NOS 2.6.1
FEATURE NOTES

SMD131803

Table of Contents

Introduction

This document contains descriptions of most of the features that are part of the NOS 2.6.1 Level 700/688 release. The articles in this document are targeted for the site analyst, however, some of the topics may be of interest to operations or the end user so we have organized this document such that each chapter may be easily copied and distributed. The Feature Note Audience Matrix should help in distribution of the articles.

These Feature Notes were developed and written by CYBER Software Support. Questions or comments regarding them may be addressed to:

Control Data Corporation
CYBER Software Support - ARH213
4201 North Lexington Avenue
Arden Hills, MN 55126-6198
USA

800-345-9903 (USA and Canada)
612-851-4131 (International)

FEATURE NOTE AUDIENCE MATRIX

Article Title	Site/Analyst	Operations	End User
Support for STORNET Extended Memory	X		
Enhanced Mainframe Logging	X	X	
PACKER	X	X	

CHAPTER 1

Support For STORNET Extended Memory.

NOS 2.6.1 L700 has been enhanced to support the STORNET Extended Memory device as a reduced cost alternative to ESM. STORNET is the functional equivalent of ESM accessed through a low speed port.

NOS can share STORNET with NOS/VE by running it in "half ESM" mode or by specifying the starting BASE address in the VE system. If using the BASE address method, care must be taken to insure that NOS and NOS/VE are not using overlapping addresses.

1.1 EQPdeck Changes.

The EQ entry for extended memory now supports SN for specifying STORNET Extended Memory. An example of this EQPdeck entry is:

```
EQ005=DT,ET=SN/D2/NC,ST=ON,SZ=4000,CH=7.
```

This entry defines 1 million words of STORNET Extended Memory for use as EQ005. This equipment could be used as a mass storage device and/or a link device for multimainframe operations. See the NOS 2 Analysis Handbook (60459300) for examples of extended memory configurations.

(This page left intentionally blank.)

CHAPTER 2

Enhanced Mainframe Logging

This release includes a new, enhanced version of the Dedicated Fault Tolerance Processor (DFT V4). Integrated with the new version of DFT are the following modifications to the NOS 2.6.1 L700/688 operating system which accommodate and take advantage of the increased amount of information provided by DFT V4:

- Allocation and structure of the DFT buffer space in central memory is dependent upon the DFT version level.
- New BML messages and extensions to current messages expand the amount of information processed and recorded in response to a mainframe error.
- System control point messages issued in response to a mainframe error are reformatted to comply with the standard format of fault symptom code messages.
- The DSDI (Deadstart Dump Interpreter) EICB directive automatically displays the additional elements added by the DFT/OS buffer restructure.

The implementation of DFT V4 is designed in such a way that the expanded interface is not imposed on older NOS systems, ensuring compatibility with a pre-L700 NOS operating system.

2.1 DFT/OS Buffer Allocation

The DFT/OS buffer is the communications block used by DFT to pass information to the operating system. Prior to NOS 2.6.1 L700/688, this buffer area was allocated and built in central memory by NOS. Now, if DFT V4 is installed, the DFT/OS buffer is built by DFT using an expanded interface structure. When DFT builds the buffer, CIP calculates the amount of central memory to be allocated for the DFT/OS buffer based on mainframe model.

CIP V9 L700 determines at deadstart time whether DFT V3 or DFT V4 is currently installed and communicates the version level to NOS. NOS 2.6.1 L700/688 includes the software to recognize this CIP information and, based on the value, the DFT/OS buffer is built with the format corresponding to the DFT level installed. If the DFT level reported to NOS is V3, the DFT/OS buffer is built by NOS, using the old DFT/OS buffer format (i.e., no change from how it worked at previous levels of NOS). If DFT V4 is installed, DFT will build the buffer using the new DFT/OS buffer format and the amount of central memory requested by CIP. This is the method used to ensure back level support.

DFT V4 is installed as part of the CIP V9 L700 installation; the version of DFT remains unchanged if L700 NOS/VE boot modules are also installed. (New with this release, the NOS/VE boot modules are installed via the *V INSTALL VE BOOT PROGRAMS* option of the deadstart *UTILITIES* display). However, running in dual state with CIP V9 L700 and NOS/VE at L688 or older requires the matching level of NOS/VE boot modules to be installed. Doing so reinstalls DFT V3.

2.2 BML Messages Enhanced

An expanded amount of information is recorded and logged for each error reported to the operating system. This includes a timestamp of when each error actually occurred and the fault symptom code associated with the error.

Refer to the Binary Maintenance Log (BML) Message Formats manual (60459940) for additions and changes to the BML messages.

2.3 1MB Console Messages

1MB issues messages at the system console for mainframe errors. Those posted at the direction of DFT have been reformatted.

2.3.1 Reformatted Messages

The system control point messages issued by 1MB as a result of mainframe errors from DFT now have the following format:

```
hh.mm.ss ERR=Demmxxx
```

hh.mm.ss	The time the error was detected, if running DFT V4 and the time was available from the wall clock chip. The time the error was logged by 1MB if running with DFT V3 or the time was not available to DFT.
D	D is a constant indicating that the fault symptom code originated from DFT.
e	Element specifier, one of the following letters: C Processor 0. D Processor 1. I IOU 0 J IOU 1 M Memory.
mm	Model number of the specified element.

xxx Model dependent data further describing the error. This normally consists of the three-character DFT analysis code:

004 Uncorrected IOU error (NIO PP)
006 Fatal IOU error (NIO PP)
008 Fatal IOU error (CIO PP)
009 Uncorrected IOU error (CIO PP)
105 Fatal CM error
106 Partial write parity error
21E Partial write parity error (990)
701 Environment warning
702 Long power warning (battery backup)
703 Short power warning

2.3.2 Other 1MB Console Messages

The following messages have not changed, but are issued by 1MB as in previous releases. They are included here for completeness.

(204) FATAL CPU ERROR

(215) FATAL MCH ERROR

POWER/ENVIRONMENT NORMAL

The first message is issued to indicate a NOS job has aborted due to an uncorrected process error.

The second message is issued as a result of 1MB detecting a maintenance channel timeout.

The third message is posted when a power or environment warning has cleared.

2.4 DSDI Updated

The DSDI EICB (Environment Interface Communication Buffer) directive dumps the additional information provided by the restructuring of the DFT/OS buffer. No new parameters or directives are required.

(This page left intentionally blank.)

CHAPTER 3

PACKER

3.1 Introduction

PACKER performs periodic maintenance on the Indirect Access Permanent File (IAPF) chain. Holes are the spaces created within the IAPF chain when fndirect access permanent files are deleted. The process of IAPF hole fragmentation, occurring during normal IAPF activity, is countered by the periodic execution of PACKER. Adjacent holes are collapsed together, files are moved to maximize the size of holes, and lost space is reclaimed if possible. The space occupied by the enlarged holes is returned to the operating system, if possible. The holes which may remain at the end of PACKER execution are larger and more useful to IAPF operations.

The operation of PACKER never leaves a disk vulnerable to system interruptions. PACKER duplicates any file it moves and then changes the file's address in the Permanent File Catalog (PFC) with a single sector rewrite. If a machine or environment failure occurs, all IAPF files are intact.

PACKER may be run in an active system environment. User access to the device is prevented while PACKER is running by setting the PF Utility Active interlock in the Mass Storage Table. Since PACKER can be run in an active system, it is useful for

dealing with emergency situations where a device is getting full during production.

For normal periodic maintenance, PACKER can be run at a time when the production load is light and/or primarily not interactive. How frequently PACKER needs to be run is site dependent. Sites which are extremely limited on disk space may want to run PACKER much more often than sites which run PACKER only to provide dynamic space management of the IAPF chain. The dynamic space management of the IAPF chain which PACKER provides frees a site from space management via PFDUMP and PFLOAD operations.

A single call to PACKER only processes a single device, however multiple copies of PACKER may be active at the same time if each is operating on a different device. PACKER has a parameter that may be specified to have it only give the statistical messages showing the current status of the device; this will not do any actual file moving.

Turning on Sense Switch 1 stops PACKER in a controlled manner. An Operator Drop or Idle Down command also terminates PACKER in a controlled manner. A second Operator Drop or Idle Down command terminates PACKER immediately and aborts the job.

3.2 Brief Overview

PACKER scans the device's catalog tracks and builds a set of tables which allows it to operate on the IAPF chain in sequential order. As PACKER scans the IAPF chain, it performs functions depending on the items encountered.

3.2.1 Lost Space Processing

Lost space is a condition where space exists on the IAPF chain which is not pointed to by a PFC entry. If lost space is surrounded by holes, PACKER will merge the lost space with one of the holes. If the lost space is preceded by a file and not followed by a hole, PACKER will create a new PFC entry for the lost space, transforming it into a normal IAPF hole. If the lost space is smaller than the absolute minimum IAPF hole size of three PRUs, the lost space can not be reclaimed and is left as lost space.

3.2.2 Hole Processing

A collection hole is a term used to describe a hole that grows by merging itself with other holes. When PACKER encounters a hole during its scan, a sequence of events begins. Unless the collection hole is already open, the newly found hole is turned into the collection hole. The collection hole is the focal point of the major operations of combining holes and moving the files. The PFC entry pointing to the collection hole is changed into a purged Direct Access Permanent File (DAPF) PFC entry. This PFC entry is changed back into an IAPF PFC entry pointing to the hole when the collection hole is changed back into a normal hole; until that occurs, the collection hole is lost space.

PACKER uses this procedure to provide for system failures during PACKER operations. If such a failure occurs, the system is unaware of the changes that PACKER is making, all files are intact and the lost space is recovered the next time PACKER is run.

3.3 *K* Display

An informative *K* Display is available. It is provided to allow an analyst to observe PACKER's state during exceptional circumstances. Activation of the *K* Display is by simply assigning the *K* Display to the job during execution.

3.4 Detailed Documentation

PACKER is fully documented in Appendix K of the NOS 2 Analysis Handbook (60459300).