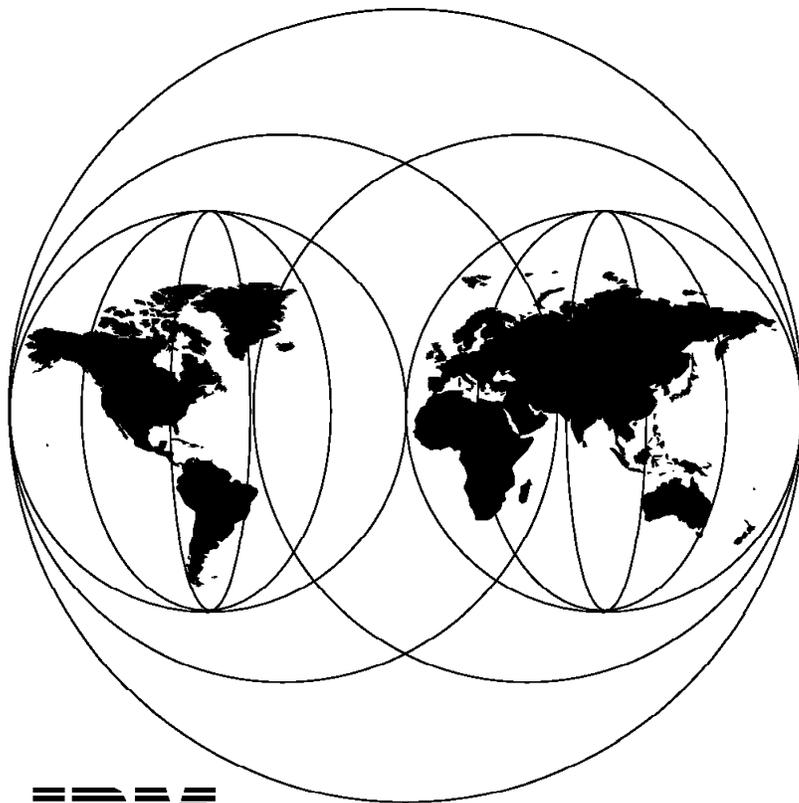


International Technical Support Organization

SG24-4847-00

P/390 and R/390 with OS/390: An Introduction

August 1996



IBM

**International Technical Support Organization
Poughkeepsie Center**



International Technical Support Organization

SG24-4847-00

P/390 and R/390 with OS/390: An Introduction

August 1996

Take Note!

Before using this information and the products it supports, be sure to read the general information under Appendix B, "Special Notices" on page 169.

First Edition (August 1996)

Comments may be addressed to:

IBM Corporation, International Technical Support Organization
Dept 541 Mail Station P099
522 South Road
Poughkeepsie, New York 12601-5400

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

© **Copyright International Business Machines Corporation 1996. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Preface	vii
How This Redbook Is Organized	vii
The Team That Wrote This Redbook	viii
Comments Welcome	viii
Chapter 1. Product Descriptions	1
1.1 Functional Flow	3
1.2 P/390 Microprocessor Complex	4
1.2.1 Channel Emulator and Device Managers	7
1.2.2 Starting the P/390 Subsystem - Overview	9
1.3 PC Server System/390 System	10
1.4 RISC/6000 S/390 Systems	12
1.4.1 RISC/6000 S/390 (Model 7012-390)	13
1.4.2 RISC/6000 S/390 (Model 7013-591)	14
1.5 Support Programs	14
1.6 Adapters and Connections	18
1.7 OS/390 on CD-ROM	19
1.8 Positioning and Usage	20
1.8.1 Selecting a System	22
Chapter 2. Configurations and the P/390 Configurator	23
2.1 Server Workload Considerations (P/390)	23
2.2 PC Server System/390 Configurations	24
2.3 RISC/6000-591 S/390 Configurations	29
2.4 3270 Connection Configurations - Overview	30
2.5 Tape Configurations	34
2.6 The P/390 Configurator	36
Chapter 3. Installation	43
3.1 PC Server System/390 OS/390 Installation	43
3.1.1 Hardware Installation	43
3.1.2 OS/2 Installation	46
3.1.3 CM/2 Installation	46
3.1.4 P/390 Support Programs	47
3.1.5 OS/390 CD-ROM Installation	48
3.2 IPL and Use MVS	49
3.3 RISC/6000 S/390 OS/390 Installation	50
3.3.1 Hardware Installation	50
3.3.2 AIX Installation	50
3.3.3 P/390 Support Programs	53
3.3.4 OS/390 CD-ROM Installation	54
3.3.5 IPLing OS/390	57
3.4 Expanding the CD-ROM Systems	58
3.5 Working With the P/390 Configurator	59
3.6 Non-RAID Installation (PC Server)	63
3.7 Installing Other MVS or OS/390 Systems	64
3.7.1 Creating and Initializing Disk Volumes	65
Chapter 4. Device Managers and Commands	69
4.1 Starting Device Managers	69
4.2 Device Managers for DASD	70

4.2.1	AWSCKD	70
4.2.2	AWSFBA	75
4.3	Device Managers for Simple Terminals	75
4.3.1	AWS3274 (P/390)	75
4.3.2	AWS3274 (R/390)	76
4.3.3	LAN3274 (P/390 only)	78
4.4	Device Managers for Tape Drives	82
4.4.1	AWSTAPE Device Manager	82
4.4.2	SCSI34x0 and AWS34x0	86
4.4.3	ISITAPE Device Manager (P/390 Only)	89
4.5	Device Managers for Unit Record	90
4.5.1	AWS2540	90
4.5.2	AWS2821 Device Manager	92
4.5.3	AWS3215 Device Manager	96
4.6	Device Managers for Communications	96
4.6.1	LAN3172 Device Manager (P/390)	97
4.6.2	LAN3172 Device Manager (R/390)	99
4.6.3	WAN3172 Device Manager (P/390)	100
4.6.4	WAN3172 Device Manager (R/390)	101
4.6.5	LCS3172 Device Manager (P/390)	101
4.6.6	LCS3172 (R/390)	103
4.6.7	MGR3172 (P/390 OS/390 Only)	104
4.6.8	AWSICA Device Manager (P/390 Only)	105
4.6.9	AWSPBS Device Manager (Normally VM and VSE)	106
4.7	Other Device Managers	106
4.7.1	AWSC370 Device Manager	106
4.7.2	LAN3088 Device Manager (P/390 only)	110
4.7.3	AWS2703 Device Manager (P/390 only)	110
4.7.4	AWSOMA Device Manager (P/390 only)	112
4.7.5	AWSPCSR (VM Only)	113
4.7.6	AWSMOUNT Command (P/390 and R/390)	113
4.7.7	AWSSTAT (P/390 only)	115
4.7.8	AWSSTART	115
4.7.9	AWSPROF Command (P/390)	116
4.7.10	CLRIO Command (P/390)	116
4.7.11	AWSCCFG Commands (P/390)	116
4.7.12	DEV2NAME	117
4.7.13	IPL Command (P/390)	117
4.7.14	ipl390 Command (R/390)	117
4.7.15	LTRENAME Command (P/390 Only)	118
4.7.16	RDEVMAP Command (P/390 Only)	118
4.7.17	BLDLIST Command (P/390)	118
4.7.18	C370MAP Command (P/390)	119
4.7.19	AWSCMLT Command (P/390)	119
4.7.20	TRCC370 Command (P/390)	119
4.8	Other Commands	119
 Chapter 5. Other Topics		 121
5.1	S/390 Manual Operations	121
5.2	Diagnostic Facilities	123
5.3	Tape Drives	127
5.4	XTAPE Product (P/390 Only)	131
5.5	OS/390 Device Planning	132
5.6	Security	133
5.7	OS/2 and Hardware Tuning	134

5.8 System/390 Activity Window	136
5.9 Device Manager Letters and Names	137
5.10 3270 Emulator Keyboards	137
5.11 Writing a Device Manager	138
5.12 Send and Receive	138
5.13 Backing Up OS/390	139
5.14 OS/390 Performance and Capacity (P/390)	140
5.15 tn3270 Emulators	149
5.15.1 xant 3270 Emulator	151
5.15.2 hcon 3270 Emulator	152
5.16 AWSICA and BSC NJE	153
5.17 Internet Address	154
5.18 LAN Connection	154
Chapter 6. Frequent Questions	155
6.1 PC Server System/390 Systems	155
6.2 RISC/6000 S/390 Systems	157
6.3 Questions Common to Both Bases	158
Appendix A. AWS2821 Translation Table (P/390)	167
Appendix B. Special Notices	169
Appendix C. Related Publications	171
Appendix D. How To Get ITSO Redbooks	173
D.1 How IBM Employees Can Get ITSO Redbooks	173
D.2 How Customers Can Get ITSO Redbooks	174
D.3 IBM Redbook Order Form	175
Index	177

Preface

This document introduces the IBM PC Server System/390 (commonly known as the P/390), with special emphasis on its use with OS/390. The discussion also includes the IBM RISC/6000 with S/390 Server on Board (commonly known as the R/390), but in less detail. The emphasis is on understanding the nature of these products, with their advantages and limitations. The reader is assumed to be a technical professional, with an MVS background. Typical configurations are described. Installation is described, briefly, as it is covered in detail in other documents. A small amount of observed performance data is included.

(181 pages)

This replaces an earlier redbook *MVS and the IBM PC Server 500 System/390*, GG24-2538. The general discussion context is for a system used for programming development, and suitable configurations are described for this purpose. *The reader is assumed to have a system programmer's knowledge of OS/390 or MVS.* This document is not an introduction to OS/390. It is an introduction to OS/390 running on the PC Server System/390 or the RISC/6000 S/390.

How This Redbook Is Organized

The descriptions in this document correspond with Version 2.1 of the P/390 support programs, level 4.0.0.1 of the R/390 support programs, and the IBM S/390 Developers' Association version of OS/390 on a CD-ROM. Later updates and releases may change some details, but the basic comments and recommendations should remain valid.

You should have *PC Server 500 System/390: Installation, Configuration, and User's Guide for MVS/ESA*, SA22-7210-1, and *PC Server 500 System/390: Messages and Codes*, SA22-7227, before installing your P/390 MVS system. These documents contain detailed installation instructions. You should also have the redbooks *Connectivity on a PC Server System/390*, SG24-4624, and *Printing with MVS on the IBM PC Server System/390*, SG24-4612, for more information in these areas.

The first chapter of this document describes the products concepts and positioning, and describes the general operation of the products.

The second chapter describes the variety of configurations that can be used, and how the P/390 configuration utility is used.

The third chapter describes, briefly, the steps required to install a system -- from Server hardware to OS/390.

The fourth chapter provides detailed information about the various device managers provided with the P/390 support programs.

The fifth chapter contains discussions of a number of areas related to OS/390 operation. A limited amount of performance information is provided.

The sixth chapter lists many frequently asked questions about these products, with associated answers.

The Team That Wrote This Redbook

This document is the result of work in the International Technical Support Organization at the Poughkeepsie Center. The author was Bill Ogden.

We extend thanks to the following people for substantial assistance in producing this document:

- Chuck Berghorn - IBM P/390 Development (Poughkeepsie)
- Bohdan Demczar - IBM Development (Poughkeepsie)
- Marty Ziskind - IBM P/390 Development (Poughkeepsie)
- Louis Voerman - IBM P/390 Development (Poughkeepsie)
- Carmine Castaldo - IBM R/390 Development (Poughkeepsie)
- Joel Naslow - IBM P/390 Development (Poughkeepsie)
- Doris Conti - IBM Development (Poughkeepsie)
- Gordon Chamberlain - Interprocess Systems, Inc. (Roswell, Georgia)

Comments Welcome

We want our redbooks to be as helpful as possible. Should you have any comments about this or other redbooks, please send us a note at the following address:

redbook@vnet.ibm.com

Your comments are important to us!

Chapter 1. Product Descriptions

The products discussed in this document are built around a single adapter card, the S/390 Microprocessor Complex. This adapter is often referred to as the *P/390 adapter*, and is explained in some detail in 1.2, "P/390 Microprocessor Complex" on page 4. The adapter is part of several products, one based on an IBM PC Server and two based on RISC/6000 systems.

To avoid writing "PC Server or RISC/6000 system" hundreds of times throughout this document, we will refer to a generic "Server" in any discussion that applies to both PC Server and RISC/6000 based products. This usage ignores the common distinction between *workstations* and *servers*, and is solely for the purpose of this document. We elected not to use the word "host" for the PC Server or RISC/6000 base system. Although it is a more logical word, it is very firmly associated with mainframes, and could be especially confusing in a document dealing with OS/390 and MVS.

The proper names for the products are *IBM PC Server S/390* and *IBM RISC/6000 with System/390 Server On Board*. They are often referred to as P/390 and R/390 systems, respectively, although these names are not strictly proper, and may confuse the adapter card with the total system.¹ The key portion of the names, "S/390," indicates the unique nature of the products. With proper setup of the underlying Server systems, they provide System/390 environments. OS/390 (MVS), VM, and VSE, with all their subsystems and applications, will run on these systems -- with very few limitations.

We will use the term "OS/390" to mean both OS/390 and any reasonably current release of MVS. For the purposes of this document, there is no distinction between OS/390 and MVS. This document is about P/390 OS/390; much of what is discussed also applies to VM and VSE systems, but they are not the subject here and are seldom mentioned. Also, as a final note on terminology, we will use "storage" rather than "memory" for P/390 storage.²

We will reference the document *P/390 & R/390: OS/390 New Owner's Cookbook*, SG24-4757, a number of times and simply refer to it as the *Cookbook*. It contains brief, specific instructions for some of the basic administrative tasks needed with OS/390. The descriptions are based on the S/390 Developers' Association OS/390 CD-ROM system, which is described later in this document. (The *Cookbook* is scheduled to be available in late 1996.)

The P/390 adapter contains a S/390 processor. It does not emulate or simulate a S/390 processor, it *is* a S/390 processor. While it is a S/390 processor, it is not a S/390 I/O subsystem. The underlying Servers are used to emulate³ the I/O subsystem and a selected set of S/390 I/O devices. Most of this document, as

¹ Thus a "P/390 system" could be either a PC Server or a RISC/6000 system, with the "P/390" referring to the S/390 adapter. In another context, "P/390" may refer to the PC Server S/390 system, in contrast to an R/390 system. The context of a discussion usually indicates which meaning is implied.

² Memory and storage are the same thing, but mainframe terminology is usually "storage" and PC/RISC terminology is usually "memory"

³ Some documents imply subtly different meanings for the words "simulate" and "emulate." This document is not so sophisticated, and arbitrarily uses the word "emulate" when describing the functions of the P/390 support programs.

well as other redbooks dealing with these products, is concerned with this emulation of S/390 I/O functions.

There are differences between these systems and a S/390 mainframe, but these differences are generally outside the normal operating system and application program interfaces. The differences include:

- System partitioning into multiple system images (“LPARs”) is not available.
- Multiprocessor functions are not available.
- Shared DASD, with a mainframe or another P/390 OS/390 system, is not available (at the time this was written).
- IOCDs configuration functions are not used. These are replaced by (simpler) I/O configuration controls through an OS/2 or AIX program, and a file known as DEVMAP.
- Integrated communications I/O, as found on IBM 9370 and 9221 systems, is emulated through various Server drivers. Not all integrated communications functions are emulated. (OS/390 does not support these integration communication adapters for VTAM use. OS/390 can use them as basic bisync devices, such as JES2 might use, but this is no longer common and is not formally supported.)
- Sysplex connections are not supported.
- Parallel bus and tag channel connections are available, using a S/370 channel adapter card and a driver program on Servers. Only certain devices are supported through this path. In particular, DASD connections are not supported at this time.
- ESCON channel connections are not available at this time.
- Integrated console attachments (such as used on IBM 4381, 9370, and 9221 systems) are not supported. The Server provides a 3270 emulator which operates as though an IBM 3174 control unit were attached. The emulator provides multiple 3270 sessions, and can be used for both console and user sessions.
- Expanded storage functions are supported. You can partition the S/390 storage into standard and expanded storage. However, there is a one-to-one tradeoff between standard and expanded storage, and most users elect to use full standard storage and no expanded storage.

In addition to emulating a S/390 I/O subsystem, the Server uses P/390 adapter interfaces to initiate IPLs, perform various S/390 resets, and provide most S/390 operator functions -- such as register and storage displays.

A package of P/390 support programs must be installed in the Server to provide the I/O emulation and console functions for programs executing in the P/390 adapter. These *P/390 support programs* are a key part of the complete products. (In keeping with the slightly confusing terminology that exists for these products, P/390 support programs are used to support the P/390 adapter, which is used in both P/390 and R/390 systems. There are two packages of P/390 support programs: one for P/390 systems and one for R/390 systems.)

1.1 Functional Flow

The functional flow shown in Figure 1 is very important in understanding these systems, and we suggest you take time to understand the concepts involved. While the figure is a high-level abstraction, it represents a Server with a P/390 adapter, running an OS/390 application. Two processors are indicated: the left side (in the figure) is the S/390 processor, and the right side is the Server processor (a Pentium or RISC processor). Each “side” has its own storage. *The processors do not share storage.* For OS/390, the P/390 adapter will normally have 128 MB storage; the size of Server memory will vary, but it will have at least 32 MB.

In the figure, an OS/390 application is executing -- this means it is executing S/390 instructions, just as though it were running on a mainframe. At the same time, the Server can be executing its own workload⁴. The flow of control might go like this:

- The OS/390 application encounters an I/O function, such as a GET macro.
- This passes control to an access method. The access method may construct a channel program and issue an EXCP request (or something similar) to OS/390. For example, this might be a read operation for a 3380 disk at S/390 address 120. As far as OS/390 is concerned, it thinks it has a “real” 3380 at this address.
- OS/390 gets control, schedules the channel program, and eventually issues an SSCH⁵ instruction. Up to this point, operation has been exactly the same as mainframe operation.
- The SSCH instruction works differently than on a mainframe. It constructs several control blocks, in control storage not visible to OS/390, and causes an interrupt in the Server system.

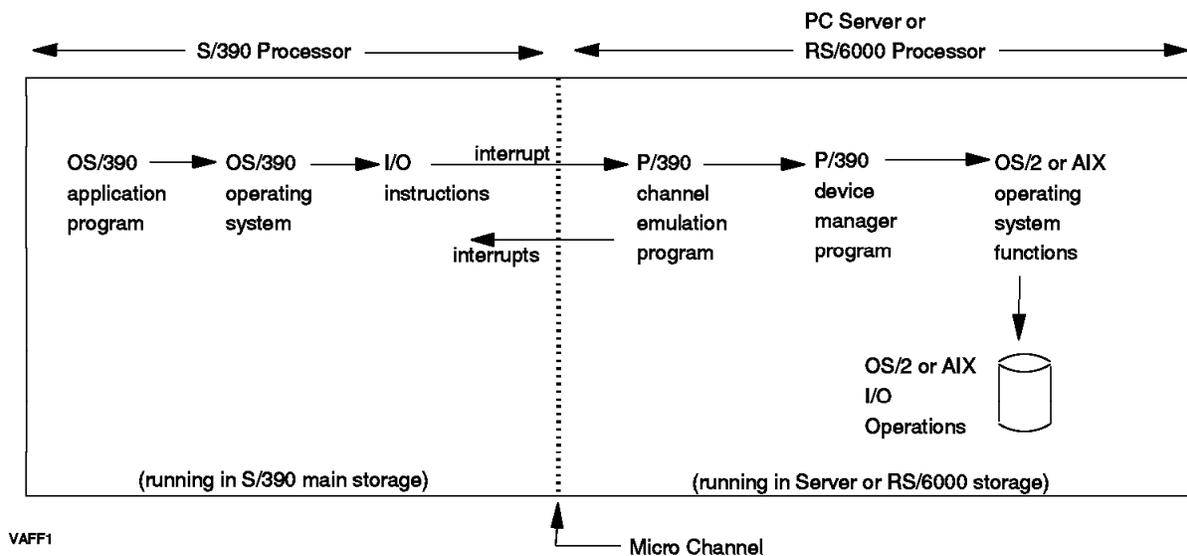


Figure 1. Conceptual Flow of Control

⁴ We will discuss the practicalities of additional Server workloads later.

⁵ Start Subchannel. This is the modern equivalent of the original SIO (Start I/O) instruction in the S/360.

- A P/390 support program (running under OS/2 or AIX) gets control (based on the interrupt from the P/390 adapter).
- It gives control to the P/390 channel emulator program. As its name implies, this program emulates many of the functions of a S/390 channel. It sustains a number of parallel operations in the “channel,” permitting OS/390 to have multiple outstanding I/O operations. It interprets portions of CCWs, and provides initial condition codes for SSCH instructions.
- The channel emulator determines which device type and address are involved (a 3380 disk at address 120, in this example, based on the definitions in a DEVMAP that is not shown), and gives control to a device manager program. At the same time, it usually returns initial status to the P/390, completing the SSCH instruction. The P/390 continues executing S/390 instructions, running in parallel with the emulation programs on the Server. In the Server, a DEVMAP (device map) is used to relate S/390 addresses (such as 120) to particular device managers (such as the one used to emulate a 3380 disk). The DEVMAP is maintained by the P/390 configurator program (operating on the Server), and is remotely similar to the IOCDs of a mainframe. The DEVMAP is described in considerable detail later.
- The P/390 support programs include many device managers, for different types of devices. In general, a device manager emulates a particular type of “real” mainframe device.⁶ In this example, the particular device manager (which would be the AWSCKD.EXE program for a P/390 system) emulates a 3380 disk drive.
- The device manager uses a real Server I/O device, and issues normal READ/WRITE instructions through OS/2 or AIX to access the device. The device manager calls the channel emulator, as needed, to transfer data to/from P/390 storage. The channel emulator does not need to interrupt the P/390 to do this; it can access P/390 storage by using a movable *window* that is accessed from the Server side.
- When the device manager completes the requested function, it notifies the channel emulator. The channel emulator then causes an I/O interruption in the P/390, and creates a CSW (channel status word) with appropriate indicators, such as channel end and device end.

While the analogy is not exact, the P/390 channel emulator functions much as a mainframe channel does, and P/390 device managers tend to perform the functions of mainframe control units.

1.2 P/390 Microprocessor Complex

This section provides slightly more detail about the system. None of this material is necessary for system use, but it may help you become more comfortable with general concepts. The S/390 processor card used is shown in Figure 2 on page 5. It is a Micro Channel adapter card. The S/390 processor is a single chip on the card. The processor has approximately 220K gates and uses 420 signal pins of the 647-pin mount. (The other pins are used for power and grounds.) It has 32KB of control storage and uses horizontal microcode with 136 bits per word. Internal data flow is 64 bits wide.

⁶ This is not exactly the case when a S/370 Channel Emulator/A adapter is used, but this is described later.

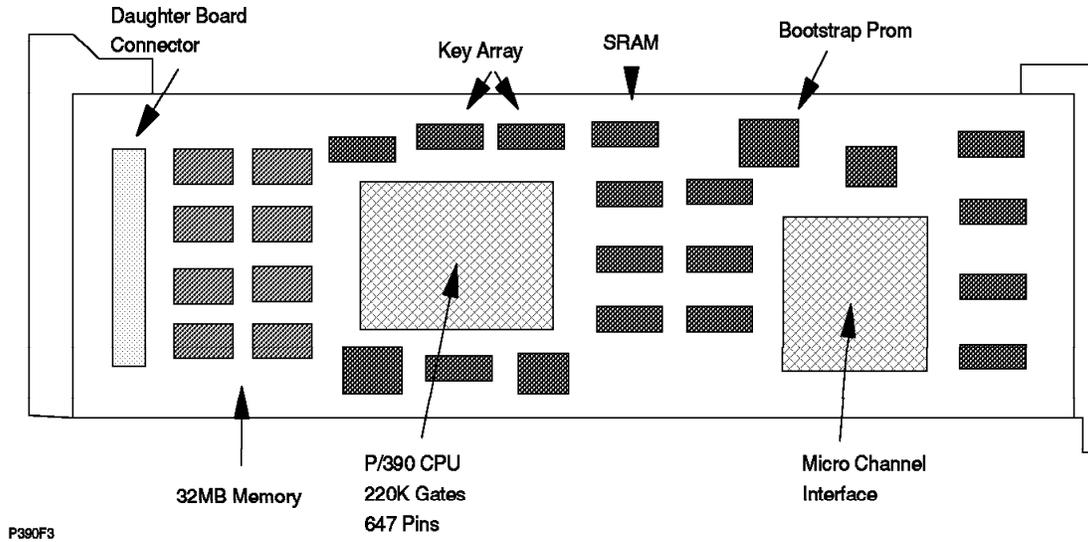


Figure 2. P/390 Processor Complex Adapter

The basic P/390 adapter has 32 MB storage. When a daughter card is used (to add 96 MB additional S/390 storage), the storage is interleaved.⁷ The 32MB storage on the S/390 processor card is not the “first 32MB.” We assume the additional 96 MB is always used when OS/390 is used. The daughter card requires a Micro Channel slot; it uses the slot for power and ground connections, but does not transfer data through the Micro Channel.

The P/390 adaptor contains its own timing circuits, and its clocking is independent of the Server. The current P/390 contains a 71 MHz clock that is divided into a four-phase clock of approximately 17.7 MHz. Different S/390 instructions require different numbers of clock cycles to complete, but the average performance is approximately 4.5 S/390 processor MIPS.

An important design goal was to avoid any modifications to OS/390 (or any other S/390 operating system used with the system). The key to doing this was to move all I/O operations to the OS/2 side of the system. No modifications are required for OS/390 to run on this system, although some reconfiguration may be appropriate.

The S/390 microprocessor is a single chip. It is controlled by microcode that is loaded when the P/390 subsystem is started.⁸ The S/390 processor (through hardware and microcode) implements the full S/390 subchannel architected interfaces; that is, a S/390 program can issue all the defined I/O instructions and work with the control blocks associated with these instructions. The subchannel control blocks (as used in all System/390 platforms) are the link between the S/390 processor and the OS/2 support programs.

When the S/390 program (usually OS/390’s IOS code) issues an SSCH instruction (Start SubChannel), an interrupt is generated for the OS/2 side of the processor. The OS/2 code (part of the I/O subsystem provided on the P/390 diskettes) can

⁷ Two different daughter cards were originally available, one with 32 MB and one with 96 MB. The 32 MB card (which brought the total P/390 adapter to 64 MB) is no longer available.

⁸ The microcode is included on the P/390 support diskettes.

access S/390 memory and can “see” the ORB, CCWs, I/O areas, and so forth, that are associated with the I/O operation. A combination of OS/2 code and P/390 microcode maintains SCHIBs that correspond to emulated subchannels.⁹

An IOCDS is not used. An equivalent mapping operation (and an IOCDS is essentially a map) is built from the DEVMAP file you build with the P/390 subsystem configurator function.¹⁰ There is a single path to each emulated device.

The S/390 processor is a full System/390 ESA processor. It executes S/390 instructions as its native instruction set. In addition:

- All of the “ESA” and “XA” instructions (such as Branch and Save) are present.
- Extended instructions are available, including string instructions, square root, cancel I/O, compression, expanded sorting, move inverse, move page(2), PER-2 and extensions, private space, data spaces, ACL protection, address limit checking, broadcast purging, subspace group, compare until substring equal, incorrect length indication suppression, set address space control fast, storage protection override, suppression on protection, and so forth.
- Some “assist microcode” is present, including Interpretive Execution (Interception format 2, PER extensions, VM data space, Storage-Key functions), add FRR, SVC assist, Obtain/Release Local Lock, Obtain/Release CMS lock).
- Low address protection, fetch protection override, and public storage key control are supported.

A few other considerations are:

- Emulated disks (3380s, for example) do not have a “CE” track. They need to be initialized as though they were minidisks. (Both the stand-alone ICKDSF program and the OS/390 version do this.)
- In general, diagnostic CCWs are not fully processed. The intention is that all “real” device recovery is done by the OS/2 programs, and OS/390 will see only successful I/O operations or simple failures (“device not ready,” for example). OS/390 will not be called upon to issue complex I/O diagnostic operations. IBM S/390 I/O maintenance tools may not work correctly (and should not be used).
- Obscure sense bits are not emulated.¹¹
- Older devices, potentially attached through a S/370 Channel Emulator/A adapter, have not been tested.
- The S/370 Channel Emulator/A adapter does not support “Read Backward” commands. (These commands were used by the SORT program, in conjunction with 3420 tape drives, many years ago. The only current usage of read backwards that we have encountered is with OS/390 standard label processing; special-case code is included to handle this usage.)

⁹ See the S/390 Principles of Operation manual for definitions of these terms.

¹⁰ Suitable DEVMAPs are included with the various operating system releases on CD-ROM that are intended for use with these systems.

¹¹ For example, a real 3420 tape drive has a sense bit that indicates that the “left reel is turning.”

- There is no hardware support for 2K storage keys; only 4K keys are supported. (VS1 used 2K keys.)

The P/390 subsystem provides several trace functions. By using the Trace icon, trace data can be displayed or sent to a file. There are two trace types:

1. Kernel trace, that records all S/390-Server interactions
2. Device trace, that records only interactions associated with a specific emulated device

The normal trace table (this is the P/390 subsystem trace table, which has no relation to the OS/390 trace table) has 2000 entries. It is most useful for debugging emulated I/O problems.

1.2.1 Channel Emulator and Device Managers

To a certain extent, the P/390 support programs are structured like mainframe hardware. The CPU and central storage communicate with channels, the channels communicate with control units, and the control units work with devices. In our case, the AWSCHAN.EXE (using the P/390 module names for these examples) program is the channel. On one side, it works with CPU interrupts, I/O requests, and storage, and on the other side it works with control units, emulated or real.

The AWSCKD.EXE device manager program is an example of an emulated control unit. Real control units can be attached through the S/370 Channel Emulator/A adaptor, although the AWSC370.EXE module is needed to interface the channel (AWSCHAN) to the adapter. The Server system and devices, to some extent, correspond to the end devices that are managed by control units. The analogy with mainframe hardware is not exact, but it is close enough to help understand the general design of the P/390 support programs.

Like a mainframe channel, AWSCHAN supports multiple concurrent activities between the CPU and various control units. A mainframe channel is often limited to eight control units, while AWSCHAN has no fixed limit to the number of device managers (emulated control units) it can manage. It is limited to 255 total devices, as seen by the P/390 configurator (which is described later).

AWSCHAN provides handling of all S/390 I/O instructions, initial handling of all CCWs, manages all accesses between device managers and the S/390 main storage¹², and manages all I/O interrupts sent to the S/390. In general, it handles many parallel functions. Internally, it uses a fixed number of buffers to pass data to/from device managers and S/390 storage. AWSCHAN is both a very complex module, and a key module for P/390 support, and its developers have continually refined and improved it. Some of these improvements account for the dramatic performance improvements included in Version 2.1 of the P/390 support programs.

The key improvements were in handling (emulated) PCI¹³ functions. OS/390 program fetch, which involves reading program text and relocation information from disk, and relocating programs as they are placed into main storage, is

¹² There are a few exceptions to this, in which other programs or adapters directly access S/390 storage, but these exceptions do not negate the general design.

¹³ Program Controlled Interruption. This is a flag in a CCW that causes the channel to interrupt the CPU when operation for that CCW is started. The channel continues to process the CCW and the rest of the channel program. After receiving the interrupt,

much faster if PCI is fully used. The same changes in AWSCHAN also improved the processing of partitioned data set directories. These two areas, PCI fetch and PDS directory processing, are two of the most important underlying functions for OS/390 performance.

The device managers are the key to emulating S/390 devices. A device manager has these characteristics:

- It emulates a device and its control unit.
- It interprets CCWs and performs whatever OS/2 I/O is required to emulate the requested I/O.
- It generates sense data as required.
- It may emulate multiple devices (such as multiple 3380 drives) and multiple device types (such as 3380 and 3390).
- It may (or may not) be multithreaded. A few major device drivers (such as AWSCKD for the P/390) are multithread, with one thread for each emulated device.
- Some serialization may be enforced to allow orderly operation by the device manager code.
- Each device manager decides how it wants to handle overlap of multiple devices. Complex drivers (such as AWSCKD) provide as much overlap as possible (and are usually limited by the underlying Server I/O design and devices).

There is no requirement for a device manager to use multiple threads; the managers for the R/390 generally do not use threads while many P/390 managers do use threads. Figure 3 shows the basic threads of a simple device manager for a P/390 base. The main line code receives an initial I/O request and provides initial status (so the SSCH instruction can complete). Final status (when the emulated I/O operation is complete) is provided by the back-end code (and thread). Asynchronous functions (such as an attention interrupt) are handled by the async code and thread.

The support programs have several interfaces to the S/390 processor card:

1. Interrupts (both ways).
2. Shared Memory. The Server can read S/390 storage through a movable window. Either real or virtual addresses can be used.
3. A communications buffer (on the adapter card). This buffer contains ICBs (interrupt control blocks) and SCHIBs (subchannel information blocks).
4. Manual operations functions such as alter/display, IPL, stop, start, and so forth.

The SSCH operation is the primary link between S/390 code and the supporting P/390 subsystem. The complete process, using the channel emulator and device manager, goes like this:

1. OS/390 issues an SSCH instruction to start an I/O operation.

the CPU can dispatch a program that processes data received from the channel thus far. The CPU could also chain additional CCWs, depending on the data obtained earlier, to the channel program.

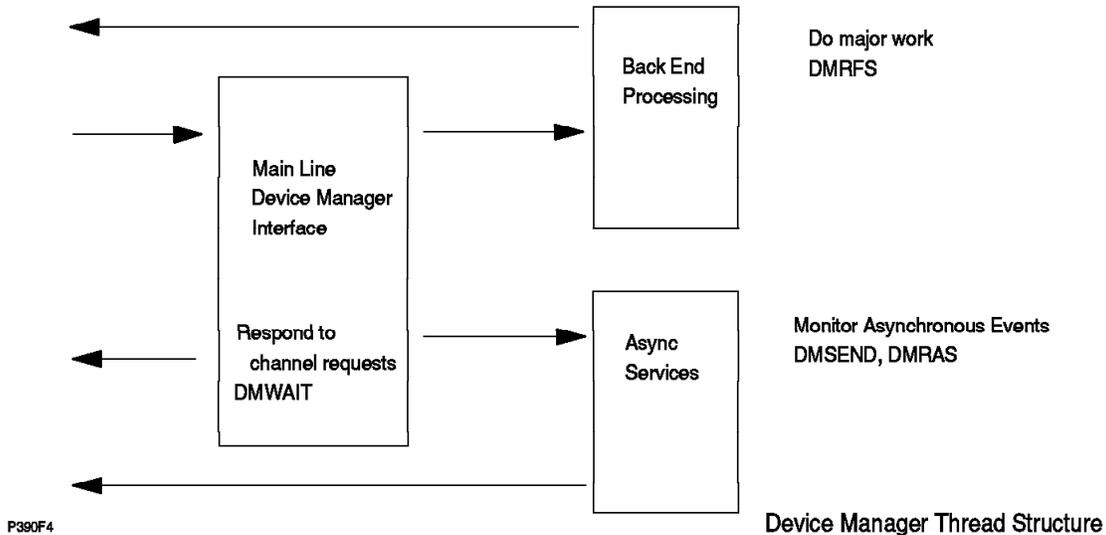


Figure 3. General Thread Structure for Device Manager

2. P/390 microcode moves data to the SCHIB, completes a ICB, and interrupts the Server.
3. The P/390 router gets control and passes control to the P/390 channel support program.
4. The channel module:
 - a. Checks if the correct device manager is available
 - b. Checks the emulated device state
 - c. Releases the S/390 with CC=0 (if the device state warrants it)
 - d. Routes the SSCH request to the device manager
5. The device manager:
 - a. Validates the CCW
 - b. Passes initial status to the channel
 - c. Interprets the CCWs, performs Server I/O, emulates CCW chaining, provides PCI interrupts, and so forth
 - d. Returns final status to the channel

The timing characteristics of this process do not exactly match the characteristics of a mainframe. If programs follow correct coding practices (and all the major MVS products do, as far as we know) there are no problems. If a program modifies CCWs too soon, there could be incompatibilities. Good program design calls for using PCIs before modifying active CCW chains, and this works properly.

1.2.2 Starting the P/390 Subsystem - Overview

When the Server system is booted, the P/390 adapter is idle. The Server must load P/390 microcode and then use control instructions to start P/390 processing. There is no requirement to use the P/390 adapter, of course. A user can boot OS/2 or AIX and use it normally. The P/390 adapter becomes active only when the appropriate commands are issued to start it. The P/390 functions are useful only when a S/390 program is available (on disk or tape) to execute. The S/390 program to be executed is normally the operating system (OS/390), but it could be a standalone program such as ICKDSF or a tape-to-disk restore program.

The P/390 functions are normally started with the IPL.CMD REXX command procedure. (The "IPL P/390" icon issues this command internally.) The startup operation goes like this:

1. Issue the IPL.CMD command (usually by clicking on an icon).
2. IPL.CMD starts AWSMAIN (one of the P/390 support programs).
3. AWSMAIN, among other things:
 - a. Loads the S/390 processor microcode (if it is not already loaded)
 - b. Loads the current DEVMAP.
 - c. Starts AWSCHAN (to begin emulated channel operations)
 - d. Builds SCHIBs to correspond to devices defined in the DEVMAP.
4. Start all the device managers. Each manager will decide (by inspecting the DEVMAP details) whether it is needed.
5. Wait for the device managers to initialize.
6. Issue an IPL function to the S/390 processor.
7. The S/390 processor executes an SSCH instruction to the I/O address specified in the IPL function.
8. The I/O operation is emulated by the appropriate device manager.
9. Control is given to the program instructions read by the initial I/O operation.
10. Operation continues, and OS/390 (or whatever was selected) loads itself.

1.3 PC Server System/390 System

The original P/390 (announced mid-1995) used a PC Server 500 as the base. The current P/390 (announced mid-1996) uses a PC Server 520 as the base. Characteristics include:

- CPU - a 133MHz Pentium (The PC Server/500 had a 90Mhz Pentium.)
- Memory - 32MB (default) to 256MB 70 ns, 2-way interleaved on a 64-bit interface. ECC error detection and correction is standard.
- Six Micro Channel and two PCI slots are included. (The Server 500 had eight Micro Channel slots.)
- Two serial ports and two parallel ports
- A SCSI adapter is standard, on the mother board. (The Server 500 did not have this.)
- An SVGA adapter is standard, on the mother board. (The Server 500 did not have this.)
- Enclosure - 18 disk bays (!), up to 38GB internal disk storage (with combinations of 1 inch, HH, and FH drives), 434 watt power supply, variable speed fans, lockable media door, tamper-evident covers, LogicLock (C2 functionality), message LED, and other features.
- Systems intended for P/390 use include:
 - A fast/wide SCSI RAID adapter with two or three channels. (The Server 500 RAID adapter had two channels.)
 - A 4mm tape drive and a CD-ROM drive.
 - Five 2.25 GB fast/wide disk drives

Early experience with the PC Server 520 indicates that the faster Pentium processor leaves more capacity available for OS/2 work (while OS/390 is also running). It does not cause OS/390 to run significantly faster, although we expect it will allow more effective parallel usage of disk drives.

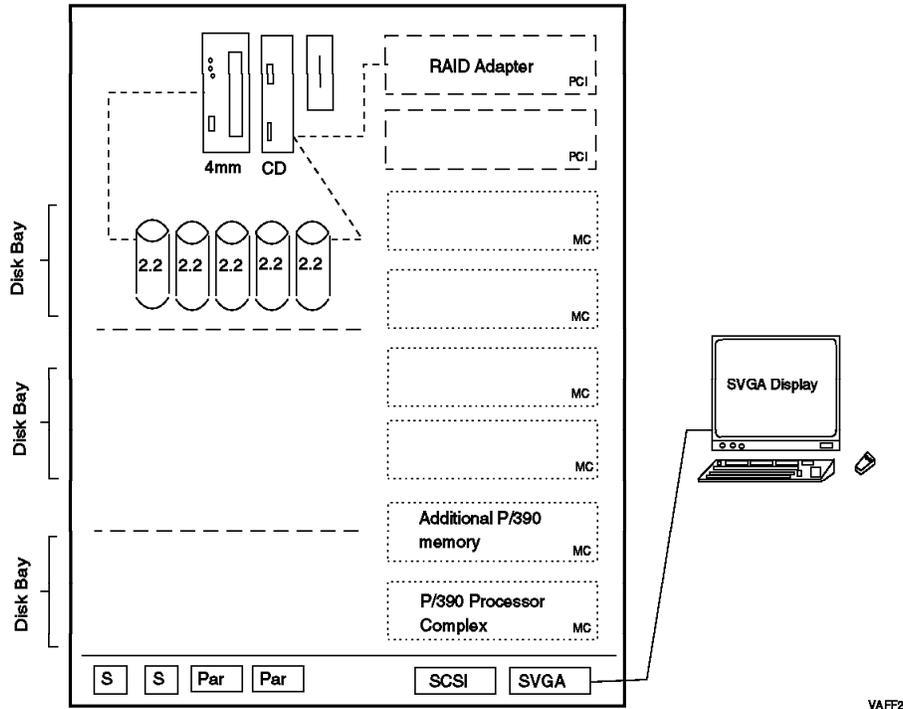


Figure 4. Basic PC Server System/390 for OS/390. This is the base configuration of a system intended for use with OS/390. Use with OS/390 implies that the additional P/390 memory "daughter card" is installed, and five 2.25 GB disk drives are installed. Later versions of the P/390 may use the planar SCSI adapter for the 4mm tape and the CD-ROM drive.

This basic system, as shown in Figure 4, leaves four Micro Channel slot and one PCI slot available for additional adapters. One potential disk bay slot may be lost in providing a SCSI connection for the 4mm tape drive, depending on the internal connections selected. The two lower disk bays can each be used for six thin (one inch high) disk drives, or three half-high disk drives. All disks must be mounted in *hot swap trays*. (Note that the integrated SCSI adapter, on the planar board, could also be used to drive the CD-ROM and 4mm tape drive since both this adapter and these devices use 8-bit SCSI interfaces. Production systems, built after this was written, may use this method.)

The RAID adapter controls the first bay of disk drives (five or six drives) and possibly the CD-ROM and 4mm tape drive. It has one or two more channels. The second channel is used for another bank of internal disk drives. The third channel can be used for yet another bank of internal disk drives, or for external SCSI devices. In general, the planar SCSI adapter would be the first choice for external SCSI connections.

Using the two lower disk bays requires two *backplanes* (into which the hot swap trays connect) and one additional *power supply*. You can order these when your initial system is built, or you can order them later.

A large number of configurations are possible, based on this initial configuration. Several of these are described in Chapter 2, "Configurations and the P/390 Configurator" on page 23. The base configuration, as shown in the figure, is completely sufficient for OS/2 and has sufficient disk space to emulate 3380 drives to load and run OS/390, with space for DLIBs, CICS, DB2, work volumes,

and a reasonable amount (a double-capacity 3380 drive, for example) of data space. The server display is used for up to five 3270 sessions, including one for the OS/390 console.

The supported operating system is Warp Server. In this case, *supported* means that IBM will accept problems reported on the system. A substantial number of systems are known to be using Warp Connect, although this is not officially *supported*.

The base Server system includes 32 MB memory, and this is sufficient for use with the P/390 subsystem. More memory should be considered if there will be significant Server memory use in parallel with P/390 activity.

The planar board provides an SVGA adapter. A display must be ordered. Use of the P/390 functions is not dependent on any particular display size or resolution, however we strongly recommend using a good quality 17-inch display in 1024x768 mode. Normal P/390 utilization will have several 3270-emulation windows open, and this becomes marginal on smaller displays or with lower resolution.

The P/390 support programs emulate certain DASD devices for OS/390 to use. The emulated DASD uses large files on normal OS/2 disks. These OS/2 disks are in either FAT or HPFS format.¹⁴ We recommend using HPFS for your OS/2 disk partitions that will contain emulated volumes. FAT partitions cannot exceed 2 GB; you will be working with much more space than this¹⁵, and attempting to use FAT partitions may cause unnecessary disk fragmentation or reloading of emulated volumes.

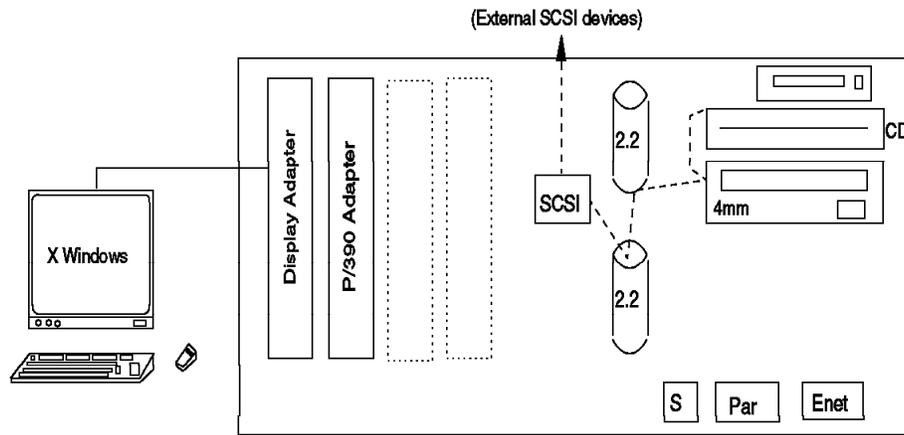
1.4 RISC/6000 S/390 Systems

Two R/390 systems based on RISC/6000 units are available. One is intended as an entry system, and the other as a high-end production system. Both use AIX as the Server operating system. AIX Version 4.1 (or later) is required. Earlier releases are not *supported* as bases for R/390 systems.

The configurations for the R/390 are not as standardized as is the PC Server P/390. Especially for the larger model, a wide range of disk configurations, display adapters, LAN adapters, and so forth, is possible. The S/390 functions place no unique requirements on the RISC/6000 base, other than two Micro Channel adapter slots for the P/390 adapter (and additional storage), and sufficient disk space for OS/390.

¹⁴ FAT is the disk format used by DOS and Windows. It is named after the File Allocation Table that it uses to manage files. The High Performance File System is unique to OS/2. OS/2 supports both file systems. A disk partition is formatted with one or the other. In general, FAT disks are used for compatibility with DOS, since DOS can work with a FAT disk but not an HPFS disk. HPFS disks (partitions) are used when DOS compatibility is not needed. HPFS has advantages in allocation granularity, fragmentation control, and error recovery.

¹⁵ A single emulated 3380-K volume is about 1.8 GB.



VAFP3

Figure 5. Basic RISC/6000 S/390 System (Model 7012-390)

1.4.1 RISC/6000 S/390 (Model 7012-390)

This system is based on the desk-top model 390.¹⁶ The basic configuration is indicated in Figure 5. Key characteristics are:

- 67 MHz POWER2 processor
- 32 MB memory standard, with 512 MB maximum
- 80 MB/sec Micro Channel, with four slots
- Integrated SCSI-2 dual-port fast/wide adapter
- Integrated Ethernet adapter
- Two internal disk bays
- CD-ROM drive
- 4mm tape drive (optional, but required for R/390)

A display must be ordered for the system. We recommend a 17-inch unit with at least 768x1024 or 1024x1024 resolution. Normal R/390 utilization will involve several 3270-emulation windows (under X Windows) and a smaller screen, or one with less resolution will impact usability.

This system, the RISC/6000-390 S/390, is not recommended for use with OS/390. The configuration shown does not have sufficient S/390 storage (there is only the 32 MB on the P/390 adapter) or disk space for OS/390. This system is intended for use with VM or VSE. It could be upgraded to run OS/390, by adding the S/390 96 MB storage daughter card, and by changing the disks to larger ones or by adding external disks. This would tend to create a “maxed out” system suitable only as an entry-level OS/390 platform.

¹⁶ The presence of the characters “390” in the base RISC model name is just a coincidence. The 7012-390 systems were in use before the S/390 adapter and functions were added to it.

1.4.2 RISC/6000 S/390 (Model 7013-591)

This is a larger system, with more adapter slots and more disk bays. A general outline is shown in Figure 10 on page 30. Characteristics include:

- 77 MHz POWER2 processor, with 256KB data cache
- 64 MB memory standard, with 2 GB maximum
- 80 MB/sec Micro Channel, with eight slots
- Integrated SCSI adapter
- SCSI-2 dual-port fast/wide adapter (uses one MC slot)
- Six internal disk bays (3.5 inch, half high)
- CD-ROM drive
- Standard display adapter
- 4mm tape drive (optional, but required for R/390)

A display must be ordered, and the same considerations apply as for the 7012-390.

1.5 Support Programs

The P/390 adapter is useless without the P/390 support programs, which are usually delivered on a set of diskettes. (There are two sets of diskettes: one for the PC Server and one for RISC/6000 Servers.) A number of components are included:

- The channel emulator, already mentioned, and a number of programs involved in interrupt handling and routing
- A considerable number of device managers
- Configurator programs, support programs, and tables
- Microcode for the P/390 adapter, and a program to load it
- Operator functions for the P/390 adapter
- Window functions and icons for OS/2 or AIX
- Documentation files for the various device managers, and one or more READ.ME files
- Trace tools
- Various other utility functions, which typically operate as OS/2 or AIX commands.

The use of many of the support programs is centered around the *configuration program*. This is the program that defines the mapping of OS/390 devices (a 3380 volume, for example) to the device managers and files used under OS/2 or AIX to emulate the devices. You will frequently use this program while setting up your system, and installing OS/390. This program is also used for systems running VM and VSE; some of the functions are specific to those operating systems. Use of the configurator is described in more detail in 2.6, "The P/390 Configurator" on page 36.

Another key support area provides console (IPL, stop/start, reset, storage display, and so forth) functions for the S/390 processor. (These functions are

provided through the “Manual Operations” icon in the “P/390” window.) Some of these functions are described in more detail in 5.1, “S/390 Manual Operations” on page 121.

The device managers are used to emulate various mainframe I/O devices. The device managers that are relevant to OS/390 are discussed, in detail, in Chapter 4, “Device Managers and Commands” on page 69.

Device Manager Names

For the P/390 system, most device managers have names beginning with the letters AWS, although a few have other names. For the R/390, there is more variation in the device manager names. The following tables always show the P/390 name, followed by the R/390 name in parenthesis, if it is different. The R/390 configurator panel, described in detail later, uses the same names as the P/390. Internally, it then translates these names to the appropriate R/390 module names. For this reason, we almost always refer to the P/390 device manager names throughout this document. The only time you, as a system administrator, would see the R/390 module names is if you list the contents of the directory containing the modules or if you edit the R/390 IPL or STOP scripts.¹⁷

Do not assume that R/390 and P/390 device managers have exactly the same characteristics because they have the same names. In some cases the equivalent P/390 and R/390 device managers are essentially the same code (within the limits of porting between OS/2 and AIX), but in other cases the underlying code is totally different. Briefly, the device managers include:

- AWSCKD (dmckd) - CKD DASD emulator, used to emulate 3380, 3390, 9345, 3375, 3350, and 3330 disk drives.
- AWSFBA (dasd) - FBA DASD emulator for 3370, 3310, 0671-4, 9332, 9335, 9336-10, FB-512 devices.
- AWSC370 (dmhuron) - S/370 channel adapter manager, used with the S/370 Channel Adapter/A card to attach “real” control units and devices. Only a limited set of control units and devices are usable through this attachment.
- AWSTAPE (dmtape) - 3803 3420/3422 emulator, which uses Server files for tape volumes.
- SCSI3420 (dm34xx) - uses 4mm or other SCSI-attached tape drives to emulate an IBM 3420 drive.
- SCSI3480 (dm34xx) - uses 4mm or other SCSI-attached tape drives to emulate an IBM 3480 drive.
- AWS3420 - is a second copy of the SCSI3420 device manager, permitting the use of a second SCSI-attached tape drive as a 3420 device.
- AWS3480 - is a second copy of the SCSI3480 device manager, permitting the use of a second SCSI-attached tape drive as a 3480 device.
- AWS2821 (printer) - Printer emulator, which emulates an IBM 2821/1403 printer, with output directed to a Server file or to a printer port.
- AWS2540 (dm2540) - Card reader emulator, uses Server files to emulate input from a card reader.

¹⁷ The R/390 module names are different for backward compatibility with a previous system. The module names may be changed in a later release to match the P/390 names.

- AWS3215 (dm3215) - Console keyboard emulator, used to emulate an IBM 3215 typewriter/keyboard. Some stand-alone utility functions (S/390) may require this, and it can be used by VM.)
- AWS3274 (dm3270) - IBM 3274 (non-SNA) control unit emulator, which provides several 3270 sessions on the Server display terminal. For P/390, it uses a special interface from CM/2. For R/390, it uses TCP/IP tn3270 sessions.
- LAN3172 (dm3172) - IBM 3172 LAN gateway (SNA), provides an SNA gateway for external OS/2, DOS, and Windows systems (usually with 3270 emulation programs) to communicate with OS/390. It can also be used for SNA connections to a mainframe or to other SNA devices such as printers. LAN3172 replaces an older device manager, AWS3172. References to AWS3172 are automatically mapped to LAN3172.
- WAN3172 - Provides SDLC connections for VTAM
- LAN3088 - IBM 3088 CTC emulation over a LAN, used only between multiple P/390 (OS/390, VM, or VSE) systems. (Not available for R/390.)
- AWSICA (sdlcdm) - Integrated Communications Adapter (ICA) support for SDLC (not with OS/390) and BSC.
- AWSPBS - Provides BSC and SDLC links using Portmaster adapters
- LAN3274 - Permits LAN (non-SNA) 3270 emulation sessions, using a simple (NetBios or TCP/IP) protocol. These sessions appear as local (coax-attached) 3270 units to OS/390. LAN3274 does not exist for R/390, where AWS3274 provides similar functions.
- LCS3172 (lcs3172tx, lcs3172rx) - Provides an interface similar to a IBM 3172 Channel Station for TCP/IP interfaces.
- MGR3172 - Provides NetView connectivity to emulated 3172 devices running LAN3172 or WAN3172.
- AWSTFA - Transparent File Access (for VM). This permits a P/390 VM user to link and access a mainframe VM minidisk.
- AWS5080 - Provides 5088-like functions using FSLA or MSLA adapters.
- AWS2703 (dm2703) - IBM 2703 communications controller emulator, uses the Server's serial (COM) ports to emulate asynchronous ports on an IBM 2703, for connections (via modems) to ASCII terminal devices. (The 2703 is no longer supported by OS/390.)
- AWSOMA - Reads a CD-ROM in OMA format. To OS/390, this appears as a 3420 or 3422 tape drive.
- AWSPCSR - Uses P/390 VM users to work directly with Server (OS/2 or AIX) files.

There are a considerable number of device managers; the following table may help categorize them:

Table 1 (Page 1 of 2). Device Manager Utilization

Manager	Description	OS/390	VM	VSE	P/390	R/390
AWSCKD (dmckd)	CKD DASD	Y	NR	NR	Y	Y
AWSFBA (dasd)	FBA DASD	N	Y	Y	Y	Y
AWSC370 (dmhuron)	Bus & tag channel	Y	Y	Y	Y	Y

Table 1 (Page 2 of 2). Device Manager Utilization

AWSTAPE (dmtape)	pseudo tape drive	Y(1)	Y	Y	Y	Y
SCSI3480 (dm34xx)	SCSI-attached tape	Y	Y	Y	Y	Y
SCSI3420 (dm34xx)	SCSI-attached tape	Y	Y	Y	Y	Y
AWS34x0	SCSI-attached tape (2)	Y	Y	Y	Y	Y
AWS2821 (printer)	1403 printers	Y	Y	Y	Y	Y
AWS2540 (dm2540)	Emulated card reader	Y	Y	Y	Y	Y
AWS3215 (dm3215)	Typewriter console	NR(3)	Y	Y	Y	Y
AWS3274	local 3270s via CM/2	Y	Y	Y	Y	N
AWS3274 (dm3270)	local 3270s via TCP/IP	Y	Y	Y	N	Y
LAN3172 (dm3172)	SNA over LAN	Y	Y	Y	Y	Y
WAN3172	SDLC connections	Y	N	N	Y	Y
AWSICA (sdlcdm)	SDLC, BSC	N(4)	Y	Y	Y	Y
AWSPBS	BSC, SDLC	N(4)	Y	Y	Y	Y
AWSX25	X.25	N(4)	Y	Y	N	Y
LAN3274	NetBIOS and TCP/IP 3270s	Y	Y	Y	Y	N
LCS3172 (lcs3172tx, lcs3172rx)	TCP/IP on 390 OS	Y	Y	Y	Y	Y
MGR3172	NetView	Y	N	N	Y	N
LAN3088	LAN as CTC	Y	Y	Y	Y	N
AWSTFA	VM file access	N	Y	Y	Y	N
AWS5080	5080 displays	N	Y	N	Y	N
AWS2703 (dm2703)	S/S terminals	N	Y	?	Y	?
AWSOMA	pseudo tape drive	Y(5)	Y	Y	Y	Y
AWSPCSRV	Access Server files	N	Y	N	Y	Y

Table notes: NR means not recommended. (1) AWSTAPE does not implement read backwards commands that are sometimes used by OS/390 standard label processing. (2) AWS3420 and AWS3480 are second copies of SCSI3420 and SCSI3480, permitting two drives of each type to be used. (3) AWS3215 is not recommended for the OS/390 console, but it can be useful for standalone utilities. (4) AWSICA and AWSPBS cannot be used with OS/390 VTAM; they can be used with non-VTAM BSC programs, such as JES2 NJE/RJE, although this is not formally supported. (5) The OMA format is not used for any OS/390 program distributions by IBM.

The multiple communications-oriented device managers can be confusing. The following table may help position them.

Table 2 (Page 1 of 2). Communications Device Managers

Manager	Use	OS/390	VM	VSE
AWS3274 (P/390)	3270s on Server display, only. For OS/390 console and a few TSO and/or CICS sessions. Appears as local, coax-attached 3270s to OS/390.	Y	Y	Y

Table 2 (Page 2 of 2). Communications Device Managers

AWS3274 (R/390)	3270s sessions through TCP/IP (on the Server) and tn3270-type emulators. Used for the MVS master console, TSO, CICS, and so forth. Users can connect through any TCP/IP path. Appears as local, coax-attached 3270s to OS/390.	Y	Y	Y
LAN3274 (P/390 only)	Two connection modes: NetBios and TCP/IP (on the server). NetBios is for OS/2 CM/2 users on the local LAN, and is very simple to set up. TCP/IP is for any tn3270-type emulator. Both modes appear as local, coax-attached 3270s.	Y	Y	Y
LAN3172	Full LAN SNA. Appears as CTC-attached 3172. Full VTAM definitions required. Can use SNA 3270 emulators, NJE, or any other SNA LAN connection.	Y	Y	Y
WAN3172	SDLC connections. Appears to VTAM as CTC-attached 3172 with LAN-like connections. WAN3172 translates real SDLC SNA connections to LAN SNA connections.	Y	N	N
AWSICA	Emulate ICA (1-2 lines). Not supported by OS/390 VTAM. Can be used (VM, VSE) for SDLC or BSC lines.	N	Y	Y
AWSPBS	Emulate ICA (more lines). Not supported by OS/390 VTAM. Can be used (VM, VSE) for SDLC or BSC lines.	N	Y	Y
LCS3172	Full TCP/IP on LAN. Appears as CTC-attached 3172, using two CTC addresses. Provides connection to OS/390 TCP/IP.	Y	Y	Y
MGR3172 (P/390 only)	Connect to NetView, and associated with VTAM resources defined for LAN3172 and WAN3172 device managers.	Y	Y	N
AWS2703	Not supported by OS/390. Sometimes used for dial-in ports to VM.	N	Y	Y

1.6 Adapters and Connections

A number of optional adapters can be used with the systems. The important ones are:

- The S/370 Channel Emulator/A adapter provides a parallel channel (“bus and tag”) connection between Servers and mainframe control units. Only a limited range of control units have been used with this connection. Please see 4.7.1, “AWSC370 Device Manager” on page 106 for more detail. The same adapter is used with both P/390 and R/390 systems.
- Token-ring and Ethernet adapters are supported for emulating an IBM 3172 Channel Station. The (emulated) 3172 can be used in a number of ways, and can provide the primary connectivity link between an OS/390 system and other systems.
- Portmaster adapters provide up to eight SDLC (or BSC, in some cases) lines. Two of these adapters may be used.
- Wide Area Connector (WAC) adapters are used to emulate the integrated communication adapters (ICAs) found with IBM 9221 and 9370 systems.

OS/390 does not generally support these devices through VTAM, but bisync (non-VTAM) can be used with these adapters. These are not used with R/390 systems. (The PS/2 Multiprotocol Adapter is no longer supported, and has been replaced by the WAC.)

- A 4mm tape drive (standard with the system) is used (via a driver program) to emulate an IBM 3420 or 3480 tape drive.
- A CD-ROM drive (standard with the system) is normally used for installing OS/2 programs. In principle, it should be possible to use it to hold read-only emulated 3380 disks (or read-only emulated 3420/3422 tape files), but we did not try this function. The CD-ROM can also be used through the OMA (Optical Media Attach) device manager.
- A SCSI-attached 3480 or 3490 type tape system can be used (with the appropriate driver) as a 3480 tape system (supported by OS/390).
- A SCSI-attached 3420 type tape system can be used (with the appropriate driver) as a 3420 tape system (supported by OS/390).
- A number of different disk drives can be used with the systems.

1.7 OS/390 on CD-ROM

IBM has available two different CD-ROM sets for OS/390. One is known as the *Preconfigured System*, and the other is the *Partners in Development or Application Developers* version. The difference is in the number of program products included with the systems.

OS/390, and all the auxiliary program products that are used with it, are *licensed program products*. You must pay the required license fees to obtain and use the products. Other things being equal, a system with more program products will have higher total license fees than a smaller system. Some program products, from IBM and other vendors, have special license fees for P/390 systems. These are often known as ESL (Entry System Level) prices. In cases where such prices do not exist, a P/390 (meaning both PC Server and RISC/6000 versions) will use the lowest tier price for the software product.

You are not required to use either of the CD-ROM OS/390 systems. However, you are required to have a proper license for any licensed software you install on your P/390 system. Note that this is a license for the software *for use on the P/390*. An existing license for another processor may not be acceptable, unless it is in the nature of a site license. After making the required license arrangements, you could use your existing MVS software, order new OS/390 (or MVS) software, purchase a custom-built (or express-built) OS/390 system from IBM, order the preconfigured CD-ROM, or use some other path to obtain your operating system and associated program products.

The Preconfigured System CD-ROM contains a basic OS/390 system, with no additional program products.¹⁸ It is usable, as is, but would normally be used as a base for adding additional software products. The Open Edition functions are not usable because they require RACF (an additional product) or equivalent.

¹⁸ OS/390 incorporates the base MVS control program, JES2, OpenEdition, DFSMS/MVS, TCP/IP, SMP/E, TIOC, TSO/E, VTAM, LE, ISPF, HLASM, DFSMSdfp, and a number of other products. These produce a minimal, usable system. Key functions not included are RACF, SDSF, and PSF.

IBM provides special P/390 support and terms for bona fide OS/390 (or VM or VSE) software developers. This is through the S/390 *Partners in Development* group (previously known as the *S/390 Developers' Association*). More information is available by telephoning 1-800-627-8363 or 1-404-835-9900 (in North America), or 49-7031-16-2809 in Europe. In other locations, please ask your IBM representative for information.

A special OS/390 CD-ROM is available for members of this association. At the time this was written, the Application Developers' CD-ROM contained the base OS/390 system, plus RACF, COBOL, C, Fortran, CICS, DB2, IMS, PSF, and a substantial list of other products.

The Preconfigured OS/390 CD-ROM has a subset of the addresses generated for the Developers' Association CD-ROM. The full set of addresses is listed here, and all the examples in this document (and other redbooks) use these addresses. The addresses are:

S/390 Addresses	Device Type	
00C	2540R	
00E	1403	
00F	3203-5	
010	3270-X	(These two 3270 devices are relics of the system used to built the CD-ROMs)
063	3270-X	
120-15F	3380	
240-25F	3380	(Reduced from 200-25F range)
260-27F	3390	(Added for OS/390 AD CD-ROM)
560-57F	3480	
580-5AF	3400	(Range extended to 5AF)
5A0-5AF	3422	
700	3270	(OS/390 console)
700-73F	3277	
900-91F	3277	
A80-ABF	3390	
E20-E27	CTC	
E40-E47	CTC	

After installing OS/390 from the CD-ROMs, you can change or add to these addresses by using HCD in the normal manner.

1.8 Positioning and Usage

Experienced MVS customers will naturally compare P/390 and R/390 systems with their mainframes. While the P/390-based systems use "real" mainframe software, OS/390 and a large variety of subsystems and applications, they should not be considered mainframes. Some differences are:

- IBM mainframes have many layers of (hardware) recovery functions, some of which (in newer systems) are very sophisticated. Major mainframes have multiple processors, and transparent recovery mechanisms operate across the complete complex.
- The Servers uses small-system disks (and the P/390 support programs do the necessary functions to emulate mainframe disks).
- Mainframe channels and control units use intricate webs of multiple paths to devices, and devices shared with multiple processors. In later mainframes, this all takes place at a level below the operating system. This sharing of devices has become a fundamental design element of production installations.

- Modern mainframes can be partitioned into multiple systems. This is often important for supporting production, development, and testing on the same mainframe. It is also important for software license management.

Who needs P/390 OS/390 or R/390 OS/390? At the time this was written, P/390 MVS had been available for one year. The range of uses has been surprising; perhaps the most interesting being a large number of UNIX software developers who want to port their products to OS/390. Potential users of P/390 OS/390 include:

- Developers (programming, testing, and so forth)
- Packaged, turn-key “solutions”
- Distributed (departmental) production operations
- “Old Iron” replacement production operations
- Other, more specialized uses

These categories have implications. For example, a development system may be a fairly minimal system, in terms of storage and DASD. It may have LAN connections (for emulated 3270 terminals), but is likely to have no channel-attached devices. It will probably have a current OS/390 system. It may have subsystems, such as DB2 and CICS, but in minimal configurations. Many of the comments in this document are oriented to this type of usage.

A distributed production system is likely to have substantial DASD (for production data), and may have channel-attached devices (especially 38xx printers, 3174 controllers, and so forth). By implication, the OS/390 or MVS system and most program products will be relatively current and may have been ordered/customized just for the smaller system. The system may have a small 37x5 communications controller for remote mainframe attachment, and/or SNA-attached 3172 controllers. It is very likely running CICS and DB2. (Both 3172s and 3174s can be emulated, using LAN functions, so the “real” devices may not be needed.)

An “old iron” replacement system may have a substantial number of internal disks (since mainframe DASD cannot be attached through the current channel adapter). It will have channel-attached devices (through the S/370 Channel Adapter/A), and these are likely to be rather old devices (card readers, punches, 27xx communications controllers, and so forth). The P/390 may be expected to run the existing level of the old system’s operating system, with few (if any) changes.

A P/390 OS/390 system is adaptable to many (but not all) of these types of environments. Limitations are due to (a) not all channel-attached devices can be used, (b) the I/O bandwidth (and DASD access arm independence) is usually less than that of a mainframe configuration, (c) meaning that practical paging rates may be lower, and (d) owners must deal with the Server side of the P/390, as well as the S/390 environment.

The “other” category is endless. For example, the P/390 could be used to demonstrate OS/390 products at a trade show. It is certainly easier to transport and set up than a mainframe, and provides better response times than a remote link. It could be a test system for the “Year 2000 Problem” by simply setting the date in the P/390 to a later date than January 1, 2000.

1.8.1 Selecting a System

We have briefly described three systems:

1. PC Server System/390
2. RISC/6000 S/390 (Model 7012-390)
3. RISC/6000 S/390 (Model 7013-591)

Some thought is needed to select the best base system for your needs. This discussion assumes you intend to run OS/390; considerations for VM or VSE may differ.

Key issues are:

- The 7012-390 is not recommended for OS/390, and is excluded from consideration. The limited number of internal disk bays and adapter slots are the key factors involved here.
- The underlying performance of the 7013-591 processor is better than that of the current PC Server. This permits the device managers to run faster, and improves overall performance. (Device manager overhead, especially for CKD emulation, is often a constraining factor for system performance.)
- The PC Server offers more expandability, without requiring external boxes. In practice, it can house 17 disk drives with 2.25 GB per drive.¹⁹
- The RISC/6000 systems use AIX as the Server operating system. AIX is more complex than OS/2, but may offer better dispatching, SMP support, and more effective use of SCSI disks. However, system administration can be more complex than with OS/2.
- The PC Server system uses OS/2 as the Server operating system. OS/2 is less complex than AIX, and may be easier to administer. However, OS/2 support of SMP hardware is more restricted than that of AIX, limiting performance growth in this direction.
- In practice, running substantial OS/2 applications while OS/390 is also active is not generally advisable. The superior speed and more sophisticated dispatching function of the AIX systems may permit significant AIX applications to run in parallel with OS/390. However, at the time this was written, we were unable to quantify this.
- More device managers, more mature (better tested), are available for the PC Server system. This advantage will change over time, of course.
- The PC Server system costs less than the RISC/6000-based system.

¹⁹ This is true at the time this was written. Disk products offer increased capacity at frequent intervals, and larger drives, in the one-inch height that best suits this Server, may become available soon.

Chapter 2. Configurations and the P/390 Configurator

This chapter describes several typical PC Server System/390 configurations, and several RISC/6000 S/390 configurations, with emphasis on adapters required and adapter slots used. There are many potential configurations possible, and the ones described are merely representative samples.

A later part of this chapter describes the P/390 configurator function. This configurator, in direct and indirect ways, contains parameters related to the Server configuration, and also to the OS/390 I/O definitions.

PC Server System/390 configurations are described in more detail, because we have had more experience with these configurations. At the time this was written, the RISC/6000 S/390 systems were new and we had much less experience with possible configurations.

2.1 Server Workload Considerations (P/390)

Experience has shown us that, if the S/390 portion is heavily loaded, there is little processing capacity left on the OS/2 side of a PC Server System/390, although the PC Server 520 may improve this situation. One reason we assume the OS/2 side is generally busy when running OS/390 on the S/390 side is that CKD disk emulation, always required by OS/390, presents a heavy workload for OS/2.

Furthermore, we have found that high utilization on the OS/2 side sometimes delays emulated DASD response to the point where OS/390 goes into error recovery, thinking that a disk I/O operation has failed. Once OS/390 begins disk error recovery, the emulation process can fail, bringing down the whole P/390 subsystem. One of the reasons is that complex CCWs associated only with error recovery are not fully emulated -- the assumption being that all "real" error recovery is performed under OS/2 by the relevant device manager. We have encountered such situations, even with fairly light OS/390 loads. OS/390 expects certain timing responses from its channels (remember that OS/390 has no special code for P/390 systems -- it assumes it is running with real hardware channels and control units), regardless of the OS/390 workload present.

For these reasons we recommend that you do not plan to run a heavy workload on OS/2 while OS/390 is running. The situation with VM or VSE is different. They normally use FBA disk emulation, which requires much less OS/2 processing, and a parallel workload on OS/2 (such as LAN Server functions) may be feasible.

The situation may be different with RISC/6000 S/390, for two reasons: the processors are faster²⁰, and AIX has a more flexible dispatching mechanism.

²⁰ This means that the effective integer processing speed is faster; this is not necessarily related to comparative processor clock speeds.

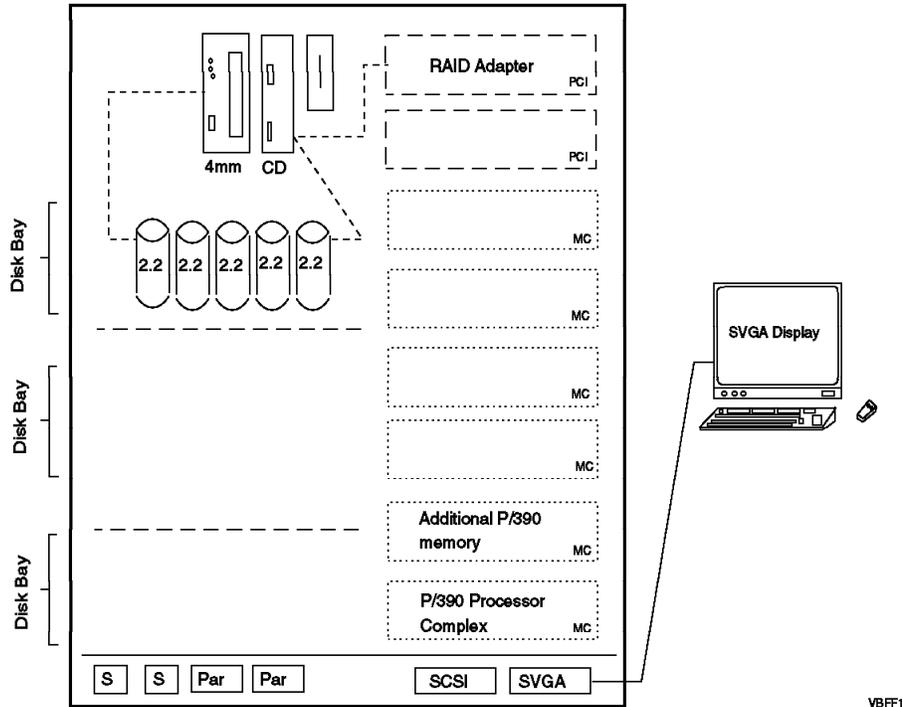


Figure 6. Basic PC Server System/390 Unit. A PC Server 520 is the base unit.

2.2 PC Server System/390 Configurations

Any PC Server System/390 system requires OS/2 as the Server operating system. Additional program products, such as CM/2 (or TCP/IP with a telnet 3270 option, or a future version of PC/3270) are needed. The P/390 support programs and files are always needed. OS/2 and the other programs require disk space. We generally allocate 200 - 250 MB for this purpose. Our most common allocation (in the sense of partitioning disks) is to create a 250 MB "C" drive (for OS/2, CM/2, and so forth), and (assuming a RAID system), a very large "D" drive for the P/390 support programs and emulated 3380 and 3390 volumes. This allocation for C is larger than required, and allows a comfortable amount of free space.

Figure 6 illustrates the basic configuration of a PC Server System/390 for use with OS/390. It contains a RAID adapter and five 2.25 GB disk drives. The RAID adapter (operating in RAID-5 mode) effectively removes the space equivalent of one drive²¹. The RAID adapter has three SCSI channels; one goes to the disks (and the 4mm tape drive and the CD-ROM drive), and the other two are unused in this configuration. The top disk bay will accept only five drives, because the SCSI address normally used for the sixth drive has been used for the 4mm tape drive. (Newer systems may use the internal SCSI adapter for the CD-ROM and 4mm drives, removing this restriction.)

The 2.25 GB disk drives, indicated in the illustration, are the standard units at the time this was written. Lower capacity drives should not be considered for use with OS/390. Larger capacity drives can be used, although there was not a

²¹ .This space is used to store redundant data that permits RAID to survive the loss of any one disk drive.

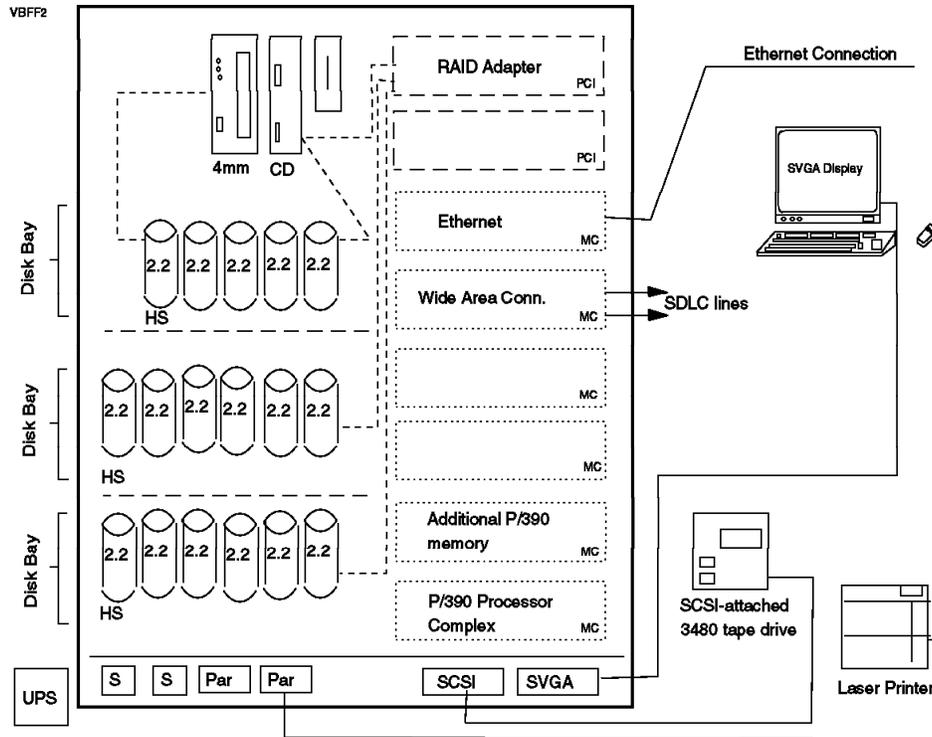


Figure 7. PC Server System/390 Configured for Additional Disks. A PC Server 520 is the base unit.

standard larger one-inch high drive available at the time this was written. All the disk drives in a RAID array should be the same size. If they are not, then all are treated as though they were the size of the smallest unit.

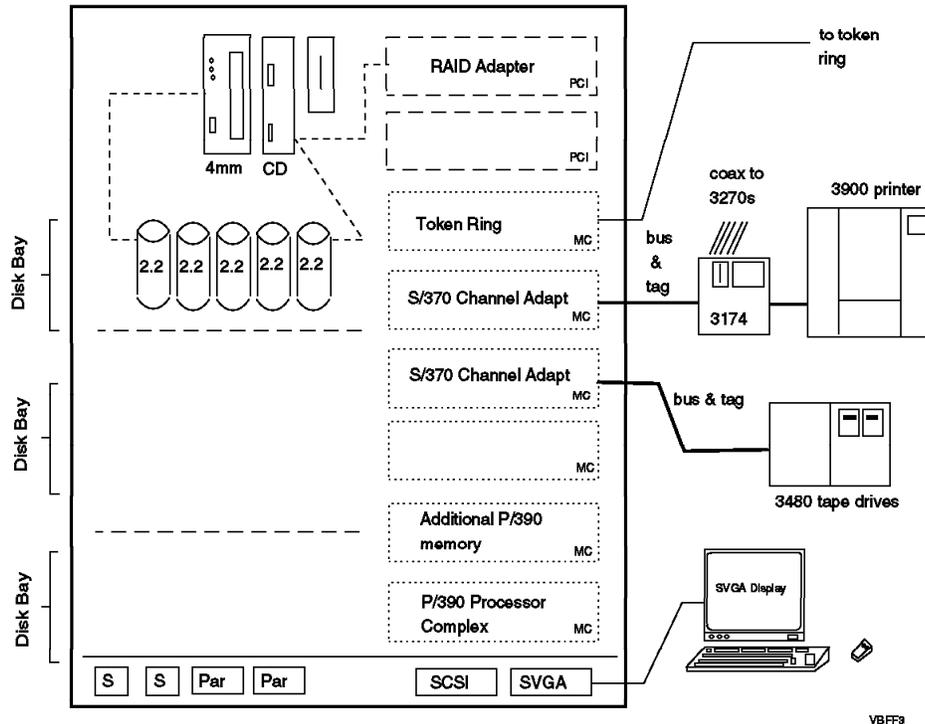


Figure 8. PC Server System/390 With Channel-Attached Devices. A PC Server 520 is the base unit.

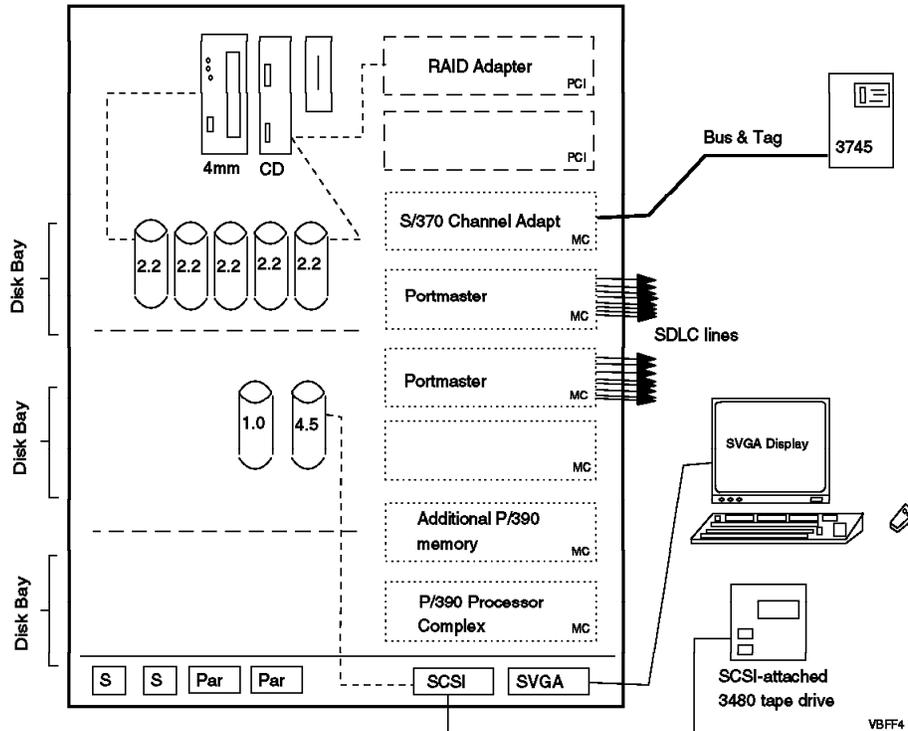


Figure 9. PC Server System/390 With Sixteen SDLC Lines. A PC Server 520 is the base unit.

A system ordered for OS/390 will always include 128 MB of S/390 storage. This currently requires two adapter slots, one for the P/390 adapter and one to provide space and power for the additional storage daughter card.

An adequate SVGA adapter is built on the planar, and no slot is required for it.

The standard PC Server System/390 has 32 MB memory (on the Pentium or OS/2 side), and this has proven to be sufficient. We do not recommend adding additional memory, unless you have specific OS/2-side requirements for it.

Figure 7 illustrates a larger PC Server System/390, with emphasis on availability. All three RAID channels are used, and each channel is defined as an array. Each array has a hot-standby disk, noted by the *HS* symbol in the illustration.

A hot-standby disk normally contains no data. If a data drive in the RAID array fails, the RAID controller will immediately (without operator intervention) rebuild the data from the failed disk on the hot-standby disk. The hot-standby disk does not replace the "extra" disk required by a RAID 5 array for storing redundant data. Thus, in the top bay in the illustration, the effective disk capacity is 3 x 2.25 GB; in effect, the fourth drive is for redundant data and the fifth drive is for the hot-standby unit.

This design requires a number of extra disk drives, but provides two levels of automatic failure recovery. Since the system is configured for high availability, an external UPS (uninterruptable power supply) is shown.

This example includes an external SCSI-attached 3480- or 3490-type tape drive, permitting tape exchange with mainframes. Such tape drives are frequently used with PC Server/390 and RISC/6000 S/390 systems. Tape drives are

discussed in more detail in 2.5, "Tape Configurations" on page 34. An Ethernet LAN adapter is shown, and would probably be used to connect to workstations running 3270 emulators. Two SDLC lines are shown, for wide-area connectivity.

This example also includes a PC laser printer. One of the device managers, AWS2821, can be configured to provide excellent OS/390 output using this device. It is also possible to drive it through PSF and PSF/2; this is discussed in detail in *Printing with MVS on the IBM PC Server System/390*, order number SG24-4612.

Figure 8 illustrates a system with emphasis on connections to mainframe control units. In this case, two parallel channels are used, and a 3174 control unit (for 3270 terminals), a large 3900 AFP printer, and 3480 tape drives are shown. Other control units can be used; there were selected for illustration and because they are typical of the devices that might be considered for use with P/390 OS/390.

The parallel channels are provided by the S/370 Channel Emulator/A adapters. A maximum of two of these can be used. Multiple control units can be chained on each of these channels, although the number of control units and the total length of the cables used are somewhat more restrictive than with mainframe channels. Mainframe disk drives cannot be attached through these channels. See 4.7.1, "AWSC370 Device Manager" on page 106 for considerations about using these channels.

This illustration also reinforces the message that the basic PC Server disk configuration, with five 2.25 GB drives, can be adequate for OS/390 operation.

Figure 9 illustrates a system with emphasis on SDLC communications. It has two PortMaster adapters, each of which can provide eight SDLC lines. A S/370 channel adapter is included. It is shown attached to a 3745 communications controller, but the channel could also continue to include printers or other channel devices.

Two disks are attached to the internal SCSI adapter, and are not part of a RAID array. This was done because the disks are different sizes than those in the RAID array. Disks in a RAID array should all be the same size; there is no restriction about sizes attached to a simple SCSI controller. In the system shown, some disks (in the RAID array) have higher availability than other disks (attached to the simple SCSI adapter). The user would probably place non-critical emulated OS/390 volumes on the SCSI-adapter disks: scratch volumes, paging volumes, perhaps JES2 spool, and so forth.

How many disks (or how much disk space) should you order for your P/390 OS/390 system? This is obviously an open question. How much data do you want to put on the disks? How many DASD volumes do you want to emulate? Do you have enough physical disk arms for your OS/390 workload? Here are a few considerations that may help your planning:

- If you will use an array model of the system (that is, a RAID model), try to order enough disks in advance. Extending a RAID array (adding additional disks) requires you to redefine the array, and with some RAID adapters you lose all the data in the process. (You would take a backup first, of course, but this may take some time for a large disk array.)
- You can add disks to an array model, and define a new array to use the new disks. This will avoid the problem of saving and restoring all your disk data,

but you will now have multiple disks (as seen by OS/2) because each array is seen as a (large) OS/2 disk.

- It is possible to have a one-disk array (using RAID 0). This is not common, but might be needed when extending a configuration.
- The RAID SCSI adapter cannot be used to attach disk drives that are not part of an array. It can be used to attach non-disk SCSI devices (which are not part of an array, of course). The RAID adapter has an external connector that can be used for disks (in a RAID array) or non-disk devices. This external connector parallels one of the internal connectors; you can use the internal disk path, or the external connector, but not both. There is a maximum of seven devices per channel.
- It is possible to have both RAID and non-RAID SCSI adapters in the same server.
- All disks attached to a RAID SCSI adapter should be the same size. If they are not, all will be used as though they were the size of the smallest disk. For example, if you have four 2GB disks and one 1GB disk, the RAID adapter will treat them as five 1GB units (if they are all defined into the same array).
- Other things being equal, a non-RAID disk may be faster but may exhibit slightly lower reliability. RAID (meaning RAID 5, the mode that is usually used) always requires access to two disks for a write operation (including OS/390 paging and spooling, of course). Trying to generalize about RAID versus non-RAID performance is difficult. In some cases, RAID might be faster due to the automatic spreading out of files across all disks in an array.
- The system normally requires the use of special “hot swap” holders (“trays”) for disks. (Disks purchased with the system are already mounted in these trays.) It can use several different models of disks, but all must be mounted in these trays.²² You can buy empty trays and mount your own disks. There are two models of trays, for “narrow” 8-bit disks and for “wide” 16-bit disks. *Be certain your disks are among those supported by the PC Server (see the announcement marketing materials).* The disk connectors must match the connectors in the appropriate tray. Order empty trays at the same time the system is ordered.
- The system has three shelves (bays) for disk drives. The first bay can accept five or six one-inch disks. (One slot may be lost “lost” to the 4mm tape drive SCSI interface.) If half-high disks are used, only three will fit in the bay. The second and third bays will each accept six one-inch disks, or three half-high disks, or a mixture.
- To use the second and third bays you must add one additional power supply, one (or two) backplane kits. (Read the information supplied with your IBM PC Server System for specific information about upgrades. Some models may have one extra backplane and the extra power supply already installed.)
- The 1.12GB and 2.25GB disk drives marketed with the IBM PC Server are one-inch units. The 4.5GB drive is half-high. Slightly older disks (1GB and 2GB) are usually half-high.²³

²² Once you acquire or convert disks using the hot-swap trays, you will find they are convenient and easy to use. They permit the installation of a large number of disks in a modest space.

²³ The author’s unit has one one-inch 1GB drive and two half-high 2GB units on the first shelf. No additional drives will fit in that bay.

- If the first bay holds five 2.25GB drives, this provides 11.25 GB total space; using three 4.5GB drives provides 13.5GB total space. Ignoring price, five disks may be better than three disks because there is more freedom of disk arm movement. Use of RAID will change this storage discussion, of course. The numbers uses here are for non-RAID models. With RAID 5, you will lose the equivalent capacity of one physical drive for each RAID 5 array. In this case, you are left with 9GB effective space for either 5x2.25 GB or 3x4.5 GB configurations. Using six 2.25 GB drives, with RAID 5, provides the maximum effective disk space.
- The Application Developer's CD-ROM OS/390 Release 1 system requires the following space:
 1. 2.8 GB for the OS/390 IPL volume (3390)
 2. 2.8 GB for a DLIB volume (3390) (if you decide to install the DLIBs)
 3. .2 GB for an SMS-managed volume for HFS files (mini-3380)
 4. 1.2 GB for the volume containing paging, spooling, catalog, PARMLIB, and so forth (3380)
 5. 1.2 GB for CICS, DB2, IMS (3380) (if you decide to install this volume)
 6. Perhaps .8 GB for TSO and scratch volumes (mini-3380). These are not included with the CD-ROM, but you are likely to want to create them.

This is 9GB, or 5GB if you do not install the DLIB and DB volumes. You can have a reasonable, basic OS/390 system by filling the first disk shelf with 2.25GB drives.

2.3 RISC/6000-591 S/390 Configurations

An R/390 system requires disk space for AIX, the P/390 support programs, other programs (such as 3270 emulators), and the emulated S/390 volumes. AIX partitions disks by using logical volumes and multiple file systems. Approximately 400-500 MB is required for base AIX (including normal workspace for /tmp and similar files). See 3.3, "RISC/6000 S/390 OS/390 Installation" on page 50 for more information about suggested disk layouts.

Figure 10 on page 30 illustrates the basic RISC/6000-591 S/390 configuration. This base system has four disk drives, with 2.2 GB each, attached to a SCSI adapter. A 4mm tape drive (required, for practical OS/390 usage) and a CD-ROM drive are also connected to the SCSI adapter. The frame can hold a maximum of six disk drives. A fast/wide SCSI adapter has two channels, and each channel can control six devices. One of the channels can be attached to external SCSI devices (in which case that channel cannot be attached to any internal SCSI devices).

One LAN adapter is shown. Only one TCP/IP "stack" can use a LAN adapter at any one time. AIX must have a functional TCP/IP system (to provide a telnet session for OS/390 consoles and TSO users). In this configuration, it would not be possible to use OS/390 TCP/IP because there is no LAN adapter for it.

This configuration leaves three free Micro Channel slots for future adapters.

This base system includes 64 MB memory, but 128MB is recommended for optimum S/390 performance. The improvement is due to AIX using the extra memory as a disk cache. This improves the performance of CKD emulation, and we have seen up to 30% improvement in selected workloads.

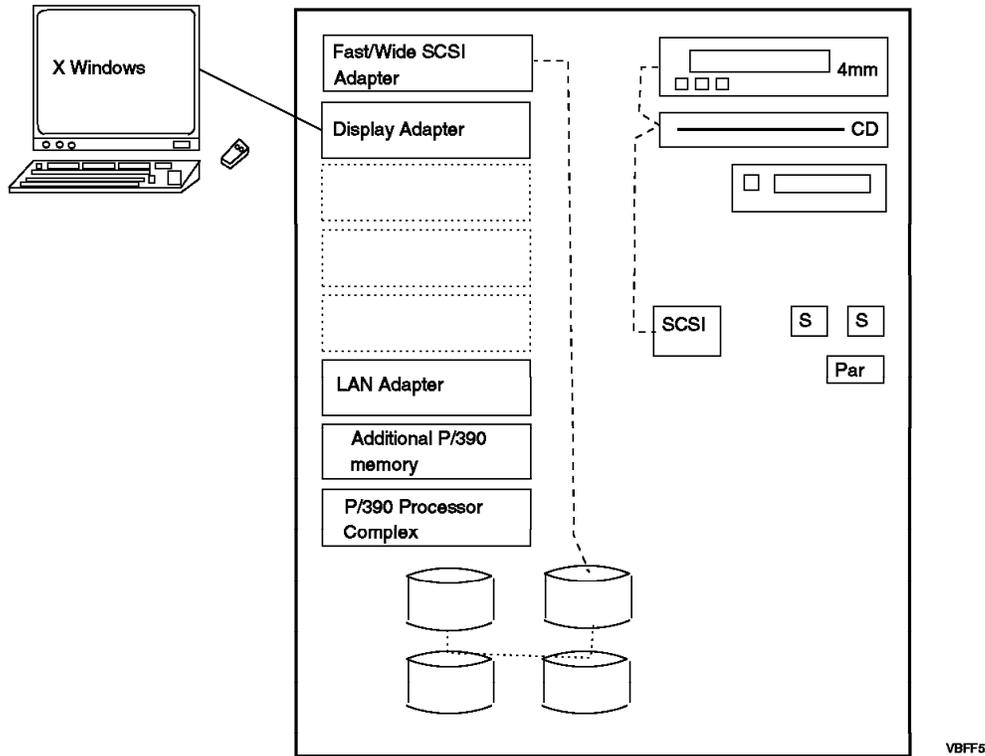


Figure 10. RISC/6000-591 S/390 Basic Configuration. There is not a “standard” configuration for a basic OS/390 system. This illustrates an entry configuration that would serve for basic OS/390 usage. The disk drives would probably be 2.25 GB units, although you have a choice of drives. See Chapter 3, “Installation” on page 43 for a discussion of the amount of disk space needed.

Figure 11 illustrates a larger system. In particular, it uses an external IBM 7137 Disk “Tower,” containing up to eight 2.2 GB drives operated as a RAID array. These towers use differential SCSI attachments (permitting longer cable lengths), and require a differential SCSI adapter. A S/370 Channel Adapter/A is also shown (the same as used with the PC/Server System/390, but with different driver code).

An external SCSI-attached 3480/3490-type tape drive is attached to the standard SCSI adapter. This permits direct interchange of tape volumes with most mainframe systems, and would be a typical device for P/390 and R/390 systems. See 2.5, “Tape Configurations” on page 34 for more information about external tape drives.

Two LAN adapters are shown, permitting both AIX and OS/390 to use TCP/IP connections.

2.4 3270 Connection Configurations - Overview

Any practical OS/390 system requires 3270 terminals -- for the system console, for TSO users, for CICS control and many CICS applications, and so forth. There are a number of ways to provide 3270 terminals for P/390s, and these are partly summarized in Figure 12 on page 32. The R/390 connections are slightly different, and are described later. Using Figure 12 on page 32,

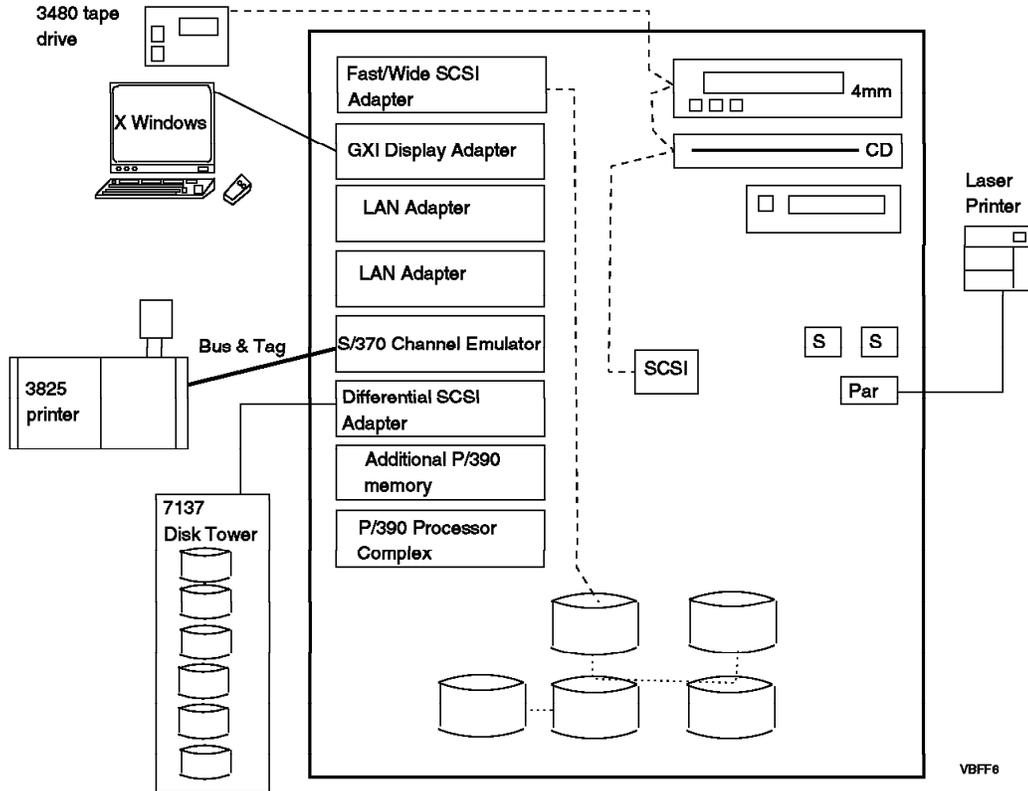


Figure 11. RISC/6000-591 S/390 Larger Configuration

1. The first method uses AWS3274 and applies only to 3270 sessions in the PC Server; that is, 3270 windows on the OS/2 display of the PC Server System/390. The *coax UCBs* notation for this method (and several other methods) means that OS/390 views these terminals as being coax connected through a channel-attached non-SNA 3174 control unit. This is the most basic method of attaching 3270 terminals, and is often used for OS/390 consoles. CM/2 recognizes a *LOCAL* attachment mode, in which it passes 3270 data streams through an API that talks with the AWS3274 device manager. *The 3270 emulator functions that use this connection are being withdrawn from CM/2. The P/390 developers intend to use the same connection mode with a future change to PC3270.* This is the easiest method for providing several 3270 sessions for OS/390. OS/390 must have a UCB for each terminal, and the P/390 configurator must have a line for each terminal. (Each 3270 emulator window is a “terminal.”)
2. The LAN3274 device manager accepts TCP/IP connections, and emulates coax-connected 3270 terminals. The TCP/IP connection uses 3270 telnet protocols. In order to use this connection, you must install TCP/IP on the Server. *A given LAN adapter can be used for only one TCP/IP product. You cannot have OS/390 TCP/IP and Server TCP/IP use the same LAN adapter port.*²⁴ Once the Server TCP/IP is installed, you can connect to it from a 3270 emulator in the Server, or from anywhere on the connected LAN. Any *telnet* emulator can be used. Again, these emulated 3270 sessions appear as

²⁴ Some LAN adapters have multiple ports, and different TCP/IP products could be running on different ports.

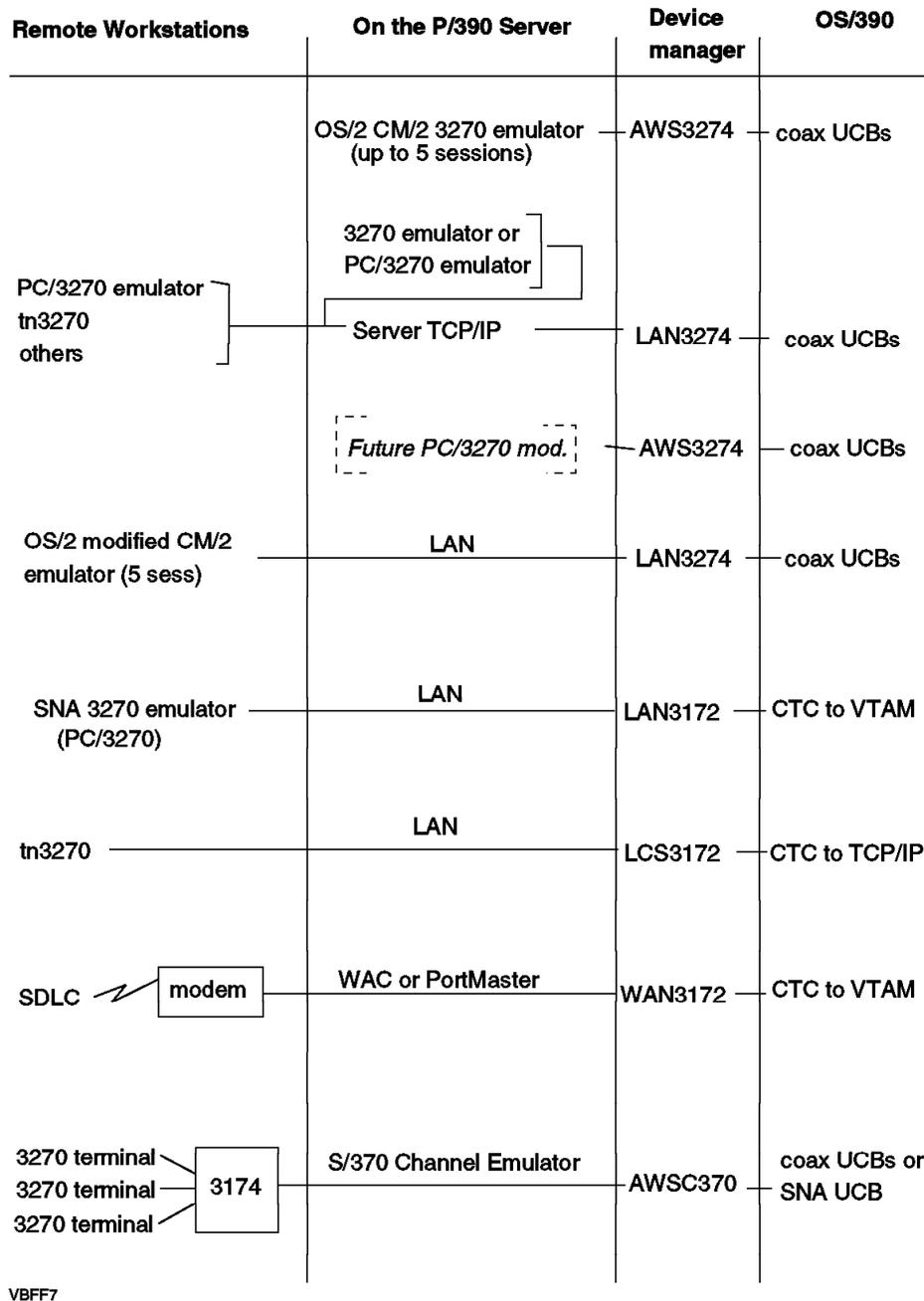


Figure 12. Methods of Connecting 3270 Terminals to the P/390

local, coax-connected terminals to OS/390. An OS/390 UCB and a P/390 configurator line are required for each emulated session.

- IBM developers have discussed a modification to the PC/3270 product, such that it can work directly with the AWS3274 device manager -- very much like the CM/2 connection shown at the top of the figure. This function was not available, at the time this was written, and was being investigated as time was available.
- The P/390 support programs include a modification for CM/2 that produces the next connection method in the drawing. It assumes workstations with OS/2 and CM/2 (plus a simple module replacement in CM/2) are connected

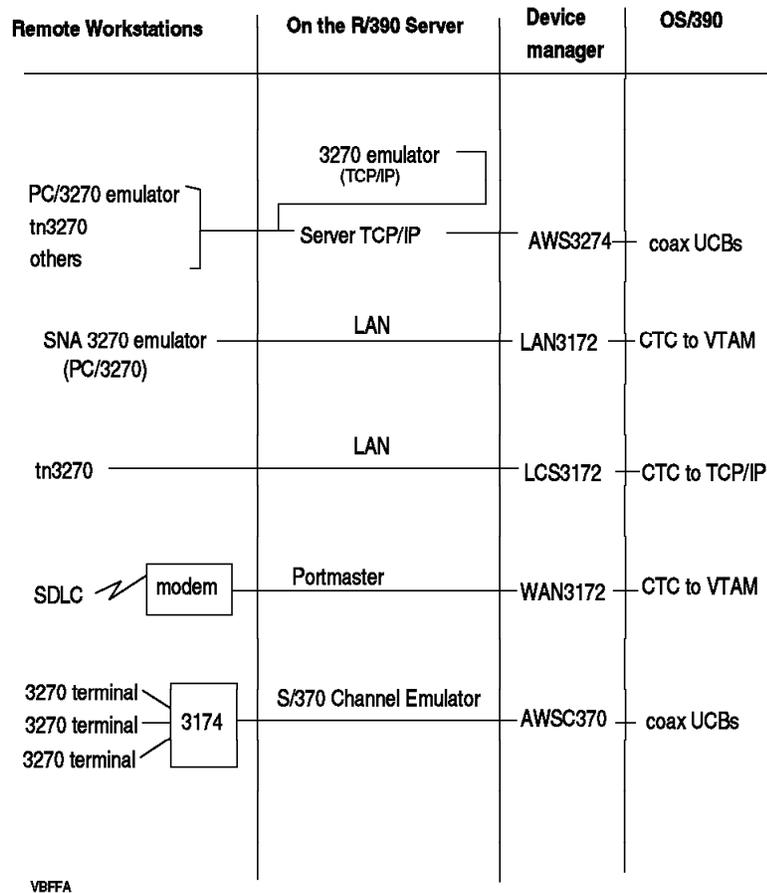


Figure 13. Methods of Connecting 3270 Terminals to the R/390. Note that the device manager names shown are the generic names used in the configurator, not the actual R/390 module names. See 1.5, "Support Programs" on page 14 for a brief discussion of device manager naming.

to the Server via a local LAN. NetBios is used by the LAN3274 device manager and the modified CM/2. This connection works only with local LANs, since NetBios is not routable in complex networks. Again, the terminals appear to be local, coax-attached terminals to OS/390. OS/390 must have a UCB for each terminal session (a CM/2 user can have up to five 3270 emulator sessions) and there must be a P/390 configurator line for each 3270 terminal (session or window). *This method will soon be obsolete, as the 3270 emulator functions are being withdrawn from new releases of CM/2.*

- The LAN3172 device manager emulates the functions of an IBM 3172 LAN station used for SNA connections. It is connected to OS/390 through a CTC (channel-to-channel) interface (also known as a 3088 interface), and is used through normal VTAM functions. There is one CTC address, regardless of the number of 3270 sessions connected this way, and only one line in the P/390 configurator. This device manager uses a Server LAN adapter (token ring or Ethernet) for general LAN connectivity. It works with SNA 3270 emulators on the LAN. There is no specific limit to the number of sessions supported, but there must be VTAM definitions for each session. The VTAM parameters are exactly the same as for a "real" 3172 unit on a mainframe. The use of the LAN adapter can be shared with other Server programs and with TCP/IP (but only one TCP/IP).

6. The LCS3172 device manager emulates the functions of an IBM 3172 LAN station used for TCP/IP connections. In this case two CTC addresses are used, regardless of the number of telnet 3270 sessions involved. Two lines are required, with consecutive unit addresses, in the P/390 configurator table. An OS/2 LAN adapter (token ring or Ethernet) is used for external connections. OS/390 TCP/IP must be installed and operational. This use of a PC Server LAN adapter is mutually exclusive with any use by the Server TCP/IP on the same adapter.
7. The next method, using WAN3172, is for remote (via modem) use. A WAC (Wide Area Connector, two lines, P/390 only) or a PortMaster (eight lines, P/390 and R/390) adapter is used. The P/390 support programs will accept dial-in connections, but cannot generate dial-out connections. Leased lines can be used.
8. The last method shown uses a S/370 Channel Emulator/A adapter, with bus and tag cable, to connect to a 3174. This, in turn, connects to 3270 terminals via coax cables. If a non-SNA 3174 is used, there must be one OS/390 UCB and one P/390 configurator line for each terminal. If an SNA 3174 is used, there is only one OS/390 UCB and one P/390 configurator line, regardless of the number of terminals. Experience has shown that SNA 3174 connections, via the channel emulator, are not reliable, and we do not recommend this usage.

The 3270 connections for R/390 systems are shown in Figure 13 on page 33

1. In the R/390, the AWS3274 device manager uses a TCP/IP interface to either local or remote tn3270-type emulators.²⁵
2. The LAN3274 device manager does not exist for the R/390.
3. The other device managers have the same functions as for the R/390.

2.5 Tape Configurations

You have several options for using tape drives with P/390 OS/390, as illustrated in Figure 14 on page 35:

1. Use a channel emulator adapter to connect to a "real" IBM 3480/3490 or IBM 3803 control unit. These control units have multiple channel interfaces. The channel adapter would represent just another channel attached to the control unit. Of course, the Server must be within channel-attachment distance to use this option. This is the least expensive way to use 3480/3490 or 3420 tapes (provided you already have the tape drives, of course). It tends to be slower than the expected speed of the tape units (due to the current channel adapter), but performance is still acceptable (unless you have small, unblocked records on tape). This option has the advantage that the whole string of tape drives attached to the control unit is available to the OS/390 system. Please read 4.7.1, "AWSC370 Device Manager" on page 106 for considerations about using channel-attached devices.
2. Purchase one or two SCSI-attached (external) 3480/3490-type tape drive, and attach them to a SCSI adapter in your system. For the P/390 this requires a

²⁵ This is almost the same function provided by one of the LAN3274 options on the P/390, and is the only case in which P/390 and R/390 device managers with the same name (AWS3274) have different functions on the two systems. The thought was that AWS3274 should always represent the most basic 3270 connection, as used for the OS/390 console, for example.

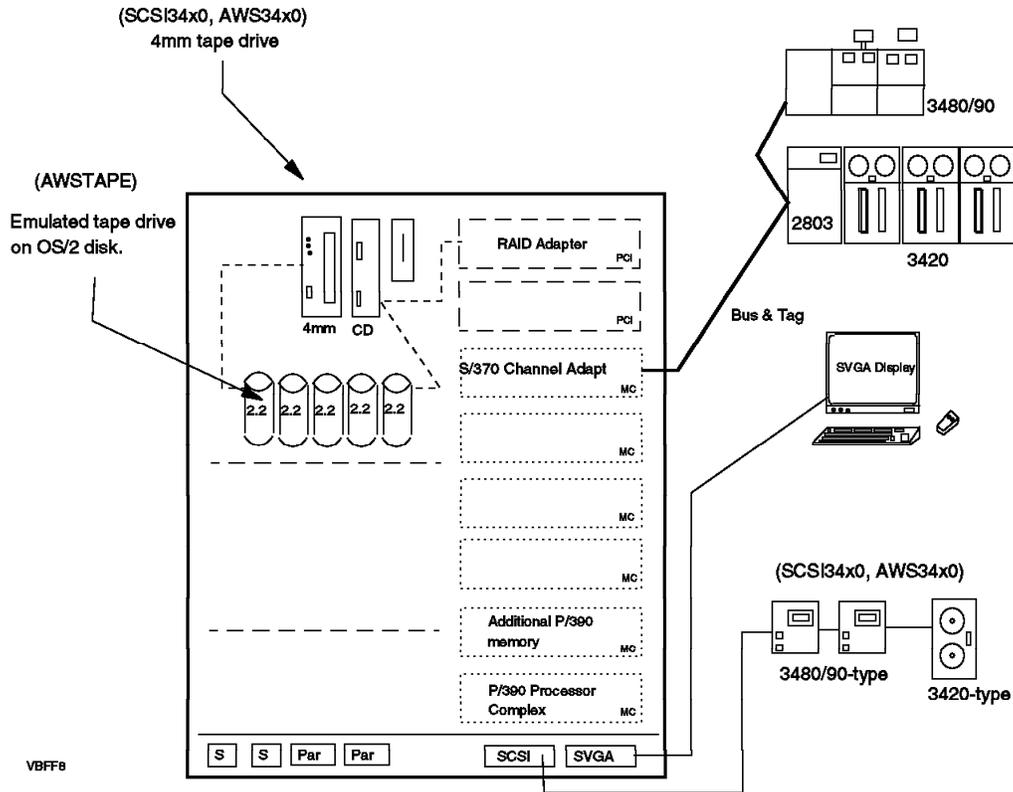


Figure 14. Ways to Configure Tape Drives. The device manager names associated with each mode of attachment are shown in parenthesis.

“single-ended” SCSI adapter in the tape drives.²⁶ IBM does not market this type of unit. We used tape drives from Overland Data, Fujitsu, and other vendors. A drive such as this is very convenient because it can interchange tape cartridges with any host OS/390 system, and has reasonable performance. These drives are relatively small, and are usually placed on top of the Server. If you do not have jobs that require more than one or two tape drives, this is an attractive option. The SCSI3480 device manager is used for the first such drive, and the AWS3480 device manager is used for the second drive.²⁷ The R/390 versions of SCSI3480 and SCSI3420 each supports more than a single SCSI drive. IBM does market a 3490E drive with a differential SCSI interface.

3. Purchase a SCSI-attached (external) 3420-type tape drive. The SCSI3420 device manager is used for these drives. A second 3420-type drive can be emulated with the AWS3420 device manager.

The SCSI3420 device manager can also be used with 3480/3490-type tape drives, and the SCSI3480 device manager can be used with 3420-type tape drives. See 4.4, “Device Managers for Tape Drives” on page 82 for more discussion.

4. Use the 4mm tape drive that is included with your system. Device manager SCSI3420 or SCSI3480 is used with the unit. Performance is good, but there

²⁶ The RAID adapter in the PC Server System/390 provides an acceptable external SCSI interface. You can use a differential SCSI tape drive if you purchase a differential SCSI adapter for your system.

²⁷ If you need more than two SCSI-attached 3480/3490 drives, see 4.4.3, “ISITAPE Device Manager (P/390 Only)” on page 89

is no compatibility with host systems. There is compatibility with other P/390 OS/390 systems using the same option, of course. Again, there is normally only one 4mm tape drive with a system. Many IBM products (and service) are available on 4mm tape, and IBM service will accept dumps on 4mm tapes.

5. Use emulated tapes, that are emulated using Server files and the AWSTAPE device manager. This may be suitable for work tapes, or for “tapes” that are retained for a while.

While performance is poor, the emulated tape can even be on diskette when using a P/390. A diskette has some obvious limitations, but it may be usable where tape processing is designed into an application, but the total amount of data involved) is relatively small.

With the P/390, you can have up to four SCSI tape drives. One (4mm tape) is included with your system. You can add more; they can be 4mm drive, 9-track drive, or 3480/3490 type drives. (You can also use 8mm drives with the R/390, and its SCSI34x0 device manager can support multiple SCSI drives.) There are four SCSI tape device managers: SCSI3480, AWS3480, SCSI3420, and AWS3420. Each device manager is used with only one device for the P/390, and this creates the limit of four SCSI tape drives. Of the four, two can emulate 3480 drives and two can emulate 3420 drives. The OS/390 UCB (address) associated with each drive must be the proper device type: 3480 or 3420. Do not use a 3490 UCB; it will not receive the correct sense codes from the SCSI3480 device manager. If you need more SCSI-attached tape drives than possible within these limits, see 4.4.3, “ISITAPE Device Manager (P/390 Only)” on page 89.

3480 emulation means that the device manager will respond to the additional device commands available on these drives. 3420 drives have fewer device commands. In general, OS/390 application programs can use the two device types interchangeably. Ignoring questions about data compression, a 4mm tape written with the SCSI3480 device manager can be read using the SCSI3420 device manager and vice versa.

2.6 The P/390 Configurator

The P/390 Configurator is an OS/2 or AIX program. Its primary purpose is to maintain a table (stored in a Server file) that connects S/390 device addresses to P/390 device managers. The basic table might look like this, for a PC Server System/390 system:

S/390 Address	Device	Device Manager
120	3380	AWCKD.EXE
121	3380	AWCKD.EXE
560	3480	SCSI3480.EXE
700	3270	AWS3274.EXE
701	3270	AWS3274.EXE
A80	3390	AWCKD.EXE

Using a table such as, if OS/390 issues an I/O instruction (such as SSCH) for address 120, the P/390 channel emulator will give control to device manager AWCKD. The device manager then reads the CCWs from S/390 memory (via the channel emulator) and emulates 3380 I/O using Server files. If the I/O operation is for device address 700, the AWS3274 device manager is called to emulate a 3270 terminal (in a window on the Server’s display). In this example,

note that device manager AWSCKD is used several times, and for different types of devices. Some device managers can appear multiple times in the configurator, and some (such as SCSI3480) cannot be used more than once.

The channel and control unit emulation is device-by-device. There is no sense of a *string* of disk drives or tape units. The channel addresses, such as 120 for a 3380, are completely arbitrary. You could have a system with three emulated 3380 drives, at addresses 001, 6AF, and EE3. Of course, your OS/390 must have UCBs generated (via HCD) at the corresponding addresses. If you have mainframe devices attached through a S/370 Channel Emulator/A adapter, these would retain their original string of addresses, if appropriate.

The configuration table is stored in a DEVMAP file. This is a normal Server (OS/2 or AIX) file. It can have any name, but the convention is to name it DEVMAP.xxx, where xxx is used to differentiate between several DEVMAP files. For example, you might have DEVMAP.MVS, DEVMAP.VSE, DEVMAP.TST, and so forth. The *current* DEVMAP is used, by default, for most P/390 functions. The *current* DEVMAP is changed with the AWSPROF command; for example:

```
D:> AWSPROF D:\DEVMAP.MVS (make DEVMAP.MVS the current DEVMAP)
```

```
D:> AWSPROF (No operand. Will display current DEVMAP name.)
```

For OS/2, you *must* enter the leading backslash, even if the file is in a root directory. For both OS/2 and AIX, you must supply the full path name. The configurator program (and the IPL process used to start the P/390 subsystem and OS/390) always use the *current* DEVMAP. If you want to change it for some reason, you must remember to use the AWSPROF command. The name of the current DEVMAP is retained across Server startups.

The complete installation process is outlined in Chapter 3, "Installation" on page 43. Once the P/390 subsystem is installed, you need to access the primary P/390 window in order to perform most P/390 functions. For OS/2, this is done by double clicking the P/390 icon on the desktop. For AIX, it is done by starting X Windows and then entering the command r390 in an AIX window.

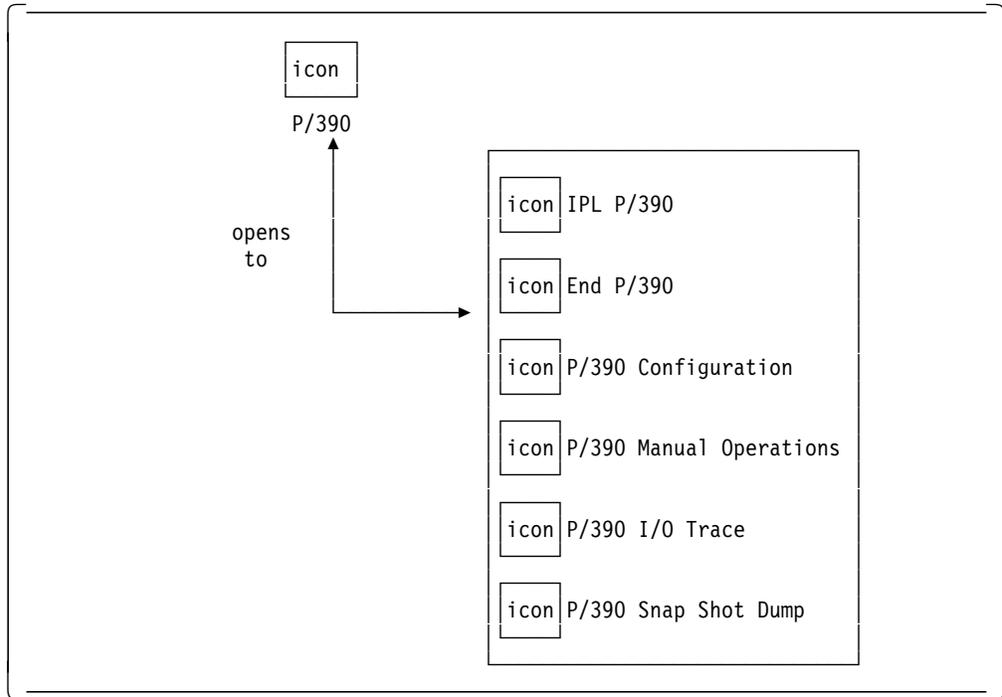


Figure 15. Base P/390 Window. Practically all P/390 administration starts from this window. The equivalent window for the R/390 is obtained by entering the command `r390` after starting X windows.

To use the P/390 configurator, open (double click with OS/2, single click with AIX) the P/390 Configuration icon. This should produce a "Configuration Password" panel such as the one shown here:

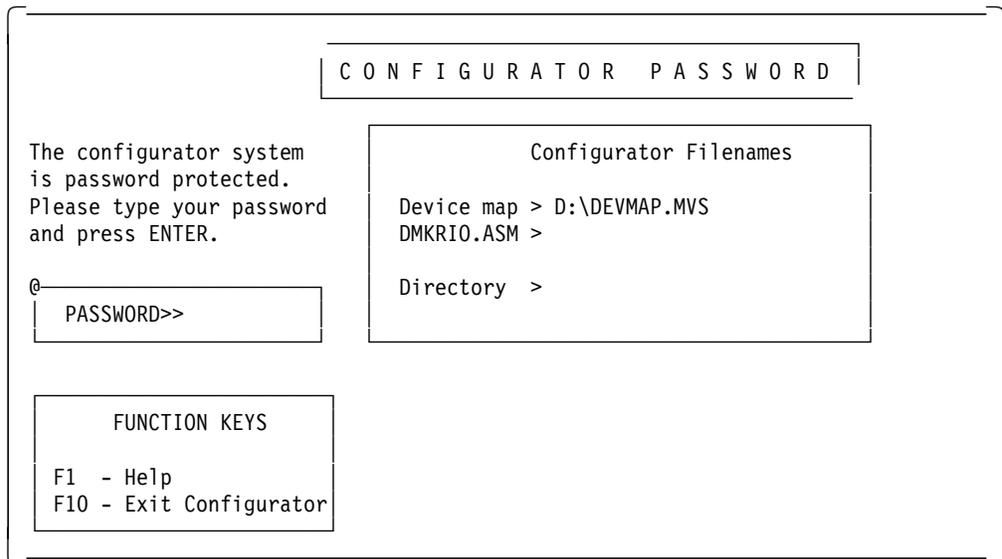


Figure 16. Configurator Password Panel. Typically, no password is set. Simply press Enter to proceed to the next screen.

By default, there is no password. Just press Enter. Before doing this (the first time you use this panel), check the upper right window, for Configurator Filenames. If there is anything in the "DMKRIO.ASM" or "Directory" lines, erase it. These parameters are for VM, and OS/390 will not work properly if these two

parameters exist. You cannot directly change the "Device map" parameter; you would need to exit from the P/390 subsystem and use the AWSPROF command to change it.

You cannot use a normal text editor with the DEVMAP files. If you alter a DEVMAP file with a text editor, it will probably not work afterwards. Only the P/390 configurator program should be used to alter DEVMAP files, and only the configurator can view a DEVMAP file in a reasonable format. (A utility, DEV2NAME, also can be used to print a DEVMAP in a reasonable format.)

Changes to DEVMAPs are detected only when the P/390 subsystem is started. For example, using the IPL icon will start (or restart) the P/390 subsystem and read the current DEVMAP. Changes made to DEVMAP while OS/390 is running (or any other S/390 system is running) are not activated until the P/390 subsystem is restarted. This always requires a reIPL for the S/390 operating system being used.

Press Enter in the password panel. This should produce the next panel:

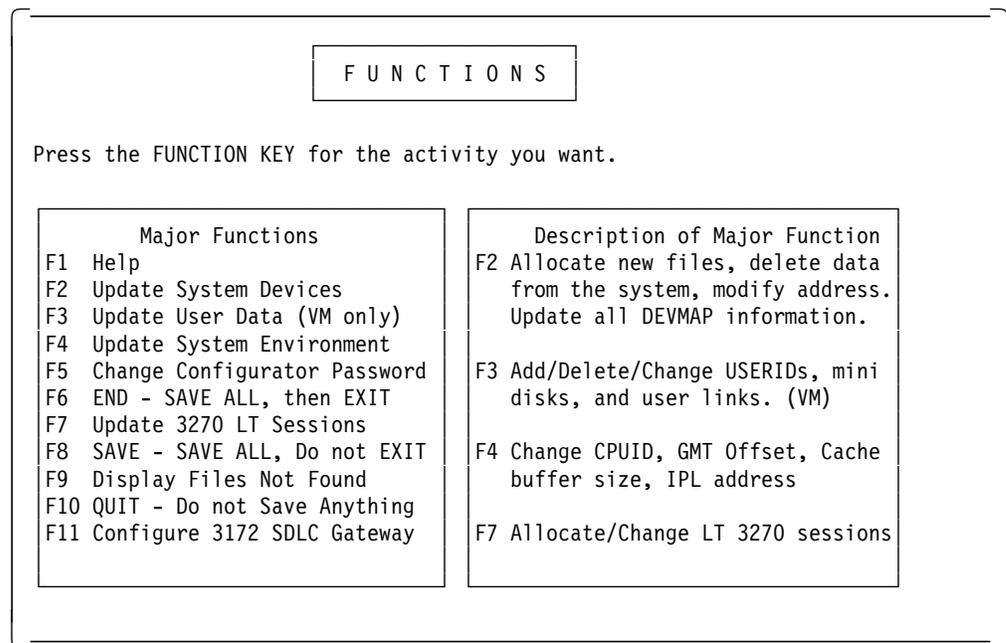


Figure 17. Main Menu of Configuration Program

Function key F2 produces the "Update System Devices" panel where all your emulated S/390 devices are defined. You will probably update this data many times, and you should devote some time to understanding this panel. It is the single most important panel in the P/390 support system. This panel displays (and alters) data in the *current* device map file. The *current* device map file is specified with the AWSPROF command. If you install one of the CD-ROM OS/390 systems, following the normal instructions, your initial device map file is D:\P390\DEVMAP.MVS for OS/2, or /mvs/devmap.mvs for AIX. Your display should look something like the following:

```

UPDATE SYSTEM DEVICES

Available disk space in Kbytes: 47,123

@
> Addr Device Label Atype Size Mgr FN/P PATH=D:\P390
> > > > > >
00E 1403
120 3380 MVSRS5S CKD 1770C 2 E:\MVSRS5.120
122 3380 SCPMV5 CKD 885C 2 E:\SCPMV5.122
580 3420 G D:\DFDSS.IPL
581 3420 G D:\ICKDSF.IPL
700 3278 3 MVS Console
701 3278 3 TSO Terminal
B71 3480 N

MGR Codes:1=AWSFBA 2=AWSCKD 3=AWS3274 4=LAN3274 5=AWS3215 6=AWS2821
7=AWS2540 8=LAN3088 9=LAN3172 A=LCS3172 B=MGR3172 C=WAN3172 D=AWS2703
E=AWSICA F=AWSPBS G=AWSTAPE H=SCSI3480 I=SCSI3420 J=AWSOMA K=AWS9346
L=AWSTFA M=AWSPCSR N=CHAN370 O=AWS3480 P=AWS3172

F1 Help ALT+F1 Key Definitions F10 Main Menu ESC No Change

```

Figure 18. Main Configuration Panel

The bottom lines of the display list various P/390 device managers and assign a single character code to each one. For example, code 2 is AWSCKD (the CKD disk emulator). This technique is used to save space in the line displayed for each emulated address. The actual device manager name, rather than the single-letter code, is saved in the DEVMAP file. The single-letter codes have no intrinsic meanings, and may change when new device managers are added.

You can use F1 to obtain help information about any of the device managers. Move the cursor to the device manager name, and press F1 to display the DOC file corresponding to that device manager.

The central box in the display is the area for all your configuration work. (In the remainder of this document, we will usually reproduce lines only from this central box.) You must use the first line (in the central box) to enter a new line. ALT+F1 displays a large list of function keys. The most important are F9 to delete a line (position the cursor on it first), ALT-F3 to turn off a device (but leave it in the table), and ALT-F5 to turn on a device.²⁸ The R/390 uses the CTL key instead of the ALT key in these multiple-key combinations.

This table contains one line for all I/O devices that your OS/390 system can access. In one sense, it is like an IODF or IOCDS on a mainframe; it *defines* the I/O system. Two columns are critical: the "Addr" column contains the (hex) address of the emulated device, and the "Mgr" column indicates which device manager should be called when OS/390 attempts to use this device. In the figure, for example, 700 is associated with device manager 3 (which is AWS3274). This emulates a local (3174-attached) 3270 terminal. In our case, this is the primary OS/390 console.

²⁸ The "off" or "on" function applies to the device where the cursor is positioned, and all devices immediately (contiguous) below it that use the same device manager.

The other columns may, or may not be relevant, depending on the device manager involved. See Chapter 4, “Device Managers and Commands” on page 69 for specific information about various device managers. You should make an entry in the “Device” column; the DASD device manager checks this value to determine device geometry and sense format. The “Label” column is unused (in an OS/390 environment)²⁹; however, we suggest you enter disk volser names here as a matter of documentation. The “Atype” column also appears to be unused in an OS/390 environment. The “Size” is used for emulated DASD, and is explained in detail in 4.2.1, “AWSCKD” on page 70. The “FN/P” field is used for emulated DASD and tape drives, and contains the OS/2 file name used for the emulated device. Do not place comments in this field; use it only if the specified device manager expects operands in this field.

In the example, there is a device at address B71, managed by the CHAN370 device manager. This is a 3480 tape drive that we attached through the S/370 Channel Emulator/A adapter. Devices 580 and 581 are defined as emulated tape drives, each using an OS/2 file in place of a tape drive. In this case, the “FN/P” field points to the OS/2 file. These files, which contain stand-alone utilities, are IPLed directly.

In the example, two disks are defined at S/390 addresses 120 and 122. The size fields indicate the number of cylinders on the emulated 3380 volumes. These are written in the indicated format: nnnnC. The FN field contains the name of the OS/2 file used to hold the emulated 3380 drive. You can use any file name, but we suggest a naming convention so that you can correlate the OS/2 name to the S/390 drive or volser.

The configurator can associate OS/2 files with tape and disk drives, as just described. The AWSMOUNT command (executed in an OS/2 command window) can change the file name associated with a device. This command is described in 4.7.6, “AWSMOUNT Command (P/390 and R/390)” on page 113. The configurator can also associate OS/2 files with simulated 2540 card readers (device manager AWS2540) and 1403 printers (AWS2821). In some cases, the device manager can dynamically create a new output file. The DOC files briefly describe these capabilities.

One other panel contains configuration information. From the menu panel shown in Figure 17 on page 39, press F4 to Update System Environment. This should produce the following panel:

²⁹ You must specify a label in order to define a new volume. OS/390 does not use this label; OS/390 gets the volser from the VTOC of the volume

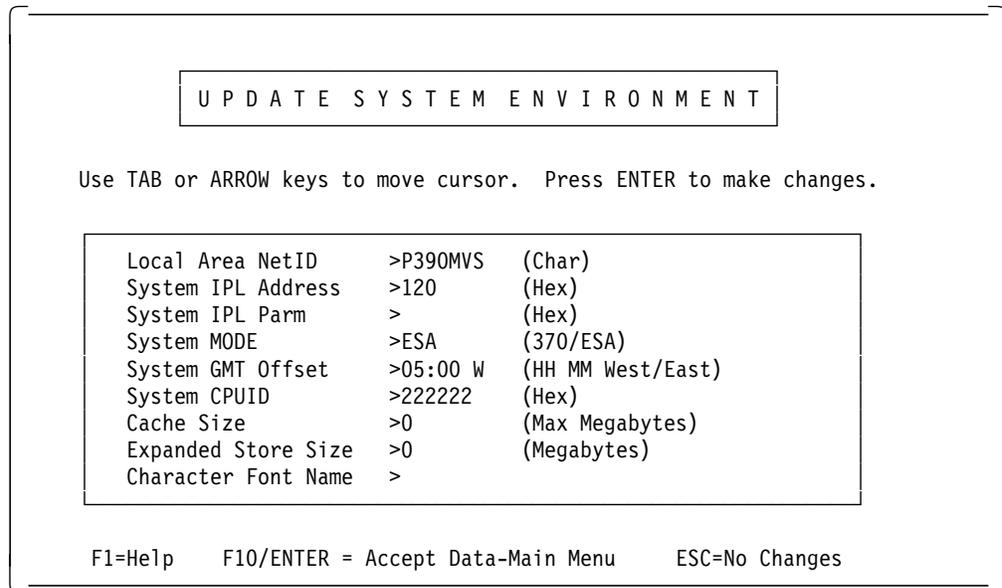


Figure 19. System Environment Panel

Set the parameters as appropriate, including the normal IPL address for your OS/390 system. Address 120, shown in the example, is the address often used by IBM-provided OS/390 systems. The “Cache Size” is for VM; leave it blank. The “Local Area NetID” is used for connecting LAN PCs with 3270 emulators, using the LAN3274 device manager, and appears only for P/390 systems.³⁰ The “Expanded Store Size” partitions your S/390 memory, and uses the indicated amount as expanded storage. Most OS/390 users leave this value zero, on the assumption that the 128 MB storage is best used as all main storage. The “Character Font Name” is for emulated 3215 consoles (often used with VM). Leave it blank. The “System CPUID” is used to form the CPUID that can be read by OS/390. You can set any value here.

The IPL address you set in this panel becomes the default IPL address that is used when you click “IPL P/390” in the primary P/390 icon window.

³⁰ This parameter assigns a NetBios network name. For your initial OS/390 installation, you will use only local 3270 sessions (local in your base OS/2 system), and a LAN name is not relevant.

Chapter 3. Installation

Your system may be delivered with everything installed and operational. However, you are likely to need to rebuild your system sometime during its lifetime. There are many reasons for this: a desire to start again “from scratch,” a really disastrous series of hardware problems, the availability of totally new operating system code, and so forth.

This chapter provides an overview of the installation processes for PC Server System/390 and RISC/6000 S/390 systems using OS/390. The two server bases are covered separately. The purpose of these overviews is to describe the general structure and goals of the installation process. We will only briefly describe some of the specific steps. Much more complete installation instructions are found in IBM publication SA22-7210-01 (for the PC Server System/390) and SA22-7228 (for the RISC/6000 S/390).

The installation discussion assumes you are using either the *Preconfigured OS/390* or the *OS/390 Application Development* system that is available on CD-ROM.

3.1 PC Server System/390 OS/390 Installation

There are a number of steps in the installation process we used:

1. Installing hardware components
2. OS/2 installation
3. CM/2 installation
4. P/390 support programs installation
5. Restoring the CD-ROM OS/390 operating system
6. Configuring P/390 support:
 - a. Configuring local 3270 terminals
 - b. Setting S/390 environmental parameters
 - c. Configuring OS/390 I/O
7. IPLing OS/390 and using it
8. Initializing emulated 3380 disks
9. Adding additional device manager functions

Each of the areas is discussed in this chapter. We assume the reader is familiar with both OS/2 and OS/390 and will not go into detail about common operations.

3.1.1 Hardware Installation

The standard IBM PC Server 520 System/390 used as the base for a P/390 MVS system has a RAID adapter and an integrated SVGA display adapter. It will have a CD-ROM drive and a 4mm tape drive. It will have some number of SCSI disk drives. The number of disk drives (and any extra power supplies and backplanes needed to install them) is discussed in Chapter 2, “Configurations and the P/390 Configurator” on page 23

The S/390 microprocessor complex (adapter card, with second card containing the additional 96 MB memory) is normally installed in the last two Micro Channel slots. The daughter card should be attached to the P/390 adapter before installing the adapter in the server.

Many P/390 MVS systems will have a S/370 Channel Emulator/A adapter card. This card can be installed in any slot. You *must* give some thought to handling the cable for this card. A very large, heavy cable connects to the card through a 62-pin D-shell connection on the back of the card (just like connectors are attached to other PS/2 adapters). This cable is about 2m long and the other end splits into standard parallel channel bus and tag connectors. The card/cable connection is not mechanically strong. Undue movement of the cable could damage the adapter card. With a little thought and care this is not a problem, but the thought and care are necessary.

Most systems will have a LAN adapter: token ring or ethernet. This can be installed in any free slot.

After the adapters are installed, the system is started and the standard reference program is run.³¹ The program will note there have been configuration changes and will offer to run the autoconfiguration process. This should be rejected until option diskettes are copied to the system partition (where the "reference diskette" files reside). Copy the option diskettes in this order:

1. Option diskettes other than the P/390 option diskette and the Channel Emulator option diskette.
2. Do not use the S/370 Channel Emulator/A option diskette
3. P/390 Option Diskette

Your system may already have these option files copied to the system area of the hard disk. The key here is to copy the options information on the P/390 option diskette *after* copying the other diskettes. In particular, the P/390 options replace files that were copied from the Channel Emulator diskette (which you do not need to copy).

After copying the option diskettes, run autoconfiguration. After it runs, start the reference program again (using F1 during the PC Server 520 boot process). Go to the Set Configuration panel, and then the Change Configuration subpanel. Be careful here; do not make any accidental changes. The display should include the following:

```
S/390 Microprocessor Computer (P/390)
  Adapter I/O Space           I/O base 1C00h-1C1Fh
  Adapter Memory Window      4 MB at E0C-----H <---check
  Interrupt Level            10 (use 10, 11, or 15)

IBM Token-Ring Network 16/4 Adapter A
  Adapter Data Rate          16 Mbps <---check
  ROM Address Range          DA000-DBFFF
  RAM Size and Address Range 16KB / DC000-DFFFF <---check
  Interrupt Level            3 <---change

S/370 Channel Emulator/A for P/370 (or P/390)
```

³¹ On some IBM PS/2 systems, this program is run from diskettes and is often identified as the "reference diskette" program (even when it resides on a hard disk).

Control Interface Location	C8000 (8K)
I/O Buffer Location	PROGRAMMED 64K > 1M <---check
I/O Interrupt Level	11

The above listing is not complete; the system will display many other parameters for these and other adapters. When using previous P/390 systems based on the PC Server 500, interrupt 12 may have been used for the Channel Emulator adapter. In the Server 520, interrupt 12 is reserved for PCI adapters. The same interrupt number may be used for the P/390 adapter and the Channel Emulator.

Also, the above listing assumes use of the older Token-Ring Network 16/4 Adapter. If you are using the more recent LAN streamer, you will not need to worry about buffer sizes. You can change a parameter by tabbing to it and pressing F5 and F6. *Change the token-ring interrupt level to something other than level 2.* We suggest level 3 if it does not interfere with anything else in your system. Verify that the token-ring adapter has a 16K RAM area; if you need to change this, you may have to juggle other addresses to avoid conflicts.³² Save any changes before exiting from the panels.

Verify your SCSI addresses, using the SCSI configuration panel. Do not change the addresses of SCSI devices supplied with the system, but be certain that any disks you added have their unique SCSI address set properly.³³

There is no need to connect the Channel Emulator or token-ring cables to anything before completing the installation process.

There are cases when the autoconfiguration program appears unable to handle the S/370 Channel Emulation/A adapter. That is, it reports *conflicts* after the autoconfiguration. The recovery from this is unusual:

1. Remove the Channel Emulation adapter
2. Run autoconfigure, and ensure that the results are good (no conflicts reported)
3. Reinstall the Channel Emulator adapter
4. Place the reference diskette in the diskette drive before powering on the system.
5. When the system *first* asks permission to run autoconfigure, reply YES.
6. This should resolve the conflicts.

The unusual nature of this recovery procedure is that *autoconfigure* internal operation is slightly different when it *first* detects a new adapter. Running autoconfigure later, from one of the reference diskette menu items, is not quite the same thing.

³² If the process described in this paragraph is new to you, find an experienced PS/2 person to help.

³³ The SCSI backplanes used with the PC Server 520 are intended to set SCSI addresses automatically. Jumper cables are provided to connect the SCSI address pins on a disk drive to the hot-swap carrier tray, which is then plugged into the backplane.

We found that the address jumper cables provided did not fit the address pins on our disks. (We were using some old drives, not the nice one-inch drives that are normally delivered with the Server.) We set the SCSI address on our disks by using the normal disk jumper pins. We then connected the disk tray to the backplane connector that corresponded to the SCSI address we used. (We do not know if this correspondence was necessary.)

Slot 6 in a disk bay is for SCSI address 5, slot 5 is for SCSI address 4, and so forth. Slot 1 is for SCSI address 0, but this slot (in the top disk bay) should not be used because SCSI address 0 is used by the 4mm tape drive.

3.1.2 OS/2 Installation

We installed Warp, using a standard distributed version. A CD-ROM version of Warp is shipped with every PC Server S/390. At the time this was written, the standard version of Warp (without any corrective service diskettes added) was supported for the PC Server 500 systems, and Warp Server for PC Server 520 systems. Our installation was absolutely standard. We did not install BootManager or DOS/Windows, but you could install them if you need them.

If you have an array (RAID) model, remember to add the IBMRAID.ADD file to the OS/2 installation diskette #1, and the statement BASEDEV=IBMRAID.ADD added to CONFIG.SYS on that diskette. (Instructions are in the documentation provided with your system.) This file should be dated 2-14-95, or later.

We used the FDISK function during Warp installation to partition our primary disk. If you have a RAID system, your RAID array will appear as a single very large disk.³⁴ We created two disk partitions:

- 250MB (HPFS) for Warp OS/2 (Drive C). This is more space than required, but it provides additional work space and room for small products.
- A large (HPFS) partition for all the remaining space (Drive D). This is about 8 GB in the most common PC Server System/390 configuration, with five 2.25 GB disks.

HPFS must be used for disk partitions larger than 2 GB; we saw no reason to have a mixture of FAT and HPFS, so we used HPFS for both partitions.

*In general, a 4mm tape drive cannot be used by both OS/2 and OS/390 after booting OS/2 with a given CONFIG.SYS file. (See 5.4, "XTAPE Product (P/390 Only)" on page 131 for an exception to this statement.) Different drivers are required in CONFIG.SYS for these two uses, and only one driver can become operational when OS/2 is initialized. Which OS/2 4mm driver might be used depends on additional products you might purchase, such as an OS/2 tape backup program. These products usually supply their own tape driver, since there is not a *standard* OS/2 tape driver. If you want to switch between OS/390 and OS/2 use, you must edit your CONFIG.SYS file and reboot the system. (OS/390 must be stopped before doing this, of course.)*

3.1.3 CM/2 Installation

CM/2 is used to provide "local" 3270 display windows for P/390 MVS.³⁵ Release 1.11 for Warp should be used. (Note that this is a special release for Warp. The CD-ROM says "for Warp" on the title side. A copy is provided with your system.)

CM/2 is shipped on a CD-ROM containing LAPS (more properly known as Network Transport Services/2, or NTS/2). This is a prerequisite for CM/2. Simply run the installation program in the NTS2 directory to install it. The LAPS configuration is not very important at this time.

CM/2 can handle multiple communication paths. For example, it can work with the P/390 (direct communication through storage, no LAN necessary), or with

³⁴ You can create multiple RAID arrays, working with the RAID utilities, but there is no reason to do this here.

³⁵ This was true at the time this was written. IBM plans to remove 3270 emulator functions from CM/2. You can also use OS/2 TCP/IP at this point, but setup is more complex. The P/390 developers plan to provide direct connection from the PC/3270 emulator product to the AWS3274 device manager, although there is no stated schedule for this function. These changes will eliminate the need for CM/2 for basic P/390 operation.

coax 3270 connections, or with a variety of LAN connections. In many cases, you probably want several of these options. CM/2 installation and setup is moderately complex. It begins with the CMSETUP program on the distribution media. (We installed from CD-ROM, and this program was in the CM2 directory.)

For your initial work, we suggest you configure only the LOCAL 3270 terminal sessions of CM/2. You can add coax and LAN operation later, if you need them for other purposes.

Observe the following instructions carefully, because they do not follow the intuitive path through CM/2 setup. You want to SETUP either a new or existing CM/2 configuration file.

1. In the appropriate panel, select "3270 Emulation through Coaxial (DFT)"³⁶
2. Click on the "Configure" button.
3. Set the number of terminal sessions. We used three, which should be regarded as a minimum, to have a less cluttered OS/2 screen; the maximum number is five. This is the number of 3270 emulation windows that appear on your OS/2 screen when you start CM/2 communications. Set the number of printer sessions to zero.
4. Click on "Advanced."
5. Double click on "Required 3270 Emulation."
6. Position the pointer on the "A" under the heading "Long Session/LU Names." Double click.
7. Change the presentation space to 33x80³⁷. Turn off "Activate presentation space print."
8. Click on "Change" for the "Alarm" option. Turn off "Screen update alarm."
9. Repeat the last steps for sessions B, C, D, and E.
10. Exit by double clicking on the upper left corner icon.
11. Click on "Close."

Later, when you want more connectivity, you can rework this configuration as needed. Once your OS/390 system is installed and in production, you might want to place a "shadow" (a common Warp term) of the CM/2 startup icon in the OS/2 startup folder. This will automatically start the 3270 emulator sessions when OS/2 is started.

3.1.4 P/390 Support Programs

There should be six P/390 diskettes with your system, and installing them is easy. Insert diskette 1 in your diskette drive and enter A:INSTALL d: (using an OS/2 command-line window), where "d" is the target drive for the installation. We used our drive D. Do not make drive A the current drive before entering the INSTALL command.

The installation process will automatically create the P390 directory on the target disk. The installation program will prompt you to load the diskettes, in order. It

³⁶ This label seems inappropriate, because we are generating a local 3270, not a coax-attached 3270 session. Nevertheless, you must select this option.

³⁷ We assume you want to emulate 3279-3 terminals. Most users prefer these.

will ask permission to update CONFIG.SYS. Reply "Y" unless your CONFIG.SYS already has the statements added by the P/390 support programs installation process.

After diskette installation is complete, you may need to edit CONFIG.SYS, using your favorite OS/2 text editor. If you have a Channel Emulator adapter or SCSI tape drives, you will need to edit it. (You will also need to edit CONFIG.SYS after installing the WAN SDLC device manager, but this is covered in a separate step.)

The P/390 support program installation process placed a number of lines at the end of the file. If you have a channel emulator, locate and remove the REM at the beginning of the lines:

```
REM DEVICE=D:\P390\CHAN370.SYS
REM RUN=D:\P390\CHAN370.EXE
```

If you have a 4mm tape drive and do not have an external 3480 drive, we suggest you enable the SCSI3480 driver for your 4mm drive. Leave the REM in place for the SCSI3420 driver. You need to edit your CONFIG.SYS to activate the line:

```
DEVICE=D:\P390\SCSI3480.SYS 00,00
```

or something similar. (This device manager is described in Chapter 4, "Device Managers and Commands" on page 69.) This enables use of your 4mm drive or an external SCSI 3480-type drive to emulate a 3420 drive. This statement is already in your CONFIG.SYS; you need to remove the "REM" from the beginning.

```
DEVICE=D:\P390\SCSI3420.SYS 00,00
```

If you have additional SCSI tape drives, you will need to have more of these DEVICE statements, with slightly different driver names and addresses (the "00,00" parameters). This customization, described in 4.4.2, "SCSI34x0 and AWS34x0" on page 86, may be done later.

This completes the installation of the standard P/390 support programs. You will need to reboot your OS/2 system to activate the changes to CONFIG.SYS. *Always shut down OS/2 properly before rebooting or turning the power off.* When you reboot, you should have a new desktop icon for "P/390."

3.1.5 OS/390 CD-ROM Installation

An emulated CKD volume, such as a 3380 volume, is a single (large) file to OS/2. The CD-ROM contains several of these large files, each being an emulated 3380 volume. Even by CD-ROM standards, these files are large. For this reason, the PKZIP utility was used to compress the files before creating the master CD-ROM. Installing the CD-ROM consists of PKUNZIPing the files as they are copied to your hard disks.

In addition to the compressed CKD volumes, the CD-ROM contains the file DEVMAP.MVS, which is a device map file already configured for the OS/390 system on the CD-ROM. It also contains two standalone utility programs (ICKDSF and a tape restore program) in the format used by the AWSTAPE device manager, and a README file.

The README file contains installation instructions. In the most basic case, installation is something like this (where drive E is the CD-ROM drive):

```
C> E:INSTALL
```

When installation completes (including loading a second CD-ROM for the Developers' Association system), you will have this:

- A new subdirectory on drive D, named MVS.
- In the MVS subdirectory will be:
 - Two or four (depending on which CD-ROM version you use) large files containing emulated 3380 volumes
 - The DEVMAP.MVS file.
 - A README file
 - Two standalone utilities (ICKDSF and DFSMSdss) in AWSTAPE format

If you need to reinstall only one of the volumes, see the instructions in 3.6, "Non-RAID Installation (PC Server)" on page 63.

3.2 IPL and Use MVS

Current OS/390 (and MVS) systems usually require an IPL Parameter in order to IPL. For the Application Development CD-ROM, this parameter is 012200. You should verify that your IPL Parameter is specified correctly in the Update System Environment panel of the Configurator.

You must start 3270 emulator sessions (for your OS/390 console and at least one TSO session) before IPLing OS/390. If you are using CM/2 for 3270 emulation, you should open (double click) the CM/2 icon, and then Start Communications (double click). Wait for the 3270 windows to appear before proceeding. (You can close the CM/2 folder, leaving the 3270 sessions running. This gives a less cluttered screen.)

Open (double click) the P390 icon, then double click the IPL P/390 option. The S/390 Activity window should appear and you should see processor activity. After a few seconds you should see OS/390's SPECIFY SYSTEM PARAMETERS message on the first 3270 session. Select this window (position the mouse pointer in the window and single click) and press Enter. (Remember that the 3270 Enter key is the right-hand Cntl key on the PC keyboard.) Continue with a normal MVS IPL.

IPL is mostly automatic. You will need to reply "NOREQ" to the JES2 startup message. VTAM and TSO will start automatically, although you may need to reply RETRY if TCAS is ready before VTAM initialization completes.

When VTAM starts it should display a logo on the 3270 sessions. Select one of the 3270 windows (with the mouse) and log onto TSO. Enter LOGON P390; the password is P390. Use the command ISPF to start ISPF.

When you are finished with OS/390, stop it cleanly.³⁸ Then double click on STOP P/390 and you are finished. Remember to shutdown OS/2 cleanly, as well. (Stopping OS/2 cleanly may be more important than stopping OS/390 cleanly. HPFS can require a considerable amount of time to verify a LARGE file system, after an OS/2 crash.)

³⁸ Not everyone does this, of course. You should at least log off any TSO sessions and stop CICS and DB2 (if running) before crashing OS/390. You can crash it by simply double clicking the STOP P/390 icon.

3.3 RISC/6000 S/390 OS/390 Installation

Installing an RISC/6000-based system follows most of the same basic concepts as installing a PC Server-based system, although many of the details are different. The normal marketing and support channels for R/390 systems are such that most systems are delivered with at least the Preconfigured OS/390 system installed, and in a ready-to-IPL state. We will briefly discuss some of the installation steps because many owners, at some point in their usage, may want to reconfigure or upgrade their hardware or software.

3.3.1 Hardware Installation

Because a much wider range of hardware configurations are possible for an R/390 (assuming it is based on an RISC/6000-591 system), we cannot provide much specific information about hardware installation. Our system had 256MB memory (left over from another project) and the following adapters:

1. Ethernet (single port)
2. P/390 adapter
3. Additional memory for P/390 adapter
4. Portmaster adapter (for SDLC lines to WAN3172)
5. Token Ring (single port)
6. Display adapter
7. Differential SCSI adapter (unused in our configuration)
8. S/370 Channel Emulator/A adapter

We also had four 2.25 GB disks attached to the integrated SCSI adapter, along with a CD-ROM drive, an 8mm tape drive, and a 4mm tape drive.

Install as much hardware as possible before installing AIX. The AIX installation process detects what hardware is present and installs drivers only for that hardware. If you add adapters later, for example, you may need to install additional driver modules.

3.3.2 AIX Installation

We installed AIX 4.1.4 from CD-ROM, using very basic installation options. This was done by loading the CD-ROM, turning the system key to Service, applying power to the system, and following the instructions on the screen.

After the AIX CD-ROM installation process starts, you are given several choices by selecting the Change/Show Installation Settings and Install option. In this panel, we elected to do a New and Complete Overwrite form of installation (as opposed to a Preservation installation). In this same panel, we selected the disk we wanted to use for AIX and elected to install the tcb (Trusted Computing Base). R/390 operation does not depend on any of these choices, and you may do something different.

The installation process took about 30 minutes once it started. When it finishes, it displays the VSM Installation Assistant Task List. At this point, we used an option to set root's password, and then used the Tasks Complete option. This causes the Installation Assistant to end, and AIX returns to a normal command line prompt. Some of the steps listed in the remainder of this R/390 installation description could be done with the Installation Assistant. We elected to use a combination of smit and basic commands. Again, it makes no difference to R/390 functions how you choose to complete these steps.

At this point, AIX had the following file systems on a single disk that constituted the rootvg volume group:

		allocated	used
hd4	/	8192 (4 MB)	46%
hd2	/usr	270336 (135 MB)	98%
hd9var	/var	8192 (4 MB)	13%
hd3	/tmp	16384 (12 MB)	4%
hd1	/home	8192 (4 MB)	5%
hd6	paging	64 LP (256 MB)	
hd5	boot	1 LP (4 MB)	
hd8	jfslog	1 LP (4 MB)	

The allocated space is given in 512-byte blocks or in LPs, which are 4 MB logical partitions. The paging logical volume is unusually large because our system had 256 MB of real memory, left over from another project.

We later found that we needed to install a number of additional programs from the AIX CD-ROM. There were:

1. The dosdir, dosread, and doswrite commands.
2. Basic printer support for the Lexmark printer we had attached to the system.
3. The bos.compat.lan files.
4. The X11.font.compat files.
5. The bos.dlc files. (The dlc files will not be needed for R/390 releases after 4.0.15.0. The LAN3172.DOC file with your R/390 support diskettes will indicate whether the dlc files are needed. It will not hurt to install them, whether needed or not.)

We found the DOS utility commands in a *bundle* that also included the printer support we wanted and CDE. (We did not particularly want CDE, but did not object to it.) This was the *Personal Productivity* bundle on the AIX CD-ROM. If you are installing a system by following these instructions, we suggest you install these additional programs now. We used the following SMIT process to install the bundle:

```
smit
  Software Installation and Maintenance
    Install and Update Software
      Install Bundles of Software (Easy Install)
        Install Bundle Contents
          Input device - /dev/cd0 (or press F4)
            select Pers-Prod Bundle
              ENTER to begin installation
                check the SUCCESS messages
```

We then used smit again to install the other components:

```
smit
  Software Installation and Maintenance
    Install and Update Software
      Install/Update Selectable Software (Custom Install)
        Install Software Products at Latest Level
          Install New Software Products at Latest Level
            Input device - /dev/cd0 (or press F4)
              SOFTWARE to install (press F4 for list)
                (It takes some time to produce the list)
```

Scroll through the list, and use F7 to mark the components you want to install. In the X11.compat group, mark "4.1.3.0 AIXwindows PC850 Fonts Compatibility", in the bos.compat group, mark "4.1.3.0 LAN COMIO Compatibility Software", and mark the whole "4.1.0.0 bos.dlc" group.

ENTER to begin installation
check the SUCCESS messages

(Note: you must reboot AIX before this LAN software is used.)

This additional installation processes took about 30 minutes, most of which is spent installing CDE. The installation process expands file systems, if required. At this point, our file systems took the following space (in 512-byte blocks):

	allocated	used
/	8192 (4 MB)	57%
/usr	434176 (217 MB)	98%
/var	8192 (4 MB)	14%
/tmp	24576 (12 MB)	5%
/home	8192 (4 MB)	5%

The total allocated space shown here is about 236 MB, plus the space used for the paging, boot, and jfs logical volumes.

We observed that the /usr file system was almost full, and knew that the P/390 support programs (and other products we might want to install) would go into this file system. An AIX file system is in a logical volume. If you expand the file system, the logical volume is automatically expanded (provided there is room in the volume group containing the logical volume). To expand a file system:

```
smit
  System Storage Management
    File System
      Add / Change / Show / Delete File System
        Journalled File System
          Change / Show Characteristics of a Journalled File System
            (Select /usr from list shown)
              SIZE (in 512-byte blocks) = nnnnnnn
                (overtpe the size field with a larger number)
                  ENTER
```

For example, the SIZE shown for /usr was 434176 512-byte blocks. We changed this to 510000 blocks, which added about 40 MB to /usr. The system automatically added a number of LPs (logical partitions) to /usr to match the requested size. We also added 32000 blocks (16 MB) to the root file system, and the same amount to the /home file system.

We added another user ID at this point. This is not required, since all the P/390 support functions and operation are under the root user ID.

```
smit
  Security and Users
    Users
      Add a User
        User NAME = OGDEN
          (no other parameters needed. Press ENTER)
      Change a User's Password
        User NAME = OGDEN
```

(enter password)

If you are using hcon for 3270 emulation, you will need to install, configure, and add users for it. This is described in 5.15.2, "hcon 3270 Emulator" on page 152.

At this point we shut down AIX, turned off the power, and verified that the P/390 adapter was installed. When AIX was restarted, the CDE (full-screen X Windows) logon screen appeared. There is an option to use a standard command-line logon instead, and we used this option to complete the installation process. (You can install and operate the R/390 under CDE if you like; we are not aware of any restrictions in this area.)

3.3.3 P/390 Support Programs

The P/390 support programs, on multiple diskettes, are installed with smit:

1. Log in as root, start X windows, use an aixterm window.
2. Enter *smitty install_latest* to start smit at the right point.
3. Specify */dev/fd0* as the input device (and insert the first diskette in the diskette drive).
4. Press ENTER twice to start the installation.
5. At the end, the messages should indicate that both the *root* and *user* portions were successful. The process takes about five minutes.

The P/390 support programs are installed in */usr/lpp/r390/bin*, which is linked to */usr/bin/r390*. That is, the files are actually installed in directory */usr/lpp/r390/bin*, but can also be accessed in directory */usr/bin/r390*. This second directory path name is shorter, and is normally used for references.

You need to add this directory to your PATH. Edit */etc/environment* and add */usr/bin/r390* to the end of the PATH definition:

```
PATH=/usr/bin:/etc:/usr/sbin:../usr/bin/X11:/sbin:/usr/bin/r390
```

There is one global environmental variable that must be defined. First change the permissions for */etc/profile*:

```
chmod 755 /etc/profile
```

and then edit */etc/profile* and add:

```
CONFIG_FILE=/mvs/devmap.mvs
PS1=' $PWD # '
export CONFIG_FILE PS1
```

at some appropriate place, using the directory and file name you will use for your system. (Our use of these names is described in the next section.) You are not required to change the PS1 prompt, as shown here; however we found usage easier if the AIX prompt line displayed the current directory. (If you have a more sophisticated prompt command, this is the time to install it.)

The P/390 licensed code and diagnostics should be installed next. They are on the IBM PC Server S/390 Advanced Diagnostics and Option Diskette, which is separate from your other R/390 diskettes. To install, insert the diskette in the reader and:

```
cd /usr/bin/r390
./loadlic
```

This process takes several minutes, and places files in */usr/lpp/r390/bin* and */etc/asw*. It uses *dosread*, so this program must be available by this point in the installation process. (Remember that */usr/bin/r390* is a link to */usr/lpp/r390/bin*.)

Descriptions in this document and most R/390 documentation refer to /usr/bin/r390.)

3.3.4 OS/390 CD-ROM Installation

A complex shell script is provided for installing various CD-ROM systems: OS/390, VM, and VSE. This shell script generally assumes you have a newly installed AIX, with sufficient free disks (not in any volume group), and it builds a complete environment. For the initial installation of a CD-ROM system, probably done before your R/390 system was delivered to you, this installation shell script is completely appropriate.

If you need to reinstall OS/390 from CD-ROM, or need to reinstall a single volume from the CD-ROM, the shell script may be too complex. The following steps describe a minimal installation, using AIX commands. It is unlikely to match exactly what you want to do, but you may be able to adapt it to what is needed.

1. Log into AIX as *root*. All of the steps assume you are working as root.
2. You will need to write large files, and you must change the *ulimit* parameter to permit this. Edit */etc/security/limits* and add the line (if you have not already done this):

```
root:
    fsize=4194303
```

to the stanza for root. This permits you to write files up to 2GB. (The number is the maximum number of 512-byte blocks.) We found we needed to log out and log in again to have this new size take effect. Issue the *ulimit* command to verify that your maximum file size is at least 4000000; if it is not, the following installation steps will fail.

3. Our system had three disks that were unused so far. We started with four disks, and directed AIX installation to *hdisk3*, leaving *hdisk0*, *hdisk1*, and *hdisk2* free. (There was no reason for selecting *hdisk3* for AIX; you can use any disk arrangement.) We now want to make a large volume group to hold the emulated S/390 volumes. We decided to use the name *mvs*. The AIX command:

```
mkvg -d 3 -y mvs hdisk0 hdisk1 hdisk2
```

will create the volume group. If your drives are larger than about 4.0 GB, or if you have a RAID array that makes a large logical drive, you will need another operand to force the physical partition (PP) size to larger than the default value of 4 MB. A sample command would be `mkvg -d 3 -y mvs -s 8 hdisk0 hdisk1 hdisk2`. This would cause all physical partitions to be 8 MB, and is suitable for a drive up to 8 GB. The operand `-s 64` would be used with a RAID array up to 64 GB. Most documentation assumes that the PP size (and the LP size) is 4 MB. If you must use a larger size than 4 MB, take care reading instructions and examples that assume a 4 MB PP/LP size.

You are not required to have a separate volume group for R/390 emulated volumes. You can place them anywhere, in any file system. However, since they are such large files, and since their use is completely separate from any other AIX usage, we expect most users will want to place them in a separate file system.

4. Use the command

```
varyonvg mvs
```

to make the volume group accessible.

5. Make a mount point for a new file system in the root directory:

```
mkdir mvs
```

creates a directory named *mvs*, and we will mount a new file system over this directory. (There is nothing special about the name *mvs*; we wanted something that would clearly indicate R/390 files and selected this name. At this point we are using the name *mvs* for a volume group, a directory, and (in the next step) for a file system; these multiple usages do not conflict.

6. We now want to make a file system that uses all (or almost all) the space in the volume group. We used this command:

```
crfs -v jfs -m /mvs -g mvs -A yes -p rw -a size=9500000
```

This will take several minutes to create a file system of size 9500000 * 512 or 4.86 GB. (This leaves a small amount of free space in our volume group. We were uncertain whether this would be needed for the *jfs* log file.) You should adjust the size parameter to whatever fits in your *mvs* volume group, of course. The file system will be mounted automatically when AIX is started.

7. Use the command:

```
lsfs
```

to determine the logical volume name that was created for the file system. In our case, it was */dev/lv00*.

8. Mount the file system:

```
mount /dev/lv00 /mvs
```

9. Now we need to mount the first CD-ROM volume. We had already determined (by listing */dev*) that the name of the drive was */dev/cd0*. We first created a mount point for it:

```
mkdir /cdrom
```

This could be any name, but *cdrom* is widely used for this purpose. We inserted the CD-ROM in the reader, and used this command:

```
mount -v cdrfs -r /dev/cd0 /cdrom
```

10. Use the *ls* command to explore the CD-ROM. The files we want to move are in the subdirectory *mvs*:³⁹

```
ls /cdrom
ls /cdrom/mvs
```

11. Unzip the required files, and place them in */mvs*:

```
/usr/bin/r390/unzip -j /cdrom/mvs/mvsv5r.zip -d /mvs
```

This will unzip the emulated MVS5R volume, which will take at least 20 minutes. We also needed the *devmap.mvs* file from this CD-ROM, and this was not in ZIP format.

```
cp /cdrom/mvs/devmap.mvs /mvs/devmap.mvs
```

³⁹ The examples and instructions here are for the MVS 5.2.2 AD CD-ROM. The OS/390 CD-ROM was not available in time to include specific instructions for it. It may use different directories and different file names, but the principles of installation are the same.

The second emulated volume we wanted was on the second CD-ROM, so we needed to do this:

```
umount /cdrom
  (change the CD-ROM)
mount -v cdrfs -r /dev/cd0 /cdrom
/usr/bin/r390/unzip -j /cdrom/mvs/scpmv5.zip -d /mvs
```

12. After this completes, check your /mvs directory and determine if the file names are all lower case. AIX is case sensitive, but OS/2 is not. The creation process for the CD-ROMs sometimes produces files with upper-case names after they are unzipped.

```
ls /mvs
devmap.mvs MVS5R.120 scpmv5.122  (<-- displayed line)
mv /mvs/MVS5R.120 /mvs/mvsv5r.120
```

You can use upper-case file names, but this is unusual in AIX and may confuse someone. We suggest you change all the names to lower case, as shown in this example.

13. Issue the commands:

```
cd /usr/bin/r390
awsprof
```

to verify that the current profile is /mvs/devmap.mvs. If it is not, issue the command:

```
awsprof /mvs/devmap.mvs
```

14. We now had the two emulated volumes we needed to IPL, and a DEVMAP. The DEVMAP we copied (devmap.mvs) was intended for P/390 instead of R/390, so we needed to correct some of the file names. In particular, we needed to change the Server file names for the emulated 3380 volumes. To start the configurator:
 - a. Start X windows with *xinit*.
 - b. In an aixterm window, enter the command *awscnf*. (You could use the command *r390* instead, to obtain the complete P/390 main menu.)
 - c. The configurator should start. Press ENTER to get through the password screen. Press F2 to Update System Devices.
 - d. The initial DEVMAP has at least four 3380 devices defined at addresses 120-123. If you look in the parameter portion of each line; these may use OS/2-style file names. You can overtype the file names. In our example, we would change address 120 (our planned IPL volume) to use /mvs/mvsv5r.120, and 122 would point to /mvs/scpmv5.122. After you enter the correct file name, the configurator should update the display to show the size (in cylinders) of the emulated volumes.
 - e. You can *turn off* the DASD devices you do not need at this time, or remove the file parameter completely. Do not leave file names with invalid AIX syntax as pointers.
 - f. Remember to save the changes when you exit the configurator. Use F10 followed by F6.

3.3.5 IPLing OS/390

Someone must log into AIX as *root* to start the P/390 subsystem and start the IPL process for OS/390. In principle, this could be changed such that these tasks could be done under some other user ID, but we did not explore this option. The standard IPL script for the R/390 automatically starts two telnet 3270 sessions (one normally becomes the OS/390 master console) and these will run under the user ID in effect when they are started. This will usually be *root*.⁴⁰ Also, any S/390 console functions will require a *root* user.

To IPL, verify that the correct IPL address and IPL Parameter are set in the Update System Environment panel of the Configurator, and then:

<code>xinit</code>	(to start X windows)
<code>r390 &</code>	(in the aixterm window)
(click IPL)	(in the P/390 main menu, which should appear)

After a few seconds, you should see two 3270 windows and the S/390 activity display (showing the PSW and a busy-state indicator). Within a minute or so, you should see the SPECIFY SYSTEM PARAMETERS message in one of the 3270 windows, and you can continue the IPL process in the usual way.

We found a few points of interest:

- Starting the `r390` process (which produces the base menu) as a background process frees the AIXwindow for other use. That is:

<code>r390</code>	ties down the AIXwindow
<code>r390 &</code>	frees the AIXwindow for other use

Either command method will work, but we recommend starting `r390` as a background process. Startup messages will still appear in the window that issued the `r390` command, even if the command is run as a background process.

- Single-click (with the mouse) to select functions in the P/390 Menu. (This is easy to say and difficult to do for anyone who frequently uses OS/2 or Microsoft Windows, where double-clicking is normal!)
- After OS/390 was running, we could close the P/390 Main Menu without seeming to harm anything. However, there is no reason to do this -- other than trying to clear a cluttered screen.
- We could use the CDE login process, and run the whole P/390 function in a CDE session. Our only problem doing this was that the `CONFIG_FILE` variable (set in `/etc/profile`) was not set for some reason. We needed to enter it, using a terminal command line in one of the CDE sessions:

```
export CONFIG_FILE=\mvs\devmap.mvs
```

before issuing the `r390` command from the same command line. You could correct this problem by placing the `CONFIG_FILE` variable in the standard profile used by CDE. The starting point for doing this is `/usr/dt/config/sys.dtpofile`. We elected to run without CDE and did not bother to update the CDE profile.

- If you want to disable the default use of CDE, do the following:

⁴⁰ The developers are aware that requiring routine operation as *root* is not desirable, and future releases may restructure the user ID needed for routine operations.

```
smit
  System Environments
  Change System User Interface
  F4 (for list)
  Select the Command-line login option
```

- The colors and structure of the P/390 Main Menu were different under CDE than under basic X windows, and not as attractive.
- The default ENTER key for 3270 sessions (under both X windows and CDE) was the carriage-return key (the NL key). This is disconcerting to 3270 users (especially those using ISPF panels). This was the result of particular 3270 emulator startup options. See 5.15.1, “xant 3270 Emulator” on page 151 for a discussion of one of the default emulators.
- The default 3270 sessions are restricted to 24-line screens and do not respond as respected to resizing. Resizing, by dragging the corner of a window, produces more blank space around the 3270 data lines, but does not make the font larger. See 5.15.1, “xant 3270 Emulator” on page 151 for more information.
- No Send and Receive commands are available with the default xant 3270 emulator.
- When running under CDE, there is a default timeout period. After a period of no screen activity, CDE will *lock* the screen. The password of whoever is logged in at the screen must be entered to unlock it. Since the screen user is normally root, this means the root password must be known to anyone who might need to use the screen.
- The standard AIX tn3270 emulator cannot be used for full-screen applications such as ISPF. You cannot use tn3270 on the R/390 system itself, or on any other AIX system to connect to OS/390 through the AWS3274 device manager.
- You will need to edit the ipl390 script (in /usr/bin/r390) to start additional device drivers such as LCS3172, LAN3172, WAN3172, and the channel emulator. The editing needed is described with each of these device managers in Chapter 4, “Device Managers and Commands” on page 69.
- The standard ipl390 script will try to start hcon, X3270, and xant (in that order) to obtain two local 3270 sessions. If you have hcon installed, you should define two TCP/IP sessions named a and b. These names are coded in the ipl390 script; you can change the hcon session names by editing the script.

3.4 Expanding the CD-ROM Systems

The CD-ROM OS/390 systems are intended as delivery vehicles. As installed, they are usable but not intended for immediate full-scale use. We strongly suggest you add several emulated volumes before beginning any significant work on your P/390 or R/390 system. For example, you should add, at least, a volume for your files (the generic name TSOPAK is often used for this), and a scratch volume. Step-by-step details for doing this are given in the Cookbook.

The Cookbook also includes a number of other steps you might want to perform before considering your system ready for production. These include RACF administration, SMS volume management, SMF setup, and so forth.

3.5 Working With the P/390 Configurator

If you installed your system from CD-ROM, following standard instructions, you will already have a current DEVMAP (located in D:\MVS\DEVMAP.MVS or /mvs/devmap.mvs). If you want to create another DEVMAP, we suggest you *copy* an existing DEVMAP to a new file, and then alter that file using the P/390 configurator. The P/390 configurator (as well as the P/390 IPL function, and practically all other P/390 functions) works with the *current* DEVMAP. You can make another DEVMAP the *current* DEVMAP with the command:

```
D:> AWSPROF D:\DEVMAP.ALT          (for OS/2)
# /usr/bin/r390/awsprof /mvs/devmap.alt  (for AIX)
```

You cannot edit a DEVMAP with a normal text editor. You can change it only with the P/390 configurator. If you attempt to alter it with an editor, it will not be usable.

If you installed a CD-ROM system, the DEVMAP provided can probably be used for your first P/390 OS/390 IPL without any changes. For the first R/390 IPL, you may need to change it slightly. When you start to customize your system (adding disk volumes, adding tape drives, adding terminals, and so forth), you must work with the configurator extensively.

Starting the Configurator was described in 2.6, “The P/390 Configurator” on page 36. When you first open the base OS/2 P/390 window, the icons are displayed horizontally. You may find that a vertical format is easier to use. You can change this using standard OS/2 display functions. Once you have a number of 3270 windows open, plus various other Server and P/390 windows, you will have a very busy screen. We shortened the titles in the base P/390 window, allowing the window to use less screen space. You can do this (under Warp) by clicking on a title with the right-hand mouse button, selecting the General page in the settings notebook, and changing the title. For example, we changed “P/390 Manual Operations” to “ManOps,” and so forth.⁴¹

This base P/390 window is important. You will use it many times. Briefly, the functions are:

- IPL P/390 starts the P/390 I/O subsystem, loads the S/390 microcode, and initiates an IPL from the standard IPL address set in the configuration panel. There are no submenus for this function.
- END P/390 stops the S/390 processor and stops the I/O subsystem. This is the proper way to terminate the whole S/390 function. After using this function, you would normally close the base P/390 window by double clicking the top left corner icon.
- P/390 CONFIGURATION begins a set of panels you will use to manage your system. Some of these are described below.
- P/390 MANUAL OPERATIONS provides a window with many pull-down submenus. These provide S/390 manual operations such as system reset, clear, external interrupt, address stops, S/390 storage display, manual IPL, and so forth.

⁴¹ If shut down properly, Warp remembers all these changes. You only need to do it once.

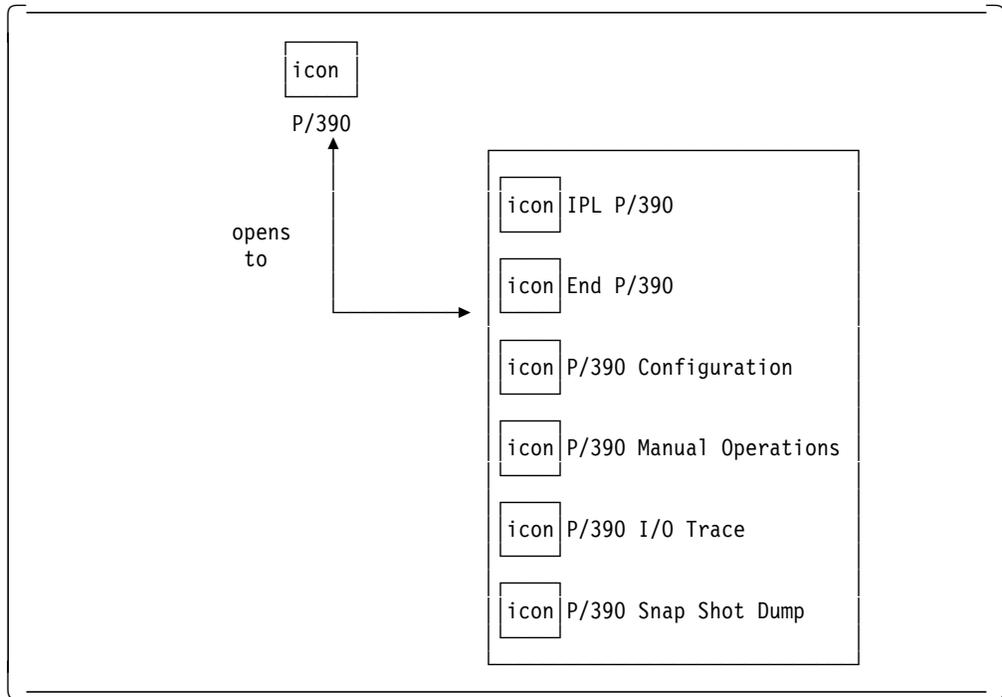


Figure 20. Main P/390 Window. The R/390 version is started with the r390 command instead of a desktop icon.

- P/390 I/O TRACE produces a window displaying trace entries produced by the P/390 support programs. Various pull-down functions are available.
- SNAP DUMP takes an immediate dump of critical P/390 control information (this does not include any S/390 storage). This function is primarily for use by the P/390 developers.

The Update System Devices panel, shown in Figure 18 on page 40, provides the key functions of the configurator. An example of adding a new disk volume will illustrate typical operation.

Entries for disks look like this:

Addr	Device	Label	Atype	Size	Mgr	FN/P
>	>	>	>	>	>	>
120	3380	P390R5	CKD	1770C	2	D:\MVS\P390R5.120
122	3380	SCPMV5	CKD	885C	2	D:\MVS\P390DX.122

In this example, two disks are defined at S/390 addresses 120 and 122. The size field indicates the number of cylinders on the emulated 3380 volume. It is written in the indicated format: nnnnC. The FN field contains the name of the OS/2 file used to hold the emulated 3380 drive. You can use any file name, but we suggest a naming convention so that you can correlate the OS/2 name to the S/390 drive or volser.

Suppose, for example, we want to define a new 3390 volume, with 500 cylinders. (We can define any number of cylinders for 3380 and 3390 volumes, up to the maximum size of the largest "real" 3380 or 3390.) Furthermore, we know that 3390 drives are already defined in our OS/390 system at addresses A80-ABF. We have checked that there is sufficient space on our D drive by using the DIR command (OS/2); the equivalent check for AIX would use the df command to

check the free space in the selected file system. For OS/2 P/390, we would use the Update System Devices panel of the configurator and enter the first line shown here:

```
Addr Device Label Atype Size Mgr FN/P
>A80 >3390 >TS0002 >CKD > 500C > 2 >D:\MVS\TS0002.A80<---enter
120 3380 P390R5 CKD 1770C 2 D:\MVS\P390R5.120
122 3380 SCPMV5 CKD 885C 2 D:\MVS\P390DX.122
```

You can use any name you like for the Server file name (entered in the FN/P field), but we recommend using a meaningful one incorporating the volser and OS/390 address. Allow for about 714,250 bytes of Server disk space per defined 3380 cylinder, or 852,486 bytes per defined 3390 cylinder. After you press RETURN, the configuration program will indicate "Creating File..." This may take some time; the program is writing the whole file in the format used by the CKD emulation program. For example, our system took 22 minutes to define a 1770 cylinder 3380 drive.

Note that creating an emulated disk volume does not initialize it (write a VTOC, among other things). It formats the created volume in the special format used by AWSCKD to emulate CKD tracks, and that is all. You must use other tools, such as ICKDSF, to initialize the volume after it is created.

The configurator panel calls the CKDALC program to actually create the file used for the emulated volume. You can call CKDALC directly, from an OS/2 or AIX command line, and not use the configurator panel to create new emulated volumes. This is described in the AWSCKD.DOC file distributed with the P/390 support programs. We recommend you use the configurator panel, but the results are the same with either method. Using CKDALC directly does not update the DEVMAP; you would need to do this manually, after using CKDALC.

You can delete emulated drives by moving the cursor to the correct line and pressing F9. The deletion process can also delete the associated Server file. (It will ask you to press F9 several times to confirm the deletions.)

The configurator can associate Server files with tape and disk drives, as just described. The AWSMOUNT command (executed in a Server command window) can change the file name associated with a device. This command is described in 4.7.6, "AWSMOUNT Command (P/390 and R/390)" on page 113. The configurator can also associate OS/2 files with simulated 2540 card readers (device manager AWS2540) and 1403 printers (AWS2821). In some cases, the device manager can dynamically create a new output file.

We suggest you try defining several devices (including DASD) with the configuration functions. You can always delete any unwanted files you create on the PC disks, and you can reload the DEVMAP.MVS file if you destroy it. (This never happened to us; the configurator program appears quite robust.) You need to be comfortable with the configuration functions.

A typical, small MVS configuration (with a limited disk configuration such as we had) might look like this for a P/390 system (see Figure 18 on page 40 for the full panel illustration):

Addr	Device	Label	Atype	Size	Mgr	FN/P
>	>	>	>	>	>	>
00E	1403				6	C=D:\PRT.00E
00F	1403				6	D:\PRINT2.TXT
120	3380	P390R5	CKD	1770C	2	D:\MVS\P390R5.120
122	3380	SCPMV5	CKD	885C	2	D:\MVS\SCPMV5.122
123	3380	P390DX	CKD	885C	2	D:\MVS\P390DX.123
130	3380	TSOPAK	CKD	885C	2	D:\MVS\TSOPAK.130
131	3380	WORK01	CKD	400C	2	D:\MVS\WORK01.131
560	3480				H	
580	3420				G	
581	3420				G	
700	3278				3	MVS Console
702	3278				3	Local TSO
703	3278				3	Local TSO
704	3278				4	LAN TSO
705	3278				4	LAN TSO
706	3278				4	LAN TSO
707	3278				4	LAN TSO
708	3278				4	LAN TSO
A80	3390	TS0002	CKD	500C	2	D:\MVS\TS0002.A80

Unit 560 is used with a SCSI-attached tape drive, to emulate a 3480 unit. This might use be the 4mm drive, appearing as a 3480 to OS/390. Units 580 and 581 are emulated tape drives, using the AWSTAPE device manager. The user would issue the AWSMOUNT command, naming an OS/2 or AIX file, in order to "mount" a tape on one of these units.

Another function of the configurator, not mentioned yet, is the Allocate/Change LT3270 function -- F7 on the configurator menu. This is used only with P/390 systems, and partly redefines CM/2 emulator terminals. It produces a window with each CM/2 3270 session listed, where each line looks something like this:

LT	Type	Adapter		LAN Trace	Node Name
		0	1		
LT 1	LOCAL	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

VCFF1

This example indicates that logical terminal A is defined as LOCAL, meaning connected to the P/390 AWS3274 device manager. With your mouse button, select one of the small icons in the LT Type box (shown arrows here). These icons should cycle the LT types among HOST, LOCAL, and LAN. For now, set all the logical terminals to LOCAL types. This is the type required for local connection to your P/390 MVS.

Select SAVE and exit from the LT Sessions panel.

Remember to save your configuration session when you make changes. Use the F6 key to do this when exiting from the configurator.

3.6 Non-RAID Installation (PC Server)

The normal PC Server System/390 uses RAID disks. This has two advantages: better availability and less fragmentation. A RAID array causes the disk drives within it to appear as a single large drive. In our examples, we allocated a 250 MB drive C and an 8+ GB drive D in our RAID array. We then allocated our emulated DASD volumes on drive D. (The same considerations about fragmentation may apply to R/390 systems. AIX's LVM may be used to create a single large logical volume from several smaller physical volumes, and the normal R/390 installation process assumes this has been done.)

If you do not use a RAID system, or if you have some non-RAID drives, or if your R/390 has not used LVM to make a large logical volume, you need to do some planning about disk usage. There is nothing complex about this planning. You must match the required sizes of your emulated volumes with your available disks.

For example, a double capacity (emulated) 3380 requires 1.2 GB. If you have several 2.2 GB disks installed, you can fit only one such emulated volume on a disk, leaving 1.0 GB unused. You could allocate a single-capacity (.6 GB) emulated volume on the same disk, leaving .4 GB unused. You could then allocate another 3380 with about 550 cylinders to use this remaining .4 GB.

With the exception of 3390-3 drives (triple capacity), all emulated volumes are *single* files to the Server, and a single file must fit on a single disk. (This might be a logical disk, in the case of RAID or LVM, but it is treated as a single disk for the purposes of file allocation.) You have the option of using odd-size disks, such as the 550 cylinder 3380 mentioned here, to minimize disk fragmentation. OS/390 appears to fully support odd-size 3380 and 3390 drives, provided the size does not exceed the largest "real" drive.

If you install one of the MVS or OS/390 CD-ROM systems on a non-RAID (or non-LVM) system, you will not be able to use the INSTALL command that is on the CD-ROM. This command, in the case of the P/390, assumes that all emulated volumes will go on your D drive. If you do not have a RAID system, it is unlikely that your D drive is large enough. In this case, you can recreate the installation process manually. You might use some of these same steps if you want to restore a single volume from the CD-ROM.⁴²

There are two CD-ROMs for the Developers' Association version of MVS 5.2.2. The general contents are:⁴³

```
CD-ROM #1
  root directory
    INSTALL.COMD      (OS/2 installation procedure)
    README.MVS
    UNZIP.EXE
  /mvs directory
```

⁴² This has happened to a number of early users. The usual problem is that various SYS1.PARMLIB members were altered without creating alternate members that were known to work. Mistakes in this area can make OS/390 fail during IPL. One solution, if you have not invested much time in other customization on SCPMV5, is to simply restore that volume again from the CD-ROM.

⁴³ The corresponding OS/390 CD-ROMs were not available at the time this was written. Their general structure will be the same, although three CD-ROMs may be required. The OS/390 AD CD-ROMs will include an additional volume (OS39H1) that is SMS-managed and is intended for HFS data sets created by OE users.

DEVMAP.MVS	(matches the distributed system)
DEVMAP.NME	(ignore this)
ICKDSF.IPL	(standalone utility in AWSTAPE format)
MIGRATE.*	(several files. See text below)
MVSV5D.ZIP	(3380 volume with DLIBs)
MVSV5R.ZIP	(3380 IPL volume)
README.MVS	(same as in root directory)
SADSS.IPL	(standalone restore utility)

CD-ROM #2

root directory	
UNZIP.EXE	(same as on first CD-ROM)
/mvs directory	
P390DX.ZIP	(3380 volume with CICS, DB2, etc)
SCPMV5.ZIP	(3380 volume; catalog, paging, spool)

The installation process is quite simple. The emulated volumes are all in PKZIP format. You simply UNZIP them onto whatever disk you wish. For example, if you have four 2.25 GB disks as drives D, E, F, and G, and if the CD-ROM drive is drive H, the commands might be:

```
(start with CD-ROM #1)
C> H:\unzip -o -j h:\mvs\mvsv5r.zip -d d:      (IPL volume)
C> H:\unzip -o -j h:\mvs\mvsv5d.zip -d e:      (DLIBs)
C> copy h:\mvs\devmap.mvs d:\devmap.mvs      (DEVMAP)
(change to CD-ROM #2)
C> H:\unzip -o -j h:\mvs\scpmv5.zip -d f:      (page, spool, ..)
C> H:\unzip -o -j h:\mvs\p390dx.zip -d g:      (CICS, DB2, ...)
```

You can also copy the smaller IPLable files for the standalone utilities, if you like. You might want to copy, or at least read, the README file. After manually installing the 3380 volumes, as shown here, you must use the P/390 configurator to correct the Server file names associated with emulated DASD volumes.

Only the MVSV5R and SCPMV5 volumes are needed to IPL. You are not required to restore MVSV5D or P390DX unless you need the contents for some reason. There are several MIGRATE files on the CD-ROM, including MIGRATE.DOC. This file explains how to install a new version of the CD-ROM without destroying your previous SCPMV5 volume, if you have a previous CD-ROM MVS or OS/390 system already installed. Much of the customization for OS/390 is reflected in data sets in the SCPMV5 volume, and the MIGRATE process permits you to preserve much of your previous customization work.

3.7 Installing Other MVS or OS/390 Systems

(The following discussion uses 3380 drives as examples, but 3390 or 9345 drives could also be used.) There are two ways to distribute an OS/390 or MVS system for a P/390 or R/390:

1. Using normal OS/390 methods, with dump/restore tapes or other formats known to OS/390
2. Using a method unique to P/390 and R/390 systems, involving Server file distribution. This is used with the OS/390 CD-ROMs, for example.

The "normal" OS/390 methods are described in the next several sections of this chapter. They involve defining your emulated 3380 volumes (described in the previous section), initializing these volumes using ICKDSF, and restoring the OS/390 tapes, using DFDSS.

An alternative to this is to use Server files as the distribution format. Remember that a complete emulated 3380 volume is a single Server file. Once an OS/390 system is built (by IBM, or whoever supplies your OS/390 package), the Server files (each containing a complete 3380 or 3390 volume) may be copied to tape or optical media. This has the advantage that the 3380/3390 format has already been established in these files; you can simply copy them (using Server tools) to your disks and they are ready to use. This bypasses the ICKDSF and DFDSS steps needed for a “normal” OS/390 installation. The CD-ROMs are produced this way.

Working with an emulated 3380 volume, using Server tools, has another advantage. The file can be compressed very effectively. Using tools in the PKZIP family⁴⁴ we typically obtained a 5:1 compression factor, making it possible to place OS/390, DLIBs, and a reasonable set of program products on two CD-ROMS.

3.7.1 Creating and Initializing Disk Volumes

If you want to restore an existing MVS system (which we assume you have on dump tapes), you must create (emulated) 3380 volumes and initialize these volumes before you can restore the dump tapes. (If you *copied* (or PKUNZIPed) volumes from a CD-ROM, you do not need to initialize anything. It was done when the original volumes were created, before they were loaded and copied to the CD-ROM master.)

The creation of emulated volumes was described in 3.5, “Working With the P/390 Configurator” on page 59. You should use the configurator to create emulated disk volumes for whatever volumes you need to restore. You must then initialize them. The initialization corresponds exactly to initializing a new drive when “real” mainframe DASD is involved. The volumes must be initialized in a minidisk format (even though you are not using VM). This is discussed in 4.2.1, “AWSCKD” on page 70. Recent versions of ICKDSF will detect this requirement automatically. If you are using an older MVS, it would be safer to use the recent stand-alone version of ICKDSF (Release 16.0A or later) to initialize emulated volumes.

If you have a working MVS (or OS/390) on your P/390, you can do the initialization under it. If not, you must use the standalone version of ICKDSF to initialize the volumes. To use the standalone ICKDSF, do the following:

Copy the ICKDSF.IPL file from one of the MVS or P/390 CD-ROMs to a Server file. This file is the ICKDSF program, in the format used by the AWSTAPE device manager. Copy the SADSS.IPL program to the Server. This is the standalone restore program, in the format used by AWSTAPE.

Use the P/390 configurator to define the devices needed to initialize your DASD volumes and restore your tapes. It should include a console (we recommend a 3215), the (emulated) tape drives for the standalone programs, the target (emulated) DASD volumes, and a tape drive for your tapes. This tape drive might be a SCSI-attached 3480/3490 type, or a channel-attached 3480/90. You can add the necessary devices to your DEVMAP.MVS configuration file, or copy DEVMAP.MVS to a new file (DEVMAP.TMP, for example), and make the changes

⁴⁴ These are produced by PKWARES, Inc.; 9025 North Deerwood Drive; Brown Deer, WI, 53223. A copy of PKUNZIP is usually included on any diskette or CD-ROM containing ZIPed files.

there. (Remember to issue the command `AWSPROF D:\MVS\DEVMAP.TMP` to make the new DEVMAP the current DEVMAP.)

An example of a configuration file might be:

```

  Addr Device Label  Atype Size  Mgr  FN/P
  >   >   >       >   >   >   >
  009 3215
  140 3380  OURMVS CKD  1770C  2   D:\OURMVS.140
  141 3380  OURCAT  CKD  1770C  2   D:\OURCAT.141
  142 3380  OURSPL  CKD  1770C  2   D:\OURSPL.142
  560 3480
  580 3420
  581 3420
                                     H
                                     G   D:\SADSS.IPL
                                     G   D:\ICKDSF.IPL

```

In this example, address 560 is a SCSI-attached tape drive that matches your dump tapes. You plan to restore the three DASD volumes listed. You will IPL from address 581 to run ICKDSF to initialize each of the emulated DASD units, then IPL from 580 to run the restore program. There is nothing special about any of the addresses shown; you can use any addresses you like.

To make things easier, go to the Environment Settings (F4 in the configuration menu) and set the IPL address to 581 (which is the tape containing ICKDSF). Save and exit from the configuration functions. Click on IPL P/390. This will start all the device managers and then attempt to IPL the indicated device. You should see a 3215 window on your OS/2 display.⁴⁵

If the IPL of ICKDSF was successful, you should see activity in the System/390 Activity window for 10-15 seconds, and then you should see PSW wait code FFFFFFFF. This indicates that ICKDSF is waiting for input. Move the mouse pointer to the the 3215 window and click (to select the window) and press RETURN. You should see the message ICK005E DEFINE INPUT DEVICE, REPLY 'DDDD, CUU' OR 'CONSOLE' OR 'CONSOLE' and so forth. The console session to initialize the first 3380 should look like this:

```

ICK005E DEFINE INPUT DEVICE, REPLY 'DDDD, CUU' OR 'CONSOLE'
ENTER INPUT/COMMAND:
console
ICK006E DEFINE OUTPUT DEVICE, REPLY 'DDD, CUU' OR 'CONSOLE'
ENTER INPUT/COMMAND:
console
ICKDSF - SA/XA/ESA  DEVICE SUPPORT FACILITIES  16.0A

ENTER INPUT/COMMAND:
init unit(140) nvfy devtyp(3380) volid(ourmvs) nocheck noval

ICK00700I DEVICE INFORMATION FOR 0C00 IS CURRENTLY AS FOLLOWS:
      PHYSICAL DEVICE = 3380
      STORAGE CONTROLLER = 3880
      STORAGE CONTROL DESCRIPTER = 03
      DEVICE DESCRIPTOR = 02
      ADDITIONAL DEVICE INFORMATION = 0000FFFF
ICK00703I DEVICE IS OPERATED AS A MINIDISK
ICK003D REPLY U TO ALTER VOLUME 0120 CONTENTS, ELSE T
ENTER INPUT/COMMAND:

```

⁴⁵ Instead of changing the IPL address in the Environmental Settings and clicking the IPL icon, you could open a Server window, switch to the P390 subdirectory and enter the command `IPL 581 CLEAR`.

```
u
ICK31061I 0120 VTOC INDEX CREATION SUCCESSFUL. VOLUME IN INDEX FORMAT
ICK061I 0120 VTOC INDEX CREATION SUCCESSFUL. VOLUME IN INDEX FORMAT
ICK01313I VOLUME CONTAINS 0 ALTERNATE TRACTS - 0 AVAILABLE
ICK01314I VTOC IS LOCATED AT CCHH = '0000 0001' AND IS 14 TRACKS
ICK00001I FUNCTION COMPLETED. HIGHEST CONDITION CODE WAS 0.
```

ENTER INPUT/COMMAND:

You should use the initialization command shown above, with appropriate unit and volser data. This performs a minimal initialization (writes the VTOC) and takes only a few seconds. You can continue to enter INIT commands to initialize all your disks.

ICKDSF communicates using wait state codes, and some of the common ones are:

- xx000022 - Unit check on IPL device
- xx000044 - Channel error on the IPL channel
- xx0000E2 - A machine check occurred
- xx111111 - Waiting for an interrupt (from the console, for example)

If you need to re-IPL, you can STOP P/390 and IPL P/390 again. If you do not STOP P/390, you will need to logically rewind the IPL tape, before IPLing it again. (Stopping the P/390 subsystem and starting it has the effect of rewinding all emulated tapes.) Use the AWSMOUNT 581 /REW command to do this.

After initializing the emulated 3380 volumes, you can restore your OS/390 tapes. Use the DFDSS program, which is IPLable from tape 580 in the system we are describing. Mount the first restore tape before starting DFDSS. (You will probably have 4mm tapes, so you need to use the internal 4mm tape drive in your system.)

Use the configuration program to change the standard IPL address to 580⁴⁶ and IPL again.

Switch to the 3215 window and press RETURN (to generate the initial interrupt). The session should look like this:

```
ADR505A DEFINE INPUT DEVICE.
input=cns1
ADR507A ENTER CONTROL CARD STATEMENT
job
ADR507A ENTER CONTROL CARD STATEMENT
msg todev=cns1,toaddr=009
ADR507A ENTER CONTROL CARD STATEMENT
restore fromdev=3480,fromaddr=b71,todev=3380,toaddr=140,volid=ourmvs
ADR507A ENTER CONTROL CARD STATEMENT
end
```

At this point the job should start. The example here restored from our SCSI-attached 3480type tape drive. Restoring from a 4mm drive is the same (with a different fromaddr value). If there is more than one tape volume,

⁴⁶ Instead of changing the standard IPL address in the Environment panel, you can enter the command IPL 580 from a Server command line; this produces some extraneous output in the window but does the IPL. You can also use an IPL pull-down window in the P/390 Manual Operations panel; however, you cannot use this for the first IPL of a S/390 program (because it does not start the I/O subsystem).

DFDSS will ask you to mount it and then DEPRESS INTERRUPT KEY. Use the P/390 Manual Operations panel (the Control pull-down) to generate an external interrupt.

The common wait codes of DFDSS include:

- 00E2 - Machine check
- 6666 - Wait for operator intervention
- EEEE - DFDSS has completed the job (You need to rewind the DFDSS tape and IPL it again to restore another tape.)
- FFFF - Wait for initial operator input.

If you used a different DEVMAP for the standalone operations, you can just switch back to your primary DEVMAP. If you used your primary DEVMAP, we suggest you return to the P/390 configuration panel and delete (or turn off) the 3215 console. OS/390 does not use it, and it will clutter up your desktop display. Place the cursor in the 3215 line and use F9 to delete it, or ALT-F3 to turn it off. Turning it off means that the device manager will not be started when the P/390 I/O system is started, and the 3215 window will not appear on your desktop. Save your configuration and exit.

Use the configuration function (F4) to change the standard IPL address to that of your OS/390 IPL volume. (This was 140 in the example shown here.) Your IPL address (and other addresses) must match addresses known to your MVS system, of course.

Double click on STOP P/390 to stop the I/O subsystem. Then start the 3270 emulator sessions you will need for your MVS console and at least one TSO session:

- For OS/2, double click on the CM/2 icon, and double click on the "Start Comm" icon. After a few seconds, the emulated 3270 windows should appear on your desktop. *You should always start CM/2 communications before IPLing OS/390.* (If you do not, the NIP routines will be unable to find a console and OS/390 goes into a disabled wait state.)
- For AIX (or OS/2 if you are using TCP/IP for all emulated terminals), start several *tn3270* sessions with AWS3274. The standard IPL shell script does this automatically, so you should not need to do it unless you have altered the IPL script or are using an unusual configuration.

Now click on IPL P/390. The S/390 Activity window should appear and you should see processor activity. After a few seconds you should see the SPECIFY SYSTEM PARAMETERS message on the first 3270 session. Select the window (with the mouse) and press Enter. (Remember that the 3270 Enter key is the right-hand Cntl key on the PC keyboard.) Continue with a normal OS/390 IPL.

Chapter 4. Device Managers and Commands

Device Managers are Server (OS/2 or AIX) programs, and are distributed as part of the P/390 Support Programs. A device manager emulates a mainframe device (or class of devices), permitting OS/390 to operate as if it were connected to the “real” device. Different device managers use different techniques to accomplish their purpose.

The list of device managers is not fixed. New device managers are being written by IBM. At least one independent vendor markets device managers. You can write your own device managers, although we do not recommend doing this.

Many device managers have the same names for P/390 and R/390 systems. This is done if the device managers provide similar functions on the two base systems. However, just because device managers have the same name on P/390 and R/390 systems, you should not assume that their functions are exactly the same. In some cases the two versions may be very similar, and in other cases they may use totally different methods to emulate their device classes.

The P/390 support program diskettes contain DOC files for all of the device managers. These files contain the most recent information about each device manager, and you should review the appropriate DOC file if you have questions about a specific device manager.

4.1 Starting Device Managers

The usual way to start the P/390 subsystem is to display the base P/390 window (Figure 15 on page 38), and then click on IPL P/390. This starts an IPL script. The script is different for the P/390 and R/390.

Among other differences, the P/390 script starts all known device managers (as listed in the IPL.CMD file). Each device manager will scan the current DEVMAP to determine whether it is needed. If there are no AWSFBA devices in the DEVMAP, for example, the AWSFBA device manager knows it is not needed and deletes itself. Starting unused device managers adds a second or two to the startup time, but is convenient in other ways. There is no need to edit the IPL command script whenever you add or delete a device type in your configuration.

The P/390 IPL script defines a function named DMSTART, and this is used to start all the device managers. For example,

```
call dmstart "AWS2821.EXE N"
```

A device manager can also be started from an OS/2 command line. The command AWSSTART is provided for this. For example,

```
AWSSTART AWS2821.EXE N
```

The double quotes are required by DMSTART, but not used with AWSSTART. You would not use the AWSSTART for normal P/390 use, but might need it in rare cases. The “N” means that a separate OS/2 window should not be created for this program.

The R/390 IPL script is not quite as sophisticated. You are expected to edit it, and comment or uncomment various lines depending on which device managers you will need. The instructions for doing this are in the appropriate DOC files,

and are sometimes noted in the descriptions in this chapter. A typical device manager start in the R/390 ipl390 file might be:

```
# $DMDIR/AWSStart dmhuron
# if [ "$?" != 0 ]
# then
#   echo "* S/370 Channel device manager failed to start."
# fi
```

Early in the ipl390 script, \$DMDIR is set to /usr/lpp/r390/bin, which is the directory where most of the R/390 device manager and support files are stored. To use the device manager in the example above, you would edit the ipl390 file, and delete the first two characters of these lines; this *uncomments* them. You might need to add additional operands to the start command, as described in the appropriate DOC file or later in this chapter. You might have, for example:

```
$DMDIR/AWSStart dmhuron -D2 -H2
if [ "$?" != 0 ]
then
  echo "* S/370 Channel device manager failed to start."
fi
```

which would start the device manager for the S/370 Channel Emulator/A adapter, and set the adapter in slot two for 4.5 MB data streaming, with high-speed transfer enabled. Other device managers for the R/390 are run as background processes. The ipl390 file contains:

```
# $DMDIR/dm3172 tok0 ent0 &
# sleep 2
```

Uncommenting these lines would start the dm3172 device manager. Two operands are specified, and the ampersand indicates that the program should be run as a background process.

4.2 Device Managers for DASD

There are two primary categories of mainframe DASD: CKD and FBA. OS/390 (and MVS) uses only the CKD architecture. VM and VSE can use either CKD or FBA. FBA is much easier to emulate using PC or RISC/6000 devices, and generally provides better performance in P/390 and R/390 operations. Since OS/390 must use CKD DASD, the characteristics and performance of the CKD device manager are critical for OS/390 operation and capacity.

4.2.1 AWSCKD

For OS/390 users, AWSCKD is the most important device manager. CKD means Count, Key, and Data. This is a type of disk format, using variable-length physical records on the disks. AWSCKD also provides ECKD (extended CKD) processing where appropriate, and the following discussion applies to CKD and ECKD. The present P/390 and R/390 systems cannot be attached to mainframe DASD units.⁴⁷ This means that all DASD must be emulated using Server disks. This is the function of AWSCKD.

⁴⁷ This is due to microsecond response times required between the channel and the DASD control units. A P/390 or R/390 cannot reliably provide the require response because their "channel" is a program (AWSCHAN) and, as such, is subject to the dispatching load and priorities of the Server.

AWSCKD emulates IBM 3390, 3380, 9345, 3375, 3350, and 3330 disk units. Only a single copy of AWSCKD is executed. It can emulate any number of these disk units, with any mixture required by the user. It uses Server (OS/2 or AIX) disks to do this. Server disks use fixed-block sectors and organization.⁴⁸

Emulating CKD (and ECKD) functions, using Server disks, is complex. Performing this function requires considerable processing in both the AWSCKD module and the AWSCHAN (emulated channel) module, and this processing often accounts for most of the Server processing required to run OS/390. There is no simple way to map CKD disk formats to the fixed-block format used by Server disks. The only general-purpose mapping possible would be based on full tracks or full cylinders of the CKD units.⁴⁹

The AWSCKD program always reads and writes an emulated full track of data, using a *track buffer*. A 3380 track is about 47K bytes; when emulating a 3380 device, AWSCKD *always* works in units of 47K bytes. Likewise, 3390 emulation works with tracks of about 57K, and 9345 emulation works with tracks of about 46K. *This is transparent to OS/390*, which continues to think it is using real mainframe DASD. If, for example, an OS/390 program needs to read a 3380 disk block that is 1K, the AWSCKD program will read the complete (emulated) track (47K), extract the requested 1K block, and send that 1K block (via the AWSCHAN channel emulator) to OS/390. If OS/390 writes a page to a paging data set, AWSCKD must first read the (emulated) track containing the targeted disk block, update the 4K block in the *track buffer*, and then rewrite the whole 47K (emulated) track. There is substantial overhead involved here, and random I/O of small disk blocks is the worst possible workload for the system.

Again, this is all transparent to OS/390. OS/390 thinks it is using real mainframe DASD, it generates whatever channel programs are appropriate for the DASD device type.

AWSCKD maintains a *track buffer* for each emulated disk unit. If your system has four 3380 and five 3390 emulated drives, AWSCKD will have nine track buffers. The track buffer acts as a cache. If a required track, from an emulated drive, is already in the track buffer, it is not read again. Changes to data in the track buffer are held until another track is referenced, at which time the track buffer is written to disk. For sequential processing, the track buffer provides relatively efficient I/O.

AWSCKD uses its own format for maintaining emulated disk data. This format is a mixture of AWSCKD's own control blocks and the OS/390 data written to the (emulated) disk. This format is not documented, and may change in future releases of the P/390 support programs. The format is the same for P/390 and R/390 systems. You cannot write Server programs (OS/2, AIX) to process data on emulated disks (unless you reverse engineer the formats, at your own risk). An emulated drive is a single, very large file to the server. A single-capacity 3380 drive, for example, appears as a single 632 MB file to the server. You can back up, restore, copy, and delete these files from the Server, because these functions ignore the internal format of the data in the file. In fact, backing up

⁴⁸ AIX uses logical blocks of 4096 bytes, built on 512-byte sectors. OS/2 uses the 512-byte sectors directly.

⁴⁹ Multi-track, track overflow, and multi-cylinder operation of CKD devices is possible, but the track and cylinder boundaries are still apparent. These functions are properly emulated by AWSCKD.

these files using Server programs is often more attractive than backing up using OS/390 programs.

Note that emulated disk drives contain whatever binary data is written by OS/390 programs. Character data is in EBCDIC. Except for the addition of control blocks needed for CKD emulation, the data is not translated in any way. If you use a Server program to inspect an emulated disk volume, you are unlikely to see any ASCII data.

The emulated types of drives have some special characteristics:

- When OS/390 requests device characteristics from an emulated disk (via a standard CCW for this purpose), the data will indicate that the volume is a minidisk, supports ECKD, has n cylinders (depending on how many you specified when you allocated the emulated volume), and is a certain model:
 - If emulating 3390, it always uses the identity of a 3390-3.
 - If emulating 3380, it will report the model corresponding to the exact number of cylinders, or the next larger model if an odd size. An emulated volume with 1-885 cylinders will be reported as a 3380-AJ4, with 886-1770 cylinders as a 3380-AE4, and with 1771-2655 cylinders as a 3380-AK4. ECKD capability is reported for all sizes of 3380s.
 - If emulating 9345, it always uses the identity of a 9345-2.
 - The P/390 allocation function will permit you to allocate an emulated DASD unit with more cylinders than the largest “real” device. However, OS/390 may not function correctly if you do this.
 - Emulated 3390 devices larger than 2518 cylinders (substantially larger than a 3390-2) require two Server files for emulation. See the AWSCKD.DOC file for more information.
 - IBM 3330, 3350, and 3375 drives can be emulated, but are not supported by OS/390. This emulation is provided for use in unusual situations and is not addressed in this document.
- They do not have “CE” tracks, used for diagnostics and spare data tracks. This means they must be initialized (ICKDSF) as minidisks. Recent versions of ICKDSF (standalone or OS/390) will detect (by reading device characteristics) that minidisk initialization is required. Other than a slightly different initialization, there is no difference in OS/390 use of the (emulated) disk. OS/390 uses it as though it were a “real” disk drive of the emulated type.
- We have used odd-sized disks (such as 500 cylinder 3380s) and know of no restrictions on the use of these with OS/390. However, we cannot state that all IBM program products fully support odd-sized disks. Most P/390 system owners tend to use emulated DASD with standard sizes for their important system volumes, and use odd-size drives for data volumes, scratch volumes, and so forth.

We have described the appearance of CKD devices in the configurator several times. An example is:

Addr	Device	Label	Atype	Size	Mgr	FN/P
>	>	>	>	>	>	>
120	3380	P390R5	CKD	1770C	2	D:\MVS\P390R5.120
or 120	3380	P390R5	CKD	1770C	2	/usr/p390/mvs/p390r5.120

When you create an emulated volume, the configurator uses the value in the Device field to determine the type of emulated device. It can be 3380, 3390, 9345, and so forth. The value in the Label field is used to create the volume label (not

the VTOC). The value in the Atype field is not used for OS/390 volumes, but is displayed as CKD. The Size field is usually the number of cylinders to be allocated. The FN/P field should always contain the fully-qualified name of the Server file.

If the Server file already exists (and contains data properly formatted for AWSCKD), the Label, Atype, and Size fields are determined from the Server file and are automatically displayed when you display the configurator panel.

Direct Allocation

Emulated volumes for use by AWSCKD are normally created using the P/390 configurator, as described in 3.5, "Working With the P/390 Configurator" on page 59. You can also use the command **CKDALC** to directly allocate a new emulated volume. The CKDALC command will not make an entry in DEVMAP, so you must do this manually. The command format is:

```
CKDALC D:\CKDDASD\TESTCKD.200 /S500 /VTEST01 /D3390
```

This would allocate a 3390 device with 500 cylinders and a volume label of TEST01. You must later initialize the volume (with ICKDSF) to create a VTOC. The Server file name, in this example, is TESTCKD.200. This does not assign the address 200 to the unit; it is just part of the name. You must make an entry in a DEVMAP in order to use the volume, and your DEVMAP entry will assign the OS/390 address. (Nevertheless, we recommend having the Server file name reflect the intended OS/390 address.) The DEVMAP entry you create must specify the full file name. When you create the DEVMAP entry, the configurator will determine if the indicated file name already exists. If it does, the existing file is used.

DASD Addresses

Mainframe disks have three or four digit (hexadecimal) addresses. The address contains channel, control unit, and specific drive identifiers. (The address information might be transformed by the I/O subsystem that operates below OS/390 on a mainframe, but we will ignore this detail.) On a mainframe, these addresses are not arbitrary -- they are determined by the hardware configuration (and I/O subsystem definition). On the P/390 system, the corresponding addresses are completely arbitrary. The device emulation programs use a simple table to translate an MVS address to an emulated device. If you had three emulated disks, you might assign them S/390 addresses of 100, 9FF, and C83. The DASD simulation programs have absolutely no sense of channels and control units.

OS/390 does have a sense of channels, control units, and "strings" of DASD units. While it can work with any addresses, we suggest you define your OS/390 I/O in terms of addresses that represent reasonable strings of DASD units. Again, this is not required. You can use arbitrary addresses and mix (emulated) disk types. However, a "normal" address convention is to your advantage, because it is less confusing to other OS/390 people.

Format Verification

It is possible for the Server file containing an emulated DASD unit to become corrupted. If this happens, the AWSCKD program is likely to fail. If AWSCKD fails, OS/390 will fail (because all its DASD has appeared to fail). The P/390 and R/390 support diskettes contain a program named CKDCHECK that may be used, when the P/390 subsystem is not running, to verify the internal format of

emulated DASD volumes. This command is explained, in detail, in *How to Verify Disk Integrity* in the Cookbook.

Performance Conflicts

The P/390 AWSCKD device manager is executed at one of two dispatching priorities. (No equivalent function is needed for the R/390 version.) By default it executes at a *normal* priority, meaning it is handled as a normal process by the OS/2 dispatcher.

OS/390 is unaware that it is using emulated DASD when it is used with a P/390 system. It assumes that its DASD units should be treated as “real” mainframe units, including all the timing characteristics of real units. Mainframe DASD operates with very tight response requirements, and OS/390 can detect failures to meet these normal responses. When this happens, OS/390 will begin error recovery and issue various HALT commands and recovery CCWs to the channel and DASD control units in an attempt to retry what it sees as a failed operation. The P/390 support programs do not always handle these recovery commands and CCWs correctly, and such retry situations must be avoided when possible.⁵⁰

Observations with early P/390 systems have always associated these problems with high OS/2 activity levels. That is, events in OS/2 prevent AWSCKD from being dispatched quickly enough to prevent “failure” detection in OS/390. We have very seldom seen this occur when the P/390 subsystem is the only workload in the OS/2 system. We have seen it more frequently when there is additional workload (such as a LAN Server, heavy communications, or large compilations) run under OS/2 while OS/390 is running.

The OS/390 symptoms of this problem are not always clear. It usually involves apparent DASD failures, often in the JES2 checkpoint data set, or in a paging data set (such as PLPA or COMMON) with complaints from ASM (auxiliary storage manager).

It is possible to run AWSCKD at a special, high OS/2 priority to avoid these problems. *We recommend doing this only if you encounter problems.* When AWSCKD runs at high priority, it introduces other effects: 3270 emulator sessions and OS/2 windows have erratic response times, among other things.

To change the priority of AWSCKD, you must edit the IPL.CMD file (in the P390 directory), using your favorite text editor. Find a line like this:

```
call dmstart "AWSCKD.EXE N"
```

and add a /PH parameter after the N. It will then read CALL DMSTART "AWSCKD.EXE N /PH". This indicates Priority High. You can also use /PN (Priority Normal), but this is the default and does not need to be specified. You must restart the P/390 subsystem for the change to be effective.

⁵⁰ OS/390's MIH (missing interrupt handler) functions address only part of the timing issues; changing values in SYS1.PARMLIB(IECIOS00) will not resolve the problems discussed here.

4.2.2 AWSFBA

This device manager is used only with VM or VSE, and is not described at length here. It emulates FBA devices with the types 3370, 3310, 0671-4, 9332, 9335, 9336-10, and FB-512. The AWSFBA.DOC file, on the P/390 support diskettes, contains more information about this device manager.

4.3 Device Managers for Simple Terminals

2.4, “3270 Connection Configurations - Overview” on page 30 discusses a number of ways to provide 3270 terminals (emulated or real) for OS/390. To help structure this discussion, we will regard any device manager that appears to OS/390 as a channel-attached 3174 non-SNA coax connection to be a “simple” terminal. This is the type of terminal that OS/390 uses for its master console, for example. Two device managers are in this group: AWS3274 and LAN3274. A channel-attached 3174 control unit would also be in this group, but is discussed separately in 4.7.1, “AWSC370 Device Manager” on page 106

4.3.1 AWS3274 (P/390)

AWS3274 allows the P/390 to emulate a local 3274 control unit and allows the P/390 to support 3270 devices such as 3277, 3278, and 3279. Practically all P/390 OS/390 systems will use this device manager.⁵¹

Each 3270 emulator session appears to OS/390 as a coax-attached local 3270. Each session is associated (by AWS3274) with an entry in the configurator, and through it, with a UCB address in OS/390. The first CM/2 3270 emulator session AWS3274 detects is associated with the lowest address AWS3274 entry in the configurator, and so forth. (The emulator functions of CM/2 will not be in future releases. The function will probably be provided by PC/3270. However, we will continue to use CM/2 in the discussion here.) If the OS/390 master console is to use an AWS3274 terminal, you must ensure that there are enough 3270 sessions available at IPL to cover the master console address.

CM/2 supports a maximum of five DFT logical terminals. Any of these can be configured as a LOCAL session, a LAN session, or a HOST session, but any HOST sessions must be defined before any LOCAL sessions. A LOCAL session is a logical terminal active on the P/390, and is what we are discussing here. A HOST session is usually a connection through a 3270 emulator adapter, connected by coax to a “real” 3174 control unit. A LAN session session is discussed later, with the LAN3274 device manager.

The CM/2 3270 emulator does not distinguish between a local and host session. Both types are configured using CM/2’s Advanced Configuration. Also, all functions available to a host session are also available to a local session, for example, GDDM graphics, font selection, cut and paste, and so forth. The only difference between a local and host session is the way 3270 data streams are processed. For a host session, the data stream is transmitted between the PS/2 and control unit over coax. The coax is attached to a 3270 Connection adapter which uses a device driver (DFTDD.SYS) to interpret the data streams and perform the appropriate inbound or outbound function. For a local session, the

⁵¹ In principle, you could use a channel-attached 3174 (P/390 and R/390) or the LAN3274 device manager (P/390 only) to provide an OS/390 master console and local TSO terminals. In practice, we have never seen a P/390 system that does not use the AWS3274 device manager.

P/390 session is physically in the same PS/2 as CM/2, so no coax or 3270 Connection card is needed. Instead, there are hooks in CM/2 to route 3270 data streams to and from the AWS3274 device manager, using CM/2's outbound and inbound router functions.

The normal setup for CM/2 3270 sessions, for use with P/390 OS/390, is described in 3.1.3, "CM/2 Installation" on page 46. In the FN/P field at the right, you may specify the name of the window. This will appear at the top of the displayed emulator window, but has no other effect. When using CM/2, there is a maximum of five AWS3274 devices, because this is the maximum number of 3270 sessions that CM/2 can produce. If you define more than five AWS3274 devices in the configurator, the higher addresses will never be used. If the OS/390 console address is among these higher addresses, it will not be found and OS/390 will enter a wait state.

The initial definition of the CM/2 emulator sessions is described in 3.1.3, "CM/2 Installation" on page 46. You can manipulate the CM/2 emulator sessions by using the configurator "Update 3270 LT Sessions" (F7) screen to specify which of the five sessions are local and which are host sessions. The formal documentation for P/390 contains instructions to define five CM/2 sessions. We found this was excessive and cluttered up the OS/2 display. We normally defined three sessions: one for the OS/390 console, and two for TSO (or CICS) use. (We did not have a 3270 Connection adapter in our systems, and had no CM/2 connection to a "real" host. If we had such a connection, we would have defined another one or two CM/2 emulator sessions for it.)

CM/2 must be running (with the local 3270 windows displayed) before IPLing OS/390. If it is not, NIP will not find a console and go into wait state 0007.

4.3.2 AWS3274 (R/390)

AWS3274 allows the R/390 to emulate a local 3274 control unit and allows the R/390 to support 3270 devices such as 3277, 3278, and 3279. Practically all R/390 OS/390 systems will use this device manager.⁵² The R/390 AWS3274 device manager uses TCP/IP (running in the Server, not in OS/390) to work with 3270 emulator sessions. These emulator sessions are typically *telnet* or *tn3270* client programs.

Each 3270 emulator session appears to OS/390 as a coax-attached local 3270. Each session is associated (by AWS3274) with an entry in the configurator, and through it, with a UCB address in OS/390. The first 3270 emulator session AWS3274 detects is associated with the lowest address AWS3274 entry in the configurator, and so forth. If the OS/390 master console is to use an AWS3274 terminal (which is the normal situation), you must ensure that there are enough 3270 sessions available at IPL to cover the master console address. (The CD-ROM OS/390 systems are set up to handle this function.)

R/390 AWS3274 supports any reasonable number of 3270 sessions, and is used with both local sessions (in the same system with the R/390) and remote connections. Again, these sessions appear to OS/390 to be coax-attached local 3270s. An OS/390 UCB (and a line in the P/390 configurator) is required for each session. AWS3274 does not distinguish between 3270 sessions in the R/390 system and sessions from remote TCP/IP users. It does not distinguish between

⁵² In principle, you could use a channel-attached 3174 to provide an OS/390 master console and local TSO terminals.

multiple sessions on one workstation and sessions on different workstations. Every 3270 session uses a configurator line entry and an OS/390 UCB.

The S/390 addresses (and UCBs) are used in ascending order, whenever a new 3270 telnet client connects. The addresses used for closed sessions are reused. In general, you will define a reasonable number of connection addresses and AWS3274 will use these as needed. The CD-ROM OS/390 systems have 96 addresses already defined for local 3270s, at addresses 700-73F and 900-91F. This may be excessive for an R/390 and you may not want to define configurator lines for this many connections.

There are operational considerations that must be noted:

1. Your OS/390 system has one (or more) master consoles defined. The CD-ROM systems, for example, look for a master console at address 700. A 3270 (in some emulated form) must be connected to this address, or IPL will fail. Someone must initiate the telnet connection that connects to the MCS address (a) after AWS3274 is started (so there is a telnet server available), but (b) before OS/390 IPL gets to the point where it senses the MCS console. (The IPL command script supplied with the R/390 diskettes automatically starts two telnet sessions at the right place in the script.)
2. Suppose you define to OS/390 (using HCD, of course) local 3270 addresses 380-39F, and you assign the MCS console to address 39F. Furthermore, assume you have defined 32 AWS3274 devices in DEVMAP, to cover these addresses. If these are your only terminals, you cannot IPL OS/390 unless you first start AWS3274 and then initiate 32 telnet sessions to it. The first telnet session will be assigned UCB address 380, the second session will be address 381, and so forth. Unless there are at least 32 telnet sessions started *before IPL*, there will be no terminal at address 39F and OS/390 will enter wait state 007 because it cannot use its console.
3. The OS/390 master console should have one of the lowest 3270 addresses that are defined for AWS3274 in the configurator. OS/390 may have lower addresses for 3270s, but if they are not defined in the configurator, they do not matter.
4. AWS3274 cannot create a 3270 session. Every telnet session must be initiated externally. The `ipl390` script is, in this sense, an external initiator and it starts two telnet sessions. These are normally used for the OS/390 master console and a TSO session.
5. AWS3274 does not distinguish between telnet sessions. An external telnet client -- someone on your network -- might connect to AWS3274 before you start your first telnet session (and before you IPL OS/390), and thus become the master console. If you use the CD-ROM system, and the IPL script on the R/390 diskettes, this is unlikely because there is a very small time window of exposure.
6. When we say *telnet* here, we mean a form of telnet that emulates 3270 terminals. There are often special telnet names for these functions, such as `tn3270`, `xant`, and so forth.

A configurator entry for AWS3274 terminals is simple, like this (assuming the device manager code for AWS3274 is 3):

Addr	Device	Mgr	FN/P
>	>	>	>
700	3278	3	
701	3278	3	
702	3278	3	
...			
71F	3278	3	

Do not use the FN/P field.

Remember that the AIX TCP/IP cannot share a LAN adapter with OS/390 TCP/IP. AWS3274 uses AIX TCP/IP. If you want users to telnet directly to OS/390, you must add another LAN adapter and use the LCS3172 device manager to provide that connectivity.

The AWS3274 telnet interface uses a nonstandard port address, offering some protection against accidental connections. Commands to connect various telnet-3270 emulators are discussed in 5.15.1, "xant 3270 Emulator" on page 151.

Note that the telnet port address is different between R/390 AWS3274 and P/390 LAN3274. In both cases the intention was to pick a random, nonstandard port address, different than the standard TCP/IP telnet daemon port address.

4.3.3 LAN3274 (P/390 only)

This device manager provides two connection methods: a special use of NetBios, and TCP/IP telnet (tn3270-type) sessions. The TCP/IP connections operate exactly as described for AWS3274 (R/390). The only difference is that the LAN3274 device manager is specified in the P/390 configuration. The NetBios function is used only by LAN3274 (P/390) and is described in this section.

Note that LAN3274 devices are seen as local 3270 devices by OS/390. You could, for example, use LAN3270 terminals as OS/390 master consoles.

The P/390 support programs provide a unique way to provide "local" 3270 terminals by using OS/2 workstations on a LAN. This solution is limited to OS/2 (and requires CM/2); it does not apply to DOS or Windows workstations, or to R/390 Servers. It needs a NetBios environment, which implies a local LAN connection (with no routers involved). To OS/390, the connections appear as channel-attached 3174 coax connections, just like those from the AWS3274 device manager.

To use this device manager, you must install several files from your P/390 diskettes on the client workstations (that is, the OS/2 systems that will be your LAN 3270 terminals).

On the P/390 system, ensure that enough 3278 terminals are defined (at addresses known to OS/390 as local 3270 terminals, of course). Assuming LAN3274 is device manager code 4, the configurator entries would appear like this:

Addr	Device	Label	Atype	Size	Mgr	FN/P
>	>	>	>	>	>	>
700	3278				3	MVS Console
702	3278				3	Local TSO
703	3278				3	Local TSO
704	3278				4	LAN1

705	3278	4	LAN2
706	3278	4	LAN3
707	3278	4	LAN4

In this example of a configuration display, 700-703 are defined as local (in the host P/390 system) terminals (device manager 3 is AWS3274), and 704-707 are LAN terminals as described in this section (device manager 4 is LAN3274). There is no particular limit to the number of LAN3274 devices you can define. Remember that you must restart the P/390 subsystem (and reIPL OS/390) after making changes to the DEVMAP.

As the LAN3274 detects connections from client OS/2 CM/2 emulators, it will assign them to AWS3274 sessions, starting at the lowest address. Each emulator session will have a different OS/390 (UCB) address. A given client might have several sessions defined for AWS3274 LAN connections; each of these sessions will be seen at a different address. A given client might not always have the same UCB addresses, depending on which clients are waiting when the P/390 subsystem is started, and the order in which other clients are started.

To start ASW3274 setup, check the System Environment panel of the P/390 configurator (F4). The first line contains the Local Area NetID. Enter a unique name (up to eight characters). Client stations (running the 3270 emulator sessions provided by the process we are describing) must specify this same NetID in their CM/2 configuration.

On your P/390 system, open an OS/2 command window. Enter **C:\IBMCOM\LAPS** to start the LAPS configuration process. (You may have installed it in a different directory or disk.) You need to verify or reconfigure several options. You need to configure IEEE 802.2 and NetBios for your LAN adapter. (The following description assumes a token-ring adapter; you may need to adapt it to other LAN types.)

1. Click on "Configure"
2. Select "Configure LAN transports" and then click on "Continue"
3. Select the appropriate adapter (token ring, in our case) and, if necessary, "Add" it.
4. If necessary, select and "Add" both IEEE 802.2 and NetBios protocols for this adapter. (You must do one at a time. The lower box on the screen displays which protocols are already added for the selected adapter.)
5. In the lower box, select the token-ring adapter and the NetBios line under it. Then click on "Edit."
6. On the NetBios parameters screen change:

Maximum sessions	64	(default was 40)
Maximum commands	64	(default was 95)
Maximum names	30	(default was 21)

(You may need larger numbers if you are running additional (non-P/390 related) NetBios applications on the OS/2 side of your P/390.) Click on "OK" to save the changes.

7. In the lower box, select the token-ring adapter and the IEEE 802.2 line under it. Then click on "Edit."
8. On the 802.2 parameters screen change:

Maximum link stations	64	(default was 8)
Maximum SAPs	7	(default was 0)
Maximum number users	7	(default was 3)

(Again, you may need larger numbers if you are running additional (non-P/390 related) LAN applications on the OS/2 side of your P/390.) Click on "OK" to save the changes.

9. Exit LAPS. (Click on "OK" "Continue" "Update CONFIG.SYS" and so forth.)
10. Shutdown and restart OS/2.
11. Before restarting OS/2, you might run the PS/2 configuration program to verify that the interrupt level for your token-ring adapter is not set to 2. If it is, change it to 3 (or any valid number other than 2.)

On each client system:

1. Verify that OS/2 (2.1 or later) and CM/2 (1.1 or later) are installed. Verify that CM/2 has 3270 emulation installed and enabled. ("Setup," "OK," verify that "Token Ring or other LAN" is selected and, "3270 emulation" is selected. Other emulators and protocols may also be present.)
2. Insert P/390 diskette 1 and (using an OS/2 command window) enter **A:INSTALL CLIENT**. Follow the prompts. A C:\P390\ subdirectory will be created with the following files in it:

```
AWSCMLT.EXE
AWSCMLT.ICO
LTRENAME.EXE
AWSDFTN.DLL
AWSDFTS.DLL
AWSNTCA.DLL
AWSSTCA.DLL
```

A new desktop icon "Update 3270 LT Sessions" will appear.

3. Double click on this icon.
4. Select one (or more) of the 3270 sessions listed and change the session type from HOST or LOCAL to LAN. Enter your P/390 NetID name in the appropriate box for each selected session.
5. Click on "Save"
6. Make the same LAPS changes that were described above for the Server P/390 system.
7. You may want to customize the 3270 emulator sessions on the clients, as described in 3.1.3, "CM/2 Installation" on page 46.
8. Shutdown and reboot OS/2.

To use the new 3270 sessions, start CM/2 on the workstation. After a few seconds, one or more 3270 windows should appear. (If not, use the Window List to find them.) Select the 3270 session that was defined as type LAN. You should see your OS/390 logo, and should be able to logon to TSO. (If not, check your LAN connections and be certain VTAM is up, the 3278 addresses (which OS/390 thinks are local, channel-attached 3270 terminals) are correctly defined to the configurator, OS/390, and VTAM.)

Please note that the LAPS parameters shown above represent minimum values that are known to work in many situations. You may specify higher values.

TCP/IP Connections

LAN3274 also accepts telnet connections, where the client provides 3270 emulation functions. In order to use this, TCP/IP must be installed and operational on the Server OS/2 system. The telnet connection is to the LAN3274 device manager, which is an OS/2 program. OS/390 is unaware that TCP/IP is being used; it sees only local 3270 terminals.

LAN3270 uses a nonstandard port address for the telnet connection; the client must specify this port address. Examples client commands are:

```
TN3270 1.234.56.789 -p 7490 -ext (OS/2)
PMANT 1.234.56.789 -p 7490 -ext (OS/2)
XANT -port 7490 -ext 1.234.56.789 (AIX)
TELNET 1.234.56.789 7490 (VM)
```

In these examples, 1.345.56.789 is the IP address. You can specify a system name instead of using the IP address if your TCP/IP connection can resolve the name. The LAN3274 port address is 7490. This is hardcoded in LAN3274, and you must specify it as part of the client connection command. (You can override the port address by changing IPL.CMD; see the LAN3274.DOC file for details.)

Connection Controls

You may want to control which LAN terminal (NetBios or TCP/IP) connects to which LAN3274 emulated terminal. For example, you may want a certain terminal to always connect to the address used for a secondary MCS console, even if the terminal is not available at IPL time. This control is provided by operands in the FN/P field. The controls are one of the following:

- **Blanks or comments** - this means that this address is used for either a NetBios or TCP/IP connection, and may be assigned to any user whose terminal is not directed to a specific connection.
- **NETBIOS** - this address will be used only for NetBios connections, but is not reserved for a specific NetBios terminal.
- **TCP/IP** - this address will be used only for TCP/IP connections, but is not reserved for any specific IP address.
- **IP address, in dotted-decimal format** - this address is reserved only for a telnet connection from the indicated IP address.
- **MAC address** - this address is reserved for a NetBios connection from the indicated MAC address. (You can use the AWSCMLT command to display the MAC address. AWSCMLT is installed when you install CLIENT P/390 software on a remote OS/2 CM/2 system.)

The configurator data might look like this:

Addr	Device	Label	Atype	Size	Mgr	FN/P
>	>	>	>	>	>	>
700	3278				3	MVS Console
702	3278				3	Local TSO
703	3278				3	Local TSO
704	3278				4	LAN1
705	3278				4	LAN2
706	3278				4	9.12.2.70
707	3278				4	9.12.0.89
708	3278				4	NETBIOS
709	3278				4	NETBIOS
70A	3278				4	400000123456
70B	3278				4	400000123654
70C	3278				4	TCPIP

In this example:

- Addresses 700-703 are managed by AWS3274 and are windows in the P/390 Server display. The contents of the FN/P field are displayed in the emulated 3270 window title line.
- Addresses 702-705 are managed by LAN3274 and are used for any connecting terminal that does not have a specific IP or MAC address reserved in a later entry. The contents of the FN/P field (since they cannot be parsed as an IP or MAC address) are ignored.
- Addresses 706-707 are managed by LAN3274 and are reserved for the indicated IP addresses.
- Addresses 708-709 are managed by LAN3274 and is used for any NetBios connection for which there is not a specific MAC entry.
- Addresses 70A-70B are managed by LAN3274 and are reserved for the specific NetBios MAC addresses shown.
- Addresses 70C-70D are managed by LAN3274 and is used for any IP connection for which there is not a specific IP address entry.

When a terminal attempts to connect to LAN3274, all the configurator entries are scanned to determine if there is a dedicated connection for the connecting terminals. If there is not, it is connected to the lowest available general-purpose LAN3274 address.

No reasonable system would have a connection list such as the one shown above; it is shown merely to illustrate the potential controls.

4.4 Device Managers for Tape Drives

Various ways of attaching tape drives are discussed in 2.5, "Tape Configurations" on page 34. The device managers associated with these are AWSTAPE, SCSI32x0, AWS34x0, and AWSC370. This last one, AWSC370, is for channel-attached drives and is discussed later in 4.7.1, "AWSC370 Device Manager" on page 106. The others are discussed here. Also described here is ISITAPE, a SCSI tape manager from an independent software vendor.

There is a general discussion of tape drives and tape usage in 5.3, "Tape Drives" on page 127.

4.4.1 AWSTAPE Device Manager

AWSTAPE emulates 3420 and 3422 drives using Server disk files. A file corresponds to a tape volume. A file can reside on either disk or diskette. It could also reside on a file server.⁵³ Unlike P/390 emulated DASD files, the a specific size for the tape file is not allocated. Normal OS/2 or AIX file expansion occurs as the "tape" is written.

To configure a 3420 (or 3422) emulated tape, use the Update System Devices menu and enter the OS/390 address of a 3420 or 3422 tape device in the Addr field, the MGR code for AWSTAPE in the Mgr field, and either a fully qualified OS/2 filename in the FN/P field or blanks. For example:

⁵³ File servers are not practical for emulated DASD volumes unless very slow response can be tolerated. However, a file server may provide acceptable performance for an emulated tape volume.

Addr	Device	Mgr	FN/P
>	>	>	>
>580	>3420	> G	>E:\TAPE\IPLDDR.580
>581	>3420	> G	>

You can use the OS/2 AWSMOUNT command to change the filename that is associated with the emulated tape or to issue Rewind and Rewind Unload commands. In this example, two tape drives are defined at addresses 580 and 581. Your OS/390 system must have 3420 (or 3422) drives defined at the corresponding addresses, of course. In this example, drive 580 has a tape volume permanently mounted (although this can be changed with AWSMOUNT). Drive 581 has no volume mounted, and AWSMOUNT must be issued before the drive is used. The AWSMOUNT command, with an example, is described in 4.7.6, "AWSMOUNT Command (P/390 and R/390)" on page 113.

You can have a maximum of 16 AWSTAPE devices, and the low-order address digit must be different for each one. For example, you cannot have AWSTAPE devices at addresses 580 and 590, because both have the same low-order digit ("0") in the address.

If you routinely use AWSTAPE devices, you will probably mount a number of tape volumes on a given drive during the course of a day. You can have any number of emulated AWSTAPE volumes on your Server disks, limited only by the amount of disk space available. An emulated tape is mounted on an emulated tape drive (with the AWSMOUNT command) in a way similar to mounting a real tape on a real tape drive.

Emulated tape volumes are Server files. The files can have any valid Server names. If you plan to use AWSTAPE emulated devices, you should decide on a naming convention for your emulated tape volumes. Something such as *volser.TAP* might be used. The TAP suffix would be sufficiently unique to identify the file as an AWSTAPE volume, and using the tape volser as the file name offers a meaningful correlation to OS/390 JCL and mount messages.

AWSTAPE Operational Characteristics

The PC file is opened when data is read or written to the tape, or when tape positioning commands, such as FORWARD SPACE FILE or FORWARD SPACE RECORD, are issued. The PC file is closed when a REWIND or REWIND UNLOAD is issued. If the PC file has the read-only attribute set, it appears as read only (no ring) to the P/390.

For a P/390 (but not for an R/390) the file can be on diskette; diskettes are convenient for storage or moving data to another P/390 system. *Remember to use the AWSMOUNT xxx /RUN command before removing the diskette.* In general, we suggest that you do not use diskettes for AWSTAPE output. Operation is slow (about 12 minutes (P/390) to write a full diskette, versus 6 seconds for the same job writing to disk) because the AWSTAPE device manager must constantly verify that enough space is available to write the next block.⁵⁴ If

⁵⁴ This is necessary because there is no way to duplicate the end-of-tape (EOT) condition associated with a "real" tape drive. The EOT condition warns that the end of tape is near, but the program can continue writing the current block (plus a few more, for labels). AWSTAPE must verify the space available frequently because other Server programs might be writing to the same disk or diskette and consuming free space.

you need an AWSTAPE file on diskette, you can create it on disk and later (with Server commands) copy it to diskette. Reading an AWSTAPE file from diskette took 4 minutes (for a 1.4MB file).

If your OS/390 tape drive definitions include (or default to) OFFLINE=NO, then MVS will consider any tape drives it detects at IPL to be ONLINE.⁵⁵ OS/390 will detect all AWSTAPE tape drives defined in your P/390 configuration file (if they correspond to tape addresses known to OS/390, of course), and they are ONLINE after IPL. When OS/390 needs to mount a tape (in response to a submitted job), it will sense/read tape drives that are ONLINE to determine if the requested tape has been premounted. If AWSTAPE drives are defined (via the P/390 configuration file) with no associated file (as in our example above), OS/390 will produce a variety of error messages when it attempts to sense/read these drives. A typical message is:

```
IOS000I 584,01,FPR,02,0E40,,01000000 .....
```

You can ignore these messages. You will sometimes get a message such as this:

```
*06 IEC613A OGDEN5,      ,583,TAPEAA TAPE POSITION ERROR
      REPLY 'R' RETRY OR 'U' CONTINUE WITH ABEND
```

This may occur before you issue the AWSMOUNT command. After you issue your AWSMOUNT command, reply "R" to this message. OS/390 should then issue the IEF503 message about an incorrect label, and you would proceed as usual.

If your output disk (or diskette) becomes full, AWSTAPE signals an end-of-tape (EOT) condition. OS/390 will call for another tape volume, and you must use the AWSMOUNT command to "mount" another volume (probably a diskette, but possibly a file on another disk). This process creates a multivolume file, just as it would if "real" tapes were used. AWSTAPE cannot fully emulate the EOT environment of a tape drive. There may be situations that cannot be handled; the results (to OS/390) would be similar to a tape running off the end of a reel before the last blocks were written. We suggest that, when possible, you do not use AWSTAPE if your expected tape size is larger than the free space on your disk (or diskette).

A REWIND UNLOAD command does not actually unload or demount the tape. It sets the sense bits so that a SENSE (or READ or WRITE) command reports "intervention required." However, if you issue other tape commands, such as REWIND or FSF, they succeed. The tape status is reset, and subsequent WRITE or READ commands proceed normally. This difference from real 3420 operation should not matter, although special programs (such as DITTO) might exhibit slightly unusual operation.

When data is written to the middle of an emulated tape, any data following it is erased and the PC file representing the tape is truncated. This allows the file to grow and shrink as the "tape" is reused. Programs that depend on the ability to rewrite records in the middle of a tape will not work with the AWSTAPE 3420 tape emulator.⁵⁶

⁵⁵ If you rework your MVS configuration, we suggest that you might want to specify OFFLINE=YES for your tape drives.

⁵⁶ Rewriting records in the middle of a tape has always been a questionable practice, and is very rarely done.

AWSTAPE does not support read backwards channel commands. *This may produce errors when attempting to process emulated tape volumes with standard labels.* OS/390 standard label processing performs read backward operations under some circumstances.⁵⁷ Our limited experimentation indicates that this may occur when opening an existing labeled tape for rewriting. In cases where we simply created an SL tape, and later (in other jobs) mounted and read it, we did not encounter problems.

The most common error is “equipment check,” which is set if the file mounted is not in AWSTAPE format. In other words, you probably AWSMOUNTed the wrong file. If that is not the case, then the file has been damaged and cannot be used with AWSTAPE.

AWSTAPE File Format

The AWSTAPE device manager automatically adds (when writing) or removes (when reading) extra control information in the file that it is using to emulate the tape. The control information helps AWSTAPE to emulate normal 3420/3422 operation. You are not concerned with these details unless you want to examine (read or write) an AWSTAPE-formatted file with a Server program. If, for example, your OS/390 program writes tape output (and you assign that tape drive to the AWSTAPE device manager) and you want to read the resulting file with an OS/2 program, the OS/2 program must know how to deal with this extra control information.

The tape file consists of a series blocks. Each block begins with a 6 byte header followed by a (possibly null) data block. The header block specifies the length of the data block. Currently, data blocks must not be longer than 4096 bytes. This is not a restriction on OS/390 programming, because AWSTAPE will automatically create or combine multiple blocks as needed.

The header is defined as follows:

```
DCL 1 T_Header,
    2 t_size fixed(16), /* Size of the following block */
    2 t_psize fixed(16), /* Size of the previous block */
    2 t_flags, /* Control flags. */
    3 t_Newrec bit, /* start of new record */
    3 t_eof bit, /* end-of-file mark */
    3 t_Endrec bit, /* End of a record */
    3 * bit, /* Must be zero */
    3 * bit, /* Reserved */
    3 reserved bit(11); /* Must be zero */
```

T_Size and T_PSize are two byte integers in PC format; that is in little end-ian form. They currently must be less than or equal to 4096 bytes. (This is the maximum size currently supported; however do not depend on it not getting larger.) Because tape records can be longer than 4096, T_Newrec and T_Endrec define the mapping of these internal tape blocks to emulated tape records.

If both Newrec and Endrec are on, the record is entirely within one block; it is t_size bytes long. If a record is longer than one block but fits in two blocks, then the first has T_Newrec set, and the second has T_endrec set. If the record is

⁵⁷ We could not determine *exactly* what triggers the read backwards function; the authors would appreciate a note from any reader who understands *exactly* what controls this action.

three blocks or longer, the middle blocks have both newrec and endrec turned off.

T_EOF is set for a filemark; if it is on, Newrec and Endrec should be off and T_size should be zero. An EOF block should be followed by header block, or another EOF block, or it should be at the end of the file.

T_size and T_Psize do not include the length of the header block. Thus, the first block on the tape has T_Psize = 0. An EOF block has T_size equal to zero, and the second of two EOF blocks in a row has both T_size and T_Psize = 0.

The Server files used for emulated tapes contain S/390 data. There is no automatic EBCDIC to ASCII conversion. The tape data can be used by Server programs, but these programs must process the extra control information in the files and perform any required data conversion. These tasks are not difficult (although floating point or packed decimal data conversion is messy), and emulated tapes are a convenient way to exchange data between Server and MVS applications. (Emulated 1403 printers and emulated 2540 card readers can be used the same way, and provide automatic EBCDIC-ASCII conversion in some cases. These may provide an easier way to send data back and forth between the Server and OS/390.)

4.4.2 SCSI34x0 and AWS34x0

The SCSI3480 and SCSI3420 device managers are very similar. One emulates a 3480 drive -- that is, it accepts the CCWs and generates the sense codes corresponding to a 3480 unit. The other emulates a 3420 drive in a corresponding way. Either of these device managers can use any one of the following types of physical tape drives:

- SCSI-attached 3480-type drive
- SCSI-attached 3490-type drive
- SCSI-attached 3420-type drive
- SCSI-attached 4mm drive
- SCSI-attached 8mm drive (R/390 only)

This may be confusing. The SCSI3480 device manager, for example, can use a 3420-type drive and still emulate (to OS/390) a 3480 drive, within the constraints of what can be done with the physical drive. Or it can use a 3490 drive (36 tracks) to emulate a 3480 (18 tracks) drive. The SCSI3420 device manager can, for example, use a 3490-type drive to emulate a 3420 unit. OS/390 should have 3480 or 3420 units generated to use these device managers. The OS/390 CD-ROM systems, for example, have 3480s generated at addresses 560-57F and 3420-8s generated at addresses 580-59F.

These device managers, when using 3420/3480/3490 types of drives, can create or read tapes used by corresponding mainframe tape drives. Many P/390 and R/390 systems have at least one of these SCSI drives, usually a 3480-type unit, in order to exchange tapes with mainframes.

For the P/390, each of these device managers can drive only a single device. For example, SCSI3480 may be used only once (in a given DEVMAP) to emulate a single 3480 drive. AWS3480 is, in essence, simply another copy of SCSI3480, and can be used to emulate a second 3480 drive (using another SCSI-attached tape unit, of course). Likewise, AWS3420 is used to emulate a second 3420 drive. The AWS34x0 modules are not *exact* copies of the SCSI34x0 modules; you cannot simply copy and rename these modules to create more emulated drives.

(The ISITAPE device manager, described later, does not have the single-device restriction.)

For the R/390, the SCSI34x0 device managers can handle more than one tape drive.

The implementation of these device managers is quite different between the P/390 and the R/390. The P/390 uses OS/2 device drivers, specified in CONFIG.SYS, for low-level device control. The R/390 uses standard AIX functions for low-level device control. OS/390 tape usage is very different than AIX tape usage, and not all tape drives used with AIX are suitable for use with R/390 OS/390.

Installation (P/390 only)

There are a number of steps involved in defining a drive for use by P/390 systems. These are:

1. Set the SCSI address of the unit. This is a number in the range 0-6. (We suggest avoiding the address 0, as this may add confusion with the internal 4mm drive whose SCSI address is also 0, but on a different SCSI channel.) You must not have duplicate SCSI addresses on the same SCSI adapter (or adapter channel in the case of the IBM RAID or F/W SCSI adapters).
2. Connect a cable between the SCSI drive and the Server, boot the server, and run whatever reference diskettes or RAID tool is needed to make the new device known to the Server hardware.
3. Boot OS/2 and edit CONFIG.SYS:

- Verify that these two lines are already in CONFIG.SYS:

```
BASEDEV=IBM2SCSI.ADD  
BASEDEV=OS2SCSI.DMD
```

- Find the lines similar to this:

```
REM DEVICE=D:\P390\SCSI3420.SYS 00,00
```

and change it as described here.

4. The "00,00" parameter has two parts, "id,ai." The id element is the SCSI address of the tape drive; you selected this when you set up your tape drive. (The 4mm drive is already installed in a typical Server, and has SCSI address 0.) The ai element is the SCSI adapter index, and is relevant if you have more than one SCSI or RAID adapter. Typically, adapter 00 is the SCSI adapter in the lowest-numbered slot, 01 is the next SCSI adapter, and so forth. However, we suggest you use the /T operand (described later in this section) to verify the SCSI adapter index numbers if you have more than one adapter.
5. There should be one CONFIG.SYS line for each SCSI tape device. The maximum would be:

```
DEVICE=D:\P390\SCSI3480.SYS id,ai  
DEVICE=D:\P390\SCSI3420.SYS id,ai  
DEVICE=D:\P390\AWS3480.SYS id,ai  
DEVICE=D:\P390\AWS3420.SYS id,ai
```

Any unused lines in CONFIG.SYS should be removed or commented out.

6. If you have, for example, the standard 4mm drive on the base RAID adapter and a 3490 drive at SCSI address 4 on a separate SCSI adapter, you would have:

```

DEVICE=D:\P390\SCSI3480.SYS 00,00 (4mm)
DEVICE=D:\P390\AWS3480.SYS 04,01 (3490)

```

7. If you use either of the AWS34x0 device managers, edit file AWSCTYPE.MGR (in the P390 directory) and, if necessary, add the lines:

```

AWS3420 3420
AWS3480 3480

```

This step causes the AWS3420 and AWS3480 device managers to be displayed at the bottom of the P/390 configurator panel, and assigns a one-character device code for them. Do not add the lines if they are already in the file. Do not reorder the other lines in the file. Next edit the file IPL.CMD and add one or both of these lines (if it is not already there) to the section containing the other dmstart commands:

```

call dmstart "awsdev.exe n /cu=AWS3420 /dd=AWS3420"
call dmstart "awsdev.exe n /cu=AWS3480 /dd=AWS3480"

```

8. Add the appropriate lines in the P/390 configurator. Note that there should be a different device manager code for each of these device managers. For example,

Addr	Device	Mgr	FN/P
>	>	>	>
>560	>3480	>H	> (SCSI3480)
>561	>3480	>0	> (AWS3480)

9. You must reboot OS/2 to make changes to CONFIG.SYS effective, and restart the P/390 subsystem to make changes to the P/390 configuration effective.

Determining the SCSI adapter index, needed to set the address parameters in CONFIG.SYS, can be difficult. If there is a problem, edit CONFIG.SYS and place a /T parameter in the first SCSI34x0 line (and comment out any other SCSI34x0 or AWS34x0 lines). For example:

```

DEVICE=D:\P390\SCSI3480.SYS 00,00 /T
REM DEVICE=D:\P390\AWS3480.SYS 04,01

```

Reboot OS/2. During startup it will display something like this:

```

SCSI3420: P/390 SCSI Tape Device Driver
          Total sequential devices detected = 03.
          Available tape unit = 00,00
          Available tape unit = 04,01
          Requested tape unit = 00,00
          Allocated tape unit = 00,00
>> Press ENTER to continue ....

```

Note the addresses of the available units and use these addresses when you edit CONFIG.SYS again. (In this example, there were three sequential device known to OS/2; one had already be claimed by some other device driver started before SCSI3480.)

Note that you must have the tape unit turned on when you boot the server AND when you IPL OS/390. If it is not on at both times, it cannot be accessed and used by the device managers. Once OS/390 is running, you can turn the tape drive off (until you want to use it, of course). If the tape drive is turned off when OS/2 is booted, you may see warning messages from OS/2. If it is turned off when OS/390 is IPLed (or if it was off when OS/2 was booted) you will see a warning window about P/390 Error, CU=SCSI348, DD=SCSI3480. If you do not

plan to use the tape drive, simply click OK to remove the warning window, and continue operation.

By the way, the IBM part number for a SCSI cable with a large Centronics-type (8 bit) connector on one end (to fit most external SCSI tape drives, and a small F/W connector on the other end (to fit the P/390 RAID adapter) is 32G3915.

Installation (R/390 only)

For the R/390, the device name of the tape drive must appear in the parameter field. Multiple tape drives would each have a different name, of course. The R/390 SCSI3480 (and SCSI3420) device manager can be assigned to multiple units in the configurator; it does not have the single-unit-per-manager limitation of the P/390.

Addr	Device	Mgr	FN/P
>	>	>	>
>560	>3480	>I	> /dev/rmt0
>561	>3480	>I	> /dev/rmt1

A tape drive can be shared between OS/390 and AIX. To return control of the drive to AIX, the following steps may be needed:

- Vary the unit offline to OS/390
- Use AWSMOUNT to issue a RUN (Rewind UNload) command
- Remove any tape from the unit.

To use the drive again with OS/390, be certain no AIX program is using it and vary it online to OS/390.

At the time this was written, it was necessary to have a tape cartridge (or reel) loaded and ready before OS/390 IPL in order for the SCSI34x0 device manager to function properly.

4.4.3 ISITAPE Device Manager (P/390 Only)

ISITAPE is available from Interprocess Systems, Inc., 11660 Alpharetta Highway Suite 455, Roswell, Georgia, 30078. Their telephone number is 770-410-1700 (fax 770-410-1773). This device manager has these characteristics:

- It will emulate 3420, 3480, 3490, and 3490E drives.
- It will support multiple (two or more) SCSI drives, with any mixture of emulated device types.
- Only one device driver (in CONFIG.SYS) is needed.
- The same device driver supports XTAPE, making it possible to coordinate the sharing of a a SCSI tape drive between OS/390 and the OS/2 environment. XTAPE (from the same vendor) is described in 5.4, "XTAPE Product (P/390 Only)" on page 131.
- It supports a wider range of tape drives, from a variety of vendors, than the SCSI34x0 device managers. See 5.3, "Tape Drives" on page 127 for a discussion of SCSI tape drives.

If your use of SCSI tape drives is minimal, perhaps limited mostly to product or fix installation using the 4mm drive, you should not need ISITAPE. If your SCSI tape usage is extensive, especially if you may require more than two SCSI drives or you purchase the newest SCSI drives, you are more likely to need ISITAPE. A

product package containing ISITAPE, XTAPE, and an OS/2 tape utility is in the \$500-\$600 price range. (XTAPE is described in 5.4, "XTAPE Product (P/390 Only)" on page 131; it also works in conjunction with the SCSI34x0 device managers and their CONFIG.SYS drivers.)

ISITAPE Installation

Installation instructions are provided with the product, and consists of:

- Copying a diskette to the P390 directory
- Adding CONFIG.SYS lines such as:

```
DEVICE=D:\P390\ISITAPE.SYS
```
- Adding a line to AWSCTYPE.MGR (add at the end of the file).

```
ISITAPE 3420 3480 3490
```
- Assigning appropriate devices to the ISITAPE device manager, using the P/390 configurator screen. The device manager will be given the next sequential identifier character, such as Q. The parameter field of each ISITAPE configurator line identifies the associated SCSI tape drive by using the sequential tape unit ID number listed in the ISITAPE.LOG file. When ISITAPE initializes during CONFIG.SYS processing, it generates a C:\ISITAPE.LOG file which contains a map of the available SCSI tape drives, and identifies each with a sequential number plus the name (vendor and model) of the tape drive.

4.5 Device Managers for Unit Record

Card readers, printers, and (sometimes) typewriter consoles are considered *unit record* devices. There are P/390 device managers to emulate these. "Real" unit record devices may be attached to a P/390 or R/390 system, using the channel adapter described in 4.7.1, "AWSC370 Device Manager" on page 106.

4.5.1 AWS2540

The simulated 2540 card reader can be used to submit jobs to OS/390 from the Server or, under certain conditions, from other workstations on the LAN. The submitted job (JCL and data) can be in ASCII or EBCDIC. With a little planning, this is very useful. For this description, we assume that you have defined a 2540R card reader in your OS/390 system, and added it to your JES2 definitions. (Using JES2 or JES3 is not required. You could allocate the card reader directly to a program, using the UNIT= parameter in JCL, but this would be unusual. We have never tried it.)

The AWS2540 device manager polls a specified directory frequently. When a file is found, it sends a device attention interrupt (for the emulated 2540 card reader) to OS/390. JES2 will then read the card input. (JES2 assumes it is a job, with proper JCL.) This is normal OS/390 JES operation.

When the device manager finds a file in the directory associated with the 2540, it checks the first two characters of the first record. If they are ASCII slashes (such as //ABC JOB ...), it assumes the file is in ASCII and automatically translates it to EBCDIC before passing it to OS/390. (If you have binary data in your job, such as an object deck, you would not use this method for sending it to OS/390.) Variable length records (ending with CR/LF (OS/2) or NL (AIX) characters, as is normal for Server text files) are padded to fixed-length 80-byte records before being passed to OS/390.

The Server directory you assign to the AWS2540 device manager could be on a LAN server, such as NetWare, IBM LAN Server, or NFS. Any workstation (DOS, Windows, OS/2, Unix) connected to the LAN (and authorized to write in the directory) could copy a file into the directory and thus submit a job to OS/390.

Only one 2540 is permitted in DEVMAP. Use the Configurator Update System Devices menu and add a 2540 device at an appropriate address in your I/O configuration and specify the MGR code for AWS2540. In the FN/P field, put a fully qualified filename. There must be an asterisk (“*”) within the name, and it must specify a subdirectory (that is, you need at least two backslash characters in the name). The directory can be located on a local drive, or on a file server. Here is an example:

Addr	Device	Mgr	FN/P
>00C	>2540	>7	>G:\RDR*.JCL

Make sure that you select an address where OS/390 expects to see a card READER (2540R, 2501, 3505). Do not confuse this with a card PUNCH, which also has the device type 2540 (2540P, 3525).

There is an optional parameter, “/NT.” Put this parameter after the file name in the FN/P field to disable all time-outs. If you do not specify /NT, then AWS2540 will remove a card deck if the S/390 doesn’t read at least once per minute (after having read the first card). See “Saving BAD card decks,” below.⁵⁸

AWS2540 tries to read the oldest decks in the directory first. There are several time and date stamps associated with Server files: created, last written, last accessed, and so forth. AWS2540 uses the “last written” date. The OS/2 file system only keeps timestamps to a 2 second resolution,⁵⁹ so files written to the directory within a 2 second interval may be read out of order.

Note that the OS/2 copy command does not update the “last written” date when it copies a file unless specifically told to do so. Use the “+ ,,” option to copy a file to a directory with a new date. For example:

```
CD G:\RDR
COPY H:\JOB LIB\REPORT4.JCL + ,,
```

Will copy the REPORT4.JCL job into G:\RDR with the current date and time.

Saving OLD Card Decks

The 2540 manager watches the directory specified in DEVMAP for new files and reads them one at a time. Once a file is read it is normally erased. If you want the 2540 to save the old files it has read, you should create a subdirectory. Call the new subdirector OLD. In the example above, you would use the OS/2 MKDIR command to create a subdirectory G:\RDR\OLD. Each time a file is completely read, it is moved to the OLD subdirectory if it exists. If the OLD subdirectory doesn’t exist, the file is erased. If a file with the same name as one just read already exists in the OLD subdirectory, it is replaced. We recommend the use of an OLD directory; it makes resubmitting a job very easy.

⁵⁸ We did not try this parameter, although we expect that it would not be needed since JES2 usually reads cards quickly. This parameter may be relevant if your OS/390 application reads cards directly, but this is very unusual in OS/390 today.

⁵⁹ This statement is for OS2 2.1. We did not verify it for Warp.

AWS2540 automatically removes a card deck if the S/390 doesn't read a card at least once per minute (after having read the first card). These card decks are either erased, or moved to a subdirectory named BAD. Using the example above, AWS2540 would try to move the file to G:\RDR\BAD. If the BAD subdirectory does not exist, the file is erased. This logic will normally not apply to OS/390, since JES2 will read all cards and depend on later processing to reject invalid data or jobs. JES2 will normally read all card input quickly, and should never encounter this one-minute timeout. In the unlikely event that you have OS/390 jobs that allocate and read the card reader directly, you should be aware of this timeout function.

ASCII/EBCDIC Translation and BINARY Card Decks

When AWS2540 reads the first card of a card deck, it checks the first few characters to see if they are "ID," "//," or "USERID." It makes this check in ASCII. If any of these match, ASCII/EBCDIC translation is assumed to be required for the whole file. Each ASCII line is delimited by an "end of line" (CR-LF for P/390, NL for R/390). Short lines are padded to 80 bytes with blanks. Long lines are split into multiple "cards."

If the first characters are not "ID," "//," or "USERID," then it is assumed that the file is binary and doesn't need translation. The file is read 80 bytes at a time until exhausted.

AWS2540 R/390 Differences

Setup and operation for the R/390 is very similar to the P/390, and the configurator line would like something such as this:

```

Addr  Device                               Mgr  FN/P
-----
>00C  >2540                               >6  >/tmp/RDR*

```

In this example, the AWS2540 device manager will monitor /tmp for any file name beginning with the letters RDR; for example, if you copy a file into /tmp/RDR17 or /tmp/RDRDATA, it will be read (assuming a JES2 reader is started, or some other application or subsystem is trying to read cards).

Warning: We created jobs on PC diskettes, and used the AIX command dosread to read the diskettes. We found that we must not read the diskette files directly to an AWS2540 input file; if this is done, random trash sometimes appears in the AWS2540 input file. We assume this is because AWS2540 starts reading the file before dosread finishes writing it. We found a two-step process worked better:

```

# dosread -a UPLOAD.JOB /tmp/holdit      read DOS diskette
# cp /tmp/holdit /tmp/RDR1              give it to AWS2540

```

An additional note: AWS2540 does not generate R/390 trace entries. Turning on device trace (from the Manual Operations panel) has no effect.

4.5.2 AWS2821 Device Manager

"Printed" output, typically from JES2, can be handled by the AWS2821⁶⁰ device manager. Output is sent to a printer (LPT1 on the OS/2 system, by default) or to a file. Output handled by this device manager is automatically translated to

⁶⁰ This device manager is named for the IBM 2821 control unit, which was part of the original S/360 series. The 2821 controlled card readers, punches, and printers.

ASCII. Other than translation to ASCII, it is not altered; it will contain JES2 separation pages and is probably oriented to 132 column printing.

This discription is primarily for the P/390. Differences for the R/390 are noted at the end.

A Server file may be specified instead of a printer. Printed output is appended to this file. After writing to the file, if no device manager control parameters are specified, AWS2821 closes the file if no new print lines arrive within 10 seconds. When more print lines arrive, the file is reopened (in append mode) and the new output added. By default, AWS2821 is not aware of different OS/390 jobs. The output may contain separator pages from JES2, but these are merely print records that are placed in the output file. You (on the Server side) must separate the output for different jobs.

The output file might reside on a LAN or NFS server and be available to multiple users. Be careful using these files, because file servers usually do not offer much protection against two programs opening the same file for output. AWS2821 always writes to the file (in append mode) and if your Server session only reads the file, there is no conflict. However the file will grow forever, and will consist of the output from many jobs (if it is JES2 writing the file, as a normal output printer device). To trim the file, or divide it into job-related files (easy with a REXX program), you need to be certain that AWS2821 (JES2) does not attempt to write to the file while you are manipulating it.

AWS2821 emulates a mainframe line printer on a PC-type printer. The emulated printer is an IBM 1403. At the time this was written, no other printer (such as a 3211 or 4248) was emulated. This means that FCB functions are not emulated. You should always generate a 1403 printer at the OS/390 address used with AWS2821. (The R/390 AWS2821 supports FCBs, as noted later.)

To configure a 2821, use the Configurator Update System Devices menu and enter the real address for the 1403 printer in the Addr field, the MGR code for AWS2821 in the Mgr field, and any optional parameters in the FN/P field:

```
Addr Device                Mgr  FN/P
-----
> 00E > 1403                > 6  > [parameters]
```

Optional Parameters:

- filename = Fully-qualified file name for printing to a file.
Default is to print to the PC printer LPT1.
LPTn (no colon) is also allowed.
The file is opened in append mode.
- /80 = Print in 80 column mode. Default is 132 columns.
Records are truncated to 80 columns. 80 column mode also sets a larger font, 132 column mode sets a smaller one.
- /I = Send a printer initialization string to an IBM 40x9 Laser Printer to switch it to PPDS mode. Default is not to send any initialization string.
- /C=filename specifies a fully-qualified file name, and this file contains advanced control parameters for AWS2821

Examples of FN/P field values:

```
LPT2
LPT1 /80
/C=D:\P390\PRTCTL.OOE
G:\printer\output.OOE
D:\mydata\script.prt /80
LPT2 /I
```

The DMSTART command (in IPL.CMD) that starts this device manager may have one parameter. It is a time (in seconds) that applies when a printer output file is open. If no printing occurs during the specified number of seconds, the output file is closed. For example:

```
'AWSSTART AWS2821.EXE n 60' /* wait 1 minute before closing files */
```

AWS2821 can accept a control file. You must create this file yourself; the P390 configuration program cannot do it automatically. To use it, include the name of the control file in the configurator line defining the emulated printer. For example,

Addr	Device	Mgr	FN/P
> 00E	> 1403	> 6	> /C=D:\P390\PRT.OOE

In our examples, we assume that Manager 6 is AWS2821; you should use the number that corresponds to AWS2821 on your system. The "/C=" characters are required. The remainder must be a fully qualified file name; we suggest using a meaningful name, such as shown in the example.

The control file is a simple ASCII file, created with any of the basic OS/2 editors. The control file will contain several lines in any order; the basic control lines are as follows:

```
// Comment lines begin with two slashes
PC= xx xx xx xx xx xx xx xx xx
EOJ=nn,oo,'ssssssssssssss'
LL=132 (or some other length)
FILE=C:\OUTLST.TXT
TR=xx xx xx, yy yy yy
LINE NUMBER
```

You are unlikely to have all of these control lines in any given instance of a printer control file. Only one instance of each statement can appear in the control file. The meanings of the parameters are:

- **PC** provides a string of printer control characters (entered in hex) that are sent to the printer every time a printer output file is opened. You must provide this string; the Cookbook contains examples for common Lexmark and HP printers. AWS2821 does not interpret or verify these codes. The maximum length of the line is 255 characters. All control files will normally have this parameter line.
- **EOJ**, if present, checks line nn (after an emulated skip to channel 1), at offset oo in the line, and attempts to match string ssss. If a match is found, then this page is the ending page of a job. After the next skip to channel 1, AWS2821 will close the printer file. (If more printed output arrives from the P/390, a new printer file is opened.) The nn and oo parameters start with 1; a page begins with line 1 and column 1. If the nn or oo parameters are zero, AWS2821 will scan the whole page or line, as appropriate.
- **LL**, which defaults to 132, sets a truncation length for print lines.

- **FILE** provides the name of a Server file for output, instead of sending output to the printer. You can specify an output file name in the configurator line instead. You can also specify a printer name, such as LPT1 or LPT2, as the file name; this would be useful if you have more than one printer.
- **TR** permits individual character translation, and can serve some of the more basic national language requirements. The “xx” operands are hexadecimal values representing S/390 bytes. The “yy” operands are hexadecimal values sent to the PC printer or file. For example, TR=C1 C2 C3, 61 62 63 would result in S/390 bytes for A, B, and C being printed as lower case a, b, and c on the PC. All x’C1’ bytes in incoming S/390 data would be translated to x’61’ before being sent to the PC printer or file, and so forth. If no TR statement is present, a generic EBCDIC to ASCII translation is used. The TR statement is limited to about 254 characters in length, permitting up to about 31 character translations. There must be an equal number of xx and yy operands, of course. The comma signals the end of the xx operands. The standard translation table (before applying any TR= functions) is shown in Appendix A, “AWS2821 Translation Table (P/390)” on page 167.
- **LINE NUMBER** causes AWS2821 to overlay the first 20 characters of every line with a field showing the line number (on the logical page, as seen by AWS2821). This is useful for determining the correct parameters for the EOJ operand.

With proper setup, AWS2821 plus a control file is a powerful combination. The ability to send control characters to the printer, before printing any output, is the key to the advanced functions. If your printer supports the necessary functions, you can send controls to rotate the output (to landscape mode), select a smaller character set (to hold more lines and characters, for example), and print on both sides of the paper (duplex).

We found that OS/2 reinitializes the printer between jobs (as OS/2 regards jobs). The initialization strings sent to the printer via AWS2821 did not affect the proper printing of output from other OS/2 applications.

Examples of control strings for Lexmark Optra printers and IBM 4019 printers are given in the Cookbook.

AWS2821 R/390 Differences

There are two key differences for the R/390 version of AWS2821:

- The control file functions are not implemented at this time. That is, you cannot specify /C=filename in the configurator parameter field. There is no way to send customized printer control strings to AIX printers.
- FCB functions are supported, so you can emulate 3211 printers in terms of FCB usage. UCS is not supported.

No AIX spooling functions or queues can be specified for the printer. That is, it must have minimal AIX definitions.

The parameter field of the configurator line should contain the device name of a printer or the full path name of a file. For example:

Addr	Device	Mgr	FN/P
>	>	>	>
00E	1403	6	/dev/lp0
00F	3211	6	/tmp/output.00f

If output is sent directly to a printer (such as /dev/lp0 in this example), it may contain typical PC printer controls, including:

```
hex 0A for line feed
hex 0C for form feed (new page)
hex 0D for carriage return
ESC 3 n (where n = lines per inch)
ESC C n (where n = lines per page)
```

4.5.3 AWS3215 Device Manager

The 3215 sessions are typewriter/keyboard sessions. These sessions are useful for stand-alone utilities and for VM virtual machines that need a console to monitor them, but do not require much input. OS/390 does not support 3215 consoles. We recommend using a 3215 console for stand-alone ICKDSF and DFSS; the emulated 3215 is easier to use than an emulated 3270. When you are finished with the stand-alone utilities, you can simply undefine the 3215 (or use the configurator option to “turn it off”).

To configure a 3215 session, use the Configurator Update System Devices menu and enter the real address for a 3215 device in the Addr field, the MGR code for AWS3215 in the Mgr field, and the title you want to display for the session in the FN/P field. Here is an example entry:

Addr	Device	Mgr	FN/P
>009	>3215	>5	>Console

There are a number of features that are quite useful for VM systems and these are explained in the DOC file on the P/390 diskettes. The same considerations do not apply to OS/390, and we will not discuss them here.

4.6 Device Managers for Communications

A number of device managers are provided for communications, but not all these are supported by OS/390. The communications device managers include:

- LAN3172, for SNA connections on a LAN
- WAN3172, for SNA connections on SDLC lines
- MGR3172, for NetView connection to an emulated 3172
- LCS3172, for TCP/IP connections on a LAN
- AWSICA, for emulating an integrated communications adapter
- AWSPBS, for emulating an integrated communications adapter (with more capacity and options)

The first three listed here are closely related, and share a number of OS/2 modules generally known as SNACOM. The same modules, with slightly different interfaces, are used in the IBM 3172 products. This helps ensure that the emulated device operation (as used by the P/390) is very similar to the real device operation, as used by a mainframe.

The implementation of communications device managers is generally quite different between P/390 and R/390 systems. The reason is that the device managers use the underlying Server functions to drive the communications links, and the Server facilities are quite different between OS/2 and AIX. For this reason, each of these device managers is described separately for P/390 and R/390 systems.

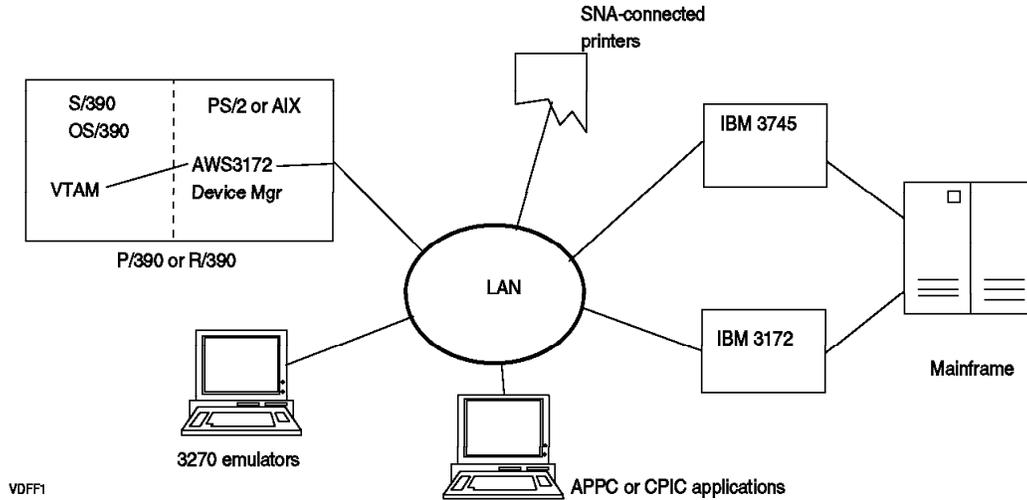


Figure 21. LAN SNA Connections. A variety of SNA connections are possible over a LAN, including full SNA interoperability with a mainframe.

4.6.1 LAN3172 Device Manager (P/390)

The LAN3172 device manager is used to provide a full SNA connection to a LAN. (Our descriptions are in terms of a token ring, but the same process should work for Ethernet.) The LAN3172 device manager emulates an IBM 3172 (Model 1, and some Model 3 functions) using a token-ring (or other) adapter in your system. Its use for workstations using SNA 3270 emulation has already been mentioned, but this is only one of its uses. For example, the configuration in Figure 21 could be used for NJE and CICS ISC links from a P/390 OS/390 system to a mainframe OS/390 system. The OS/390 and VTAM parameters are exactly what they would be for a “real” 3172 with SNA.

LAN3172 replaces AWS3172, which was distributed with earlier versions of the P/390 support programs. Externally, it does the same functions as AWS3172, but the internal code is new.

The LAN3172 emulator lets VTAM running on the P/390 think it is talking to a real channel-attached 3172. Keep in mind that OS/390 regards a 3172 as a channel-to-channel adapter (3088 or CTC).

To configure LAN3172, use the P/390 Configuration Update System Devices panel and configure a single 3088 device address. Use the manager code for AWS3172 in the Mgr field. Do not put anything in the FN/P field. For example:

Addr	Device	Mgr	FN/P
> E40	> 3088	> 9	>

More than one LAN3172 can appear in the configurator, although there should seldom be a need for this.⁶¹ One LAN3172 emulated device can use multiple LAN adapters. Only VTAM/SNA (LSA or Link Services Architecture) connections are

⁶¹ The only reason we can see for running multiple LAN3172s is if there are multiple VTAMs active. This could occur with VM running second level OS/390 or other VMs.

supported. For TCP/IP use the LCS3172 device manager; LAN3172 can coexist with LCS3172 using the same LAN adapter.

If more than one LAN3172 device is defined, they are all handled by one copy of the LAN3172 device manager. The different devices (at different OS/390 addresses) are termed *multiple subchannels* in the LAN3172.DOC file. If more than one LAN3172 device is defined, a given LAN adapter is dynamically assigned to a particular subchannel when VTAM activates a Major Node (XCA) that specifies that adapter (with the ADAPNO= parameter).

Installation Notes

Using LAN3172 means that you must have the correct VTAM parameters in OS/390, and the correct LAPS parameters in OS/2. Also, at the remote end of a connection, you must have whatever parameters are needed to match OS/390's VTAM parameters. These interactions are complex. There is a large LAN3172.DOC file on the P/390 support diskettes that covers some of these topics. There is also a redbook, *Connectivity of a PC Server System/390*, IBM order number SG24-4624, that describes several typical setups in detail.

We will not attempt to describe detailed installation here, but we will mention several of the key topics involved:

- OS/2 LAPS must be configured
- SAP addresses must be resolved. Both VTAM and CM/2 (if used for SNA connection) default to SAP address 04. You must change one of them.
- VTAM Major definitions are required
- Ethernet (if used) MAC types must be defined

LAN3172 automatically generates two files (in the P390, SNACOM, or IBMCOM directory) that are helpful for debugging:

AWS3172.LOG - contains Informational Messages

AWS3172.ERR - contains Error Message

Also, to force LAN3172 to generate debug and trace entries, add a debug parameter in IPL.CMD:

```
'AWSSTART LAN3172.EXE N DEB(#)'
```

where # = 0 = No tracing (default)

= 1 = Base tracing

= 2 = Detailed tracing

(The "N" means that an OS/2 window need not be opened while processing the command.)

If you are using OS/2 SNA applications (such as CM/2 SNA functions), you may need to increase LAPS parameters such as number of link stations and number of SAPs.

If you use SNA-connected 3270 emulators, you may need to create an SNA USSTAB. The CD-ROMs provide only a non-SNA USSTAB.

Some releases of VTAM are sensitive to buffer sizes, and 508 is a good size to use. For example,

```
IOBUF=(200,508,19,,1,20) (in SYS1.VTAMLST parameters)
```

4.6.2 LAN3172 Device Manager (R/390)

The R/390 version of LAN3172 provides the same functions as the P/390 version. The setup functions are slightly different. As with the P/390 version, LAN3172 can share a LAN adapter with LCS3172 or with AIX. However, if SNA communications are used under AIX, with the same LAN adapter, you will need to manage SAP addresses.

The DEVMAP entry has no parameter.

```
      Addr  Device                               Mgr  FN/P
-----
>      >
      E40   3088                               9
```

You must edit the `ipl390` script and uncomment the line:

```
$DMDIR/dm3172 tok0 &
```

Edit this line to refer to the correct LAN adapter. The possible names are `tok0`, `tok1`, `tok2`, `ent0`, `ent1`, and so forth.

If your R/390 release is before 4.0.15.0, you must also ensure that the `bos.dlc` files are included in your AIX system. Steps for doing this were described in 3.3.2, "AIX Installation" on page 50. If it is a later release, you must ensure that one of the following is configured:

```
dlc token          (for yoken ring)
dlc ethernet       (for DIX Ethernet)
dlc IEE_ethernet   (for 802.3 Ethernet)
```

You can determine if these are installed by using one (or more) of the following commands:

```
ls /dev/dlctoken
ls /dev/dlccether
ls /dev/dlc8023
```

If the appropriate device file does not exist, you must configure it using SMIT. To do this for a token ring:

```
smit
  Devices
    Communications
      Token ring (or Ethernet ...)
        Adapters
          Change / Show Characteristics ...
          select ...dlc Token-Ring ....
```

After configuring the dlc device manager, you may need to configure the network adapter for a token ring adapter. *You must enter alternate addresses for both ethernet and token ring adapters.*⁶² To change a token ring adapter to use a locally administered (alternate) address:

```
smit
  Devices
    Communications
      Token ring
        Adapters
          Change / Show Characteristics ...
```

⁶² This forces AIX to place the address in ODM, where the R/390 modules can find it.

```

select tok0 (or whatever adapter adapter you use)
...
Enable ALTERNATE TOKEN RING address    yes
ALTERNATE TOKEN RING address          0x400074700001
Apply change to DATABASE only         yes
...

```

A locally administered address must be twelve hex digits, beginning with 4000. The address in this example matches that in VM VTAM in one of the CD-ROM systems. OS/390 has no default addresses, so you must your own. A power-off cycle is required to established the changed address; simple rebooting may not do it. An alternate Ethernet address is also twelve hex digits.

4.6.3 WAN3172 Device Manager (P/390)

WAN3172 provides direct SDLC connections, without the need for an IBM 37x5 (or equivalent). It uses PS/2 PortMaster adapters or PS/2 Wide Area Connector (WAC) adapters for line connections. Remember that SDLC connections are synchronous connections, and normally use different modems than typical PC asynchronous connections.

A WAC adapter can have up to two SDLC lines. The number of WAC adapters in a system is limited by the number of free adapter slots. A PortMaster adapter can have up to eight SDLC lines, and there is a maximum of two PortMaster adapters in a system. In practice, a P/390 will have one of these configurations:

- No SDLC connections
- One WAC adapter for one or two SDLC lines
- One PortMaster for eight SDLC lines
- Two PortMasters for sixteen SDLC lines
- A connection to a 3745 for more SDLC lines

To configure WAN3172, use the P/390 Configuration Update System Devices panel and configure a single 3088 device address. Use the manager code for WAN3172 in the Mgr field. Do not put anything in the FN/P field. For example:

```

Addr  Device                      Mgr  FN/P
-----
> E40 > 3088                      > C  >

```

A single address is used, unless you have two PortMaster adapters (for more than eight SDLC lines). In this case, two addresses (an even/odd pair) are used; you must specify two 3088 addresses in the P/390 configurator, with adjacent addresses.

Installation

This device manager is the only P/390 device manager that is not installed by the basic P/390 support programs installation process. You must take several extra steps to complete the installation, as indicated by the DOC file.

After completing the device manager installation, you must configure the SDLC lines. This is started by using F11 in the Main Menu of the P/390 configurator. A complete example for configuring one simple SDLC line connected to an IBM 3174 (remote) control unit is given in the Cookbook.

Setting up WAN3172 for use is somewhat complex. When you complete the device manager installation, a large WAN3172.DOC file exists. You should reference this for details.

If you have a WAC adapter installed, you can use the command

```
AWSICA /q=WACDD$
```

to verify that it is installed by displaying the features that are detected for the adapter. (Go to the P390 directory in order to execute this command.)

4.6.4 WAN3172 Device Manager (R/390)

We did not have time to work with WAN3172 for the R/390 when we prepared this Redbook. The R/390 DOC files contain fairly detailed information for setup and usage.

4.6.5 LCS3172 Device Manager (P/390)

LCS3172 emulates a 3172 device that provides LAN TCP/IP connectivity for OS/390. It works with OS/390 (TCP/IP) and not OS/2 or AIX TCP/IP. The LCS3172 device manager can use the same token ring (at the same time) as the LAN3172 device manager.⁶³ Both are generated as 3088 channel-to-channel devices in OS/390, and you would need separate addresses for LCS3172 and LAN3172. LCS3172 uses an even-odd pair of addresses channel addresses, whereas LAN3172 uses only a single channel address.

The current LCS3172 device manager does not, at this time, provide the TCP/IP offload function of a 3172-3 unit.

Note. A single LAN adapter (token ring or Ethernet) can be used by only one TCP/IP driver at any one time. For example, if OS/390 is using TCP/IP (through the LCS3172 device manager) then an OS/2 TCP/IP product cannot use that same adapter. Likewise, if an OS/2 TCP/IP product is active, then OS/390 TCP/IP cannot use that adapter. Separate adapters can be used simultaneously. "Using" an adapter means that the TCP/IP communications subsystem has been started. It does not matter whether actual TCP/IP data transfer is taking place.

To configure LCS3172, use the P/390 Configuration Update System Devices panel. Configure an even/odd pair of consecutive addresses as device type 3088 for each logical 3172. Use the manager code for LCS3172 in the Mgr field. Do not enter anything in the FN/P field. For example:

Addr	Device	Mgr	FN/P
>	>	>	>
E20	3088	A	
E21	3088	A	

Up to four logical 3172 attachments (four consecutive even/odd address pairs, eight channel addresses total) may be defined. LAPS supports a maximum of four LAN adapters; if all four are devoted to LCS3172, then the eight channel addresses would be used. In general, you should only configure a single LCS3172; it can be used with multiple LAN adapters.

⁶³ Similar sharing may not work with Ethernet. Ethernet has two slightly different protocols, DIX and IEEE 802.3, and DIX is the most commonly used. SNA normally requires 802.3 protocols. A given Ethernet adapter is set (in LAPS, or in the PROTOCOL.INI file) to one protocol, either DIX or 802.3.

Installation Details

We will not attempt to describe detailed installation here; the DOC file contains detailed installation information. Key topics are:

- OS/2 TCP/IP (if used) must have a different LAN adapter than the one used by LCS3172
- OS/2 LAPS must be configured; 802.2 is required, and the DIX interface (for Ethernet) is usually required
- OS/390 TCP/IP must be configured. The user must coordinate the TCP/IP definitions with the LAPS definitions and the hardware LAN adapters.
- For the LCS3172 version released in a fixpack in late August 1996, you must add the following two lines to CONFIG.SYS:

```
DEVICE=C:\IBMCOM\PROTMAN.OS2 /I:C:\IBMCOM
DEVICE=D:\P390\AWSLCSDD.SYS
```

The PROTMAN statement must be first, and may already exist in your CONFIG.SYS file. The AWSLCSDD statement should be added anywhere in the P/390 section of CONFIG.SYS. If your LCS3172.EXE file is dated 5-13-96 (or later) you will need to manually add these statements to CONFIG.SYS. Be certain to check whether the PROTMAN statement is already there; do not enter a second copy of the statement. The LCS3172.DOC file explains the change to CONFIG.SYS in more detail.

Examples of TCP/IP profile definitions are:

```
; Hardware Definitions
; 3172 @ E20/E21 TokenRing

DEVICE UNITY LCS E20 |      (Use the Even (READ) Channel Address)
LINK TRE20 IBMTR 0 UNITY   (0 is the LAPS number)

; Hardware Definitions
; 3172 @ E22/E23 Ethernet

DEVICE UNITY LCS E22
LINK ETHER600 ETHERNET 1 UNITY
```

For debug and trace entries, you can add a debug parameter in IPL.CMD:

```
AWSSTART LCS3172.EXE w DEBUG(#)
```

where w is N (no monitoring window) or T (for a text window)

```
DEBUG(0) - NO TRACING
DEBUG(1) - CHANNEL PARTIAL / LANA PARTIAL (DEFAULT)
DEBUG(2) - CHANNEL FULL / LANA PARTIAL
DEBUG(3) - CHANNEL PARTIAL / LANA FULL
DEBUG(4) - CHANNEL FULL / LANA FULL
```

```
PARTIAL TRACE captures all MAJOR Events
FULL|PARTIAL governs MINOR Event detail
```

A WATCH WINDOW is displayed when LCS3172 is started with the Text window option. The diskette DOC file contains a detailed description of the window contents. In general, the trace information shows specific I/O events, and status

for both the LAN interface and the emulated channel interface. Performance data is shown and updated every ten seconds.

In a rough sense, the trace information is similar to that provided by other TCP/IP trace facilities. It may be very helpful when first establishing a TCP/IP environment.

A number of reported difficulties with MVS TCP/IP seem to be resolved by using a packet size of 1492 bytes (specified in the MVS TCP/IP parameters).

4.6.6 LCS3172 (R/390)

The R/390 version of LCS3172 provides the same functions as the P/390 version, but the setup details are a little different. As with the P/390 version, LCS3172 cannot share a LAN adapter with the Server; AIX TCP/IP and OS/390 TCP/IP (via LCS3172) cannot share a LAN adapter. You need to take specific steps to ensure this does not happen.

The DEVMAP entries must specify the LAN adapter to be used. If it is Ethernet, it must also specify the protocol. Your entries (always as a pair of addresses) would look like one of the following sets, assuming LCS3172 is device manager A:

Addr	Device	Mgr	FN/P
>	>	>	>
E20	3088	A	/dev/tok0
E21	3088	A	/dev/tok0
E20	3088	A	/dev/ent0,standard
E21	3088	A	/dev/ent0,standard
E20	3088	A	/dev/ent0,802.3
E21	3088	A	/dev/ent0,802.3

If you use a second token ring adapter, you would specify /dev/tok1 (or /dev/ent1, for Ethernet), and so forth. You can use various smit commands to inspect your device configuration, and determine the name of the LAN adapter you want to use for LCS3172. (The device manager identifier, shown here as "A," may be different in your system.)

You must edit the ipl390 script and uncomment these lines:

```

ifconfig tr0 inet up
sleep 1
ifconfig tr0 inet detach
sleep 1
$DMDIR/lcs3172tx &
sleep 3
$DMDIR/lcs3172rx &
sleep 10          (see note below)

```

Edit the two ifconfig statements to refer to the correct LAN adapter: tr0, tr1, en0, en1, or something similar. Note that the adapter names here are spelled slightly differently than in the DEVMAP field. If the two ifconfig statements are not in your ipl390 script, we suggest you add them. Their purpose is to ensure that the

LAN adapter is detached from AIX TCP/IP before LCS3172 attempts to acquire it. You could do this in other ways, but the method shown here seems to work.⁶⁴

There is a *sleep 10* statement in the *ipl390* script. We found this was necessary in order for the LCS3172 processes to start and acquire the LAN adapter before OS/390 sensed the 3088 device to determine if it existed.⁶⁵ At the time this was written, the developers were looking at ways to remove the need for this sleep statement, and it may not be needed with your system. (The 10 seconds specified may be excessive, but we did not experiment to determine the minimum workable time.)

The OS/390 TCP/IP definitions are the same as for a P/390 system. If your R/390 support level is 4.1.0.1 or higher, you must enable the DLPI interface in AIX. The LCS3172.DOC file contains specific instructions for this. In brief, this consists of:

1. Commenting the appropriate adapter statements in */etc/dlpi.conf*.
2. Uncommenting the corresponding statements in */etc/pse.conf*.
3. Forcing streams reconfiguration:

```
strload -u -f /etc/dlpi.conf
strload -f /etc/pse.conf
```

4. Rebooting AIX

4.6.7 MGR3172 (P/390 OS/390 Only)

MGR3172 provides a NetView interface to an emulated 3172 device. The emulated 3172 is provided by the LAN3172 or WAN3172 device managers. Only one logical 3172 is created if you use both of these device managers. The NetView function does not extend to other emulated devices, such as the emulated 3172 provided by the LCS3172 device manager (LCS3172 provides LAN connections for OS/390 TCP/IP), or lines provided by AWSICA or AWSPBS.

MGR3172 is used only with P/390 systems, and only with OS/390 (or MVS). It is specified by a DEVMAP entry such as:

Addr	Device	Mgr	FN/P
>	>	>	>
> E40	> 3088	> B	>

Where B corresponds to the MGR3172 device manager. Only one such entry may be specified in DEVMAP.

An XCA major node and a switched major node must be specified in VTAM, corresponding to the address specified in the DEVMAP entry. In addition, an IDBLK and IDNUM value must be passed to MGR3172 and the same values used in the switched major node entry. The values are passed to MGR3172 in the start command in IPL.CMD. This command can appear as:

```
call dmstart "LSApBMGR.EXE N DeBug(0) /id=07400000"
```

⁶⁴ We found that AIX TCP/IP tended to re-acquire the LAN adapter given the slightest chance. Placing the detach statement immediately before the *lcs3172* start statements partly solves this problem. We found it was necessary to "ifconfig up" first, to make the "ifconfig detach" more reliable.

⁶⁵ As part of OS/390 startup, it senses all the its defined I/O devices to see which ones are present. This is done quite soon after IPL starts. If it cannot sense a device, it assumes it is not available. In the case of 3088 devices (which is how OS/390 sees LCS3172), the device cannot be varied online later. The device must appear ready during IPL. This restriction does not exist on "real" mainframes.

In this example, the IDBLK value is 074 and the IDNUM value is 00000. The Debug value is 0 (no tracing), 1 (summary tracing), or 2 (detailed tracing). The default is no tracing.

The MGR3172.DOC file contains more information, and examples of XCA and switched major node definitions.

4.6.8 AWSICA Device Manager (P/390 Only)

AWSICA emulates the integrated communication adapter (ICA) lines that were found on IBM 9370 and some 43xx series processors. This same emulation can also appear as IBM 2703 communications lines. Wide Area Connector (WAC) adapters are used. Emulation is for SDLC or BSC protocols.

A maximum of four WAC adapters (each providing up to two lines) can be used. We strongly recommend using WAC adapters instead of the older Multiprotocol adapters. The IBM SDLC Modem/A adapter is also considered to be a Multiprotocol adapter. Note that only one interface (line) is supplied with a basic WAC adapter. You must order a second connection (or "cable") to obtain the second line, which is known as an EIB (electrical interface board).

Some Multiprotocol adapters (P/N 92G7516 or P/N 6451114) support V.25 bis modems and autodial. WAC adapters support the same autodial. Multiprotocol adapters normally use RS232D interfaces. WAC adapters use RS232D, X.21, V.35, and RS422 interfaces.

OS/390 VTAM does not support ICA lines and cannot use this device manager. VM and VSE VTAM support ICA lines and can use this adapter.

OS/390 still supports some IBM 2703 lines, and this device manager is used to emulate these lines. In practice, this is limited to BSC support, and requires an application that uses BSC at the EXCP or BTAM level. JES2 NJE and RJE, for example, work at this level and can use emulated 2703 BSC lines provided by this device manager. While writing an earlier redbook, we connected two P/390 MVS JES2 NJE systems back-to-back, using a WAC adapter in each system, and quickly had BSC NJE processing between the two systems.

There is one DEVMAP entry for each communications line. Each line has a unique S/390 address, and requires its own OS/390 UCB. For example,

```

Addr Device                               Mgr  FN/P
-----
>      >                                >  >
> 028  >                                >  E  > S=3A BSCA
> 029  >                                >  E  > S=3B BSCA

```

This example represents a WAC adapter with two lines. The adapter is in slot 3, and the two lines are A and B. The OS/390 addresses have been generated as 2703 BSC lines.

Appropriate OS/2 device drivers must be included in CONFIG.SYS for Multiprotocol or WAC adapters; they are included with the P/390 support diskettes. These device drivers are unique to AWSICA and cannot be used with any other programs. In particular, LAPS (and the WAN3172 device manager) cannot access an adapter using one of these device drivers. The drivers are:

```

DEVICE=D:\P390\AWSMPADD.SYS      (for multiprotocol)
DEVICE=D:\P390\AWSWACDD.SYS     (for WAC)

```

If this driver is used for WAC adapters, the OS/2 WAC driver (IBMWAC.OS2) must not be specified in CONFIG.SYS.

There are many optional parameters for this device manager, and you should consult the AWSICA.DOC file and the AWSWACDD.DOC file for more details.

4.6.9 AWSPBS Device Manager (Normally VM and VSE)

AWSPBS emulates integrated communications adapter (ICA) lines, and is quite similar to AWSICA. Its difference is that it uses only Portmaster adapters; it does not use Multiprotocol adapters or Wide Area Connector adapters. A single Portmaster adapter provides up to eight SDLC or BSC lines, and up to two Portmaster adapters can be installed in a P/390 or R/390 system.

Note that different Portmaster adapters (different part numbers) are used for P/390 and R/390 systems.

OS/390 VTAM does not support ICA interfaces, and this device manager cannot be used with OS/390.⁶⁶

Typical DEVMAP entries might be:

Addr	Device	Mgr	FN/P
>	>	>	>
> 030	>	> F	> 00,SDLC,N,9600,LL,PRTS,NRZI
> 031	>	> F	> 01,SDLC,N,9600,LL,PRTS,NRZI
> 032	>	> F	> 02,SDLC,N,9600,LL,PRTS,NRZI
> 033	>	> F	> 03,SDLC,N,9600,LL,PRTS,NRZI

Where F represents the code assigned to the AWSPBS device manager. The various parameters are explained in the AWSPBS.DOC file.

A special OS/2 device driver is required for AWSPBS:

```
DEVICE=D:\P390\ICARICIO.SYS D:\P390\ICAPARM.1CD
```

The second file specified (ICAPARM.1CD) contains additional control parameters for the Portmaster adapter. Again, the AWSPBS.DOC file contains additional information.

4.7 Other Device Managers

This category of *other* device managers includes the rest of the P/390 and R/390 device managers.

4.7.1 AWSC370 Device Manager

The AWSC370 manager allows real S/370 I/O control units to be attached via the S/370 Channel Emulator/A card.⁶⁷ The current version of AWSC370 supports only the S/370 Channel Emulator/A adapter, which is used with both P/390 and R/390 systems. The S/370 Channel Emulator/A adapter provides a parallel channel ("bus and tag") connection between Servers and mainframe control units. Only a

⁶⁶ An exception may exist for emulating non-VTAM BSC lines, in a way similar to that described under the AWSICA device manager. This usage has not been verified.

⁶⁷ The term CHAN370 is also used when referencing this device manager, although CHAN370.SYS and CHAN370.EXE are used in CONFIG.SYS, and are not P/390 device manager modules.

limited range of control units have been used with this connection. These include:

- IBM 3480 and 3490 tape control units and drives
- IBM 3420 tape control unit and drives
- IBM 38xx (and other AFP printers) and IBM 4248 printers
- IBM 3174 (non-SNA) control unit (for up to 32 3270-family displays)
- IBM 37x5 (that is, 3705, 3725, 3745) communications adapter in NCP mode. (EP or PEP operation has not been verified.)

The use of hardware device address 00 is not allowed. Of this list, only AFP printers and 37x5 communications control units have been verified for R/390 use. Other devices and control units *may* operate correctly with the P/390 or R/390, except for DASD and DASD control units, which cannot be attached through this adapter.

Please understand these restrictions. The S/370 Channel Emulator/A adapter is not as *tolerant* as a mainframe channel. IBM does not market it as a general-purpose parallel channel device. This means that some devices which operate correctly on mainframe channels may not operate correctly with the channel emulator. There are several areas of potential problems:

- Control signals (on the tag cables) that are too fast, or not the right shape for the adapter
- Control units that depend on obscure features of a mainframe channel, such as automatically polling for a device end if it does not arrive with the channel end signal
- Data presentation (in the bus cables) whose shape is outside the acceptance range of the adapter
- Control units that depend on fast channel responses to tag signals (where the channel adapter may need Server processing in order to generate a response)
- And so forth.

The adapter has been tested with a few devices, listed above. IBM may not support (provide assistance, corrections, or new code) other devices. (You are welcome to try them, but you are on your own.) Devices known to have problems, for example, include IBM 3274 control units, a variety of OEM tape control units (3420-, 3480-, and 3490-types), OEM 3174-type control units, and some IBM 3174 channel-attached SNA control units.

The adapter has mixed performance characteristics. It is slow when starting an I/O operation, but acceptable once the I/O operation is running. This makes it a relatively poor performer for small, unblocked records. Of course, OS/390 itself is a poor performer for small, unblocked records, so the limitations of the adapter are not unique. We should note that 38xx and 39xx printer performance with this adapter is excellent when operating in page mode.

Configuration (P/390)

For P/390, you must add the following statements to CONFIG.SYS. If they are already present, but commented out, remove the REM from the front of each line you wish to make active.

```
DEVICE=D:\P390\chan370.sys tbuf=2040 pollcount=10000  
RUN=D:\P390\chan370.exe
```

The "pollcount" parameter is needed to help match the speed of the channel adapter to the speed of the processor used by OS/2. We were unable to observe any effects caused by changing this parameter, and suggest you do not change it unless you have time to measure and observe the effects.⁶⁸

The Channel Emulator can be started at three different speeds and in three modes. You must edit the DMSTART statement for AWSC370.EXE in IPL.CMD to make these changes. The default startup parameters in IPL.CMD produce Block 3MB/sec mode.

```
/[s]Dn /[s]S /[s]B /Wttttt /Z
```

where "s" is the PS/2 slot location of the card. The default is to apply the parameter to all slots.

Dn turns on high-speed data streaming. D3 specifies 3 MB/sec and D4 specifies 4.5MB/sec. No specification produces DCI mode (which you probably do not want.)

S specifies selector mode.
B specifies byte multiplexor mode.
The default is block multiplexor mode.

ttttt sets a timeout, in seconds. The default is 10 seconds. The value 65535 disables the timeout (and may be useful for debugging).

Z activates trace debugging, including SNAP trace support.

For example, to start the S/370 Emulator manager in 4.5 MB/sec Data Streaming mode:

```
call dmstart "AWSC370.EXE N /D4"
```

(The "N" in the command indicates that no OS/2 window need be opened while processing the command.) To start two S/370 Emulator cards in different modes:

```
call dmstart "AWSC370.EXE N /5D3 /5B /1D4"
```

This will run the card in Server Adapter slot 5 in Byte Multiplexor mode at 3 MB/sec. and the card in slot 1 in Block Multiplexor mode at 4.5MB/sec. (The "N" in the command means that a window need not be opened while processing the command.)

Configuration (R/390)

We did not work with an R/390 channel adapter while writing this Redbook, and had no hands-on experience with configuration. The R/390 DOC files contain basic information for setup and usage.

⁶⁸ This count controls a loop that is executed after a channel adapter operation is started. If the channel device returns status within the time used by the loop, the status is processed as part of the initial call to the device manager. This will eliminate a future interrupt and redispach of the device manager.

Configuring Devices

There are two situations where you need to add parameters to a CHAN370 device entry in DEVMAP using the P/390 Configurator UPDATE SYSTEM DEVICES menu:

1. If the hardware unit address does not match the device address used by your P/390 operating system. The use of hardware device address 00 is not allowed.
2. If there are multiple S/370 Emulator cards in the PS/2.

These parameters may optionally be specified in the FN/P field of any devices that are assigned to the CHAN370 MGR code:

- (blank) P/390 device address is the same as the hardware unit address. Assign to S/370 Emulator card in lowest numbered PS/2 slot.
- * Same as blank.
- *dd Remap P/390 device address to hardware unit address "dd". Assign to S/370 Emulator card in lowest numbered MC slot.
- c Don't remap device address. Assign to S/370 Emulator card in MC slot "c".
- cdd Remap P/390 device address to hardware unit address "dd". Assign to S/370 Emulator card in MC slot "c".

If "c" is specified, it is checked. If there is no S/370 Emulator card in slot "c" then AWSC370.EXE terminates with an error message.

This DEVMAP entry remaps P/390 address 580 to hardware unit address 60:

```
Addr Device                      Mgr  FN/P
-----
> 580 > 3420                      > L > *60
```

Assign device 580 to S/370 Emulator card in slot 5 and 590 to card in slot 1:

```
Addr Device                      Mgr  FN/P
-----
>      >
> 580  3420                      > L > 5
> 590  3480                      L   1
```

Assign device 580 to S/370 Emulator card in slot 5 and remap to hardware unit addr 60:

```
Addr Device                      Mgr  FN/P
-----
> 580 > 3420                      > L > 560
```

Use with 3174 Control Units

"Real" IBM 3174 control units are attached to the P/390 through a S/370 Channel Emulator/A adapter. There are large numbers of 3174 units in existing OS/390 installations, and (if you are working in one of these installations) you are quite likely to find an unused unit that you can use with your P/390. The primary advantage of using a 3174 is that your existing 3270 terminals (and all their coax cables that are embedded in your walls, floors, and ceilings) can be used. (You may need to do some creative patch panel rearrangements to route the coax connections to your P/390 location.)

Remember that the S/370 Channel Adapter/A cannot use an address ending in X'00'. If your OS/390 system expects to find local 3270 terminals at addresses 800-81F (for example), you will need to generate new addresses, such as 820-83F.

We do not recommend the use of local SNA 3174 control units. We have seen a number of problems between them and the channel adapter. Also, if you have the option of ordering new 3174 controllers, we recommend using LAN-attached units such as the 3174-63R. These tend to be less expensive (LAN connections are less expensive than parallel channel electronics). Informal tests appear to indicate that token-ring connected 3174s are slightly faster than channel-attached units when the S/370 Channel Emulator/A provides the channel attachment.

We were able to use a surprising definition with a non-SNA 3174. We had a non-SNA 3174 with addresses 0C0-0DF. We defined the DEVMAP as follows:

Addr	Device	Label	Atype	Size	Mgr	FN/P
>	>	>	>	>	>	>
0C0	3278				3	LOCAL 3270
0C1	3278				3	LOCAL 3270
0C2	3278				3	LOCAL 3270
0C3	3278				4	LAN 3270
0C4	3278				4	LAN 3270
0C5	3278				N	CHANNEL 3174
0C6	3278				N	CHANNEL 3174
0C7	3278				N	CHANNEL 3174

As you can see, one OS/390 string of 3270s (defined to OS/390 as local, coax-attached non-SNA units) was defined to the P/390 configurator as a mixture of local (AWS3274) LAN (LAN3274), and channel-attached devices. The 0C5-0C7 devices corresponded to the appropriate coax connectors on the 3174. This arrangement worked correctly.

4.7.2 LAN3088 Device Manager (P/390 only)

LAN3088 emulates a channel-to-channel connection, over a LAN, with another P/390 system. It is used *only* between two P/390 systems. The R/390 is not supported, and connections to mainframes are not supported. This device manager is used (by VTAM) between two OS/390 systems. At the time this was written, had little experience with LAN3088 and will not describe it here. A full description is found in the LAN3088.DOC file on the P/390 support diskettes.

4.7.3 AWS2703 Device Manager (P/390 only)

AWS2703 allows devices to connect to the P/390 via the asynchronous port; that is, via an OS/2 COM port. It works with OS/390 devices generated as 2703s or as 2540 card punches.⁶⁹ When a modem is attached to the asynchronous port, the AWS2703 device manager acts as a 2703 Control Unit. This allows remote users to connect to the P/390 via an ASCII terminal. It is up to the S/390 code to support any terminal attached this way. OS/390 no longer supports 2703s, so you would need a rather old OS/390 (or SVS or MVT or ...) to use this interface.

When used with this device manager, an emulated 2540 card punch does not write normal OS/2 files (as you might expect, after reading about the AWS2540

⁶⁹ We realize this is a rather odd combination of devices for one device manager, but (other than the 2540 punch being output-only) they are handled in much the same way.

card reader device manager). When an OS/390 program writes to a 2540 punch associated with the AWS2703 device manager, the OS/2 system will send the "punched" data out on the indicated COM port. This does not correspond to any normal mainframe OS/390 function, but might be useful for controlling unusual devices (or even ASCII output displays) connected to your system.

For each 2703 or 2540 device, you need to specify the parameters governing its communications. To do this, use the Configurator UPDATE SYSTEM DEVICES screen. Enter the address and the manager number. In the FN/P field, enter the following parameters:

```
devname:baud,parity,data,stop,flow,dcd
```

Default Values:

```
COM1:1200,N,7,2,OFF,DCD
```

where

devname - Name of the device to be associated with this port
For example: COM1 or COM2

baud - The line speed for the port. This is any standard rate between 110-19200 baud.

parity - N(one), E(ven), O(dd), M(ark), S(pace) are the supported values. (specify only the first character of each value)

data - 7 or 8 (number of data bits)

stop - 0 or 1 (number of stop bits)

flow - OFF : No flow control
XON : XON/XOFF flow control
DTR : Hardware (DTR/DSR) handshaking

dcd - DCD : A separate thread will be started to monitor the CD signal to detect when it has been dropped.
OFF : The CD thread will not be started; if CD is dropped, the OS/2 COM driver will insert a break character (0xFF) into the input stream.

The punch emulation makes more sense when considered for VM. In this case the punch emulation is provided primarily for driving plotters. Plot files are often punched to RSCS or other virtual machines that drive a plotter. With a P/390, the extra virtual machine may not be necessary. If the plot file is generated as 80 byte records and contains all the necessary data for the plot, you can define a real CP PUNCH in your VM nucleus and have the 2703 emulator respond to VM like a punch, reading the plot (punch) records and driving the plotter. This method offers an easy way to support attended plotter operations. It is not recommended for unattended operation because little in the way of recovery or error reporting is available. How well this approach might work with OS/390 applications is unknown.

Installation and Debugging Information

The DMSTART command is usually in the IPL.CMD file. The following information is provided in the event that you experience problems and need to do more debugging.

To start the emulator, enter:

```
call dmstart 'AWS2703.EXE N "parms"'
```

(The "N" means that an OS/2 window need not be opened while processing this command.) The parms must be in double quotes to prevent OS/2 from splitting the line if an ampersand character is used. These parms include:

`/I=init_string` Use this to specify an alternate Modem Init string. The default init string is "ATE0Q1M1S0=1"

Example: 'AWSSTART AWS2703.EXE n "/I=ATE0Q1M1S1=2"'

This will override the default INIT string and set up the COM port to answer on the second ring instead of the first ring.

`/IO` This will create a file called AWS2703.DAT into which all outbound and inbound data streams are recorded. For this option to be enabled, I/O tracing for the specified 2703 device address must also be enabled. Use MANOPS Device Debug menu to enable I/O tracing.

You may have multiple uses of the AWS2703 device manager in your P/390 DEVMAP. One copy of AWS2703 will drive multiple lines or punches. In order to recognize a device as being online, the carrier-detect line is monitored. If the carrier drops, an intervention required error is sent to CP. When a device is disabled by VM, DTR on the port is dropped. It is expected that any attached devices (modems) respond properly (such as, hang up the phone, not allow a connection to be made until DTR is once again raised, and so forth).

If you are attaching a "smart" modem as a device, we recommend that result codes be disabled (ATQ1 on Hayes-compatibles) and autoanswer be enabled (ATS0=1). If everyone plays fair, the modem will not autoanswer when DTR is down.

VM provides more use for this device manager, and more information relevant to VM is found in the diskette DOC file.

4.7.4 AWSOMA Device Manager (P/390 only)

The AWSOMA device manager was designed for use with Optical Media Attach/2 functions. Briefly, these functions permit a CD-ROM to be read (by the S/390 operating system) as though it were a tape drive. Fixes and products for VSE and VM are sometimes provided in this format, although OS/390 (as far as the author is aware) does not currently use this format for products or fixes.

The OMA format is not limited to CD-ROM. You can use it to make OS/2 files readable by OS/390, and you can manipulate (by OS/390 programming, and using AWS2821 output to write an OS/2 file) the control list of which files are to be read by OS/390. This requires some planning and setup, but it can provide a fast, effective link to OS/2 files.

If you might use this technique, you should order the documents *Optical Media Attach/2 User's Guide* (SC53-1200) and *Optical Media Attach/2 Technical Reference* (SC53-1201) for detailed information about the control files and data formats used. Due to time constraints, we did not use these functions while writing this redbook. The following description is an overview of the basic operation.

When an AWSOMA device is included in a configuration, you must specify the name of an OS/2 file containing control statements. For example:

```
Addr Device Label Atype Size Mgr FN/P
>581 >3423 > > > > H >D:\OS\390\CTLFILFILE.TDF
```

The 3423 device type shown in this example may not be known in all OS/390 systems; a 3422 can be used instead. (The 3423 is a special read-only "tape drive" for use with OMA.) The D:\OS\390\CTLFILFILE.TDF file might look like this:

```
@TDF
C:\SOMEDATA.TXT TEXT
C:\PROD\XYZ.TXT TEXT
TM
EOT
```

If an OS/390 program reads from the tape drive at address 581 (in this example), the AWSOMA device manager will read OS/2 file C:\SOMEDATA.TXT, convert each text record (delimited by CR-LF) to EBCDIC, and send the record (as a tape block) to the OS/390 program. At the end of the OS/2 file, a tape mark is emulated. If the OS/390 program reads more, records from C:\PROD\XYZ.TXT are used, followed by a tape mark at the end of this file. Another read will produce another tape mark (for the TM line in the control file). Other options are available; consult the OMA documentation for details.

4.7.5 AWSPCSR (VM Only)

This device manager works with several special VM commands that are supplied with P/390 (and R/390) VM CD-ROMs. (There is no equivalent support for OS/390.) The device manager emulates a range of 3274 addresses (as though coax 3270s were attached) and interprets commands sent to these addresses.

The supplied VM commands include PCOPY (copy files either way between VM and the Server), LTRENAME (P/390 CM/2 only; rename the 3270 logical terminal session), OS2 (issue OS/2 commands from VM), and PIPE (allows CMS pipelines to issue Server commands and use Server files).

4.7.6 AWSMOUNT Command (P/390 and R/390)

This command is important. It simulates DASD and tape mounts for the OS/390 system. It also simulates tape rewinds for simulated tape drives that are really Server files. For example, if you IPL DFDSS from a simulated tape drive, and then want to IPL it again, you must "rewind" the tape. In practice, AWSMOUNT is mostly used with the AWSTAPE device manager.

The syntax is:

```
AWSMOUNT dev_addr [file_name] [options]
```

dev_addr is a three-character hexadecimal address

Optional Parameters are:

```

file_name is a FULLY QUALIFIED Server file name
/O causes the specified file name to replace the previous
  file name associated with the device. (For example, change a
  tape reel or disk pack.)
/R makes a device read-only.
/W makes a device read-write.
/Q query which file is currently associated with a device.
/D dismounts a device (such as a disk pack).
/? displays help information
/REW rewinds a simulated tape.
/RUN rewinds and unloads a simulated tape. No file is then
  associated with the tape device.
/C creates a new file (for tape output, for example).

```

An example of typical use is **D:> AWSMOUNT 580 /REW**. You can place an emulated tape file on a diskette, with a command such as **AWSMOUNT 583 A:TAPEFILE.XYZ /C**. If you do this, you must remember to **AWSMOUNT 583 /RUN** before removing the diskette.

Tapes emulated with the AWSTAPE device manager are used as labelled or unlabelled tapes. Suppose you have a job that will write a tape, and the JCL specifies a standard labelled tape with volume serial number TAPEAA.

```

//SYSUT2 DD UNIT=583,DISP=NEW,DSN=TAPEDATA,LABEL=(1,SL),
//          VOL=SER=TAPEAA

```

The JCL assigns the tape to unit 583, which (in our P/390 configuration file) is associated with device manager AWSTAPE. The entry in the configuration table does not associate an OS/2 file with this device.

Addr	Device	Label	Atype	Size	Mgr	FN/P
>	>	>	>	>	>	>
583	3420				E	

The key MVS console interaction will look like this:

```

$HASP373 OGDEN5   STARTED - INIT 1
IEF403I  OGDEN5   STARTED
IEF503I  INCORRECT VOLUME LABEL OR I/O ERROR
*IEF233A M 583,TAPEAA,,OGDEN5
          <----- at this point, switch to
                                an OS/2 window and enter:
                                AWSMOUNT 583 D:\DATA12 /C

IEC512I  LBL ERR 583,,      ,NL,TAPEAA,SL,OGDEN5
IEC704A  L 583,TAPEAA,SL,6250 BPI,OGDEN5
*O5 IEC704A REPLY 'VOLSER,OWNER INFORMATION', 'M' or 'U'
5,'TAPEAA'      <----- reply
IEE600I  REPLY TO O5 IS; 'TAPEAA'
IEC705I  TAPE ON 583, TAPEAA,SL,6250 BPI,OGDEN5
IEF234   K 583,TAPEAA,PVT,OGDEN5
IEF404I  OGDEN5 - ENDED
          <----- in OS/2 window enter:
                                AWSMOUNT 583 /RUN

```

The OS/2 command **AWSMOUNT 583 D:\DATA12 /C** will create a file on your D drive with the name DATA12. (You can use any valid OS/2 name and directory, of course.) The "/C" option is required when creating a "new" emulated tape volume. You need to respond to the standard MVS messages to create the appropriate standard labels. After the job ends, you probably want to "remove" the tape; use the **AWSMOUNT 583 /RUN** command. (RUN means Rewind UNload.) The file that was written (D:\DATA12 in this example) can later be

mounted to satisfy a mount request for another job, or used with OS/2 programs (if these programs are specially written to handle the AWSTAPE format).

You should enter a full path name when specifying a file name with the AWSMOUNT command. If you do not, it appears to default to the P390 directory.

At the time this was written, AWSMOUNT did not function with emulated CKD volumes, but did work with emulated FBA volumes.

Note that AWSMOUNT is not related to the MVS MOUNT command.

4.7.7 AWSSTAT (P/390 only)

The AWSSTAT command is not absolutely necessary, but you will find it useful when installing new device managers, or when making drastic changes in an existing configuration. It displays the status of the active device managers, and can manipulate their status. The AWSSTAT command is issued from an OS/2 prompt. All of the operands are optional. The syntax is:

```
AWSSTAT [manager_name] [/Q] [/C] [/W] [/Tseconds] [/D]
        [/L] [/K]
```

manager_name is the name of a particular device manager, such as AWSCKD

/Q queries the status of the whole P/390 subsystem. It sets a return code: 0 = running, 1 = not installed or bad driver, 2 = not powered on (?), 3 = P/390 device driver not in CONFIG.SYS, 4 = P/390 subsystem not running.

/C forces device managers waiting on semaphores to proceed

/W waits for all managers to initialize

/T waits a specified number of seconds for managers to initialize

/D displays managers that are still initializing

/L lists all active managers

/K kills all active managers and the P/390 subsystem

Most of these operands are for debugging by P/390 developers (or by someone writing their own device manager). The most common form that you would use is **D:> AWSSTAT /L**, which lists the device managers being used in your configuration. You would use this command after starting your P/390 subsystem and IPLing OS/390 (or whatever). If the only operand is the name of a device manager, AWSSTAT will kill that device manager.

4.7.8 AWSSTART

This command is used to start a device manager. You will not normally need this command unless you use the AWSSTAT command to stop a device manager. You can then restart it with the AWSSTART command. The syntax is:

```
AWSSTAT program-name [N] [parameters]
```

program-name includes the .EXE suffix, but need not include the path name, if the program is in the current directory

N tells AWSSTAT not to open a separate Server window for this program. N is almost always specified.

parameters are whatever additional parameters the device manager may need.

4.7.9 AWSPROF Command (P/390)

This command is used to identify the active DEVMAP (Configuration) file. It sets a variable in the OS2.INI file. You must set this value before using the P/390 configurator, or trying to use any other P/390 subsystem function. The command format is:

```
AWSPROF drive:\path\filename [/Q] [/C] [?]
```

The most typical command (issued from an OS/2 command line prompt) is: `D:> AWSPROF \DEVMAP.MVS`. The `/Q` (query current name) and `/C` (clear current name) are optional. By convention, the configuration device map files are always named `DEVMAP.xxx`, where the suffix is used to indicate different files. You need not enter the drive identifier if your OS/2 prompt is already at the right drive (drive D in the example above), but *you must enter the backslash before the file name*. (Normal OS/2 usage should not require the backslash, but this command appears to require it.)

The default DEVMAP file name is `C:\P390\DEVMAP.1VM`. Unless you happen to be using a device map file with this name, you need to use the AWSPROF command before doing almost any other P/390 function. (You only need to do it once; OS/2 saves the changed value.) As a side note, you will probably obtain an initial DEVMAP with your OS/390 system. If you need to create an initial DEVAMP, see 4.7.11, "AWSCFG Commands (P/390)." The `DEVMAP.1VM` configuration file is provided with the VM system that is prebuilt for the P/370. The `DEVMAP.MVS` file is provided with the OS/390 CD-ROM system discussed in this document.

The R/390 `awsprof` simply displays the name of the current device map; it cannot change the name. To change the current device map name for the R/390, you can enter an AIX command such as:

```
export CONFIG_FILE=/mvs/devmap.new
```

using the appropriate new file name. To permanently change the name, you would change `/etc/profile` or your `.profile`.

4.7.10 CLRIO Command (P/390)

This command, which has no operands, performs a function equivalent to a S/390 CLEAR I/O instruction for any control units and devices attached to a S/370 Channel Emulator/A adapter. The P/390 subsystem need not be running (but may be running) when you issue this command. It can result in lost interrupts or lost data in the P/390 OS/390 system. Do not use this command except as a last resort when attempting to work with channel-attached devices.

4.7.11 AWSCFG Commands (P/390)

The AWSCFG command provides the P/390 configurator. The command is normally issued automatically, when you click on the configuration icon. You can call it directly. One case when you need to call it directly is to create a new DEVMAP file. To do this, you should open an OS/2 window, go to the P390 subdirectory, and issue a command such as:

```
AWSCFG D:\DEVMAP.XYZ /NEW
```

The new DEVMAP may be given any name, although the convention is to always name it "DEVMAP," with a variable suffix name. You must specify a full path name, including the disk identifier. A newly created DEVMAP contains a single 3270 definition, which is really just a placeholder.

4.7.12 DEV2NAME

A DEVMAP file is not an ASCII file. You cannot display or print it cleanly. The DEV2NAME command reads a DEVMAP file (you can specify the file name as the operand, or let it default to your current active DEVMAP file), and writes a file named DEVMAP.NME in the same directory as the specified DEVMAP file. This output file is pure ASCII, and is structured for reasonable printing or for processing by a REXX program or other program.

You will find the output format obvious when you examine it.

4.7.13 IPL Command (P/390)

You normally IPL your OS/390 system by using the IPL icon in the P/390 window. (You start CM/2 sessions first, of course.) You can also IPL by selecting the IPL function in the Manual Operations window, but you cannot do the first IPL of a P/390 session this way (because the P/390 microcode has not been loaded yet). You can use the IPL command from a Server command line as another way to IPL a system. It will load the P/390 microcode, if necessary. The syntax is:

```
IPL addr [CLEAR | NOCLEAR | NOIPL] [/MAP devmap] [/MODE m] [/PARM xxx
```

The default is NOCLEAR. If *devmap* is specified, the IPL command will invoke AWSPROF to change the active DEVMAP to the specified file. MODE is either 370 or ESA. The default is set by your configuration environment selection. The PARM operand is a hex string (up to 8 hex digits) that is used as the IPL parameter. (OS/390 can use this to select an alternate nucleus, and so forth).

The NOIPL option is used to start the I/O subsystem, but not IPL the standard IPL address (specified in your "System Environment" options). After issuing an IPL NOIPL command, you could go to the Manual Operations window and perform an IPL from the address you specify there.

Some operands not listed here (WARM, CKPT, NOAUTO) are relevant only for VM systems.

4.7.14 ip1390 Command (R/390)

You normally IPL your OS/390 system by using the IPL icon in the P/390 window. You can also IPL by selecting the IPL function in the Manual Operations window, but you cannot do the first IPL of a P/390 session this way (because the P/390 microcode has not been loaded yet). You can use the IPL command from a Server command line as another way to IPL a system. It will load the P/390 microcode, if necessary. The syntax is:

```
ip1390 xxx [CLEAR | NOCLEAR | NOIPL]
```

where xxx is the IPL address. Note that there is no way to pass an IPL parameter. There are additional parameters that are more relevant to VM systems; you can list the ip1390 shell script to examine these.

4.7.15 LTRENAME Command (P/390 Only)

This command will place a name in the top line of a 3270 window, for the local 3270 emulated sessions provided by CM/2. Each 3270 window produced by CM/2 has a session name. The first is usually A, the second is usually B, and so forth. Examples of this command are:

```
D:> LTRENAME A OS/390 Console
D:> LTRENAME B First TSO Session
D:> LTRENAME C CICS Terminal
```

Use this command while CM/2 is active. The assigned names will not be retained across system startups. To make permanent name changes, use CM/2 configuration functions.

4.7.16 RDEVMAP Command (P/390 Only)

This command may be used to make mass changes to a DEVMAP file. You are unlikely to need this function unless you have a large DEVMAP. We strongly recommend you save a backup copy (under a different name) of your good DEVMAP file before experimenting with this command. (This command also saves a backup copy as DEVMAP.2VM, but you should have your own backup copy.) The syntax is:

```
RDEVMAP [d:\path\devmap.xxx] [ D=d | D=d n ] [ D=d:\old\ n:\new\ ]
```

If a DEVMAP file name is specified, it is automatically made the current DEVMAP (via the AWSPROF function). A copy of the specified devmap (before modification) is saved with 2VM as the suffix name.

The *D=d* operand will globally change the drive letter of all OS/2 files used to emulate P/390 DASD. The *D=d n* operand will change drive letter "d" to "n" every time it occurs in the DEVMAP file. The last operand form shown above will change a specified drive and path to a new one, for all occurrences in the DEVMAP file.

4.7.17 BLDLIST Command (P/390)

This command is not directly related to P/390 subsystem operation. It provides a convenient function when uploading or downloading a large group of OS/2 files to or from any host (including P/390 OS/390). The syntax is:

```
BLDLIST d:\path [ filemode | e:\path ]
```

The first operand specifies a path (directory level), not a specific file. This command generates output files that contain OS/2 SEND / RECEIVE or OS/2 COPY commands for all files in the specified directory. If a filemode (single character) is specified, the output files UPLOAD.LST and DOWNLOAD.LST are generated. The CM/2 short session name is assumed to be "A." If the second operand is a path name, COPY commands are generated in COPYFILE.LST.

A filemode character is not useful for OS/390, so you may need to edit the resulting files.

4.7.18 C370MAP Command (P/390)

This command provides a brief description of the operational environment (data rate, mapped devices) for the S/370 Channel Emulator/A adapter. There are no operands for the command. (The output lists a column for “VM”; you should read this as “OS/390.”)

4.7.19 AWSCMLT Command (P/390)

This command permits you to configure the local 3270 terminals managed by CM/2. You can do the same thing with the LT option of the configuration program (F7).

4.7.20 TRCC370 Command (P/390)

This command provides a trace of I/O commands and responses sent to the S/370 Channel Emulator/A Adapter.

4.8 Other Commands

The P/390 support software contains a number of commands that are only for VM or VSE users. That is, where VM or VSE is the operating system running in the S/390 processor. These commands include: SYSOWN, ALC, VMLOGON, and END_VSE.

Chapter 5. Other Topics

This chapter discusses a number of shorter topics that were not included in previous chapters, or that required additional comment.

5.1 S/390 Manual Operations

The P/390 and R/390 systems offer full S/390 manual operations functions. These include register displays, storage displays (with a variety of addressing), stop/start/reset functions, and a number of other displays and controls. These are all started from the Manual Operations icon in the base P/390 window (shown in Figure 15 on page 38). A primary Manual Operations window is opened by double-clicking this icon. The general form is shown in Figure 22.

The command bar options are:

- START/STOP - selecting this option toggles between the start and stop states for the S/390 processor. As on a mainframe, various control functions require the processor to be in the stopped state before the control function can be executed.
- Windows - selecting this option produces a drop-down menu with the following selections:
 - S/390 Registers - displays the standard registers
 - Communication Buffer - displays (and scrolls) a special area (not in normal S/390 addressable storage) that is unique to P/390 systems. This is primarily for P/390 developers; its use is not documented.
 - S/390 Memory (Absolute)
 - S/390 Memory (Real)
 - S/390 Memory (Virtual)
 - Primary address space
 - Secondary address space
 - Home address space
 - AR mode
 - IOCB -
 - I/O Timer -
 - Shared Memory - used by P/390 developers
 - SCHIB - permits display and alteration of the architected control block that represents an active I/O function
- IPL - provides a subwindow where an IPL address may be set, and an IPL started. This is an additional way to IPL (with an arbitrary address); the normal method is to use the IPL icon in the base P/390 window.
- Control - provides a drop-down menu with the following:
 - External Interrupt
 - Restart
 - Store Status
 - TOD Set Enable
 - Address Stop
 - Instruction Step
 - Device Interrupt - provides a function not available on mainframes, permitting the user to set status bits (such as attention or device end) and then create an I/O interruption
 - Device Debug - intended for developers, but usable for skilled S/390 systems programmers; there are four submenus, not described here

- Start Kernel Trace - starts a detailed P/390 (not S/390, and not in S/390 storage) trace. May be required when submitting a problem to the P/390 developers.
- Resets - provides several S/390 and P/390 resets:
 - System Reset Normal
 - System Reset Clear
 - Initial CPU Reset - stops the P/390 and clears outstanding interrupts
 - CPU Reset - stops the P/390 and clears outstanding interrupts
 - Load Microcode - requests the P/390 support programs (running in the Server) to reload the P/390 microcode. This function is intended for P/390 developers.
- Search - provides a window where the user may specify a byte string (in hex) and an address range (real memory), and initiate a search for the specified data
- LoadText - loads absolute object decks into S/390 storage. This is used by the developers.
- Exit - leaves the Manual Operations function
- Help - displays various help panels
- GreenCard - displays (and scrolls) the well-known S/390 reference card (although it is no longer green)

There are more manual operations than are available on some mainframes. They are not relevant to typical OS/390 users, but they are useful to hard-core systems programmers. We strongly suggest that new P/390 and R/390 owners spend some time exercising the manual operations functions to become familiar with them.

S/390 storage display is one of the most common uses of the manual operations functions. Storage display starts at address zero. You can overtype any address to display that address, and overtype data to change storage at the indicated address.

There are no significant differences between the P/390 and the R/390 manual operations functions.

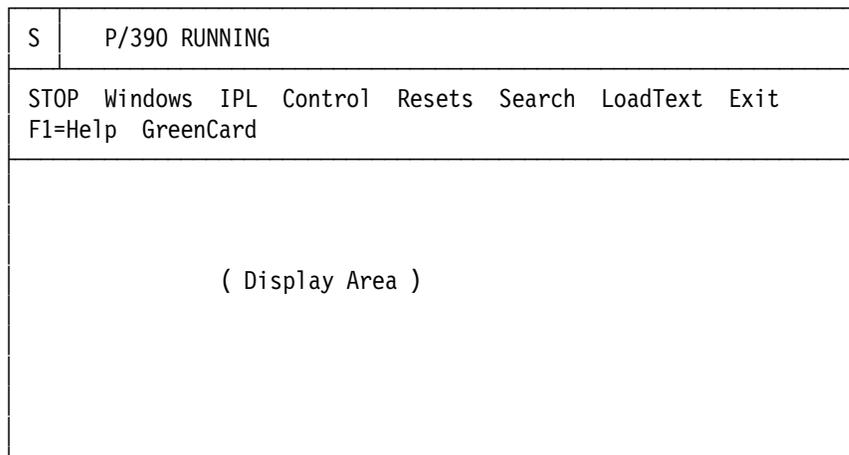


Figure 22. Manual Operations Window

5.2 Diagnostic Facilities

P/390 OS/390 has all the normal OS/390 debugging tools, of course. Additional debugging tools are provided by the P/390 subsystem environment. These fall into several areas:

- The S/390 Activity window is useful for detecting hard loops or wait states in OS/390.
- The Manual Operations window (already described; an example is in Figure 23 on page 124) provides many functions for examining S/390 registers and storage, setting address stops, and so forth.
- A unique P/390 subsystem trace can be written, showing key activities by various device managers.
- A unique P/390 snap dump may be requested from the primary P/390 icon window. The snap dump is for the P/390 developers and produces no information useful to other users.
- A trace unique to the S/370 Channel Emulator/A adapter is available.
- A trace from the emulated 3172 used for TCP/IP may be started. This provides several windows for monitoring activity. AWSERROR.LOG (in your P390 directory) is used by *some* device managers to log severe errors. You should check it periodically, because there is no automatic display of the contents.

The console functions available through the Manual Operations window should be familiar to experienced OS/390 systems programmers, and are used in the same way the functions would be used on a mainframe. The trace, snap dump, and comm area display are intended for P/390 developers and for anyone writing his own device manager. Although the formats are not documented, an experienced S/390 person will understand the IOCB, some of the trace data, and some of the SCHIB information. The comm area display and the snap dump are probably not useful for anyone other than a device manager developer.

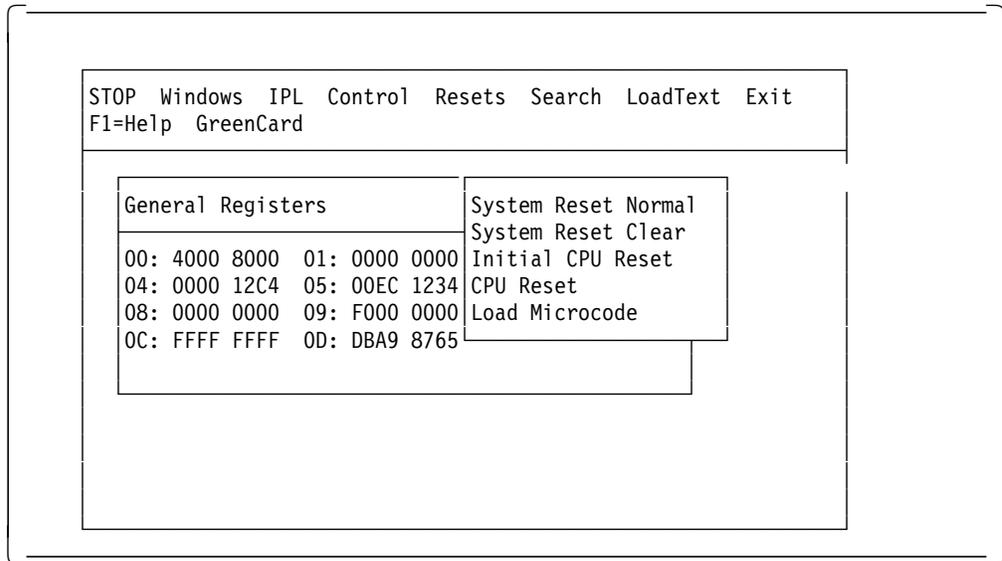
The reference diskette (or hard-disk equivalent) for the system contains a minimal hardware diagnostic check for the S/390 processor. A much more sophisticated S/390 diagnostic can be run by booting the diagnostic disk that is one of the four P/390 support program diskettes.

Trace Table

The P/390 subsystem provides several trace functions. By using the Trace icon, in the base P/390 window, trace data may be displayed or sent to a file. There are two trace types:

1. Kernel trace, that records all S/390-Server interactions
2. Device trace, that records only interactions associated with a specific emulated device

The normal trace table (this is the P/390 subsystem trace table, which has no relation to the OS/390 trace table) has 2000 entries, and wraps when it is filled. It is most useful for determining why emulated I/O failed. Each trace entry is four bytes long. There is no time stamp. You can view the trace table by double clicking the Trace Table icon in the P/390 window. The table is displayed in chronological order, oldest entry first.



and the Resets Pull-down Menu

Figure 23. Manual Operations Window with General Registers Subwindow

The trace table display includes text descriptions of each entry. This text is added by the display program. The actual trace data is shown as eight hexadecimal characters at the beginning of each line. There is no documentation available for the trace table. An experienced systems programmer will recognize the nature of many of the entries, and may find it useful for resolving device problems. The trace table is a critical debugging tool for anyone developing a device manager.

Some unconditional entries are written in the trace table. These include a few hundred entries associated with P/390 I/O subsystem startup, and a few severe error entries by various device managers. Otherwise, no trace table entries are written unless you enable a trace function. A kernel trace contains the most detail and writes a large number of entries very quickly. A kernel trace is started by selecting "Start Kernel Trace" in the Control pulldown in the Manual Operations window. It can also be started by an operand in the CHAN.TRC file.

For the P/390 only, the CHAN.TRC file (in the P390 directory) is one way to control device traces. The traces it specifies are in effect for the whole P/390 session. (It can also start the kernel trace.) This file consists of a single line, such as this:

```
KRN 00E 0C5 END (P/390 only)
```

The default CHAN.TRC file contains only the word "END." The KRN operand means to start a kernel trace. (It is not a good idea to specify kernel trace here; it has high overhead and is unlikely to capture anything useful.) In this case, 00E and 0C5 are S/390 device addresses (as specified in your DEVMAP). Trace records are written for any activity involving these units. The record should end with the END operand. The most typical use of this file would have a single device address, followed by END.

A kernel trace is not a superset of the I/O traces. It captures different types of data. I/O traces, limited to devices causing problems, are the most useful

traces. To be useful, they are typically obtained in a restricted, single-user environment.

For both P/390 and R/390, a device trace may be started from the “Device Debug” menu item in the Control window under Manual Operations. A trace option is offered. This trace is the same trace that is started with the CHAN.TRC file. In practice, this is the most useful method. You would start your OS/390 system and proceed to the problem point. You could then start a device trace (using the option in the Manual Operations window) to capture just the relevant information. When using this technique, double click on the trace icon as soon as possible after your problem occurs, so that the relevant trace entries will not be overwritten.

When you double click on the trace icon, the current trace table contents are captured and displayed. (The trace continues to run, but does not change the displayed information.) You can scroll through the displayed trace entries. You can click on “FILE” in the command bar of the trace display window. This will write the formatted trace table into AWSTRACE.LST in the P390 directory. This file name is fixed; an existing AWSTRACE.LST (in this directory) is overwritten.

In some cases, the 2000-entry P/390 trace table may not be large enough. This most frequently happens when the timing of a problem is uncertain, and you do not know that the problem occurred until some time after the event. In such cases, the key trace entries may be overwritten in the standard trace table. For the P/390, an additional set of programs, AWSTRCSP and AWSPSDD, are provided to write trace output to OS/2 files. These files can be very large, and should avoid the problems of overwritten trace entries. The setup and use of the AWSTRCSP functions is a bit complex, and is well explained in the AWSTRCSP.DOC file. The same function is not provided for the R/390 because its standard trace files are large enough to avoid overwrites in most cases.

Channel Adapter Trace (P/390)

A separate trace function is provided for the S/370 Channel Emulator/A adapter. It is started with one of these commands in a Server window:

```
D:\P390 > trcc370 ccw          (standard output)   (P/390 only)
D:\P390 > trcc370 debug       (specialized output) (P/390 only)
```

The TRCC370 function will save the trace data in file C370.TRC in the P/390 directory. The program needs the file C370TRC.CCW in order to format the output, and this file is included in your P390 directory.)

The format of the trace data is quite readable and might be used by any systems programmer familiar with low-level I/O operations. This trace function is probably more useful to an OS/390 person than the more general P/390 trace described in the previous section.

Channel Trace (R/390 Only)

The R/390 provides a special trace that is in addition to the traces described above. This channel trace is always active; there is no way to disable it. It is done very efficiently, and there appears to be little performance impact because of it. The R/390 creates two files:

```
/var/r390/ddlog0.0
/var/r390/ddlog0.1
```

Each file is limited to approximately 1 MB of data. The R/390 support programs write the ddlog0.0 file until it is full (1 MB), switch to the .1 file until it is full (1

MB) and then clear and rewrite the .0 file, and so forth. At the time this was written, these files were not cleared at IPL time, so a user must ensure he is looking at current data. More data is written in these traces than in the other P/390 and R/390 traces.

This trace function is intended only for the R/390 developers, and may not be present in future releases. We briefly document it here because it may be useful for advanced S/390 system programmers and developers. It is not a supported interface and IBM will not provide assistance for its use, or necessarily maintain the completeness or accuracy of the data. Some trace entries are relevant to specific S/390 operations, and others are relevant only to internal functions in the P/390 support programs.

Each trace entry is a single line, in one of the formats shown here. START indicates a SSCH or SIO instruction was executed. IS is the initial status for the I/O instruction. FS is the final status (or final for one CCW in a chain). CD indicates data chaining was processed. CSW and AW indicate I/O interrupts. PINTR indicates handling of pending or requested interrupts, and whether the interrupt was accepted. HostICB displays a S/390 ICB. FETCH refers to bytes taken from S/390 storage, to be written to a device. Only the first 32 bytes are traced. STORE displays data being sent to S/390 storage in response to a S/390 read I/O operation. READ displays the first three IDAs associated with an I/O operation; unlike other trace entries, it may be displayed before the START trace entry associated with it.

```

dddd-START-kkkkkkkk-oooo-eeeeeeee-cc-bbbb-ff-aaaaaaaa (SSCH or SIO)
dddd-IS-wwwwwww-hh-uu-vv-rrrr-ZERO (initial status)
dddd-IS-wwwwwww-hh-uu-vv-rrrr-SUCC-FS-hh-uu-vv-rrrr-CHAIN END
dddd-FS-hh-uu-vv-rrrr-CHAIN-eeeeeeee-cc-bbbb-ff-aaaaaaaa (final status)
dddd-FS-hh-uu-vv-rrrr-CHAIN END (final status)
dddd-CD-eeeeeeee-cc-bbbb-ff-aaaaaaaa (chain data)
dddd-CSW-ssssssss ssssssss ssssssss ssssssss-FF (sol I/O intrpt taken)
dddd-CSW-ssssssss ssssssss ssssssss ssssssss-F0 (sol I/O intrpt queued)
dddd-AS-uu-ssssssss ssssssss ssssssss ssssssss (unsolicited interrupt)
PINTR-mmmmmmm-dddd-ssssssss ssssssss ssssssss ssssssss-FF
PINTR-mmmmmmm-NO MATCH
HostICB: yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy T=tttttttt tttttttt
FETCH: xxxxxxxxxxxxxxxx...
STORE: L=bbbb F=ff D=xxxxxxxxxxxxx..
READ: IDA-1: xxxxxxxx,2: xxxxxxxx,3: xxxxxxxx

```

```

where aaaaaaaa = CCW data address
      bbbb      = CCW byte count
      cc        = CCW command code
      dddd      = unit address
      eeeeeeee = address of CCW (absolute address)
      ff        = CCW flags
      gg        = condition code (ICB response)
      hh        = host response
      kkkkkkkk = ORB word 1
      mmmmmmm  = I/O masks
      oooo      = I/O instruction operation code
      rrrr      = residual byte count
      ssssssss = CSW
      tttttttt = time, 2 words. (first in units of seconds,
                                second in nanoseconds)

      uu        = unit status
      vv        = channel status

```

wwwwwww = CAW
 yyyyyyy = ICB data

5.3 Tape Drives

You have several options for using tape drives with P/390 OS/390:

1. Use a channel emulator adapter to connect to a “real” IBM 3480/90 or IBM 3803 control unit. These control units have multiple channel interfaces and the channel adapter would represent just another channel attached to the control unit. Of course, the PC Server S/390 must be within channel-attachment distance to use this option. This is the least expensive way to use 3480/3490 or 3420 tapes, if you already have the drives for use with other systems. The 3480/90 drives will not operate at full speed (due to the current channel adapter), but performance is still acceptable (unless you have small, unblocked records on tape). This option has the strong advantage that the whole string of tape drives attached to the 3480/90 or 3803 control unit is available to the P/390 OS/390 system.
2. Purchase a SCSI-attached (external) 3480/3490-type tape drive, and attach it to the standard SCSI adapter in your system. This requires a “single-ended” SCSI adapter in the tape drive.⁷⁰ IBM does not market this type of unit. (The R/390 has a differential SCSI adapter; IBM does market a SCSI 3490E drive that matches this adapter.)

Matching the correct device manager to an OS/390 UCB is not always obvious. For example, SCSI3480 emulates a 3480 drive, regardless of whether the physical SCSI drive is 4mm, 3420-type, 3480-type, or 3490E-type. OS/390 should have a 3480 generated for the corresponding address. The ISITAPE device manager (version 2.0 or later) supports 3420, 3480, 3490 and 3490E devices (as defined in OS/390), using any of the SCSI drives shown in the following tables.

ISI has contributed the following information. (IBM makes no support statements for any particular SCSI tape model or vendor.)

Model	Type	IDRC	READ BLOCK ID	LOAD DISPLAY	SCSI34x0	ISITAPE
Fujitsu M2485B	3480	No	Full(1)	Yes	Yes	Yes
Fujitsu M2485H	3490	Yes	Full(1)	Yes	Yes	Yes
Fujitsu M2488C	3490E	Yes	Full(1)	Yes	Yes	Yes
M4 M490E	3490E	Yes	Partial(2)	Yes	No	Yes
M4 M490L	3490E	Yes	Partial(2)	Yes	No	Yes
Overland C480	3480	No	Partial(2)	No	Yes	Yes
Overland C490	3490	Yes	Partial(2)	No	Yes	Yes
Overland L490	3490	Yes	Partial(2)	No	No	Yes
Overland T490E	3490E	Yes	Partial(2)	No	No	Yes
Overland L490E	3490E	Yes	Partial(2)	No	No	Yes

⁷⁰ You can use a differential SCSI tape drive if you purchase a differential SCSI adapter for your system.

Model	Type	Density	SCSI34x0	ISITAPE
M4 9914	3420	800/1600/6250	Yes	Yes
M4 9906	3420	1600/6250	Yes	Yes
Overland 5622	3420	1600/6250	Yes	Yes
Qualstar 3414	3420	1600/6250	Yes	Yes
Qualstar 3418	3420	1600/6250	Yes	Yes

Emulated 3420/3480	SCSI34x0	ISITAPE
IBM 4mm (4/10GB)	Yes	Yes
HP 35470 4mm (4GB)	Yes	Yes
HP 35480 4mm (4/8GB)	Yes	Yes
Exabyte 8mm (8205)	Yes(3)	Yes
Exabyte 8mm (8505)	Yes(3)	Yes
DLT 2000 (20GB)	No	Yes
DLT 4000 (40GB)	No	Yes

Notes: (1) FULL includes format mode bit, (2) Partial does not include format mode bit, (3) P/390 does not officially support 8mm drives. LOAD DISPLAY refers to displaying the volser in a display that is part of the tape drive, indicating to the operator which volser to load. IDRC refers to standard 3480 compression. READ BLOCK ID refers to the read/search BLOCKID functions discussed below.

The device managers used with these drives are discussed in 4.4, "Device Managers for Tape Drives" on page 82.

If you do not require tape exchange with mainframes, only 4mm drives, 8mm drives (R/390 only), and AWSTAPE-emulated drives might be used.

IBM 3480-type tape drives offer several command functions that are not widely used. The High Speed Search (BLOCKID) commands are examples of commands that are not widely used. Some OEM 3480-type drives (using SCSI attachments) do not support these SEARCH BLOCKID commands. This becomes a problem only if an OS/390 application or subsystem uses these tape commands. Since this involves special programming, it is unlikely that many routine application programs would contain special code using these commands.

HSM (which is now DFSMSHsm) does use the High Speed Search CCWs with 3480 tape drives. If the tape drive does not support the command, the program fails. ISI⁷¹ has provided the information in the above table, indicating which drives are known to support High Speed Search. The list is not comprehensive. If you need the High Speed Search commands and your tape drive (or planned tape drive) is not listed, you should contact the tape drive vendor (or ISI, if you purchase from them) for more information.

⁷¹ Interprocess Systems, Inc., 11660 Alpharetta Highway, Suite 455, Roswell, Georgia 30076, telephone 770-410-1700. ISI is familiar with SCSI tape hardware and software used with the P/390.

At the time this was written, DFSMSHsm was the only major software user that we knew about. (If you are aware of other users, please send a note to the author at the address listed in the Acknowledgments section.)

We are aware one other restriction. It appears that a reading or writing a full 64K block *may* not work. By 64K we mean 65,535 bytes; we find that 65,534 byte I/O is consistently correct. OS/390 does not normally write blocks larger than 32K; only the dump/restore program is commonly used for longer records.

AWSTAPE Emulated Tapes

Be aware that the data format for an emulated tape drive contains extra material. This is generated and processed by the AWSTAPE device manager and is necessary because there is no other way to represent end-of-block and tape mark conditions. The format is explained in 4.4.1, “AWSTAPE Device Manager” on page 82. The AWSTAPE device manager can dynamically create new Server files when an output tape is written. The AWSMOUNT command is used to “mount” a Server file on a defined OS/390 tape drive (such as the tapes defined at 580-58F in the system discussed in this book).

The files used for emulated tapes contain OS/390 data. There is no automatic EBCDIC to ASCII conversion. The tape data can be used by AIX or OS/2 programs, but these programs must process the extra control information in the files and perform any required data conversion. These tasks are not difficult (although floating point or packed decimal data conversion is messy), and emulated tapes might be a convenient way to exchange data between Server and OS/390 applications. (Emulated 1403 printers and emulated 2540 card readers can be used the same way, and provide automatic EBCDIC-ASCII conversion in some cases.)

SCSI Tapes

3480 emulation means that the device manager will respond to the device commands available on these drives. 3420 drives have fewer device commands. In general, OS/390 application programs can use the two device types interchangeably. Ignoring questions about data compression, a 4mm tape written with the SCSI3480 device manager, for example, can be read using the SCSI3420 device manager and vice versa.

We found that we needed to use IEHINITT with completely blank 3480 tape cartridges when using our SCSI-attached 3480 drive, just like on our mainframe. We did not always need to use IEHINITT with new 4mm tapes, although it was sometimes necessary.

Data Compression

SCSI-attached 4mm and 3480/3490 drives may have data compression hardware features. Use of these features may not exactly mimic the operation of mainframe tape drives. A mainframe 3420-type drive does not offer data compression, although most will operate at two densities: 1600 bpi and 6250 bpi. The default is 6250 bpi when writing output; when reading, the existing tape density is sensed automatically. A 3480 drive may have an IDRC compression feature. Use of this compression must be explicitly requested with the DCB=TRTCH=COMP operand, which is intended for use only with standard-label tapes. You must also have configured the tape UCB with the COMPACT feature; HCD is used to add or change UCB features.

If a 4mm drive has a data compression feature, the SCSI3420 device manager will use that feature automatically for tapes that are emulated at DEN=6250 (which is the default for 3420 drives). If DEN=1600 is specified (through JCL or DCB parameters), the data compression feature is not used. Compression (if any) is by the tape device hardware. No software compression is offered. A compressed tape is automatically decompressed by the hardware (if it is present and compatible, of course) when the tape is read.

If a 4mm drive has a data compression feature, the SCSI3480 device manager will not use that feature unless OS/390 indicates that compression is to be used. If the user includes DCB=TRTCH=COMP on the appropriate DD statement, OS/390 will generate the appropriate CCW to enable compression.

If a SCSI-3480/3490 drive has a data compression feature, the SCSI3480 or ISITAPE device manager will not use that feature unless OS/390 indicates that compression is to be used. Compression is assumed to be IDRC-compatible. In all cases (as far as we know), uncompressed 3480 cartridge tapes are compatible with mainframe drives. *If you need to exchange compressed 3480 or 3490 tapes with a mainframe, you should verify compatibility with your SCSI-attached 3480/3490 drive as early as possible.*

Standard Label Processing

Under some circumstances OS/390 standard tape label processing for 3420 (or emulated 3420) tape drives issues read backward commands, with data transfer enabled. The P/390 support programs do not completely support read backward channel operations. At the time this was written, 3420 tape label processing is the only known instance of standard OS/390 functions using read backward processing.⁷² Three device managers are affected:

1. SCSI3420, which emulates a 3420 drive while operating a SCSI-attached unit.
2. CHAN370, which permits operation of “real” channel-attached devices, connect to a bus-and-tag interface.
3. AWSTAPE, which emulates a 3420 drive using an OS/2 file as the data media.

Note that all *real* 9-track tape drives support read-backward functions. No 4mm drive provide read backward functions. 3480 and 3490 drives will perform read backward functions if the block is not compressed. To deal with some of these variations, SCSI34x0 and ISITAPE will let 9-track drives perform the read-backward functions using their native hardware, and will emulate read backward on other types of drives. This is done by (1) backspacing a block, (2) reading forward, and (3) backspacing again. This permits compressed blocks to be read backward on 4mm drives, as well as on SCSI 3490 and 3490E units (when being used to emulate 3420 drives).

Changes have been made to some P/390 support modules to handle the particular read backward instance used by standard label processing. More general cases of read backward commands are *not* supported. At the time this was written, SCSI3420 (including AWS3420, as described in 4.4.2, “SCSI34x0 and AWS34x0” on page 86), ISITAPE, and CHAN370 should handle OS/390 standard label processing for 3420 drives. AWSTAPE did not, at the time this was written,

⁷² A few utilities, such as IEBCOPY, use read backward with data transfer suppressed. The P/390 support programs correctly handle this special case.

support the read backward command sometimes used by label processing. Nevertheless, we routinely use standard label AWSTAPEs; simple usage, writing an SL tape and reading it later, does not trigger the read backwards operations.

The developers are not aware of any other current OS/390 function that issues read backward commands.⁷³

5.4 XTAPE Product (P/390 Only)

This is a product marketed by Interprocess Systems, Inc., 11660 Alpharetta Highway Suite 455, Roswell, Georgia, telephone 770-410-1700, fax 770-410-1773. XTAPE is a simple utility that copies files between SCSI tapes and OS/2 disks. It is most useful for backing up (and restoring, if needed) the very large disk files used to emulate CDK or FBA volumes. It runs under OS/2 and uses the ISITAPE device manager to perform all of its tape I/O operations. This makes it possible for ISITAPE to coordinate the use of a given SCSI tape (such as the 4mm drive) from the S/390 environment and from OS/2 using XTAPE. This means that SCSI tape units can be shared between the two environments without changing CONFIG.SYS and rebooting OS/2.

The sharing of a SCSI tape drive is determined by which environment has claimed control of the tape unit. If the tape drive is considered online to the OS/390 environment, XTAPE is denied access to the drive. If you vary the unit offline to OS/390, then XTAPE can use it (from the OS/2 side).

XTAPE is simpler than typical OS/2 tape backup products. It is well suited for use with the relatively few files that constitute emulated CKD (or FBA) volumes. It is not so well suited as a general purpose OS/2 backup mechanism, where hundreds (or thousands) of files in multiple directories need to be managed. The style of usage is more attuned to S/390 tape usage than to PC tape usage.

For example, we used XTAPE to back up CKD volumes, with one volume per 4mm tape. This wastes much of the 4mm tape capacity, but is very similar to the backup style used under OS/390. XTAPE is not limited to this mode, of course.

XTAPE is an OS/2 program which, in its most direct usage, is invoked from the command line to execute one function. It can also be loaded and called interactively, using a set of entry points. The basic command-line functions are:

DUMP	Dump OS/2 files, using file names (wildcards allowed)
LOAD	Restore one or more files from tape
LABEL	Write a standard label on tape
SCAN	Scan an XTAPE and list the contents
BSF	Backspace a number of files
BSR	Backspace a number of records
FSF	Forward space a number of files
FSR	Forward space a number of records
REW	Rewind tape
RUN	Rewind and unload tape
WTM	Write a number of tape marks

⁷³ If you know of any current products that do this, please send a note to the author. (Do not list older SORT programs! Their use of read backward is well known, but we think it is in the category of ancient history.) Also, the author is interested in understanding *exactly* under what circumstances the commands are used.

5.5 OS/390 Device Planning

Most of the device examples in this document are for the Application Developers' CD-ROM for MVS 5.2.2. The Application Developers' CD-ROM for OS/390 Release 1 was not available in time for us to use it for general examples. Base information is provided here for the first release of the Application Developers' CD-ROM for OS/390.

This version uses 3390 volumes for the IPL volume and the DLIB volume. The volume names, sizes, and server file names are:

volser	type	size	OS/2 (or AIX) file name(s)
OS39R1	3390	2.8GB	OS39R1_1.A80 and OS39R1_2.A80 (IPL volume)
SCPMV5	3380	1.2GB	SCPMV5.122 (paging, spool, catalog, etc)
OS39H1	3380	.2GB	OS39H1.124 (SMS-managed, for HFS)
P390DX	3380	1.2GB	P390DX.123 (CICS, IMS, DB2)
OS39D1	3390	2.8GB	OS39D1_1.A81 and OS39D1_2.A81 (DLIBs)

Due to 32-bit design limitations in OS/2 and the current AIX, no single file is larger than 2GB. The 3390-3 volumes used for the IPL and DLIB volumes are larger than 2GB. The P/390 support programs automatically split these large 3390 volumes into two server files, and this is the reason for having two files for each of the 3390-3 volumes.

If you generate your own OS/390 system, you should give some thought to what devices should be generated. In principle, nothing unique is required for P/390 usage. In practice, you might want to plan ahead and consider these points:

1. A 2540 card reader. Very few of us still have a 2540 card reader in use. However, you should consider defining one in your P/390 OS/390 system. Only one can be used. It can be used for job submission from OS/2 or through a LAN file server, AIX, NFS, and so forth.
2. Several 1403 printers. You can define 1403 printers for JES2 output. The AWS2821 device manager can intercept this output and (after automatic ASCII conversion) send it to a file (one per defined 1403) or to PC printers. Consider defining four or five 1403s.
3. A 2540 card punch is probably not needed, although a very specialized use for one exists.
4. Consider defining three channel-attached non-SNA 3174 controllers, with up to 96 addresses for 3270 terminals. (For OS/390, you define the associated 3270 terminals, not the controller. Ninety-six terminals may be excessive, but consider defining terminals as though three separate 3174 control units were being used.) if possible. By doing this, you will be able to attach a "real" 3174 (if you have one available) in addition to the emulated 3174-type connections via LAN.
5. Consider generating one string of 3420 tape drives and one string of 3480/3490 drives, with control unit and device addresses that match "real" units in your installation. Your P/390 OS/390 system can be attached (via the S/370 Channel Emulator/A adapter) to the "real" drives. Avoid addresses that end in X'00'.
6. Generate several (at least eight) 3420 tape drives, for the AWSTAPE and SCSI3420 device managers. Specify OFFLINE=YES; this will prevent OS/390

from trying to sense tape labels on drives associated with the AWSTAPE device manager.

7. Generate several 3480 tape drives for the SCSI3480 device SCSI-attached units you intend to use.
8. Generate at least eight 3088 (CTC) addresses, for use with LCS3172, WAN3172, and LAN3172 device managers.
9. Generate more 3088 (CTC) addresses if you might attach a “real” 3172 via the S/370 Channel Emulator/A adapter. Avoid addresses that end with X’00’.
10. Generate an NCP address if there is any chance you might connect to a “real” 37x5 controller.
11. Remember that there is a maximum of 255 devices in a DEVMAP. OS/390 can have more generated devices than this, and will typically have more devices generated than are defined in a DEVMAP. In general, OS/390 should have all the devices you *might* use, and the current DEVMAP should have only the devices you *are* using.

Your I/O configuration is defined twice: once to OS/390 and once to the P/390 configuration program. The P/390 subsystem configuration definitions are very easy to change. The OS/390 definitions are not as easy to change, especially if the system owner does not have recent experience installing and managing OS/390 systems with HCD. We make the following suggestions:

1. Try to plan ahead for your OS/390 I/O configuration, and include devices you may need in the future. (This advice must be used within reason, of course. A huge I/O definition, such as one “borrowed” from a large multisystem mainframe installation, has its own disadvantages.)
2. For your P/390 subsystem configuration definitions include only devices you use at present. It is very easy to extend these definitions when you need to do so. There is no reason to define to the P/390 configurator all the devices that are defined (but unused) in your OS/390 system.

5.6 Security

How secure is a P/390 OS/390 system? This can become a complex question.

P/390 OS/390, as seen from a TSO or CICS terminal, has exactly the same security attributes as a mainframe OS/390 system. RACF (or other security monitor) controls and application designs are involved, and have the same concerns as with any mainframe OS/390. However, there are some unique concerns for P/390 and R/390 systems.

Physical access to the computer is a key concern. Physical access includes hardware access (press the buttons, take the disk drives out, and so forth) and logical access to the “primary” operator display/keyboard. A mainframe is usually in a highly-controlled environment, on a raised floor, behind secure doors, and with an operations staff in attendance. In general, P/390 OS/390 systems will not be in such secure environments.

Particular security concerns include:

- Access to the OS/390 primary console or the Manual Operations functions. With such access, a skilled user could bypass RACF controls (by changing

bits in memory), or start trusted tasks (such as writing backup tapes), and so forth. An OS/390 system will not be more secure than the security of its hardware and MCS console.

- Access to the emulated OS/390 DASD through Server programs. In principle, an OS/2 or AIX program can inspect or change OS/390 DASD data. In practice this may be difficult, because an emulated DASD is a complex object (viewed from OS/2 or AIX). The offending program would need to simulate substantial parts of volume management (find and follow DSCB pointers), deal with the unique AWSCKD device manager formats, and so forth, or employ brute-force searching techniques.
- Destruction of emulated DASD. A complete emulated volume is a single Server file, and can be deleted with a single Server delete command. This can happen accidentally, especially if meaningful file names are not used, or it could happen via LAN if peer-to-peer sharing is present. (Or if the file itself is on a LAN server, of course.)
- Copies of valuable data, such as on backup tapes (probably 4mm tapes, when using your P/390 OS/390). When a backup tape is old and recycled for other use, it still contains the old data. Even your old data may be valuable to someone.
- Theft of the whole system, or of disk drives from it.
- There are unique security problems when TCP/IP telnet is used to emulate locally-attached 3270 terminals. It might be possible, for example, for an outside user making a telnet connection at just the right time, to become the OS/390 master console if an IPL has just been started. This is discussed in 4.3.2, "AWS3274 (R/390)" on page 76. An unwanted connection as the master console is rather unlikely, due to a narrow timing window, but telnet connections for general 3270 connections are open to any TCP/IP who can access your network.
- A telnet (or ftp) connection to the Server (OS/2 or AIX) might be an exposure (especially with OS/2, where very few security controls are available in this area). If a user on the Server can access the files used to emulate DASD, in principle the user could make any changes he wants to OS/390 files. In practice, this is not easy because emulated DASD files are very large, and in an undocumented format (and contain EBCDIC character data). The most likely attack (from a server program) is to scan the file, looking for keywords (in EBCDIC). Crude programs of this nature are easy to produce.

One common thread is access to your physical system. If security is important, you must consider your physical security requirements. The other common thread is TCP/IP access to the Server itself, and to 3270 device managers running on the server.

5.7 OS/2 and Hardware Tuning

There are many areas and parameters for tuning a P/390 OS/390 system. Many of these are "standard" areas, such as the multitude of SYS1.PARMLIB members and options. Others are more unique to the P/390 supporting subsystem.

RAID Array Considerations

On array models of the PC Server, the customer sets the stripe unit size.⁷⁴ The default stripe size is 8KB. Valid choices are 8KB, 16KB, 32KB, and 64KB. Sizes larger than 8KB may yield better performance for OS/390 workloads than the default 8KB size. *Once a stripe size is selected, and data written on the drives, changing the stripe size will cause loss of all data in the array.* If you are using significant OS/2 applications, in addition to OS/390 on the S/390 side, you must consider the characteristics of those applications before selecting a stripe size.

There are two choices for “write policy” when using a RAID adapter. The default write policy is *write-through* (WT), where I/O completion status is sent after the data is written to the disk (or, perhaps, written in the disk drive’s cache -- depending on the disk drive used). To improve performance, you can change this write policy to *write-back* (WB), where the I/O completion status is returned as soon as the data is copied into the RAID adapter’s cache, but before it is actually written to a disk. Of the 4MB RAID adapter memory, over 3MB is available for this use.

Obviously, you have slightly more risk using the write-back process. If there is a power failure before data is actually written to disk, you cannot know exactly which records were written and which were not written. An uninterruptible power supply (UPS) helps with this exposure, and in other areas as well.

OS/2 CONFIG.SYS

The MAXWAIT parameter in CONFIG.SYS defines the number of seconds that an OS/2 thread waits before being assigned a higher dispatching priority. Applications that are I/O intensive could benefit from setting MAXWAIT=1 in CONFIG.SYS. (The default is 3.) Since OS/390 operation is very likely to be I/O intensive, we recommend setting MAXWAIT=1.

If your system has no FAT-formatted partitions, the DISKCACHE= device driver can be removed (commented out) of your CONFIG.SYS. This will save memory on the OS/2 side. By default, OS/2 places this device driver in CONFIG.SYS. If you have FAT-formatted disks or disk partitions, you may tune the size of the DISKCACHE. Enter “HELP DISKCACHE” (in an OS/2 command line) for more information.

The PRIORITY_DISK_IO= parameter controls whether or not an application running in the OS/2 foreground receives priority (for disk accesses) over an application running in the background. OS/390 (meaning the support functions and device managers for the S/390 side) is considered to be in the background. An OS/2 command-line window, for example, is considered foreground. We suggest you specify PRIORITY_DISK_IO=NO. (The default is YES.) This will ensure that OS/390 support functions are processed at the same priority as OS/2 foreground functions.

The HPFS.IFS device driver delivered with OS/2 has a maximum cache size of 2MB. It is specified by the CACHE:nnnn parameter of the HPFS.IFS device driver statement. The default is 10% of available RAM, with a maximum of 2MB. The default value will depend on the RAM available *at the time of installation*. We recommend you specify /CACHE:2048 for this parameter (assuming you are using HPFS file systems, of course).

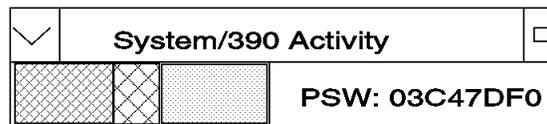
⁷⁴ A stripe is the amount of data written on a given disk before writing on the next disk in the array.

The /CRECL parameter on the HPFS.IFS device driver statement specifies the size of the largest record (that is, the length sent by a single write operation) that is eligible for the cache. The OS/2 default is 4KB. Since emulated CKD DASD always involved I/O in track-size blocks, you need to set /CRECL to a larger value than a 3380 (or 3390) track size if it is to be effective.

The HPFS driver defaults to LAZY writes. That is, I/O completion is signaled to an application when the written data is placed in the HPFS cache. LAZY writes provide a significant performance enhancement, and we suggest you keep the default parameter. LAZY writes have the same exposures as write-back cache in the event of power or system failures.

5.8 System/390 Activity Window

The System/390 Activity window is started automatically whenever the P/390 I/O subsystem is started (normally by the IPL P/390 icon). The window looks something like this:



It is the lower-left segment that is interesting -- shown as shaded in the drawing here. This is a very dynamic bar graph. It indicates the general activity of the processor, averaged over the last few milliseconds.⁷⁵ The bar graph is in color, and the colors have the following meanings:

- Blue - wait state
- Red - supervisor state activity
- Green - problem state activity
- Pink or Purple - waiting for the Server to complete an S/390 SSCH instruction (by returning CC=0, for example)

You do not need this information to install or use OS/390, but it provides reassuring indications ("my OS/390 is running") and a rough idea about how heavily it is loaded. The supervisor versus problem state information is less useful, since it is so easy to misinterpret this ratio.

The OS/2 "Pulse" window can also be interesting (if you can find room for it on your desktop), because it provides an indication of how heavily loaded the I/O subsystem (that is, OS/2) is. With a busy OS/390 system, the I/O subsystem may be the primary bottleneck.

Both the System/390 Activity window and the OS/2 Pulse window may be resized to be relatively small. We routinely have both windows running (although this does steal some amount of processing power from the OS/2 system).

⁷⁵ It appears to be updated about 10 times each second, but we have not seen any specifications that indicate this.

5.9 Device Manager Letters and Names

The single characters used to indicate different device managers in the P/390 configuration display can be confusing. The DEVMAP, as stored on disk and as used by the P/390 subsystem, contains the full name of the device manager used by each defined address. The single characters are used solely to save space in the configurator display panel.

The character-to-name association is not fixed. It is established by the AWSCTYPE.MGR file. The characters are simple sequence numbers (1...9, A...Z) for the names in this file. You can change the sequence of this file, but we suggest that you do not. Because the single-character identifiers are used in the configuration file display, they appear in many examples in this and other documents. It is tempting to reference them as fixed assignments, "Use device manager 4 for those terminals," for example. Please do not do this; use full device manager names for documentation and communication whenever possible.

At the time this was written, the associations were:

Character	P/390	R/390
1	AWSFBA	AWSFBA
2	AWCKD	AWCKD
3	AWS3274	AWS3274
4	LAN3274	AWS3215
5	AWS3215	AWS2821
6	AWS2821	AWS2540
7	AWS2540	LAN3172
8	LAN3088	LCS3172
9	LAN3172	WAN3172
A	LCS3172	AWSPBS
B	MGR3172	AWSTAPE
C	WAN3172	
D	AWS2703	AWSOMA
E	AWSICA	AWS9346
F	AWSPBS	AWSPCSR
G	AWSTAPE	CHAN370
H	SCSI3480	AWSX25
I	SCSI3420	SCSI3480
J	AWSOMA	SCSI3420
K	AWS9346	
L	AWSTFA	
M	AWSPCSR	
N	CHAN370	
O	AWS3480	
P	AWS3172	

If you added the AWS3420 device manager after the last line in AWSCTYPE.MGR, it would become device manager Q for the P/390 in this example.

5.10 3270 Emulator Keyboards

One of the challenges in using 3270 emulation programs is to find the keys used to emulate various 3270 keyboard functions. CM/2 has several nice functions for this:

- Move your mouse pointer to anywhere in a 3270 window, and click the right-hand button. This produces a small window where you can (with a

mouse click) execute PA1, PA2, CLEAR, RESET, ErEOF, ErINP, ATTN, and SysReq functions. Click (left-hand) anywhere outside this small window to remove it.

- Click on the “Keyboard” option in the top menu of a 3270 window. Then click on “Remap keyboard.” This will produce a graphic presentation of the keyboard. Click on any key to display the various uses of that key.
- Use the other keyboard menu options for performing unusual functions.

With the default CM/2 keyboard arrangement, common functions are:

- ENTER - right Ctrl key or numeric keypad Enter key
- RESET - left Ctrl key
- CLEAR - Pause key
- PA1 - Alt-Insert keys
- PA2 - Alt-Home keys
- ErEOF - Cntl-Delete keys
- ATTN - ESC key
- “not” symbol (used in PL/I) - left square bracket key
- vertical bar - right square bracket key
- cent sign - shift 6 (replacing the carot symbol)

5.11 Writing a Device Manager

Writing your own device manager, typically done to interface a new PC device as a emulated OS/390 device, is not a trivial task. Two different interfaces are available. One is programmed in C (or C++ or PL/I), and the other in PC assembly language. Examples of both are included in ZIP files on the P/390 support program diskettes.

The sample programs (including header files, link steps, source files, executable files, and object files) have many comments, but are not intended as ground-floor education for writing device managers. When UNZIPed, there are 29 files in the two examples.

We do not recommend trying to write a device manager unless you have unusual requirements that cannot be met any other way. If you *must* do this, we recommend that your IBM representative contact the P/390 development group.

5.12 Send and Receive

If your OS/390 system has IND\$FILE⁷⁶ installed, you can use the OS/2 commands SEND and RECEIVE to move files between your OS/2 and OS/390. (These same commands are provided with many 3270 emulator products.) SEND and RECEIVE can be used as line commands (in an OS/2 window), or the menu functions of the 3270 windows provided through CM/2 have panels and prompts to do the same thing.

While most PC-based 3270 emulator programs provide Send and Receive commands, fewer telnet 3270 emulators provide these functions. If your OS/390 system uses direct TCP/IP functions (which would use the LCS3172 device manager), you can use ftp to transfer files. If you use telnet 3270 emulators that

⁷⁶ This is the primary module provided by the *3270PC File Transfer* program product.

do not support Send and Receive commands, and your OS/390 does not provide TCP/IP connections, you will need to work harder to exchange files between a workstation and OS/390. One way would be to transfer a file from a workstation to the AIX server of the R/390, and then use AWS2540 card reader emulation to read the file. This multistep approach would work for simple line data where each record does not exceed 80 bytes.

5.13 Backing Up OS/390

You have two approaches for backing up your OS/390 system:

1. Use standard OS/390 backup techniques, usually the ARDRSSU program.
2. Use Server backup techniques.

With ARDRSSU, you normally dump an OS/390 volume to a number of tape cartridges. If you use the 4mm drive, you will find that any 3380 volume will fit on one tape, and there is room for several more 3380 or 3390 volumes on the same tape. That is, the 4mm tape holds much more data than a 3480/3490 cartridge. The advantage of using ARDRSSU (or something similar) is that OS/390 understands the contents of the backup tape. You can restore a single OS/390 file from the tape.

An OS/2 backup utility, such as Novaback⁷⁷ or XTAPE may be easier to use. The limitation is that OS/390 does not understand the format. This means that you cannot restore a single OS/390 file from the backup tape. You must restore a whole 3380/90 volume (which is a single file as far as OS/2, XTAPE, or Novaback are concerned). The same comments apply to AIX backup functions such as tar, backup, ADSM, and so forth.

When using an OS/2 backup program, you should not have any significant OS/390 activity. In general, we recommend stopping the P/390 subsystem while using a Server backup program.

We used both backup forms, but used Novaback or XTAPE more often -- simply because it was easier.⁷⁸ We recognized that an OS/390-format backup is probably better, because a single OS/390 file can be restored from it, so we sometimes used this format.

One factor in selecting a backup is convenience in respect to changes to CONFIG.SYS. In general, OS/2 backup programs are supplied with their own device drivers. You must install the proper device driver in CONFIG.SYS before the backup program can be used. If the same tape drive (typically the 4mm drive) is also used by OS/390, then the appropriate device driver (SCSI3480, or that series) must be installed in CONFIG.SYS. You cannot have two device drivers in CONFIG.SYS for the same tape drive. Thus, you would need to change CONFIG.SYS and reboot in order to change "ownership" of the tape drive between the driver for the OS/2 backup program and the driver for the P/390 device manager. XTAPE is an exception here, since the same driver is used for both OS/390 access and OS/2 access. (The R/390 does not have problems with

⁷⁷ From Novastor Corporation, 30961 Agoura Road, Suite 109, Westlake Village, CA 91261, telephone 818-707-9900. Novaback is often supplied with 4mm tape drives, although it is not limited to this format.

⁷⁸ Be careful: left to their own defaults, many OS/2 backup programs will backup *everything*, including a CD-ROM (if it is mounted).

driver conflicts; the same low-level access is used by the R/390 device managers and most AIX programs that use tapes.)

5.14 OS/390 Performance and Capacity (P/390)

Discussing performance or capacity planning for P/390 OS/390 is difficult for a number of reasons:

- As with many smaller systems, minor changes in the workload and timing of the workload can have major impacts on measurements. Larger systems tend to “average out” such changes.
- Assumptions that are valid for a pure S/390 environment may not be valid for the mixed S/390 - OS/2 - Server environment. This is especially apparent in the emulation of CKD (and ECKD) devices.
- OS/390 tuning is often concerned with data set placement on disks and channels. The emulated environment, especially for RAID systems, does not respond to such changes in the same way.
- Mainframes are designed to provide a balance between I/O capacity, storage, and CPU cycles. A P/390 OS/390 system lacks the same balance. In a sense, it has too much storage (128MB) and not enough I/O bandwidth to balance the S/390 CPU capacity.
- Performance and capacity of the Server are factors, and sometimes critical factors, in the performance and capacity of OS/390. The P/390 device managers, which provide all I/O to OS/390, run on the Server and are affected by its capacity and tuning.

There are many potential areas for basic P/390 OS/390 tuning, without going into the very detailed OS/390 tuning elements such as SRM, PLPA packing, and so forth. These areas include:

- Numbers of users (batch initiators, TSO users, CICS users)
- Nature of workload: TSO only, CICS only, batch only, mixed online and batch, nature of TSO and CICS transactions, use of DB2, VSAM, and so forth
- Reasonable data set placement on emulated DASD volumes
- Number of emulated DASD volumes
- Use of VIO instead of disk
- Amount of paging that is acceptable
- Customization of various parameters and procedures, such as LOGON procedures and ISPF startup procedures
- Use of traditional TSO functions versus use of Open OS/390 functions
- Extensive use of ISPF functions and product interfaces, in addition to the use of the base ISPF functions (options 0, 1, 2, 3, 6).
- Use of RAID or non-RAID units (Other considerations usually require the use of RAID, but the comparison may be interesting. Mixed RAID and non-RAID systems are also possible.)
- RAID stripe size
- Number of RAID (or SCSI) adapters
- OS/2 disk cache controls

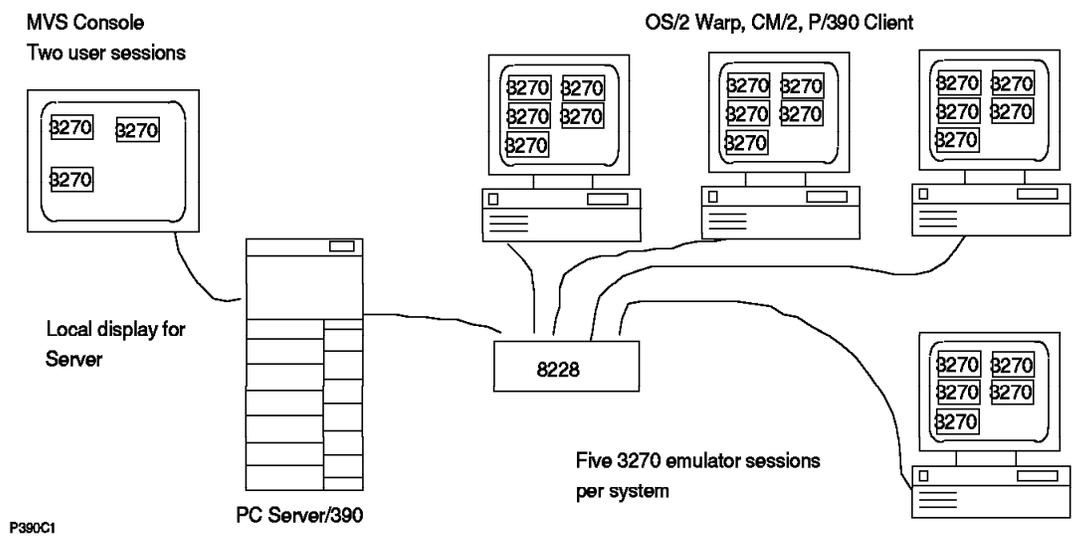


Figure 24. Measurement Configuration

We could not try all permutations of these areas within the constraints of the relatively small project described in this document. We selected the options and paths we thought were most interesting and that could be tried with the manpower and time available.

The results reported in the following sections were obtained on a non-RAID P/390 system. It was non-RAID because that was the system available for our use at the time we made the measurements. We doubt that the results would change much if rerun with a RAID system.

Workload Scripts

We needed several fixed workloads to be used while tuning the system. That is, workloads that were repeatable and measurable. A number of such standard workloads are available, and have been used for formal IBM capacity measurements. These standard workloads tend to be large and complex, especially for terminal environments. Our project was small, with less formal goals, and we felt that quite simple workload scripts were more suitable. These are described here.

For batch processing we produced a single jobsteam, in a sequential file, that could be submitted from TSO. It contained the following jobs:

- JOB1 - a moderately large assembly.
- JOB2 - an IEBCOPY step that copied SYS1.VTAMLIB to a temporary PDS.
- JOB3 - compile, link, execute a small COBOL program, which reads a small file (1000 records), sorts it, and prints it.
- JOB4 - use IEBDG to generate 200,000 records and then copy them (IEBGENER) to another data set.
- JOB5 - a trivial IEFBR14 job.
- JOB6 - use IEHLIST to list the VTOC of SCPMV5 (one of the standard disks on the developer's CD-ROM).
- JOB7 - a very small C program compile, link, execute.
- JOB8 - use IEBDG to generate 250,000 (80 byte) records and then use SORT to sort them.
- JOB9 - use IEBTPCH to print several members from SYS1.PARMLIB.

- JOB10 - a trivial IEFBR14 job.
- JOB11 - a large assembly.
- JOB12 - a trivial IEFBR14 job.
- JOB13 - use IDCAMS for a full listing of the master catalog.
- JOB14 - a small COBOL compilation.

In general, we simply measured the elapsed time to complete this job stream, using the timestamps in the SYSLOG.

We wrote a TSO session script, using REXX and EHLLAPI. The script contained random elements for think times, and a random choice of four different paths through the script. All elements of the script were always executed, but in different orders. The script included:

- Use of ISPF to both browse and edit files.
- A simple use of SDSF.
- Various basic TSO commands, such as ALLOC.
- Two submitted jobs: a small COBOL compile and an IEFBR14 job.

Much of the time of the script was spent in ISPF, scrolling through a sequential file that was being edited. While the exact sequence of commands was not representative of a normal TSO user, we thought the system load generated may be typical of a light-to-average TSO developer.

We ran parallel instances of the TSO script. We used a token-ring LAN to connect four OS/2 systems. On each of these systems we installed CM/2 and the client code provided with the P/390 that permitted a LAN 3270 session to appear as a local coax session to OS/390. Each OS/2 CM/2 system had five emulated 3270 sessions, and the TSO script was run (in parallel) in each session. By using four of these systems, we could run up to 20 TSO sessions in parallel.

When the TSO script was started, the command arguments specified the emulator session identifier, the TSO user ID, and the associated password. In addition to standard system data sets, the TSO script referenced two user-owned datasets. These were userid.TEST.SEQ (containing a sequential file used for ISPF edit) and userid.TEST.LIB (containing three members: one for browsing and two with jobs to be submitted).

We prepared twenty six user IDs (P390A - P390Z) for use with the TSO script, and prepared twenty six copies of the user-owned files.

When the TSO script ended, it reported the total elapsed time and the average response time for trivial, standard, and major commands. The script consumed about 10 seconds of S/390 CPU time (per user), and generated about 370 EXCPs (both numbers reported by the DA option of SDSF). Other measurements, using the P/390 Manual Operations I/O counts, indicate about 700 I/O's per user (when averaged down to a per user number); this would include submitted jobs, logon/logoff overhead, and what little paging occurred.

Due to cacheing by AWSCKD, OS/2, the SCSI/RAID adapters, and caches on disk drives, it is difficult to translate these I/O counts into true PC disk I/O counts.

Starting Configuration

We started with the S/390 Developer's Association version of OS/390 5.2.2, as available on CD-ROM. We made these changes immediately:

- We created two additional emulated 3380 volumes, placed on separate PC disks:
 - TSOPAK, where we placed all the user-type files associated with our batch and TSO workloads.
 - WORK01, a scratch volume.
- By changing SYS1.PARMLIB(VATLST) we made WORK01 a storage volume, and made all other volumes PRIVATE.
- We deleted the start commands for APPC, ASCH, and OMVS from SYS1.PARMLIB(COMMND00). While we used OMVS for other purposes, it was not relevant to the basic batch and TSO measurements, so we did not start these functions. This made IPL faster, and freed fixed storage areas.
- We created a new LOGON procedure that allocated fewer data sets than the logon procedure supplied with the OS/390 system.⁷⁹ The new procedure was placed in SYS1.TSOPROC, and each of the user IDs used with the TSO script was modified to use the new logon procedure. The new LOGON procedure allocates the basic ISPF panels and messages, and seven temporary data sets for ISPF. These are allocated to SYSDA, causing them to be assigned to VIO in our system.

Initial Results

Please understand that these are informal measurements that are unlikely to apply to all systems. The intention was to provide measurements for tuning comparisons, rather than to provide absolute capacity measurements. Unless otherwise noted, the following timings were based on a PC Server 500 S/390.

We ran the batch stream with one, two, and three initiators. The results for a non-RAID system were:

Initiators	Elapsed Time (min:sec)
1	5:30
2	4:51
3	4:40

I/O counts to our OS/390 volumes were:

address	volser	I/O count	seen by P/390
120	MVSV5R	1114	Product libraries
122	SCPMV5	5166	Catalog, Spool, Paging
130	TSOPAK	91	User data sets
132	WORK01	131	Scratch volume

⁷⁹ The LOGON procedure provided with the OS/390 system was designed for maximum power and function for the TSO user. It allocates a large number of system data sets that might be used with various ISPF functions, such as RACF, HCD, and so forth. We felt a smaller logon procedure would be more appropriate for typical users.

Analysis

Bandwidth to DASD is the primary bottleneck for the batch jobs. The batch jobs used standard procedures for compilations and assemblies. For other functions, the JCL specified SYSDA for all work data sets, and VIO was used for these data sets. SDSF showed almost no paging during the batch runs. The only other active function during this time was a single TSO user (the author) using SDSF to monitor activity.

We did not try more than three initiators; we expect throughput would degrade, due to disk (and AWSCKD track buffer) interference. Based on these and other measurements, we believe that two busy initiators is optimum for P/390 OS/390. More initiators might be started if they are usually waiting on seldom-used job classes. Substantial TSO usage (more than 10 active users, for example) may predicate switching to one busy initiator.

The P/390 I/O counter showed very little I/O to our scratch disk volume.⁸⁰ Almost all work datasets were going to VIO. This was good for P/390 OS/390 tuning, since it used the strongest part of the system (real memory) instead of the weakest (disk I/O). Some of the batch jobs used permanent disk data sets for input, but the large block sizes of these data sets caused only 91 I/Os to TSOPAK (where all the permanent data files were stored). This relatively low real I/O rate may not be representative of many environments, and our timing results should be viewed accordingly.

The P/390 activity indicator showed near 100% S/390 CPU usage for large parts of the elapsed time, although there were shorter periods with 20-30% CPU utilization. The OS/2 pulse indicator, which should not be regarded as an overly accurate measurement, indicated close to 100% OS/2 CPU utilization for approximately one third of the elapsed time, high CPU utilization (more than 75%) for another third, and lower (say 30%) usage for another third of the elapsed time. The SCSI indicator light showed fairly heavy utilization during the whole job stream.

TSO Results

We ran the TSO scripts with different numbers of users, and had the following results with a non-RAID system:

Number Users in Parallel	Elapsed Time Seconds	Average-response-time-(seconds)		
		Trivial	Standard	Major
1	390	.19	.82	2.28
5	416	.50	1.35	3.02
10	444	.54	1.64	3.95
15	490	.62	2.02	5.35
20	519	.70	2.92	6.71
22	556	.78	3.46	6.8

These figures are the averages for each of the simulated parallel TSO sessions. The sessions were not started at exactly the same instant; for twenty sessions,

⁸⁰ All volumes were mounted as *private* except WORK01, which was mounted as *storage*.

the startup period was spread over about 45 seconds. Multiple trials produced similar results, although the exact timings varied due to the random elements introduced in the specified think times between TSO commands. Most individual times were within 20% of the times shown; that is, individual deviations tended to be small. (We stopped at 22 users due to limitations of our test configuration, and not due to any particular limitation of P/390 OS/390.)

Scrolling within ISPF or clearing the screen were considered trivial transactions, and the average response times for such operations are shown in the trivial column. LISTALC, LISTDS, small SUBMITs, starting functions within ISPF or SDSF, and similar operations are considered standard commands and are shown in the corresponding column. Starting SDSF or SUBMITing a large job are considered major operations.

Logon time and the starting time for ISPF are not included in the response time figures, but are included in the elapsed times. Logons and ISPF startups took place, for some sessions, while other sessions were doing other functions; logon and starting ISPF were considered to be part of normal terminal functions.⁸¹

The two batch jobs submitted by each script session were always completed by the time the last session terminated. The overhead of running the batch jobs influenced the TSO sessions, but we did not notice any particular bottlenecks or stress points.

Analysis

The response times shown include LAN, emulator, HLLAPI, and REXX interpreter times. We made no attempt to analyze the overhead of these functions. Each TSO user submitted two batch jobs: one trivial, and one a small COBOL compilation. Other than these, no batch was running during the measurement periods.

Again, note that the trivial response times included the response time for simply clearing the TSO screen. We thought this is realistic since the terminal is disabled for a time, but not everyone would agree with this measurement process. We did not make separate measurements excluding screen clearing response times, but we guessed that such changes might increase the reported trivial response time by about 20%.

SDSF (using the DA subfunction) showed almost no paging during the TSO sessions. This was an unexpected finding, and reduced the tuning options we could try. (If there is no paging, there is little point in tuning the paging subsystem.)

S/390 CPU utilization varied greatly. While sessions were in logon phases, CPU utilization (and OS/2 utilization as seen by the pulse function) were very high. On the other hand, when almost all sessions were in ISPF edit or browse modes, with scrolling being the major activity, S/390 and OS/2 utilization were quite low: in the 20 - 25% range for 15 users.

⁸¹ This is stressed because some TSO evaluation scripts start measurements only after all sessions are logged onto the system.

Again, there was little I/O to our scratch and TSOPAK volumes. For 15 TSO users, with the script described above, the I/O activity reported by P/390 Manual Operations I/O counts⁸² was:

address	volser	I/O count	seen by P/390
120	MVSV5R	2872	Product libraries
122	SCPMV5	6550	Catalog, Spool, Paging
130	TSOPAK	1293	User data sets
132	WORK01	360	Scratch volume

These numbers reflected the initial system changes described above, but no other tuning. Based on this environment and the moderate TSO workload we used, we estimated that 15 active TSO developers would be a reasonable maximum for good response time with our configuration. This number includes ample headroom for additional batch jobs, or for other types of functions, and would be considered a conservative estimate.

To the extent that disk access was a bottleneck, clearly the SCPMV5 volume needed work. Three key performance functions were on this single volume: the catalog, spool (checkpoint and data), and paging (all paging data sets). While using the system we noticed, several times, that more spool space would be convenient, so we elected to create a volume just for spooling functions. For our batch jobs, the I/O counts were now:

address	volser	I/O count	seen by P/390
120	MVSV5R	1262	Product libraries
122	SCPMV5	4202	Catalog, Paging
130	TSOPAK	95	User data sets
132	WORK01	151	Scratch volume
135	SPOOL1	1741	Spool

And for 15 TSO users, the I/O counts were now:

address	volser	I/O count	seen by P/390
120	MVSV5R	3338	Product libraries
122	SCPMV5	4059	Catalog, Paging
130	TSOPAK	1376	User data sets
132	WORK01	360	Scratch volume
135	SPOOL1	2886	Spool

In both cases (TSO and batch), there was no significant improvement in elapsed times or response times. This was a surprising result.⁸³

Our conclusion was that for modest batch and TSO usage there is little point in attempting to tune OS/390 volume and data set placement. This is quite contrary to normal OS/390 experience, and highlights a unique aspect of the P/390 OS/390 environment.

⁸² Note that these numbers do not include I/O to VIO data sets, since the P/390 interface does not see these. If VIO causes a paging operation, this is counted, of course.

⁸³ At this time, we had done no OS/2 tuning and were using the default cache parameters.

Paging Capacity

We wrote a small program that “touched” a variable amount of main storage. By using a parameter on its EXEC statement, we could specify the number of megabytes (from 1 to 110) that it would use. It would sequentially store one word in the first word of each page of its large storage area. After doing this, it would WAIT for one second, and then loop through the storage area again. (The number of loops was another EXEC parameter). The net effect was a program with a very light computing load, but a large, uniform storage (paging) load. In a paging sense, it was one of the worst possible programs. The program was named PAGER.

With OS/390 running⁸⁴ and one TSO user logged onto the system, we could run PAGER using 100MB of main storage without any significant paging.⁸⁵

When told to use 110 MB, PAGER caused extensive paging -- although it did complete its loops, using minutes per loop. The purpose of this exercise was to observe the maximum sustained paging rate. We found this rate varied greatly, for reasons that we did not understand. The large, sequential paging operations should have negated the effectiveness of caches, and we did not observe (using SDSF) any swapouts due to SRM.

We observed paging rates generally between:

Paging rates (non-RAID, one paging volume) 15 - 30+ pages/second

with extremes between 6 and 65 pages/second. The paging rate was lowest (averaging about 10 pages/second) during the first loop, when page slots were being assigned to new pages. During this time OS/2 utilization was above 80%, S/390 utilization was below 20% (with the normal background rumblings of OS/390), and the SCSI subsystem was very busy.

In subsequent loops, S/390 utilization remained below 20%, OS/2 utilization was steady at 55-60%, and the SCSI subsystem remained very busy. The average rate was in the 25 to 30 pages/second range, but with the wide variations we mentioned, mostly on the higher side. In all cases, OS/390 remained responsive in CPU dispatching, but was -- as might be expected -- rather slow for any other disk accesses.

The specific storage size (110 MB) that triggered major paging is not significant; it will vary depending on how much real storage is available (128 MB in our case), OS/390 customization, and other subsystems that are running. What is significant is the paging rate that could be sustained through CKD emulation.

In proportion to its CPU and I/O capacity, the P/390 subsystem has more than ample storage. We expect that paging rates would be quite low for most applications and environments well suited to P/390 OS/390. Nevertheless, we confirmed that reasonable paging capacity is present and that it works as it should with any OS/390 system. Being, in effect, a single channel system, paging I/O directly subtracts from I/O capacity available for applications. For this reason, if no other, users should adjust workloads to keep paging at a minimum.

⁸⁴ Using the same Developer's Association CD-ROM system that was the basis for all of this document. In this case, we ran without OOS/390, ASCN, or APPN started.

⁸⁵ There was considerable paging when the program first started, probably due to initial commitment of paging space. Once this subsided, PAGER went through its loops with no additional paging, using only slightly more than one second per loop.

TSO Logon Procedure

We used the following TSO logon procedure for our performance and capacity investigations. It is a stripped-down version of the full-function logon procedure that is included (in SYS1.TSOPROC) in the Developers' Association CD-ROM. Working with this system, we copied member IBMPRODS in SYS1.LOCAL.IPSFPNLS into SYS1.SISPPENU.⁸⁶

These changes reduced the elapsed logon time, for a single user on an idle system, from 14 seconds to 5 seconds in MVS 5.2.2 on a P/390 system.

Many of the ISPF menu options, such as RACF and HCD, will not work when using the logon procedure shown here. These functions require data sets that we removed from the logon procedure, in order to make logon faster.

This is certainly not a minimum logon procedure, since it preallocates basic ISPF data sets. Nor does it allocate the minimum number of ISPF data sets. However, we think it is typical (especially in the system load it presents during logon processing) of reasonable logon procedures.

```
//ISPFPROC EXEC PGM=IKJEFT01,REGION=OM,DYNAMNBR=50,
//          PARM='%ISPFCL'
//SYSUADS DD DISP=SHR,DSN=SYS1.UADS
//SYSLBC DD DISP=SHR,DSN=SYS1.BROADCAST
//SYSPROC DD DISP=SHR,DSN=SYS1.SISPCLIB
//          DD DISP=SHR,DSN=SYS1.CLIST
//SYSHELP DD DISP=SHR,DSN=SYS1.HELP
//          DD DISP=SHR,DSN=SYS1.SISPHELP
//ISPMLIB DD DISP=SHR,DSN=SYS1.SISPMENU
//          DD DISP=SHR,DSN=SYS1.DFQMLIB
//ISPPLIB DD DISP=SHR,DSN=SYS1.SISPPENU
//          DD DISP=SHR,DSN=SYS1.SISFPLIB
//ISPSLIB DD DISP=SHR,DSN=SYS1.SISPSENU
//ISPTLIB DD DISP=SHR,DSN=SYS1.SISPTENU
//          DD DISP=SHR,DSN=SYS1.SISFTLIB
//ISPLT0 DD DISP=NEW,UNIT=SYSDA,SPACE=(CYL,(1,1)),
//          DCB=(LRECL=80,BLKSIZE=800,RECFM=FB)
//ISPLT1 DD DISP=NEW,UNIT=SYSDA,SPACE=(CYL,(1,1)),
//          DCB=(LRECL=80,BLKSIZE=800,RECFM=FB)
//ISPLT2 DD DISP=NEW,UNIT=SYSDA,SPACE=(CYL,(1,1)),
//          DCB=(LRECL=80,BLKSIZE=800,RECFM=FB)
//ISPWRK1 DD DISP=NEW,UNIT=SYSDA,SPACE=(CYL,(1,1)),
//          DCB=(LRECL=121,BLKSIZE=1210,RECFM=FBA)
//ISPWRK2 DD DISP=NEW,UNIT=SYSDA,SPACE=(CYL,(1,1)),
//          DCB=(LRECL=121,BLKSIZE=1210,RECFM=FBA)
//ISPLST1 DD DISP=NEW,UNIT=SYSDA,SPACE=(CYL,(1,1)),
//          DCB=(LRECL=121,BLKSIZE=1210,RECFM=FBA)
//ISPLST2 DD DISP=NEW,UNIT=SYSDA,SPACE=(CYL,(1,1)),
//          DCB=(LRECL=121,BLKSIZE=1210,RECFM=FBA)
//SDSFMENU DD DSN=SYS1.SISFPLIB,DISP=SHR
//SYSPRINT DD TERM=TS
//SYSIN DD TERM=TS
//SYSUDUMP DD SYSOUT=Z
//SYSABEND DD DUMMY
//SDSFDUMP DD DUMMY
```

⁸⁶ This member contained an extension of the primary ISPF menu. By copying it to SYS1.SISPPENU (the primary library of ISPF panels), we could remove SYS1.LOCAL.IPSFPNLS from the logon procedure.

Tape Performance (P/390)

We did a few tape performance measurements, to obtain a general idea of the timing to be expected from tape drives. We worked with:

1. The 4mm SCSI-attached tape drive that is standard with all P/390 and R/390 systems.
2. An older IBM 3480 tape controller and drives attached, via the S/370 Channel Emulator/A, to the P/390.
3. The same IBM 3480 drives attached to a medium mainframe.
4. A SCSI-attached 3490-type drive attached through the SCSI3480 device manager.

In the first case we simply read, using IEBGENER, a fairly long tape. (We assigned SYSUT2 to DUMMY to discard the data.) The tape contained 160 MB, in 16 KB blocks. The indicated times (in minutes:seconds) are from the removal of the mount message on the MVS console until the beginning of the tape rewind.

4mm tape on P/390	6:45	OS/390 about 20% busy
SCSI 3490 on P/390	5:15	(Fujitsu M2488C)
3480 on P/390	5:30	OS/390 about 30% busy; P/390 70-75%
3480 on mainframe	1:42	(runs at full device speed)
AWSTAPE	3:35	(Writing the file took 48:34!)

Processing with the same tape drives (the 3480s) took about three times longer on the P/390 than on a mainframe. This is an indication of the different channel speeds of the systems. It also indicates that attaching faster tape drives to the channel adapter would not improve performance. AWSTAPE (using Server files to emulate a tape drive) is quite fast when reading data, but very slow when writing data.

Working with a more complex tape format (157 MB, 14 K records of various sizes, 118 tape marks), and with a program that uses EXCP (and BLP) to copy data from disk to tape, we had these measurements:

	mainframe	P/390 3480	P/390 4mm
begin writing	0:0	0:0	0:0
rewind output	4:15	10:50	
begin readback check	4:42	11:30	
begin rewind	8:34		
finish unload	9:18	20:00	32:00
simple tape scan (w/o RUN)	2:30	7:04	9:15

This again indicates that a P/390 channel-attached drive is about three times slower than the same drive on a mainframe. It also indicates that the 4mm is a little slower (for simple reads) but considerably slower when handling tape marks.

5.15 tn3270 Emulators

The generic name *tn3270* is often used for versions of telnet that use 3270 protocols. These apply to three situations for OS/390 with the P/390 and R/390 products:

1. The LCS3172 device manager emulates an IBM 3172 TCP/IP gateway into OS/390. That is, the OS/390 TCP/IP product is being used rather than the Server TCP/IP function. OS/390 TCP/IP, with proper parameters for itself and

VTAM, will accept tn3270 connections to OS/390 for normal OS/390 use (TSO, CICS, and so forth). The tn3270 port address will normally be 23, which is the standard address for telnet.

2. The AWS3274 device manager for the R/390 accepts tn3270 connections (via AIX TCP/IP). These connections appear to OS/390 as channel-attached coax 3270s. The OS/390 master console is an example of one of these connections. The port address is 3046, which is intentionally nonstandard.
3. The LAN3274 device manager for the P/390 accepts tn3270 connections (via OS/2 TCP/IP), and these appear to OS/390 as channel-attached coax 3270s. The default port address for this is 7490, which is also nonstandard. (It can be changed by editing the IPL.CMD file.)

Unfortunately, not all *tn3270* implementations offer the same functionality. The most basic implementation emulates a simple 3277-2 type terminal, with a 24x80 display and no extended attribute functions. Older UNIX tn3270s typically implement only this level of function. Using one of these, you can connect and log onto OS/390 via R/390 AWS3274, but you may not be able to use ISPF.

The default VTAM parameters (in the mode table settings) for the CD-ROM OS/390 systems⁸⁷ indicate that applications can query a 3270 terminal to determine the maximum screen size (frequently defined as 32x80) and the availability of other functions. ISPF, for example, does this when it is started. Minimal tn3270 implementations will not recognize the query command (or the alternate-erase command that is often sent at the same time) and the end result is a closed tn3270 session and an IOS000 error message on the OS/390 console.

There are currently two solutions for this problem:

1. Use more advanced tn3270 products that support extended attributes and multiple screen sizes.
2. Change the VTAM mode tables to indicate that only basic 3270 terminals are present.

Many OS/390 users will reject the second option, primarily because they want to use 32x80 (or larger) emulated 3270 terminals. We did not investigate exactly what changes in VTAM parameters would be required to avoid the problem.

We found that the following emulators worked correctly for extended mode sessions through the R/390 AWS3274. In some cases, a parameter such as “-ext” must be specified when the emulator is started.

- PC/3270. There are IBM PC/3270 products for both OS/2 and Microsoft Windows. These products have a number of communications protocols, including TCP/IP.
- hcon. This is an AIX product providing full-function 3270 emulation with a number of supported connection protocols.
- xant. This is an unsupported module that is included with the R/390 support program diskettes. The default ipl390 script starts two xant sessions for the OS/390 console and a TSO session.
- X3270. This is another supported AIX product.

⁸⁷ These same comments apply to the most common situation in other environments, as well. There is no special problem with the CD-ROM packages.

The tn3270 module that is standard with AIX did not (at the time this was written) work with R/390 AWS3274 for ISPF usage.

5.15.1 xant 3270 Emulator

The xant program is included with the R/390 program support diskettes. It is a minimal tn3270 (3270 emulator) program. *It is not a supported product.* You might want to try using it if you have no better tn3270 program (such as hcon or X3270). Early R/390 users have used xant extensively, without problems, for the MVS console and TSO usage. Nevertheless, there is no support if you have problems with it. This section provides a brief overview of xant.

The program contains over 50 command-line options; we will describe only a few of these. The normal starting command for use with the R/390 might be one of these:

```
xant -port 3046 `hostname`
xant -port 3046 9.12.2.70
xant -port 3046 -bracket -rows 32 -sk 9.12.2.70
xant -port 3046 -geom+300+500 9.12.2.70
```

The xant program is used under X windows. You would issue one of these commands from the command line in an aixterm under X windows. The minimum parameters are the port address and the IP address. If you do not specify a port address, it will default to port 23 (which is the default telnet port, but not the one that R/390 AWS3274 uses). For the IP address, you can use several forms; two are shown here. (Use your own IP address, of course, and not the one in the examples.)

The -rows nn operand changes the number of rows in the emulated 3270 screen; the minimum is 24. In our use, we specified -rows 32 (to obtain a 3278-3 display); the emulator produced the correct number of rows but VTAM did not sense this correctly and continued using 24 rows. We did not have time to investigate this problem, but suspect it was a VTAM modetable parameter problem.

The -geom+xxxx+yyyy operand specifies the coordinates of the upper left corner of the new window (under X windows). The default appears to be +100+100, which places the emulator window near the top left corner of the X window screen. You can move the window after it is opened, of course.

The -sk operand causes a slight remapping of the keyboard; in particular, it swaps the carriage return and alt/action keys. If you use -sk, the 3270 ENTER key is the keyboard enter key (that is, the large key that is the carriage return key on a typewriter). A few users prefer this, although most ISPF users cannot tolerate it. We do not recommend using the -sk option!

The -bracket option changes the square brackets from hex BA and BB to hex AD and BD. Unfortunately, there is no real standard for square brackets in EBCDIC.

If you click your mouse in the second title line of an xant screen, you will obtain a small menu. One option is "zoom"; this makes the font larger, and is helpful if you enlarge the emulator display window on your screen.

Important, standard keyboard assignments are listed here:

3270 function	key
clear	Pause or Escape
enter	Action (unless you used the -sk option)
PA1	Ctrl F1, or Prnt Screen, or Ctrl Z
PA2	Ctrl F2, or Scroll Lock
PA3	Ctrl F3
Sysreq	Alt Prnt Screen, or Ctrl Print Screen
Attn	Ctrl Pause
Reset	Alt Escape, or Ctrl G
Erase EOF	Ctrl End, or Ctrl K
PF 13-24	Shift PF 1-12

There are many more parameters and controls for xant than listed here, and you may see more extensive or specialized descriptions elsewhere.

5.15.2 hcon 3270 Emulator

The IBM 3270 Host Connection Program (hcon) is a high-end 3270 emulation program. It supports a number of connection protocols, and supports a number of end-users connected to hcon through TCP/IP. It is a fully-supported IBM program product, with good documentation. Normal installation is from CD-ROM, using the normal smit installation process.

Once hcon is installed, you must do several administrative steps:

1. Add hcon users
2. Configure hcon sessions
3. Update /etc/services

You must login as root to administer hcon. The steps are:

```
smit
  Communications Applications and Services
    AIX 3270 Host Connection Program (HCON)
      HCON User Functions                (define sessions)
      HCON Administration Functions      (define users)
      HCON Control
```

The HCON panels are easy to use and are not described here. You must add users (using HCON Administration Functions); these are simply the AIX user IDs of those permitted to use hcon. You must configure at least one session for each user. The configuration parameters include a session name (such as "A"), the protocol (use TCP/IP), the IP address (the address of your R/390 AIX LAN adapter), and so forth. The target port number is not defined here.

Another required step is unique to R/390 usage (via AWS3274) and is not documented by hcon. You must add a unique R/390 port address to TCP/IP services for hcon:

```
smit
  Communications Applications and Services
    TCP/IP
      Further Configuration
        Client Network Services
          Add a Service
            Official Internet SERVICE NAME = hcon
            transport protocol = tcp
            Socket PORT number = 3046
```

To start hcon, you must be in an aixterm in X windows, and enlarge this window to be large enough for the 3270 emulated screen. Then enter the command “e789 a,” where “a” is a session name you created when you did your hcon configurations. It will use your existing window for the emulator session, rather than create a new window (such as xant does). To end the emulator session, enter Cntl-D twice. The emulated keyboard layout is described in the hcon manual.

The standard R/390 ipI390 script checks whether hcon is present. If it is present, the script starts sessions a and b. If you have installed hcon, you should create sessions a and b, with the appropriate TCP/IP address and port number.

5.16 AWSICA and BSC NJE

The P/390 support programs include a device manager that emulates an integrated communications adapter (ICA). An ICA is supported by VSE and VM, and is a common way to connect SDLC lines to smaller mainframes. AWSICA uses PC multiprotocol adapters or WAC (wide area communications) adapters to emulate ICA lines.

OS/390 VTAM does not support an ICA, and the AWSICA device manager is normally not used with OS/390 systems. However there are a few instances when the AWSICA device manager might be useful. OS/390 does support individual communication lines (although these are seldom used today), not connected through VTAM. These were the lines provided by IBM 2701/2703 control units and later provided by the Emulator Program (EP) on 37x5 control units. Each line is generated as an individual S/390 channel address and has its own UCB.

We used one of these lines to connect two P/390 systems through NJE, and this configuration worked quite well. The process included:

- Define a line to OS/390. This requires the use of HCD. We defined a single line at address 038, with UNIT type BSC1. This is a point-to-point bisynchronous connection using leased lines. (We defined this line on both of our P/390 systems.) We did not define any optional features for the line.
- Alter JES2PARMS to include the following:

```
NJEDEF JRNUM=1,JTNUM=1,LINENUM=2,NODENUM=10,OWNNODE=2,
        SRNUM=1,STNUM=1
LINE2  TRANSPAR=YES,UNIT=038
NODE1  NAME=P390L,LINE2
NODE2  NAME=P390R
TPDEF  BELOWBUF=(SIZE=520)
PCEDEF CNVTNUM=10
BUFDEF EXTBUF=(LIMIT=120,WARN=70)
```

In this example, the OWNNODE parameter would be “1” on the “other” system. There are many ways to define JES2 parameters, and your environment might differ from that shown here.

- We needed to reIPL to use the new MVS address. We also started JES2 with a COLD start because we had made other JES2PARM changes as well.
- When JES2 is up, we used \$SLINE commands to start the lines, and were able to route jobs and SYSOUT data between systems. (We used ROUTE parameters in JCL to control this.)

We used WAC adapters, connected through a 9600bps modem eliminator. Note that BSC requires *synchronous* modems, not *asynchronous* modems that are commonly used with PCs. Some modems offer both sync and async modes, but default to asynchronous operation.

In a similar way, BSC lines (emulated through AWSICA) could be used for BSC RJE stations, or for other application programs that support direct BSC use. We assume BTAM would work correctly with these lines, but did not try a direct BTAM application.

5.17 Internet Address

The P/390 (and R/390) developers use an Internet ftp function for distributing informal updates and for receiving traces. The ftp address is:

```
1scftp.pok.ibm.com
```

This address is used by other S/390 products, and contains many subdirectories. You want to start at subdirectory:

```
/pub/p390
```

Under this you will find several directories and a README file. Do not assume any fix or update should be used unless the comments in README or elsewhere clearly states it is for general use. (Trial fixes are present, and you probably do not want to install one of these unless it was intended for you.)

There is a directory named *incoming* that you can use to send traces to the P/390 developers. Do not send anything through this directory unless it is requested by the developers.

5.18 LAN Connection

You may have ordered a token-ring adapter with your system. This may be mounted in any slot. Once you enable the token ring in OS/2 (define drivers in CONFIG.SYS and start P390 device managers that use it), *do not disconnect it from a valid token ring*. If you must disconnect it, redefine your P/390 subsystem configuration and your OS/2 CONFIG.SYS to exclude it, or obtain a "terminator" for the token-ring adapter connector. (This is *not* a SCSI terminator.) If the token ring is disconnected, various drivers and device managers will time out when trying to use it. The time-out period is relatively long, and OS/390 will appear to hang (with solid "pink" showing in the activity indicator).

Chapter 6. Frequent Questions

This chapter lists many of the questions often asked by potential customers. Abbreviated answers are provided. More extensive material appears in earlier chapters and in the other relevant publications, listed at the beginning of this document.

6.1 PC Server System/390 Systems

Q: Can I add a very large OS/2 memory and use almost all of it as the HPFS cache? This should make the OS/390 disks appear to be much faster.

A: The standard HPFS cache is limited to 2MB. We have not made comparative studies of normal and very large cache systems for this system. Other studies (on basic OS/2 and Microsoft Windows) indicate that a very large cache may not be effective, due to the overhead of cache management.

Q. Can the OS/2 side use memory assigned to the S/390 side? Can the S/390 use memory from the OS/2 side?

A: No and no. Each side has its own dedicated memory and this cannot be shared with the other side.

Q: Can I direct certain files to use more cache? For example, could I assign the OS/390 paging datasets to cache?

A: No. There is no way to control HPFS cache at this level of detail.

Q: Can I use IDE disks (non-array, of course), if I install an IDE controller?

A: We did not try this. Except for mechanical concerns, we assume it would work. Normal internal disks *must* be mounted in the "hot swap" trays and plugged into the SCSI backplane. It would be difficult to mount different types of disks, although the space in an unused disk shelf might be adapted to hold IDE drives.

Q: Should I use the array model?

A: In general, yes. The majority of machines used during the final development and testing of the total system were RAID models. If your environment is an important, data-oriented production application, RAID offers advantages for reliability. If your environment is for program development, there may be less advantage in a RAID system. Remember that OS/390 does not see any difference. Only the OS/2 device drivers are aware that RAID is present (or not present, as the case may be). In addition to reliability, RAID offers advantages in space management; a single, very large disk (which is how a RAID array appears to OS/2) provides easier allocation planning. The RAID adapter has a 4MB cache, providing a general performance boost across all uses of the array.

Q: My older, half-high SCSI disk drives will fit in the "hot swap" trays (8-bit version). However, the small addressing cable that came with the tray will not fit my disks. What should I do?

A: Use the disks that are intended for the PC Server 500, if at all possible. You can install up to 17 (2.25GB) units; they fit properly and SCSI addressing is automatic. Viewed over the life of your system, you will probably regret investing in any other types of disks.

However, there may be situations when you are forced to use older disks. We had this situation with our early system, and had the same problem you describe. We ignored the addressing cable and used the normal SCSI jumpers to assign an address to the disk drive. We took care to assign an address that corresponded to the SCSI address for the disk slot we used. (We do not know if this was necessary or not.) Disk slot 6 has SCSI address 5, slot 5 has address 4, and so forth. Slot 1 has address 0. Slot 1 in the top shelf cannot be used because SCSI address 0 is used by the 4mm tape drive. Our older disks were half-high units, and we effectively lost half the disk connections on each shelf of the PC Server 500 (because our older disks blocked the connectors).

Q: Can I use OS/2 virtual disks to emulate OS/390 disks? They would make very fast devices.

A: In theory, yes. You could use virtual disks for scratch volumes, and even for paging and spooling. In practice, it could be awkward. The total startup process to build appropriate virtual disks would be slow. A disk must be allocated and then initialized (with ICKDSF) to build a valid VTOC. You may need to build VSAM datasets, and connect catalogs. This would need to be done every time the base system is booted. (You would also need to delete the VSAM components before stopping OS/390, or face a messy cleanup situation when you restart the system.) A clever owner could build the necessary disk image on an offline emulated 3380 volume and copy (using an OS/2 copy function) the whole disk image to a virtual disk before OS/390 is started. (Note that an OS/2 virtual disk does not exist in OS/2 pageable memory; it occupies dedicated, non-pageable memory.)

Q: Can I use external SCSI disks?

A: Yes, there is a 68-pin SCSI connector on the back of the SCSI adapters for use with external devices. If you have an array SCSI adapter, any external disks must be defined as part of an array, and are subject to the rules for forming an array. The IBM 3516 external SCSI enclosure is a practical base for external SCSI disks.

Q: Can a LAN terminal (via the LAN3274 device manager) be an MCS console?

A: Yes.

Q: Can I place an emulated 3380 volume on a LAN file server, and have several OS/390 systems share that volume? This could provide shared DASD with other P/390 OS/390 systems.

A: You can place a volume on a LAN file server, but you should verify that performance is acceptable before considering using it as a production environment. Several P/390 OS/390 systems could share the volume, but the shared DASD protection mechanisms (the reserve/release device commands, in particular) are not available. Over time, the VTOC would almost certainly become corrupted, unless the volume is used in read-only mode by all but one of the sharing OS/390 systems. Remember that the CKD device manager

ALWAYS reads and writes full track blocks. LAN bandwidth is likely to be much less than your local disk bandwidth.

Q: Will an additional RAID adapter make the system faster (as opposed to adding more disks to one RAID adapter)?

A: Yes, although we do not have precise measurements. Early indications were that an additional RAID adapter (with a suitable complement of disks) may increase general performance by up to 20%. (It was not clear how much of the increase was due to additional disks and how much was due to the additional adapter.)

Q: Can my emulated 3380 drives be on a compressed OS/2 disk?

A: We assume this would work, but did not try it. We strongly recommend against it, for performance reasons.

Q: I looked at the IPL.CMD file. It starts many device managers I do not need. Should I modify this file?

A. You can, but we recommend you do not. A device manager will detect that it is not needed and delete itself. Using the standard IPL.CMD file only adds a second or so to startup time (by starting unneeded device managers).

6.2 RISC/6000 S/390 Systems

Q: For a P/390 subsystem, does a RISC/6000 Server make a faster system?

A: Slightly, because it provides faster device emulation. The P/390 adapter is the same (and the same speed) in PC Server and RISC/6000 systems. The device managers, however, execute as native code in the server. Other things being equal, a faster server will produce a faster I/O subsystem for S/390 operation.

Q: Do OS/390 users (TSO, CICS) need to have an AIX user ID?

A: No. However, a user ID may be needed in order to access a 3270 emulator. For example, if hcon or xant (tn3270 emulators for AIX) is used, the user must login to AIX to access xant or hcon. If 3270 access is through a PC and TCP/IP (using PC/3270, for example) then no AIX user ID is needed.

Q: Can I use Unix permission bits to protect OS/390 files?

A: Yes, but the protection only applies to AIX users, not OS/390 users. The AIX user ID used to start and run the P/390 subsystem must have r/w access to all emulated DASD volumes. (This user ID is normally *root*.)

Q: Will the P/390 subsystem use my large RISC/6000 memory to its advantage.

A: In principle, yes -- at least to some extent. The AWSCKD device manager uses AIX disks for emulated DASD volumes, and AIX uses memory as a large cache. We have not measured the effects of varying Server memory.

Q. Can I copy P/390 device managers from a PC Server system to an RISC/6000 system?

A: No. The device managers are executable code on the servers. Code designed for OS/2 interfaces, and compiled for an Intel instruction set, is not usable on an RISC/6000 system.

Q: My AIX system has an 8mm tape drive. Will SCSI3480 support it?

A: Yes, if it is an internal drive. External 8mm drives are sometimes unknown devices. The problem is that different 8mm drives often have different microcode built into them and do not respond to programs in exactly the same way.

6.3 Questions Common to Both Bases

Q: Is this really a multiuser system? How many TSO users can I support? How many CICS users? What is the best way to connect these users to a P/390 OS/390 system?

A: Yes, it is a multiuser system. Numbers of TSO or CICS (or IMS or TCP/IP) users can be connected. How many depends on many variables. There is a brief discussion of P/390 capacity in 5.14, "OS/390 Performance and Capacity (P/390)" on page 140.

Q: This book talks about OS/390. We are just moving to MVS 4.3. Can an older MVS be used? Is there specific support in OS/390 for the P/390?

A: Any OS/390 or MVS can be used. No release contains specific support for the P/390. The only exception is that relatively current versions of ICKDSF must be used when initializing emulated CKD volumes. You need to have the required MVS and program product licenses for the system, of course.

Q: Can the same CD-ROMs (OS/390, VM, VSE) be used with both the P/390 and the R/390?

A: Yes

Q: You have strong warnings about attaching IBM and OEM devices to the S/370 Channel Adapters. Will this situation improve?

A: The P/390 developers are very aware of the constraints imposed in this area, and are actively investigating alternatives.

Q: Can P/390 OS/390 be used without an operator? In lights-out mode?

A. There is little unique about P/390 OS/390 in this situation. Answers for these questions depend on your environment. Do you require tape mounts? Printer operation? Is there a remote MCS console? The more restricted I/O (tape and printers, in particular) may make operatorless use of P/390 OS/390 more common. Note, however, that error messages from the Server (OS/2 or AIX) normally appear only on the server display. This limits the practicality of remote or operatorless operation.

Q: Can startup be completely automatic? From booting the Server through CICS/ESA ready?

A: We are not aware of anyone doing this. We expect that it can be done, at least for a "normal" startup. Some problems must be resolved, such as (for a

PC Server base) timed delays while CM/2 is starting emulator sessions, and a way to reply NOREQ to JES2 and RETRY to the TCAS message that OS/390 often needs. A small EEHLAPPI program would provide one approach.

Q: You refer to only one MCS console in this document. Why?

A: There are no MCS console restrictions unique to the P/390. Since these are usually rather small OS/390 systems, one MCS console is usually sufficient. Using more than one MCS console does introduce a few problems. In order to become active automatically, the console should be present at IPL. If the real connection, through a device manager, is to a TCP/IP or LAN workstation, that connection should be active at IPL. If secondary MCS consoles are not present at IPL, they can be varied online later. If these are (or might be) LAN or TCP/IP connections, you should consider the possible security implications.

Q. Standard OS/390? Are there no special messages for the P/390 environment?

A. Yes, there are unique messages from the P/390 device managers. These messages appear under OS/2 or AIX, not OS/390. They are documented in IBM publication SA22-7227.

Q: Since the S/390 processor is an adapter card, why can I not run it with any Microchannel processor?

A: It might work, but it is not available as a separate adapter card. Notice that we said it *might* work. A P/390 system is complex, with OS/390 (or VM or VSE), OS/2 or AIX, a heavily-loaded disk subsystem, rather complex device emulation routines, and (often) external connections to "real" S/360, S/370, and S/390 devices. Developing and supporting this in a few, well defined environments enables IBM to provide a stable, supportable product. Attempting to build, test, debug, and support the same functions on the full range of all possible Micro Channel systems (and all the potential adapters these might use) is an unreasonably large job.

Q: Why can I not attach "real" mainframe DASD devices, such as my IBM 3880/3380 strings, to a S/370 parallel channel adapter in my P/390 system?

A: Timing and data rates. DASD channel operations are *very* time critical. The existing P/390 channel emulations (on the Server side of the system) cannot be guaranteed to meet mainframe DASD timing and data rate requirements for attachment to mainframe DASD. There are too many variables in the timing factors which would produce unstable and unsatisfactory results.

Q: Does CKD DASD emulation really provide full function? For example, are RPS and channel disconnect/reconnect simulated? Do formatting writes work the same as on a real CKD device?

A: CKD DASD emulation provides the same end results as DASD channel programs on a mainframe. That is, your record is located and read (or written). Some channel functions (such as RPS and multiple disconnects within a channel program) are ignored. Formatting writes are emulated, since this must be done to correctly emulate a real CKD device. The net result is an accurate emulation of CKD DASD *results*. Timing will not be the same as a real CKD device. Used correctly, RPS, for example, affects timing. An emulated device may require more (or fewer, depending of cache and buffer operations in the emulation program and in OS/2) disk revolutions than a real CKD device.

ECKD emulation is automatically used for all emulated 3380, 3390, and 9345 volumes.

Q: Are all channel program functions correctly simulated? What about command and data chaining?

A: As far as we know, all normal channel programs used by current OS/390 operating systems and software products should work. It is possible that exotic channel programs, working with unbuffered (that is, very time-sensitive) devices may not work. (Note: This discussion generally refers to a S/370 Channel Adapter/A connected to older devices (an IBM 2701, for example) using unusual, customer-written channel programs.)

Read backwards is not fully supported. Some of the more exotic diagnostic CCWs (device specific) are not supported. Neither of these limitations appear to affect OS/390 operation. (Most Device recovery is performed by the OS/2 device drivers; OS/390 does not "see" complex failures and does not enter the advanced recovery functions associated with these devices.)

Q: Can I write my own device managers, to emulate other mainframe devices, or support other PC adapters?

A: Yes, but it is not a trivial job. Information is found in two ZIP files contained in the P/390 subsystem diskettes. The files are AWSDEV.ZIP and AWSDMEX.ZIP. You will require the PKUNZIP utility (or something equivalent) to unzip these files. PKUNZIP is included on the diskette. (Another ZIP file, PWSPower.ZIP, is concerned with managing a multiprotocol adapter; this is not relevant to OS/390 usage.) Device managers are written in C (or C++ or PL/I) or Intel assembly language. Examples and descriptions of both methods are provided.

Q: Can I install OS/390 and VM (and/or VSE) on the same system?

A: Yes. You would have different IPL volumes for each operating system, just like on a mainframe. You select an IPL volume through one of the Server-side support programs.

Q: Does it use a simplified version of OS/390? How many support people are needed?

A: No, a normal, full OS/390 is used. The number of support people needed varies widely. The number of people to install and customize OS/390 (with a number of program products, such as DB2, IMS, and CICS) is usually more than the number of people needed to operate such a system once it is in production. It may be reasonable to operate distributed P/390 OS/390 nodes with only a part-time administrator at each location. This person would need to know how to manage routine functions, and know who to call for more advanced problems.

Q: Can P/390 OS/390 be part of an NJE network?

A: Yes.

Q: Can I run VS1, or SVS, or MVT, or CP67, or even PCP?

A. Maybe. Only some of the older DASD is supported. In particular, 2311s and 2314s are not supported. You would need to run in 370 mode, rather than ESA mode, and this is set in the Configurator environmental variables. There is no

IBM support for use of these older operating systems, so you are on your own. The P/390 adapter does not support 2K storage keys, and this may be a limiting factor for VS1.

Q: Where are the performance bottlenecks?

A: As with most systems, performance bottlenecks are very dependent on the type of applications being used. More often than not, the bottleneck is the Server I/O system. Typical OS/390 applications and usage produce heavy I/O, especially for disks. OS/2 programs are used to emulate CKD disks, and this requires substantial processing by the Server system.

Q: Can I install multiple S/390 adapter cards and create an MP system?

A: Multiple P/370 (not 390) adapters have been installed in a single server, but only for specialized demonstrations. This does not produce an MP system because memory and interrupts are not shared among the adapters. At this time, this is not a practical solution for anything, and it is not supported by IBM.

Q: Do I need a large display?

A: It is not *required*, but it would be very nice to have. We recommend a 17-inch (1024x768) display, or larger.

Q: Can I have multiple 4mm or SCSI-attached 3480-type drives.

A: For P/390, you can have a total of four, with one associated with each of the following device managers: SCSI3480, AWS3480, SCSI3420, AWS3420. Only two of these emulate 3480 drives; the other two emulate 3420 drives. (Yes, you can emulate a 3420 when using a 3480-type drive.) The emulation mode controls the format of sense bytes returned to OS/390, and which CCWs are emulated.) The ISITAPE device manager has different limits.

Q. Why 3422 tape drives instead of 3420s? My OS/390 system is already defined with 3420 drives.

A: Your 3420 definitions will work. The 3422 provides more sense information (used by VM, for example, to autoconfigure itself).

Q: Can I use disks not listed in the IBM announcement materials?

A: The S/390 programs (such as OS/390) do not use any disks directly. This is a very important point to understand. The disks are used by Server programs that emulate OS/390-type disks. Any disk that can be used by OS/2 or AIX, in the Server model used, should work.

Q: I want to use my existing OS/390 or MVS instead of some CD-ROM system. What do I need to change?

A: No code modifications are needed. You may need some reconfiguration, however. For example, you may want to rethink your use of expanded storage. (The P/390 can provide expanded storage, but at the expense of primary storage.) You may want to rework your paging and spooling volumes. (Your P/390 OS/390 system will probably have fewer disk volumes available than your mainframe OS/390 system). You may want to prune your private catalog connections (because the corresponding volumes may not exist on your OS/390). You may want to greatly reduce your VTAM definitions -- attempting to start

many nonexistent VTAM connections slows down IPLing. JES definitions need to be reworked -- your P/390 system is unlikely to have printers and NJE connections that are on your mainframe. You should plan your workload for a lower paging rate than your mainframe accommodates. And so forth. (You must have proper licenses for program products used with your P/390 OS/390 system, of course.)

Q: Can I do a complete OS/390 CBIPO installation on this system?

A: Yes. The same concerns apply as would apply on a mainframe. You need a driver system at the right level, you need a tape drive to read the CBIPO distribution tapes, you need disk space for the DLIBs, you may need a *lot* of spool space, and so forth. Some of the CBIPO installation jobs create heavy I/O and processor workloads, so we expect these would take longer on a P/390 OS/390 system. Overall, CBIPO installation time is governed by the human interactions needed (especially to inspect the return codes of various jobs).

Q: Can my P/390 OS/390 be part of a sysplex?

A: No.

Q: How are S/390 addresses assigned to emulated devices? For example, why is the master console address 700 in your examples?

A: We used existing OS/390 generations for some of our examples. Address 700 was the "real" hardware addresses of OS/390 consoles on our mainframe. We assigned the same address in the P/390 configurator because this was the addresses OS/390 would expect. If you are adding new definitions to your OS/390 system, say several new 3380 drives, you can select any addresses you like. The S/390 addresses mean nothing to the Server hardware⁸⁸ other than a link between OS/390 and the proper device manager on the Server side. The P/390 configurator (explained in Chapter 2, "Configurations and the P/390 Configurator" on page 23) defines the links.

Q: You said I could use "small" 3380 volumes. Are there any restrictions? Can I define, say, 100-cylinder 3380 volumes for each of my TSO users?

A: Yes. Of course, you probably want to define a user catalog on each volume, with aliases from the master catalog. You must run ICKDSK on each volume when you create it.

Q: Can I generate an NCP using P/390 OS/390?

A: Yes. However, you can load it (into your 37x5) only if the 37x5 is channel attached, or if an NCP is already loaded and active. If it is attached through a token ring, using the AWSFEP device manager, you need to use a special NCP-load technique.

Q: How do I specify all the hardware path information wanted by MVSCP or HCD?

⁸⁸ An exception is addresses used with the S/370 Channel Emulator/A adapter, in which control unit and device addresses are conveyed to the external I/O device.

A: It appears that you can specify anything you like, because it is never used. (We usually did not specify any hardware path information for HCD.) Your specifications must pass the syntax checks when you assemble MVSCPIN specifications or work with HCD, but the P/390 subsystem does not use this information (and will ignore it if you try to load it). With the P/390 subsystem, there is a single path from each defined address to the associated (emulated) device. If you depend on multiple addresses/paths for a device, you may have a problem.

Q: How long should OS/390 IPL take?

A: A fairly simple IPL (that starts JES2, VTAM, and TSO, but does not have a large VTAM network, does not start APPC or OMVS, and so forth) will probably take at least 10 minutes and may take as much as 20 minutes.

Q: Can I leave all my old VTAM definitions intact?

A: Yes, but it makes starting VTAM slower (just as it does on a mainframe). A few dozen nonexistent nodes defined for VTAM will not cause much delay, but a few thousand such nodes/terminals is a different matter.

Q: How many initiators should I start?

A: We used two initiators. You need to consider your total workload (TSO, CICS, DB2, and so forth) to answer this question. Considering that there is, essentially, a single path (channel) shared by all devices and that multiple emulated disks share single disk access arms, we suggest you do not start too many initiators. (We are considering active initiators, of course. An initiator waiting for a seldom-used job class has little effect on performance.)

Q: I thought 4mm (and other PC tapes) had to be "formatted." Should I use a PC program to format 4mm tapes before using them?

A: No. The P/390 device managers (SCSI3480 and SCSI3420) write variable-length tape blocks and tape marks, just like mainframe tapes. The preformatting is a requirement for many PC programs, and is a software requirement rather than a hardware requirement.

Q: Can I program a 4mm drive just like a real tape drive?

A: Yes, but.... The 4mm drive operation is closer to that of a streaming drive, instead of a fast start/stop drive such as an IBM 3420 unit. Backspacing, or slow/erratic arrival of data blocks will result in slow operation. Read-backward operations are not supported.

Q: Are channel service and (emulated) device service priorities about the same as a mainframe? Is as much I/O operated in parallel as possible?

A: We had no way to measure this. Except for a few special cases, our impression was that I/O was "spread out" roughly equivalent to a mainframe. One special case we noticed several times involved large SVC dumps (to SYS1.DUMP00, for example). These dumps appeared to monopolize the emulated 3380, and perhaps the SCSI channel. This resulted in \$HASP263 WAITING FOR ACCESS TO JES2 CHECKPOINT VOLUME messages (and also \$HASP259 messages). Other than a few seconds delay, there were no harmful side effects.

Q: Is AWS2821 emulation limited to 1403 printers? Can it emulate a 3211 printer?

A: P/390 AWS2821 currently does not emulate FCB and UCS functions. When JES works with 3211 (or newer) printers, it typically performs at least one FCB load. 1403 printers do not have FCBs, and we recommend you use AWS2821 to emulate 1403 printers. R/390 AWS2821 does emulate FCBs, so you can define 3211s (or other more modern impact printers) for it.

Q: How fast is the system? How many MIPS? How does it compare with other OS/390 systems?

A: At the time this was written, IBM had not provided specific performance numbers. During our informal usage, we sensed performance similar to a mid-range 4381. ("Our sense" does not take the place of accurate numbers, of course. More useful information may become available after this is written.) A linear performance comparison (MIPS to MIPS) with a mainframe is not meaningful, due to the very different I/O structures of the P/390 subsystem and the mainframe.

Q: Do the tape device managers (SCSI3480, SCSI3420, ISITAPE, AWSTAPE) emulate the length of a tape?

A: No. The actual tape (4mm, 3480/3490, or disk space available for the AWSTAPE file) determines the "length" of the emulated tape. (As an interesting aside, we noticed that a tape mark written on a 4mm tape consumes approximately 1MB of tape space.)

Q: What about QIC drives?

A: QIC drives are not supported.

Q: How long can the bus and tag cables be, when using a S/370 Channel Emulator/A? How many control units can I string together?

A: We found some documentation indicating that the total cable length from the adapter to the final terminator cannot exceed 200 feet. You must subtract 15 feet for every control unit on the channel. The same documentation indicated a maximum of four control units. HOWEVER, this same documentation assumed the emulator was being used with the AFP printers it was designed to be used with. Other control units, especially if their signals are marginal for use with the emulator, may work better with shorter cables and/or fewer control units on the channel.

Q: Can I chain multiple "real" control units on a single S/370 Channel Emulator/A? Should I use multiple channel emulator adapters? What about address conflicts?

A: Yes, you can chain more than one control unit on the channel emulator, following normal S/390 rules. Please read 4.7.1, "AWSC370 Device Manager" on page 106 carefully. It explains how to change the effective address of units connected through the channel adapter. You can have more than one channel adapter. We suggest using only a single channel adapter unless you are certain you need more than one. IBM supports up to two adapters.

Q: I have a 3480 control unit attached to the channel adapter. Must I include a configuration entry for every tape drive, with each entry pointing to the AWSC370 device manager?

A: Yes. Every *device* must appear in your configuration file. The P/390 configuration program has no concept of a control unit connection.

Q: Can I dynamically change my DEVMAP file? Or my device manager parameters?

A: No. These parameters are picked up when the P/390 subsystem is started. To change the parameters you need to "END P/390" and then start it again. An exception is the small range of changes you can make with the AWSMOUNT command.

Q: I have a S/370 Channel Emulator/A adapter in my system. I sometimes use it to connect to tape drives. Do I need to terminate the adapter connector or the bus and tag connectors when the adapter is not connected to a device?

A: We do not know an "official" answer. The author did exactly the same type of movements you describe. We left our bus/tag cable connected to our 3480 tape drives and simply disconnected the cable at the Server S/390 end. We seemed to have no ill effects from this operation. However, other documentation (*Installation and Application of VSE/ESA on PS/2 Using the Personal/370 Card*, GG24-4274) states that proper termination is required.

Appendix A. AWS2821 Translation Table (P/390)

The default translation table (EBCDIC to ASCII) used by the AWS2821 device manager (P/390 version) is shown here. You can change specific character translations by using the TR= function in the printer control table, as described in 4.5.2, "AWS2821 Device Manager" on page 92.

The table is in hexadecimal. The hex value of the EBCDIC byte is used to find the appropriate character in the table, and that table byte is sent to the PC printer (or file) by the AWS2821 device manager. For example, an input x'C1' (the EBCDIC character A) is translated to x'41' (the ASCII character A). The common ASCII characters associated with some code points are shown below appropriate entries. The translation is what you would expect for common characters, but may not be what you want for non-character EBCDIC bytes.

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
1x	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2x	A3	A6	A5	A4	00	00	00	00	00	00	00	00	00	00	00	00
3x	05	06	07	08	09	0A	0B	0C	0D	0E	01	02	03	04	00	00
4x	20	81	82	83	84	85	86	87	88	89	9B	2E	3C	28	2B	7C
5x	26	8A	8B	8C	8D	8E	8F	A0	A1	A2	21	24	2A	29	3B	AA
6x	2D	2F	93	94	95	96	97	A8	99	9A	7C	2C	25	5F	3E	3F
7x	00	5E	B0	12	11	10	00	E1	EB	00	3A	23	40	27	3D	22
8x	7E	61	62	63	64	65	66	67	68	69	18	19	F3	A9	1C	1A
9x	90	6A	6B	6C	6D	6E	6F	70	71	72	E3	E2	A7	EA	9C	1B
Ax	FD	7E	73	74	75	76	77	78	79	7A	EF	AC	9D	5B	F2	F8
Bx	E0	EE	EC	E6	F9	00	F5	5C	F6	7F	1F	1E	98	5D	F4	B3
Cx	7B	41	42	43	44	45	46	47	48	49	E5	E7	16	E8	17	ED
Dx	7D	4A	4B	4C	4D	4E	4F	50	51	52	9F	13	FC	FB	91	E4
Ex	5C	1D	53	54	55	56	57	58	59	5A	F0	F1	80	E9	92	AE
Fx	30	31	32	33	34	35	36	37	38	39	00	FA	F7	0F	AF	00
	0	1	2	3	4	5	6	7	8	9						

Note that square brackets are at EBCDIC code points AD and BD (translating to ASCII 5B and 5D). Also, note that simple PC printer control characters may be created by EBCDIC bytes in the range 30 - 39. Many different EBCDIC bytes are translated to 00.

Appendix B. Special Notices

This publication is intended to help you to understand the PC Server System/390 and RISC/6000 with S/390 Server On Board, and how they are used with OS/390 (MVS). The information in this publication is not intended as the specification of any programming interfaces that are provided by any of the products discussed. See the PUBLICATIONS Section of the IBM Programming Announcements for the various products discussed for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product in this document is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594, USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

IBM	Operating System/2
AD/Cycle	ACF/VTAM
DFSMS/MVS	MVS/ESA
OS/2	PS/2
RACF	System/370
VSE/ESA	VTAM
S/390	PSF
CICS	CICS/ESA
XGA	NetFinity

Micro Channel	Personal System/2
ServerGuide	HelpCenter
ExpressPrint	OS/390
Advanced Peer-to-Peer Networking	AFP
AIX	APPN
CBIPO	CBPDO
DB2	DFSMS
ECKD	EtherStreamer
ES/9000	ESA/9000
IBMLink	IMS
LANStreamer	NetView
Portmaster	S/390
S/370	System/390
VM/ESA	VSE/ESA
VTAM	

Windows is a trademark of Microsoft Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

IEEE is a trademark of the Institute of Electrical and Electronics Engineers.

Pentium is a trademark of Intel Corporation.

Other trademarks are trademarks of their respective companies.

Appendix C. Related Publications

The following redbooks also address the PC Server System/390:

- *Printing with MVS on the IBM PC Server System/390*, order number SG24-4612
- *Connectivity on a PC Server System/390*, order number SG24-4624
- *VSE/ESA on a PC Server 500 System/390*, order number SG24-4679

This publication replaces *MVS and the IBM PC Server 500 System/390*, order number GG24-2538.

A complete list of International Technical Support Organization publications, with a brief description of each, may be found in:

International Technical Support Organization Bibliography of Redbooks,
GG24-3070

Appendix D. How To Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at URL <http://www.redbooks.ibm.com/redbooks>.

D.1 How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** — to order hardcopies in United States
- **GOPHER link to the Internet** - type GOPHER.WTSCPOK.ITSO.IBM.COM
- **Tools disks**

To get LIST3820s of redbooks, type one of the following commands:

```
TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
```

To get lists of redbooks:

```
TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
```

To register for information on workshops, residencies, and redbooks:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1996
```

For a list of product area specialists in the ITSO:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
```

- **Redbooks Home Page on the World Wide Web**

<http://w3.itso.ibm.com/redbooks/redbooks.html>

- **IBM Direct Publications Catalog on the World Wide Web**

<http://www.elink.ibm.link.ibm.com/pb1/pb1>

IBM employees may obtain LIST3820s of redbooks from this page.

- **ITSO4USA category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

With an Internet E-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an E-mail note to announce@webster.ibm.link.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

D.2 How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** (Do not send credit card information over the Internet) — send orders to:

	IBMMAIL	Internet
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	bookshop at dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU

Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications Publications Customer Support P.O. Box 29554 Raleigh, NC 27626-0570 USA	IBM Publications 144-4th Avenue, S.W. Calgary, Alberta T2P 3N5 Canada	IBM Direct Services Sortemosevej 21 DK-3450 Allerød Denmark
--	--	--

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada (toll free)	1-800-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States)** or **(+1) 415 855 43 29 (Outside USA)** — ask for:

Index # 4421 Abstracts of new redbooks
Index # 4422 IBM redbooks
Index # 4420 Redbooks for last six months

- **Direct Services** - send note to softwareshop@vnet.ibm.com

- **On the World Wide Web**

Redbooks Home Page <http://www.redbooks.ibm.com/redbooks>
IBM Direct Publications Catalog <http://www.elink.ibm.com/pbl/pbl>

- **Internet Listserver**

With an Internet E-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an E-mail note to announce@webster.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank).

D.3 IBM Redbook Order Form

Please send me the following:

Title	Order Number	Quantity

- Please put me on the mailing list for updated versions of the IBM Redbook Catalog.

First name Last name

Company

Address

City Postal code Country

Telephone number Telefax number VAT number

- Invoice to customer number

- Credit card number

Credit card expiration date Card issued to Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature is mandatory for credit card payment.

DO NOT SEND CREDIT CARD INFORMATION OVER THE INTERNET.

Index

Special Characters

/usr/bin/r390 53
/usr/lpp/r390/bin 53
\$HASP259 messages 163
\$HASP263 messages 163

Numerics

0671-4 devices 15
1403 emulation 93
1403 emulator 15
1403 printer, use 132
2314 disks 160
2540 card reader, use 132
2540 punch, DOC file 110
2540 reader, job submission 90
2703 emulation, DOC file 110
3172 control, NetView 16
3172 TCP/IP offload 101
3172s, defining 133
3211 emulation 93, 164
3215 console emulator 15
3270 configurations 30
3270 emulators, TCP/IP 149
3270 system console 16
3270 terminals 16
3270 terminals, defining 132
3270, emulation keyboard 137
3270, session name 118
3280, session configuration 119
3310 devices 15
3330 devices 15
3350 devices 15
3370 devices 15
3375 devices 15
3380 characteristics 6
3380 devices 15
3390 devices 15
3420 tape drive, options 34, 127
3420 tape drives, defining 132
3422 tape drives 161
3480 IDRC compression 129
3480 performance 149
3480 tape drives, options 34, 127
3480, configurator entry 165
3490 tape drives, options 34
3516 enclosure 156
4mm drive, usage 163
4mm tape compression 129
4mm tape drive, options 34
4mm tape drive, usage 46
4mm tape drive, use 35
4mm tape, standard 19

5088 emulation 16
8mm drive, usage 158
8mm tape drives 164
9-track tape drives 35
9332 devices 15
9335 devices 15
9336-10 devices 15
9345 devices 15

A

Activity window, S/390 136
adapter, P/390 159
addresses, S/390, standard 19
addressing, S/390 162
AIX disk requirements 29
AIX, installation 50
ALC command 119
allocation, emulated disks 60
allocation, emulated tapes 41, 61
ARDRSSU program 139
ASCII terminals 16
ASWPROF command 116
AWS2540 device manager 15, 90
AWS2540, job submission 90
AWS2540, R/390 use 138
AWS2703 device manager 16
AWS2703, DOC file 110
AWS2821 device manager 15, 92
AWS3172 device manager (obsolete) 16
AWS3215 device manager 15, 96
AWS3274 (R/390) 76
AWS3274 device manager 16
AWS3274, positioning 31
AWS3420 device manager 15, 86
AWS3480 device manager 15, 86
AWS5080 device manager 16
AWSC370 device manager 15, 106
AWSCFG command 116
AWSCHAN.EXE, use 4
AWSCKD device manager 15
AWSCKD.EXE, use 4
AWSCMLT command 119
AWSERROR log 123
AWSFBA device manager 15
AWSICA command 101
AWSICA device manager 16
AWSICA, use 153
AWSMOUNT command 82, 113
AWSMOUNT, usage 113
AWSOMA device manager 16, 112
AWSPBS device manager 16
AWSPCSRVD 16
AWSPOSDD driver 125

- AWSSTART Command 115
- AWSSTAT command 115
- AWSTAPE device manager 15, 82
- AWSTAPE performance 149
- AWSTAPE tapes, use 129
- AWSTAPE, operation 113
- AWSTAPE, usage 36
- AWSTFA device manager 16
- AWSTRACE list file 125
- AWSTRCSP program 125

B

- backup, system 139
- batch, performance 143
- BLDLIST command 118
- bos.compat files 51
- bos.dlc files 51
- BSC lines for NJE 153
- BSC links 16
- bus and tag, length 164

C

- C370MAP command 119
- capacity and performance 140
- card reader emulator 15
- CBIPO installation 162
- CCW emulation, limits 6
- CD-ROM in OMA format 16
- CD-ROM with OS/390 19
- CD-ROM, installation 48, 54
- CD-ROM, standard 19
- CDE, deleting 57
- CHAN.TRC control file 124
- CHAN370, positioning 34
- channel adapter, S/370 15
- channel adapter, trace 125
- channel emulator 106
- channel emulator, addressing 164
- channel emulator, clear 116
- channel emulator, display 119
- channel emulator, flow 4
- channel emulator, maximums 164
- channel emulator, terminators 165
- channel emulator, trace 119
- channel programs 160
- channel, ESCON 2
- channel, parallel 2
- channel, timing 9
- CKD devices 15
- CLRIO command 116
- CM/2, installation 46
- CM/2, starting 49
- CONFIG_FILE, variable 53
- CONFIG.SYS, changes 48, 139
- CONFIG.SYS, MAXWAIT 135
- configuration program 14

- configuration, I/O 39
- Configuration, P/390 24
- configurator, introduction 36
- console, integrated 2
- control units, maximum 164
- CP67 systems 160
- CPUID, configuration 41
- CTC device manager 16

D

- dasd device manager 15
- DASD requirements, OS/390 29
- DASD, initialization, OS/390 65
- DASD, mainframe 159
- DASD, planning 27
- DASD, shared 2
- DASD, small 162
- data compression, tape 129
- data link control, files 51
- debugging tools 123
- DEV2NAME command 117
- Developers' Association, CD-ROM 19
- device manager, letter 137
- device manager, writing 138
- device managers, copying 157
- device managers, introduction 14
- device managers, list 15
- device managers, writing 160
- DEVMAP, changes 39
- DEVMAP, declaration 116
- DEVMAP, dynamic changes 165
- DEVMAP, example 39
- DEVMAP, initial 116
- DEVMAP, printing 117
- DFDSS, performance 106
- DFDSS, usage 67
- diagnostics, reference diskette 123
- disk compression, OS/2 157
- disk partitions, OS/2 24
- disk trays 28
- disk, partitions 46
- DISKCACHE parameter 135
- diskettes, emulated tapes 83
- disks, addressing 73
- disks, older 155
- disks, planning 27
- DLPI interface, LCS3172 104
- dm2540 device manager 15
- dm2703 device manager 16
- dm3172 device manager 16
- dm3215 device manager 15
- dm3270 device manager 16
- dm34xx device manager 15
- dmckd device manager 15
- dmhuron 15
- dmtape device manager 15
- dosdir, program 51

dosread, program 51
doswrite, program 51
drivers, conflicts 139

E

ECKD emulation 160
Emulated 3380 on LAN 156
emulated 3380, initialization 65
emulated 3390, allocation 60
emulated tape drives 129
emulated tape, operation 113
emulated tapes, allocation 41, 61
END P/390 function 59
END_VSE command 119
error log, AWSERROR 123
ESA instruction set 6
ESCON channel 2
Expanded memory, configuration 41
expanded storage 2
extended instructions 6

F

FAT disks 135
FAT disks, use of 12
FB-512 devices 15
FBA devices 15
FSLA adapters 16
ftp address 154

H

HCD definition 162
hcon 152
hcon, use 58
hot-standby disk 26
how-swap trays 28
HPFS cache 135, 155
HPFS disks, use of 12
HPFS.IFS driver 135

I

I/O configuration 39
I/O traces 124
IBM 3088 addresses, defining 133
IBM 3480/3490 tape drives, defining 133
IBM 3516 enclosure 156
ICA adapters 2
ICKDSF, performance 106
ICKDSF, usage 65
IDE disks 155
IEHINITT, requirement 129
IND\$FILE program 138
initiators, OS/390 163
installation, hardware 43
installation, overview 43

integrated communications adapters 16
Internet address 154
IOCDS functions 2
IPL address, default 41
IPL command 117
IPL P/390 function 59
IPL, MVS 49
IPL, timing 163
IPL.CMD file 157
IPL.CMD, operation 10
ipl390 command 117
ipl390 script 58
IPLing, R/390 57
ISITAPE driver, use 36

J

JES2 output, ASW2821 92
job output, AWS2821 92
job submission, OS/2 90

K

kernel trace 124
keyboard, emulated 3270 137

L

LAN file server 156
LAN job submission 90, 132
LAN3088 device manager 16
LAN3172 device manager 16
LAN3172, positioning 33
LAN3274 device manager 16, 78
LAN3274, positioning 31, 32
LAZY writes, HPFS 136
LCS3172 device manager 16, 101
LCS3172, positioning 33
LCS3172, with R/390 138
lcs3172rx device manager 16
lcs3172tx device manager 16
length, bus and tag 164
local terminal, name 118
LOGON procedure, performance 148
LPARs 2
LTRENAME command 118

M

mainframe DASD 159
manual operations window 123
MAXWAIT, CONFIG.SYS 135
MCS console, LAN 156
MCS console, multiple 159
measurements, performance 140
memory, RISC/6000, usage 157
MGR3172 device manager 16
microcode assists 6

- minidisks, usage 162
- MIPS, performance 164
- MP systems 161
- MSLA adapters 16
- multiprocessors 2
- multiprotocol adapters 18
- multiuser system 158
- MVSCP definition 162
- MVT systems 160

N

- NCP, loading, generation 162
- NetBios 3270 sessions 16
- NetBios terminals 78
- NetID configuration 41
- NetView connection 16
- NJE, usage 160
- Novaback program 139

O

- OMA format CD-ROMs 16
- OMA, optical media attach 112
- operator, need for 158
- OS/2 disk compression 157
- OS/2 disk space 24
- OS/2 workload 23
- OS/2, installation 46
- OS/390 DASD requirements 29
- OS/390 job submission 132
- OS/390 on CD-ROM 19
- OS/390 output 132
- OS/390 volumes 132
- OS/390, installation, CD-ROM 48

P

- P/390 configuration, setup 59
- P/390 OS/390 support programs 14
- P/390 processor card 4
- P/390 subsystem configuration 133
- P/390 support programs, installation 47
- P/390, definition 1
- paging, performance 147
- parallel channel 2
- password, configurator 38
- PATH, for R/390 53
- PC printer 27
- PC Server System/390, base system 10
- performance 161
- performance and capacity 140
- performance, batch 143
- performance, logon 148
- performance, MIPS 164
- performance, paging 147
- performance, TSO 144
- permission bits, AIX 157

- PKUNZIP, usage 65
- Portmaster 18
- Preconfigured System, CD-ROM 19
- printed output 132
- printed output, AWS2821 92
- printer device manager 15
- printer emulator, 1403 15
- printer support 51
- PRIOROTY_DISK_IO parameter 135
- processor load 23
- product positioning 20
- PSW display window 136
- Pulse, display 136

Q

- QIC tape drives 164

R

- R/390 configurations 29
- R/390 tape usage 139
- R/390, definition 1
- R/390, support programs 53
- RACF, security 133
- RAID adapter, additional 157
- RAID array, extending 28
- RAID model system 155
- RAID, hot standby disk 26
- RAID, tuning 135
- RAID, write policy 135
- RDEVMAP command 118
- read backward, tapes 130
- read backwards 160
- RECEIVE command 138
- RECEIVE, with BLDLIST 118
- reference diskette, diagnostics 123
- restore, OS/390 tapes 67
- RISC/6000-390 configuration 29
- RISC/6000-390, described 13
- RISC/6000-591 configuration 29
- RISC/6000, installation 50
- RPS emulation 159

S

- S/370 channel adapter 15
- S/370 Channel Adapter/A 6
- S/370 Channel Emulator/A 18, 27, 106
- S/370 Channel Emulator/A, installation 44, 48
- S/370 Channel Emulator/A, positioning 34
- S/370 mode, configuration 41
- S/390 Activity Window 136
- S/390 addresses, standard 19
- S/390 Developers' Association 19
- S/390 mode, configuration 41
- S/390 Partners in Development 19
- S/390 processor card, chip 4

- SCHIB blocks 8
- SCSI adapter, use 27
- SCSI tape drive, options 34, 127
- SCSI tape drives, use 36
- SCSI tape performance 149
- SCSI-attached 3420 tape 19
- SCSI-attached 3480 tape 19
- SCSI, external connection 28
- SCSI, external disks 156
- SCSI3420 device manager 15, 86
- SCSI3420 driver, use 36
- SCSI3480 device manager 15, 86
- SCSI3480 driver, use 36
- SDCL links 16
- SDLC links 16
- SDLC links, VTAM 16
- SDLC, P/390 26
- sdldcm device manager 16
- security, discussion 133
- security, disks 134
- SEND command 138
- SEND, with BLDLIST 118
- Server disk space requirements 29
- shared DASD 2
- shared memory 8
- SIO instruction 3
- SNA LAN device manager 16
- SSCH instruction 3, 5
- streams interface, LCS3172 104
- support programs, installation 47
- SVGA adapter 26
- SYS1.TSOPROC 148
- SYSOWN command 119
- sysplex 2, 162

T

- tape drive emulator 15
- tape drive performance 149
- tape drive, drivers 139
- tape drives, options 34, 127
- tape drives, SCSI 161
- tape length, emulated 164
- tape, mounting 113
- tape, rewinding 113
- TCP/IP 3172 interface 16
- TCP/IP 3270 sessions 16
- TCP/IP connection 101
- TCP/IP telnet terminals 78
- telnet 3270 149
- telnet clients 76
- terminals, local 16
- threads, device manager 8
- time (TOD) offset, configuration 41
- tn3270 clients 76
- tn3270 emulator 58
- tn3270 emulators 149
- token ring, installation 154

- token-ring adapters 18
- trace functions 7
- trace table 123
- trace tools 123
- trace, channel adapter 125
- trace, output listing 125
- transparent file access, VM 16
- TRCC370 command 119, 125
- TSO, performance 144

V

- virtual disks 156
- VM operating system 160
- VMLOGON command 119
- VS1 systems 160
- VTAM definitions 163

W

- WAC, verification 101
- WAN3172 device manager 16
- WAN3172, positioning 34
- Warp, installation 46
- workload 23

X

- X Tape program 139
- X3270, use 58
- xant 149, 151
- xant, use 58
- XTAPE, product 131



Printed in U.S.A.

SG24-4847-00

