

DEPOT: DATABASE FOR ELECTRONICS PARTS AND OTHER THINGS*

C. A. Logg, P. W. Clancey, and G. Crane

*Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94309***Abstract**

DEPOT has been developed to provide tracking for the Stanford Linear Collider (SLC) control system equipment. For each piece of equipment entered in the database, a complete service, maintenance, modification, certification, location history, and, optionally, a radiation exposure history, can be maintained. To facilitate data entry accuracy, efficiency, and consistency, barcoding technology has been used extensively. DEPOT has been an important tool in improving the reliability of the microsystems controlling SLC. It is now being adopted by other systems at SLAC.

I. INTRODUCTION

It is estimated that the Stanford Linear Collider system is composed of over 100,000 individual pieces of equipment. The control system alone has numerous maintenance groups, including: timing system, beam toroids, RF, profile monitors, kicker magnets, CAMAC, microsystems, communications, beam position monitors, vacuum, collimators, and machine protection systems (MPS). Other noncontrol system maintenance groups include: power conversion, klystrons, vacuum systems, modulators, magnets, and personnel protection systems. Figure 1 illustrates the basic organization of the control system maintenance groups.

DEPOT was developed to provide tracking for the SLC Control System equipment. For each piece of equipment entered in the database, a complete service, maintenance, modification, certification, calibration, location, and radiation exposure history can be maintained. In fact, almost anything related to the equipment can be tracked.

DEPOT was a crash project. The system analysis, specification, design, and implementation were completed in the nine-month period from March 1988 through mid-December 1988. It was implemented in SPIRES (Stanford Public Information REtrieval System) [1] and PRISM [2] on an IBM 3090 running VM [3]. SPIRES is a powerful

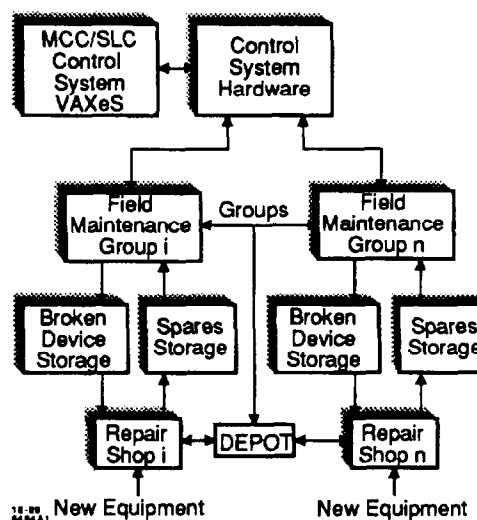


Fig. 1: Overview of SLC Control System hardware world.

fourth-generation hierarchical database and PRISM is a full-screen applications interface to SPIRES.

Initially (December 1988), the MULTIBUS micro-computer equipment used in the control of SLC was entered. Shortly thereafter (March 1989), because of high failure rates, the profile monitor system equipment was added. Since June 1989, the microsystem CAMAC crates, and their associated devices are being entered. Other control system hardware will continue to be added on a case-by-case basis.

It is impractical simply to open the database and have groups enter their equipment at whim. Each subsystem must be analyzed to determine what factors are important to track for that subsystem. In addition, conventions for the data entry must be established so that the data in the database, for a given subsystem, is analyzable.

II. OVERVIEW OF THE DEPOT PROJECT

DEPOT was initiated to provide a tool for collecting information on the equipment in SLC, and to facilitate studies on:

* Work supported by Department of Energy contracts DE-AC03-76SF00098, DE-AC03-76SF00515, and DE-AM03-76SF00010.

- *Failure Analysis:*
 - Which equipment has high failure rates?
 - Are there systematic failures which might be alleviated by redesign or modification of troublesome devices?
 - Are failures related to locations? That is, are there environmental conditions in some areas which may be causing increased failures in those areas?
- *Performance Analysis:*
 - Is there a measurable degradation in equipment performance due to aging? For example, does calibration data change with use and age, and does it indicate device performance degradation?
 - How do equipment failure rates vary with age?
- *Maintenance Analysis:*
 - How much time is spent repairing each type of device?
 - How long do devices remain in repair before they are available for spares?
 - How many devices are awaiting repair at any given time?
 - Which devices have the highest maintenance costs?
- *Quality Control:*
 - Confirm that only certified devices are placed in service.
 - Confirm that the right type of device has been placed in each location.
- *Spares stocking levels:*
 - What is the turnover in spares?
 - How many of each device should be available for spares?
 - How many are available?
- *Preventive and periodic maintenance scheduling:*
 - Provide a tool for scheduling and tracking preventive and periodic maintenance.

III. DATA COLLECTION

In order to facilitate the above, the following are necessary:

1. Each piece of equipment must be uniquely identified, including replaceable subassemblies.

2. Serial numbers, if available, must be recorded.
3. Information on the maker, model, and revision and/or variation level of each device must be kept.
4. A complete maintenance history must be maintained. It should include information on:
 - Each maintenance procedure performed.
 - Manpower required for each maintenance procedure.
 - Elapsed time a device is in maintenance.
5. A complete state and location history must be maintained. It should include information on:
 - Location and time in use.
 - Amount of time spent on spares shelf.
 - Amount of time unavailable for use.
6. Efficient, accurate, and consistent data collection and data entry must be provided for.

IV. DEPOT IMPLEMENTATION OVERVIEW

The responsibility for making DEPOT meet its requirements lies in three areas:

- Judicious database design and implementation which allow for both:
 - Common identification and configuration information to be automatically provided when needed.
 - Flexible fields in the records, which can be adapted for various types of equipment and "other things."
- Data collection and data entry tools and techniques need to be identified. These aid in data entry consistency and ease.
- A thorough analysis of any equipment which is going to be DEPOT'd is needed. This will provide for data entry ease, data consistency, and analyzability.

A. Database Design and Implementation

DEPOT is composed of several databases: the Device database, the Inventory database, the Maker database, and the Table database. The Table and Maker databases are used internally by DEPOT.

1) *The Device Database:* The Device database contains templates for each type of equipment (aka, device) entered into the Inventory database. The device record for a piece of equipment includes:

- *NICKNAME*: This is a shorthand for the model/maker/revision. It can be a name that is commonly used for a type of device, or it may be a concatenation of the actual model/maker/revision. It should be chosen with care, after the equipment analysis, to make DEPOT effective for the intended application.
- *MAKER*: Maker (manufacturer).
- *MODEL*: Model.
- *REV-VAR*: Revision and/or variation level.
- *CLASS*: This is a classification field (e.g., CAMAC, FASTBUS, NIM, VME, MULTIBUS, et al.).
- *GENERIC*: Generic further breaks down the class (e.g., CRATE, BLOWER, POWER SUPPLY, CPU, CAMERA, CHASSIS, ADC, DAC, et al.).
- *DESCRIPTION*: This is a comment field.
- *FULLNAME*: Full name of the device.

• *Optionally*:

- Subassembly and component information.
- Firmware information.
- Procedures such as:
 - Calibration.
 - Modification.
 - Certification.

Note that for any device, this information only has to be entered ONCE. When Inventory records are later created, this information is automatically linked to the Inventory record.

2) *The Inventory Database*: The Inventory database contains the actual equipment inventory records. To create an entry for a piece of equipment, one simply provides:

- ID.
- Serial number.
- Nickname of the appropriate device template.
- The Maintenance Group (which is used for protecting different groups of users from each other).

An Inventory record contains the following fields (and actually many others that aren't detailed here), which can be used to record just about any information desired. Note that the fields prefixed with a "*D-" are picked up from the Device record.

a) *Device identification*: This includes:

- *ID*: A unique identifier for the piece of equipment. It should be barcoded and permanently affixed to the piece of equipment.
- *NICKNAME*.
- *SERIAL*.
- *MG*: This is for the Maintenance Group. The different groups using DEPOT are protected from each other by this field.
- **D-MAKER*.
- **D-MODEL*.
- **D-REV-VAR*.
- *COMMENT*: This is a comment field for the individual piece of equipment. It is distinct from the Description field in the Device record.
- **D-CLASS*.
- **D-GENERIC*.
- *DRAWING*: This field is filled in by choosing the appropriate drawing package from the Device record, if there are multiple choices. If there is only one option, it is automatically chosen.

b) *State and location*: This is where the state and location history information is kept. Each record contains the following fields:

- *STATE*: Currently these are: MAINT, SPARE, SALVAGE, IN USE, OUT and UNKNOWN.
- *LOCATION*: This is an actual physical location or a logical location. A logical location is a subsystem name for a physical location. The logical location is sometimes used when a device is put into a state of IN USE or OUT. It is the name of a part of a subsystem, rather than an actual physical location name.
- *LOC.ID*: In the case of a device located in a CAMAC crate, this is the ID of the crate and the slot number. This allows a user to do a crate profile (list the devices in each slot) for a crate.
- *L.DATE*: This is the date that the device went into this state and location.

Note that there is not an "in" and "out" date associated with each state. L.DATE is the date that the device is put into the respective state. The out date is assumed to be the date of the next state and location entry, EXCEPT in the case of "IN USE," where a specific entry of state "OUT" can be made.

c) *Maintenance history:* This is where the maintenance history information is kept. Each maintenance history record has the following information:

- *MAINT.TYPE:* This is the kind of maintenance being performed. Values include: CALIB, CERT, CONFIG, INSPECT, MOD, PM, RAD, and REPAIR.
- *PROBLEM:* This is the reason the unit was sent to maintenance.
- *SYMPTOM:* In the case of a maintenance type of REPAIR, this is the symptom that the technician saw when he examined the device. In the case of other maintenance types, this field is used to record the identification of the procedure performed or other information.
- *FIX:* In the case of a maintenance type of repair, this is what was done to repair the device. In other maintenance types, this field is used to record other information (e.g., radiation dosage, calibration data).
- *PARTS:* This is to record the parts used in a repair.
- *TECH:* This is the name of the technician performing the repair.
- *M.DATE.START* and *M.DATE.DONE:* These are for recording the time the maintenance process started and finished.
- *FTE.HOURS:* This is for recording the manpower hours required for the maintenance operation.

d) *Procedures:* These fields are for storing information on procedures applicable to a device. Procedures which are entered in the Device record, are copied over to the Inventory record when the Inventory record for a piece of equipment is created. They may be subsequently modified, deleted, or added to in the Inventory record.

- *PROC.TYPE:* This is the procedure type. For example, it may be PM (preventive maintenance), INSPECT, CERT (certification), CALIB (calibration), MOD (modification), or CONFIG (configuration).
- *PROC.DAYS:* This is the interval in which the procedure is to be performed. It may be 0, which indicates it is not a periodic procedure. It may be a negative number which can be used for special flagging purposes. Or, it can be positive, in which case it indicates the procedure is a periodic procedure which is to be performed at the specified interval.
- *PROC.NAME:* This field is for the formal procedure name.

- *PROC.DESC:* This field is for a description or other comments relating to the procedure.
- *PROC.REF:* This is for a reference for the procedure.

In addition to all of the above, there are several other fields used for accounting and auditing purposes with which the casual user need not be concerned.

B. Data Entry

The above database design provides for a minimum of duplicate information input. However, data still must be entered into the database. Techniques used to facilitate data entry include:

- *Barcoded IDs:* Each device to be entered into DEPOT has a unique barcoded ID affixed to it.
- *Barcode scanners:* Data entry terminals are equipped with barcode scanners.
- *Barcode command menus:* A sequence of commands to DEPOT is usually required to perform any data entry or data access task. Barcoded command lists are often printed up. The user then simply scans the barcoded commands to perform the command sequence. This speeds up the data entry and data access process.
- *Encoding procedure identifiers into barcode:* Maintenance procedures often have written instructions on forms which need to be followed and filled in when the procedures are performed. To provide for efficient, accurate and consistent data entry, the identification of the form, and sometimes DEPOT command sequences are barcoded right onto the forms.
- *Portable data collectors with barcode scanners:* These facilitate in the field data collection. The data is sent to VM, processed, and loaded into DEPOT.

C. Equipment Analysis

As was stated earlier, there are three aspects to making DEPOT perform: the database design and implementation, the data entry, and the equipment analysis. The database design and implementation are virtually complete. Several data entry techniques have been listed, and these will continue to evolve with technology and the ingenuity of the users. The equipment analysis is where the responsibility for making DEPOT perform now lies. Before the equipment of a subsystem is entered into DEPOT, it should be thoroughly analyzed. The results of this thorough analysis will provide for ease of data entry, data consistency, data completeness, and analyzability. In the equipment analysis:

V. DEPOT REPORTS

1. A list of all the maker, models, and revision levels must be made.
2. A decision must be made on how to classify the equipment. Appropriate classification will facilitate database searching and analysis. There are two fields for classification: Class and Generic. Class is the more general classification. Some of the current Classes are: FASTBUS, CAMAC, VME, MULTI-BUS, oscilloscopes, power supplies, and PC. Generic is more specific. Some of the CAMAC Generics are: SAM, DAC, ADC, PPOM, IDOM, IDIM, CRATE, and BLOWER. Some PC generics might be CPU, monitor, serial port, and memory.
3. A unique nickname must be defined for each (model, maker, revision) triplet. This may often be just the model number and the maker. It should be chosen with consideration given to how it will be used.
4. A list must be made of which maintenance groups will be maintaining which Inventory entries in DEPOT.
5. Locations for the equipment must be defined and uniquely identified. See Ref. [4] for an example.
6. If there are any maintenance procedures to be tracked, they should be detailed and uniquely identified.
7. If a detailed maintenance history is to be maintained for some equipment, its common problems, symptoms, and fixes should be detailed. The problems, symptoms, and fixes must be entered consistently (with consistent phraseology and syntax) to be analyzable.
8. The kind of data that needs to be entered for equipment needs to be detailed. Examples here include, radiation exposure, calibration data, location data, maintenance data, etc.
9. A list of where data related to the equipment originates must be generated. This information is used to define the data collection and data entry mechanism.
10. A description of how the data in the database is going to be used, and what reports are going to be needed, should be made. It is important to assure that all the data needed for these reports is going to be entered in DEPOT.

The importance of the equipment analysis phase can not be emphasized enough. The usefulness of DEPOT from now on is almost completely dependent on the organization, consistency, and quality of the data entered into the system.

The PRISM interface to SPIRES provides an easy to use facility for report generation and formatting, when the report generation simply involves the extraction and displaying of data. Figure 2, a location history, and Fig. 3, a maintenance history report, were generated by the PRISM reporting facilities. More complex reports [5], requiring linkage of records and analysis of the data, can be generated by a knowledgeable programmer utilizing the functionality of the SPIRES system. Complex reports include: crate profiles (Fig. 4), tree displays (Fig. 5), and parts replacement report (Fig. 6).

A. Location History Report

Oct. 16, 1989 Full Location History Report			
Nickname/ID	Date In	State	Location
COMPUTROL-2-V2 16010593	03/01/89	MAINT	B034-DMG
	03/01/89	SPARE	B034-DMG
	03/01/89	IN USE	DR08
	09/25/89	MAINT	B034-DMG
	09/27/89	SPARE	B034-DMG
FSK-MODEM 16010721	10/02/89	IN USE	LI17
	03/09/89	MAINT	B034-DMG
	03/15/89	SPARE	B034-DMG
	04/19/89	IN USE	MC00
	06/07/89	MAINT	B034-DMG
	07/06/89	IN USE	LI04
	09/08/89	MAINT	B034-DMG
	09/19/89	SPARE	B034-DMG
MBCD-5-V1 16010312	10/02/89	IN USE	LI17
	02/14/89	MAINT	B034-DMG
	03/28/89	SPARE	B034-DMG
	03/30/89	IN USE	LI28
	09/19/89	MAINT	B034-DMG
	09/20/89	SPARE	B034-DMG
	10/02/89	IN USE	LI17

Fig. 2: State and Location History.

This report (Fig. 2) was generated by using the PRISM reporting facilities. It shows the state and location history for the three devices indicated.

B. Maintenance History Report

Oct. 16, 1989		Maintenance History Report				
Start Date	Done Date	Hrs	Maint Type	Symptom	Fix	Parts
<u>ID = 16008228</u>						
09/01/88	09/01/88	0.25	REPAIR	"ERRORS"	NTF	
10/20/88	10/20/88	1.00	REPAIR	FAILS SEGMENT ADDRESS TESTS	REPLACE 5M & CLEAN MULTIBUS CONNECTOR	74LS374@5M
11/15/88	11/15/88	0.50	CERT	BOARD CERTIFICATION: NONREPEATER	RHS	
12/14/88	02/08/89	1.00	REPAIR	N.T.F.	REK	
02/24/89	02/24/89	0.50	CERT	QC-123-604-01-R04	EGH	
03/23/89	03/23/89	0.50	MOD	FM-123-604-03-R00	SXN	
03/28/89	03/28/89	0.50	CERT	QC-123-604-01-R06	EGH	
04/10/89	04/17/89	0.50	REPAIR	5K NEEDS REPLACE	REPLACED	D8287@5K
04/18/89	04/18/89	0.50	CERT	QC-123-604-01-R06	EGH	
08/15/89	08/15/89	1.00	REPAIR	TDVRTO ERRORS	REPLACED BAD CHIP	8289@5B
09/08/89	09/08/89	0.26	CERT	QC-123-604-01-R07	EGH	
<u>ID = 16008232</u>						
07/22/88	07/22/88	0.25	REPAIR	"BAD"	REPLACE 5V; OPTIMIZE DATA TIMING OF 2V	74LS123@5V
07/22/88	07/22/88	0.50	CERT	BOARD CERTIFICATION	JXK	
08/18/88	08/18/88	0.50	REPAIR	"TRANSMISSION ERRORS"	DO "F" MOD	
09/09/88	09/09/88	0.25	REPAIR	FAILS NET TEST	REPLACE 5B	8289@5B
02/28/89	03/01/89	0.50	REPAIR	REPLACE 5B	REPLACE 5B WITH	8289@5B
03/02/89	03/02/89	0.50	CERT	QC-123-604-01-R04	EGH	
03/22/89	03/22/89	0.50	MOD	FM-123-604-03-R00	JPP	
03/23/89	03/23/89	0.50	CERT	QC-123-604-01-R06	EGH	
08/10/89	08/10/89	2.00	REPAIR	WON'T LOAD PIOP IMAGE	REPAIRED OPEN TRACE BETWEEN 4U-6 AND 4N-8	
09/01/89	09/01/89	0.25	CERT	QC-123-604-01-R07	EGH	
09/14/89	09/14/89	0.25	CERT	QC-123-604-01-R08	EGH	

Fig. 3: Maintenance History Report.

This Maintenance History Report (Fig. 3) was generated by the PRISM report facilities. It shows the maintenance history for two different devices.

Crate Profile for LI12/CR03 (KF12-5A31).			
Slot	Profile Device	Inventory Device	Device ID
S01	CVII*	CVII	14195
S02	PIOP-*	PIOP-D	14592
S03	PIOP-*	PIOP-D	17175
S04	PIOP-*	PIOP-D	14594
S05			
S06			
S07			
S08			
S09			
S10			
S11			
S12			
S13	SAM-E*	SAM-E (SLAC)	16083
S14	TCM-*	TCM-C	15523
S15	==(BPM IS 2-WIDE)		
S16	BPM605*	BPM605-A	16011417
S17	==(BPM IS 2-WIDE)		
S18	BPM972*	BPM972-A	16002489
S19	==(BPM IS 2-WIDE)		
S20	BPM972*	BPM972	16002505
S21	STBII	STBII	16006239
S22	==(PDUH IS 2-WIDE)		
S23	PDUH-*	PDUH-C	16001575
S24	==(SCC IS 2-WIDE)		
S25	SCC MOD*	SCC MOD 2	16967

Fig. 4: A crate profile.

Nickname	Device ID	Loc ID	Date
MBCRATE-4	16011323	(LI17)	10/02/89
COMPU-			
TROL-V1	16007626	16011323-S01	10/02/89
..COMPUTROL			
INTF-1	16010098	16007626-U77	10/02/89
.FSK-MODEM	16010603	16011323-S03	10/02/89
.MBCD-5-V1	16007647	16011323-S06	10/02/89
.1/2 MB			
RAM MBUS	16010029	16011323-S12	10/02/89
.86/30-8MH	16010722	16011323-S14	10/02/89
..SBC 337A	16010723	16010722-U48	10/02/89
..SBC 304	16010724	16010722-U72	10/02/89
.PRIM-5-V1	16008161	16011323-S16	10/02/89

Fig. 5: Tree display.

C. Crate Profile Display

The crate profile (Fig. 4) is an example of a more complex report. It originally was developed to facilitate data validation. It also has turned out to be a useful reference tool for the maintenance people. This report was generated by the commands:

```
FIND CURLOC LI12/CR03
DISPLAY CRATE
```

The CRATE parameter on the DISPLAY command invokes the code written by the programmer to create the display.

D. Tree Display

The tree display (Fig. 5) shows all the devices and their subassemblies for a particular parent device (in this case, a MULTIBUS crate). Note that LOC ID is the ID of the parent device. This report was generated by the commands:

```
FIND CURLOC LI17
DISPLAY TREE
```

The TREE parameter on the DISPLAY command invokes the code written by the programmer to create the display.

E. Parts Used Report

Parts Report for PSC* (Dates:)					
Total PSC* in DEPOT: 176					
Total number repaired: 115					
Total repairs: 361					
U57	=	42	U68	=	2
U54	=	22	VR1	=	2
U23	=	12	CR2	=	1
U46	=	9	CR4	=	1
U52	=	9	U11	=	1
U49	=	6	U15	=	1
U69	=	6	U16	=	1
U72	=	6	U20	=	1
U39	=	4	U21	=	1
U41	=	4	U3	=	1
U64	=	4	U32	=	1
U70	=	4	U39K	=	1
U9	=	4	U42	=	1
U36	=	3	U43	=	1
U44	=	3	U45	=	1
U51	=	3	U53	=	1
U60	=	3	U55	=	1
U61	=	3	U59	=	1
VR2	=	3	U63	=	1
U33	=	2	U65	=	1
U48	=	2	U66	=	1
U58	=	2	U69	=	1
U56	=	2	U70	=	1
U6	=	2	U73	=	1
U62	=	2	U75	=	1
U67	=	2	U8	=	1

Fig. 6: Parts Replacement Report.

This report (Fig. 6) shows the parts which were replaced during maintenance of the PSC modules. They are ordered from the most frequently changed parts to the less frequently changed parts. This report was generated independently of PRISM by a REXX [6] Interpreter program which calls SPIRES.

VI. DEPOT EXPERIENCE

DEPOT has been a useful tool in improving the reliability of the microsystems controlling SLC. It has made it possible to effectively track the many engineering modifications and upgrades, to both modules and systems, that

were required to improve the reliability of the microsystems. It is now being used to also track failures.

There are currently over 20,000 inventory entries into DEPOT:

- SLC controls Department—about 6500 entries.
- Power Conversion Department—about 4300 entries.
- SLD—about 2300 entries.
- Laboratory Electronics Pool—about 7400 entries.

Many more entries are being added every day, and other departments are inquiring about how DEPOT might be useful to them.

In using DEPOT interactively, the number of inventory entries does not have an effect on the response time of the system for individual records. The searching and sorting response times are primarily determined by the size of the result, and is also not affected by the number of inventory entries. The time to save an updated inventory record is also not affected by the number of inventory entries, but reflects the number of fields that are indexed. The detailed indexing facilitates the short searching response times.

VII. CONCLUSIONS

The DEPOT system itself provides a great number of options for data storage with respect to equipment tracking. The PRISM interface together with SPIRES' functionality, provide powerful tools for data searching, extracting, sorting, and reporting. However, these tools are only as good as the data entered into the system. At this point, and in the future, the primary responsibility for DEPOT's usefulness will lie in the hands of the people doing the analysis of the equipment to be entered into DEPOT, and the people doing the data collection and entry.

ACKNOWLEDGMENTS

Numerous people at SLAC contributed ideas that went into the analysis and specification of this system. In particular, I would like to acknowledge J. Kieffer and M. Ross for their input. My co-authors P. Clancey and G. Crane are due full credit for their excellent programming job. Credit is due L. Cottrell for his insight into the management of the DEPOT implementation. L. Addis deserves gracious acknowledgements for the efforts that she put into coordinating the barcode scanner evaluation. Finally, I thank J. Brown, R. Humphrey, and M. Ross for their encouragement and support.

REFERENCES

- [1] SPIRES (Stanford Public Information REtrieval System), developed and marketed by Stanford University.
- [2] PRISM is a SPIRES application development system developed and marketed by Stanford University.
- [3] IBM, Virtual Machine/System Product General Information.
- [4] C. Logg, SLC/SLAC Location Specification Proposal, ADDOC #1, Controls Department, SLAC.
- [5] P. Clancey and C. Logg, DEPOT Database Reports and Utilities, ADDOC #9, Controls Department, SLAC.
- [6] IBM, Virtual Machine/System Product, System Product Interpreter Reference, Release 5, SC24-5239-2.