versions of the Internet Services.

Configuring the REMSH Client

There is only one file on the MPE/iX system that you will need to change in order to allow use of the REMSH client. That is the file services.net.sys. However, there are some files that will need to be configured on the remote UNIX systems.

Editing the Services File

The services file associates official service names and aliases with the port number and protocol the services use. To enable REMSH use, you must edit the services file.

To edit the services file:

1. You may edit the `SERVICES.NET.SYS` file from MPE/iX. Open the services file with a text editor.

2. Verify that the following line exists in the file.

   ```
   shell   514/tcp  cmd   # remote command, no passwd used
   ```

   a. Add the line if it does not exist.

   b. If the line already exists in the file and is preceded by a pound symbol (#), delete the # and any spaces before the service name to enable the service.

3. Save the file and exit the editor program.

UNIX Configuration

The REMSH service does not prompt for user ID and passwords. That information is handled via the command line parameters and configuration on the UNIX host. See the using REMSH section for details on how the user id is determined and passed to the UNIX host.

Password information is bypassed by use of a .rhosts file in the remote user's home directory or by use of the file /etc/hosts.equiv. See the man pages of the UNIX system for details on how to set up a /etc/hosts.equiv file. A user's .rhosts file entry will consist of the MPE/iX system name and user id.

For example, to access the HP-UX system "taltos" as user "cawti," from the MPE/iX system "jhereg" while user `MANAGER.SYS`, set up a host equivalency via the /etc/hosts.equiv file, or you will create a .rhosts file in the home directory of user "cawti" on the "taltos" machine. The .rhosts file would look like this:

```
jhereg  MANAGER.SYS
```

This will cause the REMSH daemon on the UNIX host to allow a connection from `MANAGER.SYS` on "jhereg" to the "cawti" user on the host "taltos." The .rhosts file for user "cawti" would contain an entry for every host and userid that you desired to access the "taltos" host as if they were the user "cawti."

NOTE   The MPE/iX equivalent of the UNIX user id is the User.Account. An artifact of the MPE/iX implementation is that the MPE/iX information is usually reported in upper case. So be sure your .rhosts or /etc/hosts.equiv entries use the MPE/iX user ID information in upper case.

Using REMSH

The REMSH service is accessed by running the `REMSH.NET.SYS` program. You may do so under the MPE/iX CI or under the Posix shell. While the format of the commands will differ depending on how you run the program, the parameter list remains the same. We'll first look at the parameters and then look at running the program under the Posix Shell or from MPE/iX CI.

For the purposes of explaining the parameters, look at a sample invocation from the Posix shell. Detailed examples of both the Posix shell and MPE/iX invocations will follow later.

From the Posix shell you would invoke the REMSH by typing:

```
/SYS/NET/REMSH remotehost -l remoteuser  remotecommand
```

In all cases you must provide a *remotehost* and a *remotecommand*. The REMSH program will fail and generate an error message otherwise. Unless the remote system has MPE/iX type userids, you will also need to provide a -l *remoteuser* parameter as well. Otherwise the remote system will not allow the connection.

remotehost   is the name of the remote host you are attempting to connect to. The host name can be either the official name or an alias as understood by gethostbyname().

remoteuser   is the userid on the remote system you wish to be.

NOTE   The traditional UNIX implementation of REMSH makes the -l *remoteuser* parameter optional. If you do not provide a -l *remoteuser* parameter, REMSH takes your current userID and assumes that you wish to connect to the same userID on the remote system. Since the MPE version of the userID is `USER.ACCOUNT`, and the UNIX equivalent is user, it is unlikely that you will find a user on the remote system to match your id. We recommend that you always provide the -l *remoteuser* argument to REMSH.

remote- command is the command that you wish to execute on the remote machine. This command may be a CI command, a program (that meets certain criteria) or a shell script. If *remotecommand* is not specified, REMSH will terminate and provide a usage message.

NOTE   REMSH cannot be used to run commands that require a terminal interface (such as VI) or commands that read their standard error (such as `more`).

MPE/iX Examples

To run REMSH from MPE/iX prompt, type:

```
run remsh.net.sys;info="remotehost -l remoteuser [-n]
remotecommand"

jhereg(PUB): run remsh.net.sys;info="taltos -l cawti -n pwd"
/u2/home/cawti
jhereg(PUB):
```

Posix Examples

From the Posix Shell prompt, type:

```
/SYS/NET/REMSH remotehost -l remoteuser [-n] remotecommand
shell/iX> /SYS/NET/REMSH taltos -l cawti -n pwd
/u2/home/cawti
shell/iX>
```

Since you are running under the Posix shell, there are a number of shell features that you can take advantage of.

Shell metacharacters that are not quoted are interpreted on the local host; quoted metacharacters are interpreted on the remote host. The following command line appends the *remotefile* to the local file *localfile*:

```
SYS/NET/REMSH taltos -l cawti cat remotefile >> localfile
```

The following command line appends *remotefile* to the *otherremotefile*:

```
/SYS/NET/REMSH taltos -l cawti cat remotefile ">>" otherremotefile
```

The following command line runs REMSH in the background on the local system, and the output of the remote command comes to your terminal asynchronously:

```
/SYS/NET/REMSH otherhost -l remoteuser  -n remotecommand
```

The following command line causes REMSH to return immediately without waiting for the remote command to complete:

```
/SYS/NET/REMSH otherhost -l remoteuser  -n
"remotecommand 1>&- 2>&- &"
```

In REMSH, if the first parameter in its argument vector is not REMSH, it will use the value as a host name. You may symbolically link the host name to the REMSH program. A typical BSD UNIX implementation will have these links under the /usr/hosts directory.

If you have made a symbolic link to the REMSH program that is the host name (that is in our examples, `ln -s /SYS/NET/REMSH taltos`), you could generate the same result as the first example with the following:

```
shell/iX>taltos -l cawti -n pwd
/u2/home/cawti
shell/iX>
```

Troubleshooting REMSH

These are the REMSH messages with an explanation of the meaning and suggested action, if any.

Message

```
remsh: MPE/iX/X version won't support rlogin or rexec
functionality
usage: remsh host -l login -n command
```

Meaning

Be sure to provide a command to execute.

Message

```
remshd: Login incorrect
```

Meaning

There is probably an invalid entry in remote .rhosts file. Be sure *host* name and *userid* are correct. *Userid* must be in upper case. Be sure you provided an -l *userid* parameter, or that the remote system has a *userid* that matches your MPE/iX logon.

Message

```
Program requires more capabilities than allowed for the group, the user
of a temporary file, or the hierarchical directory user. (LDRERR 505)
Native mode loader message 505
Unable to load program to be run. (CIERR 625)

**** EXEC FUNCTION FAILED; subsys =517; info = 48
ABORT: REMSH.NET.SYS
NM SYS    a.00aa0270 dbg_abort_trace+$24
NM UNKN   150.00366f6c
NM UNKN   2dd.0004bbd8
[1] + Done (134) REMSH hpcsyn24 -l casc -n pwd
  262204        Abort    REMSH
```

Meaning

The first message is from running REMSH from MPE/iX name space, and the second is from running under the Posix Shell. The cause is typically lack of PM capability on the group where REMSH resides. Since REMSH is in NET.SYS, it is unlikely you will see this problem unless someone changes the capability of the NET.SYS group.

Message

```
shell/tcp: Unknown service
```

Meaning

The shell service specification is not present in the services file. Edit /etc/services or SERVICES.NET.SYS to fix.

Message

```
Can't establish stderr
```

Meaning

REMSH cannot establish secondary socket connection for stderr.

Message

```
Couldn't reopen stderr
```

Meaning

The remote command tried to reopen stderror. This is not allowed under REMSH.

Message

```
<system call>: ...
```

Meaning

Error in executing system call. Appended to this error is a message specifying the cause of the failure.

Implementation Differences

The full remote shell service typically consists of two parts: the REMSH client which allows a user on this machine to access remote hosts, and the remshd server which allows REMSH clients on other hosts to access the local host. Only the REMSH client functionality has been implemented on the MPE/iX system.

The HP-UX version of the REMSH client has an optional -n parameter that tells the client to not read from STDIN. Due to differences between MPE IO and UNIX IO the -n parameter has been hard coded into the MPE/iX client.

The HP-UX REMSH client also allows rlogin and rexec functionality. Since this implementation was designed to address the needs of users attempting to access HP-UX commands and scripts from MPE/iX stream jobs, we chose not to implement any feature needing interactive input with the remote system.

Transaction Manager

by Nagarajan and Jyothi
Commercial Systems Division

Overview

Transaction Manager (XM) is an integral part of the MPE/iX file system. It provides a facility to log all the transactions that are done on a file that is attached to XM via a system intrinsic. It ensures data integrity by protecting such files across system crashes.

Files on MPE/iX are of two categories: system files and user files. System files are label tables, directories and SSM bitmaps. User files are those created by the user. All the transactions on system files are logged onto an XM system log file on the master volume of each volume set, just as all the transactions on user files (such as IMAGE/SQL and KSAM files) are logged onto an XM user log file on the master volume.

Both these XM log files are (magic) system files and are created at volume set creation time with a fixed size.

A System Abort 2200 is generated from Transaction Management (XM) subsystem when it cannot reliably continue logging data to the system XM log file. The system XM log file is a 12Mb object allocated contiguously on disk. XM divides this into two portions called log halves, each being 6Mb in size. While data is being logged into one half, the other half is being emptied. The emptying happens when the transactions are committed and the log half is posted to disk. If we fill one half and discover that the other half is not yet empty, XM will generate SA 2200 in order to preserve the data still in the second log half. This scheme ensures data integrity by not allowing uncommitted data to be overwritten.

Typical long-running system transactions are those that involve insertion of a file into a group/directory at the beginning, initializing file contents at creation time, and others.

System Log Expansion

As the higher-end models of HP3000 become faster and faster, the frequency of occurrence of SA 2200 has been going up. To alleviate the severity of this problem, a new solution to expand the XM system log per volume set is being provided. This will allow more logging to be done before the log gets full.

Commands to Expand

The VOLUTIL tool is enhanced to provide two new commands to facilitate the expansion (or truncation after initial expansion) of the XM system log file. They are as follows:

1. showsyslog *<volset-name>*

   This command accepts a valid MPE volume set name and shows the current size of the XM system log in sectors. For this to work correctly, the master volume has to be mounted in MASTER state. The default size is 52128 sectors allocated at volset creation time.

2. altersyslog *<volset-name> <new size>*

   This command accepts the volume set name and the new size of XM syslog for it in sectors. It cannot be less than 52128 sectors and cannot be more than 2 GB. This is a very large upper limit but as disk capacities are becoming more and more, one can go for a size of about 100Mbytes. One caution is that LDEV 1 has limited reserved space and you cannot expand the XM syslog beyond that. Also, the two residual logs on the master also will be expanded to match the size of the system log.

   The downside of having a larger system log is that the XM checkpoint process will have more data to write to disk at reaching the end of each syslog half. This can have an effect on response time to on-line users.

Actual Expansion

The actual expansion of the XM system log happens when user boots the system next time. This is because the XM log files cannot be expanded/truncated while in use.

During bootup, just after recovery of the system volset, the XM system log file will be expanded and a message will be written onto the console. Also, the two residual logs that reside on the master will be expanded to match it.

For the user volume sets, right after the recovery of system transactions, the expansion of the system log happens.

You can also truncate the system log after expansion if there are any disk space or performance issues. The VOLUTIL interface is same for this operation too.

The residual logs will not be expanded until they are emptied, that is, until all the members of the given volume set are mounted in the MEMBER state.

Backdating the OS

You can backdate the OS to an earlier version with the system log in
the expanded state. The only condition is that the residual logs also
must have been expanded or truncated to match it before the
backdating is done. That would mean a complete, proper recovery of the
volume set in question has to be done before the backdate operation.

Warnings

The expansion of the system log or the residual logs may fail due to one
or more reasons. In that case, the system will report it through a
warning message (in reverse video) on the console.

If it fails due to lack of disk space, make sure that there is sufficient
contiguous space on the master that would fit the need for the
increment in the system log size. Then reboot the system so as to enable
it to complete the task of matching the sizes of residual logs with that of
XM system log.

Transaction Copy-forward

The Copy-Forward solution eliminates the frequent occurrence of SA
2200 by copying forward the activity records of slow system
transactions into the next log half and continuing logging in the current
log half.

This is an enhancement to FOS and there is no user interface needed to
make this available.

Additional System Requirements

To accomplish the task of copying forward the slow transactions, a new
system file called the `copyfwd` buffer is created on the master of each
volume set. This is done at the time of creating the volume set or at the
first time of booting the system with this version of MPE/iX.

The size of this new buffer is half of the XM system log (which is about
12 MBytes).

When It is Activated

When the XM logger sees slow system transactions at the time of
wrapping back to the start of the circular XM system log file, it
activates this solution to copy forward all such slow transactions first
into the `copyfwd` buffer, sorts them and then continues with logging of
the current transaction. This copying forward is done on such slow
transactions for a certain number of times (about 100 times, as of now),
and then, if they are not over even then, it would issue SA 2200,
concluding that there is something that can put the system in jeopardy.

Backdating the OS

While XM logger is doing `copyfwd` operation, it temporarily elevates the version of the XM system log to a higher value and brings it back to its original value once copy forward is over. But if the system crashes while the XM syslog version is at a higher level, backdating it to an earlier version of the OS wouldn't be possible. The system gives an error and XM recovery fails. So in such situations, you need to complete the recovery and then go for backdating the OS.

Checkpoint Improvement

Checkpoint utility is part of XM, and it is invoked when one-half of the XM log is full. It writes out all the dirty data pages from that half to the disk. During this activity, the system on-line response sometimes is impacted due to the high priority nature of its disk I/O and also due to the fact that you can checkpoint only one-half at a time. This makes the logging slow down if the checkpoint process is slow or it holds up critical system resources for too long. This becomes critical on high end MP machines. This solution attempts to improve the performance of XM checkpoint process so that on-line response becomes better and transaction throughput increases.

This solution achieves an improvement in performance by not holding one of the system-wide semaphores for too long while it's scanning the entire virtual address range (could be 4 GB) of a modified file. It remembers the dirty page ranges for each modified file in the log half being checked and does posting to disk only for those ranges.

This solution affects only those files that are attached to XM user log, typically IMAGE/SQL and KSAM files. It has no effect on system files such as directories and label tables.

Enabling or Disabling It

The `VOLUTIL` tool is enhanced to provide two new commands to enable or disable this solution for files on a given volume set. By default this solution is disabled, which means the checkpoint process will scan the entire file VA range for dirty pages.

The commands are as follows:

1. `showchkptstat` *<volset-name>*

   This command shows the current state of the solution, that is, whether it is enabled or disabled for the given volume set.

2. `alterchkptstat` *<volset-name>* ENABLE | DISABLE

   This command enables or disables the solution for the given volume set. If it's already in the given state, it will give a warning to that effect.

You need SM capabilities to use these two commands.

Since this solution remembers the dirty page ranges in a data structure associated with each open file, it can't affect files that were open before giving this request. However, it will come into effect for the files that were open (for the first time) after giving the request. So the best way to enable or disable the Checkpoint Improvement is to reboot the machine after giving the command.

Backdating the OS

You can backdate the OS to an earlier version without any problems since this solution doesn't put the file system in an advanced version or make any non-backward compatible changes to the file system.

New Features for DTS/TIO Subsystem

by Wendy Cheng
Commercial Systems Division

Introduction

The Datacommunications and Terminal Subsystem (DTS) consists of host (HP 3000) based software and the DTC (Datacommunications and Terminal Controller) terminal server.

The DTC family of products supports X.25 communication and Telnet access to HP 3000 systems, and TCP/IP communication to HP 9000 and non-HP systems.

The DTS allows terminal, PCs in terminal emulation mode, serial printer, and other asynchronous devices to communicate with LAN-based hosts in HP-only and multivendor networking environments.

New Features

The DTS/TIO subsystem provides the following new functionality for MPE/iX 6.0 release:

- Support of 400 DTCs.

- Support of 2000 device classes

- Configuration of the maximum number of sessions for a terminal switching port on the Host-based DTC management.

- Configuration of the number of hosts sharing the printer for a printer switching port on the Host-based DTC management.

The following enhancements were made to MPE/iX release 6.0 for the DTS/TIO subsystem to provide support for all DTC port configuration parameters of DTC version 14.4 for Host-based DTC management. Prior to release 6.0, these features were only available on the OpenView DTC Manager in a PC-based management configuration.

These new features are as follows:

- Three new modem types for attached device. The new modem types are US modem In Out, European modem In Out, and Five wires modem.

- Two new data length/parity settings. The DTS/TIO software has been enhanced to support two new parity types. The two new parity types are 8 bits data length with even parity and 8 bits data length with odd parity.

- Configuration to enable or disable flow control between the DTC and the attached device.

- Configuration of the number of stop bits on Host-based DTC ports.

- Specification of whether "Keep Alive" mechanism is to be activated on Host-based DTC ports to detect if connection is still there.

- Specification of whether port configuration parameters- data length and parity- have to be forced when binary mode is selected.

- Dynamic Configuration of new port configuration parameters.

With these DTS/TIO enhancements to MPE/iX 6.0, you can use NMMGR to configure all DTC port configuration functionality that is available in DTC download version 14.4 using Host-Based DTC management.

See the manual *Configuring System for Terminals, Printers, and Other Serial Devices* for detailed descriptions of these new features.

Support of 400 DTCs

Users can now configure up to 400 DTCs on the HP 3000 host at a time. This is up from the earlier limit of 120 DTCs in MPE/iX 5.5 release. Thus the number of DTCs that can be connected to a HP 3000 is significantly increased.

The maximum number of LDEVs supported on the HP3000 is still 4649. This means that if a high number of DTCs are configured then the DTCs can not have all ports configured as nailed ldevs. The total number of supported LDEVs can be spread over the configured DTCs in any way you want.

Support of 2000 device classes

A device class is a name that can be assigned to a single or a group of ldevs. This enables the user to refer to the ldev or the group of ldevs by a name. For example, the ldev associated with a tape driver is referred to as TAPE. Similarly, LP and DISK are the device classes for the system printer and disks respectively.

The number of device classes supported has been increased from 1000 to 2000. This enhancement increased the flexibility for those users that need many devices classes.

Three New Modem Types for Attached

To give more flexibility, support has been added for three new modem types: US modem In Out, European modem In Out, and Five wires modem. However, these new modem types will be supported for the new types of DTCs only: DTC 72MX, DTC 16iX, and DTC 16MX.

The complete list of supported modem types with MPE/iX 6.0 is:

- US modem In Out

- European modem In Out

- Five wires modem

- US modem

- European modem

- None (no modem)

Data Length/Parity List Support

With MPE/iX release 6.0, you can now use NMMGR to configure the two new parity types in the Host-based terminal, printer and host profiles. These two new parity types are: 8 bits data length with even parity and 8 bits data length with odd parity. These parity settings were added in order to provide support for serial devices that need to use these settings. Devices that would use these need to be able to send and receive binary data and also need to check parity.

These two new parity types are only supported for the newer DTCs: DTC 72MX, DTC 16iX, DTC 16MX and DTC 16RX. They are not supported for the DTC 48 or older DTCs. The full list of supported parity setting with MPE/iX release 6.0 is as given:

- 7 bit data length with even parity

- 7 bit data length with odd parity

- 7 bits data length with parity forced to 0.

- 7 bits data length with parity forced to 1.

- 8 bits data length with even parity.

- 8 bits data length with odd parity.

- 8 bits data length with none parity.

Number of stop bits

With MPE/iX release 6.0, you can now configure the number of stop bits to be used for asynchronous characters in the host_based terminal/printer profiles. The possible values are 1 or 2 stop bits. This new configuration field will be supported for the new types of DTCs only: DTC 72MX, DTC 16iX, and DTC 16MX.

Enable Flow Control

When TIO terminals, printers and host ports are created, one of the features that can be configured is a parameter that is sent to the port that determines the data flow control behavior between DTC port and connection device.

With MPE/iX release 6.0, you can now use NMMGR to configure a parameter in the Host-based TIO profiles to enable or disable flow control. If this feature is enabled in NMMGR for terminal/ printer, the data flow between the DTC port and the connection device is made with Xon/Xoff characters. If this feature is disabled, the XON/XOFF characters sent from device are not interpreted by DTC. This function is also known as "software handshake." This new feature can be supported on all DTCs except for DTC 16RX. Any value entered for this type will be ignored.

Enable Keep Alive

This new feature allows you to specify whether "keep alive" mechanism is to be activated. When there is no data transfer between the DTC and the system, then a mechanism called "keep alive" is activated to detect if the connection is still there. This new feature is only applicable to the telnet/TCP connections. It is only available for switching screens for terminal, printer and host profiles.

If this configuration is enabled, the "keep alive" mechanism is to be activated to detect if a connection is still there when there is no data transfer between the DTC and the system. This new feature can be supported on all DTCs except DTC48 without the memory extension.

Save Port Config Binary

This new feature allows you to specify whether the port configuration parameters- data length and parity get forced when binary mode is selected. If set to YES, then the port configuration remains the same when binary mode is selected. If set to NO, then data length is 8 bits and parity is forced to none when binary mode is selected. This new feature is only applicable to telnet/TCP connections. This parameter is accessed from the terminal and printer profile switching configuration screen and the host profile screen.

This new feature can apply only to the new DTCs — DTC 72MX, DTC 16MX and DTC 16iX.

Specify the Maximum Number of Sessions for a Switching Port

Host-based terminal switching allows you to connect to other systems on the LAN from a terminal connected to a DTC that is using host-based management. In previous releases, each port on the DTC was restricted to one connection per port. Connections from PAD ports are still restricted to the system that is managing the DTC. To connect to these systems, you needed to log off or close the connection before connecting to an other system.

With MPE/iX release 6.0, you can now use NMMGR to configure each DTC port to allow up to 5 connections at the same timer. If connections are to HP3000 systems using Host-based management, then only one

connection is allowed to each system. This is because each system must configure the DTC port on a specific LDEV (nailed devices). if the connections are to HP3000 systems using PC-based management, then more than one connections is allowed. This is because it is allowed to configure the DTC port to use non-nailed LDEVs. If the DTC port on the PC-based system is configured as nailed, then only one connection is allowed to that system.

This feature is configured on the terminal profile switching configuration screen. The maximum number of connections is five and the default is one.

Number Of Hosts Sharing Printer

Host-based printer switching allows the serial printer connected to a DTC port to be shared by more than one HP 3000 host. You can now use NMMGR to configure the DTC port to allow a printer to be shared by up to 5 hosts. This feature is configured on the printer profile switching configuration screen. The maximum number of systems that can share at one time is 5 and default is 1. Thus the number of HP 3000 hosts that can share a printer is increased.

Dynamic Configuration of New Port Fields

Dynamic Configuration provides the capability to perform DTS/TIO configuration tasks without the need to shutdown the HP 3000 system. With MPE/iX release 6.0, you can now change the following new fields in a terminal, printer or host profile dynamically:

- Change "Data Length/Parity list" field in a terminal or printer profile.

- Change "Number of Stop bits" field in a terminal or printer profile.

- Change "Enable Flow Control" field in a terminal or printer profile.

- Change "Enable Keep Alive" field in the profile for a switching port.

- Change "Maximum Number Of Sessions" field in the terminal profile for a switching port.

- Change "Number of Hosts Sharing Printer" field in the printer profile for a switching port.

- Change "Save Port Config in Binary" field in the terminal and printer profile for a switching port.

You can now make dynamic configuration changes of these new fields by using NMMGR to modify the configuration file NMCONFIG.PUB.SYS, then invoking the command DTCCNTRL. DTCCNTRL can be invoked by NMMGR after DTS validation has successfully completed, or later by you from the CI prompt.

Introducing ODBCLink/SE

by Dawna Whitehead
M.B. Foster Associates Limited

Overview

ODBCLink/SE is a replacement for the HP PC API. It is an implementation of Microsoft's Open Database Connectivity (ODBC) interface that enables Microsoft Windows based applications and tools to access IMAGE/SQL on the HP 3000 and ALLBASE/SQL on the HP 3000 and HP 9000, in a client/server environment. ODBCLink/SE is a part of IMAGE/SQL and ALLBASE/SQL.

In this environment, application developers and end-users can take advantage of the PC's graphical user interface (GUI) and processing power, while relying on the security, integrity, and database management capabilities of ALLBASE/SQL and IMAGE/SQL.

HP PC API with Gupta can co-exist with ODBCLink/SE, but it will no longer be supported, and will not be distributed in MPE 6.0 and subsequent releases.

ODBCLink/SE server runs under MPE/iX 5.0 or higher on the HP 3000 server, or under HP-UX Versions 9.x and 10.x on the HP 9000 Series 7xx and 8xx servers. The client side runs under Microsoft Windows 3.1 and 3.11, Windows95, or WindowsNT V3.51 and V4.0. Connection via Winsock is available in both the 16-bit and the new 32-bit version.

The driver can be used in two ways: by direct calls from a Windows program, such as Foxbase, or through an ODBC-compliant application, such as Microsoft Access or Visual Basic.

The ODBC `LongVarBinary` data type is supported for binary large objects (BLOBs), such as compressed photographs or document images.

ODBC Level 2 Compliance

ODBCLink/SE is 95 percent ODBC Level 2 compliant. These exceptions are not supported:

- SQLBrowseConnect
- SQLExtendedFetch
- SQLSetPos
- SQLSetStmtOptions
- Scalar functions
- SQL_BIT, SQL_TINYINT, and SQL_BIGINT data types

Client/Server Interface

The networked HP 3000 Series 9xx with ALLBASE/SQL and IMAGE/SQL (or the HP 9000 Series 7xx or 8xx with ALLBASE/SQL) provide the relational database environment on the server.

Security is provided by the MPE/iX or HP-UX logon system and ALLBASE/SQL or IMAGE/SQL.

ODBCLink/SE connects the client to the server with a collection of dynamic link libraries (DLLs) on the client and runs under Microsoft

Windows 3.1 and 3.11, Windows95, or WindowsNT 3.51 and 4.0.

Client application requests are routed over the network to an ALLBASE/SQL or IMAGE/SQL database on the server, and replies are returned to the client application. A listener process establishes the connection between the client application and the target database on the server. The listener works with HP ThinLAN 3000/iX on the HP 3000 or ARPA Services on the HP 9000.

With ODBCLink/SE and application software on the client, you can develop SQL applications, generate reports, and query ALLBASE/SQL or IMAGE/SQL on the server. The tested client applications include the following as well as many more:

Table 10-3   **ODBCLink/SE Tested Client Applications**

Cognos Axiant	Paradox
Cognos Impromptu	PowerBuilder
Jetform	Visual Basic
Lotus 1-2-3	Visual
Microsoft Access	Visual FoxPro
Microsoft Query	

Requirements

HP 3000 server requirements:

- MPE/iX Version 5.0 or 5.5 (or later)

- IMAGE/SQL Version B.G2.03 (or later)

- ALLBASE/SQL Version A.G2.00 (or later)

- ThinLANLink/XL must be configured and turned on

HP 9000 server requirements:

- HP-UX Version 9.0 or 10.0 (or later)

- ALLBASE/SQL Version A.G2.01 (or later)

- ARPA Services must be configured and turned on

Client requirements:

- 486 processor at a minimum, preferably a Pentium

- 8 MB memory at a minimum, preferably 16 MB

- 5 MB of free disk space

- Windows 3.1 or 3.11, Windows95, or WindowsNT V3.51 or 4.0

- Microsoft or WRQ TCP/IP software

Software provided:

- Server program for the HP 3000 or HP 9000 as appropriate

- Client programs for both 16- and 32-bit platforms.

  When using a 32-bit operating system, you should use the 32-bit
  driver if possible. However, if you have a 16-bit application running
  under Win95 or WinNT, you must install the 16-bit version of the
  driver. On Windows95 or WindowsNT, you can run both the 16- and
  32-bit versions of ODBCLink/SE. 16- and 32-bit data-sources are
  maintained separately and coexist.

- Dr. DeeBeeSpy from Syware Inc. for tracing calls to the ODBC driver

Migration from HP PC API

A data migration tool is provided to migrate data sources created for
HP PC API to ODBCLink/SE. Once the data sources are identified, the
translation is performed automatically.

Performance Considerations

ODBC access to the server is very fast. You should note the following
performance exceptions:

- Reading and sorting a large table may prevent the server from
  returning control to the client until the entire database has been
  read and sorted.

- Opening a large table in MS-Access will cause a screen full of data to
  be displayed and then appear to wait for user input. In many cases,
  data is still being downloaded to the client. Careful thought needs to
  be employed before granting ODBC access to large tables on the
  server.

- Using dynamic substitution parameters speeds up your application,
  if you are writing your own SQL to do multi-row inserts or updates.

- In general, ODBCLink/SE will not complete any request faster than
  ISQL would, and could be slower due to network overhead.

Supported Functions and Commands

These are the ALLBASE/SQL and IMAGE/SQL functions and commands supported by ODBCLink/SE:

- The complete ALLBASE/SQL and IMAGE/SQL syntax dynamic SQL

- All SQL functions

You can use the ODBC `SQLGetFunctions` call to obtain a list of the supported ODBCLink/SE driver functions.

Areas Not Fully Implemented

Some areas of certain functions are not fully implemented. They are as follows:

- SQLColAttributes

  All functions will receive a return value of:

  `Not  Implemented`

  except the following supported options:

Table 10-4   **SQLColAttributes Supported Functions**

SQL_COLUMN_COUNT	SQL_COLUMN_NAME
SQL_COLUMN_TYPE	SQL_COLUMN_LENGTH
SQL_COLUMN_PRECISION	SQL_COLUMN_SCALE
SQL_COLUMN_DISPLAY_SIZE	SQL_COLUMN_NULLABLE

- SQLSetConnectOptions

  All functions will return this message:

  `Driver not Capable (SQLState S1C00)`

  except the following supported options:

Table 10-5   **SQLSetConnect Supported Options**

SQL_AUTOCOMMIT	1=ON (default) 0=OFF
SQL_OPT_TRACE	supported by Driver Manager
SQL_OPT_TRACEFILE	supported by Driver Manager
SQL_TRANSLATE_DLL	supported
SQL_TRANSLATE_OPTION	supported
SQL_TXN_ISOLATION	supported
SQL_TXN_READ_UNCOMMITTED	Read Uncommitted (RU)
SQL_TXN_READ_COMMITTED	Read Committed (RC)

SQL_TXN_REPEATABLE_READ	Repeatable Read (RR)
SQL_TXN_SERIALIZABLE	Repeatable Read (RR)
SQL_TXN_VERSIONING	Not Implemented

• SQLSetStmtOptions are not supported.

The driver will return either of these messages:

```
Driver not Capable (SQLState S1C00)
Option value changed (SQLState 01S02)
```

Supported Data Types

ODBCLink/SE supports ALLBASE/SQL and IMAGE/SQL data types.

The following table shows the correspondence between ALLBASE/SQL or IMAGE/SQL data types and ODBC data types.

Table 10-6   **Supported Data Types**

ALLBASE/SQL or IMAGE/SQL Data Type	ODBC Data Type
CHAR, max length <255	SQL_CHAR
CHAR, max length >=255	SQL_LONGVARCHAR
VARCHAR, max length <255	SQL_VARCHAR
VARCHAR, max length >=255	SQL_LONGVARCHAR
BINARY, length <256	SQL_BINARY
BINARY, length >=256	SQL_LONGVARBINARY
VARBINARY <256	SQL_VARBINARY
VARBINARY >=256	SQL-LONGVARBINARY
LONG BINARY	SQL_LONGVARBINARY
LONG VARBINARY	SQL_LONGVARBINARY
INTEGER (32-bit)	SQL_INTEGER
SMALLINT (16-bit)	SQL_SMALLINT
DECIMAL (Internal representation is packed decimal)	SQL_DECIMAL
Image Zoned (Z)	SQL_DECIMAL
Image Packed (P)	SQL_DECIMAL
Image I3 (48-bit integer)	SQL_DECIMAL
Image I4 (64-bit integer)	SQL_DECIMAL

ALLBASE/SQL or IMAGE/SQL Data Type	ODBC Data Type
FLOAT(24), REAL, or Image E2	SQL_REAL
FLOAT(53), DOUBLE PRECISION, or Image E4	SQL_DOUBLE
Image R2 (non-IEEE float)	SQL_REAL
Image R4 (non-IEEE float)	SQL_DOUBLE
DATE	SQL_DATE
TIME	SQL_TIME
DATETIME	SQL_TIMESTAMP
INTERVAL	SQL_CHAR

ALLBASE/SQL Storage space for BLOBs should be created using a LONG VARBINARY column.

Stored Procedures

ODBCLink/SE implements all types of ALLBASE/SQL stored procedures, including those which return multiple-format results sets. You can list the stored procedures available to your ODBC session by calling `SQLProcedures`. Parameter information for stored procedures is obtained by calling `SQLProcedureColumns`.

Features Specific to ODBCLink/SE

The following is a list of features that are unique or specific to ODBCLink/SE and some details about them:

- Primary key name returned by SQLStatistics

  Most ODBC applications including MS Access, Visual Basic, and PowerBuilder require a unique primary key for a table to be updatable.

- Reading or writing to ALLBASE/SQL `LongVarBinary` items.

  Any file residing on the Hewlett-Packard server can be inserted into an ALLBASE/SQL `LongVarBinary` field with the `INSERT INTO` SQL command.

- Maximum number of statements

  ODBCLink/SE supports up to 50 concurrent statements, or cursors, per connection.

To use multiple statements when you are not writing your own SQL application, tell ODBCLink/SE to have `SQLGetInfo` report more than one statement per connection.

ALLBASE/SQL Restrictions

These are the ALLBASE/SQL and IMAGE/SQL restrictions on the ODBC grammar. These are summarized in the following table:

Table 10-7   **ALLBASE/SQL Restrictions**

Statement	Programming Considerations
CREATE TABLE	UNIQUE PRIMARY KEY must follow NOT NULL
DROP TABLE	ALLBASE/SQL does not provide CASCADE or RESTRICT
REVOKE	ALLBASE/SQL does not provide RESTRICT

- ALLBASE/SQL provides a `DATETIME` data type similar to `TIMESTAMP`.

- ALLBASE/SQL does not implement optimistic locking.

  Optimistic locking means that the DBMS does not lock data until just before an update is made, thus improving concurrency because locks are not held for long.

ANSI Character Set

The HP 3000 and HP 9000 both use the default ROMAN8 character set. Most applications running under Microsoft Windows use the 8-bit ANSI (ISO 8859/1) character set to provide support for Western European languages (including American). The first 127 characters are the same for both ROMAN8 and ANSI character sets, so only the extended and special characters are different.

If your client application uses these extended or special characters, then the data from the server must be converted from ROMAN8 to ANSI, and the data returning to the server must be converted from ANSI to ROMAN8. A translation DLL has been provided to accomplish this task.

Unsupported ALLBASE/SQL and IMAGE/SQL Statements

There are several reasons why some ALLBASE/SQL and IMAGE/SQL statements are not supported in the ODBCLink/SE environment:

- ODBCLink/SE does not support embedded SQL, therefore statements that work only with embedded SQL are not supported. See details that follow.

- Statements that relate to functions, such as connection, that have been replaced by functions resident on the client. See details that follow.

- Some functions are best restricted to the DBA; statements which control these functions are accessible only through Interactive Structured Query Language (ISQL) on the server.

- ODBCLink/SE internally prepares an SQL statement to be dynamically preprocessed. Therefore, you cannot use some statements such as EXECUTE IMMEDIATE.

Statements that Work Only with Embedded SQL

Client application software that accesses ALLBASE/SQL and IMAGE/SQL databases through ODBCLink/SE does not support embedded SQL. The application software calls the database functions directly; there is no need for preprocessing.

Because the following ALLBASE/SQL and IMAGE/SQL statements function only with embedded SQL, they are not supported by ODBCLink/SE:

Table 10-8   **Functions Not Supported by ODBCLink/SE**

CLOSE CURSOR	BEGIN DECLARE SECTION
DECLARE CURSOR	DESCRIBE
END DECLARE SECTION	EXECUTE
EXECUTE IMMEDIATE	FETCH
INCLUDE	OPEN
PREPARE	REFETCH
SQLEXPLAIN	WHENEVER

Statements Replaced by Functions on the Client

Some ALLBASE/SQL and IMAGE/SQL functions have been moved from the server to the client with ODBCLink/SE.

The following ALLBASE/SQL and IMAGE/SQL statements are not supported because they have been replaced by other functions resident on the client:

Table 10-9   **Replaced Functions**

CONNECT	DISCONNECT
RELEASE	RESET
SET CONNECT	SET MULTITRANSACTION
START DBE	START DBE NEW
START DBE NEWLOG	STOP DBE

Utilities

The utilities provided with ODBCLink/SE do the following:

- Test and verify that the driver is functioning correctly.

- Determine which version of ODBCLink/SE is running.

- Delete unwanted drivers.

- Delete unwanted translation DLL's.

Tools on the Server

The following server tools are available to allow monitoring of the ODBCLink/SE listener process:

- HP GlancePlus/XL (HP 3000) Program

- HP GlancePlus/UX (HP 9000) Program

- HP-UX Process Status (ps) Command

- Monitoring ALLBASE/SQL Activity with `SQLMON`

- ODBCUTSE for testing/troubleshooting the host connection or table access

- The parameter to have error messages logged to the `ODBCLOG` file: `Trace SQL calls on the server`.

Tools on the Client

The tools on the client that you can use to monitor or troubleshoot the ODBCLink/SE client process:

- The configuration parameter, `Trace ODBC calls on the client`, is used to log information to the ODBCLink.LOG file.

- The application, Dr. DeeBeeSpy, is used to trace calls to the ODBC driver. Dr. DeeBeeSpy is shipped with ODBCLink/SE.

ODBCLink

ODBCLink/SE is a "Special Edition" of M. B. Foster Associates Limited ODBCLink option of DataExpress. ODBCLink, the full product, offers the following features in addition to the features of ODBCLink/SE:

- Support for serial and/or modem connections

- Read/Write access to KSAM files

- Read/Write access to MPE files

- Native read/write access to TurboIMAGE databases

- Support for PowerHouse dictionaries including PDL and PowerHouse subfiles

- Read/Write access to Suprtool, Dictionary/3000, HP SD, and PowerHouse Qschemac via DataExpress DataViews

- Support for TPI keyword indexes used by Omnidex and Superdex

- Access to Oracle databases on HP 3000 and HP 9000 platforms

- Access to Multiple DBEnvironments

- Access to HP intrinsics and procedures via remote procedure call

- Enhanced security via field level access controls

You may obtain an upgrade from ODBCLink/SE to the full product ODBCLink by contacting M. B. Foster Associates at 1-800-ANSWERS, (800-267-9377) or 613-448-2333, or by visiting the M. B. Foster website at `http://www.mbfoster.com`.

Documentation

For further information on ODBCLink/SE, refer to the *ODBCLink/SE Reference Manual* (36217-90406) (revised June 1998).

Introducing Samba/iX

by Wendy Cheng
Commercial Systems Division

Overview

Samba/iX is a suite of programs which work together to allow clients to access a server's file space and printers via the Server Message Block (SMB) protocol. Samba is the Server Message Block (SMB) file server that runs on MPE/iX shell operating system on MPE/iX release 6.0. It allows the MPE/iX shell operating system to act as a file and printer server for SMB clients which are mainly for Microsoft LAN Manager and compatible products. These compatible clients can be Windows for Workgroups, Windows 95, Windows NT, or other clients.

Initially Samba/iX was written for UNIX, now it also runs on MPE/iX, NT, and other operating systems. Samba/iX provides the file and printer sharing services between NT and HP 3000 systems. In practice, the user can redirect disks and printers to MPE disks and printers from Windows NT, Windows 95, LAN Manager, or other clients.

A generic UNIX program that is part of the Samba/iX suite has also been ported to MPE/iX Shell operating system. This program allows UNIX or MPE users to use an FTP-like interface to access file space and printers on any other SMB servers. This capability enables these operating systems to act like a LAN server or Windows NT server.

Major Components

Samba/iX contains programs like SMBD and NMBD to provide server functionality and also contains utilities like SMBCLIENT or NMBLOOKUP to provide client functionality. The following lists the major components of Samba/iX:

- A SMBD program is part of Samba/iX suite. A SMBD process is a server that can provide file and printer services to clients using the SMB protocol. This is compatible with the LAN Manager protocol and can service LAN Manager clients. This program provides selected server directories accessible across the LAN. Typical clients are PCs that use Windows 95 or Windows NT to access the server "share" as "network drive."

- A NMBD program provides browsing support, it helps clients to view which workgroups, servers and shares are available on a LAN.

- A `SMBCLIENT` utility is part of Samba suite. `SMBCLIENT` is a client that can access to a SMB (Server Message Block) server. It provides a user interface similar to that of FTP program. Mainly, it is used on UNIX based servers as they typically do not have a Windows File Manager available.

- A `NMBLOOKUP` utility allows to query `WINS` name-servers and to assist with manual `NETBIOS` name resolution.

- A `TESTPARMS` program provides the user to test the Samba configuration file for correctness. If this program reports no problems, the `SMBD` process will successfully load the configuration.

- A `SAMP-SMB.CNF` file is a sample Samba/iX configuration file that contains the runtime configuration information for the `SMBD` program. This file consists of sections and parameters. A global section is defined to contain the general parameters and defaults. This file also contains the information required for each share and defines attributes like associated directory path, read or write access for each share. This file resides in the `/usr/local/samba/lib` directory.

File and Printer Sharing Using Samba/iX

With the availability of Samba/iX on a MPE/iX 6.0 system, you can set up your HP 3000 system as a Samba server. In order to set up the HP 3000 system as a Samba server, you have to create a configuration file that basically contains the information required for file and printer sharing services. This Samba configuration file contains information on various shares you may access. This file resides in `/usr/local/samba/lib` directory and is read when the `SMBD` server is run. Samba/iX supports both file and printer sharing services. As a result, you can define not only file shares but also printer shares.

A listener process is then started under the control of INETD. The listener process waits for incoming client connection requests and creates child processes as needed. The Samba/iX server side validates user name and password which are sent by the client and grants access to the requested share if it is appropriate.

When the client connection is established, the client can then send requests to the Samba/iX server that allow them to access shares, open files, read and write files.

At the DOS prompt on a PC client, you can use `NET VIEW` command to retrieve information about available shares from the Samba/iX server. You can use NET USE command to connect to server shares and to map them to disk drivers. These network drivers are then accessible to regular PC applications. You don't need to use DOS commands for connecting or accessing shares, this can also be done from the Windows File Manager.

You can also access server based printers using Samba/iX. Printer sharing works in such a way that the client creates a file on server directory associated with the printer, and then the server process triggers a configurable command to push that file into the MPE spooler.

Starting Samba/iX

Before you start to run Samba/iX server or client components, you should have set up the TCP/IP networking on your HP 3000 system as well as your PC. On the HP 3000 system side, you should have a proper IP address and subnet mask configured in NMMGR as well as `NETCONTROL START` successfully executed. You must choose to start SMBD and NMBD either as Daemons or under control of INETD.

Disable Resource Sharing

If your system has `NBDAEMON.PUB.HPLANMGR` running, then SMBD and NMBD will not be able to use ports 137, and 139 as NBDAEMON already binds to them. The workaround solution is to stop the PDSERVER process. The NBMON and NBDAEMON processes will not start because of this. This can be done by modifying the file `PDSSERV.NET.SYS` by changing the line 7 from 1 to 0. This will setup PDSSERVE for non-reserved servers.

The following shows the steps of making non-reserved servers:

1. Modify the file `PDSSERV.NET.SYS` and change the line 7 from 1 to 0.

2. Shutdown the network.

3. Stream `JCONFJOB.NET.SYS`.

4. Start the network back up.

5. The nscontrol `status=services` should show non-reserved PDSERVERs.

Verify Link Configuration

The default assumes that LAN link configuration in NMMGR is SYSLINK. You need to run the following command to get IP address and subnet mask of your HP 3000 system, you will need this information for future Samba/iX configuration file updates with the "interfaces" parameter.

1. Log into your MPE system

2. Hello manager.sys

3. Enter the command **Netcontrol status; net=LAN1**

The following example displays when you run the command **netcontrol status; net = lan1**.

```
    NETWORK NAME:       LAN1
```

```
NETWORK IP ADDRESS:   $0F0DC750 15.13.188.80
NETWORK SUBNET MASK: $0FF000000 255.0.0.0
```

Starting SMBD and NMBD as Daemons

1. Logon as MGR.SAMBA

2. Copy the sample configuration file `smap-smb.conf`, `samp-printcap` and `samp-user.map` to `smb.conf`, `printcap` and `user.map`. Modify the entries to suit your Samba/iX environment. The `samp-smb.conf`, `samp-princap` and `samp-user.map` files reside in the `/usr/local/samba/lib` directory.

3. Check your Samba/iX configuration files with TESTPARM utility. The TESTPARM utility resides in the `/usr/local/samba/bin` directory. Run the following command:
   shell/ix> testparm /usr/local/samba/lib/smb.conf.

4. Start your SMBD listener and NMBD server.

5. Use the jobs supplied as `JSMB.SAMBA.SYS` and `JNMB.SAMBA.SYS` and stream them.

6. Use SHOWJOB to see if the jobs stay alive, it can look as follows:

```
JOBNUM      STATE    JIN      JLIST      JOB NAME

J#30        EXEC     10S      LP         JNMB,MGR.SAMBA

J#31        EXEC     10S      LP         JSMB,MGR.SAMBA
```

Starting SAMBA/iX Under the INETD Control

If you choose to run SMBD and NMBD processes under control of INETD, you should have new entries in `SERVICES.NET.SYS` and `INETDCNF.NET.SYS` and then you have to create symbolic links to make `SERVICES.NET.SYS` link to `/etc/services` and `INETDCNF.NET.SYS` symbolic links to `/etc/inetd.conf` respectively. Perform the following steps:

1. Logon as `manager.sys`.

2. Copy `SERVSAMP.NET.SYS` file to `SERVICES.NET.SYS` if `SERVICES.NET.SYS` doesn't exist. The Following two entries should exist in file `SERVICES.NET.SYS`:

```
nmbp 137/udp

smbp 139/tcp
```

3. Copy `INCNFSMP.NET.SYS` file to `INETDCNF.NET.SYS` if `INETDCNF.NET.SYS` doesn't exist, the following two entries should exist in file `INETDCNF.NET.SYS`:

```
nmbp    dgram   udp    wait   MGR.SAMBA   /SYS/SAMBA/NMBD   nmbd

smbp    stream  tcp    wait   MGR.SAMBA   /SYS/SAMBA/SMBD   smbd
```

4. Use the following two commands to create symbolic links to make `SERVICES.NET.SYS` link to `/etc/services` and `INETDCNF.NET.SYS` links to `/etc/inetd.cnf` respectively:

```
:newlink /etc/services, /SYS/NET/SERVICES

:newlink /etc/inetd.conf, /SYS/NET/INETDCNF
```

5. Stream `JINETD.NET.SYS` or use `INETD.NET.SYS -c` (if INETD job was already running) to start SMND listener and NMBD server.

6. Use `SHOWOUT JOB= Jobnumber`

7. Print `Oxxx.OUT.HPSPOOL` to check for any problems in the spool files.

In case of problems, check for the job listings for useful error messages and look into the Samba/iX log file `/usr/local/samba/var/log.smb` and `log.nmb` for hints. You can control the amount of log messages with the "debug level" directive inside the config file.

Stopping Samba/iX

It is important to shutdown Samba/iX before bringing the system down. You can use the following commands to stop Samba/iX:

1. Use **SHOWJOB** to see if the jobs stay alive, it can look as follows:

```
JOBNUM      STATE      JIN      JLIST      JOB NAME

J#30        EXEC       10S      LP         JNMB,MGR.SAMBA

J#31        EXEC       10S      LP         JSMB,MGR.SAMBA
```

2. Use the following two commands to stop Samba/iX:

```
:abortjob          #smbjobnumber

:abortjob          #nmbjobnumber
```

Logging Method

The Samba/iX programs write trace messages into a log file called `log.smb` or `log.nmb` by default. The log files reside in `/usr/local/samba/var` directory.

The level of detail is controlled by a "debug level" option in the Samba configuration file which is smb.conf file. The debug level is an integer from 0 to 5. The high this value is, the more detail will be logged to the log file about the activities of the server. At level 0, only critical errors and serious warnings will be logged. Level 1 generates a small amount of information about operations. Level above 1 will generate considerable amount of trace messages which can be used for trouble shooting problems.

Samba/iX Version

The current version of Samba/iX on 6.0 is based on Samba/iX 1.9.16p9;
the port uses the LIBBSD package (libraries and include files) available
on Jazz (*http://jazz.external.hp.com*). This is to resolve the BSD style
routines that are not part of POSIX to successfully compile with
HPC/iX (c89 under the POSIX shell.)

Developer's Kit for Java/iX

by *Mike Yawn*
Commercial Systems Division

Background

The Java development environment (including compiler, runtime, and class libraries) was first ported to MPE in 1996, and various versions have been available as unsupported freeware since that time. Beginning with MPE/iX 6.0, a version of Java is now bundled with the MPE operating system.

Java/iX for the HP 3000 was initially developed as an unsupported "freeware" product by engineers within Hewlett-Packard. With the release of MPE/iX version 6.0, the current Java/iX version is being bundled into FOS as the first step in turning Java/iX into a fully supported HP product. While HP is committed to providing support for Java in MPE/iX, the current release contains a few limitations that you should be aware of:

- This first release of Java/iX will be supported by the Response Center on a "best effort" basis.

- Performance has been a secondary concern to reliability in our engineering efforts up to this point. While Java/iX includes HP's own Just-In-Time (JIT) compiler which allows Java programs to be converted into native PA-RISC code at runtime, customers may find that the overall performance for some Java applications will be less at this time than on other platforms.

- This release of Java/iX does not pass 100% of the Java Compatibility Kit test suite, which prevents it from being officially labeled as 'Java Compatible' at this time.

- Sun is developing a functionality specification for Java running on server systems such as the HP 3000. It is likely that we will target this specification for the functionality to be included in future releases of Java/iX. Thus it is possible that some of the Java APIs which are not normally used in server-side Java programs, but which are included in the current Java/iX release, may not be supported in future versions.

Work is actively being done to address these issues.

Installing or Updating Java

If you have not downloaded and installed any version of Java from the JAZZ web server or an Interex freeware tape, then you can safely install the version of Java included with this release of MPE/iX. If you have installed another version of Java, you should check to see whether

the one included with the OS is more recent. If you have installed any version of Java on your system, you can determine the version by typing the following at the POSIX shell prompt (or at a CI prompt if you have set up the proper UDCs to allow Java to be accessed from the Command Interpreter):

```
java -version
```

You can compare this to the version number in the readme file (README.INSTALL.JAVA) after updating your MPE/iX system to 6.0. If the 6.0 version is more recent, you can install it by typing the following at the MPE CI prompt:

```
:STREAM JINSTJDK.INSTALL.JAVA
```

Shell or CI?

You can run Java from the MPE Command Interpreter or from the POSIX shell. These are the differences between the two environments:

- The MPE Command Interpreter will launch Java programs faster, because of the efficiency of MPE's CREATEPROCESS mechanism compared to POSIX fork() and exec().

- The POSIX shell provides intelligent default values for several environment variables, so the user does not, in most cases, have to provide values.

Java uses six environment variables, and Java will be easier for you to work with if you provide values for these via a command file or UDC (if you use Java from the MPE CI) or a shell script or *.profile* (if you use Java from the POSIX shell). These variables are:

- SYSNAME
- PATH (or HPPATH)
- CLASSPATH
- LD_LIBRARY_PATH
- THREADS_FLAG
- DISPLAY

Descriptions of these variables and their typical settings follow.

SYSNAME Variable

Example of setting SYSNAME from the shell:

```
shell/iX> export SYSNAME=hp3000
```

The SYSNAME variable is tested by some of the shell scripts provided with Java. It does not need to be set in the MPE CI environment, since the Java shell scripts are not used when running from the CI. It must be set to hp3000, exactly as shown above.

PATH (or HPPATH) Variable

Example of setting `PATH` from the shell:

```
shell/iX> export PATH=$PATH:/usr/local/java/latest/bin
```

Example of setting `HPPATH` from the CI:

```
:setvar HPPATH HPPATH+",/usr/local/java/latest/bin/PA-RISC/green_threads"
```

These variables are not Java-specific, but rather the standard path variables used to search for executable program files. The Java executable program files are in the directory shown for the `HPPATH` variable. When executing Java from the CI, the actual program binary files are executed. When executing Java from the shell, we do not execute the Java program files directly, but instead execute shell "wrapper" scripts that are in the directory indicated by the `PATH` statement above. These shell scripts set standard env variables for us (such as `CLASSPATH` and `LD_LIBRARY_PATH`, shown below), so that we don't need to specify values for these unless we wish to add to or replace the standard settings.

CLASSPATH Variable

Example of setting `CLASSPATH` from the shell:

```
shell/iX> export CLASSPATH=$CLASSPATH:(add your stuff here)
```

Example of setting `CLASSPATH` from the CI:

```
:setvar CLASSPATH "/usr/local/java/latest/lib/classes.zip:(add your stuff here)"
```

Like the `PATH` variable, the `CLASSPATH` variable is a colon-separated list of directories (or zipped archive files) to search. While the `PATH` and `HPPATH` variables list the directories to search for program files, the `CLASSPATH` variable lists the directories to search for Java classes.

The standard Java classes (such as those that make up the `java.lang`, `java.io`, and `java.net` packages) are all bundled together in a single ZIP format archive file, called `classes.zip` and stored at the location shown in the `:SETVAR` command above. When running from the shell, the wrapper script sets the `CLASSPATH` for you to include this file; when running from the CI, you must create a `CLASSPATH` variable to include the `classes.zip` file. In either case, any additional classes, such as application or applet-specific classes you have written or obtained from a third party, must also be listed in the `CLASSPATH`. Because locations on the `CLASSPATH` are searched in the order you specify, best performance is obtained when you order the directories on the `CLASSPATH` so that the directories containing the most frequently-accessed classes are listed first. (In most cases, this means that the standard `classes.zip` file should come before any others).

LD_LIBRARY_PATH Variable.

Example of setting `LD_LIBRARY_PATH` from the shell:

```
shell/iX> export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:(add your stuff here)
```

Example of setting `LD_LIBRARY_PATH` from the CI:

```
:setvar LD_LIBRARY_PATH "/usr/local/java/latest/lib/PA-RISC/green_threa
ds:
(add your stuff here)"
```

Again, we specify a colon-separated list of directories to search. In this case, we're providing the locations to search for library (XL) files that contain native (that is, platform-specific) code. The java classes—all the files that go into `classes.zip`—are the same on all platforms. The platform-specific code is written in C, and compiled into shared libraries (XLs on MPE). The standard Java runtime libraries are in the location shown on the MPE `:SETVAR` command above; the Java shell wrappers will include this location automatically when running Java from the shell. Users usually do not need to add additional locations to this variable; it is used only if you have written (or obtained from a third party) Java code that includes native (non-portable) code. One example of such a library is the TurboIMAGE/XL class library for MPE, which includes an XL that has the interfaces to the IMAGE/SQL intrinsics.

THREADS_FLAG Variable

Example of setting from the shell:

```
shell/iX> export THREADS_FLAG=green
```

The `THREADS_FLAG` variable is only needed when running from the shell, and tells the java shell wrappers to use the binaries found in the `green_threads` subdirectory (when running from the CI, we've already indicated this in our setting of the `HPPATH` variable). There are two possible ways to implement multithreading when porting Java to a new platform: using kernel threads (with binaries in the `native_threads` directory) or using library threads (with binaries in the `green_threads` directory). The current MPE implementation uses `green_threads`, although a `native_threads` version may be developed in the future. If so, the `THREADS_FLAG` variable could be set to `native` to select the kernel threads implementation (and the `HPPATH` variable would be changed for CI users to select the `native_threads` directory).

DISPLAY Variable

Example of setting from the shell:

```
shell/iX> export DISPLAY=sysname:0
```

Example of setting from the CI:

```
:setvar DISPLAY "sysname:0"
```

The `DISPLAY` value is used by the AWT and Applet classes to locate the display device. Only X-Windows compatible display devices (UNIX workstations, X-terminals, or PCs running X-Windows emulation software) are supported for the display device. Most HP 3000 Java

users probably won't need the AWT on the HP 3000; user-interface code is usually downloaded to a Web browser or run in some other client-server type of configuration. We certainly recommend this for your own development:

• Design the user interface as a separable module that can be executed on a client, and leave application logic and database access modules on the server.

You may, however, encounter Java applications that aren't designed this way, and the MPE AWT implementation will allow you to execute these applications, if you can provide an X-Windows capable display device accessible from the HP 3000.

Java Directory Structure

Java is installed in MPE's hierarchical file system under the path `/usr/local/java`. There are several additional directory levels beyond this point.

The next level down specifies the Java version. For example, Version 1.1.5 of Java is installed under `/usr/local/java/jdk1.1.5.` There is also a symbolic link called `latest` that you can use to access the most recent version of Java, so that you do not need to change shell scripts, jobstreams, and command files every time you install a new version of Java. Thus, you can use `/usr/local/java/latest/...` as the path to any Java components to use the latest installed version. When you install new versions, you can remove this link and create a new one. If you have software that must be run with a particular version of Java, you can always specify a specific version directory rather than `latest`.

Two major hierarchies, `lib` and `bin`, are under `/usr/local/java/latest`, as well as the optional hierarchies, `demo` and `doc`, and a number of files such as the latest README file.

The `bin` hierarchy contains Java program files and shell scripts used to run those programs. Under `/usr/local/java/latest/bin` are shell scripts used to run each of the Java components (described in the next section). So, for example, `/usr/local/java/latest/bin/java` is the script to run the latest version of the Java interpreter. If the `PATH` (or `HPPATH`) variable is set correctly, it is not necessary to specify this entire path; `java MyClass` would be sufficient.

The actual program binaries are two levels further down in the hierarchy. Under `/usr/local/java/latest/bin` is the directory `PA-RISC`. In the future, an additional subdirectory may be created at this level to contain IA-64 binaries. Under the `PA-RISC` directory is the directory `green_threads`. This directory holds the actual Java programs. Green threads is a simulated threads library provided by JavaSoft. If at some point we choose to create a version of Java that uses MPE's kernel threads, a `native_threads` subdirectory will be

created at this level to hold that version of the binaries. The `lib` hierarchy follows the same structure as the `bin` hierarchy, with `PA-RISC` and `green_threads` subdirectories. One file residing at the top level of the `lib` hierarchy, `classes.zip`, is an archive file that contains all of the standard Java classes. You do not have to unzip it to access this archive (in fact, you can access it faster as a single zip file than as hundreds of individual disk files). At the bottom level of the hierarchy are a number of specialized libraries that are dynamically loaded when needed for certain Java tasks. For example, `libnet.sl` contains the Java networking code, and `libawt.sl` contains code used by the Abstract Windowing Toolkit.

The `demo` directory holds a number of demo programs.

The `doc` directory holds HTML format API documentation.

Using Java

Writing Java Programs

There are no unique tools on the HP 3000 for writing Java programs; you just select your favorite word processor and enter the program text. You can use either a traditional HP 3000 editor such as EDIT/3000 or HP EDIT, or you can use the `vi` editor from within the POSIX shell. The naming of a Java source file should follow the following conventions:

- The name must be followed by the extension `.java`

- The name is case-sensitive. While not a requirement, it is conventional to use mixed case in class names, with an initial capital and additional capitals used if the class name contains more than one word (for example, `HelloWorld.java`).

- Java files must be in bytestream format. VI will create bytestream files for you. If you use another editor such as HPEDIT, you will need to use the shell's `tobyte` command to convert the source file to bytestream format.

Because of these requirements, if you use an editor that does not support full POSIX naming conventions, you must either use file equations or rename your source file to meet the requirements.

If your program is part of a package, it needs to be placed in an HFS directory whose name is the same as the package name.

For an example, consider the following, perhaps the simplest possible Java program:

```
public class HelloWorld
{
    public static void main(String[] args) {
        System.out.println("Hello, World!");
        }
}
```

Because the class name is `HelloWorld`, this program needs to be saved as `HelloWorld.java`.

Compiling a Program

The compiler for Java programs is called javac, so the above program is compiled simply by:

```
javac HelloWorld.java
```

This command can be entered at either a CI prompt or at a shell prompt. The compiler will produce a class file called `HelloWorld.class`.

Executing a Program

To run the above program, type at either the CI or shell prompt:

```
java HelloWorld
```

Note that the `.class` extension isn't needed (and will cause an error if specified).

Where to Go for More Information

For an introductory book on Java, *Java in a Nutshell*(O'Reilly, 1997), now available in a second edition covering JDK 1.1, is an excellent work. Many Java users have recommended *Teach Yourself Java in 21 Days*(O'Reilly, 1997) as another good starting point. Also highly recommended is *Thinking in Java* (Prentice-Hall, 1998) but also downloadable over the Web from *http://www.EckelsObjects.com/javabook.html*.

For more advanced topics, you'll have no problems finding books focused on the AWT, JavaBeans, JDBC, or just about any other Java topic. The books published by O'Reilly and Associates, the JavaSoft Press series published by Prentice Hall, and the official specification books (*The Java Language Specification*, the *Java Virtual Machine Specification*, and the *Java API Specification* (two volumes)), published by Addison-Wesley all are frequently recommended on the Internet by Java users. Your best bet is to ask others, in a forum such as USENET or SIG JAVA, for recommendations, or just be prepared to spend some time browsing at the bookstore before making your selection.

If you want to find resources on the World Wide Web, a search for "Java" in your Web browser will find more hits than you can possibly read in your lifetime. The following two URLs are good starting points:

- The Official JavaSoft home page *http://java.sun.com* is updated very frequently. Check here for new releases of add-on products, such as the Beans Development Kit (BDK), Java Workshop, or Java Web Server. On-line tutorials and reference documents are available, plus access to developer resources and a wealth of other information.

- The Home Page for Java on the HP 3000
  http://jazz.external.hp.com/src/java contains the latest MPE-specific Java information and downloadable software.

SIG JAVA is the Interex Special Interest Group for Java on all HP Platforms. SIG JAVA holds discussions via the JAVA mailing list. You can subscribe by sending an email to *java-request@interex.org* with the message "subscribe java *yourname*" in the body of the message. You can then send email to the entire mailing list by addressing a message to *java@interex.org*.

The *comp.lang.java.** newsgroups are very active, holding discussions on all aspects of Java.

MPE-specific Java issues will also be discussed on the *comp.sys.hp.mpe USENET* newsgroup or its mirrored mailing list, *HP3000-L*.

Hewlett-Packard NetWorker Client for MPE/iX

by Lalitha Pejavar and Alex Early
Commercial Systems Division

Overview

Available now for HP 3000 users is NetWorker from Legato, a client/server application that provides advanced storage management capabilities to a heterogeneous environment of servers and desktop computers.

Benefits

These are the key benefits of using NetWorker:

- Complete Solution for HP Environments

  NetWorker is the most widely-implemented backup utility in open system environments. It provides HP 3000 users with better integration in a multiplatform environment, allowing them to back up data to an UNIX, NT, or Netware server.

- Support for HP's DLT Library System

  NetWorker is fully compatible with HP's Advanced DLT Library System enabling all server and desktop data to be automatically backed up, indexed, and policy-managed for either user-based file recovery or complete disaster recovery.

- Centralized Enterprise Management

  NetWorker's client/server architecture and centralized storage model allow integration into network management frameworks. Legato NetWorker integration modules allows SNMP-based administration of NetWorker servers using HP OpenView, or complete enterprise administration using HP OpenView IT/Operations.

- Superior Protection for On-Line Databases

  NetWorker supports on-line backup and restore operations for Oracle, Informix, and application data of SAP R/3, Oracle Financials and PeopleSoft HRMS.

Features

These are the key features of the NetWorker Client for MPE/iX:

- Provides an easy-to-operate graphical user interface and maintains a file index and media index to track data backed up and recovered.

- Supports a wide range of industry tape (DAT) and DLT libraries and furnishes a sophisticated media manager which allows for unattended backup and restore using autochange library.

- Provides for concurrent device support for backup and recovery operations using up to 16 backup devices simultaneously.

- Utilizes most of the features of 7x24 True-Online Backup since NetWorker uses TurboSTORE/iX 7x24 for backup. However, it will also work with FOS STORE and TurboSTORE/iX.

- Supports automatic backups of MPE volume sets. Also provides the ability to run two or more scheduled backups in parallel.

- Provides an optional Legato NetWorker Archive Application which gives the ability to take a snapshot of finished project files, directories, or MPE volume sets residing on primary media and safely stored on removable media.

Required Hardware and Software

Required hardware:

- HP 3000 any Series 9xx as a client

- HP 9000 server or Net Server

- DLT auto-library or DAT auto-changer

    - HP DLT 4/48, 2/28

    - STK 9714, 9710

- Networking connection (that is, FDDI, LAN, or 100VG-ANY LAN; 100VG ANY-LAN or FDDI LAN recommended)

Required software:

- NetWorker Client for MPE/iX

- NetWorker Server software (`LEGNETWR`), HP-UX Network Edition

- Auto-library license for the server (`LEGAUTOC`)

HP Response Center Support

The NetWorker Client for MPE/iX currently has software supported through Legato, and NetWorker server software (`LEGNETWR`) has software support available through HP or Legato.

Legato NetWorker is the only third-party storage management solution supported by HP Worldwide Customer Support Operations. HP Response Center engineers have been fully trained on NetWorker operations. HP support contracts are available for NetWorker from your HP sales representative.

Documentation

The Hewlett-Packard Network Client Kit (B5475AA) contains the Legato Network Client software, *Legato NetWorker ClientPak for MPE/iX Installation Guide*, and *Legato NetWorker ClientPak for MPE/iX User's Guide*.

FTP Enhancements

by Raghuram B.
Commercial Systems Division

The following is a description of new features available with FTP/iX.

Implementation of FTPSRVR under INETD

Prior to Release 6.0, the FTP/iX server, FTPSRVR, was initiated by the FTPMON process by the `JFTPSTRT.ARPA.SYS` job. For 6.0, the `JFTPSTRT.ARPA.SYS` job is no longer used.

Instead the job, `JINETD.NET.SYS` is streamed to initiate the FTP server. To properly configure a system for FTP, please read the "Starting FTP" chapter of the *Installing and Managing HP ARPA File Transfer Protocol Network Manager's Guide.*

Anonymous FTP Functionality

FTP/iX anonymous logon allows a restricted access to the MPE/iX file system via the FTP/iX server, without the need for a logon account. The remote user can access the FTP/iX anonymous environment by specifying either one of the accepted user names 'anonymous' or 'ftp' (these login id's are currently defacto standards).

The system administrator can decide whether or not to have anonymous FTP for his MPE FTP server.

The anonymous ftp user can access files that are rooted at /FTPGUEST/PUB. While any attempt to traverse the MPE/iX directory structures beyond /FTPGUEST/PUB is denied, the FTP user can access, create, or delete files and directories below the defined root. This ensures that the security of the HP3000 system is intact.

Except for "site stream" command all the other FTP/iX commands will be executed, but under anonymous restrictions based at the defined absolute root path of /FTPGUEST/PUB.

Here is a sample Anonymous login:

```
ftp> open pilot5
Connected to pilot5
220 HP ARPA FTP Server [A00080ZZ] (C) Hewlett-Packard Co. 1990
Name (pilot5:manager): anonymous
331-Guest login ok, send ident as password.
Password:
230 User logged on
Remote system type is MPE/iX.
ftp>pwd
257-"/" is the current directory.
257 "Anonymous FTP" is the current session.
ftp>
```

Support for Non-Standard MPE/iX File Types

The following types of MPE/iX files can be transferred to a remote host using FTP.

- Standard MPE files with fixed, variable or bytestream format.

- Files with either binary or ASCII data types.

In addition, the latest enhancements enable the transfer of the following types of non-standard files between two HP3000 machines.

- Privileged files (files with PRIV file code)

- Circular files

- Message files

- RIO files

- CM KSAM files

- NM KSAM files

- HPSPOOL files

The type of the file is transparent to the user. The user need not specify the file type being transferring.

Of these, SPOOL files are transferred as fixed ascii files. The above non-standard file transfers are not possible between an MPE and an non-MPE system. However, SPOOL files can be transferred as fixed ascii files from an MPE to a non-MPE system.

The client (FTP) and server (FTPSRVR) must be compatible to take advantage of this feature. The above file types can be transferred between like FTP/iX systems without the need for buildparms on the command line. Buildparms included on the command line are ignored if the file is a non-standard file type. The use of buildparms is accepted only for standard file types (file code of zero).

Non-standard file type transfers to incompatible FTP/iX versions result in a data transfer failure message. You cannot append to non-standard MPE/iX file types with the use of this new feature. Non-standard file transfers to non-MPE/iX systems also result in a data transfer failure. Non-compatibility of client and server file transfer requests result in appropriate error messages.

Example

```
FILENAME   CODE   ------------LOGICAL RECORD-----------   ----SPACE----

                  SIZE   TYP         EOF       LIMIT R/B   SECTORS #X MX

DB000      PRIV   128W   FB           6           6   1        16  1  1
```

```
ftp> get db000
200 PORT command okay
150 File: db000 opened; data connection will be opened
2816 bytes received in 0.02 seconds (130.95 Kbytes/sec)
ftp>
```

FILENAME	CODE	-----------LOGICAL RECORD-----------			----SPACE----			
		SIZE	TYP	EOF	LIMIT R/B	SECTORS	#X	MX
BIGRIO1		128B	FAR	8193	8192 3	4128	1	8

```
ftp> get bigrio1
200 PORT command okay
150 File: bigrio1 opened; data connection will be opened
1054934 bytes received in 3.69 seconds (279.27 Kbytes/sec)
ftp>
```

The Configurable Server Time-Out

Current patch releases of FTP/iX 5.0 and 5.5 have a hard-coded infinite time-out value applied to the data connection timer of the FTP/iX server. The FTP/iX client, which shares this data connection, has a configurable time-out which is user-controlled through the TIMEOUT command.

The latest enhancements provide a configurable server time-out. This allows the client side to control the time-out value that the server applies to its data connection timer. To set the time-out value, execute the SITE TIMEOUT command with a value from 0 (infinite) to 3000 seconds.

Syntax

```
SITE TIMEOUT [num-secs]
```

The time-out value indicates how long the FTP server should wait for a message from the remote FTP client before giving up. The allowable range is 0 to 3000. A value from 1 to 3000 indicates a time-out value in seconds. A value of 0 means no time-out (that is, wait forever). If num-secs is not specified, the current remote time-out value is displayed. Otherwise, this command sets the connection time-out to num-secs seconds.

Example

Command for the server to time out after 5 minutes (300 seconds).

```
ftp>site timeout 300
200 TIMEOUT command ok.
ftp>
```

Hash Command

While transferring files, if the user desires to receive progress status during file transmission, the hash command provides this capability. This is a user command at the client end, which allows the user to toggle state of the hash command. If the toggle state is 'on' a hash symbol (#) is displayed every 1024 characters transferred. The default is set to 'off'.

Example

```
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> get TESTFILE
200 PORT command successful.
150 Opening BINARY mode data connection for TESTFILE (145915 bytes)

################################################################
################################################################
########
226 Transfer complete
145915 bytes received in 0.32 seconds (438.45 Kbytes/sec)
```

REPLACE Option

This FTP/iX client controlled option allows the user to override the default file placement strategy of FTP/iX (unconditional replacement of the target file).

Executing the client REPLACE command will toggle the current state (on or off). If the REPLACE option state, displayed by the REPLACE command is on, FTP/iX will unconditionally replace the specified target file. If the transfer state is set to off and a target file with the specified name exists on the target system, the FTP/iX client will prompt the user with a question (asking if it is okay to continue with the requested transfer). Otherwise, if the target file doesn't exist, FTP/iX will create the target file and continue with the transfer.

The user also has control over the REPLACE option by externally setting the environment variable FTPREPLACE to the desired state prior to entering the FTP/iX subsystem TRUE - on; FALSE - off). The default state for this option is TRUE (on).

Example

Using the REPLACE Option with the GET Command:

```
ftp>get abc def
200 PORT command ok.
150 File:  DEF;REC=-80,1,F,ASCII;DISC=204800,8 opened; data connection
will be opened
226 Transfer complete.
ftp>REPLACE
REPLACE option set to off (FTPINFO 99).
ftp>get abc def
File def exists! Do you want to replace it?>y
200 PORT command ok.
150 File:  DEF;REC=-80,1,F,ASCII;DISC=204800,8 opened; data connect
opened
ftp>get abc def
File def exists! Do you want to replace it?>n
New target file name: wxy
150 File:  WXY;REC=-80,1,F,ASCII;DISC=204800,8 opened; data connect
opened
226 Transfer complete.
```

MOVE Option

This FTP/iX user controlled option allows the user to selectively specify on the GET, PUT, MGET, or MPUT commands whether to delete the source file after confirmation of a successful file transfer to the target

system. MPE/iX file security and access rights may prevent successful removal of the original source file despite the use of the MOVE command.

Syntax

```
{ get }

{ put }

{     } source file [target file][;buildparms] [;move]

{ mget}

{ mput}

 Parameters :  none

 Operation : This option requests that the specified source
             file be physically moved from the source
             location to the specified target location.
```

Example

Example Use of The MOVE Option with the MGET Command

```
ftp>ls *
200 PORT command ok.
150 File: LISTFILE @,6 opened; data connection will be opened
abc
def
ghi
klm
226 Transfer complete.
20 bytes received in 0.08 seconds (0.26 Kbytes/sec)
ftp>mget *;move
200 PORT command ok.
150 File: LISTFILE @,6 opened; data connection will be opened
226 Transfer complete.
20 bytes received in 0.06 seconds (0.31 Kbytes/sec)
mget abc? y
200 PORT command ok.
150 File: ABC opened; data connection will be opened
226 Transfer complete.
100 bytes received in 0.03 seconds (3.49 Kbytes/sec)
mget def? y
200 PORT command ok.
150 File: DEF opened; data connection will be opened
226 Transfer complete.
100 bytes received in 0.03 seconds (3.05 Kbytes/sec)
mget ghi? y
200 PORT command ok.
150 File: GHI opened; data connection will be opened
226 Transfer complete.
100 bytes received in 0.04 seconds (2.57 Kbytes/sec)
mget klm? y
200 PORT command ok.
150 File: KLM opened; data connection will be opened
226 Transfer complete.
100 bytes received in 0.03 seconds (3.37 Kbytes/sec)
ftp>ls *
200 PORT command ok.
150 File: LISTFILE @,6 opened; data connection will be opened
226 Transfer complete.
```

The Implementation of CI Environment Variables for Additional FTP/iX Status Information

The following status information will be provided through these CI environment variables:

FTPXFERFILES number of files successfully transferred during the last FTP/iX execution.

FTPREQFILES total number of files requested for transfer during the last FTP/iX execution.

FTPREPLACE reflects the state of the file placement policy of FTP/iX: either unconditionally (TRUE) or conditionally (FALSE) replace the target file(s) specified in the user commands GET, PUT, MPUT, and MGET (see the REPLACE option for additional details).

The following environment variables are currently supplied with FTP/iX:

FTPLASTERR identifies the last FTP error encountered.

FTPLASTREPLY identifies the last FTP protocol reply message string.

The addition of the aforementioned environment variables will provide the interactive and batch FTP/iX environments with additional status information for control of processes outside the FTP/iX subsystem.

Example Use of Environment Variables

```
:showvar FTP@
FTPLASTERR = 0
FTPLASTREPLY = 221 Server is closing command connection
FTPXFERFILES = 0
FTPREQFILES = 0
FTPREPLACE = TRUE
            .
            .
            .
ftp>mget *
200 PORT command ok.
150 File: LISTFILE @,6 opened; data connection will be opened
226 Transfer complete.
20 bytes received in 0.07 seconds (0.29 Kbytes/sec)
mget abc? y
200 PORT command ok.
150 File: ABC opened; data connection will be opened
226 Transfer complete.
100 bytes received in 0.03 seconds (3.49 Kbytes/sec)
mget def? y
200 PORT command ok.
150 File: DEF opened; data connection will be opened
226 Transfer complete.
100 bytes received in 0.04 seconds (2.57 Kbytes/sec)
mget ghi? n
mget klm? n
ftp>quit
:
:showvar FTP@
FTPLASTERR = 0
FTPLASTREPLY = 221 Server is closing command connection
FTPXFERFILES = 2
```

```
FTPREQFILES = 4
FTPREPLACE = TRUE

:if FTPREQFILES <> 4 and FTPXFERFILES = 2 then
```

DNS BIND/iX and Syslog/iX

by Wendy Cheng
Commercial Systems Division

DNS BIND, which stands for Berkeley Internet Name BIND/iX Domain, is an implementation of the Domain Name System (DNS) and is the most commonly used of the Domain Name System (DNS). Now, the complete implementation of DNS BIND/iX runs on MPE/iX shell operation system on MPE/iX release 6.0. Initially, DNS BIND/iX was written for UNIX. Now, the latest version of DNS BIND/iX 8.1.1 runs on MPE/iX systems.

Overview of DNS BIND/iX

DNS BIND/iX is a domain name system which consists of a client-server mechanism. The name-servers comprise the server half of the DNS's client-server mechanism. The name-server is the NM program `NAMED` which maintains information about some part of the DNS called a zone and has capabilities to retrieve information regarding other zones. The clients are resolver routines provided as NMRL libraries. The resolvers are clients that query the name servers, interpret the responses and send the answers to the requester.

DNS BIND/iX makes your domain names visible to the internet as well as handling client requests to resolve domain names within your domain and external domains. Prior to DNS BIND/iX, the HP 3000 users had to rely on other machines or other operating system to host their organization DNS information. Now, you can host it on HP 3000 systems.

The latest version of DNS BIND/iX 8.1.1 is available for MPE/iX release 6.0 with the following features:

- DSN Dynamic Updates

- DNS Change Notification

- Completely new configuration syntax

- More efficient zone transfers

The BIND/iX contains the following major utilities and administration tools:

`nslookup`	The nslookup utility can be used by users to query the name servers interactively
`DIG`	Domain Information Groper.
`named-xfer`	The named-xfer utility is used to transfer the zone information and called by name-server internally.

dnsquery   The dnsquery tool can be used to provide you all the
                DNS detail information.

Overview of Syslog/iX

Syslog is the standard event logging system for Syslog/iX UNIX. Now
the Syslog/iX can run on the MPE/iX shell operating system on MPE/iX
release 6.0. With the features of Syslog/iX available on MPE/iX
systems, the event messages can be logged to files, terminal devices, or
even forward to other syslog systems. Syslog/iX can accept data from
the local system via an AF_UNIX socket or from any system on the
network via an AF_INET UDP socket on port 514. DNS BIND/iX uses
Syslog/iX as the event logging subsystem.

Types of Names Servers

DNS BIND/iX provides two types of name-servers: primary masters
and secondary masters. A primary master name server gets the data for
the zones it is authorized for from files on the host it runs on. A
secondary master name server gets its zone data from another name
server authorized for the zone. When a secondary name-server starts
up, it contacts the name-server it updates from and, if necessary, pulls
the zone data over. This is referred to as a zone transfer. With DNS
BIND/iX available on MPE/iX systems, you can create the data for your
zone and set up a primary master name-server, then you can set up
secondary master name-servers that load their data from primary
servers. Once they are set up, the secondary servers will periodically
query the primary servers to keep the zone data up to date.

Major Files Used for DNS BIND/iX

The following lists the major files required for DNS BIND/iX:

NAMED.CONF   The NAMED.CONF file is a configuration file of the DNS
                BIND/iX, which contains runtime configuration for the
                name-server called NAMED. This file resides in
                /BIND/PUB/etc directory.

Data Files   The files from which primary name-servers load their
                zone data are called data files or zone files. They are
                also referred to as db files or database files. The data
                files contain resource records that describe the zone.
                The resource records describe all the hosts in the zone.

There are two types of the data files. The files mapping addresses to
hostnames are called db.ADDRdb or zone.ADDR, where ADDR is the
network address. The files mapping hostnames to addresses called
db.DOMAIN or zone.DOMAIN.

Loopback Address

Files A name-server needs one additional `db.ADDR` or `zone.ADDR` file to cover the loopback network. The loopback address is a special address that hosts use to direct traffic to themselves. This network is always 127.0.0. and the host number is always 127.0.0.1. Therefore, the name of this file is zone.127.0.0.

Root Cache Data

File Besides your local data files for your zone information, the name servers also needs to know where the name servers for the root domain are. This file is needed to initialize cache of Internet Domain Name Servers. This file is also called hint file.

Configure Masters

The DNS BIND/iX allows users to configure both master zone or slave zone in your configuration file. Here are the sample configuration units for a master zone or slave zone shown below:

Sample Configuration for a Master Zone

```
zone "43.10.15.IN-Addr.ARPA  {
      type master
      file "db.15.10.43"
}
```

The file db.15.10.43 will have entries like:

```
IN     SOA     bindserver.india.hp.com
                104             ;  Serial
                36000           ;  Refresh every 10 hours
                3600            ;  Retry every hour
                360000          ;  Expire after 100 hours
                36000           ;  Minimum ttl is 10 hours

IN     SOA     bindserver.india.hp.com

1         IN    PTR     m1.india.hp.com.
2         IN    PTR     m2.india.hp.com.
3         IN    PTR     m3.india.hp.com.
```

There are several key parameters defined in the zone file record:

Serial The "serial" field is one of the important field defined in the zone file. Every time you make a change to a database file, you must increment its serial number. Only by doing this will secondary servers know they need to reach into your system and pull out new name server data, a procedure is known as a "zone transfer."

Refresh	The refresh field specifies the time interval that must elapse between each poll of the primary by the secondary name server (here 36,000 seconds or 10 hours). If the "serial number" has been updated on the primary server, the secondary name-server assumes its data is stale and requests updated information as a "zone transfer."
Retry	If the secondary fails to reach the primary name-server after the refresh period, it starts trying to connect every retry seconds, here 3600 seconds or 1 hour.
Retry Expire	If the secondary fails to contact the primary name-server for expire seconds, the secondary expires its data. Expiring the data means the secondary stops giving out answers about the data because the data are too old to be useful.
Minimum	The minimum time-to-live value, which specifies how long other servers should cache data from the name-server, here 36,000 seconds or 10 hours.

Sample Configuration Unit for a Slave Zone

```
zone "41.10.15.IN-ADDR.ARPA" {
      type slave;
      file "db.15.10.41";
      masters {
                15.70.188.45;
      };
}
```

The IP address of the server that is primary for that domain is specified in the masters { } section of the configuration unit. There could be more than one master for a given zone.

When the name-server comes up, looking at this configuration, it makes a connection with the name-server running on 15.70.188.45 and does a zone transfer, if it is required. It makes a local copy of this file too.

Configuration Migration

The configuration file in BIND/iX 8.1.1 version is called "`named.conf`" which has a completely new syntax. The configuration file in BIND 4.x version was called "`named.boot`."

The migration utility "`named-bootconf.pl`" is available with DNS BIND/iX and can be used to convert 4.x of named.boot file to 8.1.1 version of named.conf file. The utility "`named-bootconf.pl`" resides in /BIND/PUB/bin directory.

How to Run Syslog/iX

The following describes the important steps to start Syslog/iX:

1. Log on to the system as `MGR.SYSLOG`.

2. Create, examine and adjust the Syslog configuration file `syslog.conf`. The syslog.conf file resides in /SYSLOG/PUB directory.

3. You stream the following job to start Syslog/iX:

   `STREAM JSYSLOGD.PUB.SYSLOG`

4. You can stop Syslog/iX by issuing the command : `ABORTJOB`

How to Run DNS BIND/iX

The following describes the major steps to start DNS BIND/iX:

1. Log on to system as `MGR.BIND`.

2. Create, examine and customize the configuration file `named.conf` for your own system environment. The named.conf file resides in /BIND/PUB/etc directory.

3. You need to stream the following job to start DNS BIND/iX:

   `STREAM JNAMED.PUB.BIND`

4. Add your server's IP address as the first name-server entry in /etc/resolv.conf for all MPE and HPUX hosts that you wish to use this server for resolution queries. On MPE hosts, make sure that there is a symbolic link to make `RESLVCNF.NET.SYS` link to `/etc/resolv.cnf`.

5. You can stop DNS BIND/iX by issuing the command :`ABORTJOB`.

Sample Configuration File for SysDiag/iX

When you want to use Syslog/iX, you should examine File for SysDiag/iX and adjust the syslog configuration file syslog.conf.

The syslog.conf file resides in /SYSLOG/PUB directory.

The following is the sample configuration file for Syslog/iX that you can find in the system:

```
##
## :TELL @.@
## *.emerg
##
## Write to the :console
##
*.alert
##
## :TELL @.SYSLOG
##
*.crit @.SYSLOG
##
```

```
## :TELL MANAGER.SYS
## *.err MANAGER.SYS
##
## Forward to syslogd on another host via UDP
##
*.warning @some.host.running.syslogd
##
*.info
##
## Write to a file
##
*.debug /tmp/syslog.log
```

The error messages coming from a program are classified into critical informative and alert types of messages. The syslog configuration file tells the syslog daemon how to post these messages. They could be sent to the console or to a log file, or to another machine.

Test Your Name Server

The nslookup utility can be used interactively, using nslookup much like other programs, such as ftp. That is, if you invoke this program without command-line arguments, it displays a prompt and waits for your command:

> server mpe3000

```
Default Name Server: mpe3000.cup.hp.com Address: 15.13.199.80
```

By default, nslookup performs queries based on host names you submit; just enter a host name after the prompt:

```
> romeo
Server: mpe3000.cup.hp.com
Address: 15.13.199.80

Name: romeo.cup.hp.com
Address: 15.13.194.242

> 15.12.194.242
Server: mpe3000.cup.hp.com
Address: 15.13.199.80

Name: romeo.cup.hp.com
Address: 15.12.194.242
```

You can check the resource records information about name server:

```
> set type=ns
> mpeworld
Name Server: mpeworld.cup.hp.com
Address: 15.13.199.80

origin = dns.cup.hp.com
mail addr = dns-admin.dns.cup.hp.com
serial = 96092255
refresh = 10800 (3 hours)
retry = 3600 (1 hour) expire = 604800 (7 days)
minimum ttl = 86400 (1 day)
```

Debugging Levels for Troubleshooting

Debugging levels 1 through 14 can be specified when Troubleshooting the DNS BIND/iX server is started. The level of detail can be controlled in the JNAMED.PUB.BIND file. The higher the debugging value, the more detail will be logged about the activities of the server.

The following describes how you can set up the debugging level in the
`JNAMED.PUB.BIND`:

```
JNAMED.PUB.BIND:
 !job jnamed,mgr.bind;outclass=,2
 !run named;info="-f /BIND/PUB/etc/named.conf"
 !eo
```

The parameter "-d <level>" can be specified in the info string of the run
command.

Example

```
!run named;info=" -d 4 -f /BIND/PUB/etc/named.conf"
It will set the debugging level 4.
```

The logging and trace messages can be seen in the spool files.

Example

```
:showjob

JOBNUM    STATE    IPRI    JIN    JLIST    JOB NAME
#J2       EXEC             10S    LP       FTPMON,FTP.SYS
#j3        EXEC            10S      LP       JINETD,MANAGER.SYS

#17       EXEC             10S    LP       JNAMED,MGR.BIND
#S44      EXEC             20     20       MGR.BIND

:showout job=#j17

:print Oxxx.out.hpspool
```

Documentation

Documentation for the DNS BIND/iX and Syslog/iX is available in the
Configuring and Managing MPE/iX Internet Services that is shipped
with the MPE/iX FOS.

More DNS BIND/iX Resources

Additional information is available in this book and on these websites:

- *DNS & BIND* is a book which was written by Paul Albitz and
  Cricket Lui and published by O'Reilly & Associates, Inc.

- *http://www.isc.org*

- *http://www.academ.com/nanog/oct1997/bind8/index.html*

11   Product Release History

This chapter contains tables that provide information on the currently
supported Commercial Systems MPE/iX releases and products, and the
systems supported for the 6.0 release.

Product Changes by Releases

The following table provides information on the currently supported Commercial Systems MPE/iX releases and products. Included are the MPE/iX release or SUBSYS VUF and a list of products introduced. It also provides information on significant changes made to a release.

Table 11-1 **MPE/iX Product Releases**

Release	SUBSYS	Date Code	Product(s) Introduced/Added
C.50.00		R3504	MPE/iX Release 5.0 (Core Software Release) Workload Manager (B3879AA) Shared Globals ARPA Bundling
	C.50.02	R3513	MPE/iX-Express 2 based on General Release 5.0 MOVER Utility Information Access SQL/iX LASERRX/MPE ALLBASE/4GL Developer and Runtime
	C.50.03	R3538	MPE/iX-Express 3 based on General Release 5.0 Information Access Server/iX Symbolic Debug/iX ALLBASE/SQL G.1 IMAGE/SQL G.1 New Systems: HP 3000 Series 969KS/x00 (x = 1- 4) HP 3000 Series 996/80, 996/x00 (x = 1- 8) New SCSI Disks: ST31200N (1 GB Single-Ended) ST31230N (1 GB Single-Ended, Low Profile) ST32550N (2 GB Single-Ended) ST15150N (4 GB Single-Ended) ST32550W (2 GB Fast/Wide) ST15150W (4 GB Fast/Wide)

Release	SUBSYS	Date Code	Product(s) Introduced/Added

C.55.00		R3628	MPE/iX Release 5.5 (Non-Platform Release)
			
			HP Loader Dependent Libraries
			Subsystem Dump Facility
			HP Stage/iX
			HP Patch/iX
			TurboSTORE/iX 7x24 True-Online Backup
			HP Optical Disk Libraries:
			   C1150B - 40GB
			   C1160B - 80GB
			   C1170B - 100GB
			TCP/IP Network Printer Support
			Telnet/iX Server
			DTS/TIO Dynamic Configuration
			Online System Device Configuration

	C.55.01	R3638	PowerPatch 1 based on Release 5.5 ALLBASE/SQL CAST function HP Telnet/iX—full functionality CCSY Access Server 979/x00KS support High Availability Disk Array support
	C.55.02	R3715	Express 2 based on Release 5.5 CI enhancements VPlus enhancements DTS/TIO new functionality

	C.55.03	R3732	Express 3 based on Release 5.5
			
			COBOL II/iX enhancements
			100VG-AnyLAN Network Adapter introduced
			100Base-T Network Adapter introduced
			ALLBASE/SQL release G2
			ODBCLink/SE ODBC driver introduced
			TurboIMAGE/XL B-Tree and scalability enhancements
			IMAGE/SQL B-Tree enhancement
			9 GB Disk Drive
			NMMGR Node Maintenance Manager enhancements
			New SCSI Disks:
			   ST32272N, ST32272WC (2 GB)
			   ST34572N, ST34572WC (4 GB)
			   XP34361W (4 GB)

Release	SUBSYS	Date Code	Product(s) Introduced/Added
		R3748	Express 4 based on Release 5.5 Year 2000 enhancements MPE/iX new date intrinsics 997/100-500 support Enhancements for: Transact V and Transact/iX Dictionary/3000 COBOL II/iX Inform/V HP ALLBASE/BRW QUERY/iX
		R3813	PowerPatch 5 based on Release 5.5 989/100, 989/200, 989/400, 989/600 support 997/600, 997/800 support A3699A - HP-IB I/O Bus Converter New SCSI Disks: A4909A - 18GB, WD A5239A - 18GB, WD Enhancements for Image/SQL and TurboIMAGE/XL: Dynamic data set expansion for master sets Performance enhancement for TPI Image Row-level locking 80 GB Image data sets ODBCLink/SE E.56.12
C.60.00		R3821	MPE/iX Release 6.0 (Platform Release) User-defined job Queue FTP enhancements Java for MPE/iX Samba/iX Performance enhancements System limit enhancements DNS BIND/iX

Supported Releases

The naming conventions for the different types of releases have been changed slightly to clarify the type of release being discussed. The terms used to describe or refer to the releases are:

Mainline Release A mainline release involves the recompilization and reintegration of all software release components (FOS, SLT, and SUBSYS tapes). The release number is changed (for example, 5.0 or 5.5) and the update "UU" field of the V.UU.FF is changed. There are two types of mainline releases: *Platform* and *Non-Platform*.

Platform Release A platform release (previously also known as a "core" release) is a subset of a mainline release. Typically, the release number ends with a "0" such as 5.0. Platform releases are *automatically distributed* to all customers with support contracts.

Non-Platform Release A non-platform release (previously known as a "major" release) is a subset of the mainline release. The release number typically ends with a "5" such as 5.5. Non-platform releases must be *explicitly ordered* by customers.

Both platform and non-platform releases can be referred to as *mainline releases* when not discussing distribution or extended support life.

The following matrix provides information on the supported Commercial Systems MPE/iX mainline releases. It lists the currently supported releases and the SPUs they are supported on. The matrix also provides all known factory support termination dates. When a mainline release becomes unsupported, the factory will not provide any support services for that release. Online calls are not accepted and patches are not created; customers are advised to roll to a supported release.

- In general, HP will support the two most recent mainline releases.

- In order to facilitate customers moving to new releases, we will support the oldest release for at least six months after the new release ships. In other words, Release "N-2" will still be supported for at least six months after Release "N" ships.

- The release of an Express Release or a Powerpatch on any particular release does not extend its support life.

Given the rate with which we ship releases, this new strategy implies that any particular release will be supported for at least three years, and in most cases, most releases will be supported for a significantly longer period of time.

This new strategy is effective now. This implies that Release 5.0, which first shipped February 17, 1995, will become obsolete as early as six months after Release 6.0 ships.

Table 11-2   **Supported System Release Matrix**

Supported Releases	Supported Systems	Support Termination Date
@Release 5.0 (C.50.xx)	920,922,925,930,932,935,948,949,950,955,960, 980-100,980-200,980-300,980-400,9x7,9x7LX, 9x7RX,9x110,992/300,992/400,992/100DX, 992/200DX,992/300DX,992/400DX,991DX, 995/100DX,995/200DX,995/300DX,995/400DX, 995/500DX,995/600DX,995/700DX,995/800DX, 991CX,995/100CX,995/200CX,995/300CX, 995/400CX,995/500CX,995/600CX,995/700CX, 995/800CX	June 1, 1999
Release 5.5 (C.55.xx)	920,922,925,930,932,935,948,949,950,955,960,9x7, 9x7LX,9x7RX,9x7SX,9x8LX,9x8RX,939KS,939KS/0 20, 959KS/100,959KS/200,959KS/300,959KS/400, 969KS/100,969KS/200,969KS/300,969KS/400, 969KS/120,969KS/220,969KS/320,969KS/420, 979KS/100,979KS/200,979KS/300,979KS/400, 980-100,980-200,980-300,980-400,987/150RX, 987/150SX,987/200RX,987/200SX, 989/100,989/200,989/400,989/600,990,990DX, 992/100,992/200,992/300,992/400,992/100DX, 992/200DX,992/300DX,992/400DX,991DX, 995/100DX,995/200DX,995/300DX,995/400DX, 995/500DX,995/600DX,995/700DX,995/800DX, 991CX, 995/100CX,995/200CX,995/300CX, 995/400CX, 995/500CX,995/600CX,995/700CX, 995/800CX 996/80,996/100,996/200,996/300, 996/400,996/500,996/600,996/700,996/800, 997/100,997/200,997/300,997/400,997/500, 997/600,997/800	6 months following next Mainline release after 6.0; approximately the end of year 2000

Supported Releases	Supported Systems	Support Termination Date
Release 6.0 (C.60.xx)	920,922,925,930,932,935,948,949,950,955,960,9x7, 9x7LX,9x7RX,9x7SX,9x8LX,9x8RX,929KS/020, 939KS,939KS/020,959KS/100,959KS/200, 959KS/300,959KS/400,969KS/100,969KS/200, 969KS/300,969KS/400,969KS/120,969KS/220, 969KS/320,969KS/420,979KS/100,979KS/200, 979KS/300,979KS/400,980-100,980-200,980-300, 980-400,987/150RX,987/150SX,987/200RX, 987/200SX,989KS/100,989KS/200,989KS/400, 989KS/600,990,990DX,992/100,992/200,992/300, 992/400,992/100DX,992/200DX,992/300DX, 992/400DX,991DX,995/100DX,995/200DX, 995/300DX,995/400DX,995/500DX,995/600DX, 995/700DX,995/800DX,991CX,995/100CX, 995/200CX,995/300CX,995/400CX,995/500CX, 995/600CX,995/700CX,995/800CX996/80,996/100, 996/200,996/300,996/400,996/500,996/600,996/700, 996/800, 997/100,997/200,997/300,997/400,997/500, 997/600,997/800	6 months following second mainline release after 6.0

12   Catalog of User Documentation

This chapter provides listings of customer manuals for the HP 3000 computer system. The listings are divided into two sections:

- "MPE/iX 6.0 New or Updated Manuals," which lists all manuals that have been introduced or changed since the MPE/iX 5.5 Release.

- "Manual Collections," which lists manuals by collections in alphabetical order. For detailed information on a particular manual or manual collection, refer to the *MPE/iX Documentation Guide*.

If your contract includes Material-Based Services, you will receive both software and manual revisions. For additional copies of new or revised manuals, you can order Manual Update Services (MUS).

Many of the learning products listed in this chapter can be individually ordered by calling HP Parts Direct Ordering at 800-227-8164. Specify the part number of the manual you are interested in when ordering.

MPE/iX 6.0 New or Updated Manuals

This section lists customer manuals introduced or updated from
MPE/iX-Express 2 based on General Release 5.5 through MPE/iX 6.0
Release.

Table 12-1   **MPE/iX 6.0 New or Updated Manuals**

Manual Title	Part No.	Latest Edition
ALLBASE/SQL Database Administration Guide	36216-90214	8/97
ALLBASE/SQL Reference Manual	36216-90215	10/98
ALLBASE/SQL Message Manual	36216-90213	8/97
Asynchronous Serial Communications Programmer's Reference Manual	32022-90052	10/98
Communicator 3000 MPE/iX Release 6.0	30216-90269	10/98
Configuring and Managing MPE/iX Internet Services	32650-90859	10/98
Configuring Systems for Terminal, Printer, and Other Serial Devices	32022-90051	10/98
HP 3000 MPE/iX System Software Maintenance Manual	30216-90272	10/98
HP 3000/iX Network Planning and Configuration Guide	36922-90037	10/98
HP ARPA File Transfer Protocol User's Guide	36957-90158	10/98
HP IMAGE/SQL Administration Guide	36385-90012	8/97
HP-PB 100VG-AnyLAN Network Adapter Installation and Service Guide	B5425-90001	8/97
HP-PB 100Base-T Network Adapter Installation and Service Guide	B5427-90001	8/97
HP Telnet/iX User's Guide	36957-90156	10/98
Installing and Managing HPARPA File Transfer Protocol Manager's Guide	36957-90157	10/98
Installation/Update Procedures for SHAREPLEX	B3933-90005	6/97
Legato NetWorker ClientPak for MPE/iX Installation Guide	B5475-90001	12/97

Manual Title	Part No.	Latest Edition
Legato NetWorker ClientPak for MPE/iX User's Guide	B5475-90002	12/97
Legato NetWorker ClientPak for MPE/iX Release 5.0 Supplement	B5475-90005	2/98
MPE/iX Commands Reference Manual	32650-90864	10/98
MPE/iX Documentation Guide	32650-90866	10/98
MPE/iX Error Messages Addendum	32650-90874	10/98
NBSpool for MPE Reference Guide	B3933-90002	6/97
NetBase for MPE Reference Guide	B3933-90001	6/97
NetBase SQL Shadowing Handbook	B3482-90006	6/97
NS 3000/iX Error Messages Reference Manual	36923-90041	10/98
NS 3000/iX NMMGR Screens Reference Manual	36922-90038	10/98
NS 3000/iX Operations and Maintenance Manual	36922-90039	10/98
ODBCLink/SE Reference Manual	36217-90407	7/98
TurboIMAGE/XL Database Management System Reference Manual	30391-90010	8/97
Using the Node Management Service (NMS) Utilities	32022-90053	10/98
Vista Plus for MPE User's Guide	B3933-90003	6/97
Vista Plus for MPE Administrator's Guide	B3933-90004	6/97

Manual Collections

Table 12-2   **Manual Collections**

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
MPE/iX Operating Systems						
CI Programming Quick Reference Pocket Card	32650-90269	32650-90316	Dec-90	n	y	y
Communicator 3000 (6.0)	30216-90269	30216-90269	Oct-98	n	y	y
Communicator 3000 (PowerPatch 5 based on 5.5)	30216-90257	30216-90257	Jul-98	n	n	y
FCOPY Reference Manual	32212-90003	32212-90008	Jun-92	y	y	y
HP 3000 MPE/iX System Software Maintenance Manual	30126-90223	30216-90272	Oct-98	n	y	y
HP 3000 Series 9X8LX Computer Systems: Commands Reference	B3813-90011	B3813-90012	Apr-94	y	y	y
HP 3000 Series 9X8LX Computer Systems: Getting Started	B3813-90003	B3813-90014	Apr-94	y	y	y
HP 3000 Series 9X8LX Computer Systems: Task Reference Manual	B3813-90009	B3813-90010	Apr-94	y	y	y
HP 3000 Series 9X8LX Computer Systems: Understanding Your System	B3813-90001	B3813-90013	Apr-94	n	y	y
HP Easytime/XL Quick Reference Card	B1940-90001	B1940-90611	Jun-92	n	n	n
HP Easytime/XL User's Guide	B1940-90002	B1940-90602	Sep-91	n	n	n
MPE/iX Commands Reference Manual	32650-90003	32650-90864	Oct-98	n	y	y
MPE/iX Day to Day Tasks	A1707-90004	A1707-96034	Jun-92	n	y	y
MPE/iX Documentation Guide	32650-90144	32650-90866	Oct-98	y	y	y
MPE/iX Error Messages Addendum		32650-90874	Oct 98	n	y	y
* The documentation website address is: *http://www.docs.hp.com.*						

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
MPE/iX Error Messages Manual Vol 1	32650-90066	32650-90481	Apr-94	y	y	y
MPE/iX Error Messages Manual Vol 2	32650-90152	32650-90851	Jul-96	y	y	y
MPE/iX Error Messages Manual Vol 3	32650-90368	32650-90483	Apr-94	y	y	y
MPE/iX Glossary of Terms and Acronyms	32650-90146	32650-90391	Jun-92	y	y	y
MPE/iX Quick Reference Guide	32650-90032	32650-90873	Oct-98	n	y	y
MPE/iX System Utilities Reference Manual	32650-90081	32650-90490	Apr-94	y	y	y
SORT-MERGE/XL General User's Guide	32650-90082	32650-90499	Apr-94	y	y	y
SORT-MERGE/XL Programmer's Guide	32650-90080	32650-90080	Jan-91	y	y	y
Using HP 3000: Advanced Skills Tutorial	32650-90872	32650-90872	Oct-98	y	y	y
Using HP 3000: Fundamental Skills Tutorial	32650-90871	32650-90871	Oct-98	y	y	y
Development Tools and Distributed Computing						
Accessing Files Programmer's Guide	32650-90017	32650-90394	Jun-92	y	y	y
ALLBASE NET User's Guide	36216-90031	36216-90101	Apr-94	y	y	y
ALLBASE/SQL Advanced Application Programming Guide	36216-90100	36216-90099	Apr-94	y	y	y
ALLBASE/SQL C Application Programming Guide	36216-90023	36216-90080	Jun-92	y	y	y
ALLBASE/SQL COBOL Application Programming Guide	36216-90006	36216-90081	Jun-92	y	y	y
ALLBASE/SQL Database Administration Guide	36216-90005	36216-90214	Aug-97	y	y	y

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
ALLBASE/SQL FORTRAN Application Programming Guide	36216-90030	36216-90079	Jun-92	y	y	y
ALLBASE/SQL Message Manual	36216-90009	36216-90213	Aug-97	y	y	y
ALLBASE/SQL Pascal Application Programming Guide	36216-90007	36216-90082	Oct-92	y	y	y
ALLBASE/SQL Performance and Monitoring Guidelines	36216-90102	36216-90103	Apr-94	y	y	y
ALLBASE/SQL Reference Manual	36126-90001	36216-90215	Oct-98	n	y	y
Asynchronous Serial Communications Programmer's Reference Manual	32022-61001	32022-90052	Oct-98	n	y	y
BASIC Interpreter Reference Manual	30000-90026	30000-90026	Nov-78	n	n	n
BASIC/3000 Compiler Reference Manual	32103-90001	32103-90001	Sep-77	y	n	n
BASIC/V to HP Business BASIC/V Conversion Guide	32115-90004	32115-90004	Sep-87	y	n	n
Berkeley Sockets/iX Reference Guide	32650-90363	32650-90372	Apr-94	y	y	y
Command Interpreter Access and Variables Programmer's Guide	32650-90011	32650-90493	Apr-94	y	y	y
Compiler Library/XL Reference Manual	32650-60014	32650-90029	Oct-88	y	n	y
Data Dictionary Managing Information Network Primer	5958-8527	5958-8527	Nov-86	n	n	n
Data Entry and Forms Mgmt System VPLUS Reference Manual	32209-90001	32209-90001	Nov-87	y	n	y
Data Entry and Forms Mgmt System VPLUS Technical Addendum	32209-90021	32209-90021	Apr-90	y	n	y
Data Types Conversion Programmer's Guide	32650-60010	32650-90015	Oct-89	y	y	y

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
DBChange Plus Technical Addendum for MPE/iX Release 4.0	36386-90005	36386-90005	Jun-92	y	n	y
DBChange Plus User's Guide	36386-90001	36386-90001	Feb-91	y	n	y
DCE for the HP 3000	B3821-90001	B3821-90002	Oct-95	y	n	y
Dictionary/3000 Reference Manual	32244-90001	32244-90001	Dec-87	y	n	y
EDIT/3000 Reference Manual	03000-90012	32650-90385	Aug-80	y	y	y
Getting Started as an MPE/iX Programmer	32650-90008	32650-90421	Jun-92	y	y	y
Getting Started with HP IMAGE/SQL	36385-90008	36385-90011	Dec-94	y	y	y
Getting Started with HP Software Revision Controller (SRC)	30234-60002	30234-90002	Nov-88	n	n	n
Getting Started with TRANSACT	32247-60002	32247-90007	Jul-88	y	n	y
High Level Screen Management Intrinsics Library Reference Manual	32424-60001	32424-90002	Nov-87	y	n	y
HP 3000 Basic for Beginners	03000-90025	03000-90025	Nov-72	n	n	n
HP ALLBASE Query User Guide Kit	32426-64001	32426-64001	Oct-89	n	n	n
HP ALLBASE/4GL Developer Administration Manual	30601-90201	30601-90205	May-92	y	n	y
HP ALLBASE/4GL Developer Quick Reference Guide	30601-90210	30601-90211	May-92	y	n	y
HP ALLBASE/4GL Developer Reference Manual Vol 1	30601-90202	30601-90206	May-92	y	n	y
HP ALLBASE/4GL Developer Reference Manual Vol 2	30601-90204	30601-90208	May-92	y	n	y
HP ALLBASE/4GL Installation Manual	30601-64212	30601-64213	May-92	n	n	n

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
HP ALLBASE/4GL Self-Paced Training Guide	30601-90203	30601-90207	May-92	y	n	y
HP ALLBASE/4GL Software Update Notice B.06 Release	5961-7797	5063-3725	Feb-93	y	n	y
HP ALLBASE/BRW Reference Manual	35360-90051	35360-90052	Jan-92	y	n	y
HP ALLBASE/BRW SW Update Notice for MPE/iX (BRW A.01.50)	35360-90204	35360-90203	Feb-94	y	n	y
HP ALLBASE/BRW Tutorial	35360-90201	35360-90202	May-92	y	n	y
HP Browse/XL User's Guide	36384-90001	36384-90001	Oct-90	y	n	y
HP Business BASIC Programmer's Guide	32115-90003	32115-90003	Jul-87	y	n	y
HP Business BASIC Quick Reference Guide	32115-90002	32115-90002	Jul-85	n	n	n
HP Business BASIC Reference Manual	32115-90001	32115-90001	Jul-87	y	n	y
HP Business BASIC/XL Migration Guide	32715-60002	32715-90003	Oct-89	y	n	y
HP Business BASIC/XL Reference Manual	32715-60001	32715-90001	Oct-89	y	n	y
HP C Programmer's Guide	92434-90002	92434-90009	Jul-96	y	n	y
HP C/iX Library Reference Manual	30026-90001	30026-90004	Oct-92	y	n	y
HP C/iX Reference Manual	31506-90005	31506-90011	Jun-92	y	n	y
HP COBOL II/V Reference Manual	32233-90001	32233-90001	May-89	y	n	y
HP COBOL II/XL Migration Guide	31502-60011	31500-90004	Oct-88	y	n	y
HP COBOL II/XL Programmer's Guide	31500-90002	31500-90014	Jul-91	y	n	y
HP COBOL II/XL Quick Reference Guide	31500-90003	31500-90015	Jul-91	y	n	y
HP COBOL II/XL Reference Manual	31500-90001	31500-90013	Jul-91	y	n	y

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
HP EDIT Quick Reference Guide	30316-90005	30316-90017	Dec-90	n	n	n
HP EDIT Reference Manual	30316-90001	30316-90016	Dec-90	y	n	y
HP FORTRAN 77/iX Migration Guide	31501-90004	31501-90023	Jun-92	y	n	y
HP FORTRAN 77/iX Programmer's Guide	31501-90011	31501-90022	Jun-92	y	n	y
HP FORTRAN 77/iX Reference	31501-90010	31501-90021	Jun-92	y	n	y
HP GlancePlus User's Manual (for MPE/iX Systems)	B1787-90001	B1787-90008	Apr-92	y	n	y
HP IMAGE/SQL Administration Guide	36385-90001	36385-90012	Aug-97	y	y	y
HP Link Editor/iX Reference Manual	32650-90030	32650-90309	Dec-90	y	n	y
HP Link Editor/iX Technical Addendum	32650-09476	32650-90845	Oct-95	y	n	y
HP Pascal/iX Migration Guide	31502-60011	31502-90004	Nov-87	y	n	y
HP Pascal/iX Programmer's Guide	31502-90002	31502-90023	Jun-92	y	n	y
HP Pascal/iX Reference Manual	31502-90001	31502-90022	Jun-92	y	n	y
HP Performance Collection Software User's Manual	50700-90022	50700-90038	Apr-92	n	n	n
HP RPG/iX Pocket Guide	30318-90002	30318-90002	Oct-89	n	n	n
HP RPG/iX Programmer's Guide	30318-60001	30318-90001	Jul-89	y	n	y
HP RPG/iX Reference Manual	30318-60002	30318-90011	Dec-93	y	n	y
HP RPG/iX Reference Manual Software Update Notice	30318-90016	30318-90017	Mar-95	y	n	y
HP RPG/iX Utilities Reference Manual	30318-60003	30318-90006	Oct-89	y	n	y
HP Search/XL User's Guide	36383-90001	36383-90001	Oct-90	y	n	y

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
HP Software Revision Controller (SRC) Implementation Guide	30234-60002	30234-90003	Nov-88	n	n	n
HP Software Revision Controller (SRC) Quick Reference Card	30234-60002	30234-90005	Nov-88	n	n	n
HP Software Revision Controller (SRC) User's Guide	30234-60002	30234-90001	Nov-88	n	n	n
HP Software Revision Controller/XL Product Information Update	30234-60002	30234-90006	Nov-86	n	n	n
HP Symbolic Debugger/iX User's Guide	31508-90003	31508-90014	Jun-92	y	n	y
HP System Dictionary/XL COBOL Definition Extractor	32257-90001	32257-90001	Dec-87	y	n	y
HP System Dictionary/XL General Reference Manual Vol 1	32256-90004	32256-90004	May-88	y	n	y
HP System Dictionary/XL General Reference Manual Vol 2	32256-90005	32256-90005	May-88	y	n	y
HP System Dictionary/XL Intrinsics Reference Manual	32256-90002	32256-90002	May-88	y	n	y
HP System Dictionary/XL SDMAIN Reference Manual	32256-90001	32256-90001	May-88	y	n	y
HP System Dictionary/XL Self-Paced Customer Training	32254-91001	32254-91001	Aug-87	n	n	n
HP System Dictionary/XL Utilities Reference Manual	32256-90003	32256-90003	May-88	y	n	y
HP Toolset/XL Reference Manual	36044-60001	36044-90001	Jan-84	y	n	y
HP TRANSACT Documentation Update Notice	32247-90028	32247-90028	Oct-96	y	n	y
HP TRANSACT Quick Reference Guide	32247-90020	32247-90027	Oct-96	y	n	y
HP TRANSACT Reference Manual	32247-60003	32247-90026	Apr-94	y	n	y

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
INFORM/V User's Guide	32246-60002	32246-60002	Mar-88	y	n	y
Introduction to MPE/XL for MPE V Programmers	30367-90005	30367-90005	Oct-89	y	n	y
Introduction to MPE/XL for MPE V System Administrators	30367-90003	30367-90017	Dec-90	y	n	y
IPC Communications Programmer's Guide	32650-90019	32650-90019	Nov-87	y	y	y
ISQL Reference Manual for ALLBASE/SQL & IMAGE/SQL	36216-90096	36216-90095	Apr-94	y	y	y
KSAM/3000 Reference Manual	30000-90079	32650-90386	Jun-92	y	y	y
Learning HP EDIT	30316-90002	30316-90015	Dec-90	y	n	y
Message Catalogs Programmer's Guide	32650-90021	32650-90021	Mar-90	y	y	y
Migration Process Guide	30367-90007	30367-90019	Jun-92	y	n	y
MPE Segmenter Reference Manual	30000-90011	30000-90011	Aug-96	y	n	y
MPE V to MPE XL: Getting Started Mentor's Guide	30367-90004	30367-90004	Oct-89	y	n	y
MPE V to MPE XL: Getting Started Self-Paced Training	30367-90002	30367-90002	Oct-89	y	n	y
MPE/iX AIF: OS Reference Manual	36374-90001	36374-90013	Dec-94	n	y	y
MPE/iX Developer's Kit Reference Manual Vol 1	36430-90001	36430-90007	Apr-94	y	y	y
MPE/iX Developer's Kit Reference Manual Vol 2	36430-90002	36430-90008	Apr-94	y	y	y
MPE/iX Intrinsics Reference Manual	32650-90028	32650-90821	Dec-94	y	y	y
MPE/iX Shell and Utilities Reference Manual Vol 1	36431-90001	36431-90007	Apr-94	n	n	.pdf
MPE/iX Shell and Utilities Reference Manual Vol 2	36431-90003	36431-90008	Apr-94	n	n	.pdf
MPE/iX Shell and Utilities User's Guide	36431-90002	36431-90006	Apr-94	n	n	.pdf

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
Native Language Programmer's Guide	32650-90022	32650-90207	Sep-91	n	y	y
ODBCLINK/SE Reference Manual	36217-90406	36217-90407	Jul-98	n	n	y
Process Management Programmer's Guide	32650-90023	32650-90023	Nov-87	y	y	y
QUERY/V Reference Manual	30000-90042	32650-90387	May-87	y	y	y
Resource Management Programmer's Guide	32650-90024	32650-90024	Nov-87	y	y	y
RPG/V Reference Manual	32104-90001	32104-90001	Sep-89	y	n	n
SPL to HP C/XL Migration Guide	30231-60001	30231-90001	Oct-89	y	n	y
Switch Programming Guide	32650-60030	32650-90014	Nov-87	y	n	y
System Debug Reference Manual	32650-90013	32650-90824	Apr-94	y	y	y
The POSIX.1 Standard: A Programmer's Guide ISBN-0-8053-9605-5	-	36430-90006	Apr-94	n	n	n
Trap Handling Programmer's Guide	32650-90026	32650-90026	Mar-90	y	n	y
TurboIMAGE/XL Database Management System Reference Manual	30391-90001	30391-90010	Aug-97	y	y	y
Up and Running with ALLBASE/SQL	36389-90011	36389-90016	Jun-92	y	y	y
User Logging Programmer's Guide	32650-90027	32650-90027	Jul-88	y	y	y
Using KSAM/XL	32650-90168	32650-90487	Mar-94	y	y	y
Using VPLUS/V: An Introduction to Forms Design	32209-90004	32209-90004	Aug-86	y	n	y
High Availability						
Auto/Restart/XL User's Guide	36375-90001	36375-90004	Oct-92	y	y	y
Installation/Upgrade Procedures for SHAREPLEX	B3933-90005	B3933-90005	Jun-97	n	n	.pdf

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
Legato NetWorker ClientPak for MPE/iX Installation Guide	B5475-90001	B5475-90001	Dec-97	n	n	.pdf
Legato Networker ClientPak for MPE/iX Release 5.0 Supplement	B5475-90005	B5475-90005	Feb-98	n	n	.pdf
Legato NetWorker ClientPak for MPE/iX Users Guide	B5475-90002	B5475-90002	Dec-97	n	n	.pdf
Mirrored Disk/iX User's Guide	30349-90003	30349-90005	Jun-92	y	y	y
NBSpool for MPE Reference Guide	B3933-90002	B3933-90002	Jun-97	n	n	.pdf
NBSpool Release Notes Version 9.7	B3933-90008	B3933-90008	Jun-97	n	n	.pdf
NetBase for MPE Reference Guide	B3933-90001	B3933-90001	Jun-97	n	n	.pdf
NetBase Release Notes Version 9.7	B3933-90007	B3933-90007	Jun-97	n	n	.pdf
NetBase SQL Shadowing Handbook	B3482-90006	B3482-90006	Jun-97	n	n	.pdf
VISTA Plus for MPE Administrators Guide	B3933-90004	B3933-90004	Jun-97	n	n	.pdf
VISTA Plus for MPE User's Guide	B3933-90003	B3933-90003	Jun-97	n	n	.pdf
VISTA Release Notes Version A.07	B3933-90006	B3933-90006	Jun-97	n	n	.pdf
System Hardware and Diagnostics						
CPU Upgrade Manual (9x9KS)	-	A2375-90012	Jan-98	n	n	.pdf
Expansion Cabinet Installation Guide (99X/Family)	-	A1809-90006	May-97	n	n	.pdf
Firmware Update Guide (99X/Family)	-	A1820-90002	May-97	n	n	.pdf
Firmware Update Quick Reference (99X/T-Class)	-	A1820-90003	May-97	n	n	.pdf

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
HP 3000 9X9KS Site Preparation and Requirements Guide	-	A2375-90073	Nov-96	n	n	.pdf
HP 3000 and HP 9000 PA-RISC Computer System Support Log	09740-90013	09740-96001	May-90	n	n	n
HP 3000 CS 99x/890/T500 Families Operator's Guide	A1809-90009	A1809-96019	Apr-97	n	n	n
HP 3000 Series 9X8lx/RX Computer Systems Inst. and Configuration Guide	A2051-90006	A2051-96006	Oct-93	n	n	n
HP 3000/9x9KS Installation Guide	-	-	-	n	n	.pdf
HP PA-RISC Computer Systems Integrated Cabinet Installation Guide (9X9KS)	-	A2375-90007	Jan-98	n	n	.pdf
HP PA-RISC Computer Systems System Upgrade Guide (9X9KS)	-	A2375-90010	Feb-97	n	n	.pdf
I/O Upgrade Manual (9X9KS)	-	A2375-90027	Jan-98	n	n	.pdf
Installation Guide (99X Family)	-	A1809-90001	Oct-97	n	n	.pdf
Internal Peripherals Update Guide (9X9KS)	-	A2375-90008	Jan-98	n	n	.pdf
Operator's Guide (99X Family)	-	A1809-90009	Jun-97	n	n	.pdf
Site Preparation and Requirements Guide (99X Family)	-	A1809-90002	May-97	n	n	.pdf
System Memory Upgrade Guide (9X9KS)	-	A2375-90009	Jan-98	n	n	.pdf
System Upgrade Installation Guide (CS 99X Family) (990/992 to 996)	-	A3310-90002	May-97	n	n	.pdf
System Upgrade Installation Guide (CS 99X Family) (991/995 to 996)	-	A3310-90001	May-97	n	n	.pdf
Networking and Communications						

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
APPC Subsystem on MPE/XL Node Manager's Guide	30294-61002	30294-90007	Jun-92	n	n	n
Configuring and Managing Host-Based X.25 Links	36939-61004	36939-90054	Oct-98	n	y	y
HP ARPA File Transfer Protocol User's Guide	36957-61002	36957-90158	Oct-98	n	y	y
HP SNA Products Remote System Configuration Guide	J2220-61025	30292-90008	Mar-95	y	y	y
HP SNMP/XL User's Guide	36922-61029	36922-90036	Mar-94	y	n	y
HP Telnet/iX User's Guide	36957-90154	36957-90156	Oct-98	n	y	y
HP-PB 100Base-T Network Adapter Installation and Service Guide	B5427-90001	B5427-90001	Aug-97	n	y	y
HP-PB 100VG-AnyLAN Network Adapter Installation and Service Guide	B5425-90001	B5425-90001	Aug-97	n	y	y
LU 6.2 API Application Programmer's Reference Guide	30294-61000	30294-90008	Jun-92	n	n	n
Managing Host-Based X.25 Links Quick Reference Card	36939-61003	36939-90051	Dec-94	n	y	y
NetIPC 3000/XL Programmer's Reference Manual	36920-61005	5958-8600	Oct-89	y	n	y
NS Cross-System NFT Reference Manual	36920-61003	5960-1634	Jan-89	n	n	n
NS3000/iX Error Messages Reference Manual	36923-61000	36923-90041	Oct-98	n	y	y
NS3000/iX NMMGR Screens Reference Manual	36922-61003	36922-90038	Oct-98	n	y	y
SNA IMF Programmer's Reference Manual	30293-61005	30293-90009	Jun-92	y	n	y
SNA IMF/XL Node Manager's Guide	30293-61000	30293-90010	Jun-92	n	n	n
SNA Link/iX Node Manager's Guide	30291-90009	30291-90009	Mar-94	y	y	y
SNA NRJE Node Manager's Guide	30292-61000	30292-90007	Oct-92	y	n	y

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
SNA NRJE User/Programmer Reference Manual	30292-61001	30292-90006	Oct-92	y	n	y
Using NS3000/iX Network Services	36920-61000	36920-90008	May-94	y	y	y
Using SNA IMF Pass Thru	30293-61008	30293-90006	Dec-90	y	n	y
Network and System Management						
Configuring and Managing MPE/iX Internet Services	32650-90835	32650-90859	Oct-98	n	y	y
Configuring Systems for Terminals, Printers, and Other Serial Devices	32022-61000	32022-90051	Oct-98	n	y	y
Customizing Terminal and Printer Type Files with the Workstation Configurator	5959-2870	32022-90031	Feb-94	n	y	y
HP 3000/iX Network Planning and Configuration Guide	36922-61023	36922-90037	Oct-98	n	y	y
HP OpenView System Manager Manager's Guide	36936-61002	36936-90202	Mar-95	n	n	n
HP OpenView System Manager User's Guide	36936-61001	36936-90201	Mar-95	n	n	n
HP Security Monitor/iX Manager's Guide	32650-90455	32650-90498	Apr-94	n	y	y
HP Security Monitor/iX User's Guide	32650-90454	32650-90497	Apr-94	y	y	y
Installing and Managing HP ARPA File Transfer Protocol Network Manager's Guide	36957-61001	36957-90157	Oct-98	n	y	y
Manager's Guide to MPE/iX Security	32650-90474	32650-90473	Apr-94	y	y	y
Managing Spooler Operations Quick Reference Pocket Card	32650-90268	32650-90488	Apr-94	n	y	y
MPE/iX HP 3000 Series 99X Software Startup Manual	36123-90046	36123-90046	Apr-94	n	n	n
Native Mode Spooler Reference Manual	32650-90166	32650-90867	Oct-98	y	y	y

Manual Title	Previously Used CO Number	Part Number	Latest Edition	Laser ROM 11/97 LR	Instant Info 6.0	View on the WEB*
New Features of MPE/iX: Using the Hierarchical File System	32650-90351	32650-90492	Apr-94	y	y	y
NS3000/iX Operations and Maintenance Reference Manual	36922-61005	36922-90039	Oct-98	n	y	y
Openview DTC Technical Reference Manual	5961-9820	5961-9882	Oct-94	n	n	n
Performing System Management Tasks	32650-90004	32650-90854	Jul-96	y	y	y
Performing System Operation Tasks	32650-90137	32650-90484	Apr-94	y	y	y
STORE and TurboSTORE/iX Products Manual	B5151-90001	B5151-90002	Jul-96	y	y	y
System Startup, Configuration, and Shutdown Reference Manual	32650-90042	32650-90855	Jul-96	y	y	y
Troubleshooting Terminal, Printer and other Serial Devices	32022-61002	32022-90030	Oct-93	y	y	y
User's Guide to MPE/iX Security	32650-90472	32650-90471	Apr-94	y	y	y
Using HP OpenView DTC Manager	D2355-90001	D2355-95018	Jan-93	n	n	n
Using the HP 3000 Workload Manager	B3879-90001	B3879-90002	Dec-94	y	y	y
Using the Node Management Services (NMS)Utilities	32022-61005	32022-90053	Oct-98	n	y	y
Volume Management Reference Manual	32650-90045	32650-90491	Apr-94	y	y	y
* The documentation website address is: *http://www.docs.hp.com*.						

Index

Index

R

releases
  mainline, 271
  non-platform, 271
  platform, 271
  product history, 267
  supported, 271
remote shell, 210
REMSH client, 210, 213
  functionality, 210
  troubleshooting, 213
RPG/iX 4-digit year
    enhancement, 144

S

Samba/iX, 236
SCSI device configuration, 34
Sherlock, 66
Stage/iX, 49
STORE
  with NetWorker Client, 251
supported devices, 107
supported releases, 271
SupportLine
  MPE/iX patches available, 18
  via Internet, 18
Syslog/iX, 61, 260

T

TCP connections, 53
Transaction Manager, 52, 216
TurboIMAGE/XL
  dynamic detail dataset
    expansion fix, 166
  enhancement summary, 156
  enhancements, 154
TurboSTORE/iX
  with NetWorker Client, 251

U

update
  pre-planning, 25
updating to MPE/iX 6.0, 24
upgrading to a 997, 31
user
  account naming structure, 39
  documentation list, 275
user-defined job queues, 42

V

version compatibility, 25
VPLUS enhancements, 181

Y

year 2000

  and ALLBASE/SQL, 177
  VPLUS enhancements, 181
year 2000 enhancements, 119,
    181
year 2000 web page, 118