Part (1)
200101294-7
02 MAR 2001

APPLICATION OF AUTOMATION AND PROCEDURES TO ENABLE HIGH-SPEED RECOVERY AND RELOCATION OF COMPUTER WORKLOADS

Background Of The Invention

This invention generally relates to the recovery and relocation of computer workloads. More specifically, the invention relates to methods and systems using high speed, automated procedures for this purpose.

For various reasons, it may be necessary or desirable to recover or reconfigure computer systems or computer workloads. For instance, natural disasters, such as a flood or hurricane, may do substantial damage to a computer system or a computer network. If the system is complex, reconstructing the system may be difficult and time consuming. Attempts have been made to automate or simplify the recovery or reconfiguration process, but these attempts have not passed beyond simple addressing layouts.

Summary Of The Invention

An object of this invention is to enable high-speed recovery and relocation of computer workloads.

Another object of the present invention is to use a definition language for the expression of requirements

This Sheet Has Been Replaced



for computer systems and associated networking and peripherals.

A further object of this invention is to improve productivity for business continuity and recovery services personnel involved in the management and support of customer recoveries and tests in the server environment.

These and other objectives are attained with a method and system for enabling high-speed recovery and automation of computer workloads. The method comprises the steps of expressing requirements for a computer system and associated networking and peripherals, allowing a customer to specify a recovery instruction, processing the recovery instruction at a recovery site, and using a computer to process the recovery instruction to assign resources at the recovery site to reconfigure the computer system.

Preferably, the expressing step includes the step of using a resource definition language to express the requirement of the computer system. The recovery instruction may include process definitions, system control program information, peripheral requirements, networking information, channel extension capabilities and tape vaulting information. Preferably, the computer is used to analyze, map, reconfigure recovery hardware

This Sheet Has Been Replaced



and begin restoration of the customer to a recovery configuration.

Further benefits and advantages of the invention will become apparent from a consideration of the following detailed description, given with reference to the accompanying drawings, which specify and show preferred embodiments of the invention.

Brief Description Of The Drawings

Figure 1 is a flow chart illustrating a first, customer side part of a method embodying the present invention.

Figure 2 shows a second, recovery side part of the method.

Figure 3 illustrates a particular task that is used in the preferred embodiment of this invention.

Figure 4 diagrammatically shows a computer system that may be used to carry out the present invention.

Detailed Description Of The Preferred Embodiments

The present invention, generally, is a method and system to record requirements, analyze system configurations,

This Sheet Has Been Replaced



and map them to hardware and software inventory at a designated recovery location while reducing, or minimizing, the need for human intervention. This process reduces or minimizes the burden of reconfiguring hardware and software at the recovery center, and can be effectively used to reduce a process that formerly took days down to hours.

More specifically, the present invention uses a resource definition language for the expression of requirements for computer systems and associated networking and peripherals. The resource definition language is used by the customer to specify a complete recovery instruction, from process definition, system control program information, peripheral requirements, networking information, channel extension capabilities, and tape vaulting information in a common descriptive system. This specification preferably is analyzed and processed automatically, enabling included programs and processes to automatically assign resource at the recovery site and reconfigure the environment.

This invention allows complete control of specifications (including networking, channel extension, esoteric device names, console addresses and tape inventory control) as well as rapid (within minutes) evaluation of the disaster recovery test event or actual recovery processing.

This Sheet Has Been Replaced



The present invention may be used to analyze, map, reconfigure recovery hardware and begin restoration of a customer to a hot-site recovery configuration in a matter of minutes. In addition, the invention may be linked directly to director switching technology to automate the assignment of device after selection has been made.

In the implementation of the present invention, with reference to Figure 1, as represented at 102 and 104, the customer inputs an expression of resources and logical/procedural recovery instructions. From the customer input data, a customer proposed worksheet 106 is prepared and electronically transmitted 110 to the recovery management site requirement library 112.

At the recovery site, with reference to Figure 2, requirements and instructions for each customer are stored in a library 202, along with requirements definitions and directives 204, and the customer requirements and instructions are analyzed 206. Also, a scan is performed of available inventory and assign assets 210, a list of available assets 212 is made, and assets available for assignment are analyzed 214. Then, as represented at 216, resource and asset characteristics is made, instructions are generated at 220 to reconfigure assets to match requirements, and recovery 222 is performed using reconfigured assets.

This Sheet Has Been Replaced



Figure 3 illustrates a task referred to as the server getfiles task. In this task, a customer prepared definition worksheet 302 is input to a communications server 304, which serves as a firewall. When the task begins work, as represented by step 306, the server directory entries are read at step 310. If no new files are in the work to do queue, the task returns to step 306. If, however, new files are in that work to do queue, the task proceeds to step 312, where a check is made to determine if the work item is from a valid customer. If the work item is not from a valid customer, the administrator is notified at step 314.

If the work item is from a valid customer, the external file is transferred from behind the firewall to a queue area at step 316. Then, at step 320, the destination is determined, a copy is sent to the project library, and an entry is made at the customer requirements repository library at step 322.

As will be understood by those of ordinary skill in the art, the present invention may be carried out on any suitable computer or computer network. Figure 4 illustrates, as an example, a computer system 400 of a type that may be used in the practice of this invention. The computer system includes a series of central processing units 402, which perform the calculations and logic operations required to execute programs. A main

This Sheet Has Been Replaced



memory system 404 and an auxiliary memory system 406 constitute the memory of the computer system.

A bus 410 serves as the main information highway, interconnecting the other components of the computer system. Input/output channels 412 are connected to bus 406, and telecommunications control units 414 are, in turn, connected to I/O channels 412. A group of direct access storage devices 416 are connected to I/O channels 412 via controller 420. Similarly, tape devices 422 and work stations 424 are connected to the I/O channels via controllers 430 and 432 respectively.

For example, the present invention has been successfully reduced to practice using the resource definition language Common Entity Definition Language, which allows for the expression of requirements for the IBM Server/390 system. Appendices A-Q describe this reduction to practice in detail. In particular, Appendices A and B are documentation user and documentation customer manuals, and Appendix C describes the Common Entity Definition Language. Appendices D, E, F and G are the source code for tasks referred to, respectively, as AMFSGT - Server Get Task, BRSRDSCH - Read Scheduler, MJF Autoconfig GEN(VM), and GETIOCP Library Control. Appendices H, I and J are the source code for MJFJFR Preprocessor RDWS, MJFMCM multicriteria match, and MJFSDC summarizer. Appendices K, L and M are the source code

This Sheet Has Been Replaced



for tasks SCRAMIEE - execution shell, SCRAMIEE Profile, and conversion aid MJFAIS analyze IOCP. Appendices N, O, P and Q are the source codes for conversion aids MJFALD - align data filter, MJFALI align input filter, MJFGAID and MGFPAIM.

While it is apparent that the invention herein disclosed is well calculated to fulfill the objects stated above, it will be appreciated that numerous modifications and embodiments may be devised by those skilled in the art, and it is intended that the appended claims cover all such modifications and embodiments as fall within the true spirit and scope of the present invention.

This Sheet Has Been Replaced



APPENDIX A

This Sheet Has Been Replaced



INTERNATIONAL BUSINESS MACHINES CORPORATION
GLOBAL SERVICES
BUSINESS CONTINUITY AND RECOVERY SERVICES
SERVER/390 RECOVERY AND CONTINUITY AUTOMATION MANAGER
USER GUIDE AND REFERENCE

Document Number SCRAM-390-UGR

February 15, 2000

Business Continuity and Recovery Services Productivity Team

Business Continuity and Recovery Services
Building 183
800 North Frederick Avenue
Gaithersburg, MD 20879

IBM Confidential until Announce

This Sheet Has Been Replaced



This Sheet Has Been Replaced



Contents

Preface v

Summary of Changes vii

1.0 Introduction 1
1.1 Setting Expectations 1
1.2 Terminology 1

2.0 Server/390 Continuity and Recovery Automation Manager 3
2.1 Process Overview 3
2.2 Why Server/390 Continuity and Recovery Automation Manager is Different 4

3.0 An Overview of the SCRAM Environment 7
3.1 How SCRAM Processing Works - An Overview 7

4.0 Processing Overview 9
4.1 Operational Flow 9

5.0 File and Library Naming Conventions 11
5.1 Directory Structures 11
5.2 File Identifiers 11
5.3 Naming conventions 12
5.3.1 Customer Requirements Directories 12
5.3.2 Customer Requirements Filenames 13
5.3.3 IBM Resource Directories Directories 13
5.3.4 SCRAM Component Directories 13

6.0 SCRAM Commands 15
6.1 SCRAMIEE ACCESS 15
6.2 SCRAMIEE ANALYZE 16
6.3 SCRAMIEE COMPRESS 17
6.3.1 SCRAMIEE DISCARD 17
6.3.2 SCRAMIEE EDIT 17
6.3.3 SCRAMIEE EXPAND 18
6.3.4 SCRAMIEE FILELIST 18
6.3.5 SCRAMIEE GENERATE 19
6.3.6 SCRAMIEE LIBLIST 20
6.3.7 SCRAMIEE MAP 20
6.3.8 SCRAMIEE QCONTRACT 21
6.3.9 SCRAMIEE QSEARCH 22
6.3.10 SCRAMIEE RELEASE 23
6.3.11 SCRAMIEE SUMMARIZE 23

7.0 Messages and Codes by Issuing Component 25
7.1 ACG Automated Configuraiton Generator (VM Directory Builder) 25
7.2 AIS Analyze IOCP Specification 25
7.3 ALI ALIgn IOCP records in canonical format 25
7.4 ALC Analyze Logical Control units 25
7.5 ALD Analyze Logical Devices 25
7.6 GAD Generate Address Definitions 25
7.7 JFR Joined Format Requirement 26

This Sheet Has Been Replaced



7.8 PAM Pathing Assignment Manager 26
7.9 MCM Multi-Criteria Mapping 26
7.10 SDC - Summarize Device Counters 26

This Sheet Has Been Replaced



Preface

Applicability

Objective: *To improve productivity for Business Continuity and Recovery Services personnel involved in the management and support of customer recoveries and tests in the Server/390 environment.*

Scope: *This document describes the requirements for Server/390 processor complex recovery configuration generation and maintenance.*

Compliance: *Compliance with these procedures is mandatory and subject to Audit inspection. Any noncompliance must be documented in a risk acceptance.*

Note:

- Within this document, the reference to IBM is to be understood to mean IBM and wholly owned subsidiaries. This includes references to information classification.
- If there is a conflict between this standard and local laws or industrial relations agreements in a country, the conflict and management's decision on deviations from this standard must be documented.
- If this Standard requires that a process be defined, that process needs must be documented unless stated otherwise. Required documentation may be done at any level within an organization.

Definition: Within this document, the term *Provider of Service* is used to refer to the organization and/or management which is providing information processing services, and is to be considered synonymous with local terms used at various IBM sites, such as:

- Supplier of Service
- Service Provider
- Service Delivery
- Information Systems (I/S) Service

Document Control:

- This document will be reviewed annually and will be re-issued when revisions are necessary and approved. Obsolete copies of this document should be destroyed as soon as practical and shall be the responsibility of the holder.

Change History: (Only the changes introduced in the most recent version will be itemized and marked by revision code.)

- Draft Release Beta 001 - February 15, 2000
- Control requirements for systems exempted from this standard are clarified in the document Preface.

This Sheet Has Been Replaced



This Sheet Has Been Replaced



Summary of Changes

20000215 Changes:

1. First Edition

This Sheet Has Been Replaced



This Sheet Has Been Replaced



1.0 Introduction

System/390 Continuity and Recovery Automation Manager is an aid designed to improve communications between the Business Continuity and Recovery Services Service Delivery Teams and their customers.

In addition, parts of the aid address the automated mapping of customer requirements to available resources, and the assisted generation of recovery configurations for customers.

1.1 Setting Expectations

System/390 Continuity and Recovery Automation Manager (hereafter, SCRAM) is an aid, not a programming system. It is intended to reduce the burden on a skilled systems programmer by providing rapid, auditable communications of customer recovery requirements in a standardized, stylized format.

SCRAM is ***not*** intended to allow the complete automated mapping of customer requirements into recovery configurations. It is designed to process the most common cases, minimize keying effort, and generally produce about conversion rate between 75 to 85 percent of source inputs. A successful use of the tool as specified in this document, will be defined as having converted 75 percent of source requirements into recovery configuration specifications without additional manual intervention in one working day or less.

SCRAM allows all International Business Machines Corporation Global Services, Business Continuity and Recovery Services, Service Delivery North America sites to employ a common tool base without necessarily providing an optimized tool for each specific requirement.

Lastly, SCRAM should be view as a working set of programs that are undergoing rapid evolution to a more comprehensive system.

1.2 Terminology

As used in this document, the term:

System will refer to a singular instance of a System/390 processor complex and its associated peripheral equipment.

Source System will refer to a customer system to be managed or recovered on IBM resources.

Target System will refer to an IBM owned or provided system on which the recovery is to be performed.

Event refers to an actual recovery or a scheduled recovery test exercise

Requirements Definition WorkSheet (RDWS) refers to a specific stylized expression of the components involved in a Source System.

Data Server Refers to a dual-server mechanism designed to transport customer information over the Internet to a secured server outside the IBM firewall, and then to retrieve that data, bring it inside the IBM firewall, and to place the data in the appropriate structures for processing

IOCP Mixed human and machine readable input to the standalone version of the IBM Input Output Control Program

This Sheet Has Been Replaced



Directory, VM Directory A sequence of logical and physical machine characteristics designed to define the resources assigned to a virtual representation of an IBM Server/390™ processor, either in whole or in part.

Sorted Date Format A millenium-compliant date stamp in the form "yyyymmdd" where yyyy is the year value, mm the numeric value of the month, and dd is the numeric value of the day of the month. For the purposes of this document, common era reckoning and the Gregorian calendar are assumed.

This Sheet Has Been Replaced



2.0 Server/390 Continuity and Recovery Automation Manager

2.1 Process Overview

The process of preparing to exercise a contingency management recovery involves a number of discrete steps:

1. Understanding what resources are to be recovered.

   Depending on circumstances, a customer may choose to recover their Server/390 system in whole or in part, depending on what s/he defines as "critical workloads." In addition, a customer contemplating recoveries at other facilities must face the challenge of deciding on levels of service, levels of recovery effort, and a number of other variables.

2. Enumerating and expressing requirements

   Once critical workloads are identified and sized, and once resources required to support these workloads are identified, the customer then confronts the challenge of expressing those requirements to any service provider or providers involved in their recovery exercise. Typically, that exercise has been burdensome, because to date no facility existed to integrate the expression of requirements in a clear, consistent manner. What made matters worse was the fact that customer configurations were not static. The dynamic nature of many facilities today means that integration of recovery management into change management was a necessity, but often difficult to achieve.

3. Scheduling sufficient resources to meet expressed needs.

   Once required resources are enumerated and expressed, a similar (not necessarily mono-morphic) process is required to identify, schedule, and allocate sufficient inventory on behalf of the recovery service provider to handle the recovery or test exercise involved.

4. Understanding the characteristics of the target configuration

   Recovery service providers can rarely provide a perfect one-to-one correspondence of requirements to resources in inventory. The fact is that at least some of the time, resources will have to be substituted based on an understanding of the technical characteristics of the dimension or dimensions along which the substitution is required. To give a concrete example, it is usually permissible to substitute a larger direct access storage device (DASD) for a smaller one, even if it means wasting available space. Likewise, it may be possible to schedule a central electronics complex with fewer but faster processor engines to substitute for a configuration with more and slower individual engines.

5. Mapping requirements to resources

   Once the detailed requirements and the characteristic capabilities of the inventory are thoroughly understood, the process of mapping requirements to resources is conceptually straightforward. In the real world, however, this process is often quite difficult. The sources of trouble in this stage boil down to two principal causes: (a) transcriptive error when transferring source or target descriptions into a matching process (b) process error arising from the sheer volume of individual items to match, usually found as unintended process termination resulting in an incomplete end condition.

   Depending on the customer and the requirement, this may entail understanding and defining both the physical (addressing layout) and logical (VOLSER and allocation) configuration.

6. Validating acceptability of proposed target configuration

   A key step in the overall process is validating the acceptability of the proposed target configuration. Given the number of process delays, incomplete or missing specifications, *et cetera*,

This Sheet Has Been Replaced



this step is often "short circuited." The most desirable outcome would be to have sufficient time and resources for the customer to validate the adequacy of the proposed recovery environment, as planned and mapped to date.

7. Preparing pre-test configuration and setup specifications

   Knowing that the resources are properly mapped and that the proposed resource map satisfies the customer needs, the next logical task is the preparation of information to direct pre-test and setup organizations on the specific switching, configuration requirements, networking connections, and other physical connectivity issues involved in performing the physical layout of the recovery target system. This may involve such chores as preparing switching instructions and network connectivity directions. This prepared information then needs to be communicated to the responsible pre-test and setup groups.

8. Building the recovery environment

   Once the specific configuration directions are available, actual construction of the recovery environment may begin. This is usually performed by pre-test personnel, working in teams with appropriate networking and hardware infrastructure groups.

9. Certifying that the built environment meets specifications

   Another key checkpoint is certifying that the environment as built actually meets the specifications. Subteams working with the event management operation may be required to test accessibility of devices, perform basic system administration and space management tasks, demonstrate physical (if not logical) network connectivity, *etc.*

10. Turning over the environment to the customer

    The final phase of the recovery preparation involves turning over the built environment to the customer so that they can begin the recovery of their business processes.

2.2 Why Server/390 Continuity and Recovery Automation Manager is Different

SCRAM departs from current account management and support practices in several key areas:

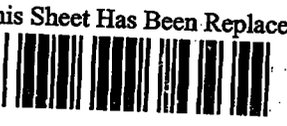
1. SCRAM uses a *standardized* worksheet that expresses both physical (*e.g.,* I/O device definition) and logical (*e.g.,*volume label and recovery technique) requirements in a single definition.
2. SCRAM is designed to promote the rapid transfer of information between the customer and the recovery service provider with a minimum of re-keying and transfer.
3. SCRAM is designed to map resources dynamically, with a minimum of "table maintenance." While some maintenance of control information (principally the floor IOCP definitions) is required, SCRAM seeks to minimize the burden on the pre-tester.
4. SCRAM is designed with operational performance in mind. It is capable of processing configurations from several hundred to several hundred thousand devices in acceptable times - processing rates may exceed 10 requirements definitions/second. Some ultra-large configurations made possible by later releases of FiCon™ could theoretically be processed within a 24-hour window.
5. SCRAM is designed to be architecturally scalable, with support for extended device addressability built into the tool. It can accommodate a theoretical maximum of 16,777,215 devices per source system image. It is designed to accommodate multi-tiered ESCON™ configurations, as well as FiCon™, though full exploitation of these features is not in the current release.
6. SCRAM is modular. It is designed to be implemented in an evolutionary manner. That is, SCRAM is designed to be the first in a series of related productivity aids, all using a common information interchange format.

This Sheet Has Been Replaced



7. SCRAM is designed to facilitate "virtual teaming." By employing a single common repository of both source and target system definitions, SCRAM can facilitate the assignment of labor across recovery centers, allowing for greater synergy between the service delivery teams.
8. SCRAM is designed to be extensible. Architectural provisions have been made to interface with other key support subsystems such as the CIS scheduling system and hardware management tools under development. The SCRAM architecture provides for incorporating scheduling, effort recording, and problem reporting functions.
9. SCRAM is designed to support recoveries that are conducted under the Virtual Machine operating environment, with later releases supporting recovery-on-native hardware.
10. SCRAM is designed to integrate the expression of requirements for RME and other value-added recovery management offerings in the Server/390 environment.

This Sheet Has Been Replaced



This Sheet Has Been Replaced



This Sheet Has Been Replaced



3.0 An Overview of the SCRAM Environment

Functionally, SCRAM breaks into two large components, each of which we will discuss in more detail.

1. The Data Server

   The SCRAM Data Server is actually a pair of servers, one operating outside the IBM corporate firewall and actually interfacing with the customer, and a second server operating inside the firewall and routing customer-provided data into the appropriate locations for processing.

2. The Execution Environment

   The SCRAM Data Server is actually a pair of servers, one operating outside the IBM corporate firewall and actually interfacing with the customer, and a second server operating inside the firewall and routing customer-provided data into the appropriate locations for processing.

3.1 How SCRAM Processing Works - An Overview

The question then becomes, "How are the SCRAM components deployed to match the needs of the recovery process?"

1. Understanding what resources are to be recovered.

   By driving the process toward a common worksheet or "checklist" their Server/390 system in whole or in part, depending on what s/he defines as "critical workloads." In addition, a customer contemplating recoveries at other facilities must face the challenge of deciding on levels of service, levels of recovery effort, and a number of other variables.

2. Enumerating and expressing requirements

   SCRAM attacks problems in enumerating and expressing requirements. First, it formalizes the physical and logical definitions of the entities to be recovered. Second, it standardizes the expression of those requirements. Third, it integrates both physical and logical definitions in a unique and predictable way.

   Expression is also facilitated in that SCRAM processing does away with the "base plus delta" management style that manual processing encouraged. By forcing a complete refresh of the information we are more certain to pick up all required changes and less likely to commit errors when applying changes to a base specification.

3. Scheduling sufficient resources to meet expressed needs.

   SCRAM allows improvements in the scheduling process by fostering precise expression of requirements earlier in the recovery generation process for scheduled events. This allows schedulers and pre-testers to adjust the assigned resources to more precisely meet requirements, and to more easily recognize differences between contracted and actual resources assigned.

4. Understanding the characteristics of the target configuration

   SCRAM has a way of improving service by allowing a limited subset of requirements to be automatically substituted for conformable resources. The support in the current version is limited to 3380 and 3390 direct access storage units (and their emulated equivalents).

   While SCRAM does not provide other substitutions at this time, the general data architecture and the base code allow for this facility to be extended.

5. Mapping requirements to resources

   SCRAM provides a way to efficiently map requirements to resources by matching characteristics of allocatable units across system images. SCRAM MAP processing is designed to map source to target systems and considerably reduce the need for manual intervention. SCRAM maps at the

allocatable unit level. Support is provided for mapping most controllers, direct access storage devices, tapes, printers, and a variety of other input/output (I/O) gear. Some devices do not map completely, and may require manual follow-up. However, tests have shown that SCRAM can reduce the need for manual mapping by seventy-five to ninety-six percent.

In addition, customers requesting pre-allocation and setup of their logical unit characteristics (Volume Serial (VOLSER), Volume Table of Contents (VTOC), indexing, *et cetera*.) may define those characteristics, and tools using the SCRAM input data may be invoked to assist in the formatting of jobstreams to accomplish those tasks.

6. Validating acceptability of proposed target configuration

   SCRAM does not routinely "validate" the acceptability of a customer's requirement, or the target configuration definition. However, by identifying components that cannot be automatically mapped, and separately enumerating them, SCRAM enables resource allocators schedulers, and managers to more rapidly identify potential problems so that they can be discussed with the customer.

7. Preparing pre-test configuration and setup specifications

   SCRAM is designed to produce outputs that can directly be used by pre-test organizations in the form of VM directory entries, and eventually native Input/Output Control Program (IOCP) and Input/Output Definition File (IODF) decks. In addition, SCRAM integrates with network mapping sub-components, and is designed to produce outputs directly readable by ESCON and FiCon director switching tools currently under development as part of the SCRAM environment.

8. Building the recovery environment

   SCRAM is designed to produce definitions and setup information that can be used to control both the physical and logical pre-allocation of resources to an event. This reduces the burden on, and improves the accuracy of setup and build processes because SCRAM minimizes the number of times data is transcribed and moved between programming subsystems designed to support recovery environment.

9. Certifying that the built environment meets specifications

   SCRAM assists the pre-tester to validate that the built environment meets the specifications by enabling the rapid identification of exception conditions, and by providing a comprehensive audit trail from customer request to resource assignment and generation. By tracking the progress of a request through the SCRAM programming flow, we gain enhanced positive control and make the verification task easier.

10. Turning over the environment to the customer

    The use of SCRAM can make the overall delivery easier and quicker, and enable the pre-test and other personnel to approach their task with greater confidence in the event of a real recovery, as it enables the pre-test and setup teams to work with more current definitions with greater speed and accuracy, which will lead to improved customer satisfaction.

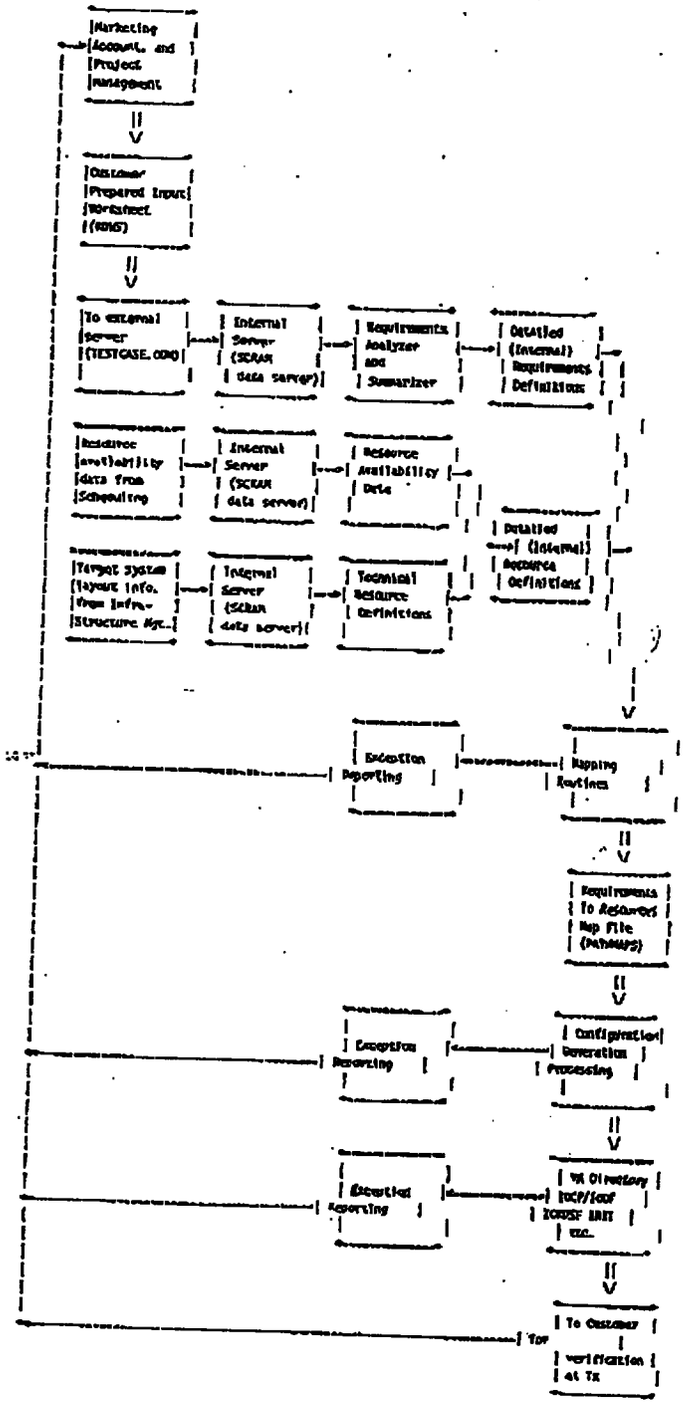
This Sheet Has Been Replaced



This Sheet Has Been Replaced

4.0 Processing Overview

4.1 Operational Flow



The above diagram shows the overall operational flow of the SCRAM process.

This Sheet Has Been Replaced



This Sheet Has Been Replaced



5.0 File and Library Naming Conventions

A key part of the SCRAM process is its ability to use coordinated sets of shared files for processing. These coordinated sets of files form the basis of an implementation that can be extended and interfaced with other efforts, such as scheduling tools and machine hardware switch control mechanisms.

The authors trust that you have some understanding of the IBM Shared File System (SFS), a part of the Conversational Monitor System (CMS) which is an ancient IBM computing support environment. An education on CMS is beyond the scope of this document. You will have to learn about file and directory naming conventions, however, as they affect the SCRAM process.

5.1 Directory Structures

The SCRAM server automatically places customer files in directories for you. Directory names are structures that allow an individual file to be assigned into a directory or subdirectory. You may be familiar with the concept of a directory by a different name. You may call it a "folder." Unlike Lotus™ folders, however, SFS directories use dots (.) rather than backslashes (\) to denote separation of different levels. Thus a folder called PROJECTS would be represented in Lotus™ as:

```
...\Folders and Views\PROJECTS
```

Where the '...' represents the "invisible" structure of the drive letter and the operating system directory. In SFS, the analogous structure is represented as:

```
SERVERNAME:MYUSERID.FOLDERSANDVIEWS:PROJECTS
```

Please note that in SFS, all the names are in uppercase, and that there are no spaces allowed in between separators. That is important.

5.2 File Identifiers

The name of a file in Notes™ is essentially arbitrary. In fact, Notes™ goes to great lengths to shield the user from the actual structure of the data elements involved. If you are familiar with Windows™ the file structure of a Lotus™ Notes™ file might look something like:

```
C:\Lotus\Notes\Working\L45MAIL.NSF
```

But all you're used to seeing is the "INBOX" which has been established with the Lotus™ Notes™ environment as an alias to that file.

In CMS, files have a three part structure to their name:

- A filename, consisting of 1 to 8 alphanational characters.
- A filetype, also consisting of 1 to 8 alphanational characters.

  By convention and tradition, some filetypes have special meaning, typically products and commands used inside the CMS environment. For the purposes of SCRAM, we are introducing the following as reserved types:

  - RDWS - used to identify Requirements Data Work Sheets
  - DEVPATHS - an intermediate working file type used to expand the definition of a device
  - PATHMAPS - a file containing a one-to-one mapping or correspondence of requirements and resources enumerated on an allocatable unit basis.

This Sheet Has Been Replaced



- JOURNAL - a file containing messages and key process history information
- ERRORLOG - a file containing a record of outright processing errors and other procedural failures
- PROTODIR - a file of statements in the form of a virtual machine (VM) directory entry, but which may require some manual review before incorporation into a running VM system.
- IOCPSEG - a file of statements in the form of an Input/Output Control Program definition segment, but which may require some manual review before incorporation into a hardware systems generation.
- SHORTAGE - a file of entries in DEVPATHS format enumerating what requested devices could not be mapped onto the target system.
- SURPLUS - a file of entires in DEVPATHS format enumerating what target system resources exist in excess of the expressed customer requirement in the RDWS being processed.

• A filemode, consisting of one alphabetic and one numeric character. The numeric character has a special reserved meaning within CMS. It defaults to '1' and is usually ignored or not specified. For our purposes, we will always assume that it is not specified. The alphabetic character designates a search order sequence from A to Z. Using the CMS ACCESS or the SCRAMIEE ACCESS commands, a particular set of files may be accessed in a specific position or sequence within the limitation of the available twenty-six choices. Some filemodes are reserved. S,Y,Z and P are typically "off-limits" for historical reasons.

5.3 Naming conventions

Now that we know a little more about naming objects in the CMS/SFS world, we need to talk about naming conventions. While CMS and SFS don't enforce naming conventions on anyone, SCRAM does enforce a basic set of conventions to enable users to more easily locate and access data.

The basis of the naming convention is the notion that like files should be stored together on a common directory structure. "Likeness" is the informal expression of a property of data morphology which involves the logical, areal, and temporal locality of reference to a given object. Objects which have a high degree of logical, areal, and temporal affinity are said to be "alike" in their storage needs.

But you're interested in a practical, not theoretical, point. The naming convention in SCRAM is based on the type of object we're considering.

5.3.1 Customer Requirements Directories

Customer requirements are stored in directories named with the following structure:

```
WSCVMU:EBRS1.CDREPOS.cccccccc.kkkkkkkk.f.dddddddd
```

Where:

cccccccc is the customer number

kkkkkkkk is the contract number

f is a flag value, 'T' for test events and 'D' for recovery definitions

dddddddd is the sorted-date associated with the event

This Sheet Has Been Replaced

5.3.2 Customer Requirements Filenames

Customer requirements are stored in directories named with the following structure:

```
nnnnnnnn Filetype m
```

Where filetype is one of:

RDWS the customer-supplied requirements definition file

DEVPATHS the extended format for both RDWS and scheduled devices

NCP NCP Source definition

CHX Channel Extended definition file

TMS Temporary Management Services - used by the *ASSETLIST processor

5.3.3 IBM Resource Directories Directories

IBM Resource Definition files are stored in directories named with the following structure:

```
WSCVMU:EBRS1.RESOURCES.ttttttt.lll.
```

Where:

ttttttt is the library type (IOCPLIB)

lll is the location code (BLD, GBG, STF)

5.3.4 SCRAM Component Directories

Customer requirements are stored in directories named with the following structure:

```
WSCVMU:EBRS1.Component.libtypen.
```

Where:

Component defines the SCRAM component

- SCRAM - the SCRAM execution shell
- SVMCODE - the SCRAM inside-the-firewall server virtual machine

libtypen defines the type of materials

- PRODLIB - holds production level code
- TESTLIB - holds test level code
- DOCLIB - holds documentation files in both host and PC formats
- SAMPLIB - holds sample materials and other demonstrations

This Sheet Has Been Replaced



This Sheet Has Been Replaced



6.0 SCRAM Commands

SCRAM processing uses a lexical user interface (LUI) model, commonly referred to as a command-line.

Interfacing to SCRAM is controlled by the SCRAMIEE command processor. The following subcommands are available:

- ACCESS
- ANALYZE
- COMPRESS
- DISCARD
- EDIT
- EXPAND
- FILELIST
- GENERATE
- LIBLIST
- MAP
- NCP
- QCONTRACT
- QSEARCH
- RELEASE
- SUMMARIZE

6.1 SCRAMIEE ACCESS

Establish the ability to view and/or change files on a disk or directory

Subcommand Format

```
        +---<diskno>--------+
ACCESS -|                   |----> Filemode </Extension>
        +---<directoryid>---+
```

ACCESS works identically to the CMS ACCESS command within the SCRAMIE environment.

Example 1:

You want to establish access to your workdisk as your primary filemode, the SCRAM control files to be searched next, and the test code library following. You would enter the following command sequence:

```
SCRAM ACCESS .WORKDISK A
SCRAM ACCESS .CONTROL  B/A
SCRAM ACCESS .TESTLIB  C
```

Example 2:

You want to establish access to your workdisk as your primary filemode, the SCRAM control files to be searched next, and the production code library following. You then want to work on a customer test and assign the working folder to mode F:

```
SCRAM ACCESS .WORKDISK A
SCRAM ACCESS .CONTROL  B/A
SCRAM ACCESS .PRODLIB  C
SCRAM ACCESS WSCVMU:EBRS1.CDREPOS.CUSTOMER.CONTRACT.EVNTDATE.T.    F
```

Notes: To UNDO the operation of the SCRAM ACCESS command, use the SCRAM RELEASE command.

6.2 SCRAMIEE ANALYZE

ANALYZE provides a way of taking customer input or input from the CIS scheduling tool and preparing it for SCRAM MAP processing. An output file is written containing the detailed device information and pathing requirements (the DEVPATHS file).

Subcommand Format

```
                +---< REQUIREMENTS nnnnnnnn tttttttt m
ANALYZE   ----|
                +---< RESOURCES    nnnnnnnn tttttttt m
```

Example:

You want to prepare an RDWS file for customer system CUSTSYS for processing by the SCRAM MAP program. It is located on your 'W' disk.

You would enter the following command:

```
SCRAM ANALYZE REQUIREMENTS CUSTSYS RDWS W
```

The system might reply with something like:

```
MJFJFR004I 00541 records available for input.
MJFJFR005I 00500 records accepted for input.
MJFJFR006I 00541 records processed.
MJFJFR001I Exiting. Rc= 0
```

As a result, the file: CUSTSYS DEVPATHS W will be created. It is needed as input to the SCRAM MAP stage.

This Sheet Has Been Replaced



6.3 SCRAMIEE COMPRESS

COMPRESS - allows the user to save disk space in the VM Shared File System by storing archival files in a compressed format. The name of the library can be chosen by the user, but should be relevant to the event. A desirable practice would be to use the SHORTNAME value associated with this event from the CUSTOMER TABLE.

Subcommand Format

```
COMPRESS InFname InFtype InFmode INTO LibraryName
```

Example 1:

You have finished processing customer BEST$ONE. You want to save the files associated with this test, but you do not want them taking up a lot of disk space. You can use the SCRAM COMPRESS command to pack them into a single library prior to deleting the individual files.

Issue the command:

```
SCRAM COMPRESS * * W INTO BEST$ONE
```

This would result in the packing of all files currently on the disk or directory currently accessed as 'W' into the compressed library BEST$ONE. You can then use the SCRAM DISCARD command to delete the original (full-size) files.

Notes:

See the SCRAM QCUSTOMER and SCRAM DISCARD commands for more information.

6.3.1 SCRAMIEE DISCARD

DISCARD- removes a file from an accessed disk or directory.

Subcommand Format

```
DISCARD TargetFName TargetFType TargetFMode
```

DISCARD acts as the equivalent to a CMS ERASE command.

6.3.2 SCRAMIEE EDIT

EDIT - Invokes the CMS eXtended EDITor (XEDIT) so that you can review, manipulate, or change the contents of disk files.

Subcommand Format

```
SCRAM EDIT TargetFname TargetFType TargetFmode
```

This Sheet Has Been Replaced



EDIT invokes the CMS XEDIT editor. Your PROFILE XEDIT may be require to give you the most familiar version of editor behaviour.

6.3.3 SCRAMIEE EXPAND

EXPAND - allows the user to recover files from an archive processed by the SCRAM COMPRESS subcommand.

Subcommand Format

```
EXPAND OutFname OutFtype OutFmode FROM LibraryName
```

Example 1:

You want to see extract an RDWS associate with a source system SOURCSYS from the archive associated with customer MYFAVE. You have established access to the filesets in question, and want to put the output in the directory currently accessed as your "K" disk.

You would issue the command:

```
SCRAM EXPAND SOURCSYS RDWS K FROM MYFAVE
```

This would result in extracting and rexpanding the file SOURCESYS RDWS from the archive MYFAVE and placing the output onto the disk or directory currently accessed at mode letter K.

Notes:

1. This is the opposite of SCRAM COMPRESS, but EXPAND operates on only one file at a time.

6.3.4 SCRAMIEE FILELIST

FILELIST - display the contents of a currently accessed disk or directory. Equivalent to the CMS FILELIST command.

Subcommand Format

```
            ---< Fname     Ftype     Fmode
FILELIST --|
            ---< *         *         *
```

FILELIST returns results equivalent to a CMS FILELIST command.

Example:

You want to see the contents of your work directory, currently accessed at mode 'B'

You would enter the following command:

This Sheet Has Been Replaced



The system might reply with something like:

```
DIRBUILD FILELIST A1 V=169 TRUNC=169 SIZE=9 LINE=4 ALT=0
DIRECTORY NAME = WSCVMU.EBRS1.CDREPOS.CUSTNUMB.CONTRACT.Z0000130.T
Cmd  Filename FileType FM Recfm  LRECL    Records    Blocks Date     Time
     RDWS     DSECT    B1 V       79        209         3 1/16/00  22:35:33
     SAMPLE   RDWS     B1 V      658        588        88 1/15/00  13:05:39
     PATHMAPS DSECT    B1 V       84        330         5 11/29/99 15:08:29
     SAMPLE   PATHMAPS B1 V     1487       1260       445 10/27/99 15:23:55
     SAMPLE   HEADER   B1 V       77          1         1 10/27/99 15:22:49
     MKSTANZA EXEC     B1 V       72         26         1 8/31/99  10:38:01
     SYS2     TXT      B1 V       41       1389        15 6/25/99  15:17:15
     RMSINITL EXEC     B1 V      109         75         1 6/07/99  8:38:08
     TESTINIT RDWS     B1 V      658         11         2 6/07/99  8:29:19
     TESTRUN3 RDWS     B1 V      658         11         2 6/07/99  8:26:55
     TESTRUN2 RDWS     B1 V      658         11         2 6/07/99  8:22:09
     TESTRUN1 RDWS     B1 V      658         11         2 6/07/99  8:21:03
     TESTRUN0 RDWS     B1 V      658         11         2 6/07/99  8:20:11

1=Help     2=Refresh    3=Quit     4=Cancel   5=Sort(Dir)   6=Sort(size)
7=Backward 8=Forward    9=FL /n    10=Share   11=XEDIT/LIST 12=Cursor

===>

                                                  X E D I T  1 File
```

6.3.5 SCRAMIEE GENERATE

GENERATE controls the actual building of a system generation information from lists of resources and requirements. (PATHMAPS file).

Subcommand Format

```
                +---< VMDIRECT FROM reqname PATHMAPS reqfm
GENERATE  ------|
                +---< IOCP     FROM reqname PATHMAPS reqfm
```

Where: reqname1 PATHMAPS reqfm is the file identifier (name: reqname1 type: PATHMAPS mode: reqfm) of the file containing the processed requirement s and their corresponding assigned resources.

IOCP is not a supported option at this time.

Example 1:

You want to prepare an generate a VM directory entry for a customer system to recovery their environment on an IBM Business Continuity and Recovery Services processor. The mapped customer requirements file is CUSTREQT PATHMAPS W.

You would enter the following command:

This Sheet Has Been Replaced

The system will then display a panel allowing you to further customize the directory entry you are about to generate. You will need to know, for example:

- Accounting code information
- Password to be assigned to the newly created userid
- Privilege classes to be assigned
- Additional standard includes for this customer
- Whether or not to generate the disks as shared minidisks or as attached DASD units

Notes

- A linemode option is available through the programming API.

6.3.6 SCRAMIEE LIBLIST

LIBLIST - allows the user to display the contents of a compressed library. Individual files may be added, deleted, expanded, or extracted.

Subcommand Format

```
LIBLIST LibraryName
```

Example

You have finished processing customer BEST$ONE. You have packed the files associated with this test into BEST$ONE PACKLIB on your 'W' disk. You want to verify the contents of the library before exiting.

Issue the command:

```
SCRAM LIBLIST BEST$ONE
```

Notes:

- See the SCRAM COMPRESS and SCRAM EXPAND commands for more information.

6.3.7 SCRAMIEE MAP

MAP controls the actual building of a system generation information from lists of resources and requirements. (DEVPATHS files).

Subcommand Format

```
MAP  reqname DEVPATHS reqfm  resname DEVPATHS resfm
```

Where:

- reqname1 DEVPATHS reqfm is the file identifier (name: reqname1 type: DEVPATHS mode: reqfm) of the file containing the processed requirements

This Sheet Has Been Replaced



- resname1 DEVPATHS resfm is the file identifier (name: resname1 type: DEVPATHS mode: resfm) of the file containing the processed requirements

The resulting output will be written to the working disk, with a file identifier of: reqname1 PATHMAPS.

Example

You want to prepare the working file that establishes a correspondence between a customer request (CUSTREQT) you have processed using the SCRAM ANALYZE REQUIREMENTS subcommand and IBM Business Continuity and Recovery Services resources. on the system you have prepared as SYSTEM0A (by using the SCRAM ANALYZE RESOURCES subcommand)

You would enter the following command:

```
SCRAM MAP CUSTREQT DEVPATHS A SYSTEM0A DEVPATHS A
```

This will create a file CUSTREQT PATHMAPS A, containing the correspondence map.

6.3.8 SCRAMIEE QCONTRACT

QCONTRACT provides a way of returning information on what customer and contract is associated with a particular set of Shared File System directories (library indices.)

Subcommand Format

```
                +---< (CUSTOMER=aaaaaaaa
QCONTRACT ----|-- < (CONTRACT=aaaaaaaa
                +---< (SHORTNAME=aaaaaaaa
```

Example 1:

You want to see what customer number is associated with contract DMK9661

You would enter the following command:

```
SCRAM QCUSTOMER (CONTRACT=DMK9661
```

The system might reply with something like:

```
MJFQCC002I Initializing.
MJFQCC003I Processing in progress. Please wait.
CUSTOMER CONTRACT   SHORTNAME DESCRIPTION
3456543  DMK966I    SYSTEST2 Test Case

MJFQCC006I 0001 records found.
```

Notes:

This Sheet Has Been Replaced



1. This is a useful command to return the SHORTNAME (to name the archive file from the SCRAMIEE COMPRESS command.
2. Alternatively, it can be used to look up a shortname or description that will tell you where a particular customer's data can be found in the file system. For example, assume you want to look up what directories store information about an upcoming test to be conducted on 02/29/2000. You know from other documentation that MYFAVE's contract number is: ABCD0001.

   You can issue the QCONTRACT subcommand:

```
SCRAM QCUSTOMER (CONTRACT=ABCD0001
```

   The system might reply with something like:

```
MJFQCC002I Initializing.
MJFQCC003I Processing in progress. Please wait.
   CUSTOMER CONTRACT   SHORTNAME DESCRIPTION
   7777777  ABCD0001   MYFAVE    My Favourite Customer

MJFQCC006I 0001 records found.
```

Now you can derive that your customer's entries can be found in the SFS directory known as:

```
WSCVMU:EBRS1.CDREPOS.7777777.ABCD0001.20000229.T.
```

Using the standard naming convention of:

```
Server:Userid.Function.Customer.Contract.EventDate.Type
```

6.3.9 SCRAMIEE QSEARCH

QSEARCH- display the current sequence in which disks or libraries are accessed and searched for files.

Subcommand Format

```
QSEARCH
```

QSEARCH returns results equivalent to a CMS QUERY ACCESSED command.

Example

You want to see the sequence in which your disks and libraries are accessed and searched for files: You would enter the following command:

```
SCRAM QSEARCH
```

This Sheet Has Been Replaced



The system might reply with something like:

```
Mode  Stat     Files  Vdev  Label/Directory
A      R/W       129  DIR   WSCVMU:MYUSERID.WORKDISK
B      R/W        39  DIR.  WSCVMU:EBRS1.SCRAM.DOCLIB
C      R/W        24  DIR   WSCVMU:EBRS1.SCRAM.CONTROL
D      R/O       319  DIR   WSCVMU:EBRS1.SCRAM.PRODLIB
L      R/O      2500  DIR   WSCVMT:TOOLS.WSCLOCAL
P      R/O      2102  DIR   WSCVMT:TOOLS.PRODUCTS
S      R/O       708  190   CMS190
T/S    R/O      4336  DIR   WSCVMT:TOOLS.WSCTOOLS
U      R/O       193  DIR   WSCVMT:TOOLS.USAPPL
Y/S    R/O      1303  19E   CMS19E
Z/Z    R/O      1280  19D   HELP
```

6.3.10 SCRAMIEE RELEASE

RELEASE - terminate the ability to view and/or change files on a disk or directory

Subcommand Format

```
RELEASE Filemode </Extension>
```

RELEASE works identically to the CMS RELEASE command within the SCRAMIEE environment.

Example You have established access to your workdisk as your primary filemode, the SCRAM control files to be searched next, and the test code library following. You want to release access to all:

```
SCRAM RELEASE A
SCRAM RELEASE B
SCRAM RELEASE C
```

Notes

- To UNDO the operation of the SCRAM RELEASE command, use the SCRAM ACCESS command.

6.3.11 SCRAMIEE SUMMARIZE

SUMMARIZE provides a way of creating a high-level listing of resources or requirements associated with a given event information (the SUMMARY file).

This allows the pre-tester or other person responsible for processing to compare the actual resource demands with the contract allocation (via the scheduling system), and to inform the account manager or project manager of significant discrepancies.

Subcommand Format

This Sheet Has Been Replaced



This Sheet Has Been Replaced


```
              +--< REQUIREMENTS nnnnnnn1 RDWS      m >--
SUMMARIZE ---|                                        |--< (NODISPLAY
              +--< RESOURCES    nnnnnnn2 DEVPATHS m >--
```

Output would be written to nnnnnnn1 SUMMARY A or nnnnnn2 SUMMARY A, respectively.

Use the (NODISPLAY option to supress display output to the console.

Example

You want to check that a customer defintion matches their contract requirements.

You would enter the following commands:

```
SCRAM SUMMARIZE REQUIREMENTS CUSTSYS  RDWS     W
SCRAM SUMMARIZE RESOURCES    AVAILABL DEVPATHS W
```

The first invocation will create CUSTSYS SUMMARY W and the second will create AVAILABL SUMMARY W. By comparing the contents of the two files, a determination can be made as to whether or not appropriate contract resources have been allocated.

A partial output from SUMMARIZE follows:

```
MJFSDC001I 20000118 22:37:07 000000 AFP      devices specified.
MJFSDC001I 20000118 22:37:07 000074 BSC1     devices specified.
MJFSDC001I 20000118 22:37:07 000003 CFS      devices specified.
MJFSDC001I 20000118 22:37:07 000207 CTC      devices specified.
MJFSDC001I 20000118 22:37:07 000207 SCTC     devices specified.
MJFSDC001I 20000118 22:37:08 000002 3172     devices specified.
MJFSDC001I 20000118 22:37:08 000003 3174     devices specified.
MJFSDC001I 20000118 22:37:08 000001 3203     devices specified.
MJFSDC001I 20000118 22:37:08 000030 3270     devices specified.
MJFSDC001I 20000118 22:37:08 000001 3274     devices specified.
MJFSDC001I 20000118 22:37:08 000030 3277     devices specified.
MJFSDC001I 20000118 22:37:08 000002 3278     devices specified.
MJFSDC001I 20000118 22:37:08 000007 3380     devices specified.
MJFSDC001I 20000118 22:37:08 000072 3390     devices specified.
MJFSDC001I 20000118 22:37:08 000004 3420     devices specified.
MJFSDC001I 20000118 22:37:09 000020 3480     devices specified.
MJFSDC001I 20000118 22:37:09 000021 3490     devices specified.
MJFSDC001I 20000118 22:37:09 000009 3705     devices specified.
MJFSDC001I 20000118 22:37:09 000033 3745     devices specified.
MJFSDC001I 20000118 22:37:09 000009 3791L    devices specified.
MJFSDC001I 20000118 22:37:09 000001 3800     devices specified.
MJFSDC001I 20000118 22:37:09 000002 3880     devices specified.
MJFSDC001I 20000118 22:37:09 000046 3990     devices specified.
MJFSDC001I 20000118 22:37:10 000000 9395     devices specified.
MJFSDC005W 20000118 22:37:10 000008 Overlapped/duplicate assignments defined.
```

This Sheet Has Been Replaced



7.0 Messages and Codes by Issuing Component

The following is a listing of messages and identifiers by component:

7.1 ACG Automated Configuraiton Generator (VM Directory Builder)

```
MJFACG     Automated Configuration Generator
MJFACG020E Invalid contract number. Cannot be blank.
MJFACG021E Path Map definition file selection terminated.
MJFACG021E Invalid contract number. Only A-Z,a-z,0-9 allowed.
MJFACG028E Include file:'"||IncludeFileName IncludeFileType||"' not found.
MJFACG030I Invalid contract name. Cannot be blank.
MJFACG060E Invalid guest userid. Cannot be blank.
MJFACG061E Invalid guest userid. Only A-Z,a-z,0-9 allowed.
MJFACG0B0E Invalid privilege classes. See installation documentation.
MJFACG090E Invalid default central storage value must be 16M < CS < 2048M
MJFACG091E Invalid maximum central storage value must be 16M < CS < 2048M
MJFACG092E Invalid expanded storage value:       must be  0M < XS < 2048M
MJFACG093E Invalid central storage values: Default < Max required.
MJFACG100I Path Map definition file selected.
MJFACG110I You selected: cccccccc PATHMAPS cc
MJFACG111R OK to proceed? <Y|N>
MJFACG120E Invalid number of processors. Value must be 1 < CPUs < 10
MJFACG128E IOCP definition file missing. Respecify.
MJFACG130E Invalid machine setting. Check list for allowed values.
MJFACG132W Warning! Not all processors allow Mode=370.
MJFACG133R Press <ENTER> to continue.
MJFACG900I nnnnnn Device records processed.
MJFACG901I Processing DASD|TAPE|PRNT|COMM|CTCA records for tttt
MJFACG920I Processing Complete. Time= nnnnnn sec. Input= rrrrrrr Residue= nnnnnn. cc%
```

7.2 AIS Analyze IOCP Specification

```
MJFAIS     Analyze IOCP Specification
MJFAIS001E Error accessing 'filename filetype filemode' RC=nnnnnn
MJFAIS002I PROCESSING ENDED. nnnn SEC. ELAPSED
```

7.3 ALI ALign IOCP records in canonical format

```
MJFALI     Input ALIgnment processor
           none
```

7.4 ALC Analyze Logical Control units

```
MJFALC     Analyze Logical Control units
           none
```

7.5 ALD Analyze Logical Devices

```
MJFALD     Analyze Logical Devices
           none
```

7.6 GAD Generate Address Definitions

This Sheet Has Been Replaced

7.7 JFR Joined Format Requirement

```
MJFJFR     Analyze Requirements Definition WorkSheet
MJFJFR001I Exiting. RC= nnnnnn
MJFJFR002E Error nnn reading filename filetype filemode
MJFJFR003I Exiting.
MJFJFR004I nnnnn records available for input
MJFJFR005I nnnnn records accepted as input
MJFJFR006I nnnnn records processed.
```

7.8 PAM Pathing Assignment Manager

```
MJFPAM     IOCP device Path Assignment Manager
MJFPAM001I nnnnnn UNIT SPECIFICATIONS PROCESSED.
MJFPAM002I nnnnnn DEFINITION RECORDS PROCESSED.
MJFPAM003I nnn.nn SECONDS ELAPSED.
```

7.9 MCM Multi-Criteria Mapping

```
MJFMCM     Multi-Criteria Mapping
MJFMCM001I Device type: tttttttt nnnnn devices required.
MJFMCM002W Device type: ttttt nnnnnn required/ nnnnnn available.
MJFMCM003I Initializing.
MJFMCM008E Insufficient ttttt devices in target configuration
MJFMCM009E Unknown source device type encountered.
MJFMCM010W Resource shortages noted.
MJFMCM011W Substituted ddddd - nnnnnnnnn (aaaa) for ddddd-nnnnnnnnn (aaaa)
MJFMCM012E Substitutions exhausted for dddd - nnnnnnnnnn
MJFMCM020E Requested tttttttt not available. xxxx is not defined/wrong type.
MJFMCM021E Requested tttttttt not available. xxxx assignment is restricted.
MJFMCM022I Assignment restricted for tttttttt xxxx.
MJFMCM023E Requested resource tttttttt xxxx not available.
MJFMCM024I Forcing assignment for sourcetype sourceaddr to targettype targeta
MJFMCM030E Source file not found. Rc=nnnnn
MJFMCM031E Source file not found. Rc=nnnnn
MJFMCM032E Target file not found. Rc=nnnnn
MJFMCM033E Target file not found. Rc=nnnnn
MJFMCM040I Processing tttttttt nnnnn definitions found.
MJFMCM041I Processing tttttttt complete.
MJFMCM042I Processing tttttttt nnnnn definitions processed.
MJFMCM050I Processing complete. nnnnn devices nnnn.nn seconds nn.nn dev/sec.
```

7.10 SDC - Summarize Device Counters

```
MJFSDC     Summarize Device Counters
MJFSDC001I yyyymmdd tt:tt:tt nnnnnn tttttttt devices specified.
MJFSDC002I Initializing.
MJFSDC003I Processing in progress. Please wait.
MJFSDC004I Processing complete. Exiting.
MJFSDC005W Warning! nnnnn Overlapped/duplicate assignments defined.
MJFSDC007E Definition overlap begins at record nnnnnnn.
```

< < END OF DOCUMENT > >

APPENDIX B

This Sheet Has Been Replaced



INTERNATIONAL BUSINESS MACHINES CORPORATION
GLOBAL SERVICES
BUSINESS CONTINUITY AND RECOVERY SERVICES
SERVER/390 RECOVERY AND CONTINUITY AUTOMATION MANAGER
REQUIREMENTS DEFINITION WORKSHEET USER GUIDE

Document Number SCRAM-390-UG

January 15, 2000

Business Continuity and Recovery Services Productivity Team

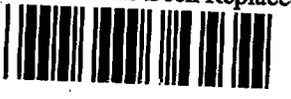
Business Continuity and Recovery Services
Building 183
800 North Frederick Avenue
Gaithersburg, MD 20879

IBM Confidential until Announce

This Sheet Has Been Replaced



This Sheet Has Been Replaced



Contents

Notice v

Summary of Changes vii

1.0 Communicating Your Technical Specifications to IBM Business Continuity and Recovery Services 1

1.1 Background 1

2.0 Specifying recovery requirements using a Requirements Definition Work Sheet 3

2.1 The Requirements Definition Worksheet File (RDWS) 3

2.2 RDWS Directives 3

2.3 RDWS Definitions 3

2.3.1 A Partial Example of the Requirements Definition Work Sheet (RDWS) 7

2.3.2 Another Example of the Requirements Definition Work Sheet (RDWS) 12

3.0 Specifying networking requirements using the Network Control Program (NCP) source datasets 15

4.0 Specifying asset and tape packing lists for recovery/test exercises 17

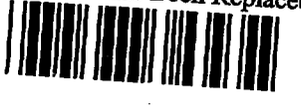
5.0 Specifying channel extender requirements 19

6.0 Transmission Protocols 23

This Sheet Has Been Replaced



This Sheet Has Been Replaced



Preface

Applicability

Objective: *To improve productivity for Business Continuity and Recovery Services personnel involved in the management and support of customer recoveries and tests in the Server/390 environment.*

Scope: *This document describes the requirements for Server/390 processor complex recovery configuration generation and maintenance.*

Compliance: *Compliance with these procedures is mandatory and subject to Audit inspection. Any noncompliance must be documented in a risk acceptance.*

Note:

- Within this document, the reference to IBM is to be understood to mean IBM and wholly owned subsidiaries. This includes references to information classification.
- If there is a conflict between this standard and local laws or industrial relations agreements in a country, the conflict and management's decision on deviations from this standard must be documented.
- If this Standard requires that a process be defined, that process needs must be documented unless stated otherwise. Required documentation may be done at any level within an organization.

Definition: Within this document, the term *Provider of Service* is used to refer to the organization and/or management which is providing information processing services, and is to be considered synonymous with local terms used at various IBM sites, such as:

- Supplier of Service
- Service Provider
- Service Delivery
- Information Systems (I/S) Service

Document Control:

- This document will be reviewed annually and will be re-issued when revisions are necessary and approved. Obsolete copies of this document should be destroyed as soon as practical and shall be the responsibility of the holder.

  Change History: (Only the changes introduced in the most recent version will be itemized and marked by revision code.)
  - Draft Release Beta 001 - January 15, 2000
  - Control requirements for systems exempted from this standard are clarified in the document Preface.

This Sheet Has Been Replaced



This Sheet Has Been Replaced



Summary of Changes

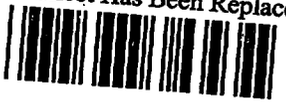
2000Orb5 Changes:

1. First Edition

This Sheet Has Been Replaced



This Sheet Has Been Replaced



1.0 Communicating Your Technical Specifications to IBM Business Continuity and Recovery Services

1.1 Background

The use of new tools to speed the creation of customer recovery environments has mandated that a Electronic Document Interchange (EDI) standard be developed for communicating information about your system to IBM Business Continuity and Recovery Services. This data is at a detailed, system programming and support level. The EDI standard is called the Requirements Definition WorkSheet (RDWS).

The RDWS is a stylized, portable way of electronically communicating your System/390 recovery configuration requirements directly to Business Continuity and Recovery Services with a minimum of human intervention. This enables Business Continuity and Recovery Services, in turn, to improve service turnaround through the use of more extensive automation and improve general readiness by allowing you to routinely update your configuration data between tests or real recovery events.

In addition, customers who have contracted for additional services such as the Recovery Management Services offerings will find that the use of a single reference document will also enable

1. One change control point for key configuration information
2. Automated processing of Direct Access Storage Device (DASD) volume formatting and labelling.
3. Reduction in service time for system recovery by compressing or eliminating re-IPLs of a recovery starter in some (not all) circumstances.

This Sheet Has Been Replaced



This Sheet Has Been Replaced



2.0 Specifying recovery requirements using a Requirements Definition Work Sheet

2.1 The Requirements Definition Worksheet File (RDWS)

The RDWS consists of a series of character vectors, each vector consisting of one or more tags and one or more associated values. The record format is considered to be static, but may be extended in future releases of DIRBUILD.

The RDWS may be logically viewed as a series of directives and definitions.

2.2 RDWS Directives

*** (Comment)** is used to provide readability and other textual information.

***HEADER** is used to hold information on the source of the request, and its intended disposition. ***HEADER is a mandatory directive.** Fields in the *HEADER directive include:

CUST= Records the Business Continuity and Recovery Services customer number associated with this request.

SYSTEM= Records the common name of the system being processed. If the system is in actuality a part of a multi-image environment, this field should record the name of the overall processor complex.

CONTRACT= Records the contract number of the event being processed.

EDATE= Indicates the date associated with the event being processed. Note the format must be in Date(S) (sorted) format.

EVENTTYPE= has two values: D is used for disaster recovery preparedness, while T is used to indicate that this RDWS is associated with a specific readiness test.

***CINFO** is used to hold information on the source of the request, and its intended disposition. ***CINFO is a mandatory directive.** Fields in the *CINFO directive include:

CONTACT=xxxxxxxxxxxxxxxxxxxxxxxx Records a name to contact at your location in case Business Continuity and Recovery Services personnel have questions about the content of the transmission. The contact name field is limited to 32 characters. Please specify name in the format: Surname, Firstname

CPHONE= Records a the telephone number associated with the contact name. Include any required country/area codes or prefixing information.

EMAIL= Records an internet or intranet EMAIL address associated with the contract name.

```
*CINFO CONTACT=Greyhame, Gandalf  CPHONE=301-555-1212 EMAIL=GANDALF@MyHomeNode.Net
```

***CNAME** Prefixes up to 40 characters giving a descriptive name of the customer and/or system. For example,

```
*CNAME Very Good Customer SystemA
```

***CHANEXTN c <*CHXSOURCE nnnnnnnn>** Instructs Business Continuity and Recovery Services that recovery requires specially defined channel extender devices on the target system. Valid values are:

- Y - Channel extender defintition information is being supplied in a separate file.
- N - No channel extenders are present on this sytem.

This Sheet Has Been Replaced



If *CHANEXTN Y is specified, the CHXSOURCE operand indicates the name of the file containing the channel extender defintions.

If *CHANEXTN Y is specified, see the section below on preparing and transmitting channel extender definition information to Business Continuity and Recovery Services. Examples:

```
*CHANEXTN Y <*CHXSOURCE* nnnnnnnn>
*CHANEXTN N
```

*CONSOLE xxxxxxx tttttt-mmmmmmmmm cccccccc Instructs Business Continuity and Recovery Services that the following addresses (RDEVS,UCBS) are to be reserved as system console address.

Code one *CONSOLE directive for each device you want assigned to the console device class

*CONSOLE should be used ONLY if you are running in a native or LPAR recovery environment. Users recovering their systems in a configuration where VM is in control of the real hardware may omit this statement.

- xxxxxxx is the hexadecimal device addreess
- tttttt is the device type
- mmmmmmmm is the device model
- cccccccc is the console class short name

```
*CONSOLE  0000009     3278-3        MSTRCONS
*CONSOLE  0000609     3278-3        ALTCONS
```

*ENVIRONMENT c Instructs Business Continuity and Recovery Services that the recovery or test is to be managed using one of three possible environments below. Consult with your account manager or project manager if you need to code options N or L, below.

N — Coding *ENVIRONMENT N instructs Business Continuity and Recovery Services that the specified recovery defintions are to be mapped onto a native machine (i.e., one with NO multi-image manager active).

L — Coding *ENVIRONMENT L instructs Business Continuity and Recovery Services that the specified recovery defintions are to be mapped onto an LPAR-only multi-image managed system.

V — Coding *ENVIRONMENT V instructs Business Continuity and Recovery Services that any recovery performed from these definitions is permitted in a virtual machine environment. *ENVIRONMNENT V is the default.

*ESOTERIC xxxxxxx tttttt-mmmmmmmmm eeeeeeee Instructs Business Continuity and Recovery Services that the following addresses (RDEVS,UCBS) are to be reserved as MVS esoteric device identifier. Use the directive only if you are configuring a worksheet for a processor that will be recovered natively (or in a customer-defined LPAR).

Code one *ESOTERIC directive for each device you want assigned to a specific esoteric device class.

- xxxxxxx is the hexadecimal device addreess
- tttttt is the device type
- mmmmmmmm is the device model
- eeeeeeee is the esoteric short name

```
*ESOTERIC 0000180 3490-E      349E
*ESOTERIC 0001181 3422-1      9TRK
```

This Sheet Has Been Replaced



***FIXEDADD xxxxxxx ttt Comment** Instructs Business Continuity and Recovery Services that the following addresses (RDEVS,UCBS) must be mapped to identical addresses on the target system. This is used when defining MCS consoles, or DASD that requires a static location.

```
*FIXEDADD 0000009 3278 MCS Console
*FIXEDADD 0001F00 3330 DASD with strange I/O Protocol
```

***LPAR** The *LPAR directive (along with the *PROCESSOR directive) is used to express requirements for processor resources on the target configuration.

```
*LPAR *LPN=nnnnnnnn *LPID=oo *NP=pp *CS=cccc *ES=eeeee *WT=www *DED=f *ADMOPT=a
```

Where

- nnnnnnnn = the name of the partition in question
- oo = the partition identifier number
- pp = is the number of logical processors defined to this lpar
- cccc = is the quantity of configured central storage, in MegaBytes (MB).
- eeeee = is the quantity of configured expanded storage, in MegaBytes (MB).
- www = is the LPAR weighting factor for this partition
- f = flag indicating whether the LPAR uses dedicated processors (D) or shared processors (S)
- a = flag indicating whether or not the LPAR uses the ADM option (Y=yes, N=no)

PROCESSOR** The *PROCESSOR directive (along with the *LPAR directive) is used to express requirements for processor resources on the target configuration. ****PROCESSOR is a mandatory directive.

```
*PROCESSOR *TYPE=xxxx *MDL=yyy *NPROC=pp *SN=sssssss *CS=cccc *ES=eeeee
```

Where

- xxxx = is the device model of the processor (e.g., 9672)
- yyy = is the model number of the processor (e.g., RX4)
- pp = is the number of processors active on the machine.
- sssssss = is the serial number of the machine
  if you have NO serial number dependent software, you may code ANY in this field.
- cccc = is the quantity of configured central storage, in MegaBytes (MB).
- eeeee = is the quantity of configured expanded storage, in MegaBytes (MB).

NCPINFO x <*NCPLEVEL= nnnnnnnnnn *SSPLEVEL=nnnnnnnn *NCPSOURCE= cccccccc** The *NCPINFO directive indicates whether or not an NCP source deck will be transmitted for mapping onto Business Continuity and Recovery Services supplied hardware. ****NCPINFO is a mandatory directive. Valid values are:

- Y - NCP generation information is being supplied in a separate file
- N - No NCP generation information is being supplied to Business Continuity and Recovery Services

If *NCPINFO Y is specified, the *NCPLEVEL= and *SSPLEVEL= operands indicate the level of the libraries used to generate the NCP and SSP code. If the level used is not available at the recovery site, or cannot be provided by IBM Business Continuity and Recovery Services personnel, a process exception will be generated, and the customer will be notified to make arrangements to port or archive key libraries to manage the network specificaiton process.

If *NCPINFO Y is specified, see the section below on preparing and transmitting NCP information to Business Continuity and Recovery Services.

Code one *NCPINFO directive for each NCP image to be transmitted to Business Continuity and Recovery Services

Examples:

This Sheet Has Been Replaced



```
*NCPINFO  Y  *NCPLEVEL= 7.8        *SSPLEVEL=4.8  *NCPSOURCE= NCPNAME1
*NCPINFO  N
```

***SCPLPAR nnnnnnnn xxxxxxxx vvvvvvv yyyy Comment** Instructs Business Continuity and Recovery Services that the system control program to be activated in LPAR nnnnnnnn is defined by the literal xxxxxxxx, at version/release level vvvvvvv with a PUT/service update level of yyyy.

*SCPLPARis functionally equivalent to *SCPTYPE, but has an additional (first) operand of an LPAR name or identifier. ***SCPLPAR is a required directive in an LPAR environment.** Use of the *SCPLPAR directive will not directly affect resource allocation processing, but is reserved for informational use. Valid *SCPLPAR values for this release level are:

- *SCPLPAR ..... MVS/ESA
- *SCPLPAR OS/390
- *SCPLPAR ..... VM/ESA

Examples:

```
*SCPLPAR  LPAROLD1 MVS/ESA V5R2L2 9802 JES2
*SCPLPAR  LPARPROD OS/390  R2.5   9901 JES2
*SCPLPAR  LPARTEST OS/390  R2.7   9902 JES3
*SCPLPAR  LPARSLOW VM/ESA  V2R4.0 9904 CMS15
```

***SCPTYPE xxxxxxxx vvvvvvv yyyy Comment** Instructs Business Continuity and Recovery Services that the primary system control program is defined by the literal xxxxxxxx, at version/release level vvvvvvv with a PUT/service update level of yyyy. **Code *SCPTYPE when running native or when all LPARs have identical SCP service levels.** Use of the *SCPTYPE directive will not directly affect resource allocation processing, but is reserved for informational use. Valid *SCPTYPE values for this release level are:

- *SCPTYPE MVS/ESA
- *SCPTYPE OS/390
- *SCPTYPE VM/ESA

Examples:

```
*SCPTYPE  MVS/ESA V5R2L2 9802 JES2
*SCPTYPE  OS/390  R2.5   9901 JES2
*SCPTYPE  OS/390  R2.7   9902 JES3
*SCPTYPE  VM/ESA  V2R4.0 9904 CMS15
```

***ASSETLIST c <*LISTSOURCE= nnnnnnnn>** Instructs Business Continuity and Recovery Services that the a list of assets (tapes, customer supplied equipment, etc.,) accompanying the event is to be defined for audit purposes in the file named nnnnnnnn.

'c' is either Y or N. Use of the *ASSETLIST directive will not directly affect resource allocation processing, but is reserved for informational use. Examples:

```
*ASSETLIST N
*ASSETLIST Y *LISTSOURCE= PICKLIST
```

If *ASSETLIST Y is specified, see the section below on preparing and transmitting asset list information to Business Continuity and Recovery Services.

Code one *ASSETLIST directive for each recovery event.

This Sheet Has Been Replaced



2.3 RDWS Definitions

The RDWS definition consists of a series of records, one record for each device or control unit that the customer intends to recover on the target system, each conforming to the following format: Optional data elements are identified with a plus-sign (+) before the field name. Optional elements may be defaulted to blanks (x'40'). Note that all tags (i.e., ' *tagname= ') are considered mandatory.

FIELD NAME	TYPE	DATATYPE/ LENGTH	VALUE	DESCRIPTION
RDEFDRCT	TAG	CL8	'*DEFINE '	Constant
	PAD	CL1	' '	
DEVTYPE1	VALUE	CL4		Device Type Code the IBM machine type numeric designator (e.g., 9395, 3590, 3990, etc.)
DEVTYPE2	RESERVED	CL4	Reserved for future use	
	PAD	CL1	' '	
DEVADDR	VALUE	CL7		Device Address (Hexadecimal address) padded with leading zeroes. i.e., 0000240 or 000FFFF.
	TAG	CL12	' *CNTLUNIT= '	
CUATTACH	VALUE	CL7		Control Unit Address (Hexadecimal address) padded with leading zeroes. i.e., 0000240 or 000FFFF.
	TAG	CL9	' *MODEL= '	Model Information
DEVMODEL	VALUE	CL8		While optional, it is desirable to fill in this field. The following devices have a default model assigned by the process: 3390 = 3390-3 3380 = 3380-K 3480 = 3480E 3490 = 3490E 3278 = 3278-2 3179 = 3179-S3G 3472 = 3472-3G
	TAG	CL11	' *FEATURE= '	
+DEVFEATR	VALUE	CL40		Feature Info Note: For control units with the EMC

This Sheet Has Been Replaced



				TimeFinder feature, code *FEATURE= as TIMEFINDER(v.r.l) where v, r, and l are the version release and level of the TimeFinder host software package.
	TAG	CL11	' *DYNAMIC= '	
DEVDYNAM	VALUE	CL3		Dynamic=YES|NO
	TAG	CL9	' *LPARS= '	
LPNAMES	VALUE	15*CL9		15 fields of 8 characters separated by X'40' listing the LPARs to which the device is attached, if any. Code blanks if none.
	TAG	CL11	' *SYSNAME= '	
+ BASENAME	VALUE	CL8		Name of system or complex if multi-image managed
	TAG	CL8	' *PATH= '	
+ PATHSET	VALUE	10*CL3		10 fields of 2 hex digits separated by x'40' listing the paths to which the device is attached.
	TAG	CL8	' *LINK= '	
+ LINKSET	VALUE	10*CL3		10 fields of 2 hex digits separated by X'40' listing the LINKS to which the device is attached.
	TAG	CL11	' *CONNECT= '	
+ TARGETSYS	VALUE	CL9		Name of other LPAR or system to which a xCTC or 3088 is to be attached
+ TARGETADR	VALUE	CL7		Address on other

This Sheet Has Been Replaced



				LPAR or system to which a XCTC or 3088 is to be attached
	TAG	CL9	' *LABEL= '	
+ DEVLABEL	VALUE	CL8		Label of device (in case of DASD, give volume label)
	TAG	CL7	' *SMS= '	
+ SMSFLAG	VALUE	CL1		Is this device an SMS device? Value: Y or N
	TAG	CL6	' *SG= '	
+ STORGPID	VALUE	CL16		Name of SMS storage group to which this device belongs if any.
	TAG	CL8	' *VTOC= '	
+ VTOCPARM	VALUE	CL16		If special VTOC initialization parameters are required, the first 16 characters may be stored here. e.g., (0,1,15)
	TAG	CL9	' *INDEX= '	
+ INDXPARM	VALUE	CL16		If an INDEX VTOC is used, specify the INDEX parameters here. e.g., (0,2,90) Only the first 16 characters may be stored here.
	TAG	CL8	' *VVDS= '	
+ VVDSPARM	VALUE	CL16		If an VVDS VTOC is used, specify the VVDS parameters here. e.g., (0,2,45) Only the first 16 characters may be stored here.
	TAG	CL9	' *RTECH= '	
+ RECVRTYP	VALUE	CL2		One character flag indicating one of the following choices: 'B' Business

This Sheet Has Been Replaced



				Continutity Volume (EMC only) 'C' Restore CP Volume 'D' Restore dataset level 'F' Restore full volume 'I' Restore Index VTOC volume 'M' Restore Minidisk (Perform ICKDSF for VM volume) 'O' Restore OS volume (Perform ICKDSF for native OS volume) 'P' Primary volume of duplex/mirror pair 'S' Secondary volume of duplex/mirror pair 'V' Restore VVDS volume
+ RECVTOOL	VALUE	CL14		If the customer requires a special utility to restore this data, specify the first 14 characters of the name of the tool here: e.g., DDR, FDR, DFDSS, IDCAMS, etc.
	TAG	CL10	' *EMUNIT= '	
+ EMUNIT	VALUE	CL2		One character flag indicating one of the following Y = Device is emulated on another real device type N = Device is not emulated (N is the default)
	TAG	CL9	' *RDMFR= '	
+ RDEVMFR	VALUE	CL8		Manufacturer of real device on which this emulated device is mapped

This Sheet Has Been Replaced



	TAG	CL10	' *RDTYPE= '	
+ RDEVTYP	VALUE	CL5		Type of the real device on which this emulated device is mapped
	TAG	CL10	' *RDMODL= '	
+ RDEVMDL	VALUE	CL5		Model designator of real device on which this emulated device is mapped
	TAG	CL10	' *RDUNIT= '	
+ RMDAUNIT	VALUE	CL2		Allocation unit type for this emulated device Choose one of: B = FBA blocks C = Cylinders
	TAG	CL9	' *RDALOC= '	
+ RMDALOC	VALUE	CL10		Count of usable allocation unit on this device expressed in integer count in terms of RMDAUNITs, above

Notes

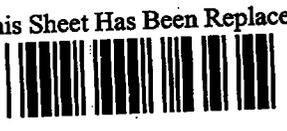
1. While a great deal of material is optional and can be defaulted to blanks, the subect of emulated devices deserves some special attention at this point.

   If the emulated devices match the original IBM device type in all particulars, (especially number of usable cylinders during volume format/allocate), it may be coded as a "native" device and no further action is required on the part of the person filling out the RDWS.

   On the other hand, if the emulated device does not precisely match the characteristics of the original device (e.g., emulating an IBM 3390-9 volume on an IBM RAMAC Scalable Array (RSA) volume), then *EMUNIT=Y must be coded, and the subsequent tags (*RDMFR= to *RDALOC=) must be fully filled in.

   This additional data is required in the event that the mapping of source to target resources requires the substitution of one emulating subsystem for another, for example, the customer has EMC™ Hypervolumes™ defining a non-standard 3390 allocation which must then be mapped onto an IBM RSA Flexvolume.

This Sheet Has Been Replaced



2.3.1 A Partial Example of the Requirements Definition Work Sheet (RDWS)

In the example below, please note that '...' denotes an ellipsis, not a required place-holding character.

```
*
*HEADER CUST=01234567 SYSTEM=MINITEST CONTRACT=Z123456 EDATE=19990229 ETYPE=T
*
*CINFO CONTACT=Greyhame, Gandalf   CPHONE=301-555-1212 EMAIL=GANDALF@MyHomeNode.Net
*
*CNAME VERY_GOOD_CUSTOMER
*
*ENVIRONMENT Y
*
*ESOTERIC 0000401       3490-E          P349
*ESOTERIC 0000402       3490-E          P349
*
*
*FIXEDADD 0000301 3278 MCS Console
*FIXEDADD 0000500 3880 Very old style controller
*FIXEDADD 0000561 3350 Antique DASD that is still referenced by UCB
*
*CONSOLE  0000301 3278-3   MASTCONS
*
*SCPTYPE  OS/390 V2R7  9903 JES2
*
*NCPINFO  Y  *NCPLEVEL= 7.8        *SSPLEVEL=4.8  *NCPSOURCE= NCPIMAG1
*NCPINFO  Y  *NCPLEVEL= 7.8        *SSPLEVEL=4.8  *NCPSOURCE= NCPIMAG2
* ------------------------------------------------------------------
*PROCESSOR *TYPE=9672 *MDL=R55 *NPROC=05 *SN=0123456 *CS=1024 *ES=02048
* ------------------------------------------------------------------
*DEFINE 3390    0000280 *CNTLUNIT= 0000280 *MODEL= 6      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *PATH= 02 03 04 05  ...
*DEFINE 3390    0000200 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *LABEL= SYSRES *SMS= N ...
*DEFINE 3390    0000201 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *LABEL= SPOOL1 *SMS= N ...
*DEFINE 3390    0000202 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *LABEL= SPOOL2 *SMS= N ...
*DEFINE 3390    0000203 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *LABEL= PAGE01 *SMS= N ...
*DEFINE 3390    0000204 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *LABEL= PAGE02 *SMS= N ...
*DEFINE 3390    0000205 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *LABEL= USER01 *SMS= N ...
*DEFINE 3390    0000206 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *LABEL= USER02 *SMS= N ...
*DEFINE 3390    0000207 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *LABEL= USER03 *SMS= N ...
*
*DEFINE 3880    0000500 *CNTLUNIT= 0000200 *MODEL= 11     *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *PATH= 06 07 ...
*DEFINE 3350    0000501 *CNTLUNIT= 0000200 *MODEL=        *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE ... *PATH= 06 07 ...
*
*DEFINE 3274    0000300 *CNTLUNIT= 0000300 *MODEL= 2A     ....
*DEFINE 3278    0000301 *CNTLUNIT= 0000300 *MODEL= 3      ....
*
*DEFINE 3490    0000400 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE  ---- *PATH= 08 09 0A 0B  ...
*DEFINE 3490    0000400 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE  .... *LABEL=        *SMS= N ...
*DEFINE 3490    0000401 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE  .... *LABEL=        *SMS= N ...
*DEFINE 3490    0000402 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE  .... *LABEL=        *SMS= N ...
*DEFINE 3490    0000403 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= NONE  .... *LABEL=        *SMS= N ...
```

2.3.2 Another Example of the Requirements Definition Work Sheet (RDWS)

The following example illustrates the use of both the *PROCESSOR and the *LPAR directive.

In the example below, please note that '...' denotes an ellipsis, not a required place-holding character.

This Sheet Has Been Replaced



```
*
*HEADER CUST=01234567 SYSTEM=FRAGMENT CONTRACT=Z123456 EDATE=19990229 ETYPE=T
*
*CINFO CONTACT=Greyhame, Gandalf   CPHONE=301-555-1212 EMAIL=GANDALF@MyHomeNode.Net
*
*CNAME VERY_GOOD_CUSTOMER
*
*CHAMEXTN Y CMXSOURCE DEFEXTNO
*
*
*ENVIRONMENT Y
*
*FIXEDADD 0301 3278 MCS Console
*FIXEDADD 0500 3880 Very old style controller
*FIXEDADD 0501 3350 Antique DASD that is still referenced by UCB
*
*NCPINFO  Y *NCPLEVEL= 7.8 *SSPLEVEL=4.8  *NCPSOURCE= MYNCP01
*
* ---------------------------------------------------------------
*PROCESSOR *TYPE=9672 *MDL=RX5 *NPROC=10 *SN=0123456 *CS=1424 *ES=02048
* ---------------------------------------------------------------
*LPAR *LPN=LP01     *LPID=01 *NP=03 *CS=0128 *ES=00512 *WT=050 *DED=N *ADMOPT=N
*LPAR *LPN=LP02     *LPID=02 *NP=07 *CS=0256 *ES=00512 *WT=050 *DED=N *ADMOPT=N
*SCPLPAR  LP01      OS/390 V2R7  9903 JES2
*SCPLPAR  LP02      OS/390 V2R5  9903 JES2
* ---------------------------------------------------------------
*DEFINE 3990   0000200 *CNTLUNIT= 0000200 *MODEL= 6      *FEATURE= NONE      *DYNAMIC= NO  *LPARS= LP01 ... *PATH= 02 03 04 05  ...
*DEFINE 3390   0000200 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE      *DYNAMIC= NO  *LPARS= LP02 ... *LABEL= SYSRES *SMS= N
*DEFINE 3390   0000201 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP01 ... *LABEL= SPOOL1 *SMS= N ...
*DEFINE 3390   0000202 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP01 ... *LABEL= SPOOL2 *SMS= N ...
*DEFINE 3390   0000203 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP01 ... *LABEL= PAGE01 *SMS= N ...
*DEFINE 3390   0000204 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP01 ... *LABEL= PAGE02 *SMS= N ...
*DEFINE 3390   0000205 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP01 ... *LABEL= USER01 *SMS= N ...
*DEFINE 3390   0000206 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP01 ... *LABEL= USER02 *SMS= N ...
*DEFINE 3390   0000207 *CNTLUNIT= 0000200 *MODEL= 3      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP01 ... *LABEL= USER03 *SMS= N ...
*
*DEFINE 3880   0000500 *CNTLUNIT= 0000200 *MODEL= 11     *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP02 ... *PATH= 06 07 ...
*DEFINE 3350   0000501 *CNTLUNIT= 0000200 *MODEL=        *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP02 ... *PATH= 06 07 ...
*
*DEFINE 3274   0000300 *CNTLUNIT= 0000300 *MODEL= 2A     ....*DYNAMIC= YES
*DEFINE 3278   0000301 *CNTLUNIT= 0000300 *MODEL= 3      ....*DYNAMIC= YES
*
*DEFINE 3490   0000400 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP02 .... *PATH= 08 09 0A 0B  ...
*DEFINE 3490   0000400 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP02 .... *LABEL=        *SMS= N ...
*DEFINE 3490   0000401 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP02 .... *LABEL=        *SMS= N ...
*DEFINE 3490   0000402 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP02 .... *LABEL=        *SMS= N ...
*DEFINE 3490   0000403 *CNTLUNIT= 0000400 *MODEL= E      *FEATURE= NONE *DYNAMIC= NO  *LPARS= LP02 .... *LABEL=        *SMS= N ...
```

This Sheet Has Been Replaced



This Sheet Has Been Replaced



3.0 Specifying networking requirements using the Network Control Program (NCP) source datasets

Customers who wish to have their NCPs automatically convered to run on Business Continuity and Recovery Services 3745 hardware may do so by sending their source files to the Server as separate files from the RDWS.

Use the transmission techniques that are described in the next section of this document. Send each NCP as a separate file, following the prescribed naming conventions. Please ensure that the file(s) sent are the original source before any processing by NDF.

Once the source files have been identified, you will need to place a *HEADER directive as the first (comment) line in the NCP source file before transmitting it to Business Continuity and Recovery Services.

*HEADER Is used to hold information on the source of the request, and its intended disposition. *HEADER is a mandatory directive. Fields in the *HEADER directive include:

- CUST= Records the Business Continuity and Recovery Services customer number associated with this request.
- SYSTEM= Records the common name of the NCP being processed.
- CONTRACT= Records the contract number of the event being processed.
- EDATE= Indicates the date associated with the event being processed. Note the format must be in Date(S) (sorted) format.
- ETYPE= has two values: D is used for disaster recovery preparedness, while T is used to indicate that this NCP definition is associated with a specific readiness test.

```
*HEADER CUST=01234567 SYSTEM=NCPNAM1  CONTRACT=Z123456 EDATE=19990229 ETYPE=T
```

Business Continuity and Recovery Services will process these files, re-assigning lines, TICs, and channels to match our hardware configurations. LIC weights and scanner utilization will be determined by NCP parameters so it is important that the SPEED parameter accurately reflects how the lines are being used.

For file naming conventions, please see the section on transmitting data to the server, below.

This Sheet Has Been Replaced



This Sheet Has Been Replaced



4.0 Specifying asset and tape packing lists for recovery/test exercises

Customers who wish to enumerate their tape lists for audit/handling control purposes may do so by specifying a ASSETLIST file and sending it to the Server as a file separate from the RDWS. Each record of the ASSETLIST file has the following format:

*ASSET cccccccc eeeeeeee tttttttt ddddddddddddddddddddddddddddddd

Where:

cccccccccc represents an asset class (e.g., MODEM, CONNECTR, CABLE, LIBRARY, DEVICE, TAPE - the generic TAPE will refer to both reel and cartridge tapes).

eeeeeeee represents an external identifier (in the case of tape, a volume serial number or VOLSER identifier).

tttttttt represents an asset type. For tapes, code one of the following:

- 3422 (includes 3420)
- 3480E
- 3490
- 3490E

ddddddddddddddddddddddddddddddd represents up to 32 bytes of descriptive information.

Use the transmission techniques that are described in the next section of this document.

Once the source files have been identified, you will need to place a *HEADER directive as the first (comment) line in the ASSETLIST source file before transmitting it to Business Continuity and Recovery Services The actual file format is RECFM=V LRECL=80.

***HEADER** is used to hold information on the source of the request, and its intended disposition. *HEADER is a mandatory directive. Fields in the *HEADER directive include:

CUST= Records the Business Continuity and Recovery Services customer number associated with this request.

SYSTEM= Records the common name of the system being processed. If the system is in actuality a part of a multi-image environment, this field should record the name of the overall processor complex.

CONTRACT= Records the contract number of the event being processed.

EDATE= Indicates the date associated with the event being processed. Note the format must be in Date(S) (sorted) format.

ETYPE= has two values: D is used for disaster recovery preparedness, while T is used to indicate that this tape packing list definition is associated with a specific readiness test.

This Sheet Has Been Replaced



```
*
*HEADER CUST=01234567 SYSTEM=SYSTEM4  CONTRACT=Z123456 EDATE=19990229 ETYPE=T
*
* ---------------------------------------------------
* A sample ASSETLIST file for a very small but varied test
* ---------------------------------------------------
*ASSET 3277GA     HK12774  TERMINAL Custom 3277GA display
*ASSET CABLE      7X2274H  CABLE    Special cable for 3277GA
*ASSET TAPE       ABC123   3422     Antique archive tape
*ASSET TAPE       XYZ980   3480E
*ASSET TAPE       XYZ981   3480E
*ASSET TAPE       XYZ982   3480E
*ASSET TAPE       XYZ983   3480E
*ASSET TAPE       MM3391   3490
*ASSET TAPE       HQ4392   3490E
*ASSET TAPE       HQ4393   3490E
```

For file naming conventions, please see the section on transmitting data to the server, below.

This Sheet Has Been Replaced



5.0 Specifying channel extender requirements

Customers who wish to operate surviving remote facilities using channel extension hardware may specify the configuration of those channel extenders by building a description file and transmitting that file to the Business Continuity and Recovery Services server as a separate file from the RDWS.

Use the transmission techniques that are described in the next section of this document. Send each channel extender definition as a separate file, following the prescribed naming conventions.

The content of the channel extender file is under development. The information below should be treated as preliminary until such time as a formal release is made.

*DEFCHX is a single line directive, one *DEFCHX record is required for every logically extended device. Each record consists of tag delimited operands, which appear below. Operands of the *DEFCHX directive are:

*APPADDR= xxxxxxx The apparent address of the device as attached to the customer system.

*APPCHPID= xx The apparent Channel Path identifier of the device as attached to the customer system.

*APPLNKID= xx The apparent link identifier of the device as attached to the customer system.

*APPTYPE= cccc The apparent device type of the device as attached to the customer system.

*APPMODEL= cccccccccccc The apparent model characteristics of the device as attached to the customer system.

*VENDOR= cccccccc The vendor name of the channel extender hardware, truncated to 8 characters.

*BOXID= cccccccc The identifier of the channel extender device itself. Often called the nodename of the extender.

*SLOTID= cccccccc The identifier of the card or slot within the box that is used to establish this connection. The data is optional, the tag is not.

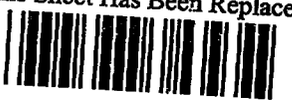
*PORTID= cccccccc The identifier of the port on the card or in the slot within the box that is used to establish this connection. The data is optional, the tag is not.

*LINEID= cccccccccccccccc The identifier of the line to connect to the remote channel extender This id may be a circuit designator, or an internal designator agreed upon by the customer and IBM.

In the following record description, optional data elements are identified with a plus-sign (+) before the field name. Optional elements may be defaulted to blanks (X'40'). Note that all tags (i.e., '*tagname= ') are considered **mandatory.**

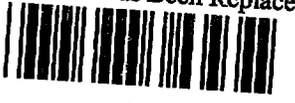
The layout of the record is as follows:

This Sheet Has Been Replaced



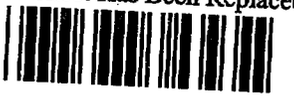
FIELD NAME	TYPE	DATATYPE/ LENGTH	VALUE	DESCRIPTION
RDEFCHNX	TAG	CL8	'*DEFCHX '	Constant
	PAD	CL1	' '	
	TAG	CL10	'*APPADDR= '	
RCXAADDR	VALUE	CL7		Apparent address of extended device 7 hexits, pad with leading zeroes.
	PAD	CL1	' '	
	TAG	CL11	'*APPCHPID= '	
RCXACHP	VALUE	CL2	' '	Apparent channel path identifier of extended device
	PAD	CL1	' '	
	TAG	CL11	'*APPLNKID= '	
+RCXALNK	VALUE	CL2	' '	Apparent link identifier of extended device Use only if ESCON extension needed
	PAD	CL1	' '	
	TAG	CL10	'*APPTYPE= '	
RCXATYPE	VALUE	CL5	' '	Type designator of Extended device
	PAD	CL1	' '	
	TAG	CL11	'*APPMODEL= '	
RCXAMODL	VALUE	CL8	' '	Model designator of Extended device
	PAD	CL1	' '	
	TAG	CL9	'*VENDOR= '	
RCXVEND	VALUE	CL8		Channel extender technology provider valid values are: CNT PARADYNE COMPUTERM ????????
	PAD	CL1	' '	
	TAG	CL9	'*BOXID= '	
+RCXIDENT	VALUE	CL8		Identifier or

This Sheet Has Been Replaced



				nodename of the extender being defined
	PAD	CL1	' '	
	TAG	CL9	'*SLOTID= '	
*RCXSLOT	VALUE	CL8		Physical slot attachment inside the given extender
	PAD	CL1	' '	
	TAG	CL9	'*PORTID= '	
*RCXPORT	VALUE	CL8		Physical port attachment inside the given extender
	PAD	CL1	' '	
	TAG	CL9	'*LINEID= '	
RCXLINE	VALUE	CL16		Communications line attachment for the given extender

This Sheet Has Been Replaced



Once the proper extender characteristics are defined, files have been identified, you will need to place a *HEADER directive as the first (comment) line in the channel extender defintion file before transmitting it to Business Continuity and Recovery Services.

***HEADER** is used to hold information on the source of the request, and its intended disposition. ***HEADER is a mandatory directive.** Fields in the *HEADER directive include:

CUST= Records the Business Continuity and Recovery Services customer number associated with this request.

SYSTEM= Records the common name of the system being processed. If the system is in actuality a part of a multi-image environment, this field should record the name of the overall processor complex.

CONTRACT= Records the contract number of the event being processed.

EDATE= Indicates the date associated with the event being processed. Note the format must be in Date(S) (sorted) format.

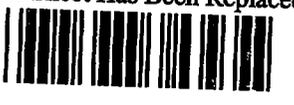
ETYPE= has two values: D is used for disaster recovery preparedness, while T is used to indicate that this definition is associated with a specific readiness test.

```
*HEADER CUST=01234567 SYSTEM=CHANEXT1 CONTRACT=Z123456 EDATE=19990229 ETYPE=T
```

Business Continuity and Recovery Services will process these files, and route the configuration information to the appropriate channel extender suppliers. IBM personnel will not validate or reconfigure channel extension facilites without customer and supplier participation.

For file naming conventions, please see the section on transmitting data to the server, below.

This Sheet Has Been Replaced



6.0 Transmission Protocols

All input must originate as an EBCDIC encoded file. That is, it must have originally been created on a Server/390 or equivalent architecture machine.

If your host system has TCP/IP connectivity, sending the files using the naming convention described below is the most direct method of transfer.

If your host system does not have direct TCP/IP connectivity, you may download the files to a system with FTP cabability and transmit them as noted below provided that they are moved as BINARY objects (not TEXT files).

For the purposes of the current implementation only, the only supported method for trasmitting data to Business Continuity and Recovery Services is the use of TCP/IP FTP protocol. The target FTP server is:

TESTCASE.BOULDER.IBM.COM

You will need to use a directory to hold your files. The directory name should be:

BRS\TOIBM\CustomerNumber\ContractNumber\

Use the MKDIR command to create the directory as specified.

The actual data files should be named as follows:

RDWS files CustomerNumb.ContractNumb.EventDate.Systemname.Flag.RDWS.BIN

NCP source definition files CustomerNumb.ContractNumb.EventDate.NCPname.Flag.NCP.BIN

Channel Extender source definition files
CustomerNumb.ContractNumb.EventDate.Systemname.Flag.CHX.BIN

Asset list shipping definition files CustomerNumb.ContractNumb.EventDate.Listname.Flag.TMS.BIN

Where

EventDate is in sorted date format (YYYYMMDD)

Flag is either 'T' indicating a scheduled test event or 'D' indicating an archive source for disaster recovery

For the sake of the following example, assume you have stored your configuration data on the VM Shared File System. Assume further that you have a customer number of 0001234, and a contract number of DMK9661.

An edited sample of the FTP session follows:

This Sheet Has Been Replaced



```
FTP TESTCASE.BOULDER.IBM.COM
   VM TCP/IP FTP V2R4
   Connecting to TESTCASE.BOULDER.IBM.COM, port 21
   220 - EZAFTSRV IBM VM V2R4 at TESTCASE.IBM.COM, 17:04:12 on 08/03/99
   220 Connection will close if idle for more than 5 minutes..
   USER (Identify yourslef to the host):
anonymous
   >>> USER ANONYMOUS
   331 Send password please.
email@server       (your email address)
   >>> PASS ********
   230 ANONYMOUS logged in;
   Command:
CD  /SOFTWARE/TOIBM/BRS/
   Command:
PUT localname 0001234.DMK966I.20000219.MYSYSTEM.T.RDWS.BIN
   Command:
PUT localname 0001234.DMK966I.20000219.MYNCP001.T.NCP.BIN
   Command:
PUT localname 0001234.DMK966I.20000219.MYNCP002.T.NCP.BIN
   Command:
PUT localname 0001234.DMK966I.20000219.MYSYSTEM.T.CHX.BIN
   Command:
PUT localname 0001234.DMK966I.20000219.MYLIST.T.TMS.BIN
   Command:
QUIT
   221 Goodbye.
```

Customers having security concerns about transmission of their data over public networks are encouraged to submit their resource requirement definitions on tape and to employ such carriers as will meet their requirements for appropriate secured transport. All such special arrangements need to be approved with the Business Continuity and Recovery Services account manager after consultation with technical support personnel.

This Sheet Has Been Replaced



APPENDIX C

This Sheet Has Been Replaced



Understanding and Utilizing Common Entity Definition Language/390

©2000 IBM Corporation

Global Services
Business Continuity and Recovery Services
System Support North America
Productivity Tools Development Team

This Sheet Has Been Replaced

Agenda

- Understanding Common Entity Definition Language/390
- Using CEDL/390 to control SCRAM/IEE processing
- Library and directory control
- Aiding customer migration to CEDL/390 and RDWS processing

This Sheet Has Been Replaced

Understanding Common Entity Definition Language/390

This Sheet Has Been Replaced

Understanding Common Entity Definition Language/390

- CEDL/390 was created to provide a consistent, complete representation of both physical and logical device characteristics
  - Physical Characteristics
    - Device types, models, features, etc.
  - Logical device characteristics
    - Emulation parameters
    - Optional VOLSER and logical recovery information

This Sheet Has Been Replaced

Understanding CEDL/390

- Expressed as a series of long variable length statements in a stylized format
- Always coded in EBCDIC character set
- Four basic types of statements in CEDL
  - Comments
  - Directives
  - Definitions
  - Auxiliary Statements

This Sheet Has Been Replaced

Understanding CEDL/390: Comments

This Sheet Has Been Replaced

Understanding CEDL/390

• Comments

– Begin with an initial asterisk (*), followed by one or more spaces

Freeform

– "Assembler" style

– Example:

```
* This entry is an example of a comment
```

This Sheet Has Been Replaced

Understanding CEDL/390

- Directives
  - Consist of a variable length statement beginning with an asterisk (*) followed by a reserved word:
    - *HEADER
    - *CINFO
    - *CNAME
    - *CHANEXTN
    - *CPROCESSOR
    - *CONSOLE
    - *ENVIRONMENT
    - *ESOTERIC
    - *FIXEDADD
    - *LPAR
    - *PROCESSOR
    - *NCPINFO
    - *SCPLPAR
    - *SCPTYPE
    - *ASSETLIST

This Sheet Has Been Replaced

Understanding CEDL/390: Directives

This Sheet Has Been Replaced

CEDL/390 Directives

- Directives are CEDL statements that begin with an asterisk-reserved word combination, with no intervening spaces, and are followed by one or more optional data values, which may or may not have explicit tags associated with them:

```
*RESERVEDWORD value *TAG=value2
```

- They are used to control processing and resource allocation

This Sheet Has Been Replaced

This Sheet Has Been Replaced

CEDL/390 Directives - *HEADER

*HEADER

- Used to identify the source of the request and the disposition of the source datasets
- *Mandatory directive*
- Must precede all other directives
- Must be included as first item in all related files

```
*HEADER CUST=nnnnnnnn SYSTEM=XXXXXXXX CONTRACT=XXXXXXXX EDATE=yyyymmdd EVENTTYPE=c
```

CEDL/390 Directives - *CINFO

- ***CINFO**
  - Used to identify a contact in the customer or submitting organization who is responsible for the construction and maintenance of the request
  - ***Mandatory directive***

```
*CINFO CONTACT=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa CPHONE=nnn-nnn-nnnn EMAIL= aaaaaaaa @ nnnnnnn.sss

*CINFO CONTACT=GREYHAME, GANDALF                 CPHONE=301-555-1212 EMAIL= Gandalf @ MyNode.Net
```


This Sheet Has Been Replaced

CEDL/390 Directives - *CINFO

– Fields in the *CINFO directive include:

- **CONTACT=xxxxxxxxxxxxxxxxxxxxxxxx**

  Records a name to contact at your location in case Business Continuity and Recovery Services personnel have questions about the content of the transmission. The contact name field is limited to 32 characters. Please specify name in the format: Lastname, Firstname

- **CPHONE=**

  Records a the telephone number associated with the contact name. Include any required country/area codes or prefixing information.

- **EMAIL=**

  Records an internet or intranet EMAIL address associated with the contract name.

This Sheet Has Been Replaced

CEDL/390 Directives - *CNAME

- *CNAME
  - Used to provide a descriptive name of the customer or system involved with this request.
  - Mandatory directive

```
*CNAME VGCC - Very Good Customer Corp (SYSTEMA)
```

  - 40 characters long, free form

This Sheet Has Been Replaced

CEDL/390 Directives -*CHANEXTN

- *CHANEXTN
  - Instructs Business Continuity and Recovery Services that recovery requires specially defined channel extender devices on the target system.
  - Valid values are:
    - Y - Channel extender definition information is being supplied in a separate file.
    - N - No channel extenders are present on this system.
  - If *CHANEXTN Y is specified, the CHXSOURCE operand indicates the name of the file containing the channel extender definitions.

```
*CHANEXTN Y <*CHXSOURCE=nnnnnnnn>
*CHANEXTN N
```


This Sheet Has Been Replaced

CEDL/390 Directives -*CONSOLE

*CONSOLE

- Instructs Business Continuity and Recovery Services that the following addresses (RDEVS,UCBS) are to be reserved as system console address.

Code one *CONSOLE directive for each device you want assigned to the console device class

****CONSOLE should be used ONLY if you are running in a native or LPAR recovery environment. Users recovering their systems in a configuration where VM is in control of the real hardware may omit this statement.***

This Sheet Has Been Replaced

CEDL/390 Directives -*CONSOLE

```
*CONSOLE xxxxxxx ttttttttt mmmmmmmm ccccccccc
```

- xxxxxxx is the hexadecimal device address (7 hexits)
- ttttttttt is the device type
- mmmmmmmm is the device model
- ccccccccc is the console class short name

```
- *CONSOLE  0000009     3278 3          MSTRCONS
- *CONSOLE  0000609     3278 3          ALTCONS
```

This Sheet Has Been Replaced

CEDL/390 Directives - *ENVIRONMENT

- ***ENVIRONMENT**
  - Instructs Business Continuity and Recovery Services that the recovery or test is to be managed using one of three possible environments below. Consult with your account manager or project manager if you need to code options N or L, below.

    *ENVIRONMENT N requests resource allocation onto a native machine (ie., one with NO multi-image manager active).

    *ENVIRONMENT L requests resource allocation onto an LPAR-only multi-image managedsystem.

    *ENVIRONMENT V instructs that any recovery performed from these definitions is permitted in a virtual machine environment.

    ****ENVIRONMNENT V is the default.***

This Sheet Has Been Replaced

CEDL/390 Directives -*ESOTERIC

- ***ESOTERIC**
  - **Instructs Business Continuity and Recovery Services that the following addresses (RDEVS,UCBS) are to be reserved to a specific MVS esoteric device identifier.**
  - ***Use the directive only if you are configuring a worksheet for a processor that will be recovered natively (or in a customer-defined LPAR).***
  - **Code one *ESOTERIC directive for each device you want assigned to a specific esoteric device class.**

This Sheet Has Been Replaced

CEDL/390 Directives -*ESOTERIC

```
- *ESOTERIC xxxxxxx ttttttt mmmmmmm eeeeeeee
                xxxxxxx  is the hexadecimal device address (7 hexits)
                ttttttt is the device type
                mmmmmmm is the device model
                eeeeeeee is the esoteric short name

- *ESOTERIC 0000180 3490 E        349E
- *ESOTERIC 0001181 3422 1        9TRK
```


This Sheet Has Been Replaced

CEDL/390 Directives - *FIXEDADD

- ***FIXEDADD**
  - Instructs Business Continuity and Recovery Services that the following addresses (RDEVS,UCBS) must be mapped to identical addresses on the target system. This is used when defining MCS consoles, or DASD that requires a static location. *CONSOLE and *ESOTERIC directives are mapped into *FIXEDADD statements internally.

```
- *FIXEDADD xxxxxxx tttt Comment

- *FIXEDADD 0000009 3278 MCS Console

- *FIXEDADD 0001F00 3330 DASD with strange I/O Protocol
```

This Sheet Has Been Replaced

CEDL/390 Directives - *LPAR

- ***LPAR**
- **The *LPAR directive (along with the *PROCESSOR directive) is used to express requirements for processor resources on the target configuration.**
- ***Use the *LPAR directive <u>only</u> if *ENVIRONMENT L or *ENVIRONMENT N is coded***

This Sheet Has Been Replaced

CEDL/390 Directives - *LPAR

*LPAR *LPN=nnnnnnnn *LPID=oo *NP=pp *CS=cccc *ES=eeeee *WT=www *DED=f *ADMOPT=a

- **Where**
  - nnnnnnnn = the name of the partition in question
  - oo = the partition identifier number
  - pp = is the number of logical processors defined to this LPAR
  - cccc = is the quantity of configured central storage, in MegaBytes
  - eeeee = is the quantity of configured expanded storage, in MB
  - www = is the LPAR weighting factor for this partition
  - f = flag indicating whether the LPAR uses dedicated or shared processors (D or S)
  - a = flag indicating whether or not the LPAR uses the ADM option (Y=yes, N=no)

This Sheet Has Been Replaced

CEDL/390 Directives -*PROCESSOR

- ***PROCESSOR**
- **The *PROCESSOR directive (along with the *LPAR directive) is used to express requirements for processor resources on the target configuration.**
- ****PROCESSOR is a mandatory directive.***
- ***PROCESSOR *TYPE=xxxx *MDL=yyy *NPROC=pp *SN=sssssss *CS=cccc *ES=eeeee**
- **Where**
- **xxxx = is the device model of the processor (e.g., 9672)**
- **yyy = is the model number of the processor (e.g., RX4)**
- **pp = is the number of processors active on the machine.**
- **sssssss = is the serial number of the machine**
  If you have NO serial number dependent software, you may code ANY in this field.
- **cccc = is the quantity of configured central storage, in MegaBytes (MB).**
- **eeeee = is the quantity of configured expanded storage, in MegaBytes (MB).**

This Sheet Has Been Replaced

CEDL/390 Directives - *NCPINFO

- ***NCPINFO**
- **The *NCPINFO directive indicates whether or not an NCP source deck will be transmitted for mapping onto Business Continuity and Recovery Services supplied hardware. *NCPINFO is a mandatory directive. Valid values are:**
  - **Y - NCP generation information is being supplied in a separate file**
  - **N - No NCP generation information is being supplied**
- *NCPINFO x <*NCPLEVEL=nnnnnnnnnn *SSPLEVEL=nnnnnnnn *NCPSOURCE= cccccccc

This Sheet Has Been Replaced

CEDL/390 Directives - *NCPINFO

- *If *NCPINFO Y is specified, the *NCPLEVEL= and *SSPLEVEL= operands indicate the level of the libraries used to generate the NCP and SSP code.*
- *If the level used is not available at the recovery site, or cannot be provided by IBM Business Continuity and Recovery Services personnel, a process exception will be generated, and the customer will be notified to make arrangements to port or archive key libraries to manage the network specification process.*
- *Code one *NCPINFO directive for each NCP image to be transmitted*

```
*NCPINFO  Y  *NCPLEVEL= 7.8          *SSPLEVEL=4.8  *NCPSOURCE= NCPNAME1
*NCPINFO  N
```


This Sheet Has Been Replaced

CEDL/390 Directives -*SCPLPAR

- ***SCPLPAR**
  - **Instructs Business Continuity and Recovery Services that the system control program to be activated in LPAR nnnnnnn is defined by the literal xxxxxxxx, at version/release level vvvvvvvv with a PUT/service update level of yyyy.**
  - ****SCPLPAR is functionally equivalent to *SCPTYPE, but has an additional (first) operand of an LPAR name or identifier.***
  - ****SCPLPAR is a required directive in an LPAR environment.***
  - ***Use of the *SCPLPAR directive will not directly affect resource allocation processing, but is reserved for informational use.***

This Sheet Has Been Replaced

CEDL/390 Directives -*SCPLPAR

- **Valid *SCPLPAR values for this release level are:**

```
*SCPLPAR ..... MVS/ESA
*SCPLPAR ..... OS/390
*SCPLPAR ..... VM/ESA

*SCPLPAR  LPAROLD1 MVS/ESA V5R2L2 9802 JES2
*SCPLPAR  LPARPROD OS/390  R2.5   9901 JES2
*SCPLPAR  LPARTEST OS/390  R2.7   9902 JES3
*SCPLPAR  LPARSLOW VM/ESA  V2R4.0 9904 CMS15
```


This Sheet Has Been Replaced

CEDL/390 Directives -*SCPTYPE

- ***SCPTYPE xxxxxxxx vvvvvvvv yyyy Comment**
  - **Instructs Business Continuity and Recovery Services that the primary system control program is defined by the literal xxxxxxxx, at version/release level vvvvvvvv with a PUT/service update level of yyyy.**
  - ***Use of the *SCPTYPE directive will not directly affect resource allocation processing, but is reserved for informational use.***
- Valid *SCPTYPE values for this release level are:
  - *SCPTYPE MVS/ESA
  - *SCPTYPE OS/390
  - *SCPTYPE VM/ESA
- Examples:
- *SCPTYPE MVS/ESA V5R2L2 9802 JES2
- *SCPTYPE OS/390 R2.5 9901 JES2
- *SCPTYPE OS/390 R2.7 9902 JES3
- *SCPTYPE VM/ESA V2R4.0 9904 CMS15

This Sheet Has Been Replaced

CEDL/390 Directives -*ASSETLIST

- ***ASSETLIST c <*LISTSOURCE= nnnnnnnn>**
  - **Instructs Business Continuity and Recovery Services that the a list of assets (tapes, customer supplied equipment, etc.,) accompanying the event is to be defined for audit purposes in the file named nnnnnnn.**
  - **'c' is either Y or N.**
  - **Use of the *ASSETLIST directive will not directly affect resource allocation processing, but is reserved for informational use.**
  - ***Code one *ASSETLIST directive for each recovery event.***

- Examples:
- *ASSETLIST N
- *ASSETLIST Y *LISTSOURCE= PICKLIST

This Sheet Has Been Replaced

CEDL/390 Definitions

This Sheet Has Been Replaced

CEDL/390 Definitions

- The ***DEFINE** statement in CEDL/390 is a long fixed length record in a tag-and-value format that provides information about the physical and logical characteristics of a single specific device.
- The specific characteristics of the fields are documented in the customer guide
- We will examine each of the tags in turn
- *NOTE: record alignment (spacing) is critical, as the programming filters depend on a fixed-format record.*

This Sheet Has Been Replaced

CEDL/390 Definitions - *DEFINE Type/Address

- *DEFINE
  - *DEFINE begins the CEDL/390 definition of a specific physical unit with its associated logical characteristics.
  - *DEFINE has a two part data value immediately after the *DEFINE constant and before any other tags in the record:
    - 8 characters, left aligned, of the device type (e.g.3390, 3990, CFS, CFR, CTCA, 3745, etc)
    - a space
    - 7 hexadecimal digits, right aligned, and padded with zeroes, indicating the device address.

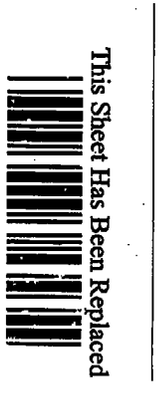
```
*DEFINE  3390     00009F2
0---+---1----+----2----+----3----
```


This Sheet Has Been Replaced

CEDL/390 Definitions - *CNTLUNIT

- ***CNTLUNIT=**
- **Defines the 7-hexit device address of the control unit to which the device is attached.**
- **For devices which act as their own control units, (e.g., 3850, 1403), code the unit address here as well.**
- ***This is a mandatory data field.***

```
*DEFINE 3390      0009F02 *CNTLUNIT= 00009F00
*DEFINE 3850      0000300 *CNTLUNIT= 00000300
```

CEDL/390 Definitions - *MODEL

- *MODEL=
  - Indicates the model type of this unit.
  - 8 character data value
  - Some device types have default models assigned, so that a null value or NONE creates (internally) a device with the default characteritic.

```
3390 = 3390 -   3
3380 = 3380 -   K
3480 = 3480 -   E
3490 = 3490 -   E
3278 = 3278 -   2
3179 = 3179 - S3G
3472 = 3472 -  3G
```

```
*DEFINE 3390      0009F02 *CUATTACH= 00009F00 MODEL= 9
```

This Sheet Has Been Replaced

CEDL/390 Definitions -*FEATURE

- ***FEATURE=**
  - 40 character value indicating feature codes important to generate this device.
  - Default is NONE (blanks)
  - ***This is an optional item***
  - ***Note: For control units equipped with the EMC™ TimeFinder™ feature, code TIMEFINDER(v.r.l), where v.r.l are the version, release, and level of the host TimeFinder™ software***

```
*DEFINE 3390 0009F02 *CNTLUNIT=0009F00 *MODEL=03          *FEATURE= NONE

*DEFINE 5550 0009F00 *CNTLUNIT=0009F00 *MODEL= 1          *FEATURE= TIMEFINDER(3.1.2)
```


This Sheet Has Been Replaced

CEDL/390 Definitions - *DYNAMIC

- ***DYNAMIC**
  - **Is dynamic allocation permitted for this device?**
  - **Default is "NO"**
  - **Values are "YES" or "NO"**
  - ***Optional item***

- **Example:**
- `*DEFINE 3390 ... *FEATURE=NONE *DYNAMIC=YES`

This Sheet Has Been Replaced

CEDL/390 Definitions - *LPARS

- ***LPARS=**
  - **If the host system is partitioned, does this device connect to more than one partition? If yes, enter the partition names as a series of up to 15 eight character tokens, separated by blanks.**
  - **Code blanks if none**
  - ***Up to 15 LPAR names may be coded. Support limitations on the number of LPARS definable on a given processor type/model may apply.***
  - ***Mandatory item only if *ENVIRONMENT L***
  - **Example:**

```
*DEFINE ...*LPARS= LPAR0001 LPARTEST LPARPROD LPARSLOW...
```


This Sheet Has Been Replaced

CEDL/390 Definitions - *SYSNAME

- ***SYSNAME=**
  - **Name of system or complex if multi-image managed**
  - **Default is blank (none)**
  - ***Optional item***
- **Example:**
  - `*DEFINE 3990 ... *LPARS=... *SYSNAME=HOMENODE ...`

This Sheet Has Been Replaced

CEDL/390 Definitions - *PATH

- ***PATH=**
  - **10 x 3 Character fields (2 hexits and a blank as separator). To which paths does this device attach?**
  - ***Optional field, may be defaulted to blanks***
- **Example:**
- `*DEFINE 3390 … *PATH= 01 02 03 06 08 1A 1B 1C 1D 4F …`

This Sheet Has Been Replaced

CEDL/390 Definitions - *LINK

- ***LINK=**
  - **10 x 3 Character fields (2 hexits and a blank as separator). To which links does this device attach?**
  - ***Optional field, may be defaulted to blanks***
- **Example:**
- `*DEFINE 3390 ... *LINK= 01 02 03 06 08 1A 1B 1C 1D 4F ...`



This Sheet Has Been Replaced

CEDL/390 Definitions -*CONNECT

- ***CONNECT=**
  - **Does this device communicate with another image or system?**
  - **If yes, code *CONNECT= with a two-part data value:**
  - ***Optional data value, default is blank (no)***

- TARGETSYS CL9 Name of other image or system (Left align)
- TARGETADR CL7 Address on other image to which this device attaches (right aligned, padded with zeroes)

Examples:

```
*DEFINE CTCA ... *CONNECT= OTHERSYS 0000250 ...
*DEFINE 3088 ... *CONNECT= PRIMESYS 0000260 ...
```


This Sheet Has Been Replaced

CEDL/390 Definitions - *LABEL

- ***LABEL=**
  - **Volume serial identifier for Direct Access Storage Devices (DASD)**
  - **Code this value ONLY if the customer wants to pre-label/pre-allocate their recovery DASD**
  - ***Optional data value***
  - ***Default is blanks***

- **Example:**
- `*DEFINE 3390 , 0000407 ... *LABEL= MYPACK ...`

This Sheet Has Been Replaced

CEDL/390 Definitions - *SMS

- ***SMS=**
  - **Is this device a participant in OS/390 Systems Managed Storage?**
  - ***This tag is optional. Values may be omitted. Blank defaults to 'N'***
  - ***You do not need to code the tag if the customer will be performing recovery and initializing their own DASD.***

- **Example:**
- `*DEFINE 3390 0000407 *SMS= N ...`

This Sheet Has Been Replaced

CEDL/390 Definitions - *SG

- ***SG=**
  - **Name of the SMS Storage Pool to which this device belongs, if any**
  - ***Optional tag***
  - ***Do not code a value if *SMS=N or where *SMS= blank***

- **Examples:**
- `*DEFINE 3390 0000407 *SMS=Y *SG= FASTPACK ...`
- `*DEFINE 3390 0000408 - *SMS=N *SG=`

This Sheet Has Been Replaced

CEDL/390 Definitions - *VTOC

- ***VTOC=**
  - **Used to specify if VTOC initialization parameters are required.**
  - ***Code this value ONLY if the customer requests the IBM pre-allocate or recover their volumes.***
  - **The first 16 characters are permitted here.**
  - **For example, a ICKDSF VTOC(0,1,15) statement would be entered as *VTOC= (0,1,15)**

This Sheet Has Been Replaced

CEDL/390 Definitions - *INDEX

- ***INDEX=**
  - **Used to specify if INDEXed VTOC initialization parameters are required.**
  - ***Code this value ONLY if the customer requests the IBM pre-allocate or recover their volumes.***
  - **The first 16 characters are permitted here.**
  - **For example, a ICKDSF INDEX(0,2,90) statement would be entered as *INDEX=(0,2,90)**

This Sheet Has Been Replaced

CEDL/390 Definitions - *VVDS

- ***VVDS=**
  - **Used to specify if VVDS (VSAM) initialization parameters are required.**
  - ***Code this value ONLY if the customer requests the IBM pre-allocate or recover their volumes.***
  - **The first 16 characters are permitted here.**
  - **For example, a ICKDSF VVDS(0,2,45) statement would be entered as *VVDS= (0,2,45)**

This Sheet Has Been Replaced

CEDL/390 Definitions - *RTECH

- ***RTECH=**
  - **Two part data field.**
    - **1 Character Flag indicates formatting choice**
    - **14 character field indicates any special recovery utility required.**
  - ***Code this value ONLY if the customer requests the IBM pre-allocate or recover their volumes.***

This Sheet Has Been Replaced

CEDL/390 Definitions - *RTECH

- ***RTECH=**
  - **Flag values and their meaning**
    - B = EMC™ TimeFinder™ Business Continuity Volume
    - C = Restore VM volume in CP format
    - D = Dataset level restore
    - F = Full volume restore
    - I = Restore Index VTOC volume
    - M = Restore Minidisk Volume (ICKDSF for a full pack minidisk)
    - O = Restore OS volume (perform ICKDSF for "native" volume)
    - P = Primary unit of duplex/mirrored pair
    - S = Secondary unit of duplex/mirrored pair
    - V = Restore VVDS volume

This Sheet Has Been Replaced

CEDL/390 Definitions - *RTECH

- ***RTECH=**
  - **Field 2 (RECVTOOL)**
    - **If the customer requires a special utility to restore this volume, indicate here and flag for processing exception later.**
      - **Examples: VMBSAR, DDR, IDCAMS, etc.**

This Sheet Has Been Replaced

CEDL/390 Definitions - *EMUNIT

- ***EMUNIT=**
  - **One character flag indicating whether or not the device specified is in fact emulated on another real device type**
  - **Values: Y or N (Y=emulated, N=not)**
  - ***Default is: N, blank is read as "N"***
  - **Example:**
    - `*DEFINE 3390 0000502 … *EMUNIT=Y …`

This Sheet Has Been Replaced

CEDL/390 Definitions - *RDMFR

- ***RDMFR=**
  - **Manufacturer of real device on which emulated device is mapped.**
  - **Length 8 Characters**
  - ***Do not code if *EMUNIT=N***
  - **Example:**
    - *DEFINE 3390 0000502 … *EMUNIT=Y *RDMFR= EMC
      …

This Sheet Has Been Replaced

CEDL/390 Definitions - *RDTYPE

- ***RDTYPE=**
  - **Type of the real device on which emulated device is mapped.**
  - **Length 10 Characters**
  - ***Do not code if *EMUNIT=N***

Example:

```
*DEFINE 3390    0000502 … *EMUNIT=Y … *RDTYPE= 5550 …
```


This Sheet Has Been Replaced

CEDL/390 Definitions - *RDMODL

- *RDMODL=
  - Model of the real device on which emulated device is mapped.
  - Length 10 Characters
  - *Do not code if *EMUNIT=N*
  - Example:

```
*DEFINE 3390 0000502 .. *EMUNIT=Y .. *RDTYPE= 5550 *RDMODEL= 2 ..
```

This Sheet Has Been Replaced

CEDL/390 Definitions - *RDUNIT

- ***RDUNIT=**
  - **Real Device allocation UNIT**
  - **1-Character Flag**
  - **Allocation unit type for this emulated device.**
  - **Choose one of:**
    - **'F' = FBA Blocks**
    - **'C' = CKD Cylinders**
- ***Do NOT code if *EMUNIT= N***

This Sheet Has Been Replaced

CEDL/390 Definitions - *RDALOC

- *RDALOC=
- Count of usable allocation units on this device expressed as a nine-digit integer count in terms of RDUNITs
- *Do NOT code if *EMUNIT= N*

This Sheet Has Been Replaced

CEDL/390 Auxiliary Statements

This Sheet Has Been Replaced

CEDL/390 Auxiliary Statements

- Auxiliary Statements are expressions in CEDL/390 that are used to control ancillary processes, such as:
  - Channel Extender Definitions
  - Asset Transfer Control Lists
- They provide a natural extension to the CEDL/390 architecture that allows it to be extended to control resource processing beyond the immediate hardware and software involved in system mapping

This Sheet Has Been Replaced

CEDL/390 Auxiliaries - *DEFCHX

- *DEFCHX
  - A CEDL/390 Auxiliary Statement in a separate file of type CHX
  - *Optional*
  - *Use ONLY if you have a requirement to define an extended channel set*

This Sheet Has Been Replaced

This Sheet Has Been Replaced

CEDL/390 Auxiliaries - *DEFCHX

- *DEFCHX
  - A CEDL/390 Auxiliary Statement in a separate file of type CHX
  - ***Optional***
  - *Use ONLY if you have a requirement to define an extended channel set*

CEDL/390 Auxiliaries - *DEFCHX

- *DEFCHX tags:
  - *APPADDR= apparent address of the extended device (7)
  - *APPCHPID= apparent channel path of the extended device (2)
  - *APPLNKID= apparent link identifier of the extended device (2)
  - *APPTYPE= apparent device type (5)
  - *APPMODEL= apparent model characteristics (8)
  - *VENDOR= vendor name of the extender hardware (8)
  - *BOXID= identifier of the extender itself. Often called 'nodename' (8)
  - *SLOTID= identifier of the slot within the box used for this line or circuit (8)
  - *PORTID= identifier of the port in the slot used for this line or circuit (8)
  - *LINEID= line or circuit designator (16)
- *Numbers in parentheses are data field lengths*

This Sheet Has Been Replaced

CEDL/390 Auxiliaries - *ASSET

- *ASSET
  - A CEDL/390 Auxiliary Statement in a separate file of type TMS (Temporary Management Service)
  - *Optional*
  - *Use ONLY if you have a requirement to manage and inventory customer assets*

This Sheet Has Been Replaced

CEDL/390 Auxiliaries - *ASSET

- *ASSET CLASS IDENTIFIER subTYPE Descriptor
  - Class (8 alphanumeric)
    - CABLE CONNECTR DEVICE, MODEM, LIBRARY, TAPE
      - (TAPE refers to both reel and cartridge tapes)
  - Identifier (8 alphanumeric)
    - An external identifier for this asset. For example, in the case of a TAPE asset, specify the external volume serial identifier (VOLSER)
  - subTYPE (8 alphanumeric)
    - Additional reference information about the device. For example, in the case of TAPE, code one of (3422|3480E|3490|3490E|3590)
  - Descriptor (32 alphanumeric)
    - 32 characters of additional descriptive information

This Sheet Has Been Replaced

Using CEDL/390 to Control SCRAM/IEE Processing

This Sheet Has Been Replaced

Using CEDL/390 to Control SCRAM/IEE Processing

- When analyzing a customer request (RDWS), *CONSOLE and *ESOTERIC directives will be mapped into *FIXEDADD directives and mapping will proceed
- When you want to reserve an address range on the target system, SCRAMIEE EDIT the DEVPATHS file and delete the reserved addresses prior to running the MAP step. This removes the devices from consideration for allocation.

This Sheet Has Been Replaced

Library Control

This Sheet Has Been Replaced

Library Control

- Single server is on WSCVM
- Filepool is WSCVMU:
- High level directory is EBRS1.
- Customer data resides in .CDREPOS directories
- Program code resides in .SCRAM.xxxxxLIB
  (DOCLIB, TESTLIB, PRODLIB,SAMPLIB)
- Control information is in: .SCRAM.CONTROL

This Sheet Has Been Replaced

Aiding Customer Migration to CEDL/390

This Sheet Has Been Replaced

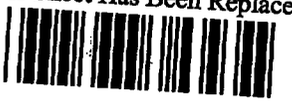
Aiding Customer Migration to CEDL/390

- Migration aids are planned
  - Analyze IOCP (not IODF) and convert device type, model and address
  - Prototype aid under joint development
  - Does not address VOLSER and other logical information
  - Not anytime soon

This Sheet Has Been Replaced

APPENDIX D

This Sheet Has Been Replaced



pilation successful

piler Options

```
  NOALTERNATE
    CEXEC     (AMFSGT   CEXEC    B1)                    RECFM=F,LRECL=1024
  NOCOMPILE   (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG      (I)
  NOIEXEC
    LIBLEVEL  (*)
    LINECOUNT (55)
    MARGINS   (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT     (AMFSGT   LISTING  B1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN     (AMFSGT   EXEC     B1 Y ┬ Y M Y ┬ Y ) 0A \  ) RECFM=V,LRECL=114
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 3
```

149

This Sheet Has Been Replaced

```
42  /* ------------------------------------------------------------ */
43  MainPgm:  Nop;
44            Rcx = 0
45            Call ProcessProfile
46            Call SetupAccess
47            Call GetWorkToDo
48            Call IdentifyFiles
49            Call GetActualFiles
50            Select
51                When Argstring = "(CLEANSERVER" then call CheckFileAndKillOutsideCopyIfOK
```


This Sheet Has Been Replaced

```
Do Sel Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
   1  /* ------------------------------------------------------------------ */
   2  /*  AMFSGT - SCRAM GeTfiles                                           */
   3  /*  (c) 2000 IBM Corporation All Rights Reserved.                     */
   4  /*  Patent Application Pending                                        */
   5  /* ------------------------------------------------------------------ */
   6  /* ================================================================== */
   7  /* Usage: AMFSGT (option                                              */
   8  /*        Where option may be:                                        */
   9  /*                                                                    */
  10  /*        CLEANSERVER = used to delete files from the outside-the-    */
  11  /*                      firewall-server once it's established they    */
  12  /*                      have been transferred to the local system     */
  13  /*                      and have a readable header.                   */
  14  /*                                                                    */
  15  /* NOTE: To set up the program, you will need to establish the highest*/
  16  /*       Level directory on your local server.                        */
  17  /*       This is the value in AMFSGT PROFILE under the keyword        */
  18  /*       BaseDirectory = 'x'                                          */
  19  /*       This can be done by using the CMS command:                   */
  20  /*           CREATE DIRECTORY 'x'                                     */
  21  /*                                                                    */
  22  /* ================================================================== */
  23  /* Change History                                                     */
  24  /* YYYYMMDD WHO CHANGE                                                */
  25  /* -------- --- ------------------------------------------------------*/
  26  /* 20000112 AMF Initial Release                                       */
  27  /* 20000214 AMF Notification to userid list in AMFSGT PROFILE         */
  28  /*                                                                    */
  29  /* ------------------------------------------------------------------ */
  30             Address CMS;
  31             Trace Off;
  32             Parse Upper Arg Argstring.
  33  /* ------------------------------------------------------------------ */
  34  Literals: Nop;
  35            InFileID = "FTP DIROUTP *"
  36            ContractMasterFileID  = "CONTRACT TABLE *"
  37            WorkMode = "W"
  38            ProfileFileID = "AMFSGT PROFILE *"
  39  LiteralsEnd: Nop;
  40  /* ------------------------------------------------------------------ */
  41  /*                                                                    */
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
      1     52                Otherwise nop;
            53             End;
            54             Call ExitRtn Rcx
            55    MainPgmEnd: Exit
            56    /* ---------------------------------------------------------------- */
            57    /*                                                                  */
            58    /* ---------------------------------------------------------------- */
            59    ExitRtn:    Parse Arg RCX
            60               'CP SPOOL CONS CLOSE STOP'
            61                Exit RCX
            62    ExitRtnEnd: Return;
            63    /* ---------------------------------------------------------------- */
            64    /*                                                                  */
            65    /* ---------------------------------------------------------------- */
            66    SetupAccess: Nop;
            67               'CP LINK TCPMAINT 592 592 RR'
            68               'ACC 592 R'
            69               'CP SPOOL CONS * CLASS L START'
            70               'CP TERM MORE 1 1'
            71    SetupAccessEnd: Return;
            72    /* ---------------------------------------------------------------- */
            73    /*                                                                  */
            74    /* ---------------------------------------------------------------- */
            75    GetWorkToDo: Nop;
            76               "MAKEBUF"
            77               Bufferid = rc
            78               QUEUE 'testcase files4me'
            79               QUEUE 'quote site testcase.boulder.ibm.com'
            80               QUEUE 'user ebrs1 z9tx3q8'
            81               QUEUE 'cd /software/toibm/brs'
            82               QUEUE 'dir *.* (disk'
            83               QUEUE 'quit'
            84
            85               FTP 'sng-bb2.boulder.ibm.com'
            86               Rcs = rc
            87               "CP SLEEP 02 SEC"
            88               "DROPBUF" Bufferid
            89               "PIPE DISK FTP DIROUTP *" ,
            90                    "| COUNT LINES" ,
            91                    "| VAR INQUESZ"
            92               Say Date(S) time() "AMFSGT0041" INQUESZ "files for transfer"
```


This Sheet Has Been Replaced

```
94  /* ---------------------------------------------------------------- */
95  /*                                                                  */
96  /* ---------------------------------------------------------------- */
97  GetActualFiles: Nop;
98              "MAKEBUF"
99              Bufferid = rc
100             QUEUE 'testcase files4me'
101             QUEUE 'quote site testcase.boulder.ibm.com'
102             QUEUE 'user ebrsl z9tx3q8'
```

This Sheet Has Been Replaced

```
Do Sel Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          103             QUEUE 'cd /software/toibm/brs'
          104             "PIPE DISK WORK QUEUE A" ,
          105                  "| STACK"
          106             QUEUE 'quit'
          107
          108             FTP 'sng-bb2.boulder.ibm.com'
          109
          110             Rcs = rc
          111             Say "Waiting....."
          112             "CP SLEEP 03 SEC"
          113             "DROPBUF" Bufferid
          114             Say Date(S) Time() ,
          115                 "AMFSGT005I" WorkQueue.0/3 "files selected on TESTCASE."
          116             Select
      1   117                 When RCS = 0 then nop;
      1   118                 Otherwise do
  1   2   119                     Say Date(S) time() "AMFSGT012T Error '",
  1   2   120                         || RCs ||"' during file transfer."
  1   2   121                     Say Date(S) time() "AMFSGT012I Ending."
  1   2   122                     Call ExitRtn Rcs
      2   123                     end;
          124             End;
          125   GetActualFilesEnd: Return;
          126   /* ---------------------------------------------------------- */
          127   /*                                                            */
          128   /* ---------------------------------------------------------- */
          129   IdentifyFiles: Nop;
          130             QElement = 1
          131             WQElement = 1
          132            "PIPE DISK" InFileID ,
          133                "| SORT 42.3 DESCENDING 46.2 DESCENDING " ,
          134                "        49.2 ASCENDING 51.2 ASCENDING" ,
          135                "| STEM TestCaseFiles."
          136
          137             "PIPE DISK" ContractMasterFileID  ,
          138                "| DROP FIRST 1" ,
          139                "| STEM ValidContracts."
          140
          141             FilesToTransfer = TestCaseFiles.0
          142
          143             Do Xfer = 1 to FilesToTransfer
```


This Sheet Has Been Replaced

```
1       144     Parse Var TestCaseFiles.Xfer ,
1       145           . . . . . Month.Xfer Day.Xfer Hour.Xfer ":" ,
1       146          Minute.Xfer TestCaseFileID.Xfer
1       147     Parse Upper Var TestCaseFileID.Xfer ,
1       148               TCCustomer.Xfer ,
1       149             "." TCContract.Xfer ,
1       150             "." TCEDate.Xfer ,
1       151             "." TCFileID.Xfer ,
1       152             "." TCFlag.Xfer ,
1       153             "." TCType.Xfer ,
```

This Sheet Has Been Replaced

```
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
  1        154                    ".BIN"
  1        155
  1        156                TCFileIDCheck.Xfer = TCCustomer.Xfer ,
  1        157                                     TCContract.Xfer ,
  1        158                                     TCEDate.Xfer ,
  1        159                                     TCFileID.Xfer ,
  1        160                                     TCFlag.Xfer ,
  1        161                                     TCType.Xfer
  1        162
  1        163                Select
  1    1   164                    When (Words(TCFileIDCheck.Xfer)<>6) then do
  2    2   165                          Say Date(S) Time() "AMFSGT024E" ,
  2    2   166                              "Invalid transfer file:'" ,
  2    2   167                              ||TCCustomer.Xfer||"' requesting service."
  2    2   168                          Iterate XFer
  1    2   169                          end;
  1    1   170                    When ((Word(TCFileIDCheck.Xfer,5)<>'D')  & (Word(TCFileIDCheck.Xfer,5)<>'T') ) the
                 C n do
  2    2   171                          Say Date(S) Time() "AMFSGT024E" ,
  2    2   172                              "Invalid transfer file:'" ,
  2    2   173                              ||TCCustomer.Xfer||"' Type='"||TCType.Xfer||"'"
  2    2   174                          Iterate XFer
  1    2   175                          end;
  1    1   176                    When (Datatype(Word(TCFileIDCheck.Xfer,3),'W')<>1) then do
  2    2   177                          Say Date(S) Time() "AMFSGT025E" ,
  2    2   178                              "Invalid transfer file:'" ,
  2    2   179                              ||TCCustomer.Xfer||"' Date='"||TCEDate.Xfer||"'"
  2    2   180                          Iterate XFer
  1    2   181                          end;
  1    1   182                     Otherwise nop;
  1        183                End;
  1        184
  1        185
  1        186                "PIPE STEM ValidContracts." ,
  1        187                     "| LOCATE 1.8 /"||TCCustomer.Xfer||"/" ,
  1        188                     "| VAR FoundCustomer.Xfer"
  1        189
  1        190                "PIPE STEM ValidContracts." ,
  1        191                     "| LOCATE 10.8 /"||TCContract.Xfer||"/" ,
  1        192                     "| VAR FoundContract.Xfer"
  1        193
```

156

This Sheet Has Been Replaced

```
1       1.,         Select
1   1   195             When Left(FoundCustomer.Xfer,14) = "FOUNDCUSTOMER." then do
2   2   196                 Say Date(S) Time() "AMFSGT006E",
2   2   197                     "Invalid customer number: '",
2   2   198                     ||TCCustomer.Xfer||"' requesting service."
2   2   199                 Iterate XFer
1   2   200                 end;
1   1   201             Otherwise nop;
1       202         End;
1       203         Select
```


This Sheet Has Been Replaced

```
▸ Source Listing                                AMFSGT   EXEC     #1
Compiler REXX/370 3.0  LVL PQ00112    Time: 11:51:37                Date: 2000-02-16                Page:    6
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0

 1   1    204                    When Left(FoundContract.Xfer,14) = "FOUNDCONTRACT." then do
 2   2    205                         Say Date(S) Time() "AMFSGT008E" ,
 2   2    206                             "Invalid contract number: '" ,
 2   2    207                             ||TCContract.Xfer ||"' requesting service" ,
 2   2    208                             "for customer:" TCCustomer.Xfer
 2   2    209                         Iterate XFer
 1   2    210                         end;
 1   1    211                    Otherwise nop;
 1        212                End;
 1        213            SFSCustomer.Xfer      = BaseDirectory||TCCustomer.Xfer||"."
 1        214            SFSContract.Xfer      = SFSCustomer.Xfer||TCContract.Xfer||"."
 1        215            SFSType.Xfer          = SFSContract.Xfer||TCFlag.Xfer||"."
 1        216            SFSFileSet.Xfer       = SFSType.Xfer||TCEDate.Xfer||"."
 1        217
 1        218            Upper SFSCustomer.Xfer
 1        219            Upper SFSContract.Xfer
 1        220            Upper SFSType.Xfer
 1        221            Upper SFSFileSet.Xfer
 1        222
 1        223            "PIPE CMS CREATE DIRECTORY" SFSCustomer.Xfer     "|HOLE"
 1        224            "PIPE CMS CREATE DIRECTORY" SFSContract.Xfer     "|HOLE"
 1        225            "PIPE CMS CREATE DIRECTORY" SFSType.Xfer         "|HOLE"
 1        226            "PIPE CMS CREATE DIRECTORY" SFSFileSet.Xfer      "|HOLE"
 1        227
 1        228            LocalFileID.Xfer = "CMS ACCESS" SFSFileSet.Xfer WorkMode
 1        229
 1        230            Upper LocalFileID.Xfer
 1        231            QElement1 = QElement + 1
 1        232            QElement2 = QElement + 2
 1        233            WorkQueue.QElement   = "CMS ACCESS" SFSFileSet.Xfer WorkMode
 1        234            WorkQueue.QElement1  = "LCD" WorkMode
 1        235            WorkQueue.QElement2  = "GET" TestCaseFileID.Xfer TCFileID.XFer||"."||TCType.Xfer||"."||Wo
     C rkMode "(REPL"
 1        236               QElement = QElement + 3
          237            End;
          238
          239            WorkQueue.0 = QElement - 1
          240            "PIPE Stem WorkQueue." ,
          241               "|> WORK QUEUE A"
          242            "PIPE Stem WorkQueue." ,
          243               "| NLOCATE \LCD\" ,
```

158

This Sheet Has Been Replaced

```
244             "| Stem LogQueue."
245          "PIPE LITERAL ANFSGT090I" Date(S) Time() ,
246             "| > AMFSGT WORKQUE  A"
247          "PIPE STEM LogQueue." ,
248             "| >> AMFSGT WORKQUE A"
249
250          Do ThisUser = 1 to NotifyList.0
251              "SENDFILE AMFSGT WORKQUE A TO" NotifyList.ThisUser
252          End;
253          "PIPE CMS ERASE AMFSGT WORKQUE A | HOLE"
```

This Sheet Has Been Replaced

```
=> Source Listing                                    AMFSGT   EXEC     B1
1-Compiler REXX/370 3.0  LVL PQ00112    Time: 11:51:37                Date: 2000-02-16                Page:    7
:  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                  254   IdentifyFilesEnd: Return;
                  255   /* ---------------------------------------------------------------- */
                  256   /*                                                                  */
                  257   /* ---------------------------------------------------------------- */
                  258   CheckFileAndKillOutsideCopyIfOK: Nop;
                  259             "PIPE DISK WORK QUEUE A" ,
                  260                   "| NLOCATE /LCD/" ,  /* Don't need the LCD directives   */
                  261                   "| STEM CheckFiles."
                  262             Element = 1
                  263
                  264             Do Reference = 1 by 2 to CheckFiles.0
              1   265                   Line1 = Reference
              1   266                   Line2 = Reference + 1
              1   267                   "ACCESS" Word(CheckFiles.Line1,3) "W"
              1   268                   TargetFile.Line2 = Word(CheckFiles.Line2,3)
              1   269                   TargetFile.Line2 = Translate(TargetFile.Line2,' ','.')
              1   270                   "PIPE DISK" TargetFile.Line2 ,
              1   271                         "| LOCATE /*HEADER/" ,
              1   272                         "| VAR VALIDITYTEST"
              1   273                   Select
              1   1   274                    When ValidityTest = "VALIDITYTEST" then do
              2   2   275                         Say Date(S) Time() "AMFSGT027E Invalid" ,
              2   2   276                             ||"/unreadable file:" Word(CheckFiles.Line2,3)
              1   2   277                         end;
              1   1   278                    otherwise do
              2   2   279                         QCommand.Element = "DELETE" Word(CheckFiles.Line2,2)
              2   2   280                         Element = Element + 1
              1   2   281                         end;
              1       282                End;
                  283             End;
                  284
                  285             QCommand.0 = Element - 1
                  286             "PIPE STEM QCommand."
                  287                "|> DELETE QUEUE A"
                  288
                  289             "MAKEBUF"
                  290             Bufferid = rc
                  291             QUEUE 'testcase files4me'
                  292             QUEUE 'quote site testcase.boulder.ibm.com'
                  293             QUEUE 'user ebrs1 z9tx3q8'
                  294             QUEUE 'cd /software/toibm/brs'
```

160
This Sheet Has Been Replaced

```
295             "PIPE DISK DELETE QUEUE A" ,
296                 "| STACK"
297             QUEUE 'quit'
298
299           FTP 'sng-bb2.boulder.ibm.com'
300             Rcs = rc
301             Say "Waiting....."
302             "CP SLEEP 03 SEC"
303             "DROPBUF" Bufferid
304             Say Date(S) Time() ,
```

This Sheet Has Been Replaced

Source Listing


```
305                 "AMFSGT005I" QCommand.0 "files deleted on TESTCASE."
306   CheckFileAndKillOutsideCopyIfOKEnd: Return;
307   /* ---------------------------------------------------------------- */
308   /*                                                                  */
309   /* ---------------------------------------------------------------- */
310   ProcessProfile: Nop;
311             "PIPE DISK" ProfileFileID ,
312                 "| LOCATE /NOTIFY =/" ,
313                 "| STEM NotifyList."
314
315              Do Who = 1 to NotifyList.0
1    316            Parse Var NotifyList.Who . '=' '"' NotifyList.Who '"'
317              End;
318
319             "PIPE DISK" ProfileFileID ,
320                 "| LOCATE /BASEDIRECTORY =/" ,
321                 "| VAR BaseDirectoryRecord"
322
323             Parse Var BaseDirectoryRecord . "=" BaseDirectory .
324   ProcessProfileEnd: Return;
```

This Sheet Has Been Replaced

X Lines 324

e of compiled program in bytes 22232

al messages	Informational	Warning	Error	Severe	Terminating
0	0	0	0	0	0

ishing time of compilation: 11:51:37

This Sheet Has Been Replaced

npilation successful

npiler Options

```
  NOALTERNATE
    CEXEC      (AMFSGT   CEXEC    B1)                    RECFM=F,LRECL=1024
  NOCOMPILE    (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG       (I)
  NOIEXEC
    LIBLEVEL   (*)
    LINECOUNT  (55)
    MARGINS    (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT      (AMFSGT   LISTING  B1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN      (AMFSGT   EXEC     B1 Y ┬ Y M Y ┬ Y ]  0A \  ) RECFM=V,LRECL=114
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 3
```


This Sheet Has Been Replaced

=> Source Listing AMFSGT EXEC B1

```
Compiler REXX/370 3.0  LVL PQ00112    Time: 11:51:37          Date: 2000-02-16          Page:    2
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
    1   /* ------------------------------------------------------------------ */
    2   /*   AMFSGT - SCRAM GeTfiles                                          */
    3   /*   (c) 2000 IBM Corporation All Rights Reserved.                    */
    4   /*   Patent Application Pending                                       */
    5   /* ------------------------------------------------------------------ */
    6   /* ================================================================== */
    7   /* Usage: AMFSGT (option                                              */
    8   /*        Where option may be:                                        */
    9   /*                                                                    */
   10   /*        CLEANSERVER = used to delete files from the outside-the-    */
   11   /*                      firewall-server once it's established they    */
   12   /*                      have been transferred to the local system     */
   13   /*                      and have a readable header.                   */
   14   /*                                                                    */
   15   /* NOTE: To set up the program, you will need to establish the highest*/
   16   /*       Level directory on your local server.                        */
   17   /*       This is the value in AMFSGT PROFILE under the keyword        */
   18   /*       BaseDirectory = 'x'                                          */
   19   /*       This can be done by using the CMS command:                   */
   20   /*           CREATE DIRECTORY 'x'                                     */
   21   /*                                                                    */
   22   /* ================================================================== */
   23   /* Change History                                                     */
   24   /* YYYYMMDD WHO CHANGE                                                */
   25   /* -------- --- ----------------------------------------------------- */
   26   /* 20000112 AMF Initial Release                                       */
   27   /* 20000214 AMF Notification to userid list in AMFSGT PROFILE         */
   28   /*                                                                    */
   29   /* ------------------------------------------------------------------ */
   30              Address CMS;
   31              Trace Off;
   32              Parse Upper Arg Argstring
   33   /* ------------------------------------------------------------------ */
   34   Literals: Nop;
   35             InFileID = "FTP DIROUTP *"
   36             ContractMasterFileID  = "CONTRACT TABLE *"
   37             WorkMode = "W"
   38             ProfileFileID = "AMFSGT PROFILE *"
   39   LiteralsEnd: Nop;
   40   /* ------------------------------------------------------------------ */
   41   /*                                                                    */
```

This Sheet Has Been Replaced

```
42  /* ---------------------------------------------------------------- */
43  MainPgm:  Nop;
44            Rcx = 0
45            Call ProcessProfile
46            Call SetupAccess
47            Call GetWorkToDo
48            Call IdentifyFiles
49            Call GetActualFiles
50            Select
1   51            When Argstring = "(CLEANSERVER" then call CheckFileAndKillOutsideCopyIfOK
```

This Sheet Has Been Replaced

```
> Source Listing                                    AMFSGT   EXEC     B1
Compiler REXX/370 3.0  LVL PQ00112    Time: 11:51:37                    Date: 2000-02-16                    Page:    3
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
      1    52                 Otherwise nop;
           53               End;
           54               Call ExitRtn Rcx
           55    MainPgmEnd: Exit
           56    /* ------------------------------------------------------------------- */
           57    /*                                                                     */
           58    /* ------------------------------------------------------------------- */
           59    ExitRtn:    Parse Arg RCX
           60                'CP SPOOL CONS CLOSE STOP'
           61                Exit RCX
           62    ExitRtnEnd: Return;
           63    /* ------------------------------------------------------------------- */
           64    /*                                                                     */
           65    /* ------------------------------------------------------------------- */
           66    SetupAccess: Nop;
           67                'CP LINK TCPMAINT 592 592 RR'
           68                'ACC 592 R'
           69                'CP SPOOL CONS * CLASS L START'
           70                'CP TERM MORE 1 1'
           71    SetupAccessEnd: Return;
           72    /* ------------------------------------------------------------------- */
           73    /*                                                                     */
           74    /* ------------------------------------------------------------------- */
           75    GetWorkToDo: Nop;
           76                "MAKEBUF"
           77                Bufferid = rc
           78                QUEUE 'testcase files4me'
           79                QUEUE 'quote site testcase.boulder.ibm.com'
           80                QUEUE 'user ebrs1 z9tx3q8'
           81                QUEUE 'cd /software/toibm/brs'
           82                QUEUE 'dir *.* (disk'
           83                QUEUE 'quit'
           84
           85                FTP 'sng-bb2.boulder.ibm.com'
           86                Rcs = rc
           87                "CP SLEEP 02 SEC"
           88                "DROPBUF" Bufferid
           89                "PIPE DISK FTP DIROUTP *" ,
           90                     "| COUNT LINES" ,
           91                     "| VAR INQUESZ"
           92                Say Date(S) time() "AMFSGT004I" INQUESZ "files for transfer"
```

This Sheet Has Been Replaced

```
 93   GetWorkToDoEnd: Return;
 94   /* ---------------------------------------------------------------- */
 95   /*                                                                  */
 96   /* ---------------------------------------------------------------- */
 97   GetActualFiles: Nop;
 98               "MAKEBUF"
 99               Bufferid = rc
100               QUEUE 'testcase files4me'
101               QUEUE 'quote site testcase.boulder.ibm.com'
102               QUEUE 'user ebrsl z9tx3q8'
```

This Sheet Has Been Replaced

```
==> Source Listing                                    AMFSGT   EXEC     B1
   Compiler REXX/370 3.0  LVL PQ00112    Time: 11:51:37                Date: 2000-02-16                Page:    4
    Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                 103              QUEUE 'cd /software/toibm/brs'
                 104              "PIPE DISK WORK QUEUE A" ,
                 105                  "| STACK"
                 106              QUEUE 'quit'
                 107
                 108              FTP 'sng-bb2.boulder.ibm.com'
                 109
                 110              Rcs = rc
                 111              Say "Waiting....."
                 112              "CP SLEEP 03 SEC"
                 113              "DROPBUF" Bufferid
                 114              Say Date(S) Time() ,
                 115                  "AMFSGT005I" WorkQueue.0/3 "files selected on TESTCASE."
                 116              Select
       1         117                  When RCS = 0 then nop;
       1         118                  Otherwise do
   1   2         119                      Say Date(S) time() "AMFSGT012T Error '",
   1   2         120                          || RCs ||"'during file transfer."
   1   2         121                      Say Date(S) time() "AMFSGT012I Ending."
   1   2         122                      Call ExitRtn Rcs
       2         123                      end;
                 124              End;
                 125   GetActualFilesEnd: Return;
                 126   /* ---------------------------------------------------------------- */
                 127   /*                                                                  */
                 128   /* ---------------------------------------------------------------- */
                 129   IdentifyFiles: Nop;
                 130              QElement = 1
                 131              WQElement = 1
                 132             "PIPE DISK" InFileID ,
                 133                "| SORT 42.3 DESCENDING 46.2 DESCENDING " ,
                 134                "       49.2 ASCENDING 51.2 ASCENDING" ,
                 135                "| STEM TestCaseFiles."
                 136
                 137             "PIPE DISK" ContractMasterFileID  ,
                 138                "| DROP FIRST 1" ,
                 139                "| STEM ValidContracts."
                 140
                 141             FilesToTransfer = TestCaseFiles.0
                 142
                 143             Do Xfer = 1 to FilesToTransfer
```


This Sheet Has Been Replaced

```
1       144
1       145
1       146
1       147
1       148
1       149
1       150
1       151
1       152
1       153
```

```
Parse Var TestCaseFiles.Xfer,
          . . . . . Month.Xfer Day.Xfer Hour.Xfer ":" ,
      Minute.Xfer TestCaseFileID.Xfer
Parse Upper Var TestCaseFileID.Xfer ,
            TCCustomer.Xfer ,
            " " TCContract.Xfer ,
            "." TCEDate.Xfer ,
            "." TCFileID.Xfer ,
            "." TCFlag.Xfer ,
            "." TCType.Xfer ,
```

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
 1        154                    ".BIN"
 1        155
 1        156                TCFileIDCheck.Xfer = TCCustomer.Xfer ,
 1        157                                     TCContract.Xfer ,
 1        158                                     TCEDate.Xfer ,
 1        159                                     TCFileID.Xfer ,
 1        160                                     TCFlag.Xfer ,
 1        161                                     TCType.Xfer
 1        162
 1        163                Select
 1    1   164                    When (Words(TCFileIDCheck.Xfer)<>6) then do
 2    2   165                          Say Date(S) Time() "AMFSGT024B" ,
 2    2   166                              "Invalid transfer file:'" ,
 2    2   167                              ||TCCustomer.Xfer||"' requesting service."
 2    2   168                          Iterate XFer
 1    2   169                          end;
 1    1   170                    When ((Word(TCFileIDCheck.Xfer,5)<>'D')  & (Word(TCFileIDCheck.Xfer,5)<>'T') ) the
              C n do
 2    2   171                          Say Date(S) Time() "AMFSGT024E" ,
 2    2   172                              "Invalid transfer file:'" ,
 2    2   173                              ||TCCustomer.Xfer||"' Type='"||TCType.Xfer||"'"
 2    2   174                          Iterate XFer
 1    2   175                          end;
 1    1   176                    When (Datatype(Word(TCFileIDCheck.Xfer,3),'W')<>1) then do
 2    2   177                          Say Date(S) Time() "AMFSGT025E" ,
 2    2   178                              "Invalid transfer file:'" ,
 2    2   179                              ||TCCustomer.Xfer||"' Date='"||TCEDate.Xfer||"'"
 2    2   180                          Iterate XFer
 1    2   181                          end;
 1    1   182                     Otherwise nop;
 1        183                End;
 1        184
 1        185
 1        186                "PIPE STEM ValidContracts." ,
 1        187                     "| LOCATE 1.8 /"||TCCustomer.Xfer||"/" ,
 1        188                     "| VAR FoundCustomer.Xfer"
 1        189
 1        190                "PIPE STEM ValidContracts." ,
 1        191                     "| LOCATE 10.8 /"||TCContract.Xfer||"/" ,
 1        192                     "| VAR FoundContract.Xfer"
 1        193
```


This Sheet Has Been Replaced

```
1       194
1   1   195     Select
2   2   196         When Left(FoundCustomer.Xfer,14) = "FOUNDCUSTOMER." then do
2   2   197              Say Date(S) Time() "AMFSGT008E"
2   2   198                  "Invalid customer number: '"
2   2   199                  ||TCCustomer.Xfer||"' requesting service."
1   2   200              Iterate XFer
1   1   201              end;
1       202         Otherwise nop;
1       203     End;
            :   Select
```

This Sheet Has Been Replaced

```
1   1   204                     When Left(FoundContract.Xfer,14) = "FOUNDCONTRACT." then do
2   2   205                         Say Date(S) Time() "AMFSGT008E" ,
2   2   206                             "Invalid contract number: '" ,
2   2   207                             ||TCContract.Xfer ||"' requesting service" ,
2   2   208                             "for customer:" TCCustomer.Xfer
2   2   209                         Iterate XFer
1   2   210                         end;
1   1   211                     Otherwise nop;
1       212                 End;
1       213                 SFSCustomer.Xfer      = BaseDirectory||TCCustomer.Xfer||"."
1       214                 SFSContract.Xfer      = SFSCustomer.Xfer||TCContract.Xfer||"."
1       215                 SFSType.Xfer          = SFSContract.Xfer||TCFlag.Xfer||"."
1       216                 SFSFileSet.Xfer       = SFSType.Xfer||TCEDate.Xfer||"."
1       217
1       218                 Upper SFSCustomer.Xfer
1       219                 Upper SFSContract.Xfer
1       220                 Upper SFSType.Xfer
1       221                 Upper SFSFileSet.Xfer
1       222
1       223                 "PIPE CMS CREATE DIRECTORY" SFSCustomer.Xfer     "|HOLE"
1       224                 "PIPE CMS CREATE DIRECTORY" SFSContract.Xfer     "|HOLE"
1       225                 "PIPE CMS CREATE DIRECTORY" SFSType.Xfer         "|HOLE"
1       226                 "PIPE CMS CREATE DIRECTORY" SFSFileSet.Xfer      "|HOLE"
1       227
1       228                 LocalFileID.Xfer = "CMS ACCESS" SFSFileSet.Xfer WorkMode
1       229
1       230                 Upper LocalFileID.Xfer
1       231                 QElement1 = QElement + 1
1       232                 QElement2 = QElement + 2
1       233                 WorkQueue.QElement  = "CMS ACCESS" SFSFileSet.Xfer WorkMode
1       234                 WorkQueue.QElement1 = "LCD" WorkMode
1       235                 WorkQueue.QElement2 = "GET" TestCaseFileID.Xfer TCFileID.XFer||"."||TCTYpe.Xfer||"."||Wo
                      C rkMode "(REPL"
1       236                 QElement = QElement + 3
        237             End;
        238
        239             WorkQueue.0 = QElement - 1
        240             "PIPE Stem WorkQueue." ,
        241                "|> WORK QUEUE A"
        242             "PIPE Stem WorkQueue." ,
        243                "| NLOCATE \LCD\" ,
```

This Sheet Has Been Replaced

```
244             "| Stem LogQueue."
245          "PIPE LITERAL AMFSGT090I" Date(S) Time() ,
246             "| > AMFSGT WORKQUE  A"
247          "PIPE STEM LogQueue." ,
248             "| >> AMFSGT WORKQUE A"
249
250          Do ThisUser = 1 to NotifyList.0
1   251          "SENDFILE AMFSGT WORKQUE A TO" NotifyList.ThisUser
252          End;
253          "PIPE CMS ERASE AMFSGT WORKQUE A | HOLE"
```


This Sheet Has Been Replaced

```
                254  IdentifyFilesEnd: Return;
                255  /* ---------------------------------------------------------------- */
                256  /*                                                                  */
                257  /* ---------------------------------------------------------------- */
                258  CheckFileAndKillOutsideCopyIfOK: Nop;
                259            "PIPE DISK WORK QUEUE A" ,
                260                  "| NLOCATE /LCD/" ,  /* Don't need the LCD directives   */
                261                  "| STEM CheckFiles."
                262            Element = 1
                263
                264            Do Reference = 1 by 2 to CheckFiles.0
      1         265                  Line1 = Reference
      1         266                  Line2 = Reference + 1
      1         267                  "ACCESS" Word(CheckFiles.Line1,3) "W"
      1         268                  TargetFile.Line2 = Word(CheckFiles.Line2,3)
      1         269                  TargetFile.Line2 = Translate(TargetFile.Line2,' ','.')
      1         270                  "PIPE DISK" TargetFile.Line2 ,
      1         271                        "| LOCATE /*HEADER/" ,
      1         272                        "| VAR VALIDITYTEST"
      1         273             :          Select
      1    1    274                        When ValidityTest = "VALIDITYTEST" then do
      2    2    275                             Say Date(S) Time() "AMFSGT027E Invalid" ,
      2    2    276                                 ||"/unreadable file:" Word(CheckFiles.Line2,3)
      1    2    277                             end;
      1    1    278                        otherwise do
      2    2    279                             QCommand.Element = "DELETE" Word(CheckFiles.Line2,2)
      2    2    280                             Element = Element + 1
      1    2    281                             end;
      1         282                  End;
                283            End;
                284
                285            QCommand.0 = Element - 1
                286            "PIPE STEM QCommand."
                287                "|> DELETE QUEUE A"
                288
                289            "MAKEBUF"
                290            Bufferid = rc
                291            QUEUE 'testcase files4me'
                292            QUEUE 'quote site testcase.boulder.ibm.com'
                293            QUEUE 'user ebrsl z9tx3q8'
                294            QUEUE 'cd /software/toibm/brs'
```


This Sheet Has Been Replaced

```
295             "PIPE DISK DELETE QUEUE A" ,
296                 "| STACK"
297             QUEUE 'quit'
298
299           FTP 'sng-bb2.boulder.ibm.com'
300             Rcs = rc
301             Say "Waiting....."
302             "CP SLEEP 03 SEC"
303             "DROPBUF" Bufferid
304             Say Date(S) Time() ,
```

This Sheet Has Been Replaced

=> Source Listing AMFSGT EXEC B1


```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                 305                 "AMFSGT005I" QCommand.0 "files deleted on TESTCASE."
                 306   CheckFileAndKillOutsideCopyIfOKEnd: Return;
                 307   /* ---------------------------------------------------------------- */
                 308   /*                                                                  */
                 309   /* ---------------------------------------------------------------- */
                 310   ProcessProfile: Nop;
                 311             "PIPE DISK" ProfileFileID ,
                 312                 "| LOCATE /NOTIFY =/" ,
                 313                 "| STEM NotifyList."
                 314
                 315              Do Who = 1 to NotifyList.0
               1 316                 Parse Var NotifyList.Who . '=' '"' NotifyList.Who '"'
                 317              End;
                 318
                 319             "PIPE DISK" ProfilePileID ,
                 320                 "| LOCATE /BASEDIRECTORY =/" ,
                 321                 "| VAR BaseDirectoryRecord"
                 322
                 323             Parse Var BaseDirectoryRecord . "=" BaseDirectory .
                 324   ProcessProfileEnd: Return;
```


This Sheet Has Been Replaced

X Lines 324

e of compiled program in bytes 22232

al messages	Informational	Warning	Error	Severe	Terminating
0	0	0	0	0	0

ishing time of compilation: 11:51:37

This Sheet Has Been Replaced

APPENDIX E

This Sheet Has Been Replaced



Compilation Summary BRSRDSCH EXEC

Compiler REXX/370 3.0 LVL PQ26073 Time: 15:50:56 Date: 2000-02-03 Page: 1

ilation successful

iler Options

```
 NOALTERNATE
   CEXEC      (BRSRDSCH CEXEC     G1)                    RECFM=F,LRECL=1024
 NOCOMPILE    (S)
 NOCONDENSE
 NODLINK
 NODUMP
   FLAG       (I)
 NOIEXEC
   LIBLEVEL   (*)
   LINECOUNT  (55)
   MARGINS    (1 *)
 NOOBJECT
   OPTIMIZE
   PRINT      (BRSRDSCH LISTING   G1)                    RECFM=V,LRECL=121
 NOSAA
 NOSLINE
   SOURCE
   SYSIN      (BRSRDSCH EXEC      G1 R § R U R § R \ (   PD) RECFM=V,LRECL=90
 NOTERMINAL
 NOTESTHALT
 NOTRACE
 NOXREF
Minimum Library Level required: 3
```

180

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
     1  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
     2  /*                                                                                                     */
     3  /*              **  BRSRDSCH EXEC **                                                                   */
     4  /*                                                                                                     */
     5  /*  Function:                                                                                          */
     6  /*                                                                                                     */
     7  /*      Build file of formatted IOCP records suitable for use in VM                                    */
     8  /*      directory for BRS customer disaster or test.                                                   */
     9  /*                                                                                                     */
    10  /*      Respond to queries for list of contracts for a specific date                                   */
    11  /*      or lists of dates for a specific contract.                                                     */
    12  /*                                                                                                     */
    13  /*      Help prompt if null, question mark, or questionable input                                      */
    14  /*                                                                                                     */
    15  /*                                                                                                     */
    16  /*  Description:                                                                                       */
    17  /*                                                                                                     */
    18  /*      This EXEC uses the GAI ALLOCV2 file in WSCVMU:WSCBRS.ASCHED to                                 */
    19  /*      identify tape, disk, CPU and N/W controller devices scheduled                                  */
    20  /*      for a customer test.                                                                           */
    21  /*                                                                                                     */
    22  /*      The file created by the EXEC contains 'IOCP records'                                           */
    23  /*      for all scheduled tape, disk, and N/W controller devices found                                 */
    24  /*      in $VALID$ IOCPRECS file in WSCVMU:WSCBRS.BRSTOOLS.                                            */
    25  /*                                                                                                     */
    26  /*      The file created lists resources for the entire test, whether                                  */
    27  /*      using one or multiple CPUs.                                                                    */
    28  /*                                                                                                     */
    29  /*      This file is best used as a template to create a final IOCP                                    */
    30  /*      record configuration for a test since records will be present                                  */
    31  /*      for devices scheduled whether or not they are actually used                                    */
    32  /*      in a particular test.  Also the scheduler system does not                                      */
    33  /*      know how devices are distributed amongst multiple CPUs                                         */
    34  /*      scheduled for a test, so the resulting file constitutes the                                    */
    35  /*      devices for the test complex, not for a particular CPU when                                    */
    36  /*      multiple CPUs are available.                                                                   */
    37  /*                                                                                                     */
    38  /*      CTC devices are not scheduled, since they are considered part                                  */
    39  /*      of standard equipment for a CPU.                                                               */
    40  /*                                                                                                     */
    41  /*                                                                                                     */
```


This Sheet Has Been Replaced

```
42  /* Documentation                                                    */
43  /*                                                                  */
44  /*     See file 'VALIDS SCRIPT' or 'VALIDS LIST3820' on             */
45  /*     WSCVM's shared file system 'WSCVMU:WSCBRS.BRSDOCS' for a      */
46  /*     more complete description of this and related EXECs and      */
47  /*     how they may be used.                                        */
48  /*                                                                  */
49  /*                                                                  */
50  /* Invoking syntax:                                                 */
51  /*                                                                  */
```


This Sheet Has Been Replaced

```
52  /*    BRSRDSCH ContractNumber ContractDate <StartTime> <CustName>      */
53  /*           <( LISTCOntract>                                          */
54  /*           <( LISTDAte>                                              */
55  /*           <( LISTCUstomer>                                          */
56  /*           <( NOPRint>                                               */
57  /*           <( FN=filename>                                           */
58  /*                                                                     */
59  /*                                                                     */
60  /*  Invoking syntax examples:                                          */
61  /*                                                                     */
62  /*    BRSRDSCH    R31417 8/3/1998                                      */
63  /*                                                                     */
64  /*    BRSRDSCH    R31417 8/3/1998 0800                                 */
65  /*                                                                     */
66  /*    BRSRDSCH    R31417 8/3/1998 (NOPR                                */
67  /*                                                                     */
68  /*    BRSRDSCH    R31417 (LISTCO                                       */
69  /*                                                                     */
70  /*    BRSRDSCH    8/3/1998 (LISTDA                                     */
71  /*                                                                     */
72  /*    BRSRDSCH    custname (LISTCU                                     */
73  /*                                                                     */
74  /*                                                                     */
75  /*  Input:                                                             */
76  /*                                                                     */
77  /*    ContractNumber:  The usual, like 'R31417', or 'SVS'              */
78  /*                                                                     */
79  /*    ContractDate:    The usual start date in 'mm/dd/yyyy' format,    */
80  /*                        like '7/23/1998'                             */
81  /*                                                                     */
82  /*    StartTime:       The usual start time in 'hhmm' format,          */
83  /*                        like '1730'.  This field is optional and     */
84  /*                        only needs to be used when ContractNumber    */
85  /*                        and ContractDate do not provide sufficient   */
86  /*                        resolution.                                  */
87  /*                                                                     */
88  /*    CustName:        Any portion of customer name.  The field is     */
89  /*                        optional and only needs to be used when      */
90  /*                        ContractNumber and the other fields do not   */
91  /*                        provide sufficient resolution.               */
92  /*                                                                     */
```


This Sheet Has Been Replaced

```
 93   /*
 94   /*  Required ancillary file                                          */
 95   /*                                                                   */
 96   /*    CAI ALLOC:      Table of scheduler information,                */
 97   /*                    in WSCVMU:WSCBRS.ASCHED directory on WSCVM      */
 98   /*                                                                   */
 99   /*    $VALID$ IOCPRECS:   Table of template IOCP records,            */
100   /*                    in WSCVMU:WSCBRS.ASCHED directory on WSCVM      */
101   /*                                                                   */
102   /*                                                                   */
```

This Sheet Has Been Replaced

```
103   /* Output:                                                          */
104   /*                                                                  */
105   /*    'cnlbl' IOCPRECS file on user's A disk containing reverse-mapped */
106   /*       DEDICATE statements for native customer MVS starter systems. */
107   /*                                                                  */
108   /*    Filename 'cnlbl' is constructed from first four and last three  */
109   /*    characters of customer name.                                  */
110   /*                                                                  */
111   /*    Hardcopy of 'cnlbl' IOCPRECS via ANYOUT print unless user     */
112   /*       specifies NOPRint option                                   */
113   /*                                                                  */
114   /*    Help prompt if null, question mark, or questionable input     */
115   /*                                                                  */
116   /*                                                                  */
117   /* Changes:                                                         */
118   /*                                                                  */
119   /*    7/19/98    Original by S. W. Wood                             */
120   /*    7/23/98    Add additional comments to output boilerplate      */
121   /*               Add unknown devices as comment records in output file */
122   /*    7/24/98    Separate DASD: TAPE: and FEP: list entries with commas*/
123   /*    7/25/98    Handle case where no resources are scheduled       */
124   /*    8/01/98    Add query capabilities                             */
125   /*    8/03/98    Add ANYOUT (PROF(FLATPROF) of resulting file       */
126   /*    8/19/98    Add colon to start time formatting                 */
127   /*    9/03/98    Handle null contract number condition for std. search */
128   /*               Improve error message for not found conditions     */
129   /*    9/03/98    Remove comma at end of device lists in comments    */
130   /*               Remove translation of error comments in output file */
131   /*    9/11/98    When no contract, also check customer name         */
132   /*   11/25/98    Add NOPRint option to disallow hardcopy print      */
133   /*               Add VMRES: to comment boilerplate                  */
134   /*               Note that resources are those for entire test      */
135   /*   12/04/98    Add FN= option to force filename                   */
136   /*               Add LPAR records boilerplate                       */
137   /*   12/05/98    Fix 'cnlbl' to disallow slash '/' in filename      */
138   /*    3/24/99    Add header to display output                       */
139   /*               Add customer name to optional search criteria      */
140   /*                 when generating IOCP records                     */
141   /*               Ensure all resources for given contract/date/time/ */
142   /*                 customer are reported in output IOCPRECS file    */
143   /*    3/25/99    Add excfn to messages                              */
```

185

This Sheet Has Been Replaced

```
145  /*  8/10/99  Handle changes per CMR #42947                          */
146  /*           Fix NOPR position bug                                  */
147  /*           Use GAI ALLOCV2 Y2K version rather than GAI ALLOC      */
148  /*  8/14/99  Use "3174s,FEPS:" on VM directory comment line         */
149  /*  8/15/99  Insert OPTIONS: BRSLAB options defaults                */
150  /*           Separate terminal suites from tape rooms               */
151  /*  8/16/99  Add CALL RECORD: boilerplate to CONTRACT comment line  */
152  /*           Use last three char of DASD location                   */
153  /*  8/17/99  Add printer IOCP records to output                     */

144  /*  Handle NOPR and FN = options with others                        */
```

This Sheet Has Been Replaced

```
154   /*    8/19/99    Parse SYSTEM: components; ensure fits on line          */
155   /*               Ensure gateway components fit on line                   */
156   /*               Ensure channel extendor components fit on line          */
157   /*    8/20/99    Ensure list of event numbers reported                   */
158   /*    8/25/99    Add LISTALLCPU option to identify odd CPUs              */
159   /*    9/24/99    Add date/time/location of $VALID$ IOCPRECS file used    */
160   /*    9/27/99    Eliminate duplicates in list of event numbers           */
161   /*                                                                       */
162   /*                                                                       */
163   /*  Wish list:                                                           */
164   /*                                                                       */
165   /*                                                                       */
166   /*                                                                       */
167   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
168
169   /*%PAGE */
```




```
170  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
171  /*                                                                                         */
172  /* Save caller info and environment                                                        */
173  /*                                                                                         */
174  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
175
176  'identify (lifo'
177  parse upper pull caller . callernode .
178  parse source . . excfn excft .
179
180  dropbuf 0
181  'makebuf'
182  svmb = rc
183
184  if callernode <> 'WSCVM' then
185    do
186      say ' '
187      say 'Sorry, this EXEC must be run at WSCVM'
188      say ' Terminating.'
189      'dropbuf' svmb
190      exit 999
191    end
192
193
194  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
195  /*                                                                                         */
196  /* See what he gave us to work with                                                        */
197  /*                                                                                         */
198  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
199
200  parse upper arg cnum cdate ctime cname '(' options
201
202  if (cnum = '') | (cnum = '?') then
203    call say_parms
204
205  listco = 0
206  listcu = 0
207  listda = 0
208  listal = 0
209  nopr   = 0
210  ffn    = 0
```


This Sheet Has Been Replaced

```
211   odds   = ''
212
213                                   /* look for NOPRint function          */
214   nopr   = index(options,'NOPR')
215
216                                   /* look for FN=filename option        */
217   ffn    = index(options,'FN=')
218   if ffn <> 0 then
219     do
1    220       forcefn = substr(options,ffn+3)
```


This Sheet Has Been Replaced

```
1       221
1       222       forcefn = word(forcefn,1)          /* use only FN= portion of options    */
        223     end
        224
        225
        226   if (options = '') | (nopr <> 0)        /* see if caller wants list function  */
        227                                          | (ffn <> 0) then
        228     do                                   /* no; he wants IOCPREGS built        */
1       229
1       230
1       231       if index(cnum,'/') <> 0 then     /* handle null contract number cond.  */
1       232         do
2       233
2       234           cname = ctime                /* move everything over one position  */
2       235           ctime = cdate
2       236           cdate = cnum
2       237           cnum  = ''
1       238         end
1       239
1       240
1       241       if datatype(ctime,'N') <> '1' then /* handle null start time condition  */
1       242         do
2       243
2       244           cname = ctime                /* assume 'ctime' is customer name    */
2       245           ctime = ''
1       246         end
1       247
1       248       parse var cdate mm '/' dd '/' yyyy /* clean up the date caller gave us   */
1       249
1       250       if length(mm) < 2 then
1       251         mm = insert('0',mm)
1       252
1       253       if length(dd) < 2 then
1       254         dd = insert('0',dd)
1       255
1       256       if yyyy = '99' then
1       257         yyyy = '1999'
1       258       else
1       259         if yyyy = '00' then
1       260           yyyy = '2000'
1       261
```


This Sheet Has Been Replaced

```
      262     end
      263
      264   else
      265
      266     do                                  /* caller wants a list function      */
1     267                                         /* LISTCOntract function found       */
1     268       listco = index(options,'LISTCO')
1     269
1     270                                         /* LISTDAte     function found       */
1     271       listda = index(options,'LISTDA')
```

This Sheet Has Been Replaced

```
1       272
1       273
1       274       listcu = index(options,'LISTCU')          /* LISTCUstomer function found        */
1       275
1       276
1       277       listal = index(options,'LISTAL')          /* LISTALlcpu   function found        */
1       278
1       279       if (listco = 0) & (listda = 0) & (listcu = 0) & (nopr = 0),
1       280         & (ffn = 0) & (listal = 0) then
1       281           call say_parms
1       282
1       283       if listda <> 0 then
1       284         do
2       285                                   /* list contracts on a specific date  */
2       286                                   /* clean up the date caller gave us   */
2       287           parse var cnum mm '/' dd '/' yyyy
2       288
2       289           if length(mm) < 2 then
2       290             mm = insert('0',mm)
2       291
2       292           if length(dd) < 2 then
2       293             dd = insert('0',dd)
2       294
2       295           if yyyy = '99' then
2       296             yyyy = '1999'
2       297           else
2       298             if yyyy = '00' then
2       299               yyyy = '2000'
2       300
2       301           say 'Listing contracts for date' mm'/'dd'/'yyyy
2       302           say ' '
2       303           say 'CNTR_NUM DATE---- STRT CUSTOMER_NAME------------'
2       304           say ' '
2       305
2       306           call read_gai_alloc
2       307
2       308           exit(1)
1       309
1       310         end
1       311       else if listco <> 0 then
1       312         do
                                      /* list dates for a specific contract */
```


This Sheet Has Been Replaced

```
2       313
2       314             say 'Listing dates for contract' cnum
2       315             say ' '
2       316             say 'CNTR_NUM DATE---- STRT CUSTOMER_NAME----------'
2       317             say ' '
2       318
2       319             call read_gai_alloc
2       320
2       321             exit(2)
2       322
```

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
 1       323          end
 1       324
 1       325        else if listcu <> 0 then
 1       326          do
 2       327                                  /* list dates for a specific customer */
 2       328
 2       329                                  /* ensure we have most of the name    */
 2       330            parse upper arg cusname          '(' options stuff
 2       331            cusname = strip(cusname)
 2       332
 2       333            say 'Listing dates for customer' cusname
 2       334            say ' '
 2       335            say 'CNTR_NUM DATE---- STRT CUSTOMER_NAME------------'
 2       336            say ' '
 2       337
 2       338            call read_gai_alloc
 2       339
 2       340            exit(3)
 2       341
 1       342          end
 1       343
 1       344        else if listal <> 0 then
 1       345          do
 2       346                                  /* identify odd CPUs in list          */
 2       347
 2       348            say 'Listing odd CPU info as of' date() 'at' time()
 2       349            say ' '
 2       350            say ' '
 2       351
 2       352            call read_gai_alloc
 2       353
 2       354            do ods = 1 to words(odds)
 3       355              say word(odds,ods)
 2       356            end
 2       357            exit(4)
 2       358
 1       359          end
 1       360
         361      end                         /* end of list function parm process  */
         362
         363
```


This Sheet Has Been Replaced

/*%PAGE */

This Sheet Has Been Replaced

```
365   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
366   /*                                                                       */
367   /* Main code to build IOCPRECS file                                      */
368   /*                                                                       */
369   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
370
371
372                                  /* * * * * * * * * * * * * * * * * * * * */
373                                  /*                                       */
374                                  /* read the schedule file                */
375                                  /*                                       */
376                                  /* * * * * * * * * * * * * * * * * * * * */
377
378   sdate     = mm || '/' || dd || '/' || yyyy
379   scontract = cnum
380   sstart    = ctime
381   scus      = strip(cname)
382
383   call read_gai_alloc
384
385   if sfound then nop
386     else exit (998)
387
388                                  /* * * * * * * * * * * * * * * * * * * * */
389                                  /*                                       */
390                                  /* get the IOCPRECS file                 */
391                                  /*                                       */
392                                  /* * * * * * * * * * * * * * * * * * * * */
393
394   call read_IOCPRECS
395
396                                  /* * * * * * * * * * * * * * * * * * * * */
397                                  /*                                       */
398                                  /* But first, save some boilerplate      */
399                                  /*  for VM directory comments section    */
400                                  /*                                       */
401                                  /* * * * * * * * * * * * * * * * * * * * */
402
403                                  /* build some asterisks                  */
404   asts = ''
405   do ix = 1 to 74
```

196

This Sheet Has Been Replaced

```
406       asts = asts || '*'
407     end
408
409     hdr = ''
410     hdx = 0
411
412                                        /* build stuff line by line          */
413     hdx = hdx + 1
414     hdr.hdx = '* Created by' caller 'at' callernode 'using' excfn,
415         'on' date() 'at' time()
```


This Sheet Has Been Replaced

```
416
417   hdx = hdx + 1
418   hdr.hdx = "* '$VALID$ IOCPRECS" brstfm"' located at" iordir " ..."
419   hdx = hdx + 1
420   hdr.hdx = "* ... was created on" iordate "at" iortime
421
422   hdx = hdx + 1
423   hdr.hdx = '***  Devices scheduled for test  ***'
424
425   hdx = hdx + 1
426   hdr.hdx = '************** START 'gaccust substr(asts,1,(74 - length(gaccust) - 22))
427
428   hdx = hdx + 1
429   hdr.hdx = '* NAME:' gaccust
430
431                                  /* separate terminal suites from      */
432                                  /*   tape suites                      */
433
434   taposuites = ''
435   termsuites = ''
436
437   do ts = 1 to words(gasuite)
1     438
1     439     thisuite = word(gasuite,ts)
1     440
1     441     if index(thisuite,'TAPE') <> 0 |,
1     442        index(thisuite,'POOL') <> 0 |,
1     443        index(thisuite,'SWING') <> 0 then
1     444       tapesuites = tapesuites thisuite
1     445     else
1     446       termsuites = termsuites thisuite
1     447
448   end
449
450   if index(gatape,'POOL') <> 0 then
451       tapesuites = tapesuites ''PRONTO_POOL'
452
453   hdx = hdx + 1
454   hdr.hdx = '* TERMINAL SUITE:' termsuites
455
456   hdx = hdx + 1
```


This Sheet Has Been Replaced

```
1        457    hdr.hdx = '* TAPE ROOM:' tapesuites
         458
         459    do ccc = 1 to words(gacpu) by 3
1        460
1        461      hdx = hdx + 1
1        462      hdr.hdx = '* SYSTEM:' word(gacpu,ccc) word(gacpu,ccc+1),
1        463         word(gacpu,ccc+2)
1        464
         465    end
         466
```


This Sheet Has Been Replaced

```
467   call set_dirmsg ('DASD:' gadasd)
468
469   call set_dirmsg ('TAPE:' gatape)
470
471   call set_dirmsg ('3174s,FEPS:' ga3174)
472
473   hdx = hdx + 1
474   hdr.hdx = '* PRINTERS:' gaprtr
475
476   call set_dirmsg ('CHANNEL EXTENDERS:' gachext)
477
478   call set_dirmsg ('GATEWAYS:' gagatew)
479
480   hdx = hdx + 1
481   hdr.hdx = '* CTCS:                    *** unscheduled: add as required ***'
482
483   hdx = hdx + 1
484   hdr.hdx = '* BRS STARTER STRING: *** unscheduled: add as required ***'
485
486   hdx = hdx + 1
487   hdr.hdx = '* DEDICATED DEVICES:  *** unscheduled: add as required ***'
488
489   hdx = hdx + 1
490   hdr.hdx = '* VMRES: xxxx          *** supply from local schedule   ***'
491
492   hdx = hdx + 1
493   hdr.hdx = '* TEST START DATE:' substr(gacdate,5,2) || '/' ||,
494                                  substr(gacdate,7,2) || '/' ||,
495                                  substr(gacdate,1,4)
496
497   hdx = hdx + 1
498   hdr.hdx = '* TEST START TIME:' substr(gastart,1,2)':'substr(gastart,3,4),
499             '  HOURS:' gahours
500
501   hdx = hdx + 1
502   hdr.hdx = '* CONTRACT NUMBER:' gacontract,
503             '  CALL RECORD: yyyyy'
504
505   hdx = hdx + 1
506   hdr.hdx = '* EVENT NUMBER:    ' gaevno
507
```

This Sheet Has Been Replaced

```
508    hdx = hdx + 1
509    hdr.hdx = '* OPTIONS: NRBV=NO  MDISK=NO  PWSUFFIX=pw'
510
511    hdx = hdx + 1
512    hdr.hdx = '*************************************************************************',
513     || '*************'
514
515    hdx = hdx + 1
516    hdr.hdx = '** 'gaccust'       LPAR RECORDS
517
```


This Sheet Has Been Replaced

```
518   hdx = hdx + 1
519   hdr.hdx = '*************************************************************************',
520    || '**************'
521
522   hdx = hdx + 1
523   hdr.hdx = '**'
524
525   hdx = hdx + 1
526   hdr.hdx = '**          # ID    CSTR ORIG RSVC ORSC ESTR ORIG RSVE CP A ADDR PARMS--- WT  ADM'
527
528   hdx = hdx + 1
529   hdr.hdx = '**'
530
531   hdx = hdx + 1
532   hdr.hdx = '1parname 1 1 ESA 1248 0000 ---- ---- 0512 0000 ---- 06 N ---- -------- 015 Y'
533
534   hdx = hdx + 1
535   hdr.hdx = '**'
536
537   hdx = hdx + 1
538   hdr.hdx = '*************************************************************************',
539    || '**************'
540
541   hdx = hdx + 1
542   hdr.hdx = '** 'gaccust'       IOCP RECORDS
543
544   hdx = hdx + 1
545   hdr.hdx = '*************************************************************************',
546    || '**************'
547
548   hdx = hdx + 1
549   hdr.hdx = '**'
550
551   hdx = hdx + 1
552   hdr.hdx = '**          CUST                    BRS'
553
554   hdx = hdx + 1
555   hdr.hdx = '**DEVICE ADDR QTY LPAR-ASSGN ADDR',
556      'COMMENTS ----------------------------'
557
558   hdx = hdx + 1
```


This Sheet Has Been Replaced

```
559   hdx:hdx = '00'
560
561                              /* * * * * * * * * * * * * * * * * * * * */
562                              /*                                       */
563                              /* now provide IOCPREC for each          */
564                              /*   device in DASD, tape, 3174 list     */
565                              /*                                       */
566                              /* * * * * * * * * * * * * * * * * * * * */
567
568   out = ''
```

This Sheet Has Been Replaced

```
          569   odk = 0
          570
          571
          572   do j = 1 to words(gadasd)                  /*   look for DASD                  */
1         573
1         574     call search_IOCPRECS( word(gadasd,j) 'DASD')
1         575
          576   end
          577
          578
          579   do j = 1 to words(gatape)                  /*   look for tape                  */
1         580
1         581     call search_IOCPRECS( word(gatape,j) 'TAPE')
1         582
          583   end
          584
          585
          586   do j = 1 to words(ga3174)                  /*   look for 3174                  */
1         587
1         588     call search_IOCPRECS( word(ga3174,j) '3174')
1         589
          590   end
          591
          592
          593   do j = 1 to words(gaprtr)                  /*   look for printers              */
1         594
1         595     call search_IOCPRECS( word(gaprtr,j) 'PRTR')
1         596
          597   end
          598
          599
          600                                              /* * * * * * * * * * * * * * * * * * */
          601                                              /*                                   */
          602                                              /*  Now emit final output file       */
          603                                              /*                                   */
          604                                              /* * * * * * * * * * * * * * * * * * */
          605
          606   ofn = '$S$W$W$'                            /*  setup name for output file       */
          607
          608   cnlbl = substr(gaccust,1,4) || substr(gaccust,length(gaccust)-2,3) || '$'
          609   cnlbl = translate(cnlbl,'BPHACLRS',' .-&,()/')
```


This Sheet Has Been Replaced

```
610   ofn = cnlbl
611   if ffn <> 0 then
612     ofn = forcefn
613
614   oft = 'IOCPRECS'
615
616                                   /*  setup name for work file          */
617   oxn = '$S$X$X$'
618   oxt = 'IOCPRECS'
619
```

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          620
          621   'SET CMSTYPE HT'                          /*  erase files if they exist        */
          622
          623   'state' ofn oft 'A'
          624   if rc = 0 then
          625     'erase' ofn oft 'A'
          626
          627   'state' oxn oxt 'A'
          628   if rc = 0 then
          629     'erase' oxn oxt 'A'
          630
          631   'SET CMSTYPE RT'
          632
          633                                             /*  write out the work file          */
          634   'EXECIO' odx   'DISKW' oxn oxt 'A (STEM OUT. FINIS'
          635
          636                                             /*  sort the IOCPRECS part of output */
          637   call sort_file
          638
          639                                             /*  write out the header stuff       */
          640   'EXECIO' hdx   'DISKW' ofn oft 'A (STEM HDR. FINIS'
          641
          642                                             /*  read the sorted file             */
          643   'EXECIO * DISKR' oxn oxt 'A (STEM OUT.'
          644
          645                                             /*  re-translate BRS ADDR field      */
          646   if out.0 > 0 then
          647     do iw = 1 to out.0
    1     648
    1     649                                             /*  skip any comment lines           */
    1     650       if substr(out.iw,1,1) <> '*' then
    1     651         do
    2     652           daddr   = substr(out.iw,30,4)
    2     653
    2     654           daddr   = translate(daddr, '0123456789', 'abcdefghij')
    2     655
    2     656           out.iw  = substr(out.iw,1,29) || daddr || substr(out.iw,34)
    1     657         end
    1     658
          659     end
          660
```

206

This Sheet Has Been Replaced

```
661
662  'EXECIO' odx 'DISKW' ofn oft 'A (STEM OUT. FINIS'          /* write out the output file */
663
664                                                            /* erase the work file       */
665  'SET CMSTYPE HT'
666  'state' oxn oxt 'A'
667  if rc = 0 then
668    'erase' oxn oxt 'A'
669  'SET CMSTYPE RT'
670
```


This Sheet Has Been Replaced

```
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          671   if nopr = 0 then
          672     do
 1        673       'ANYOUT' OFN OFT 'A',
 1        674         '(SYON DUPLEX SYSVAR(X NO N YES D YES H YES) PROF(FLATPROF)'
          675     end
          676   else
          677     nop
          678
          679   exit 0                          /* Get out                          */
          680
          681
          682   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
          683   /*                                                                    */
          684   /* Various subroutines follow                                       */
          685   /*                                                                    */
          686   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
          687
          688
          689   /*%PAGE */
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
   690   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
   691   /*                                                                                                     */
   692   /*  Read GAI ALLOC file                                                                                */
   693   /*                                                                                                     */
   694   /*  Input                                                                                              */
   695   /*                                                                                                     */
   696   /*    scontract = contract number we are looking for                                                   */
   697   /*    sdate     = date of contract we are looking for                                                  */
   698   /*    sstart    = start time of contract we are looking for                                            */
   699   /*    scus      = name of customer we are looking for                                                  */
   700   /*                                                                                                     */
   701   /*    listco    = non-zero, 'cnum' set to contract number                                              */
   702   /*    listcu    = non-zero, 'cusname' set to customer name or                                          */
   703   /*                  portion thereof                                                                    */
   704   /*    listda    = non-zero, 'mm', 'dd', 'yyyy' set to contract date                                    */
   705   /*                                                                                                     */
   706   /*  Output                                                                                             */
   707   /*                                                                                                     */
   708   /*    ga....    set to values found in the GAI ALLOC file                                              */
   709   /*                                                                                                     */
   710   /*    sfound set to '0' if values not found in the GAI ALLOC file                                      */
   711   /*    sfound set to '1' if values found in the GAI ALLOC file                                          */
   712   /*                                                                                                     */
   713   /*                                                                                                     */
   714   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
   715
   716   read_gai_alloc: nop;
   717
   718   address CMS
   719
   720   'getfmadr'
   721   pull aschedfilemode
   722   aschedfm = word(aschedfilemode,2)
   723
   724   'SET CMSTYPE MT'
   725   'ACC WSCVMO:WSCBRS.ASCHED' aschedfm   /* find the disk with the file   */
   726   'SET CMSTYPE RT'
   727
   728   fn = 'GAI'
   729   ft = 'ALLOCV2'
   730   fm = aschedfm
```


This Sheet Has Been Replaced

```
731   gaifn = fn
732   gaift = ft
733   gaifm = fm
734   rn = 0
735                                   /*  save location for report          */
736   'LISTF' gaifn gaift gaifm '(SEARCH LIFO'
737   parse upper pull . . . gaiodir
738   'LISTF' gaifn gaift gaifm ' (stack lifo alldates full dtolu'
739   parse upper pull . . . . . gailudate gailutime
740
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          741   pp = index(gaiodir,':')
          742   if pp <> 0 then
          743     do
 1        744       gaiodir = delstr(gaiodir,pp,1)
 1        745       gaiodir = insert('&gml.',gaiodir,pp-1)
          746     end
          747
          748   gacontract = ''
          749   gacdate = ''
          750   gaccust = ''
          751   gacpu = ''
          752   gadasd = ''
          753   gatape = ''
          754   ga3174 = ''
          755   gaid = ''
          756   gastart = ''
          757   gahours = ''
          758   gaprtr = ''
          759   gachext = ''
          760   gagatew = ''
          761   lastcontract = '9999999999999'
          762   sfound = '0'
          763
          764   /* debug
          765   say 'contract, date, starttime, customer searched = ',
          766    cnum  mm dd yyyy  sstart cusname
          767   */
          768
          769
          770   'SET CMSTYPE HT'
          771   'state' gaifn gaift gaifm
          772
          773   if rc = 0 then
          774     do forever
 1        775
 1        776     'SET CMSTYPE RT'
 1        777     if read() = 'EOF' then leave
 1        778       else do
 2        779
 2        780                                   /*  pick up contract number          */
 2        781         newcontract =  substr(inline,78,7)
```

This Sheet Has Been Replaced

```
2       783         newdate = substr(inline,26,8)   /* yyyymmdd now                */
2       784                                         /*  pick up start time        */
2       785         gayyyy  = substr(newdate,1,4)
2       786         gamm    = substr(newdate,5,2)
2       787         gadd    = substr(newdate,7,2)
2       788         if sstart <> '' then
2       789           newstart = substr(inline,35,4)
2       790         else
2       791           newstart = ''
```


This Sheet Has Been Replaced

```
Do Sal  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
 2        792
 2        793          /* pick up customer name                    */
 2        794        if scus <> '' then
 2        795          newcust    = strip(substr(inline,1,24))
 2        796        else
 2        797          newcust = ''
 2        798
 2        799          /*  see if we're doing list option          */
 2        800          /*    for specific contract                 */
 2        801        if listco <> 0 then
 3        802          do
 3        803
 3        804            scontract = cnum
 3        805            sdate = '??/??/????'
 3        806
 3        807            if cnum = newcontract then
 4        808              do
 4        809
 4        810                /* display only if different from last*/
 4        811                if  (newcontract = gacontract),
 4        812                  & (newdate = gacdate),
 4        813                  & (newstart = gastart),
 4        814                  & (newcust = gaccust) then nop
 4        815                else
 4        816                  say newcontract || ' ' newdate newstart newcust
 4        817
 4        818                gaccust    = newcust
 4        819                gacdate    = newdate
 4        820                gastart    = newstart
 4        821                gacontract = newcontract
 4        822
 4        823                sfound = '1'
 3        824
 3        825              end
 2        826
 2        827          end
 2        828        else
 2        829          /*  see if we're doing list option          */
 2        830          /*    for specific customer                 */
 2        831        if listcu <> 0 then
 3        832          do
```

213

This Sheet Has Been Replaced

```
3       833             scontract = '????????'
3       834             sdate = '??/??/????'
3       835
3       836             if index(newcust,cusname) <> 0 then
3       837               do
4       838
4       839                                 /* display only if different from last*/
4       840                 if  (newcontract = gacontract),
4       841                   & (newdate = gacdate),
4       842                   & (newstart = gastart),
```


This Sheet Has Been Replaced

```
4       843                 & (newcust = gaccust) then nop
4       844               else
4       845                 say newcontract || ' ' newdate newstart newcust
4       846
4       847               gaccust     = newcust
4       848               gacdate     = newdate
4       849               gastart     = newstart
4       850               gacontract = newcontract
4       851
4       852               sfound = '1'
4       853
3       854             end
2       855         end
2       856
2       857       else
2       858
2       859                                 /*  see if we're doing list option    */
2       860                                 /*    for specific date               */
2       861       if listda <> 0 then
3       862         do
3       863
3       864           scontract = '????????'
3       865           sdate = mm'/'dd'/'yyyy
3       866
3       867           if (gamm || '/' || gadd || '/' || gayyyy) = sdate then
4       868             do
4       869
4       870                                 /* display only if different from last*/
4       871               if  (newcontract = gacontract),
4       872                 & (newdate = gacdate),
4       873                 & (newstart = gastart),
4       874                 & (newcust = gaccust) then nop
4       875               else
4       876                 say newcontract || ' ' newdate newstart newcust
4       877
4       878               gaccust     = newcust
4       879               gacdate     = newdate
4       880               gastart     = newstart
4       881               gacontract = newcontract
4       882
4       883               sfound = '1'
```


This Sheet Has Been Replaced

```
3       884             end
3       885
2       886           end
2       887         else
2       888
2       889                                 /*  see if we're doing list option    */
2       890         if listal <> 0 then     /*    for odd CPUs                    */
2       891           do
3       892
3       893
```


This Sheet Has Been Replaced

```
=> Source Listing
Compiler REXX/370 3.0  LVL PQ28073    Time: 15:50:56          Date: 2000-02-03
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
  3       894            gacpu    = strip(substr(inline,95,15))
  3       895            extr = word(gacpu,1)
  3       896            if (extr <> ''),
  3       897              then
  3       898              do
  4       899
  4       900                                  /* display only if different from list*/
  4       901                odf = '0'
  4       902                do ods = 1 to words(odds)
  5       903                  if extr = word(odds,ods) then
  5       904                    do
  6       905                      odf = '1'
  5       906                    end
  4       907                end
  4       908
  4       909                if odf then
  4       910                  do
  5       911
  5       912  /*                  say newcontract || ' ' newdate newstart newcust extr
  5       913
  5       914  */
  5       915
  4       916                  end
  4       917                else
  4       918                    odds = odds extr
  4       919
  4       920                gaccust    = newcust
  4       921                gacdate    = newdate
  4       922                gastart    = newstart
  4       923                gacontract = newcontract
  4       924
  4       925                sfound = '1'
  4       926
  3       927              end
  3       928
  2       929            end
  2       930
  2       931          else
  2       932
  2       933                                  /*  must be looking for contract,     */
  2       934                                  /*    date, and maybe start time,     */
```


This Sheet Has Been Replaced

```
2       935                                 /*    and maybe customer name        */
2       936
2       937         if (newcontract = scontract),
2       938          & ((gamm || '/' || gadd || '/' || gayyyy) = sdate),
2       939          & (newstart = sstart),
2       940          & ( (index(newcust,scus) <> 0) & (scus <> '') | scus = '')
2       941          then
2       942           do
3       943
3       944             sfound = '1'
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
 3       945
 3       946            if lastcontract = newcontract then do
 4       947
 4       948              gadasd  = gadasd  || ' ' || strip(substr(inline,110,63))
 4       949              gatape  = gatape  || ' ' || strip(substr(inline,174,32))
 4       950              ga3174  = ga3174  || ' ' || strip(substr(inline,214,23))
 4       951              tacpu   = strip(substr(inline,95,15))
 4       952              if words(tacpu) < 1 then
 4       953                tacpu = tacpu 'CPU?'
 4       954              if words(tacpu) < 2 then
 4       955                tacpu = tacpu 'main?'
 4       956              if words(tacpu) < 3 then
 4       957                tacpu = tacpu 'exp?'
 4       958              gacpu   = gacpu tacpu
 4       959              gasuite = gasuite || ' ' || strip(substr(inline,62,15))
 4       960                                /*  only pick up unique event numbers */
 4       961              taevno     = strip(substr(inline,238,7))
 4       962              fndtae = '0'
 4       963              do tg = 1 to words(gaevno)
 5       964                if taevno = word(gaevno,tg) then
 5       965                  fndtae = '1'
 4       966              end
 4       967              if fndtae then
 4       968                nop
 4       969              else
 4       970                gaevno  = gaevno  || ' ' taevno
 4       971              gaprtr  = gaprtr  || ' ' || strip(substr(inline,246,17))
 4       972              gachext = gachext || ' ' || strip(substr(inline,264,17))
 4       973              gagatew = gagatew || ' ' || strip(substr(inline,282,8))
 4       974
 3       975            end
 3       976
 3       977          else do
 4       978
 4       979            lastcontract = newcontract
 4       980
 4       981            gaccust    = strip(substr(inline,1,24))
 4       982            gacdate    = strip(substr(inline,26,8))    /* yyyymmdd now */
 4       983            gastart    = strip(substr(inline,35,4))
 4       984            gahours    = strip(substr(inline,54,5))
 4       985            gasuite    = strip(substr(inline,62,15))
```


This Sheet Has Been Replaced

```
4   987                 [illegible] strip(substr(inline,78,7))
4   988     tacpu       = strip(substr(inline,95,15))
4   989     if words(tacpu) < 1 then
4   990       tacpu = tacpu 'CPU?'
4   991     if words(tacpu) < 2 then
4   992       tacpu = tacpu 'main?'
4   993     if words(tacpu) < 3 then
4   994       tacpu = tacpu 'exp?'
4   995     gacpu = tacpu
            gadasd      = strip(substr(inline,110,63))
```

gacontract = strip(substr(inline, 78, 7)

tacpu = strip(substr(inline, 95, 15))

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
 4       996             gatape     = strip(substr(inline,174,32))
 4       997             garcs      = strip(substr(inline,207,6))
 4       998             ga3174     = strip(substr(inline,214,23))
 4       999             gaevno     = strip(substr(inline,238,7))
 4      1000             gaprtr     = strip(substr(inline,246,17))
 4      1001             gachext    = strip(substr(inline,264,17))
 4      1002             gagatew    = strip(substr(inline,282,8))
 4      1003
 3      1004           end
 3      1005
 3      1006   /*
 3      1007           say ''
 3      1008           say 'Customer, rec# = ' gaccust rn
 3      1009           say 'dasd = ' gadasd
 3      1010           say 'tape = ' gatape
 3      1011           say '3174 = ' ga3174
 3      1012           say 'suite = ' gasuite
 3      1013           say 'cpu   = ' gacpu
 3      1014           say 'event = ' gaevno
 3      1015           say 'prtr  = ' gaprtr
 3      1016           say 'chext = ' gachext
 3      1017           say 'gatew = ' gagatew
 3      1018           say 'entire inline = ' inline
 3      1019   */
 3      1020
 2      1021           end /* matching contract/date/time/customer found          */
 2      1022
 2      1023       else
 2      1024         do
 3      1025                                  /* pass all the records            */
 3      1026           nop
 2      1027         end
 1      1028       end                        /* not EOF on file                 */
 1      1029
 1      1030
        1031   end                           /* end of do forever               */
        1032
        1033   'SET CMSTYPE RT'
        1034
        1035   if sfound then nop
        1036     else do
```


This Sheet Has Been Replaced

```
1       1037
1       1038        if sstart <> '' & sstart <> "SSTART" then
1       1039          dstartm = sstart
1       1040        else
1       1041          dstartm = 'N/A'
1       1042
1       1043        if scontract <> '' & scontract <> "SCONTRACT" then
1       1044          dscontr = scontract
1       1045        else
1       1046          dscontr = 'N/A'
```

This Sheet Has Been Replaced

```
Compiler REXX/370 3.0  Level 11Z00073   Time 10:00:00.00
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
  1      1047
  1      1048        if scus <> '' & scus <> "SCUS" then
  1      1049          dscus = scus
  1      1050        else
  1      1051          dscus = 'N/A'
  1      1052
  1      1053        say excfn 'did not find contract:' dscontr 'for date:' sdate,
  1      1054         'at time:' dstartm 'for customer:' dscus
  1      1055
         1056      end
         1057
         1058    'rel' aschedfm                  /*  release the disk              */
         1059
         1060    return                          /* end of read_gai_alloc routine  */
         1061
         1062
         1063    /*%PAGE */
```

Source Listing BRSRDSCH EXEC

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
      1064   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
      1065   /*                                                                                               */
      1066   /*  Read IOCPRECS                                                                                */
      1067   /*                                                                                               */
      1068   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
      1069
      1070   read_IOCPRECS: nop;
      1071
      1072   address CMS
      1073
      1074   'getfmadr'
      1075   pull brstfilemode
      1076   brstfm = word(brstfilemode,2)
      1077
      1078   'SET CMSTYPE HT'
      1079   'ACC MSCVMU:WSCDRS.BRSTOOLS' brstfm   /* find the disk with the file   */
      1080   'SET CMSTYPE RT'
      1081
      1082   'EXECIO * DISKR $VALID$ IOCPRECS' brstfm '(STEM IOCP.'
      1083
      1084   'LISTF $VALID$ IOCPRECS' brstfm '(SEARCH LIFO'
      1085   parse upper pull . . . iordir
      1086   svexcdir = excdir
      1087
      1088   'LISTF $VALID$ IOCPRECS' brstfm '(STACK LIFO ALLDATES FULL DTOLU'
      1089   parse upper pull . . . . . iordate iortime
      1090
      1091   'rel' brstfm                        /*  release the disk                */
      1092
      1093   return
      1094
      1095
      1096   /*%PAGE */
```


This Sheet Has Been Replaced

```
1097  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1098  /*                                                                                             */
1099  /*  Search IOCPRECS                                                                            */
1100  /*                                                                                             */
1101  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1102
1103  Search_IOCPRECS: procedure expose iocp. out. odx
1104
1105  parse upper arg location type .
1106
1107  if type = 'DASD' then
1108    tloc = tailstr(location,3,3)
1109  /*
1110    tloc = substr(location,1,3)
1111  */
1112
1113  if type = 'TAPE' then
1114    tloc = tailstr(location,3,3)
1115
1116  if type = '3174' then
1117    tloc = substr(location,2,3)
1118
1119  if type = 'PRTR' then
1120    tloc = tailstr(location,3,3)
1121
1122  svodx = odx
1123
1124  do jx = 1 to iocp.0
1125
1126
1127    if tloc      = word(iocp.jx,8) then      /*  location matches argument          */
1128  /*  if index(location,word(iocp.jx,8)) <> 0 then
1129
1130  */
1131      do
1132        odx = odx + 1
1133        daddr = substr(iocp.jx,30,4)
1134
1135        if substr(iocp.jx,1,1) <> '*' then  /*  don't translate comment lines       */
1136
1137          daddr   = translate(daddr, 'abcdefghij', '0123456789')  /*  translate so hex sort works  */
```

This Sheet Has Been Replaced

```
2       1138
2       1139          out.odx = substr(iocp.jx,1,29) || daddr || substr(iocp.jx,34)
1       1140        end
1       1141
        1142    end
        1143
        1144    if odx <> svodx then nop
        1145      else
        1146      do
1       1147        cmt =  '*** ' type location 'not in $VALID$ IOCPRECS ***'
```

This Sheet Has Been Replaced

2001012947
02 MAR 2001

Source Listing


```
1       1148        odx = odx + 1
1       1149        out.odx = cmt
1       1150        say cmt
        1151      end
        1152
        1153    return
        1154
        1155
        1156    /*%PAGE */
```

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          1157  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
          1158  /*                                                                                           */
          1159  /*  READ Subroutine                                                                          */
          1160  /*                                                                                           */
          1161  /*                                                                                           */
          1162  /*
          1163  ** Read one line of input from file FN+FT+FN into variable INLINE,
          1164  ** return to caller with '0' in case of success or with 'EOF' if
          1165  ** end of file was encountered.
          1166  ** If an error was encountered during the read operation then
          1167  ** write the appropriate message, close input, and stop
          1168  ** the program with return code 16.
          1169  **
          1170  ** arg: none
          1171  **
          1172  ** out: 0         successfully read
          1173  **      'EOF'     end of file
          1174  **
          1175  ** ref: none
          1176  **
          1177  ** set: INLINE    next input record
          1178  */
          1179  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
          1180
          1181  Read: nop;
          1182
          1183      cmd='EXECIO 1 DISKR' fn ft fm '( VAR INLINE'
          1184      cmd
          1185      If rc=2 Then
          1186        Return 'EOF'
          1187      If rc<>0  Then
          1188        Do
1         1189          say excfn 'RC = ' rc ' CMD = ' cmd
1         1190          'FINIS' fn ft fm
1         1191          'dropbuf' svmb
1         1192          exit 16
          1193        End
          1194      rn = rn + 1                    /* count records in this file        */
          1195      RETURN 0
          1196
          1197
```


This Sheet Has Been Replaced

This Sheet Has Been Replaced



```
1199  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1200  /*                                                                                       */
1201  /*  Function: TAILSTR                                                                    */
1202  /*                                                                                       */
1203  /*    SUBSTR function, except it starts from the end of a string                         */
1204  /*    rather than the beginning                                                          */
1205  /*                                                                                       */
1206  /*    Arguments:                                                                         */
1207  /*                                                                                       */
1208  /*       instr - input string of any length                                              */
1209  /*       st    - start of string, using end byte as 1                                    */
1210  /*       lt    - length of string                                                        */
1211  /*                                                                                       */
1212  /*  Changes                                                                              */
1213  /*                                                                                       */
1214  /*     10/25/97  - Original by S.W.Wood                                                  */
1215  /*                                                                                       */
1216  /*                                                                                       */
1217  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1218
1219  tailstr: procedure
1220
1221   parse arg instr, st, lt .
1222
1223   li = length(instr)
1224   rl = min(lt,li)
1225   rs = li - st + 1
1226  /*
1227   rs = li - rl + 1     /* here, the start is the beginning of the str. */
1228  */
1229   xxyy = substr(instr,rs,rl)
1230
1231  return xxyy
1232
1233
1234  /*%PAGE */
```

230

This Sheet Has Been Replaced

```
      1235  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
      1236  /*                                                                   */
      1237  /*  SET_DIRMSG                                                       */
      1238  /*                                    ensure no line longer than 'mlth'  */
      1239  /*                                                                   */
      1240  /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
      1241
      1242  set_dirmsg: procedure expose hdx hdr.
      1243
      1244  parse upper arg textstr ':' list
      1245
      1246  mlth = 64
      1247  amsg = ''
      1248  stop = '0'
      1249
      1250  vmmsg = list
      1251
      1252  do until words(vmmsg) = 0
1     1253
1     1254    do xx = 1 to 9999 until stop | (length(amsg) > mlth)
2     1255
2     1256      amsg = amsg || word(vmmsg,xx) || ','
2     1257      if words(vmmsg) <= xx  then
2     1258        stop = '1'
2     1259
1     1260    end
1     1261
1     1262                                  /*  remove final comma              */
1     1263    amsg  = delstr(amsg,lastpos(',',amsg))
1     1264                                  /*  reset vmmsg                     */
1     1265    vmmsg = subword(vmmsg,xx + 1)
1     1266
1     1267    hdx = hdx + 1
1     1268    hdr.hdx = '*' textstr || ':' amsg
1     1269
1     1270    amsg = ''
1     1271
      1272  end
      1273
      1274  return
      1275
```

This Sheet Has Been Replaced

This Sheet Has Been Replaced



1276
1277
/* BDR2MB */

```
1276   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1279   /*                                                                       */
1280   /*  Sort_File routine                                                    */
1281   /*                                                                       */
1282   /*  Sort proposed IOCPRECS file by                                       */
1283   /*    BRSADDR field 30-33                                                */
1284   /*                                                                       */
1285   /*                                                                       */
1286   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1287
1288   Sort_File: nop
1289
1290                                      /*  sort the created file into temp   */
1291     'SET CMSTYPE HT'
1292     queue '30 33'
1293     'sort' oxn oxt 'A' oxn oxt 'A'
1294     'SET CMSTYPE RT'
1295
1296   return
1297
1298   /*%PAGE */
```


This Sheet Has Been Replaced

> source listing

BRSRDSCH EXEC

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
1299   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1300   /*                                                                                                       */
1301   /*  Say_Parms routine                                                                                    */
1302   /*                                                                                                       */
1303   /*  Explain routine options to caller                                                                    */
1304   /*                                                                                                       */
1305   /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1306
1307   Say_Parms: nop
1308
1309       say ''
1310       say 'Build VM directory IOCP records for customer test,'
1311       say '  or query WSCVM copy of BRS schedule records'
1312       say ''
1313       say 'Syntax is:' excfn 'ContractNumber ContractDate <ContractTime>',
1314   '<CustomerName>'
1315       say '                <(option>'
1316       say '  where "option" is one of the following:'
1317       say ''
1318       say '  <LISTCOntract>  for list of tests for a specific contract'
1319       say '  <LISTDAte>      for list of tests for a specific date'
1320       say '  <LISTCUstomer>  for list of tests for a specific customer'
1321       say '  <LISTALl>       for list of CPUs in scheduler file'
1322       say '  <NOPRint>       to NOT print hardcopy of IOCPRECS built'
1323       say '  <FN=filename>   to force filename'
1324       say ''
1325       say "  Date format must be 'mm/dd/yyyy'"
1326       say ''
1327       say "  Time format must be 'hhmm', if present"
1328       say ''
1329       say 'Examples:'
1330       say ''
1331       say '    'excfn'   R31417   8/3/1998          /* build IOCP records                    */'
1332       say '    'excfn'   SVS  4/1/1999  custname  /* build IOCP records for "custname"  */'
1333       say '    'excfn'   R31417   8/3/1998 (NOPR  /* build IOCP recs, no print          */'
1334       say '    'excfn'   R31417   (LISTCO         /* search for contracts               */'
1335       say '    'excfn'   8/3/1998 (LISTDA         /* search for dates                   */'
1336       say '    'excfn'   custname (LISTCU         /* search for customer                */'
1337       say '    'excfn'   xxxxx    (LISTAL         /* search for CPUs                    */'
1338       say ''
1339      'dropbuf' svmb
```

This Sheet Has Been Replaced

```
1340        EXIT 0
1341
1342
1343    /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
1344    /*                                                                                           */
1345    /*        **  END OF BRSRDSCH EXEC **                                                       */
1346    /*                                                                                           */
1347    /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
```

This Sheet Has Been Replaced

: Lines 1347

of compiled program in bytes 53088

1 messages	Informational	Warning	Error	Severe	Terminating
0	0	0	0	0	0

shing time of compilation: 15:50:57

Compilation statistics

BRSRPSCH EXEC

This Sheet Has Been Replaced

APPENDIX F

This Sheet Has Been Replaced



IM Compiler REXX/370 3.0 LVL PQ28073 Time: 15:46:24 Date: 2000-02-03 Page: 1

==> Compilation Summary MJFACG EXEC G1

message(s) reported. Highest severity code was 4 - Warning

Compiler Options

```
  NOALTERNATE
    CEXEC    (MJFACG    CEXEC     G1)                    RECFM=F,LRECL=1024
  NOCOMPILE  (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG     (I)
  NOIEXEC
    LIBLEVEL (*)
    LINECOUNT (55)
    MARGINS  (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT    (MJFACG    LISTING   G1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN    (MJFACG    EXEC      G1   §   U   §   \ \-  \ D) RECFM=V,LRECL=163
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 4
```


This Sheet Has Been Replaced

```
 Compiler REXX/370 3.0  LVL PQ26073    Time: 15:48:24                Date: 2000-02-03                Page:    2
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
 ==> Source Listing                    MJFACG EXEC G1
       1  /*
       2  /*        MJFACG   EXEC - A Rexx procedure to generate VM directory    */
       3  /*                        skeletons for BRS                            */
       4  /*                                                                     */
       5  /*        (c) 1998 IBM GS and M.J. Firley                              */
       6  /*                                                                     */
       7  /*                                                                     */
       8  /*        Called by       :                                            */
       9  /*                                                                     */
      10  /*  Change History:                                                    */
      11  /*                                                                     */
      12  /*  VER YYMMDD WHO DESCRIBE                                            */
      13  /*  --- ------ --- ----------------------------------------------------*/
      14  /*                                                                     */
      15  /*                                                                     */
      16                Address CMS;
      17                Trace Value Off
 +FANGA0550W Unsupported TRACE options will default to OFF
      18                SeparatorRecord = "* "
      19
      20                Parse Upper Arg Argstring
      21                Parse Var Argstring . "(" OperationalModeOption
      22                OperationalModeOption = Left(OperationalModeOption,1)
      23                Select
      24                    When OperationalModeOption = "L" then Do
      25                         LineMode = 1
      26                         Parse Var Argstring BuildVMDataFileID "("
      27                         "PIPE CMS LISTFILE" BuildVMDataFileID "|STACK"
      28                         Parse Upper Pull SFname SFtype SFmode .
      29                         Select
      30                             When Left(SFName,6,' ') = "DMSLST" then do
      31                                  "VMFCLEAR"
      32                                  Say "MJFACG028E Missing control file:" ,
      33                                      BuildVMDataFileID
      34                                  Exit 28
      35                                  end;
      36                             Otherwise nop;
      37                         end;
      38                         End;
      39                    Otherwise LineMode = 0
```

This Sheet Has Been Replaced

```
40
41
42
43
44
45
46
47
48
49
```

```
End;
ForceReshow
"PIPE CMS IDENTIFY"
    " | Stem Identity."
Parse Var Identity . ThisUser "AT" ThisNode "VIA" ConnectName
TodaysDate = Date(S)
ThisYear   = Left(TodaysDate,4)
ThisMonth  = Substr(TodaysDate,2)
```

F3

This Sheet Has Been Replaced

```
                50              ThisDay     = Right(TodaysDate,2)
                51
                52              Call ProcessProfile
                53              Select
          1     54                  When LineMode = 0 then nop;
          1     55                  Otherwise do
    1     2     56                       "PIPE DISK" BuildVMDataFileID ,
    1     2     57                           "| STEM BuildVMDataStack."
    1     2     58                       Do InputControlRecord = 1 to BuildVMDataStack.0
    2     2     59                           Interpret BuildVMDataStack.InputControlRecord
    1     2     60                       end;
    1     2     61                       Call ProcessSelection
    1     2     62                       Exit
          2     63                       End;
                64              End;
                65              Call ShowPanel
                66              Exit
                67
                68
                69  ShowPanel:   Nop;
                70               D1 = Left(' ',79,' ')
                71               D2 = '0000000A'
                72               D3 = Left(' ',70,' ')
                73               D4 = Left(' ',8,' ')
                74               D5 = Left(' ',2,' ')
                75               D6 = Left('GUESTSYS',8,' ')
                76               D7 = Left('NNNNnnnn',8,' ')
                77               D8 = Left('EFG',8,' ')
                78               D9 = Right('256',4,'0')
                79               D10= Right('512',4,'0')
                80               D11= Right('1024',4,'0')
                81               D12= Right('1',2,'0')
                82               D13= "ESA"
                83               D14= "0000"
                84               D15= 'PARMLIST'
                85               D16= 'N'
                86               D17= 'starter '
                87               D18= 'profile '
                88               D19= 'Y'
                89               D20= 'Y'
                90               D21= 'Y'
```


This Sheet Has Been Replaced

```
92                  D23= 'D'
93                  CursorPos = "D2"
94                  "EUDEXEC2"
95                  Address DISPLAY;
96  RedisplayPanel: Nop;
97                  ForceReshow    = 0
98                  Address DISPLAY;
99                  "USE PANEL MJFACG10"
100                 "CURSOR" CursorPos
```

91

D22 = 'A'

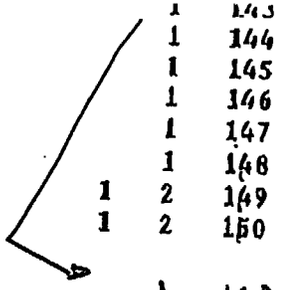
This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                101                "DISPLAY (LOAD UNLOAD"
                102                DisplayMessage = Left(' ',80,' ')
                103                ContractNumber = D2
                104                ContractName   = D3
                105                PathMapFileName= D4
                106                PathMapFileMode= D5
                107                GuestLogonID   = D6
                108                AccountCode    = D7
                109                PrivilegeVector= D8
                110                CStorDefault   = D9
                111                CStorMax       = D10
                112                XStorDefined   = D11
                113                NumberOfCPUs   = D12
                114                MachineMode    = D13
                115                IPLAddress     = D14
                116                IPLParmList    = D15
                117                IncludeStarter = D16
                118                IncludeFileName= D17
                119                IncludeFileType =D18
                120                IncludeDASD    = D19
                121                IncludeTape    = D20
                122                Include3270    = D21
                123                IncludePrinters= D22
                124                DedicatedMDISKS= D23
                125                DI = Left(' ',79,' ')
                126                Select
          1     127                   When Rstatus = "ENTER" then do
1         2     128                      Call CheckInputs
1         2     129                      Select
1         3     130                         When ForceReshow = 1 then Signal RedisplayPanel
1         3     131                         Otherwise nop;
1         2     132                      End;
1         2     133                      Call ProcessSelection
1         2     134                      Address DISPLAY;
          2     135                      end
          1     136                   When Rstatus = "PF1"  then Call HelpRtn
          1     137                   When Rstatus = "PF13" then Call HelpRtn
          1     138                   When Rstatus = "PF3"  then Call ExitRtn 0
          1     139                   When Rstatus = "PF15" then Call ExitRtn 0
          1     140                   When Rstatus = "PF7"  then Call ExitRtn 8
          1     141                   When Rstatus = "PF19" then Call ExitRtn 8
```

This Sheet Has Been Replaced

```
1   143
1   144   When Rstatus = "PF20" then Call ExitRtn P
1   145   When Rstatus = "PF12" then Call ExitRtn 0
1   146   When Rstatus = "PF24" then Call ExitRtn 0
1   147   When Rstatus = "PF1"  then Call HelpRtn
1   148   When Rstatus = "PF13" then Call HelpRtn
1   2 149 Otherwise do
1   2 150       01 = "You pressed:" Rstatus " That is not an valid selection. Please retry."
              Signal RedisplayPanel

    1 142
```



When Rstatus = "PF8" then Call ExitRtn F

This Sheet Has Been Replaced

Source Listing

```
! Compiler REXX/370 3.0  LVL PQ28073     Time: 15:48:24          MJFACG   EXEC     G1
: Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                                                                Date: 2000-02-03                              Page:    5
   1   2   151                          Call ExitRtn 4
                                        |
FANGA00773I Instruction may never be executed
       2   152                          End;
           153                  End;
           154                  Address CMS;
           155   ShowPanelEnd: Return;
           156
           157   ProcessSelection: Nop;
           158                    Address CMS;
           159                    MyTime = Time(R);
           160                    MyTime = Time(E);
           161                    Call BuildDirProlog
           162                    Select
       1   163                       When IncludeStarter = "Y" then do
   1   2   164                             Address CMS "PIPE DISK" IncludeName IncludeType "*"
   1   2   165                                 "|>>" GuestLogonID "PROTODIR" PathMapFileMode
       2   166                             end
       1   167                       Otherwise nop;
           168                    End;
           169                    Select
       1   170                       When IncludeDASD = "Y" then do
   1   2   171                             Call BuildDASD
       2   172                             end
       1   173                       Otherwise nop;
           174                    End;
           175                    Select
       1   176                       When IncludeTape = "Y" then do
   1   2   177                             Call BuildTape
       2   178                             end
       1   179                       Otherwise nop;
           180                    End;
           181                    Select
       1   182                       When IncludePrinters = "Y" then do
   1   2   183                             Call BuildPrinters
       2   184                             end
       1   185                       Otherwise nop;
           186                    End;
           187                    Select
       1   188                       When Include3270 = "Y" then do
   1   2   189                             Call BuildCommDevs
```

This Sheet Has Been Replaced

```
191
192
193
194
195
196
197
198
199
```

```
        Otherwise nop;
End;
Call BuildResidueList
ElapsedTime = Time(E)
ElapsedTime = Format(ElapsedTime,4,0)
PercentConverted = ((InRecords-Residue)/Inrecords)*100
PercentConverted = Format(PercentConverted,3,0)
Select
     When LineMode = 0 then do
```

190

End

This Sheet Has Been Replaced

```
Source Listing
M Compiler REXX/370 3.0  LVL PQ28073     Time: 15:46:24          MJFACG  EXEC     G1
f  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                                                                Date: 2000-02-03                                Page:    6
    1   2   200                          D1 = "MJFACG920I Processing Complete. Time=" ElapsedTime "sec." "Input="||
          C InRecords "Residue="||Residue PercentConverted||"%"
    1   2   201                          Signal RedisplayPanel
        2   202                         end;
        1   203                       Otherwise do
    1   2   204                          Say  "MJFACG920I Processing Complete. Time=" ElapsedTime "sec." "Input="||
          C InRecords "Residue="||Residue PercentConverted||"%"
        2   205                       end;
            206                   End;
            207   ProcessSelectionEnd: Return;
            208
            209   SelectMapFile: Nop;
            210                  Address CMS;
            211                  "CP SLEEP 02 SEC"
            212                  "VMFCLEAR"
            213                  Say "MJFACG110I You selected:" PathMapFileName "PATHMAPS" PathMapFileMode
            214                  GoodAnswer = 0
            215                  Do until GoodAnswer
    1       216                     Say "MJFACG111R OK to proceed? <Y|N>"
    1       217                     Parse Upper Pull Answer
    1       218                     ThisAnswer = Left(Space(Answer,0),1)
    1       219                     Select
    1   1   220                         When ThisAnswer = "Y" | ThisAnswer = "N" then Goodanswer = 1
    1   1   221                         Otherwise iterate
    1       222                     end;
            223                  End
            224                  Select
        1   225                     When ThisAnswer = "Y" then nop;
        1   226                     Otherwise do
    1   2   227                          D1 = "MJFACG021E Path Map definition file selection terminated."
    1   2   228                          CursorPos = "D4"
    1   2   229                          Signal RedisplayPanel
        2   230                          end;
            231                End;
            232               Address CMS "STATEW" PathMapFileName "PATHMAPS" PathMapFileMode
            233               Rcs = rc
            234               Select
        1   235                   When RCS = 0 then do
    1   2   236                       Say   "MJFACG100I Path Map definition file selected."
        2   237                       end;
        1   238                   Otherwise do
```

This Sheet Has Been Replaced

```
1   2   240                    ... "MJFACG128E IOCP definition file missing. Respecify."
1   2   241                    CursorPos = "D4"
    2   242                    Signal RedisplayPanel
        243                    end;
        244                End;
        245 SelectMapFileEnd: Return;
        246 /*
        247 /*                                                          */
        248 ExitRtn:     Parse Upper Arg Retcode .                       */
```

239

D1 = "MJFACG128E IOCP definition file missing. Respecify"

248

This Sheet Has Been Replaced

```
                249                 Select
          1     250                      When Retcode = "B" then do
      1   2     251                           Retcode = 0
      1   2     252                           Queue "MJFACM"
          2     253                           end;
          1     254                      When Retcode = "P" then do
      1   2     255                           Retcode = 0
      1   2     256                           Queue "XEDIT" GuestLogonID "PROTODIR *"
          2     257                           end;
          1     258                      Otherwise nop;
                259                 End;
                260 ExitRtnEnd:     Exit Retcode
                261 /*
                262 CheckInputs:    Nop;                                          */
                263                 Call CheckContractNumber
                264                 Call CheckContractName
                265                 Call CheckGuestID
                266                 Call CheckPrivClasses
                267                 Call CheckStorageLimits
                268                 Call CheckMachineMode
                269                 Call CheckCPUCount
                270                 Call CheckIncludeFile
                271                 Call SelectMapFile
                272                 Call CheckDASDBuild
                273 CheckInputsEnd: Return;
                274
                275 HelpRtn:        Nop;
                276                 Address CMS;
                277                 "HELP CONFIGURE"
                278                 Address DISPLAY
                279                 ForceReshow = 1
                280                 Signal RedisplayPanel
                281 HelpRtnEnd:    Return;
                282
                283 CheckStorageLimits: Nop;
                284                 Select
          1     285                      When  ((CstorDefault < CstorDefaultLow)|(CstorDefault>CStorDofaultHigh)) then do
      1   2     286                           CursorPos = "D9"
      1   2     287                           DI = "MJFACG090E Invalid default central storage value must be 16M < CS < 2048M"
          2     288                           Signal RedisplayPanel
                289                      end;
```

This Sheet Has Been Replaced

```
                                    CursorPos = "D9"
1   2   292                         D1 = "MJFACG093E Invalid central storage values: Default < Max required.      "
1   2   293                         Signal RedisplayPanel
    2   294                     end;
    1   295                     When (CstorMax      < CstorDefaultLow | CstorMax >   CStorDefaultHigh) then do
1   2   296                         CursorPos = "D10"
1   2   297                         D1 = "MJFACG091E Invalid maximum central storage value must be 16M < CS < 2048M"
1   2   298                     Signal RedisplayPanel
    2   299                 end;

    1   290

when CstorMax < CstorDefault then do
```


This Sheet Has Been Replaced

```
      1   300                When ((XStorDefined < XStorDefinedLow)|(XStorDefined > XStorDefinedHigh)) then do
    1 2   301                    CursorPos = "D11"
    1 2   302                    D1 = "MJFACG092E Invalid expanded storage value:       must be  0M < XS < 2048M"
    1 2   303                    Signal RedisplayPanel
      2   304                end;
      1   305                Otherwise nop;
          306             End;
          307   Trace Value Off;
                              |
FANGA0055OW Unsupported TRACE options will default to OFF
          308   CheckStorageLimitsEnd: Return;
          309
          310   CheckCPUCount: Nop;
          311             Select
      1   312                 When (NumberOfCPUs < NumberOfCPUsMin | NumberOfCPUs > NumberofCPUsMax) then do
    1 2   313                     CursorPos = "D12"
    1 2   314                     D1 = "MJFACG120E Invalid number of processors. Value must be 1 < CPUs < 10"
    1 2   315                     Signal RedisplayPanel
      2   316                 end;
      1   317                 Otherwise nop;
          318             End;
          319   CheckCPUCountEnd: Return;
          320
          321   CheckContractName: Nop;
          322             Select
      1   323                 When Space(ContractName,0) = "" then do
    1 2   324                     CursorPos = "D3"
    1 2   325                     D1 = "MJFACG030E Invalid contract name. Cannot be blank."
    1 2   326                     Signal RedisplayPanel
      2   327                 end;
      1   328                 Otherwise nop;
          329             End;
          330   CheckContractNameEnd: Return;
          331
          332   CheckContractNumber: Nop;
          333             Select
      1   334                 When Space(ContractNumber,0) = "" then do
    1 2   335                     CursorPos = "D3"
    1 2   336                     D1 = "MJFACG020E Invalid contract number. Cannot be blank."
    1 2   337                     Signal RedisplayPanel
      2   338                 end;
```


This Sheet Has Been Replaced

```
1   2   340                When Datatype(ContractNumber,'A') <> 1 then do
1   2   341                    CursorPos = "D3"
1   2   342                    D1 = "HJFACG021E Invalid contract number. Only A-Z,a-z,0-9 allowed."
    2   343                    Signal RedisplayPanel
    1   344                end;
        345                Otherwise nop;
        346            End;
        347    CheckContractNumberEnd: Return;
        348
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          349   CheckGuestID: Nop;
          350               GuestLogonID = Strip(GuestLogonID,'B',' ')
          351               Select
       I  352                   When Space(GuestLogonID,0) = "" then do
    1  2  353                       CursorPos = "D6"
    I  2  354                       D1 = "MJFACG060E Invalid guest userid. Cannot be blank."
    1  2  355                       Signal RedisplayPanel
       2  356                   end;
       1  357                   When Datatype(GuestLogonID,'A') <> 1 then do
    1  2  358                       CursorPos = "D6"
    1  2  359                       D1 = "MJFACG061E Invalid guest userid. Only A-Z,a-z,0-9 allowed."
    1  2  360                       Signal RedisplayPanel
       2  361                   end;
       I  362                   Otherwise nop;
          363               End;
          364   CheckGuestID: Return;
                |
FANPAR0071W Duplicate label: Only first occurrence on line 349 used
          365
          366   CheckPrivClasses: Nop;
          367               PrivilegeVector = Strip(PrivilegeVector,'B',' ')
          368               Do DefinedClass = 1 to Length(PrivilegeVector)
    I     369                  ThisClass = Substr(PrivilegeVector,DefinedClass,1)
    1     370                  Select
    1  1  371                     When Pos(ThisClass,AllowedPrivClasses) > 0 then nop;
    1  1  372                     otherwise do
    2  2  373                          CursorPos = "D8"
    2  2  374                          D1 = "MJFACG080E Invalid privilege classes. See installation documentation."
    2  2  375                          Signal RedisplayPanel
    1  2  376                          end;
    1     377                  End;
          378               End;
          379   CheckPrivClassesEnd: Return;
          380
          381   CheckMachineMode: Nop;
          382               MachineMode = Space(MachineMode,0)
          383               Select
       1  384                    When WordPos(MachineMode,AllowedMachineModes) > 0 then nop
       1  385                    Otherwise do
    1  2  386                       CursorPos = "D13"
    1  2  387                       D1 = "MJFACG130E Invalid machine setting. Check list for allowed values."
```


This Sheet Has Been Replaced

```
1   2   388             Signal RedisplayPanel
    2   389             end
        390         End;
        391         Select
    1   392             When MachineMode = "370" then do
1   2   393                 Address CMS "VMFCLEAR"
1   2   394                 Say "MJFACG132W Warning! Not all processors allow Mode=370."
1   2   395                 Say "MJFACG133R Press <ENTER> to continue."
1   2   396                 Parse Upper Pull .
    2   397             end;
```


This Sheet Has Been Replaced

```
> Source Listing                                MJFACG   EXEC     G1
 Compiler REXX/370 3.0  LVL PQ28073    Time: 15:48:24                Date: 2000-02-03                Page:    10
  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                 |
       1   398                 Otherwise nop;
           399               End;
           400   CheckMachineMode: Return;
                 |
FANPAR0071W Duplicate label: Only first occurrence on line 381 used
           401
           402   CheckIncludeFile: Nop;
           403             Select
       1   404                 When IncludeStarter = "Y" then do
   1   2   405                     Address CMS "STATE" IncludeFileName IncludeFileType "*"
   1   2   406                     RcST = Rc
   1   2   407                     Select
   1   3   408                         When RcST = 0 then nop;
   1   3   409                         Otherwise do
   2   4   410                             CursorPos = "D18"
   2   4   411                             D1 = "MJFACG028E Include file:'"||IncludeFileName IncludeFileType||"' no
             C t found."
   2   4   412                             Signal RedisplayPanel
   1   4   413                             end;
   1   2   414                     End;
       2   415                 end;
       1   416                 Otherwise nop;
           417             End;
           418   CheckIncludeFileEnd: Return;
           419
           420   BuildDirProlog: Nop;
           421             Address CMS;
           422             Userid = "USER" Left(GuestLogonID,8,' ')
           423             Password = Left(ContractNumber,6,' ')
           424             Storage = CStorDefault||"M"  CStorMax||"M"
           425             VMDescriptor  = Userid Password Storage PrivilegeVector
           426             AccountInfo   = " ACCOUNT" AccountCode "DIST"
           427             ChangeRecord  = "*" Userid() Date(S) Time() "MJFACG Build"
           428             CustomerRecord= "*" ContractName
           429             AuthorRecord  = "*" Userid() Date(S) Time()
           430             NumberOfOptionLines = 3
           431             BaseOptions.1 = " OPTION NODEDICATE QUICKDSP MAINTCCW RMCHINFO TODENABLE"
           432             BaseOptions.2 = " OPTION CFUSER"
           433             BaseOptions.3 = " STDEVOPT LIBRARY CTL"
           434             IPLRecord     = " IPL" IPLAddress "PARM" IPLParmList
           435
```

This Sheet Has Been Replaced

```
497         "PIPE Var VMDescriptor    | >" GuestLogonID "PROTODIR" PathMapFileMode
498         "PIPE Literal * |          >>" GuestLogonID "PROTODIR" PathMapFileMode
499         "PIPE Var ChangeRecord    |>>" GuestLogonID "CHNGHIST" PathMapFileMode
440         "PIPE Var AccountInfo     |>>" GuestLogonID "PROTODIR" PathMapFileMode
441         "PIPE Var CustomerRecord  |>>" GuestLogonID "PROTODIR" PathMapFileMode
442         "PIPE Var AuthorRecord    |>>" GuestLogonID "PROTODIR" PathMapFileMode
443         "PIPE Literal * |          >>" GuestLogonID "PROTODIR" PathMapFileMode
444
445         Do Cpuid = 1 to NumberOfCPUs
```

This Sheet Has Been Replaced

```
=> Source Listing                                MJFACG   EXEC     G1
( Compiler REXX/370 3.0  LVL PQ26073    Time: 15:48:24                Date: 2000-02-03                Page:   11
: Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
   1       446                 CPURecord = ' CPU ' Right(Cpuid,2,'0')
   1       447                 "PIPE Var CPURecord     |>>" GuestLogonID "PROTODIR" PathMapFileMode
           448             End;
           449             MachineType   = " MACHINE" MachineMode NumberOfCPUs
           450
           451             "PIPE Var MachineType  |>>" GuestLogonID "PROTODIR" PathMapFileMode
           452             Do OptionLine = 1 to NumberOfOptionLines
   1       453                 "PIPE Var BaseOptions.OptionLine  |>>" GuestLogonID "PROTODIR" PathMapFileMode
           454             End;
           455
           456             Select
       1   457                 When XStoreDefined > 0 then do
   1   2   458                     XStoreRecord = " XSTORE" XStorDefined||"M"
   1   2   459                 "PIPE Var XStoreRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
       2   460                 end
       1   461                 Otherwise nop;
           462             End;
           463
           464             "PIPE Var IPLRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
           465             "PIPE Var SeparatorRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
           466 BuildDirPrologEnd: Return;
           467
           468 BuildDASD: Nop
           469             Select
       1   470                 When DedicatedMDISKs = "D" then call BuildDedicatedDASD
       1   471                 Otherwise call BuildSharedDASD
           472             End;
           473 BuildDASDEnd: Return;
           474
           475 BuildTape: Nop
           476             Address CMS;
           477             Do TapeType = 1 to Words(TapeTypes)
   1       478                 OldCtlGroup  = '0000'
   1       479                 ThisType = Word(TapeTypes,TapeType)
   1       480                 TypeHeader = "'* " ThisType "DEVICES"
   1       481                 "PIPE Var  TypeHeader |>>" GuestLogonID "PROTODIR" PathMapFileMode
   1       482                 Say "MJFACG901I Processing TAPE records for" ThisType
   1       483                 Do Record = 1 to DevMap.0
   2       484                     Select
   2   1   485                         When word(DevMap.Record,2) = ThisType then do
   3   2   486                             Parse Var DevMap.Record . "MAPS_TO" . . TargetAddress .
```

257

This Sheet Has Been Replaced

```
3   2   487                    TargetAddress = Right(Right(Space(TargetAddress,0),4),5,' ')
3   2   488                    NewCtlGroup  =Word(DevMap.Record,5)
3   2   489                    Select
3   3   490                          When (NewCtlGroup = OldCtlGroup ) then nop;
3   3   491                          Otherwise do
4   4   492                                "PIPE Var SeparatorRecord |>>" GuestLogonID "PROTODIR" Path
                 ! C MapFileMode
3   4   493                                end
3   2   494                    End;
3   2   495                    DeviceRecord = " DEDICATE" Right(Right(Space(Word(DevMap.Record,3),0),4)
```


This Sheet Has Been Replaced

```
=> Source Listing                              MJFACG   EXEC     G1
| Compiler REXX/370 3.0  LVL PQ28073    Time: 15:48:24                Date: 2000-02-03                Page:   12
  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0

                 C ,5,' ') TargetAddress "*" ThisType "CTLR" Word(DevMap.Record,5)
    3   2   496                                "PIPE Var DeviceRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
    3   2   497                                OldCtlGroup  = Word(DevMap.Record,5)
    2   2   498                                end;
    2   1   499                             Otherwise nop;
    2       500                          end;
    2       501                          Select
    2   1   502                                When (Record // 5000) = 0  then do
    3   2   503                                  Say "MJFACG900I" Right(Record,6,' ') "Device records processed."
    2   2   504                                  end;
    2   1   505                                Otherwise nop;
    2       506                          end;
    1       507                      end;
    1       508                      TypeTrailer = "* END " ThisType "DEVICES"
    1       509                      "PIPE Var TypeTrailer |>>" GuestLogonID "PROTODIR" PathMapFileMode
            510                  end;
            511   BuildTapeEnd: Return;
            512
            513   BuildPrinters: Nop
            514                  Address CMS;
            515                  "PIPE DISK" PathMapFileName "PATHMAPS" PathMapFileMode ,
            516                      "| Stem DevMap."
            517                  Do PrinterType = 1 to Words(Printypes)
    1       518                      ThisType = Word(Printypes,PrinterType)
    1       519                      TypeHeader = "* " ThisType "DEVICES"
    1       520                      "PIPE Var  TypeHeader |>>" GuestLogonID "PROTODIR" PathMapFileMode
    1       521                      Say "MJFACG901I Processing PRNT records for" ThisType
    1       522                      Do Record = 1 to DevMap.0
    2       523                          Select
    2   1   524                              When word(DevMap.Record,2) = ThisType then do
    3   2   525                                  Parse Var DevMap.Record . "*MAPS_TO" . . TargetAddress .
    3   2   526                                  TargetAddress = Right(Right(Space(TargetAddress,0),4),5,' ')
    3   2   527                                  DeviceRecord = " DEDICATE" Right(Right(Space(Word(DevMap.Record,3),0),4)
                 C ,5,' ') TargetAddress "*" ThisType "CTLR" Word(DevMap.Record,5)
    3   2   528                                  "PIPE Var DeviceRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
    2   2   529                                  end;
    2   1   530                              Otherwise nop;
    2       531                          end;
    2       532                          Call PostProgressMsg
    1       533                      end;
    1       534                      TypeTrailer = "* END " ThisType "DEVICES"
```

259

This Sheet Has Been Re-...

```
595                 "PIPE Var TypeTrailer |>>" GuestLogonID "PROTODIR" PathMapFileMode
596             end;
597 BuildPrintersEnd: Return;
598
599 BuildCommDevs: Nop
600             Address CMS;
601             "PIPE DISK" PathMapFileName "PATHMAPS" PathMapFileMode ,
602                 "| Stem DevMap."
603             CommCtlrTypes   = "3172 3174 3705 3725 3745"
604
```

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          545              CommDeviceTypes = "3270 3286"
          546              CTCATypes       = "SCTC CTCA CTC 3088"
          547
          548              Do CommType = 1 to Words(CommDeviceTypes)
 1        549                  OldCtlGroup  = '0000'
 1        550                  ThisType = Word(CommDeviceTypes,CommType)
 1        551                  TypeHeader = "* " ThisType "DEVICES"
 1        552                  "PIPE Var  TypeHeader |>>" GuestLogonID "PROTODIR" PathMapFileMode
 1        553                  Say "MJFACG901I Processing COMM records for" ThisType
 1        554                  Do Record = 1 to DevMap.0
 2        555                      Select
 2   1    556                          When word(DevMap.Record,2) = ThisType then do
 3   2    557                              NewCtlGroup  =Word(DevMap.Record,5)
 3   2    558                              Select
 3   3    559                                  When (NewCtlGroup = OldCtlGroup ) then nop;
 3   3    560                                  Otherwise do
 4   4    561                                      "PIPE Var SeparatorRecord |>>" GuestLogonID "PROTODIR" Path
            C MapFileMode
 3   4    562                                      end
 3   2    563                              End;
 3   2    564                              DeviceRecord = " SPECIAL" Right(Right(Space(Word(DevMap.Record,3),0),4),
            C 5,' ') "3270" "    " ThisType "CTLR" Word(DevMap.Record,5)
 3   2    565                              "PIPE Var DeviceRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
 3   2    566                              OldCtlGroup  = Word(DevMap.Record,5)
 2   2    567                              end;
 2   1    568                          Otherwise nop;
 2        569                      end;
 2        570                      Call PostProgressMsg
 1        571                  end;
 1        572                  TypeTrailer = "* END " ThisType "DEVICES"
 1        573                  "PIPE Var TypeTrailer |>>" GuestLogonID "PROTODIR" PathMapFileMode
          574              end;
          575
          576              Do CTCAType  = 1 to Words(CTCATypes)
 1        577                  OldCtlGroup  = '0000'
 1        578                  ThisType = Word(CTCATypes,CTCAType)
 1        579                  TypeHeader = "* " ThisType "DEVICES"
 1        580                  "PIPE Var  TypeHeader |>>" GuestLogonID "PROTODIR" PathMapFileMode
 1        581                  Say "MJFACG901I Processing CTCA records for" ThisType
 1        582                  Do Record = 1 to DevMap.0
 2        583                      Select
```


This Sheet Has Been Replaced

```
2   1   584
3   2   585
3   2   586
3   2   587
        C etConnect.Record "*" ,
3   2   588
3   2   589
        C ')
3   2   590
        C 5,' ') ,
```

```
When word(DevMap.Record,2) = ThisType then do
    NewCtlGroup  =Word(DevMap.Record,5)
    Call SeparateCtlUnits
    Parse Var DevMap.Record . "*MAPS_TO" . "*CONNECT=" TargetSys.Record Targ

    TargetSys.Record = Left(TargetSys.Record,8,' ')
    TargetConnect.Record = Right(Right(Space(TargetConnect.Record,0),4),5,'

    DeviceRecord = " SPECIAL" Right(Right(Space(Word(DevMap.Record,3),0),4),
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
 3   2   591                                  "CTCA" TargetSys.Record ,
 3   2   592                                  Right(Right(Space(Word(DevMap.Record,3),0),4),5,' ') "   *" ThisType
           ;  C  "CTLR" Word(DevMap.Record,5)
 3   2   593                                  "PIPE Var DeviceRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
 3   2   594                                  OldCtlGroup  = Word(DevMap.Record,5)
 2   2   595                                  end;
 2   1   596                              Otherwise nop;
 2       597                          end;
 2       598                          Call PostProgressMsg
 1       599                      end;
 1       600                      TypeTrailer = "* END " ThisType "DEVICES"
 1       601                      "PIPE Var TypeTrailer |>>" GuestLogonID "PROTODIR" PathMapFileMode
         602                  end;
         603
         604
         605                  Do CtlrType = 1 to Words(CommCtlrTypes)
 1       606                      OldCtlGroup  = '0000'
 1       607                      ThisType = Word(CommCtlrTypes,CtlrType)
 1       608                      TypeHeader = "* " ThisType "DEVICES"
 1       609                      "PIPE Var  TypeHeader |>>" GuestLogonID "PROTODIR" PathMapFileMode
 1       610                      Say "MJFACG901I Processing COMM records for" ThisType
 1       611                      Do Record = 1 to DevMap.0
 2       612                          Select
 2   1   613                          When word(DevMap.Record,2) = ThisType then do
 3   2   614                              Parse Var DevMap.Record . "*MAPS_TO" . . TargetAddress .
 3   2   615                              TargetAddress = Right(Right(Space(TargetAddress,0),4),5,' ')
 3   2   616                              NewCtlGroup  =Word(DevMap.Record,5)
 3   2   617                              Call SeparateCtlUnits
 3   2   618                              CtlrRecord   = " DEDICATE" Right(Right(Space(Word(DevMap.Record,3),0),4)
           ;  C ,5,' ') TargetAddress
 3   2   619                              "PIPE Var CtlrRecord   |>>" GuestLogonID "PROTODIR" PathMapFileMode
 3   2   620                              OldCtlGroup  = Word(DevMap.Record,5)
 2   2   621                              end;
 2   1   622                          Otherwise nop;
 2       623                      end;
 2       624                      Call PostProgressMsg
 1       625                  end;
 1       626                  TypeTrailer = "* END " ThisType "DEVICES"
 1       627                  "PIPE Var TypeTrailer |>>" GuestLogonID "PROTODIR" PathMapFileMode
         628              end;
         629
```


This Sheet Has Been Replaced

```
630
631   BuildCommDevsEnd: Return;
632
633   BuildResidueList: Nop;
634               "PIPE DISK" PathMapFileName "PATHMAPS" PathMapFileMode ,
635                   "| NLOCATE /3310/" ,
636                   "| NLOCATE /3330/" ,
637                   "| NLOCATE /3340/" ,
638                   "| NLOCATE /3350/" ,
639                   "| NLOCATE /3370/" ,
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          640                 "| NLOCATE /3380/" ,
          641                 "| NLOCATE /3390/" ,
          642                 "| NLOCATE /3995/" ,
          643                 "| NLOCATE /9341/" ,
          644                 "| NLOCATE /9342/" ,
          645                 "| NLOCATE /9344/" ,
          646                 "| NLOCATE /9345/" ,
          647                 "| NLOCATE /9390/" ,
          648                 "| NLOCATE /9391/" ,
          649                 "| NLOCATE /9392/" ,
          650                 "| NLOCATE /9393/" ,
          651                 "| NLOCATE /9395/" ,
          652                 "| NLOCATE /5500/" ,
          653                 "| NLOCATE /3420/" ,
          654                 "| NLOCATE /3422/" ,
          655                 "| NLOCATE /3480/" ,
          656                 "| NLOCATE /3490/" ,
          657                 "| NLOCATE /3495/" ,
          658                 "| NLOCATE /3590/" ,
          659                 "| NLOCATE /AFP/" ,
          660                 "| NLOCATE /1403/" ,
          661                 "| NLOCATE /3203/" ,
          662                 "| NLOCATE /3211/" ,
          663                 "| NLOCATE /3800/" ,
          664                 "| NLOCATE /3820/" ,
          665                 "| NLOCATE /3900/" ,
          666                 "| NLOCATE /4245/" ,
          667                 "| NLOCATE /4248/" ,
          668                 "| NLOCATE /3172/" ,
          669                 "| NLOCATE /3174/" ,
          670                 "| NLOCATE /3270/" ,
          671                 "| NLOCATE /3286/" ,
          672                 "| NLOCATE /3705/" ,
          673                 "| NLOCATE /3725/" ,
          674                 "| NLOCATE /3745/" ,
          675                 "| NLOCATE /SCTC/" ,
          676                 "| SPECS 1.71" ,
          677                 "|>" GuestLogonID "RESIDUE" PathMapFileMode
          678               ResidualRecord = "* UNPROCESSED/PARTIALLY PROCESSED DEVICES"
          679               "PIPE Var ResidualRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
          680               "PIPE DISK" GuestLogonID "RESIDUE" PathMapFileMode ,
```


This Sheet Has Been Replaced

```
681                    "|>>" GuestLogonID "PROTODIR  PathMapFileMode
682                  "PIPE DISK" GuestLogonID "CHNGHIST" PathMapFileMode ,
683                    "|>>" GuestLogonID "PROTODIR" PathMapFileMode
684                  "PIPE DISK" PathMapFileName "PATHMAPS" PathMapFileMode ,
685                    "| COUNT LINES" ,
686                    "| VAR InRecords"
687                  "PIPE DISK" GuestLogonID    "RESIDUE" PathMapFileMode ,
688                    "| COUNT LINES" ,
689                    "| VAR Residue"
690  BuildResidueListEnd: Return;
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                /
          691
          692   ProcessProfile: Nop;
          693                 "PIPE DISK SCRAMIEE  PROFILE *" ,
          694                       "| STEM ConfigRecord."
          695                 Do Record = 1 to ConfigRecord.0
    1     696                     INTERPRET ConfigRecord.Record
          697                 End;
          698   ProcessProfileEnd: Return;
          699
          700   SeparateCtlUnits: Nop;
          701                 SeparatorRecord = "* SOURCE " ThisType  "CTLR" Word(DevMap.Record,5)
          702                 Select
      1   703                       When (NewCtlGroup = OldCtlGroup) then  nop;
      1   704                       Otherwise do
    1 2   705                             "PIPE Var SeparatorRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
      2   706                             end
          707                 End;
          708   SeparateCtlUnitsEnd: Return;
          709
          710   PostProgressMsg: Nop;
          711                 Select
      1   712                       When (Record // 5000) = 0  then do
    1 2   713                         Say "MJFACG900I" Right(Record,6,' ') "Device records processed."
      2   714                         end;
      1   715                       Otherwise nop;
          716                 end;
          717   PostProgressMsgEnd: Return;
          718
          719   BuildDedicatedDASD: Nop
          720               Address CMS;
          721               "PIPE DISK" PathMapFileName "PATHMAPS" PathMapFileMode ,
          722                   "| Stem DevMap."
          723               Do DASDType = 1 to Words(DASDTypes)
    1     724                   OldCtlGroup  = '0000'
    1     725                   ThisType = Word(DASDTypes,DASDType)
    1     726                   TypeHeader = "* " ThisType "DEVICES"
    1     727                   "PIPE Var  TypeHeader |>>" GuestLogonID "PROTODIR" PathMapFileMode
    1     728                   Say "MJFACG901I Processing DASD records for" ThisType
    1     729                   Do Record = 1 to DevMap.0
    2     730                       Select
    2 1   731                           When word(DevMap.Record,2) = ThisType then do
```


This Sheet Has Been Replaced

```
3   2   732                         NewCtlGroup  =Word(DevMap.Record,5)
3   2   733                         Parse Var DevMap.Record . "*MAPS_TO" . . TargetAddress .
3   2   734                         TargetAddress = Right(Right(Space(TargetAddress,0),4),5,' ')
3   2   735                         Call SeparateCtlUnits
3   2   736                         DeviceRecord = " DEDICATE" Right(Right(Space(Word(DevMap.Record,3),0),4)
        C ,5,' ') TargetAddress "*" ThisType "CTLR" Word(DevMap.Record,5)
3   2   737                         "PIPE Var DeviceRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
3   2   738                         OldCtlGroup  = Word(DevMap.Record,5)
2   2   739                         end;
2   1   740                   Otherwise nop;
```

This Sheet Has Been Replaced

```
=> Source Listing                                MJFACG   EXEC     G1
| Compiler REXX/370 3.0  LVL PQ28073    Time: 15:48:24                Date: 2000-02-03                Page:   17
: Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
   2        741                    end;
   2        742                    Select
   2   1    743                          When (Record // 5000) = 0  then do
   3   2    744                            Say "MJFACG900I" Right(Record,6,' ') "Device records processed."
   2   2    745                            end;
   2   1    746                          Otherwise nop;
   2        747                    end;
   1        748                end;
   1        749                TypeTrailer = "* END " ThisType "DEVICES"
   1        750                "PIPE Var TypeTrailer |>>" GuestLogonID "PROTODIR" PathMapFileMode
            751             end;
            752  BuildDedicatedDASDEnd: Return;
            753
            754  BuildSharedDASD: Nop
            755             Address CMS;
            756             "PIPE DISK" PathMapFileName "PATHMAPS" PathMapFileMode ,
            757                 "| Stem DevMap."
            758             Do DASDType = 1 to Words(DASDTypes)
   1        759                 OldCtlGroup  = '0000'
   1        760                 ThisType = Word(DASDTypes,DASDType)
   1        761                 TypeHeader = "* " ThisType "DEVICES"
   1        762                 "PIPE Var  TypeHeader |>>" GuestLogonID "PROTODIR" PathMapFileMode
   1        763                 Say "MJFACG901I Processing DASD records for" ThisType
   1        764                 Do Record = 1 to DevMap.0
   2        765                     Select
   2   1    766                         When word(DevMap.Record,2) = ThisType then do
   3   2    767                             NewCtlGroup  =Word(DevMap.Record,5)
   3   2    768                             Parse Var DevMap.Record . "*MAPS_TO" "*DEFINE" TargetType TargetAddress
          C . "*LABEL=" TargetLabel "*" .
   3   2    769                             TargetAddress = Right(Right(Space(TargetAddress,0),4),5,' ')
   3   2    770                             Call SeparateCtlUnits
   3   2    771                             DeviceRecord = " MDISK" Right(Right(Space(Word(DevMap.Record,3),0),4),5,
          C ' ') TargetType "DEVNO" TargetAddress "MR ALL"
   3   2    772                             "PIPE Var DeviceRecord |>>" GuestLogonID "PROTODIR" PathMapFileMode
   3   2    773                             OldCtlGroup  = Word(DevMap.Record,5)
   2   2    774                             end;
   2   1    775                         Otherwise nop;
   2        776                     end;
   2        777                     Select
   2   1    778                           When (Record // 5000) = 0  then do
   3   2    779                             Say "MJFACG900I" Right(Record,6,' ') "Device records processed."
```

269

This Sheet Has Been Replaced

```
2  2  780                                end;
2  1  781                               Otherwise nop;
2     782                          end;
1     783                     end;
1     784                     TypeTrailer = "* END " ThisType "DEVICES"
1     785                     "PIPE Var TypeTrailer |>>" GuestLogonID "PROTODIR" PathMapFileMode
      786                end;
      787  BuildSharedDASDEnd: Return;
      788
      789  CheckDASDBuild: Nop;
```

This Sheet Has Been Replaced

```
> Source Listing                                    MJFACG   EXEC     G1
 Compiler REXX/370 3.0  LVL PQ28073    Time: 15:48:24                    Date: 2000-02-03                    Page:    18
  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          :
          790              Select
       1  791                   When DedicatedMDISKS = 'S' then nop
       1  792                   When DedicatedMDISKS = 'D' then nop
       1  793                   Otherwise do
    1  2  794                      CursorPos = "D23"
    1  2  795                      D1 = "MJFACG022E Invalid selection. Must be 'D' or 'S'."
    1  2  796                      Signal RedisplayPanel
       2  797                   end;
          798              End;
          799   CheckDASDBuildEnd: Return;
```


This Sheet Has Been Replaced

: Lines 799

: of compiled program in bytes 62552

il messages Informational Warning Error Severe Terminating
5 1 4 0 0 0

following lines have been flagged

151 307 364 400

or No. Line

71 364 400

550 17 307

773 151

lshing time of compilation: 15:48:25

This Sheet Has Been Replaced

APPENDIX G

This Sheet Has Been Replaced



Compilation Summary

GETIOCP EXEC G2

pilation successful

piler Options

```
NOALTERNATE
  CEXEC      (GETIOCP  CEXEC    G2)                    RECFM=F,LRECL=1024
NOCOMPILE    (S)
NOCONDENSE
NODLINK
NODUMP
  FLAG       (I)
NOIEXEC
  LIBLEVEL   (*)
  LINECOUNT  (55)
  MARGINS    (1 *)
NOOBJECT
  OPTIMIZE
  PRINT      (GETIOCP  LISTING  G2)                    RECFM=V,LRECL=121
NOSAA
NOSLINE
  SOURCE
  SYSIN      (GETIOCP  EXEC     G2 R § R U R § R \  (  PD) RECFM=V,LRECL=71
NOTERMINAL
NOTESTHALT
NOTRACE
NOXREF
Minimum Library Level required: 0
```

274

This Sheet Has Been Replaced

```
Do Sel Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                1  /*********************************************************************/
                2  /* Retrieve WSC master IOCP from WSCMVS                               */
                3  /* MVSPGE = 503F,  OSSPGE = 528C                                     */
                4  /*********************************************************************/
                5  Address Command
                6  Parse Upper Arg member fm .
                7  'CP LINK MVSDISKS 528C 528C RR'
                8  'ACCESS 528C M'
                9  If Length(member) = 0 Then Do
  1   1        10    Say 'Enter one of the following member names as a parameter:'
  1   1        11    Say '- IZP9021C for the 9021/982 C system'
  1   1        12    Say '- IZP9021P for the 9021/9X2 P system'
  1   1        13    Say '- IZP9021V for the 9021/982 V system'
  1   1        14    Say '- IZP9021Z for the 9021/982 Z system'
  1   1        15    Say '- IZP9672W for the 9672/RX3 W system'
  1   1        16    Say '- IZP9672L for the 9672/RX3 L system'
      1        17    Say '- CF1LPAR  for the 9674/C03 9674SG41'
      1        18    Say '- CF2LPAR  for the 9674/C03 9674SG42'
      1        19    Say ' '
      1        20    Say 'Example: GETIOCP IZP9021C'
  1   1        21    Say ' '
  1   1        22    Say 'Or choose from the following MEMBERS in SYS1.IOCP.MASTER:'
  1   1        23    'LISTDS SYS1.IOCP.MASTER M (PDS'
  1   1        24    Say 'Enter: "CP TERM CHARDEL OFF" if member has an "@" in it.'
  1   1        25    Say ' '
  1            26  End
               27  Else Do
  1   1        28    'FILEDEF IN M1 DSN SYS1.IOCP.MASTER (MEMBER' member
  1   1        29    If fm = '' Then
  2   1        30     'FILEDEF OUT DISK' member 'IOCP A'
  1   1        31    Else 'FILEDEF OUT DISK' member 'IOCP' fm
  1   1        32    'MOVEFILE IN OUT'
  1   1        33    If rc ¬= 0 Then
  2   1        34      Say 'Please enter a valid parameter or enter no parameter for help'
  1            35  End
               36  'RELEASE M (DETACH'
               37  Exit
```


This Sheet Has Been Replaced

Compilation Statistics

GETJOCP EXEC G2

XX Lines 37

ze of compiled program in bytes 5480

tal messages	Informational	Warning	Error	Severe	Terminating
0	0	0	0	0	0

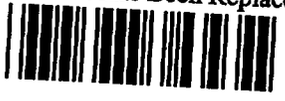
oishing time of compilation: 15:50:57

276

This Sheet Has Been Replaced

APPENDIX H

This Sheet Has Been Replaced



⇒ Compilation Summary MJFJFR EXEC G1
| Compiler REXX/370 3.0 LVL PQ28073 Time: 15:50:54 Date: 2000-02-03 Page: 1

essage(s) reported. Highest severity code was 4 - Warning

piler Options

```
  NOALTERNATE
    CEXEC      (MJFJFR   CEXEC    G1)                    RECFM=F,LRECL=1024
  NOCOMPILE    (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG       (I)
  NOIEXEC
    LIBLEVEL   (*)
    LINECOUNT  (55)
    MARGINS    (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT      (MJFJFR   LISTING  G1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN      (MJFJFR   EXEC     G1 R § R U R § R \  [   PD) RECFM=V,LRECL=85
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 3
```

278



```
=> Source Listing                                    MJFJFR  EXEC  A1
I Compiler REXX/370 3.0  LVL PQ28073    Time: 15:50:54          Date: 2000-02-03          Page:    2
: Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                   1  /*                                                                  */
                   2  /*   MJFJFR EXEC (c) 1999 N.J. Firley and IBM Corporation           */
                   3  /*                                                                  */
                   4  /*   A REXX procedure to control mapping a BRS Requirements         */
                   5  /*         definition worksheet (RDWS) to a DEVPATHS file for       */
                   6  /*         use with the DIRBUILD tool                               */
                   7  /*                                                                  */
                   8            Address CMS;
                   9            Trace Value Off;
                                      |
-FANGA0055OW Unsupported TRACE options will default to OFF
                  10            Parse Arg InFileName InFileType InFileMode .
                  11            InFileID = InFileName InFileType InFileMode
                  12            OutFileID = InFileName "DEVPATHS" InFileMode
                  13            RMSFileID = InFileName "RMSCNTRL" InFileMode
                  14            ControlFile2ID = InFileName "PROCNTRL" InFileMode
                  15
                  16            "PIPE DISK" InFileID ,
                  17                 "| LOCATE /*DEFINE/ " ,
                  18                 "| STEM InputRecord."
                  19            Rcs = rc
                  20            Select
        1         21               When rcs = 0 then nop;
        1         22               Otherwise do
    1   2         23                    Say "MJFJFR002E Error" Rcs "reading" InFileID
    1   2         24                    Say "MJFJFR003I Exiting."
        2         25                    end;
                  26            end;
                  27
                  28
                  29            OutputRecord.0 = InputRecord.0
                  30
                  31            DEVPATHSTrailer =
                  32                 "*LOC=" Left(' ',2,' ') ,
                  33                 "*ALIAS=" Left(' ',10,' ') ,
                  34                 "*PLEXID=" Left(' ',8,' ') ,
                  35                 "*DEVSER=" Left(' ',12,' ') ,
                  36                 "*RESTRICT=" Left(' ',32,' ') ,
                  37                 "*HWPATHING=" Left(' ',374,' ')
                  38
                  39            Say "MJFJFR004I" Right(Space(InputRecord.0,0),5,'0') "records" ,
```

279



This Sheet Has Been Replaced

```
          40                          available for input.
          41          Do ThisRecord = 1 to InputRecord.0
1         42              OutputRecord.ThisRecord = InputRecord.ThisRecord ,
1         43                    DEVPATHSTrailer
1         44              Select
1    1    45                    when ThisRecord // 500 = 0 then do
2    2    46                        Say "HJFJFR005I" Right(Space(ThisRecord,0),5,'0') "records" ,
2    2    47                                         "accepted for input."
1    2    48                        end;
1    1    49                    otherwise nop
```


This Sheet Has Been Replaced



```
f  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
   1        50              End;
            51           End;
            52
            53
            54           "PIPE Stem OutputRecord."  ,
            55                "|>" OutFileID
            56           Rcs = rc
            57
            58           "PIPE DISK" InFileID ,
            59                "| NLOCATE /*DEFINE/" ,
            60                "|>>" ControlFile2ID
            61
            62           Select
        1   63               When Rcs = 0 then do
   1    2   64                   Say "MJFJFR006I" Right(Space(ThisRecord-1,0),5,'0') "records" ,
   1    2   65                                    "processed."
        2   66                   end;
        1   67               Otherwise nop;
            68           End;
            69           Say "MJFJFR001I Exiting. Rc=" Rc
```


This Sheet Has Been Replaced

XX Lines 69

ze of compiled program in bytes 7168

tal messages	Informational	Warning	Error	Severe	Terminating
1	0	1	0	0	0

e following lines have been flagged

ror No.	Line
550	9

nishing time of compilation: 15:50:54

This Sheet Has Been Replaced

APPENDIX I

This Sheet Has Been Replaced



message(s) reported. Highest severity code was 4 - Warning

mpiler Options

```
   NOALTERNATE
     CEXEC      (MJFMCM    CEXEC     G1)                    RECFM=F,LRECL=1024
   NOCOMPILE    (S)
   NOCONDENSE
   NODLINK
   NODUMP
     FLAG       (I)
   NOIEXEC
     LIBLEVEL   (*)
     LINECOUNT  (55)
     MARGINS    (1 *)
   NOOBJECT
     OPTIMIZE-
     PRINT      (MJFMCM    LISTING   G1)                    RECFM=V,LRECL=121
   NOSAA
   NOSLINE
     SOURCE
     SYSIN      (MJFMCM    EXEC      G1 R²§ R²U R²§ R²\ \    \ D) RECFM=V,LRECL=164
   NOTERMINAL
   NOTESTHALT
   NOTRACE
   NOXREF
   Minimum Library Level required: 4
```

284

This Sheet Has Been Replaced

```
    1  /*                                                                  */
    2            Address CMS;
    3            Trace Value Off;
                              |
+FANGA0550W Unsupported TRACE options will default to OFF
    4            Parse Upper Arg Argstring
    5  /* ---------------------------------------------------------------- */
    6  /*                                                                  */
    7  /* ---------------------------------------------------------------- */
    8  Literals: Nop;
    9            WorkFmode = "W"
   10            Rcx = 0
   11  LiteralsEnd: Nop;
   12  /* ---------------------------------------------------------------- */
   13  /*                                                                  */
   14  /* ---------------------------------------------------------------- */
   15  MainPgm:  Nop;
   16            Call ParseInput
   17            Call InitializeWorkFiles
   18            Call CheckInputFiles
   19            Call GetInputFileStructure
   20            Call ProcessProfile
   21            Call MergeControlInformationOnSource
   22            Call MergeControlInformationOnTarget
   23            Call LoadSourceAndTargetFiles
   24            Call MatchGeneralAllocations
   25  MainPgmEnd: Call ExitRtn Rcx
   26  /* ---------------------------------------------------------------- */
   27  /*                                                                  */
   28  /* ---------------------------------------------------------------- */
   29  ExitRtn:  Parse Arg Rcx
   30            Exit Rcx
   31  ExitRtnEnd: Return;
   32  /* ---------------------------------------------------------------- */
   33  /*                                                                  */
   34  /* ---------------------------------------------------------------- */
   35  ParseInput: Nop;
   36            Parse Var Argstring SourceFName SourceFtype SourceFmode ,
   37                                TargetFname TargetFtype TargetFmode ,
   38                                "(" OptionString
   39            Select
```

This Sheet Has Been Replaced



```
1   2    41          Loc_WorkDiskParm = WordPos('WORKDISK',OptionString) +1
1   2    42          WorkFmode = Word(OptionString,Loc_WorkDiskParm)
2   2    43          WorkFmode = Strip(WorkFmode,'B',' ')
    2    44          end;
    1    45       Otherwise nop;
         46    End;
         47
         48    Select
    1    49       When SourceFtype <> 'DEVPATHS' then do

    1    40          when WordPos('WORKDISK',optionString)>0 then do
```


This Sheet Has Been Replaced

Source Listing MJFMCM EXEC G4

```
Compiler REXX/370 3.0  LVL PQ28073    Time: 15:45:45                Date: 2000-02-03                Page:    3
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
  1   2    50                  Say Date(S) Time() "MJFMCM024E Invalid input file." ,
  1   2    51                      "Source must be type DEVPATHS."
  1   2    52                  Call ExitRtn 24
      2    53                  end;
      1    54              When TargetFtype <> 'DEVPATHS' then do
  1   2    55                  Say Date(S) Time() "MJFMCM024E Invalid input file." ,
  1   2    56                      "Target must be type DEVPATHS."
  1   2    57                  Call ExitRtn 24
      2    58                  end;
      1    59              Otherwise nop;
           60          End;
           61  ParseInputEnd: Return;
           62  /* ---------------------------------------------------------------- */
           63  /*                                                                  */
           64  /* ---------------------------------------------------------------- */
           65  InitializeWorkFiles: Nop;
           66          "PIPE CMS IDENTIFY | STEM WhoAmI."
           67          NodeName   =  Space(Word(WhoAmI.1,3),0)
           68
           69          SourceFileID    = SourceFname SourceFtype SourceFmode
           70          TargetFileID    = TargetFname TargetFtype TargetFmode
           71          SourceCtrlID    = SourceFname "PROCNTRL *"
           72          TargetCtrlID    = TargetFname "PROCNTRL *"
           73          SurplusFileID   = SourceFname "SURPLUS"  WorkFmode
           74          ShortageFileID  = SourceFname "SHORTAGE" WorkFmode
           75          JournalFileID   = SourceFname "JOURNAL"  WorkFmode
           76          ErrorFileID     = SourceFname "ERRORLOG" WorkFmode
           77          MergedMapFileID = SourceFname "PATHMAPS" WorkFmode
           78
           79          AuditStamp.1= Date(S) Time() "* MJFMCM Initialized."
           80          AuditStamp.2= Date(S) Time() "* Run by:" Userid() "AT" Nodename
           81          AuditStamp.3= Date(S) Time() "* Source:" SourceFileID
           82          AuditStamp.4= Date(S) Time() "* Target:" TargetFileID
           83          AuditStamp.5= Date(S) Time() "* SCntrl:" SourceCtrlID
           84          AuditStamp.6= Date(S) Time() "* TCntrl:" TargetCtrlID
           85          AuditStamp.0= 6
           86          "PIPE STEM AuditStamp." ,
           87                  "| >>" JournalFileID  ;
           88          "PIPE LITERAL" Date(S) Time() Userid() ,
           89                  "| >>" ErrorFileID
           90          "PIPE LITERAL *" Date(S) Time() Userid() ,
```

287

This Sheet Has Been Replaced

```
92              "PIPE LITERAL *" Date(S) Time() Userid() ,
93                      "|  >" ShortageFileID
94              "PIPE LITERAL *" Date(S) Time() Userid() ,
95                      "|  >" MergedMapFileID
96      InitializeWorkFilesEnd: Return;
97      /* ------------------------------------------------------------------ */
98      /*                                                                    */
99      /* ------------------------------------------------------------------ */
100     CheckInputFiles: Nop;
```

91 "| >" SurplusFile ID

This Sheet Has Been Replaced

```
f  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                  /*
                 101            "PIPE CMS LISTFILE" SourceFileID "(LABEL" ,
                 102                 "| STEM SourceFileIDData."
                 103
                 104            LengthOfDevpathRecord = Word(SourceFileIDData.2,5)
                 105            Select
           1     106                When DataType(LengthOfDevpathRecord,'N') = 1 & ,
           1     107                     LengthOfDevPathRecord > 0 then nop;
           1     108                Otherwise do
     1     2     109                     ErrorMsg = Date(S) time() "MJFMCM028E Invalid" ,
     1     2     110                                "or missing source file. Exiting."
     1     2     111                     Say ErrorMsg
     1     2     112                     "PIPE LITERAL" ErrorMsg ,
     1     2     113                          "|>>" ErrorFileID
     1     2     114                     Call ExitRtn 28
           2     115                     end;
                 116             End;
                 117
                 118            "PIPE CMS LISTFILE" TargetFileID "(LABEL" ,
                 119                 "| STEM TargetFileIDData."
                 120            LengthOfDevpathRecord2 = Word(SourceFileIDData.2,5)
                 121            Select
           1     122                When DataType(LengthOfDevpathRecord2,'N') = 1 & ,
           1     123                     LengthOfDevPathRecord2 > 0 then nop;
           1     124                Otherwise do
     1     2     125                     ErrorMsg = Date(S) time() "MJFMCM028E Invalid" ,
     1     2     126                                "or missing target file. Exiting."
     1     2     127                     Say ErrorMsg
     1     2     128                     "PIPE LITERAL" ErrorMsg ,
     1     2     129                          "|>>" ErrorFileID
     1     2     130                     Call ExitRtn 28
           2     131                     end;
                 132             End;
                 133
                 134
                 135
                 136            "PIPE DISK" SourceFileID ,
                 137                    "| NLOCATE /*DEFINE/" ,    /* Only define statements */
                 138                    "| NLOCATE /* /" ,         /* and comments are allowd*/
                 139                    "| STEM SourceResidue."
                 140
                 141            "PIPE DISK" TargetFileID ,
```


This Sheet Has Been Replaced

```
                                "| NLOCATE /*DEFINE/  ,  /* only define statements */
                                "| NLOCATE /* /" ,       /* and comments are allowd*/
                                "| STEM TargetResidue."

                        Select
                            When SourceResidue.0 >  0 then do;
                                ErrorMsg = Date(S) time() "MJFMCH006E Invalid" ,
                                           "source definitions found. Exiting."
                                Say ErrorMsg
                                "PIPE LITERAL" ErrorMsg ,
```


This Sheet Has Been Replaced

```
f  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
    1   2   152                       "|>>" ErrorFileID
    1   2   153                   Call ExitRtn 8
        2   154                   end;
        1   155               Otherwise nop;
            156           End;
            157           Select
        1   158               When TargetResidue.0 >  0 then do;
    1   2   159                   ErrorMsg = Date(S) time() "MJFMCM009E Invalid" ,
    1   2   160                              "target definitions found. Exiting."
    1   2   161                   Say ErrorMsg
    1   2   162                   "PIPE LITERAL" ErrorMsg ,
    1   2   163                       "|>>" ErrorFileID
    1   2   164                   Call ExitRtn 9
        2   165                   end;
        1   166               Otherwise nop;
            167           End;
            168   CheckInputFilesEnd: Return;
            169   /* ---------------------------------------------------------------- */
            170   /*                                                                  */
            171   /* ---------------------------------------------------------------- */
            172   GetInputFileStructure: Nop;
            173             "PIPE DISK MJFPDV COMMON *" ,
            174                 "| STEM DevPathRecordLayout."
            175             "PIPE DISK" SourceFileID ,
            176                 "| LOCATE /*DEFINE/ " ,
            177                 "| TAKE 1" ,
            178                 "| VAR DEVPATHSRecord"
            179              Do Statement = 1 to DevPathRecordLayout.0
    1       180                 Interpret DevPathRecordLayout.Statement
            181              End;
            182   GetInputFileStructureEnd: Return;
            183   /* ---------------------------------------------------------------- */
            184   /*                                                                  */
            185   /* ---------------------------------------------------------------- */
            186   ProcessProfile: Nop;
            187               "PIPE DISK SCRAMIEE  PROFILE *" ,
            188                   "| STEM ConfigRecord."
            189               Do Record = 1 to ConfigRecord.0
    1       190                   INTERPRET ConfigRecord.Record
            191               End;
            192               "PIPE LITERAL" SelectedTypes ,
```


This Sheet Has Been Replaced

```
193                     "| SPLIT AFTER BLANK  ,
194                     "| SORT UNIQUE" ,
195                     "| JOIN *" ,
196                     "| VAR SelectedTypes"
197     ProcessProfileEnd: Return;
198     /* ------------------------------------------------------------------ */
199     /*                                                                    */
200     /* ------------------------------------------------------------------ */
201     MergeControlInformationOnSource: Nop;
202                 "PIPE DISK" SourceCtrlID ,
```


This Sheet Has Been Replaced

f Do Sel Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0

Source Listing MJFMCH EXEC G1

```
                203                "| LOCATE /*ENVIRONMENT/ " ,
                204                "| STEM RecoveryEnvironment."
                205             Select
            1   206                 When RecoveryEnvironment.0 > 0 then nop
            1   207                 Otherwise do
        1   2   208                       JournalMsg.1 = Date(S) Time() Userid() ,
        1   2   209                            "MJFMCM104I Invalid Recovery Environment" ,
        1   2   210                                 "keyword specified."
        1   2   211                       JournalMsg.2 = Date(S) Time() Userid() ,
        1   2   212                            "MJFMCM105I Must be one of: <V|L|N>"
        1   2   213                       JournalMsg.0 = 2
        1   2   214                       Do Msg = 1 to 2
        2   2   215                          Say JournalMsg.Msg
        1   2   216                       End;
        1   2   217                       "PIPE STEM JournalMsg." ,
        1   2   218                            "|>>" JournalFileID
        1   2   219                       "PIPE STEM JournalMsg." ,
        1   2   220                            "|>>" ErrorFileID
        1   2   221                       Call ExitRtn 104
            2   222                       end;
                223             End;
                224             RecoveryManager = Space(Word(RecoveryEnvironment.1,2),0)
                225             Select
            1   226                 When RecoveryManager = "V" then do
        1   2   227                       JournalMsg.1 = Date(S) Time() Userid() ,
        1   2   228                            "MJFMCH100I Selecting VM as recovery" ,
        1   2   229                                 "manager."
        1   2   230                       JournalMsg.2 = Date(S) Time() Userid() ,
        1   2   231                            "MJFMCH101I Bypassing forced addressing" ,
        1   2   232                            "on source."
        1   2   233                       JournalMsg.0 = 2
        1   2   234                       Do Msg = 1 to 2
        2   2   235                          Say JournalMsg.Msg
        1   2   236                       End;
        1   2   237                       "PIPE STEM JournalMsg." ,
        1   2   238                            "|>>" JournalFileID
            2   239                       end;
            1   240                 Otherwise do
        1   2   241                       JournalMsg.1 = Date(S) Time() Userid() ,
        1   2   242                            "MJFMCH100E Selecting NATIVE/LPAR" ,
        1   2   243                                 "recovery type."
```


This Sheet Has Been Replaced

```
1   2   244                 JournalMsg.2 = Date(S) Time() Userid() ,
1   2   245                      "MJFMCM102E Function Not Supported. Exiting."
1   2   246                 JournalMsg.0 = 2
1   2   247                 Do Msg = 1 to 2
2   2   248                    Say JournalMsg.Msg
1   2   249                 End;
1   2   250                 "PIPE STEM JournalMsg." ,
1   2   251                      "|>>" JournalFileID
    2   252                 end;
        253             End;
```

This Sheet Has Been Replaced

Do Sel Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0

Source Listing MJFMCH EXEC

```
254   MergeControlInformationOnSourceEnd: Return;
255   /* ---------------------------------------------------------------- */
256   /*                                                                  */
257   /* ---------------------------------------------------------------- */
258   MergeControlInformationOnTarget: Nop;
259                 "PIPE DISK" TargetCtrlID ,
260                     "| LOCATE /*FIXEDADD/ " ,
261                     "| STEM FixedAllocations."
262                 Select
1   1 263               When FixedAllocations.0 > 0 then Call RestrictTarget
1   1 264               Otherwise do
1 2 2 265                     JournalMsg.1 = Date(S) Time() Userid() ,
1 2 2 266                         "MJFMCM200W No restricted addresses found"
1 2 2 267                             "on recovery target."
1 2 2 268                     JournalMsg.2 = Date(S) Time() Userid() ,
1 2 2 269                         "MJFMCM201R Confirm no devices required? <Y|N>"
1 2 2 270                     JournalMsg.0 = 2
1 2 2 271                     "PIPE STEM JournalMsg." ,
1 2 2 272                         "|>>" JournalFileID
1 2 2 273                     Say JournalMsg.1
1 2 2 274                     Do until Response = Y | Response = N
2 2 2 275                        Say JournalMsg.2
2 2 2 276                        Parse Upper Pull OK .
2 2 2 277                        Response = Left(Space(OK,0),1)
1 2 2 278                     end;
1 2 2 279                     "PIPE LITERAL" Date(S) Time() Userid() Response ,
1 2 2 280                         "|>>" JournalFileID
  2 2 281                     end;
  2 2 282                     Select
  3 3 283                         When Response = "Y" then nop;
  3 3 284                         Otherwise do
1 4 4 285                           JournalMsg.1 = Date(S) Time() Userid() ,
1 4 4 286                               "MJFMCM202I Exiting at user request."
1 4 4 287                         "PIPE LITERAL" JournalMsg.1 ,
1 4 4 288                             "|>>" JournalFileID
1 4 4 289                           Say JournalMsg.1
  4 4 290                           end;
  2 2 291                     End;
    2 292               End;
    2 293 MergeControlInformationOnTargetEnd: Return;
    2 294 /* ---------------------------------------------------------------- */
```


This Sheet Has Been Replaced

```
295  /*                                                                  */
296  /* ---------------------------------------------------------------- */
297  RestrictTarget: Nop;
298            "PIPE DISK" TargetCtrlID ,
299                 "| LOCATE /*FIXEDADD/ " ,
300                 "| STEM" TargetFixedAddresses.
301            Do ThisOne = 1 to TargetFixedAddresses.0
1   302             Parse Var TargetFixedAddresses.ThisOne ,
1   303                       . TargetFixedAddress.ThisOne ,
1   304                       TargetFixedAddressDevtype.ThisOne .
```

296

```
> Source Listing                                MJFMCM   EXEC     G1
 Compiler REXX/370 3.0  LVL PQ28073    Time: 15:45:45                Date: 2000-02-09                Page:    8
  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                 ?
   1       305                  TargetFixedAddress.ThisOne = ,
   1       306                         Right(Space(TargetFixedAddress.ThisOne,0),Len_D_DPDEFADDR-1,'0')
   1       307                  TargetFixedAddressDevType.ThisOne = ,
   1       308                         Left(Space(TargetFixedAddressDevType.ThisOne,0),Len_D_DPDEFUNIT+1,' ')
   1       309                  LocatorKey.ThisOne = "*DEFINE " ,
   1       310                                       TargetFixedAddressDevtype.Thisone ,
   1       311                                       TargetFixedAddress.ThisOne
   1       312
   1       313                  "PIPE DISK" TargetFileID ,
   1       314                       "| STEM TargetBaseRecords."
   1       315
   1       316                  "PIPE STEM" TargetBaseRecords. ,
   1       317                       "| LOCATE /"||LocatorKey.ThisOne||"/" ,
   1       318                       "| VAR" ForceTargetSafety.ThisOne
   1       319
   1       320                  Select
   1   1   321                       When Left(ForceTargetSafety.ThisOne,5) <> "FORCE" then nop
   1   1   322                       Otherwise do
   2   2   323                           JournalMsg.1 = Date(S) Time() Userid() ,
   2   2   324                              "MJFMCM211E Requested fixed assignment of:"
   2   2   325                           JournalMsg.2 = Date(S) Time() Userid() ,
   2   2   326                                 "           " TargetFixedAddressDevtype.ThisOne ,
   2   2   327                                               TargetFixedAddress.ThisOne
   2   2   328                           JournalMsg.3 = Date(S) Time() Userid() ,
   2   2   329                              "MJFMCM212I No conformable device found." ,
   2   2   330                              "Exiting."
   2   2   331                           JournalMsg.0 = 3
   2   2   332                           "PIPE STEM" JournalMsg. ,
   2   2   333                               "|>>" JournalFileID
   2   2   334                           "PIPE STEM" JournalMsg. ,
   2   2   335                               "|>>" ErrorFileID
   2   2   336                           Do Msg = 1 to JournalMsg.0
   3   2   337                              Say JournalMsg.Msg
   2   2   338                           End;
   2   2   339                           Call ExitRtn 211
   1   2   340                           end;
   1       341                  End;
   1       342                  "PIPE STEM" TargetBaseRecords. ,
   1       343                       "| LOCATE /"||LocatorKey.ThisOne||"/" ,
   1       344                       "| CHANGE /*RESTRICT=" Left(' ',32,' ') "/*RESTRICT= R"||TargetFixedAddressDevT
      C ype.ThisOne||TargetFixedAddress.ThisOne||Left(' ',17,' ')||"/" ,
```


This Sheet Has Been Replaced

```
1       345             "| VAR TargetDeltaRecords..nisOne"
1       346
1       347         "PIPE STEM" TargetBaseRecords. ,
1       348             "| NLOCATE /"||LocatorKey.ThisOne||"/" ,
1       349             "| STEM TargetBaseRecords."
1       350
1       351         JournalMsg.1 = Date(S) Time() Userid() ,
1       352             "HJFMCM210I Forcing target allocation for:" ,
1       353                 TargetFixedAddressDevtype.ThisOne ,
1       354                 TargetFixedAddress.ThisOne
```




Source Listing


```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
   1      355
   1      356                  "PIPE LITERAL" JournalMsg.1 ,
   1      357                     "|>>" JournalFileID
   1      358                  Say JournalMsg.1
          359               End;
          360
          361               TargetDeltaRecords.0 = ThisOne - 1
          362
          363               Do ThisOne = 1 to TargetFixedAddresses.0
   1      364                     NextOne = TargetBaseRecords.0 + ThisOne
   1      365                     TargetBaseRecords.NextOne = TargetDeltaRecords.ThisOne
   1      366                     TargetBaseRecords.0 = NextOne
          367               End;
          368
          369               "PIPE STEM TargetBaseRecords."
          370               "| HEXSORT " Pos_D_DPDEFADDR||"."||Len_D_DPDEFADDR "ASCENDING" ,
          371               "| > " TargetFileID
          372
          373    RestrictTargetEnd: Return;
          374    /* ---------------------------------------------------------------- */
          375    /*                                                                  */
          376    /* ---------------------------------------------------------------- */
          377    ProcessRestrictedSourceAssignments: Nop;
          378    /* This routine to be filled in when Native/LPAR support available  */
          379    ProcessRestrictedSourceAssignementEnd: Return;
          380    /* ---------------------------------------------------------------- */
          381    /*                                                                  */
          382    /* ---------------------------------------------------------------- */
          383    MatchGeneralAllocations: Nop;
          384              Do Type = 1 to words(SelectedTypes)
   1      385                 ThisDeviceType = Space(Word(SelectedTypes,Type),0)
   1      386                 Call ExtractThisDeviceType ThisDeviceType
   1      387                 LimitValue = Min(NumberOfSources,NumberOfTargets)
   1      388                 InfoMsg = "MJFMCM040I Processing " Left(ThisDeviceType,8,' ') ,
   1      389                                   Right(SourceOfType.0,5,' ') "definitions found."
   1      390                 Say InfoMsg
   1      391                 Select
   1   1  392                    When ( NumberOfSources < 1 & NumberOfTargets < 1 ) then do
   2   2  393                       Iterate Type
   1   2  394                       end;
   1   1  395                    When ( NumberOfSources < 1 & NumberOfTargets > 1)  then do
```


This Sheet Has Been Replaced

```
2   2   396
2   2   397
2   2   398
1   2   399
1   1   400
2   2   401
2   2   402
2   2   403
1   2   404
1   1   405
```

```
        Call MapResourceMismatches
        Call PurgeTargetType
        Iterate Type
        end;
When ( NumberOfSources > 0 & NumberOfTargets < 1)  then do
        Call MapResourceMismatches
        Call PurgeSourceType
        Iterate Type
        end;
When ( NumberOfSources > 0 & NumberOfTargets > 0 )  then do
```

300

```
2   2   406   Do ThisUnit = 1 to LimitValue
3   2   407     Call TestModelCompatability
3   2   408     Select
3   3   409       When ModelsAreCompatible then do
4   4   410         NewRecord.ThisUnit = ,
4   4   411               Left(SourceOfType.ThisUnit,LengthOfDevpathRecord+10,' ') ,
4   4   412               "*MAPS_TO " ,
4   4   413               TargetOfType.ThisUnit
3   4   414         end;
3   3   415       Otherwise do
4   4   416         ErrorMsg = "MJFMCM012E Substitutions" ,
4   4   417                    "exhausted for:" ,
4   4   418                     SourceDevType.ThisUnit ||"-"|| SourceModelType.ThisUnit
                C ,
4   4   419                     NumberOfSources||"/"||NumberOfTargets
4   4   420         Say ErrorMsg
4   4   421         "PIPE LITERAL" Date(S) Time() Userid() ErrorMsg ,
4   4   422               "|>>" JournalFileID
4   4   423         "PIPE LITERAL" Date(S) Time() Userid() ErrorMsg ,
4   4   424               "|>>" ErrorLogFileID
4   4   425         Select
4   5   426           When (ThisUnit-1)<LimitValue then do
5   6   427           NumberOfTargets = ThisUnit - 1
4   6   428           end;
4   5   429           Otherwise nop;
4   4   430         End;
4   4   431         Leave ThisUnit
3   4   432         end;
3   2   433     End;
2   2   434   End;
2   2   435
2   2   436   NewRecord.0 = Min((ThisUnit - 1),LimitValue)
2   2   437     "PIPE STEM NewRecord. " ,
2   2   438           "| HEXSORT WORD 3" ,
2   2   439           "|>>" MergedMapFileID
2   2   440     Call MapResourceMismatches
2   2   441     Call PurgeSourceType
2   2   442     Call PurgeTargetType
2   2   443     Iterate Type
2   2   444
1   2   445   end;
```


This Sheet Has Been Replaced

```
1   2   446
1   1   447                     Otherwise do
2   2   448                         Iterate Type
1   2   449                         end;
1       450                  End;
1       451
1       452                 InfoMsg = "MJFMCM041I Processing" Left(ThisDeviceType,8,' ') "complete."
                                    |
+FANGA00773I Instruction may never be executed
1       453                 Say InfoMsg
```


This Sheet Has Been Replaced

```
⇒ Source Listing                    MSFMCH EXEC                    Date: 2000-02-03
Compiler REXX/370 3.0  LVL PQ28073    Time: 15:45:45
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----
          454            End;
          455
          456  MatchGeneralAllocationsEnd: Return;                                  */
          457  /* ------------------------------------------------------------------ */
          458  /*                                                                    */
          459  /* ------------------------------------------------------------------ */
          460  ExtractThisDeviceType: Parse Upper Arg ThisDeviceType .
          461                 ThisDeviceType = Left(Strip(ThisDeviceType,'B',' '),5,' ')
          462                 Drop SourceOfType.
          463                 Drop TargetOfType.
          464                 DeviceModelField = Pos_D_DPMODEL||"."||Len_D_DPMODEL
          465                 "PIPE STEM SourceRecord."  ,
          466                     "| LOCATE /*DEFINE  "||ThisDeviceType||"/"  ,
          467                     "| SORT" DeviceModelField "ASCENDING"  ,
          468                     "| Stem SourceOfType."
          469                 "PIPE STEM TargetRecord."  ,
          470                     "| LOCATE /*DEFINE  "||ThisDeviceType||"/"  ,
          471                     "| SORT" DeviceModelField "ASCENDING"  ,
          472                     "| Stem TargetOfType."
          473                 NumberOfSources = SourceOfType.0
          474                 NumberOfTargets = TargetOfType.0
          475  ExtractThisDeviceTypeEnd: Return;                                    */
          476  /* ------------------------------------------------------------------ */
          477  /*                                                                    */
          478  /* ------------------------------------------------------------------ */
          479  /*                                                                    */
          480  /*       We've mapped all the matching devices we can, now note       */
          481  /*            unfilled requests to the shortage file                  */
          482  /*            and overages to the surplus file                        */
          483  MapResourceMismatches: Nop;
          484           Call LogResourceMismatch
          485           Select
          486              When NumberOfSources > NumberOfTargets then call MapTheShortage
1         487              Otherwise call MapTheSurplus
          488           End;
          489  MapResourceMismatchesEnd: Return;                                    */
          490  /* ------------------------------------------------------------------ */
          491  /*                                                                    */
          492  /* ------------------------------------------------------------------ */
          493  MapTheShortage: Nop;
          494            Do ShortFall = NumberOfTargets by 1 to NumberOfSources
```


This Sheet Has Been Replaced

```
1       495                Index = (ShortFall - NumberOfTargets) + 1
1       496                ShortageRecord.Index = Left(SourceOfType.ShortFall,LengthOfDevpathRecord,' ')
        497             End;
        498             ShortageRecord.0 = Index
        499             "PIPE STEM ShortageRecord. " ,
        500                  "| DROP FIRST 1" ,
        501                  "|>>" ShortageFileID
        502  MapTheShortageEnd: Return;
        503  /* ------------------------------------------------------------------ */
        504  /*                                                                    */
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                505  /* ------------------------------------------------------------------ */
                506  MapTheSurplus:  Nop;
                507           Do Surplus = NumberOfSources by 1 to NumberOfTargets
             1  508              Index = (Surplus - NumberOfSources) + 1
             1  509              SurplusRecord.Index = Left(TargetOfType.Surplus,LengthOfDevpathRecord,' ')
                510           End;
                511           SurplusRecord.0 = Index
                512          "PIPE Stem SurplusRecord." ,
                513           "| DROP FIRST 1" ,
                514           "|>>" SurplusFileID
                515  MapTheSurplusEnd: Return;                                          */
                516  /* ------------------------------------------------------------------ */
                517  /*                                                                    */
                518  /* ------------------------------------------------------------------ */
                519  LogResourceMismatch: Nop;
                520           Select
             1  521              When NumberOfSources >  NumberOfTargets then do
             1  522                   Say "MJFNCM008E Insufficient" ThisDeviceType "devices in target configuration"
             1  523                   WarningMsg =  ,
             1  524                              "MJFHCM002W Device type:" Left(ThisDeviceType,5,' '),
             1  525                                         Right(NumberOfSources,6,' ') "required /" ,
             1  526                                         Right(NumberOfTargets,6,' ') "available."
             1  527                   Say WarningMsg
             1  528                   Shortage = Shortage + 1
             1  529                   "PIPE LITERAL" Date(S) Time() USERID() WarningMsg ,
             1  530                        "|>>" ErrorLogFileID
                531                end;
                532              Otherwise nop;
                533           End;
                534  LogResourceMismatchEnd: Return;                                    */
                535  /* ------------------------------------------------------------------ */
                536  /*                                                                    */
                537  /* ------------------------------------------------------------------ */
                538  LogUnknownSourceTypes: Nop;
                539
                540           "PIPE Stem SourceRecord." ,
                541              "| DROP FIRST 1" ,
                542              "| COUNT LINES " ,
                543              "| Var ResidualTypes "
                544
                545           Select
```

H22

305

This Sheet Has Been Replaced

```
1  2  546    When ResidualTypes > 0 then do
1  2  547      "PIPE Stem SourceRecord." ,
1  2  548           "| DROP FIRST 1" ,
1  2  549           "| COUNT LINES " ,
1  2  550           "| >>" ShortageFileID
1  2  551
1  2  552           WarningMsg = ,
1  2  553                        "MJFMCM009E Unknown source device type encountered."
1  2  554           Say WarningMsg
1  2  555           "PIPE LITERAL" Date(S) Time() USERID() WarningMsg ,
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8-
                                    "|>>" ErrorLogFileID.
 1   2   556                    end;
     2   557                Otherwise nop
     1   558            End;
         559    LogUnknownSourceTypesEnd: Return;                                    */
         560    /* ------------------------------------------------------------------*/
         561    /*                                                                   */
         562    /* ------------------------------------------------------------------*/
         563    TestModelCompatability: Nop;
         564            Parse Var SourceOfType.ThisUnit "*DEFINE " SourceDevType.ThisUnit "*" .
         565            SourceDevType.ThisUnit = Left(Word(SourceDevType.ThisUnit,1),5,' ')
         566            Parse Var SourceOfType.ThisUnit . "*MODEL= " SourceModelType.ThisUnit"*" .
         567
         568            Parse Var TargetOfType.ThisUnit "*DEFINE " TargetDevType.ThisUnit "*" .
         569            TargetDevType.ThisUnit = Left(Word(TargetDevType.ThisUnit,1),5,' ')
         570            Parse Var TargetOfType.ThisUnit . "*MODEL= " TargetModelType.ThisUnit "*" .
         571
         572            Select
         573                 When SourceDevType.ThisUnit = "3380" & Space(SourceModelType.ThisUnit,0) = "" then d
     1   574                     C o
                                 SourceModelType.ThisUnit = "NONE" then do
 1   2   575                     SourceModelType.ThisUnit = Left(Default3380Model,8,' ')
 1   2   576                     end;
     2   577                 When SourceDevType.ThisUnit = "3380" & SourceModelType.ThisUnit = "NONE" then do
     1   578                     SourceModelType.ThisUnit = "NONE" then do
 1   2   579                     SourceModelType.ThisUnit = Left(Default3380Model,8,' ')
 1   2   580                     end;
     2   581                 When SourceDevType.ThisUnit = "3390" & Space(SourceModelType.ThisUnit,0) = "" then d
     1   582                     C o
                                 SourceModelType.ThisUnit = "NONE" then do
 1   2   583                     SourceModelType.ThisUnit = Left(Default3390Model,8,' ')
 1   2   584                     end;
     2   585                 When SourceDevType.ThisUnit = "3390" & SourceModelType.ThisUnit = "NONE" then do
     1   586                     SourceModelType.ThisUnit = "NONE" then do
 1   2   587                     SourceModelType.ThisUnit = Left(Default3390Model,8,' ')
 1   2   588                     end;
     2   589                Otherwise nop;
     1   590            End;
         591
         592            Select
         593                 When TargetDevType.ThisUnit = "3380" & Space(TargetModelType.ThisUnit,0) = "" then d
     1   594
```

This Sheet Has Been Replaced

```
1   2   595                 TargetModelType.ThisUnit = "NONE"
1   2   596                 TargetModelType.ThisUnit = Left(Default3380Model,8,' ')
    2   597                 end;
    1   598             When TargetDevType.ThisUnit = "3380" & TargetModelType.ThisUnit = "NONE" then do
1   2   599                 TargetModelType.ThisUnit = "NONE"
1   2   600                 TargetModelType.ThisUnit = Left(Default3380Model,8,' ')
    2   601                 end;
    1   602             When TargetDevType.ThisUnit = "3390" & Space(TargetModelType.ThisUnit,0) = "" then d
          C o
```


This Sheet Has Been Replaced

```
Source Listing
ompiler REXX/370 3.0  LVL PQ28073    Time: 15:45:45  MJFMCM EXEC Date: 2000-02-08
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9
                                TargetModelType.ThisUnit = "NONE"
 1   2   603                    TargetModelType.ThisUnit = Left(Default3390Model,8,' ')
 1   2   604                    end;
     2   605                When TargetDevType.ThisUnit = "3390" & TargetModelType.ThisUnit = "NONE" then do
     1   606                    TargetModelType.ThisUnit = "NONE"
 1   2   607                    TargetModelType.ThisUnit = Left(Default3390Model,8,' ')
 1   2   608                    end;
     2   609                Otherwise nop;
     1   610            End;
         611
         612            Select
         613                When Space(TargetDevType.ThisUnit,0) = "3380" then Hierarchy= Sequence3380
     1   614                When Space(TargetDevType.ThisUnit,0) = "3390" then Hierarchy= Sequence3390
     1   615                Otherwise Hierarchy = ""
     1   616            End;
         617
         618            SourceHierarchy.ThisUnit  = WordPos(SourceModelType.ThisUnit,Hierarchy)
         619            TargetHierarchy.ThisUnit  = WordPos(TargetModelType.ThisUnit,Hierarchy)
         620
         621            Select
         622                When ( SourceDevType.ThisUnit   = TargetDevType.ThisUnit   ) &  ,
     1   623                     ( SourceModelType.ThisUnit = TargetModelType.ThisUnit )    ,
     1   624                     then ModelsAreCompatible = 1
     1   625                When ( SourceDevType.ThisUnit   = TargetDevType.ThisUnit   ) &  ,
     1   626                     ( SourceHierarchy.ThisUnit <= TargetHierarchy.ThisUnit )   ,
     1   627                     then do
     1   628                          ErrorMsg="MJFMCM011W Substituted" Left(Space(TargetDevType.ThisUnit,0),5,'
 1   2   629                                 ! C ') ,
                                              || "-"|| Left(Space(TargetModelType.ThisUnit,0),8,' ') ,
                                              ||"("|| Left(Space(Word(TargetOfType.ThisUnit,3),0),4,' ')||")" ,
                                                "for" Left(Space(SourceDevType.ThisUnit,0),5,' ') ,
                                              || "-"|| Left(Space(SourceModelType.ThisUnit,0),8,' ')
                                              || "("|| Left(Space(Word(SourceOfType.ThisUnit,3),0),4,' ')||")"
 1   2   630                     Say ErrorMsg
 1   2   631                     "PIPE LITERAL" Date(S) Time() Userid() ErrorMsg ,
 1   2   632                          "|>>" JournalFileID
 1   2   633                     ModelsAreCompatible = 1
 1   2   634                     end;
 1   2   635               Otherwise ModelsAreCompatible = 0
 1   2   636           End;
 1   2   637   TestModelCompataibility: Return
 1   2   638
     2   639
     1   640
         641
         642
```

309

This Sheet Has Been Replaced

```
643   /* ---------------------------------------------------------------- */
644   /*                                                                  */
645   /* ---------------------------------------------------------------- */
646   LoadSourceAndTargetFiles: Nop;
647             "PIPE DISK" SourceFileID ,
648                 "|STEM SourceRecord."
649             "PIPE DISK" TargetFileID ,
650                 "|STEM TargetRecord."
651   LoadSourceAndTargetFilesEnd: Return;
652   /* ---------------------------------------------------------------- */
```

This Sheet Has Been Replaced

```
653   /*                                                                  */
654   /* ---------------------------------------------------------------- */
655   PurgeTargetElement: Nop;
656           "PIPE STEM TargetRecord." ,
657                 "| NLOCATE /*DEFINE   "||TargetType TargetAddr||"/" ,
658                 "| Stem TargetRecord."
659   PurgeTargetElementEnd: Return;
660   /* ---------------------------------------------------------------- */
661   /*                                                                  */
662   /* ---------------------------------------------------------------- */
663   PurgeSourceElement: Nop;
664           "PIPE STEM SourceRecord." ,
665                 "| NLOCATE /*DEFINE   "||SourceType SourceAddr||"/" ,
666                 "| Stem SourceRecord."
667   PurgeSourceElementEnd: Return;
668   /* ---------------------------------------------------------------- */
669   /*                                                                  */
670   /* ---------------------------------------------------------------- */
671   PurgeSourceType: Nop;
672           "PIPE STEM SourceRecord." ,
673                 "| NLOCATE /*DEFINE   "||ThisDeviceType||"/" ,
674                 "| Stem SourceRecord."
675   PurgeSourceTypeEnd: Return;
676   /* ---------------------------------------------------------------- */
677   /*                                                                  */
678   /* ---------------------------------------------------------------- */
679   PurgeTargetType: Nop;
680           TargetDevType = Left(Space(ThisDeviceType,0),9,' ')
681           "PIPE STEM TargetRecord." ,
682                 "| NLOCATE /*DEFINE   "||ThisDeviceType||"/" ,
683                 "| Stem TargetRecord."
684   PurgeTargetTypeEnd: Return;
685   /* ---------------------------------------------------------------- */
686   /*                                                                  */
687   /* ---------------------------------------------------------------- */
```


This Sheet Has Been Replaced

XX Lines 687

ze of compiled program in bytes 49728

tal messages	Informational	Warning	Error	Severe	Terminating
2	1	1	0	0	0

s following lines have been flagged

452

:or No. Line

550 3

773 452

lishing time of compilation: 15:45:46



APPENDIX J

This Sheet Has Been Replaced



essage(s) reported. Highest severity code was 4 - Warning

piler Options

```
  NOALTERNATE
    CEXEC      (MJFSDC   CEXEC    G1)                    RECFM=F,LRECL=1024
  NOCOMPILE    (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG       (I)
  NOIEXEC
    LIBLEVEL   (*)
    LINECOUNT  (55)
    MARGINS    (1 *)
  NOOBJECT
314 OPTIMIZE
    PRINT      (MJFSDC   LISTING  G1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN      (MJFSDC   EXEC     G1 R § R U R § R \ (   PD) RECFM=V,LRECL=99
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 4
```

This Sheet Has Been Replaced

```
> Source Listing                                MJFSDC   EXEC     V1
 Compiler REXX/370 3.0  LVL PQ28073    Time: 15:50:54                Date: 2000-02-03                Page:     2
  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
            1   /*    MJFSDC - Summarize Device Count                                                  */
            2
            3           Address CMS;
            4           Trace Value Off;
FANGA00550W Unsupported TRACE options will default to OFF
            5           Parse Upper Arg InFileName InFileType InFileMode .
            6           InFileID = InFileName InFileType InFileMode
            7           OutFileID = InFileName "SUMMARY" InFileMode
            8           Rcs = 0
            9
           10   MainPgm: Nop;
           11           Say "MJFSDC002I Initializing."
           12           Call ProcessProfile
           13           Say "MJFSDC003I Processing in progress. Please wait."
           14           "PIPE DISK" InFileID ,
           15                "| COUNT LINES " ,
           16                "| VAR TotalStatements"
           17           Say "MJFSDC006I" Right(TotalStatements,6,'0') "input records. "
           18           "PIPE DISK" InFileID ,
           19                "| LOCATE /*DEFINE/" ,
           20                "| COUNT LINES " ,
           21                "| VAR TotalDefinitions"
           22           Say "MJFSDC006I" Right(TotalDefinitions,6,'0') "units defined."
           23           "PIPE DISK" InFileID ,
           24                "| LOCATE /*DEFINE/" ,
           25                "| HEXSORT UNIQUE 19.7  " ,
           26                "| COUNT LINES " ,
           27                "| VAR UniqueDefinitions"
           28           Say "MJFSDC006I" Right(UniqueDefinitions,6,'0') "unique units defined."
           29           "PIPE CMS ERASE" OutFileID "| HOLE"
           30           Do Index = 1 to words(SelectedTypes)
1          31                ThisType.Index = word(SelectedTypes,Index)
1          32                "PIPE DISK" InFileID ,
1          33                     "| LOCATE /"||ThisType.Index||"/" ,
1          34                     "| COUNT LINES " ,
1          35                     "| VAR" Count.Index
1          36                OutLine.Index = "MJFSDC001I" Date(S) Time() ,
1          37                                Right(Count.Index,6,0) Left(ThisType.Index,8,' ') ,
1          38                                "devices specified."
1          39                "PIPE LITERAL" OutLine.Index  ,
```

315

This Sheet Has Been Replaced

```
1                                   "|>>" OutFileID
                                End;
                                Select
                                    When TotalDefinitions <> UniqueDefinitions then do
                                        Delta = TotalDefinitions - UniqueDefinitions
                                        ErrorLine = "MJFSDC005W" Date(s) Time() ,
                                            Right(Delta,6,'0') ,
                                            "Overlapped/duplicate assignments defined."
                                        Say Errorline
                                        "PIPE LITERAL" ErrorLine ,
```


This Sheet Has Been Replaced

```
> Source Listing                                    MJFSDC   EXEC     V1
Compiler REXX/370 3.0  LVL PQ28073    Time: 15:50:54                Date: 2000-02-03                Page:    3
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
  1   2     50                                  "|>>" OutFileID
  1   2     51                             Rcs = 4
        2     52                          end
        1     53                    Otherwise nop
              54              End;
              55              "PIPE DISK" InFileID ,
              56                   "| HEXSORT 1.40 " ,
              57                   "| STEM TempCopy."
              58
              59              "PIPE CMS  ERASE" InFileID "| HOLE"
              60
              61              "PIPE STEM TempCopy. | HEXSORT 20.5 |>" InFileID
              62
              63              "PIPE DISK" InFileID ,
              64                   "| SPECS 1.40" ,
              65                   "| HEXSORT 19.7 " ,
              66                   "| STEM AllUnitDefinitions."
              67
              68              "PIPE DISK" InFileID ,
              69                   "| SPECS 1.25" ,
              70                   "| HEXSORT UNIQUE 19.7 " ,
              71                   "| STEM UniqueUnitDefinitions."
              72
              73              Do ThisRecord = 1 to UniqueUnitDefinitions.0
  1           74                   Select
  1   1       75                       When AllUnitDefinitions.ThisRecord =  UniqueUnitDefinitions.ThisRecord then nop;
  1   1       76                       Otherwise do
  2   2       77                            ErrorLine="MJFSDC007E Definition overlap begins at record" ThisRecord
  2   2       78                            Say ErrorLine
  2   2       79                            "PIPE LITERAL" ErrorLine ,
  2   2       80                                 "|>>" OutFileID
  1   2       81                            ThisRecord = UniqueUnitDefinitions.0
  1           82                            end;
              83                   End;
              84              End;
              85              Say "MJFSDC004I Processing complete. Exiting."
              86   MainEnd:   Call ExitRtn Rcs
              87
              88   ExitRtn:   Parse Arg Retcode
              89              Exit Retcode
              90   ExitRtnEnd: Return;
```

317

This Sheet Has Been Replaced



```
 91
 92  ProcessProfile: Nop;
 93                  "PIPE DISK SCRAMIEE. PROFILE *"  ,
 94                       "| STEM ConfigRecord."
 95                  Do Record = 1 to ConfigRecord.0
 96                     INTERPRET ConfigRecord.Record
 97                  End;
 98                  "PIPE LITERAL" SelectedTypes  ,
 99                     "| SPLIT AFTER BLANK"  ,
100                     "| SORT UNIQUE" ,
```


This Sheet Has Been Replaced

```
        101                     "| JOIN *" ,
        102                     "| VAR SelectedTypes"
        103   ProcessProfileEnd: Return;
```

319

This Sheet Has Been Replaced

> Compilation Statistics
Compiler REXX/370 3.0 LVL PQ26073 Time: 15:50:54 HJFSDC EXEC 01 Date: 2000-02-03 Page: 5

X Lines 103

e of compiled program in bytes 11064

al messages Informational Warning Error Severe Terminating
1 0 1 0 0 0

following lines have been flagged

or No. Line

550 4

shing time of compilation: 15:50:54

320

This Sheet Has Been Replaced

APPENDIX K

This Sheet Has Been Replaced



Compilation Summary

(Compiler REXX/370 3.0 LVL PQ28073 Time: 15:50:50 Date: 2000-02-03 Page: 1

SCRAMIEE EXEC G1

nessage(s) reported. Highest severity code was 4 - Warning

npiler Options

```
NOALTERNATE
  CEXEC     (SCRAMIEE CEXEC    G1)                    RECFM=F,LRECL=1024
NOCOMPILE   (S)
NOCONDENSE
NODLINK
NODUMP
  FLAG      (I)
NOIEXEC
  LIBLEVEL  (*)
  LINECOUNT (55)
  MARGINS   (1 *)
NOOBJECT
  OPTIMIZE
  PRINT     (SCRAMIEE LISTING  G1)                   RECFM=V,LRECL=121
NOSAA
NOSLINE
  SOURCE
  SYSIN     (SCRAMIEE EXEC     G1 R § R V R § R \    °   GD) RECFM=V,LRECL=98
NOTERMINAL
NOTESTHALT
NOTRACE
NOXREF
Minimum Library Level required: 4
```

322

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
Source Listing                    SCRAMIEE EXEC  G1
             1 /* ------------------------------------------------------------- */
             2 /*   SCRAMIEE - Server/390 Continuity and Recovery Automation Manager */
             3 /*   (c) 2000 IBM Corporation All Rights Reserved.               */
             4 /*   Patent Application Pending                                  */
             5 /* ------------------------------------------------------------- */
             6 /* ============================================================= */
             7 /* Usage: SCRAMIEE ScramCommand                                  */
             8 /* YYYYMMDD WHO Change Description                               */
             9 /* -------- --- ------------------------------------------------ */
            10 /* 20000115 MJF Initial Release                                  */
            11 /* ============================================================= */
            12           Address CMS;
            13           Trace Value Off;
                                 |
-FANGA00550W Unsupported TRACE options will default to OFF
            14           Parse Upper Arg ArgString
            15           Rcs = 0
            16           Call ProcessProfile
            17 /* ------------------------------------------------------------- */
            18 /*                                                               */
            19 /* ------------------------------------------------------------- */
            20 Main:      Nop;
            21           Select
       1    22               When Word(Argstring,1) = "ALLOCATE" then do
     1 2    23                    Call MakeNewDirectory
       2    24                    end;
       1    25               When Word(Argstring,1) = "ACCESS" then do
     1 2    26                    Call AccessDirectory
       2    27                    end;
       1    28               When Word(Argstring,1) = "RELEASE" then do
     1 2    29                    Call ReleaseDirectory
       2    30                    end;
       1    31               When Word(Argstring,1) = "ANALYZE" then do
     1 2    32                    Call Analyze
       2    33                    end;
       1    34               When Word(Argstring,1) = "CUSTOMIZE" then do
     1 2    35                    Call Customize
       2    36                    end;
       1    37               When Word(Argstring,1) = "RESTRICT" then do
     1 2    38                    Call Restrictions
       2    39                    end;
```


This Sheet Has Been Replaced

```
1   2    41          Call MatchResources
    2    42          end;
    1    43     When Word(Argstring,1) = "HELP" then do
1   2    44          Call TellHelp
    2    45          end;
    1    46     When Word(Argstring,1) = "GENERATE" then do
1   2    47          Call Generate
    2    48          end;
    1    49     When Word(Argstring,1) = "COMPRESS" then do
    1    40     when Word(Argstring,1) = "MAP" then do
```

This Sheet Has Been Replaced

Do Sel Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0

Source Listing SCRAMIEE EXEC G1

```
 1   2    50                  Call Compress
     2    51                  end;
     1    52                When Word(Argstring,1) = "EXPAND" then do
 1   2    53                  Call Expand
     2    54                  end;
     1    55                When Word(Argstring,1) = "LIBLIST" then do
 1   2    56                  Call ListTheLibrary
     2    57                  end;
     1    58                When Word(Argstring,1) = "RELEASE" then do
 1   2    59                  Call ReleaseDirectory
     2    60                  end;
     1    61                When Word(Argstring,1) = "DISCARD" then do
 1   2    62                  Call Purge
     2    63                  end;
     1    64                When Word(Argstring,1) = "FILELIST" then do
 1   2    65                  Call ListFiles
     2    66                  end;
     1    67                When Word(Argstring,1) = "EDIT" then do
 1   2    68                  Call EditFiles
     2    69                  end;
     1    70                When Word(Argstring,1) = "SUMMARIZE" then do
 1   2    71                  Call Summarize
     2    72                  end;
     1    73                When Word(Argstring,1) = "QSEARCH" then do
 1   2    74                  Call QuerySearch
     2    75                  end;
     1    76                When Word(Argstring,1) = "QCONTRACT" then do
 1   2    77                  Call QueryContract
     2    78                  end;
     1    79                When Word(Argstring,1) = "QUIT" then do
 1   2    80                  Say "MJFCMD012I Exiting."
 1   2    81                  Exit 8
     2    82                  end;
     1    83                Otherwise do
 1   2    84                  Say Date(S) Time() "MJFCMD008E Invalid command:" ,
 1   2    85                        "'"||Word(Argstring,1)||"'."
 1   2    86                  Rcs = 8
     2    87                  end;
          88                End;
          89  MainEnd:      Exit Rcs
          90  /* ---------------------------------------------------------------- */
```

This Sheet Has Been Replaced

```
92   /* ---------------------------------------------------------------- */
93   MakeNewDirectory: Nop;
94                CommandTarget = Delword(ArgString,1,1)
95                CommandTarget = Strip(CommandTarget,'L',' ')
96                BaseDirectoryID = Strip(BaseDirectoryID,'B',' ')
97                Address CMS "CREATE DIRECTORY" BaseDirectoryID||CommandTarget
98                Rcs = rc
99   MakeNewDirectoryEnd: Nop;
100  /* ---------------------------------------------------------------- */


91   /* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - */
```


This Sheet Has Been Replaced

```
Compiler REXX/370 3.0  LVL PQ28073    Time: 15:50:50          Date: 2000 02 03
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                                SCRAMJEE EXEC  G1
       101  /*                                                                  */
       102  /* ---------------------------------------------------------------- */
       103  AccessDirectory:       Nop;
       104                 CommandTarget = Delword(ArgString,1,1)
       105                 Address CMS "ACCESS" CommandTarget
       106                 Rcs = rc
       107  AccessDirectoryEnd:    Return;
       108  /* ---------------------------------------------------------------- */
       109  /*                                                                  */
       110  /* ---------------------------------------------------------------- */
       111  ReleaseDirectory:  Nop;
       112                 LastWord = Words(Argstring)
       113                 CommandTarget = Word(ArgString,LastWord)
       114                 Address CMS "RELEASE" CommandTarget
       115                 Rcs = rc
       116  ReleaseDirectoryEnd:   Return;
       117  /* ---------------------------------------------------------------- */
       118  /*                                                                  */
       119  /* ---------------------------------------------------------------- */
       120  Analyze:       Nop;
       121                 Specification = Delword(ArgString,1,1)
       122                 Upper Specification
       123                 Parse Var Specification ObjectType Object
       124                 Select
     1 2   125                      When Left(Space(ObjectType,0),3) = "RES" then do
   1   2   126                           Call MakeDEVPATHSfromSchedule
     2   127                           end;
     1   128                      When Left(Space(ObjectType,0),3) = "REQ" then do
   1   2   129                           Call MakeDEVPATHSfromRDWS
     2   130                           end;
     1   131                      Otherwise do
   1   2   132                           Say "MJFCMD024E Object type unsupported."
   1   2   133                           Rcs = 24
     2   134                           end;
       135                 End;
       136  AnalyzeEnd:   Return;
       137  /* ---------------------------------------------------------------- */
       138  /*                                                                  */
       139  /* ---------------------------------------------------------------- */
       140  Restrictions:      Nop;
       141                 CommandTarget = Delword(ArgString,1,1)
```

Source Listing

This Sheet Has Been Replaced

```
143                 Rcs = rc
144   RestrictionsEnd:  Return;
145   /* ---------------------------------------------------------------- */
146   /*                                                                  */
147   /* ---------------------------------------------------------------- */
148   TellHelp:       Nop;
149                   Address CMS "HELP SCRAMIER MENU"
150                   Rcs = rc
151   TellHelpEnd:    Return;
```

142 Address CMS "MJFUMC" Command Target

This Sheet Has Been Replaced

```
1 Compilor REXX/370 3.0  LVL PQ28073    Time: 15:50:50                Date: 2000-02-03
E  Do Sol  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                                  SCRAMIBE EXEC  91
                  /* ---------------------------------------------------------- */
                  /*                                                            */
                  /* ---------------------------------------------------------- */
                  MatchResources: Nop;
                                  CommandTarget = Delword(ArgString,1,1)
                                  Address CMS "MJFMCM" CommandTarget "( WORKDISK" Word(CommandTarget,3)
                                  Rcs = rc
                  MatchResourcesEnd:  Return;
                  /* ---------------------------------------------------------- */
                  /*                                                            */
                  /* ---------------------------------------------------------- */
                  Generate:       Nop;
                                  CommandTarget = Delword(ArgString,1,1)
                                  Select
                                      When Word(CommandTarget,1) = "VMDIRECT" then do
                                           Parse Var CommandTarget "FROM" ,
                                                InPathMapID
                                           "MJFACG"
                                           end;
                                      Otherwise do
                                           Say Date(S) Time() "MJFCMD012E Invalid operand:" CommandTarget
                                           Rcs = 8
                                           end;
                                  End;
                  GenerateEnd:    Return;
                  /* ---------------------------------------------------------- */
                  /*                                                            */
                  /* ---------------------------------------------------------- */
                  Compress:       Nop;
                                  CommandTarget = Delword(ArgString,1,1)
                                  Parse Var CommandTarget FileSpecs "INTO" TargetLibName .
                                  WorkingFileMode = Subword(FileSpecs,3)
                                  "FCOPY" FileSpecs TargetLibName "PACKLIB" WorkingFileMode "(INSERT PACK ISODATE"
                                  Rcs = rc
                  CompressEnd:    Return;
                  /* ---------------------------------------------------------- */
                  /*                                                            */
                  /* ---------------------------------------------------------- */
                  Expand:         Nop;
                                  CommandTarget = Delword(ArgString,1,1)
                                  Parse Var CommandTarget FileSpecs "FROM" TargetLibName .
```

This Sheet Has Been Replaced

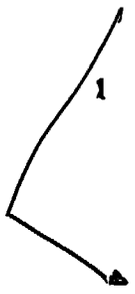
```
194                 "FCOPY" TargetLibName "PACKLIB" WorkingFileMode "(UNPACK SELECT" FileSpecs
195                 Rcs = rc
196   ExpandEnd:    Return;
197   /* ---------------------------------------------------------------- */
198   /*                                                                  */
199   /* ---------------------------------------------------------------- */
200   Purge:        Nop;
201                 CommandTarget = Delword(ArgString,1,1)
202                 Say Date(S) Time() "MJFCMD010W Warning! This will destroy '"||CommandTarget||"'."
```

193 WorkingFileMode = Subword(Filespecs, 3)

This Sheet Has Been Replaced

```
f  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                                    SCRAMIEE EXEC  G1
          203              Say Date(S) Time() "MJFCMD011R Do you want to discard these files? <Y|N>"
          204              Do Until Response = "Y" | Response = "N"
    1     205                 Parse Upper Pull ThisResp .
    1     206                 Response = Left(Space(ThisResp,0),1)
          207              End;
          208              Select
        1 209                 When Response = "Y" then do
    1   2 210                      "PIPE CMS DISCARD" CommandTarget "| HOLE"
        2 211                      end;
        1 212                 Otherwise do
    1   2 213                      Say Date(S) Time() "MJFCMD012I Cancelled at user request."
        2 214                      end
          215              End;
          216   PurgeEnd:  Return;
          217   /* ---------------------------------------------------------------- */
          218   /*                                                                  */
          219   /* ---------------------------------------------------------------- */
          220   Customize: Nop;
          221              CommandTarget = Delword(ArgString,1,1)
          222              Address CMS "XEDIT" Word(ArgString,2) "RESCNTRL" Word(Argstring,4)
          223              Address CMS "MJFUMC" CommandTarget
          224              Rcs = rc
          225   CustomizeEnd: Return;
          226   /* ---------------------------------------------------------------- */
          227   /*                                                                  */
          228   /* ---------------------------------------------------------------- */
          229   ListFiles: Nop;
          230              "FILELIST" Delword(ArgString,1,1)
          231   ListFilesEnd: Return;
          232   /* ---------------------------------------------------------------- */
          233   /*                                                                  */
          234   /* ---------------------------------------------------------------- */
          235   EditFiles: Nop;
          236              "XEDIT" Delword(ArgString,1,1)
          237   EditFilesEnd: Return;
          238   /* ---------------------------------------------------------------- */
          239   /*                                                                  */
          240   /* ---------------------------------------------------------------- */
          241   Summarize: Nop;
          242              CommandTarget = Delword(ArgString,1,2)
          243              Parse Var CommandTarget CommandTarget1 "(" DisplayOption .
```


This Sheet Has Been Replaced



```
245                 Select
246                     When Space(DisplayOption,0) = "NODISPLAY" then nop
247                     Otherwise do
248                         Address CMS "XEDIT"  Word(CommandTarget1,1) "SUMMARY" Word(CommandTarget1,3)
249                         end;
250                 End;
251                 Rcs = rc
252 SummarizeEnd: Return;
253 /* ------------------------------------------------------------------- */
```

244 Address CMS "MJF30C" CommandTarget1

This Sheet Has Been Replaced

Source Listing

SCRAMJEE EXEC G1

```
E Do Sal Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                254 /*                                                                  */
                255 /* ---------------------------------------------------------------- */
                256 ProcessProfile: Nop;
                257             "PIPE DISK SCRAMJEE  PROFILE *"  ,
                258                  "| STEM ConfigRecord."
                259             Do Record = 1 to ConfigRecord.0
              1 260                INTERPRET ConfigRecord.Record
                261             End;
                262             "PIPE CMS ACCESS WSCVMU:WSCBRS. M | HOLE"
                263             "PIPE CMS ACCESS WSCVMU:WSCBRS.BRSTOOLS. N | HOLE"
                264 ProcessProfileEnd: Return;
                265 /* ---------------------------------------------------------------- */
                266 /*                                                                  */
                267 /* ---------------------------------------------------------------- */
                268 QuerySearch: Nop;
                269             "CMS QUERY ACCESSED"
                270 QuerySearchEnd: Return;
                271 /* ---------------------------------------------------------------- */
                272 /*                                                                  */
                273 /* ---------------------------------------------------------------- */
                274 QueryContract: Nop;
                275             "MJFQCC" Delword(ArgString,1,1)
                276 QueryContractEnd: Return;
                277 /* ---------------------------------------------------------------- */
                278 /*                                                                  */
                279 /* ---------------------------------------------------------------- */
                280 ListTheLibrary: Nop;
                281             "FCOPY" Word(Delword(ArgString,1,1),1) "PACKLIB * (LIST"
                282 ListTheLibraryEnd: Return;
                283 /* ---------------------------------------------------------------- */
                284 /*                                                                  */
                285 /* ---------------------------------------------------------------- */
                286 MakeDEVPATHSfromRDWS: Nop;
                287             "MJFJFR" Object
                288 MakeDEVPATHSfromRDWSEnd: Return;
                289 /* ---------------------------------------------------------------- */
                290 /*                                                                  */
                291 /* ---------------------------------------------------------------- */
                292 MakeDEVPATHSfromSCHEDULE: Nop;
                293           CommandTarget = Delword(ArgString,1,2)
                294           Select
```


This Sheet Has Been Replaced

```
1   2   296          Parse Var CommandTarget "FOR" TargetCustomerNumber ,
1   2   297                                  "ON"  ScheduleDate ,
1   2   298                                  "AT"  TargetLocationCode ,
1   2   299                                  "ASSIGNED" TargetIOCPIdentifier .
    2   300          end;
    1   301        When Word(CommandTarget,1) = "IN" then do
1   2   302          Parse Var CommandTarget "IN" InRDWSFn InRDWSFt InRDWSFm .
1   2   303          "MJFJFR" InRDWSFn InRDWSFt InRDWSFm
1   2   304          Signal MakeDEVPATHSfromSCHEDULEEnd
```

1 2 295 when Word(Command Target, 1) = "FOR" then do

This Sheet Has Been Replaced

```
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
 Source Listing
 M compiler REXX/370 3.07 LVL PQ28073 SCRAMBLE EXEC G1   Time: 15:50:50   Date: 2000-02-08
      2  305                end;
      1  306             Otherwise do
   1  2  307               Say "MJFCMD050 Invalid RESOURCES selection. Must be" ,
   1  2  308                   "'FOR' or 'IN'."
   1  2  309               Exit 50
      2  310               end;
         311          End;
         312          ResourceLocation = Left(Space(TargetLocationCode,0),1)
         313
         314           Select
      1  315             When ResourceLocation = "B" then do
   1  2  316                  Location = "BLD"
   1  2  317                  IOCPLib = BaseDirectoryID||".RESOURCES.IOCPLIB.BLD"
   1  2  318                  "PIPE CMS ACCESS" IOCPLib "I | HOLE"
      2  319                  end;
      1  320             When ResourceLocation = "G" then do
   1  2  321                  Location = "GBG"
   1  2  322                  IOCPLib = BaseDirectoryID||".RESOURCES.IOCPLIB.GBG"
      2  323                  end;
      1  324             When ResourceLocation = "S" then do
   1  2  325                  Location = "STF"
   1  2  326                  IOCPLib = BaseDirectoryID||".RESOURCES.IOCPLIB.STF"
      2  327                  end;
      1  328             Otherwise nop;
         329           End;
         330          "PIPE CMS ACCESS" IOCPLib "I | HOLE"
         331          Say "MJFCMD210I IOCP Library" IOCPLib "accessed at filemode 'I'."
         332
         333          Select
      1  334              When Location = "GBG" then do
   1  2  335                  Call ProcessGBGSchedule
      2  336                  End;
      1  337              Otherwise do
   1  2  338                  Say "MJFCMD000T Support not yet available for" ,
   1  2  339                      "locations other than 'GBG'."
   1  2  340                  Exit 0
      2  341                  end;
         342          End;
         343
         344  MakeDEVPATHSfromSCHEDULEEnd: Return;
         345  /* ---------------------------------------------------------- */
```

335

This Sheet Has Been Replaced

```
347  /* ------------------------------------------------------------------ */
348  ProcessGBGSchedule: Nop
349                  ScheduleCustomer = TargetCustomerNumber
350                  ScheduleDateKey = Substr(ScheduleDate,4,2) || ,
351                                    "/" || Right(ScheduleDate,2) || ,
352                                    "/" || Left(ScheduleDate,4)
353
354                  "BRSRDSCH" ScheduleCustomer ScheduleDateKey "(NOPRINT FN="||ScheduleCustomer
355                  Rcs = rc
356  /*                                                                */
```


This Sheet Has Been Replaced

Source Listing

SCRAMJEE G1

```
Do Sel Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          356
          357                  Select
       1  358                      When Rcs = 0 then nop
       1  359                      Otherwise do
    1  2  360                           Say "MJFCMD205E Error" Rcs "in BRSRDSCH processing."
    1  2  361                           Exit Rcs
       2  362                           end;
          363                  End;
          364
          365                  OutFn = CustomerNumber
          366
          367                 "STATE" OutFn "IOCPRECS *"
          368                  Rcst= rc
          369                  Select
       1  370                      When Rcst= 0 then nop
       1  371                      Otherwise do
    1  2  372                           Say "MJFCMD206E Error" Rcst "locating DEVPATHS input file."
    1  2  373                           Exit Rcst
       2  374                           end;
          375                  End;
          376
          377                  "MJFPSO" OutFn "IOCPRECS *"
          378                  Rcsp= rc
          379                  Select
       1  380                      When Rcsp= 0 then nop
       1  381                      Otherwise do
    1  2  382                           Say "MJFCMD207E Error" Rcsp "creating DEVPATHS file."
    1  2  383                           Exit Rcsp
       2  384                           end;
          385                  End;
          386
          387  ProcessGBGScheduleEnd: Return;
```

337

This Sheet Has Been Replaced

=> Compilation Statistics SCRAMBLER G1

IX Lines 387

ze of compiled program in bytes 26080

al messages Informational Warning Error Severe Terminating
1 0 1 0 0 0

following lines have been flagged

or No. Line

550 13

shing time of compilation: 15:50:50

This Sheet Has Been Replaced

APPENDIX L

This Sheet Has Been Replaced



```
AllowedMachineModes= "370 XA ESA XC"
AllowedPrivClasses = "EFGHIJKLMNOPQRSTUVWXYZ0123456789"
CStorDefaultLow    = 1
CstorDefaultHigh   = 2048
CstorMaxLow        = 1
CstorMaxHigh       = 2048
NumberofCPUsMin    = 1
NumberofCPUsMax    = 12
XstorDefinedLow    = 0
XstorDefinedHigh   = 2048

BaseDirectoryID    = "WSCVMU:EBRS1;"

BlockCount33701          = 558000
BlockCount33702          = 712752
BlockCount9332400        = 360036
BlockCount9332600        = 554800
BlockCount9335           = 804714
BlockCount933610         = 920115
BlockCount933620         = 1672881

CylinderCount33301       = 404
CylinderCount333011      = 808
CylinderCount3350        = 555
CylinderCount3375        = 959
CylinderCount3380K       = 2655
CylinderCount3380E       = 1770
CylinderCount33901       = 1113
CylinderCount33902       = 2226
CylinderCount33903       = 3339
CylinderCount33909       = 10017
CylinderCount93451       = 1440
CylinderCount93452       = 2156
CylinderCount3380Default = 885
CylinderCount3390Default = 3339

Default3380Model         = J
Default3390Model         = 3
Sequence3380             = D J E K
Sequence3390             = 1 2 3 6 9

DASDTypes = "3310 3330 3340 3350 3370 3380 3390 9341 9342 9344 9345 9390 9391 9392 9393 9395 5500 3995"
TapeTypes = "3420 3422 3480 3490 3590 3495"
CommTypes = "3171 3172 3277 3278 3272 3274 3286 3270 3705 3725 3745 BSC1 BSC2 BSC3 TWX"
PrinTypes = "APP  1403 3203 3211 3800 3820 3900 4245 4248"
MiscTypes = "SCTC CTC  3088 R86K DUMMY 2740 2741P 3995 3791L"

CommCtlrs = "SCTC 3174 3745 CFS 3088 3791L 3705 9725"
MiscCtlrs =  CFS  3088 DUMMY RS6K 9032"
```

This Sheet Has Been Replaced

```
DASDCtlrs = "3880 3990 9343"
PrinCtlrs = "3811 3820 3827 3900 4245 AFP1 AFP3 3203"
TapeCtlrs = "3803 3835 3422 3480 3490 3590 3495"

ChanTypes   = "BL BY CNC CVC CFS CTC"
CtlrTypes   = DASDCtlrs TapeCtlrs CommCtlrs PrinCtlrs MiscCtlrs
DeviceTypes = DASDTypes TapeTypes CommTypes PrinTypes MiscTypes

SelectedTypes = DeviceTypes CtlrTypes
```


This Sheet Has Been Replaced

This Sheet Has Been Replaced



APPENDIX M

essage(s) reported. Highest severity code was 4 - Warning

piler Options

```
  NOALTERNATE
    CEXEC      (MJFAIS   CEXEC    A1)                    RECFM=F,LRECL=1024
  NOCOMPILE    (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG       (I)
  NOIEXEC
    LIBLEVEL   (*)
    LINECOUNT  (55)
    MARGINS    (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT      (MJFAIS   LISTING  A1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN      (MJFAIS   EXEC     A1 V:┬ V:M V:┬ V:] P°  PGD) RECFM=V,LRECL=84
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 3
```

This Sheet Has Been Replaced

Source Listing MJFAIS EXEC A1

```
Compiler REXX/370 3.0  LVL PQ28073    Time: 16:58:26          Date: 2000-02-03          Page:    2
 Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          1   /*  Copyright 1998 IBM Corporation                                              */
          2   /*  Patent application pending                                                  */
          3   /*                                                                             */
          4   /*  MJFAIS  - a REXX procedure to control the analysis and parsing             */
          5   /*            of IOCP input Streams                                            */
          6   /*                                                                             */
          7               Address CMS;
          8               Trace Value Off;
'ANGA00550W Unsupported TRACE options will default to OFF
          9               Parse Upper Arg InFileName InFileType InFileMode .
         10               Time(R);
         11
         12               Say "Enter Location:"
         13               Parse Upper Pull Location .
         14               Location = Left(Strip(Location,'L',' '),8,' ')
         15               Say "Enter System Name or blank:"
         16               Parse Upper Pull RealSystemName .
         17               ImageName = Left(Strip(RealSystemName,'L',' '),8,' ')
         18               Say "Enter Image Name or blank:"
         19               Parse Upper Pull ImageName .
         20               ImageName = Left(Strip(ImageName,'L',' '),8,' ')
         21
         22               InFileID = InFileName InFileType InFileMode
         23               WkFileID = InFileName "$$WORK$$" InFileMode
         24               ChFileID = InFileName "$$CHPS$$" InFileMode
         25               CUFileID = InFileName "$$CNTL$$" InFileMode
         26               DevFileID = InFileName "DEVPATHS" InFileMode
         27
         28               "ESTATEW" InFileID
         29               Rcs = rc
         30               Select
     1   31                   When Rcs = 0 then nop;
     1   32                   Otherwise do
 1   2   33                       Say "MJFAIS001E Error accessing" ,
 1   2   34                           "'"||InFileID||"'", RC="||Rcs
 1   2   35                       Exit Rcs
     2   36                       end;
         37               End;
         38
         39               "PIPE DISK" InFileID ,
```

This Sheet Has Been Replaced

```
40                "| ASMCONT "                  /* Aggregate multiple lines/statement */
41                "| NLOCATE 1:1 '*'"          /* Ignore comment records              */
42                "| MJFALI "                  /* ALign the Input records             */
43                "| SORT 10.8 ASCENDING" )    /* Sort by input specification type    */
44                ">" WkFileID
45
46
47          "PIPE DISK" WkFileID ,
48                "| LOCATE 10.10 /CNTLUNIT/" ,     /* Find all control units      */
49                ">" CUFileID
```

This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
     50
     51            "PIPE DISK" WkFileID ,
     52                 "| LOCATE  2.20 /IODEVICE/" ,     /* Find all devices          */
     53                 "|>" DevFileID
     54
     55            "PIPE CMS ERASE" WkFileID "| HOLE"     /* Don't need the big work   */
     56                                                    /* file anymore              */
     57            "PIPE DISK" DevFileID ,
     58                 "| MJFALD" ,                      /* Reformat input stream     */
     59                 "|>" WkFileID                     /* First, IODEVICE records   */
     60
     61            "PIPE DISK" CUFileID ,
     62                 "| MJFALC" ,                      /* Reformat input stream     */
     63                 "|>>" WkFileID                    /* Include CNTLUNITs as well */
     64
     65            "PIPE CMS ERASE" DevFileID "| HOLE"
     66
     67            "RENAME" WkFileID DevFileID           /* Input now reformatted     */
     68
     69            "PIPE DISK" DevFileID ,               /* Generate Address Details  */
     70                 "| MJFGAD" Location RealSystemName ImageName ,
     71                 "|>" WkFileID
     72            "PIPE CMS ERASE" DevFileID "| HOLE"
     73
     74            "RENAME" WkFileID DevFileID           /*                           */
     75
     76            "PIPE DISK" DevFileID ,
     77                 "| MJFPAM" CuFileID ,            /* Add pathing information   */
     78                 "|>" WkFileID
     79
     80            "PIPE CMS ERASE" DevFileID "| HOLE"
     81            "PIPE CMS ERASE" CUFileID  "| HOLE"
     82
     83            "RENAME" WkFileID DevFileID           /*                           */
     84
     85            ElapsedTime = Time(E);
     86            Say "MJFAIS002I PROCESSING ENDED." Format(ElapsedTime,4,1) "SEC. ELAPSED"
```


This Sheet Has Been Replaced

Lines 86

of compiled program in bytes 7680

messages	Informational	Warning	Error	Severe	Terminating
1	0	1	0	0	0

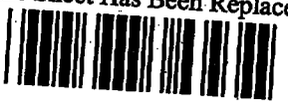
following lines have been flagged

r No.	Line
550	8

shing time of compilation: 16:58:26

This Sheet Has Been Replaced

This Sheet Has Been Replaced



APPENDIX N

pilation successful

piler Options

```
  NOALTERNATE
    CEXEC      (MJFALD   CREXX    A1)                    RECFM=F,LRECL=1024
  NOCOMPILE    (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG       (I)
  NOIEXEC
    LIBLEVEL   (*)
    LINECOUNT  (55)
    MARGINS    (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT      (MJFALD   LISTING  A1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN      (MJFALD   REXX     A1 V T V M V T V ) P0  PGD) RECFM=V,LRECL=84
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 3
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          1   /* Copyright 1998 IBM Corporation                                          */
          2   /*  Patent application pending                                            */
          3   /*                                                                        */
          4   /*  MJFALD  - a REXX filter    to control the analysis and parsing        */
          5   /*            of IOCP input Streams                                       */
          6   /*                                                                        */
          7   VariablePool:    Nop;
          8   VariablePool_End: Nop;
          9   /* */
         10   Procedure: Nop;
         11            Signal On Error
         12            Record       = ""
         13            Do  ThisRecord = 1 by 1
  1      14                'readto'  Record.ThisRecord
  1      15                Upper Record.ThisRecord
  1      16                Call BuildCanonicalDeviceRecord;
  1      17                'output' Record.ThisRecord
         18            End
         19   Error:   Nop;
         20            Select
    1    21                When RC = 12 then nop;
    1    22                Otherwise Exit RC
         23            End;
         24            Exit RC*(RC-=12)
         25   /* */
         26   BuildCanonicalDeviceRecord: Nop;
         27            Parse Var Record.ThisRecord . "IODEVICE" Specifications
         28            Specifications = Strip(Specifications,'L',' ')
         29            Select
    1    30                 When Left(Specifications,4) = "UNIT" then nop
    1    31                 Otherwise do
  1 2    32                      Parse Var Record.ThisRecord . "UNIT=" UnitType
  1 2    33                      UnitType = Left(UnitType,5)
  1 2    34                      UnitType = Strip(UnitType,'T',' ')
  1 2    35                      UnitType = Strip(UnitType,'T',',')
  1 2    36                      UnitToken = "UNIT="||UnitType||","
  1 2    37                      TokenLength = Length(UnitType) + 5  /* Plus character filler */
  1 2    38                      Parse Var Record.ThisRecord Filler "IODEVICE" Trailer
  1 2    39                      PositionOfUnitToken = Pos("UNIT",Trailer)
  1 2    40                      Remainder = DelStr(Trailer,PositionOfUnitToken,TokenLength)
  1 2    41                      UnitToken = Strip(UnitToken,'T',' ')
```

This Sheet Has Been Replaced

```
1   2    42                    Remainder = Strip(Remainder,'L',' ')
1   2    43                    DeviceRecord = Filler "IODEVICE" UnitToken||Remainder
1   2    44                    Record.ThisRecord = DeviceRecord
    2    45                    end
         46              End;
         47   BuildCanonicalDeviceRecordEnd: Return;
```

This Sheet Has Been Replaced

(Lines 47

) of compiled program in bytes 5936

il messages Informational Warning Error Severe Terminating
0 0 0 0 0 0

.shing time of compilation: 16:58:32

This Sheet Has Been Replaced

APPENDIX O

This Sheet Has Been Replaced



```
=> Compilation Summary                          MJFALI   REXX    A1
i Compiler REXX/370 3.0  LVL PQ28073   Time: 16:58:32          Date: 2000-02-03          Page:    1

npilation successful

npiler Options

  NOALTERNATE
    CEXEC     (MJFALI   CREXX    A1)                  RECFM=F,LRECL=1024
  NOCOMPILE   (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG      (I)
  NOIEXEC
    LIBLEVEL  (*)
    LINECOUNT (55)
    MARGINS   (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT     (MJFALI   LISTING  A1)                  RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN     (MJFALI   REXX     A1 V:┬ V:M V:┬ V:] P°  PGD) RECFM=V,LRECL=71
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 3
```




This Sheet Has Been Replaced

```
=> Source Listing                                    MJFALI   REXX    a1
! Compiler REXX/370 3.0  LVL PQ26073    Time: 16:58:32          Date: 2000-02-03                    Page:    2
: Do Sel Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          1   /*  Copyright 1996 IBM Corporation                                                    */
          2   /*  Patent application pending                                                        */
          3   /*                                                                                    */
          4   /*  MJFALI  - a REXX filter    to control the analysis and parsing                    */
          5   /*            of IOCP input Streams.  This stage ALIgns records.                      */
          6   /*                                                                                    */
          7   VariablePool:     Nop;
          8   VariablePool_End: Nop;
          9   /* */
         10   Procedure: Nop;
         11            Signal On Error
         12            Record       = ""
         13            Do  Forever
  1      14               'readto Record'
  1      15               Upper Record
  1      16                  Select
  1   1  17                     When Word(Record,1) = "CHPID" then do
  2   2  18                          StartingColumn = Pos('CHPID',Record)
  2   2  19                          PadLength      = 10-StartingColumn
  2   2  20                          OutRecord = Left(' ',PadLength,' ') || Record
  2   2  21                          'OUTPUT' OutRecord
  1   2  22                          end
  1   1  23                     When Word(Record,1) = "RESOURCE" then do
  2   2  24                          StartingColumn = Pos('RESOURCE',Record)
  2   2  25                          PadLength      = 10-StartingColumn
  2   2  26                          OutRecord = Left(' ',PadLength,' ') || Record
  2   2  27                          'OUTPUT' OutRecord
  1   2  28                          end
  1   1  29                     Otherwise do
  2   2  30                          'OUTPUT' Record
  1   2  31                     End;
  1      32                  End
         33            End
         34   Error:    Nop;
         35            Select
     1   36               When RC = 12 then nop;
     1   37               Otherwise Exit RC
         38            End;
         39
         40            Exit RC*(RC-=12)
         41   /* */
```

This Sheet Has Been Replaced

(X Lines 41

ze of compiled program in bytes 4632

cal messages	Informational	Warning	Error	Severe	Terminating
0	0	0	0	0	0

ıishing time of compilation: 16:58:32

This Sheet Has Been Replaced

APPENDIX P

This Sheet Has Been Replaced



```
=> Compilation Summary                    MJFGAD  REXX    A1
| Compiler REXX/370 3.0  EVL PQ28073   Time: 16:58:32        Date: 2000-02-03        Page:  1

pilation successful

piler Options

  NOALTERNATE
    CEXEC      (MJFGAD   CREXX    A1)                    RECFM=F,LRECL=1024
  NOCOMPILE    (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG       (I)
  NOIEXEC
    LIBLEVEL   (*)
    LINECOUNT  (55)
    MARGINS    (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT      (MJFGAD   LISTING  A1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINE
    SOURCE
    SYSIN      (MJFGAD   REXX     A1 T § T U T § T \ P°  PGD) RECFM=V,LRECL=122
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 3
```


This Sheet Has Been Replaced

```
Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
          1   /*                                                                                              */
          2   /*       MJFGAD - Generate Address range for Devices                                             */
          3   /*                          for BRS                                                             */
          4   /*                                                                                              */
          5   /*       (c) 1997 IBM GS and M.J. Firley                                                        */
          6   /*           Patent application pending                                                         */
          7   /*                                                                                              */
          8   /*                                                                                              */
          9   /*       Called by          : CONFIGURR                                                         */
         10   /*                                                                                              */
         11   /*  Change History:                                                                             */
         12   /*                                                                                              */
         13   /*  VER YYMMDD WHO DESCRIBE                                                                     */
         14   /*  --- ------ --- ------------------------------------------------------------                 */
         15   /*  001 981025 MJF Initial test version                                                         */
         16   /*                                                                                              */
         17   VariablePool:   Nop;
         18            DeviceCount       = 0
         19            RecordCount       = 0
         20            AddrCount         = 0
         21            MsgStem           = "MJFGAD"
         22   VariablePool_End: Nop;
         23   /*                                                                                              */
         24   Procedure: Nop;
         25           Trace Off;
         26           Signal On Error
         27           Parse Upper Arg Location RealSystemName ImageName .
         28           Record       = ""
         29           DeviceRecord = "*DEFINE  "
         30           ProcessingDeviceRecord = 0
         31           Do  Forever
1        32              'readto Record'
1        33              RecordCount = RecordCount + 1
1        34              AddrCount = 1
1        35              ThisRecord = Record
1        36
1        37              Parse Var ThisRecord . "ADDRESS=" BaseAddr
1        38              Parse Var ThisRecord . "UNIT=" UnitType "," .
1        39              Parse Var ThisRecord . "MODEL=" Model "," .
1        40              Parse Var ThisRecord . "FEATURE=" Feature "," .
1        41              Parse Var ThisRecord . "DYNAMIC=" Dynamic "," .
```

This Sheet Has Been Replaced

```
1         42          Parse Var ThisRecord . "CUNUMBR=" CUNumber " "
1         43          DeviceNickName = Left(ThisRecord,8,' ')
1         44          DeviceSerialNumber = Left('0',8,'0')
1         45          Call ProcessModelInfo;
1         46          Call ProcessFeatureInfo;
1         47          Call ProcessDynamicInfo;
1         48          Call ProcessLPARAssigned;
1         49          Call ProcessAttachedToCtlUnit;
1         50          Select                  /* Addresses coded as (Base,Range) */
1    1    51              When Left(Space(BaseAddr,0),1) = "(" then do
```

This Sheet Has Been Replaced

```
 2   2    52                    Parse Var BaseAddr "(" BaseAd "," AddrCount ")" .
 2   2    53                    Select
 2   3    54                         When Right(Space(BaseAd,0),1) = ")" then do
 3   4    55                                   AddrCount =  1
 3   4    56                                   BaseAd = Strip(BaseAd,'T',')')
 2   4    57                                   end;
 2   3    58                         Otherwise nop;
 2   2    59                    End;
 2   2    60                    BaseAddr = Right(BaseAd,4,'0')
 2   2    61                    Select
 2   3    62                         When BaseAddr = '0000' then do
 3   4    63                              Select
 3   5    64                                   When CUNumber <> "" then do
 4   6    65                                        BaseAddr = CUNumber
 3   6    66                                        end;
 3   5    67                                   Otherwise nop
 3   4    68                              end;
 2   4    69                              end;
 2   3    70                         Otherwise nop
 2   2    71                    End;
 2   2    72                    Parse Var ThisRecord . "UNIT=" UnitType ","
 1   2    73                    end    /* Addresses coded as ADDRESS=,UNITADD= */
 1   1    74               When Left(Space(BaseAddr,0),1) <> "(" then do
 2   2    75                    Parse Var BaseAddr BaseAddr "," RestOfSpec
 2   2    76                    BaseAddr = Right(BaseAddr,4,'0')
 2   2    77                    Select
 2   3    78                         When Pos("UNITADD=",ThisRecord) > 0 then do
 3   4    79                              Parse Var ThisRecord . "UNITADD=" UnitAddr "," .
 3   4    80                              BaseAddr = Right(BaseAddr,4,'0')
 3   4    81                              AddrCount= 1
 2   4    82                              end;
 2   3    83                         Otherwise do
 3   4    84                              BaseAddr = Right(BaseAddr,4,'0')
 3   4    85                              AddrCount= 1
 2   4    86                              end;
 2   2    87                    End;
 1   2    88                    end;
 1   1    89          Otherwise nop;
 1        90          End;
 1        91                    Select
 1   1    92                         When BaseAddr = '0000' then do
```

This Sheet Has Been Replaced

```
2   2    93
2   3    94
3   4    95
2   4    96
2   3    97
2   2    98
1   2    99
1   1   100
1       101
1       102
```

```
                Select
                    When CUNumber <> "0000" then do
                        BaseAddr = CUNumber
                        end;
                    Otherwise nop
                end;
                end;
            Otherwise nop
        End;
DeviceCount = DeviceCount + AddrCount
```

This Sheet Has Been Replaced

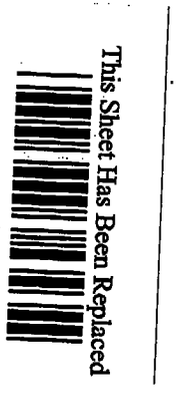
```
=> Source Listing                                  HJFGAD   REXX     A1
M Compiler REXX/370 3.0  LVL PQ28073    Time: 16:58:32                Date: 2000-02-03                Page:    4
E  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0

    1        103                Range       = AddrCount - 1
    1        104                AttachesTo  = "*CNTLUNIT=" Left(CUBaseID,10,' ')
    1        105                ImageName   = Left(Space(ImageName,0),8,' ')
    1        106                SystemName  = Left(Space(SystemName,0),8,' ')
    1        107                Location    = Left(Space(Location,0),2,' ')
    1        108                LPARAssigned= Left(LPARAssigned,144,' ')
    1        109                Feature     = Left(Feature,50,' ')
    1        110                DevSer      = Left(Space(DeviceSerialNumber,0),8,' ')
    1        111                Select
    1    1   112                     When AddrCount = 1 then do
    2    2   113                          TargetAddress = Space(BaseAddr,0)
    2    2   114                          TargetAddress = Right(TargetAddress,4,'0')
    2    2   115                          UnitType      = Left(Space(UnitType,0),8,' ')
    2    2   116                          OutRecord = "*DEFINE  " UnitType TargetAddress AttachesTo Model Feature
             C Dynamic LPARAssigned ,
    2    2   117                                   "*ALIAS=" DeviceNickName "*LOCATION=" Location "*SYSNAME=" RealS
             C ystemName ,
    2    2   118                                   "*IMAGENAME=" ImageName "*DEVSER=" DeviceSerialNumber "*RESTRICT
             C =" Left(' ',32,' ') ,
    2    2   119                                   "*CONNECT=" Left(' ',16,' ') "*LABEL=" Left(' ',8,' ')
    2    2   120                                   "*SMS=   *SG=" Left(' ',16,' ') "*VTOC=" Left(' ',32,' ') ,
    2    2   121                                   "*VVDS=" Left(' ',32,' ') "*RTECH=" Left(' ',16,' ')
    2    2   122                          EMUnit  = "*EMUNIT=" Left(' ',2,' ') ,
    2    2   123                                    "*RDHFR=" Left(' ',8,' ') ,
    2    2   124                                    "*RDTYPE=" Left(' ',5,' ') ,
    2    2   125                                    "*RDMODL=" Left(' ',5,' ') ,
    2    2   126                                    "*RDUNIT=" Left(' ',2,' ') ,
    2    2   127                                    "*RALOC=", Right(' ',10,' ')
    2    2   128                          OutRecord = OutRecord || " " || EMUnit
    2    2   129                          "OUTPUT" OutRecord
    1    2   130                     end;
    1    1   131                     Otherwise Do Index = 0 to Range
    2    2   132                          BaseAddr = Strip(BaseAddr,'L','(')
    2    2   133                          BaseAddr = Strip(BaseAddr,'T',',')
    2    2   134                          BaseAddr = Strip(BaseAddr,'T',')')
    2    2   135                          BaseAddr = Right(BaseAddr,4,'0')
    2    2   136                          TargetAddress = d2x(x2d(Space(BaseAddr,0)) + Index)
    2    2   137                          TargetAddress = Right(TargetAddress,4,'0')
    2    2   138                          UnitType      = Left(UnitType,8,' ')
    2    2   139                          OutRecord = "*DEFINE  " UnitType TargetAddress AttachesTo Model Feature
             C Dynamic LPARAssigned ,
```

This Sheet Has Been Replaced

```
2   2   140
        C ystemName ;
2   2   141
        C =" Left(' ',32,' ') ;
2   2   142
2   2   143
2   2   144
2   2   145
2   2   146
2   2   147
```

```
"*ALIAS=" DeviceNickName "*LOCATION=" Location "*SYSNAME=" ReaIS
"*IMAGENAME=" ImageName "*DEVSER=" DeviceSerialNumber "*RESTRICT
"*CONNECT=" Left(' ',16,' ') "*LABEL=" Left(' ',8,' ')
"*SMS=   *SG=" Left(' ',16,' ') "*VTOC=" Left(' ',16,' ') ,
"*INDEX=" Left(' ',16,' ') "*RTECH=" Left(' ',16,' ')
EMUnit  = "*EMUNIT= " Left(' ',2,' ') ,
          "*RDMFR= "  Left(' ',8,' ') ,
          "*RDTYPE= " Left(' ',5,' ') ,
```

This Sheet Has Been Replaced

```
 2   2   148                                    "*RDHODL= " Left(' ',5,' ') ,
 2   2   149                                    "*RDUNIT= " Left(' ',2,' ') ,
 2   2   150                                    "*RALOC= "  Right(' ',10,' ')
 2   2   151                          OutRecord = OutRecord || " " || EMUnit
 2   2   152                          "OUTPUT" OutRecord
 1   2   153                      End;
 1       154                  End
         155              End
         156  Error:      Nop;
         157              Select
     1   158                  When RC = 12 then nop;
     1   159                  Otherwise Exit RC
         160              End;
         161
         162              Say MsgStem||"001I"  Right(RecordCount,6) "DEFINITIONS WERE PROCESSED."
         163              Say MsgStem||"002I"  Right(DeviceCount,6) "UNIT SPECIFICATIONS GENERATED."
         164              Exit RC*(RC-=12)
         165  /* */
         166  /* */
         167  ProcessModelInfo: Nop;
         168              Select
     1   169                   When space(Model,0) = "" then Model = "NONE"
     1   170                   Otherwise do
 1   2   171                       Parse Var Model Model "," .
 1   2   172                       Model = Space(Model,0)
     2   173                       end
         174              End;
         175              Model   = "*MODEL= " Left(Model,6,' ')
         176  ProcessModelInfoEnd: Return;
         177
         178  ProcessFeatureInfo: Nop;
         179              Select
     1   180                   When space(Feature,0) = "" then Feature = "NONE"
     1   181                   When space(Feature,0) = "AUTOPOLL" then Feature = "AUTOPOLL"
     1   182                   Otherwise do
 1   2   183                       Parse Var Feature "(" FeatureList ")" "," .
 1   2   184                       Feature = FeatureList
     2   185                       end
         186              End;
         187              Feature = "*FEATURE=" Left(Feature,50)
         188  ProcessFeatureInfoEnd: Return;
```


This Sheet Has Been Replaced

```
   189
   190  ProcessDynamicInfo: Nop;
   191              Dynamic = Space(Dynamic,0)
   192              Select
     1 193                  When Space(Dynamic,0) = "" then do
   1 2 194                  DynamicValue = "NO "
     2 195                  end;
     1 196                When Substr(Dynamic,1,1) = "Y" then do
   1 2 197                  DynamicValue="YES"
     2 198                  end;
```

This Sheet Has Been Replaced

```
      1   199                   Otherwise do;
   1  2   200                        DynamicValue = "???"
      2   201                        end;
          202              End;
          203              Dynamic = "*DYNAMIC=" Left(DynamicValue,3,' ')
          204    ProcessDynamicInfoEnd: Return;
          205
          206    ProcessLPARAssigned: Nop;
          207              Parse Var ThisRecord . "PART=" LPARAssigned .
          208              Select
      1   209                   when Pos('PART=',ThisRecord) > 0 then do
   1  2   210                        Parse Var ThisRecord . "PART=" LPARAssigned .
      2   211                        end;
      1   212                   when Pos('PARTITION=',ThisRecord) > 0 then do
   1  2   213                        Parse Var ThisRecord . "PARTITION=" LPARAssigned .
      2   214                        end;
      1   215                   Otherwise do
   1  2   216                        LPARAssigned = ""
      2   217                        end
          218              End;
          219              LPAR = Space(LPARAssigned,0)
          220              Select
      1   221                   When Space(LPAR,0) = "" then do
   1  2   222                        LPARAssigned = "NOASSIGN"
      2   223                        end
      1   224                   Otherwise do;
   1  2   225                        Parse Var LPARAssigned LPARAssigned "," .
      2   226                        end;
          227              End;
          228              LPARAssigned = "*LPARS=" Left(LPARAssigned,144,' ')
          229    ProcessLPARAssignedEnd: Return;
          230
          231
          232    ProcessAttachedToCtlUnit: Nop;
          233                 Parse Var ThisRecord . "CUNUMBR=" CUBaseStr
          234                 Select
      1   235                      When Left(CUBaseStr,1) = "(" then do
   1  2   236                           Parse Var CUBaseStr "(" CUBaseID ")" . "," .
   1  2   237                           Parse Var CUBaseID CUBaseID "," .
      2   238                           end;
      1   239                    Otherwise do
```

This Sheet Has Been Replaced

```
1   2   240                    CUBaseID = Left(CUBase...,4)
1   2   241                    CUBaseID = Strip(CUBaseID,'T',' ')
1   2   242                    CUBaseID = Strip(CUBaseID,'T',' ')
    2   243                 end;
        244             End;
        245             CUBaseID = Right(CUBaseID,4,'0')
        246 ProcessAttachedToCtlUnitEnd: Return;
        247
```

This Sheet Has Been Replaced

(X Lines 247

ze of compiled program in bytes 17784

tal messages	Informational	Warning	Error	Severe	Terminating
0	0	0	0	0	0

ishing time of compilation: 16:58:32

This Sheet Has Been Replaced

APPENDIX Q

This Sheet Has Been Replaced



```
=> Compilation Summary.                          MJFPAM   REXX     A1
1 Compiler REXX/370 3.0  LVL PQ28073    Time: 16:58:32                Date: 2000-02-03          Page:    1

Compilation successful

Compiler Options

  NOALTERNATE
    CEXEC      (MJFPAM   CREXX    A1)                    RECFM=F,LRECL=1024
  NOCOMPILE    (S)
  NOCONDENSE
  NODLINK
  NODUMP
    FLAG       (I)
  NOIEXEC
    LIBLEVEL   (*)
    LINECOUNT  (55)
    MARGINS    (1 *)
  NOOBJECT
    OPTIMIZE
    PRINT      (MJFPAM   LISTING  A1)                    RECFM=V,LRECL=121
  NOSAA
  NOSLINK
    SOURCE
    SYSIN      (MJFPAM   REXX     A1.V § V U V § V \ P°  PGD) RECFM=V,LRECL=93
  NOTERMINAL
  NOTESTHALT
  NOTRACE
  NOXREF
  Minimum Library Level required: 3
```

371

This Sheet Has Been Replaced

```
=> Source Listing .                    MJFPAM  REXX    A1
1 Compiler REXX/370 3.0  LVL PQ28073   Time: 16:50:32          Date: 2000-02-03          Page:    2
F  Do Sel  Line C ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0
                1  /*                                                                          */
                2  /*       MJFPAM - generate PAth MAP for devices                              */
                3  /*                                                                          */
                4  /*       (c) 1997 IBM GS and M.J. Firley                                    */
                5  /*           Patent application pending                                     */
                6  /*                                                                          */
                7  /*                                                                          */
                8  /*       Called by        : MJFAIS                                          */
                9  /*                                                                          */
               10  /*  Change History:                                                         */
               11  /*                                                                          */
               12  /*  VER YYMMDD WHO DESCRIBE                                                 */
               13  /*  --- ------ --- ----------------------------------------------------    */
               14  /*  001 981025 MJF Initial test version                                     */
               15  /*                                                                          */
               16  VariablePool:    Nop;
               17            DeviceCount       = 0
               18            RecordCount       = 0
               19            AddrCount         = 0
               20            CURecord          = "00000000"
               21            MsgSten           = "MJFPAM"
               22  VariablePool_End: Nop;
               23  /* */
               24  Procedure: Nop;
               25          RunTime = Time(R)
               26          RunTime = Time(R)
               27          Signal On Error
               28          Parse Upper Arg TargetFileID
               29          Record       = ""
               30          DeviceString = "*DEFINE  "
               31          ProcessingDeviceString = 0
               32          Do  Forever
          1    33             'readto Record'
          1    34             RecordCount = RecordCount + 1
          1    35             ThisRecord = Record
          1    36             Parse Var ThisRecord . "*CNTLUNIT=" CUBaseID .
          1    37             Call FindAssociatedPathInfo CUBaseID
          1    38             PathMask = Left(PathMask,48,' ')
          1    39             LinkMask = Left(LinkMask,48,' ')
          1    40             OutputRecord = Record "*PATH="||PathMask "*LINK=" LinkMask
          1    41            "OUTPUT" OutputRecord
```

This Sheet Has Been Replaced

```
1          42              Select
1    1     43                  When RecordCount // 1000 = 0 then do
2    2     44                       Say "HJFPAM001I"  Right(RecordCount,6) "UNIT SPECIFICATIONS PROCESSED."
1    2     45                       end;
1    1     46                  Otherwise nop;
1          47              End;
           48          End
           49
           50   Error:     Nop;
           51              Select
```

This Sheet Has Been Replaced

```
      1    52                When RC = 12 then nop;
      1    53                Otherwise Exit RC
           54             End;
           55              RunTime = Time(E)
           56              Say "MJFPAM002I"  Right(RecordCount,6) "DEFINITION RECORDS PROCESSED."
           57              Say "MJFPAM003I"  Format(RunTime,6,2) "SECONDS ELAPSED."
           58   /* */
           59   FindAssociatedPathInfo: Parse Arg CUBaseID .
           60                "CALLPIPE DISK" TargetFileID  ,
           61                     "| LOCATE /CUNUMBR="||CUBaseID||"/" ,
           62                     "| VAR CURecord"
           63                Parse Upper Var CURecord . "PATH=" PathIDs .
           64                Parse Upper Var CURecord . "LINK=" LinkIDs .
           65
           66                PathIDs = Strip(PathIDs,'B',' ')
           67                PathIDs = Strip(PathIDs,'T',',')
           68                Select
      1    69                    When Left(PathIDs,1) = "(" then do
    1 2    70                         Parse Var PathIDs PathIDs ")"
    1 2    71                         PathIDs = Strip(PathIDs,'L','(')
    1 2    72                         PathIDs = Strip(PathIDs,'T',')')
      2    73                         end;
      1    74                    Otherwise nop;
           75                end;
           76                PathIDs = Translate(PathIDs,' ',',')
           77                PathMask = Left(PathIDs,50,' ')
           78
           79                LinkIDs = Strip(LinkIDs,'B',' ')
           80                LinkIDs = Strip(LinkIDs,'T',',')
           81                Select
      1    82                    When Space(LinkIDs,0) = "" then do
    1 2    83                         LinkIDs = "NONE"
      2    84                         end;
      1    85                    When Left(LinkIDs,1) = "(" then do
    1 2    86                         Parse Var LinkIDs LinkIDs ")"
    1 2    87                         LinkIDs = Strip(LinkIDs,'L','(')
    1 2    88                         LinkIDs = Strip(LinkIDs,'T',')')
      2    89                         end;
      1    90                    Otherwise nop;
           91                end;
           92                LinkIDs = Translate(LinkIDs,' ',',')
```

This Sheet Has Been Replaced

```
94    FindAssociatedPathInfoEnd: Return;
      LinkMask = Left(LinkIDs,26,' ')
```

This Sheet Has Been Replaced

XX Lines 94

ze of compiled program in bytes 8520

tal messages	Informational	Warning	Error	Severe	Terminating
0	0	0	0	0	0

nishing time of compilation: 16:58:32

This Sheet Has Been Replaced

* C L M S *

Claims

1. A method for enabling high-speed recovery and automation of computer workloads, comprising the steps:

expressing requirements for a computer system and associated networking and peripherals;

allowing a customer to specify a recovery instruction;

processing the recovery instruction at a recovery site; and

using a computer to process the recovery instruction to assign resources at the recovery site to reconfigure the computer system.

2. A method according to Claim 1, wherein the expressing step includes the step of using a resource definition language to express the requirement of the computer system.

3. A method according to claim 1, wherein the recovery instruction includes process definitions, system control program information, peripheral requirements, networking information, channel extension capabilities and tape vaulting information.

This Sheet Has Been Replaced

* C L M S *

4. A method according to Claim 1, wherein the step of using the computer includes the step of using the computer to analyze, map, reconfigure recovery hardware and begin restoration of the customer to a recovery configuration.

5. A system for enabling high-speed recovery and automation of computer workloads, comprising:

means for expressing requirements for a computer system and associated networking and peripherals;

means for allowing a customer to specify a recovery instruction;

means for processing the recovery instruction at a recovery site; and

a computer to process the recovery instruction to assign resources at the recovery site to reconfigure the computer system.

6. A system according to Claim 5, wherein the expressing means includes means for using a resource definition language to express the requirement of the computer system.

This Sheet Has Been Replaced

* C L M S *

7. A system according to claim 5, wherein the recovery instruction includes process definitions, system control program information, peripheral requirements, networking information, channel extension capabilities and tape vaulting information.

8. A system according to Claim 5, wherein the computer includes means to analyze, map, reconfigure recovery hardware and begin restoration of the customer to a recovery configuration.

9. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine for performing method steps for enabling high-speed recovery and automation of computer workloads, said method steps comprising:

expressing requirements for a computer system and associated networking and peripherals;

allowing a customer to specify a recovery instruction;

processing the recovery instruction at a recovery site; and

using a computer to process the recovery instruction to assign resources at the recovery site to reconfigure the computer system.

This Sheet Has Been Replaced

10. A program storage device according to Claim 9, wherein the expressing step includes the step of using a resource definition language to express the requirement of the computer system.

11. A program storage device according to claim 9, wherein the recovery instruction includes process definitions, system control program information, peripheral requirements, networking information, channel extension capabilities and tape vaulting information.

12. A program storage device according to Claim 9, wherein the step of using the computer includes the step of using the computer to analyze, map, reconfigure recovery hardware and begin restoration of the customer to a recovery configuration.

This Sheet Has Been Replaced