TO:          Distribution
FROM:        Joan Archer
DATE:        30 May 75
RE:          Multics Change Requests

Enclosed are copies of Multics Change Requests which were approved from 1 May 75 to 15 May 75.
SUMMARY:

Change the ipc_event call mechanism to work properly with internal procedures.

REASONS:

Several programs set up event call channels to internal procedures. When these procedures get invoked, they receive invalid display pointers. The only reason these procedures work today is that they do not reference non-local automatic variables or parameters. If MCR 1098 were to be implemented, these programs would stop working. New users of ipc_ could easily fail by referencing non-local variables.

IMPLICATIONS:

The user community must be informed of the interface change to ipc_$decl_ev_call_chn well before implementation of MCR 1098.

DETAILED PROPOSAL:

Work should proceed in 3 stages:

1) The entry ipc_$dcl_event_call_channel must be added, which receives an entry value as its second argument.
instead of a pointer. This entry value must be stored in the event call structure and used in calling the procedure at wakeup time. This change will have no ill effect on the system. Programs that work now will continue to work.

2) All system callers to ipc_$decl_ev_call_chn must be changed to use ipc_$dcl_event_call_channel. This involves about thirty procedures.

3) After allowing time for users to convert to the new interface, implement MCR 1098 and remove ipc_$decl_ev_call_chn.
**Entry: ipc_$create_ev_chn**

This entry creates an event-wait channel in the current ring.

**Usage**

```plaintext
declare ipc_$create_ev_chn entry (fixed bin(71),
fixed bin(35));

call ipc_$create_ev_chn (channel_id, code);
```

1) `channel_id` is the identifier of the event channel. (Output)
2) `code` is a standard status code; see Status Code Values below. (Output)

**Entry: ipc_$delete_ev_chn**

This entry destroys an event channel previously created by the process.

**Usage**

```plaintext
declare ipc_$delete_ev_chn entry (fixed bin(71),
fixed bin(35));

call ipc_$delete_ev_chn (channel_id, code);
```

1) `channel_id` is as above. (Input)
2) `code` is as above. (Output)

**Entry: ipc_$dcl_event_call_channel**

This entry changes an event-wait channel into an event-call channel.

**Usage**

```plaintext
declare ipc_$dcl_event_call_channel entry (fixed bin(71), entry,
ptr, fixed bin, fixed bin(35));

call ipc_$dcl_event_call_channel (channel_id, procedure,
data_ptr, priority, code);
```

1) `channel_id` is as above. (Input)
2) procedure

Is a pointer to a procedure entry point to be invoked when an event occurs on the specified channel. (Input)

3) data_ptr

Is a pointer to data to be passed to and interpreted by that procedure entry point. (Input)

4) priority

Is a number indicating the priority of this event-call channel as compared to other event-call channels declared by this process for this ring. If, upon interrogating all the appropriate event-call channels, more than one is found to have received an event, the lowest-numbered priority will be honored first, and so on. (Input)

5) code

Is as above. (Output)

**Entry:** ipc$_decl_ev_walt_chn

This entry changes an event-call channel into an event-wait channel.

**Usage**

```
declare ipc$_decl_ev_walt_chn entry (fixed bin(71),
   fixed bin(35));

call ipc$_decl_ev_walt_chn (channel_id, code);
```

1) channel_id is as above. (Input)

2) code is as above. (Output)

**Entry:** ipc$_drain_chn

This entry resets an event channel so that any pending events (i.e., events which have been received for that channel) are removed.

**Usage**

```
declare ipc$_drain_chn entry (fixed bin(71), fixed bin(35));

call ipc$_drain_chn (channel_id, code);
```

1) channel_id is as above. (Input)
TITLE: Remove backup time setting entries from hcs

AUTHOR: A. Kobziar

SUMMARY:
Change the reloader to call hphcs_$set_backup_times and delete the corresponding hcs_entry. Delete hcs_$set_backup_dump_time as the dumper already calls the hphcs version. A warning for the hcs_entries was issued last fall (with MCR 693).

REASONS:
The removal of user settable dates creates a problem whenever links are reloaded under the -debug option because the reloader does not treat the date-time information of each link separately, but rather collectively by using the time in the parent directory. If this directory is created by the reload, then its date-time will be later than the time stored in the link (dir list) record on the dump tape and the normal action is to skip the processing of this record. Adding code for individual link processing would slow down the reloader and not work correctly on multiple incremental tapes; this choice was opted in favor of always reloading links in the -debug case. Also the default setting of the -trim option (which deletes everything in an existing directory that is not in a dir_list record) will be changed from on to off if
the -debug option is used. This will prevent deletion of new segments/directories in an existing new directory of the same name.

IMPLICATIONS:

A user reload (-debug option) will always replace existing links and segments that are named the same as those found on the tape. Also, trimming (deleting all segments and links in an existing directory that are not on the tape) will not be the default. If trimming is desired, then -trim must follow the -debug option: reload -debug -trim.
TITLE: Separate privileges from the access authorization field

AUTHOR: A. Kobziar

Status: A

Expires: 11/3/75

Written: 05/05/75

DOCUMENTATION CHANGES


355  BOS  Salvager  MPM (Vol, Sect.)

Ring Zero  PIM (AN #) 75

Ring One  SysDaemon/Admin.

RunTime  MPAM (Sect.)

User Cmd/Subr.

MSSM (Sect.)

Objections/Comments:

Info Segs

Other (Name)

None (Reason)

Use these headings: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional)

SUMMARY: Add two new functions, get_max_authorization_ and get_privileges_, which along with the existing function get_authorization_, will allow ring 1 circumvention of the SPS entry hcs_$get_authorization.

REASONS: hcs_$get_authorization returns the privilege bits embedded in an unused part of the authorization field. Ring 1 procedures currently store this authorization as an access class. Although the bits used for privileges are ignored in accessibility calculations, it is an error to store these bits, as privileges are dynamic, and a salvager can assume that all bits outside of those defined for the access class should be zero—if any are on then the access class could have been clobbered.

DETAILED PROPOSAL: The new functions will join get_authorization_ in the procedure get_process_id_. Once obtained from ring 0 the information will be stored in internal static with all undefined bits zeroed, except for the get_privileges_ entry which will always call ring 0, as privileges are dynamic.

declare get_max_authorization_ entry returns(bit(72)aligned);

declare get_privileges_ entry returns(bit(36)aligned);
SUMMARY: Install a new program, syserr_log_man, in bound_user_control, to manage permanent syserr log segments in ring 4.

REASONS: Increased use of the syserr log (particularly for protection auditing) requires a mechanism to keep a permanent on-line copy of the syserr log. Rather than fill up *system_control_dir* with possibly many permanent syserr log segments, the mechanism below is proposed. Also, the marketing requirement for protection auditing requires a mechanism of this type.

IMPLICATIONS: Logs will be searched and addressed efficiently. The decision on when a part of the log is obsolete and can be deleted is removed from ring 0. Log summary and analysis tools will be able to refer to a full history of syserr messages, enhancing statistical analyses of system operation of various kinds.

DETAILED PROPOSAL:

Permanent syserr log structure

Permanent syserr log segments will be kept in *system_control_dir/syserr_log_dir*. Each time the syserr log is copied from ring 0 (see MTR-103 and MCH-320), a new log segment will be created, named with the date and time of creation, and containing only those syserr messages not previously copied, rewritten to minimize storage.
A log addressing segment will be maintained containing

- the date and time range of messages
- the sequence number range of messages
- the date and time this log segment was deleted (to help in searching backup maps for old log segments)

This addressing segment allows quick and convenient searching of log segments by log analysis tools (print_syserr_log and daily_syserr_process), since one need not initiate log segments that do not contain relevant messages. It also makes possible automatic round-robin management of the permanent log segments.

**Access to permanent syserr logs**

To initialize the automatic copying of syserr logs from ring 0 (MCr-820), the directory \( \text{system_control_dir}\backslash \text{syserr_log_dir} \) must be created manually, and its ACL and ring 4 segment initial ACL set. ACLs should be set as follows. User ID's that only need to read the log should have "s" on syserr_log_dir, and "r" on its ring 4 segment initial ACL. This results in "r" access to both the log addressing segment and the log segments themselves. User ID's that need to be able to copy the syserr log from ring 0 (like the Initializer, or any other process designated to perform automatic log copying) should have "sma" to syserr_log_dir and "rw" on its ring 4 segment initial ACL. This results in "r" access to log segments (because, once the log is copied, the segment should not be modified), and "rw" on the log addressing segment.

The safety switch is set on for all log segments and the log addressing segment, to prevent accidents.

If, during an attempt to copy the syserr log into ring 4, the log addressing segment is missing or damaged, automatic copying of the syserr log from ring 0 is disabled, with a sys_log message on the Initializer terminal. At sites where the loss of syserr messages is not tolerable, the operator can shut the system down before the ring 0 syserr log wraps around over previously uncopied messages.

**syserr_log_man entry points:**

- **as_auto_copy_init**
  - **auto_copy_init**
    - set up this process as the handler for the log copying wakeup from syserr_logger (see MCr-820).

- **as_auto_copy_log**
  - **auto_copy_log**
    - copy the syserr log in response to a log copying wakeup.

- **manual_copy_log**
  - copy the syserr log manually.
All entry points require hphcs_access. If called without hphcs_access, they do no damage. The "as_" prefixes indicate special entry points for the answering service, which use sys_log to log errors instead of com_err.
**SUMMARY:**

Install improved version of the teco editor, which uses EIS coding improvements, uses iox_ instead of ios_, allows editing of segments >64k, and fixes several bugs. The coding will also be brought up to system programming standards.

**IMPLICATIONS:**

1. The procedure search_file in boundQedx_ will be called by teco to evaluate regular expressions. This will involve a change to the bind file for boundQedx_. (This change is covered by another MCR.)

2. The bind file for bound_teco_ will be changed to remove the following names from the segment (the entry points will be retained by the binder): TECO, list_temporary_segments, list_temp_segs.
3. The teco search rules will be changed from (in order searched):
   a) working dir
   b) home dir
   c) >system_library_tools
   d) >system_library_auth_maint

to:
   a) working dir
   b) home dir
   c) containing dir for bound_teco_

Therefore the teco macro library must be in the same directory as bound_teco_.

4. The teco macro library will be brought up to system programming standards. Obsolete or special purpose macros will be eliminated. Embedded pathnames will be eliminated.

5. The teco$abort implementation has been changed. A user can no longer simulate teco$abort with a call to signal_.

DETAILED PROPOSAL:

1. teco is now coded in Version 2 PL/I.

2. teco will allow any segment up to sys_info$max_seg_size words in length. The editing buffer size reflects the actual text length and certain editing sequences will result in added overhead.

3. A new teco command, n, is being added to allow a qedx style regular expression search.

4. Red and black shift characters are not included in error messages.
TITLE: New Reload Options during System Startup

AUTHOR: D. R. Vinograd

SUMMARY:

Replace the "update" command in system_startup by two new commands:

reload_notrim

and

reload_system_release

The first will call reload with the -notrim control argument and the second will call a new entry point reload$system_release.

REASONS:

The reload_notrim command is necessary to the maintenance of the development hierarchy. The reload_system_release command is required for system distribution to enable backup to replicate completely the information on a system release tape onto an existant hierarchy.

IMPLICATIONS:

Sites must be advised of the new options and when to use them. The present reload key word will
operate in the same way it does today. The invocation of the reload_system_release key word will cause the deletion of any segment or directory in the user's hierarchy that does not conform with the hierarchy description on the release tape.

PROPOSAL:

Two new backup control flags will be created to control directory name trimming and the suppression of "dtm" checking. These flags will only be set by the entry reload$system_release and will not be control arguments.
TITLE: Fix bug in mst generator

AUTHOR: E. Stone

- Coded in [ ] PL/I [ ] ALM [ ] other
- Explain in DETAILED PROPOSAL [ ]
- Planned for System MR [ ]
- Fixes Bug Number(s) [ ]
- Documented in MT8 [ ]
- User/Operations-visible [ ]
- Interface change? [ ] yes [ ] no
- Incompatible change? [ ] yes [ ] no
- Performance: [ ] Better [ ] Same [ ]
- Replaces MCR [ ]

CATEGORY (Check One):
- Lib. Maint. Tools [ ]
- Sys. Anal. Tools [ ]
- Sys. Prog. Tools [ ]
- 355 [ ]
- Document [ ]
- Specify One or More [ ]

Objections/Comments:

Use these headings: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional)

SUMMARY:

Change the mst generator to clear the extended access field of ACL entries.

Convert the mst generator to use iox_ and hence to use the tape_mult_, vfile_, and discard_ io modules.

REASONS:

The extended access field of segments whose ACL is specified on the header currently is set to blanks, e.g. 040040040040.
**TITLE:** set_tty command

**AUTHOR:** R. Coren

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CODED IN</td>
<td>X</td>
<td>FL</td>
<td>I</td>
<td>ALM</td>
</tr>
</tbody>
</table>

**COMMENT:**

- Planned for System MR
- Fixes Bug Number(s)
- Documented in MTB: 355

**User/Operations-visible**

- Interface change? [X] yes
- Incompatible change? [X] yes
- Performance: [X] Better
- Replaces MCR

**DATE**

- Written: 04/25/75
- Status: A 11/06/75
- Expires: 11/06/75

**SUMMARY:**

Provide a user command, set_tty, to allow a user to specify his terminal type for the purposes of code conversion, and to set the modes for I/O on his terminal. This replaces the iomode command.

**REASONS:**

Failure (for whatever reason) to read a terminal's answerback can cause a terminal to behave in an undesirable fashion; the current methods of redress open to the user are both cumbersome and incomplete. The proposed command is flexible enough to cover most situations; in general, the -type control argument by itself is sufficient to do what the user wants.

**IMPLICATIONS:** None

**DETAILED PROPOSAL:**

See attached draft MPM documentation.

**NOTE:** The command uses the Answering Service's "Initial Modes Table", and will recognize the possibility of setting a Selecterm's tabs as soon as the Answering Service does.
set_tty

**NAME:** set_tty, stty

The `set_tty` command may be used to modify the type of the user’s terminal and/or the modes associated with terminal I/O. The type as specified by this command is used for determining character conversion and delay timings, and has no effect on communications line control.

**usage**

```
set_tty -control_args-
```

where control_args are one or more of the following:

- **-type device_type** causes the user’s terminal type to be set to device_type, which may be any one of the following:
  - 1050
  - 2/41
  - TTY37 or tty37 (Teletype Model 37)
  - TTY33 or tty33 (Teleprompter Model 33)
  - TTY38 or tty38 (Teletype Model 38)
  - R300 or tnc300 (Terminet 300/1200)
  - ARDS or ards (ARDS or Tektronix)
  - ASCII or ascii (General ASCII terminal)

  The default modes for the new terminal type will be turned on.

- **-modes mode_string** sets the modes for terminal I/O according to mode_string, which is a string of mode names separated by commas, each one optionally preceded by "~" to turn the specified mode off. For a list of valid mode names, see the description of the tty_ I/O module in the Subroutines volume of the UNIX manual. Modes not specified in mode_string will be left unchanged. See Notes, below.

- **-reset** turns off all modes which are not set in the default modes string for the current terminal type.
-tabs

specifies that the device has software-settable tabs, and that the tabs are to be set. This control argument currently has effect only for terminet 300-like devices.

-print

causes the terminal type and modes to be printed on the terminal. If any other control arguments are specified, the type and modes printed will reflect the result of the command.

Notes

The effect of this command can be viewed as performing the following steps in the specified order:

1. If -type is specified, set the specified type, and turn on the default modes for that type.

2. If -reset is specified, turn off all modes which are not set in the default modes string for the current terminal type.

3. If -modes is specified, turn on and off those modes specified explicitly in the mode_string.

4. If -tabs is specified, and the terminal has settable tabs, set the tabs.

5. If -print is specified, print the type and modes on the terminal.
**TITLE:** Install new tty dim tools  
**AUTHOR:** M. Grady  

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Status</th>
<th>DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td>Written</td>
<td>04/29/75</td>
</tr>
<tr>
<td>X Sys. Anal. Tools</td>
<td>Expires</td>
<td>11/06/75</td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**DOCUMENTATION CHANGES**

- BOS
- Salvager MPM (Vol., Sect.)
- Ring Zero PLMS (AN #) 51
- Ring One
- MosN (Vol., Sect.)
- SysDemon/Admin.
- MPAM (Vol., Sect.)
- User Cmmd/Subr.
- MSAM (Vol., Sect.)

**Objections/Comments:**

- None (Reason)

**SUMMARY:**

Install two tools which are useful for tty dim problem analysis:

1) **tty_dump**, - dumps and interprets the ring 0 tty data for a particular person/line.

2) **tty_analyze**, - analyzes tty dim segments which have been extracted from a dump.

**REASONS:**

These tools have been cleaned up for MCS, and should be installed.
This command causes the contents of the control blocks and buffers associated with a specified TTY line to be printed on the user's terminal.

**Usage**

```
tty_dump identifier
```

where identifier is one of the following:

1. a person identifier, in which case, if the specified person is logged in from a terminal, the control blocks associated with the communications line to that terminal are dumped;

2. a line identifier of the form used in the "lines file" (e.g. "tty004"). If this form is used, the control blocks associated with the specified line are dumped whether or not the line is dialed up.
**Name:** tty_analyze,

This command analyzes the contents of a copy of tty_buf which it extracts from a dump. It performs certain tests to verify the consistency of the contents of tty_buf.

**Usage:**

```
tty_analyze erfno -control_arg-
```

where:

1. **erfno** is the number of the dump from which the copy of tty_buf is to be extracted.

2. **control_arg** may be -long (or -lg). If present the contents of each input and output buffer will be printed, otherwise only the addresses of the buffers will be printed.
**SUMMARY:**

Fix bug in pll_abs (etc.) causing it to abort when a pathname argument is shorter than the appropriate suffix, ie.

`.pll`, `.fortran` or `.alm`
## Multics Change Request

### TITLE: Make retriever handle up to 250 requests

**AUTHOR:** M. A. Meer

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Status Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td>Written: 05/01/75</td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td>Expires: 11/06/75</td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td>A 05/06/76</td>
</tr>
</tbody>
</table>

### SUMMARY:

Change the reloader to accept up to 250 retrieve requests (or up to 125 if not the same name as on tape) in a single file.

Also allow up to 50 renames among the requests.

### REASONS:

Current limit of up to 50 requests (25 if names not the same as on tape) is sometimes found to be inconvenient by some users.
**MULTICS CHANGE REQUEST**

**TITLE:** Replace `get_library_segment` with the `library_fetch` command

**AUTHOR:** Gary C. Dixon

**Planned for System:** MR 3.0

**Fixes Bug Number(s):** not applicable

**Documented in MTB:** revised MTB-166 attached

**Incompatible Change:** yes

**User/Operations-visible Interface Change:** yes

**Coded in:** (B)PL/I ( )ALM ( )other-see below

**Performance:** ( )better ( )same ( )worse

**Replaces MCR:** 11058

**DOCUMENTATION CHANGES (specify one or more):**
- `get_library_segment.info`
- `library_fetch.info`, `pc.info`

**HEADINGS ARE:** SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (optional)

---

**SUMMARY:**
Replace the `get_library_segment` command and all of its assorted control files and its online library search program (`get_primary_name`) with the new `library_fetch` command. The command is documented in the attached writeup.

**REASON:**
The `library_fetch` command uses library descriptors to find entries in a library. This allows information about library structure and contents to be concentrated in a library descriptor rather than in the `library_fetch` command. Since this information is shared by the `library_fetch`, `library_info`, `library_map`, `library_print`, and `library_cleanup` commands, libraries can be added or changed without having to modify all of these commands.

`library_fetch` also implements a full renaming capability for fetched entries, which `glsl` couldn't do. In addition, its search names allow the star convention and its renaming facility uses the equals convention. It provides an output log facility which records when each library entry was fetched from the library.

**IMPLICATIONS:**
Advantages: Adding or changing the organization of the Multics System Libraries will be easier to do because only the library descriptor program will have to be modified (and perhaps the `multics_library_search_procedure`), rather than all of the library maintenance commands. The star convention and full renaming facilities will make it easier to fetch library entries according to the user's needs. These tools can also be
used for user-defined libraries, thus providing a sub-system library maintenance tool.

Disadvantages: Users will have to change from using gIs to using library_fetch. Because the overhead of the library descriptor mechanism used by library_fetch is larger than the specialized mechanism used by gIs, the response time and CPU time required to fetch a library entry will be longer.

DETAILED PROPOSAL:

The change from gIs to if will be performed according to the following installation plan:

1) Install the library_fetch command, library_fetch.info and pending_changes.info (indicating coming deletion of gIs).
2) delete gIs and its control segments and subroutines at some suitable time in the future.

By default, library_fetch will search the same libraries that gIs does now.

The Design Review process for library_fetch concluded that the command had too many control arguments, and that the output arguments documented in MTB-166 should be excluded from the final command. This changes has been reflected in the attached document.

From several subsequent Design Reviews on other topics and from discussions with MCR Board members, I have concluded that it is best for a command to have control arguments indicating the defaults, as well as control arguments which vary from the defaults. This has been reflected in the attached document by the introduction of four new arguments: -brief (opposing -long); -no_chase (opposing -chase); -entry (opposing -container and -components); and -omit (opposing -retain). It was much more natural to document the defaults in the resulting command description, and I feel this is generally a good practice.

Care was taken in the attached document to describe first the most often used control arguments.

I have also attached a section of the Library Maintenance PLM referenced by the attached library_fetch documentation which describes how library_fetch uses library descriptors.
The library_fetch command copies entries from a library into the user's working directory. Control arguments allow copying the entries into another directory or renaming them as they are copied; select which library entry names are placed on the copy; allow copying the library entry which contains a matching entry instead of the matching entry itself (e.g., copy the archive which contains a matching archive component); or copying all of the components of the containing entry. A documentation facility is provided for recording in a file the status of each entry which is copied.

This command uses a library descriptor and library search programs, as described in Section IV.

Usage

library_fetch -search_names= -control_args=

where:

1. search_names are entry names which identify the library entries to be copied. The Multics star convention may be used to identify a group of entries with a single search name. Up to 30 search names may be given in the command. If none are given, then any default search names specified in the library descriptor are used.

2. control_args are selected from the following list of control arguments and can appear anywhere in the command:

   -library library_name,
   -lb library_name

identifies a library which is to be searched for entries matching the search names. The Multics star convention may be used to identify a group of libraries to be searched. Up to 30 -library control arguments may be given in each command. If none are given, then any default library names specified in the library descriptor are used.
-name, -nm indicates that all of the names on each matching library entry are to be placed on the copy. See the discussion of naming considerations under "Notes" below.

-primary, -pri indicates that the first name of each matching library entry is to be placed on the copy. See the discussion of naming considerations under "Notes" below.

-match indicates that, for each matching library entry, the entry names which match any of the search names are to be placed on the copy. See the discussion of naming considerations under "Notes" below. This is the default.

-into path identifies the directory into which library entries are copied and indicates how they are renamed. An absolute or relative pathname may be given. The directory portion of the pathname identifies the directory into which each library entry is copied. The final entry name of the pathname is used to rename each library entry name being placed on the copy, under control of the Multics equal convention. Only one -into control argument may appear in a command line. If -into is not given, matching entries are copied into the user's working directory and no renaming occurs.

-chase indicates that the target of a matching library link is to be copied.

-no_chase indicates that a warning message is to be printed when a matching link is found in the library, and that no copying is to occur. This is the default.

-long, -lg causes the pathname of each matching entry to be printed on the user's terminal as the entry is copied.

-brief, -bf suppresses printing the pathname of matching entries. This is the default.
-container
causes the library entry which contains each matching entry to be copied, instead of the matching entry itself. See the discussion under "Notes" below.

-components
causes all of the component library entries of a matching library entry to be copied, rather than just the matching entry itself. It also causes all components of a library entry containing a matching component to be copied. See the discussion under "Notes" below.

-entry, -et
causes each matching library entry itself to be copied. This is the default.

-search_name search_name,
identifies a search name which begins with a minus (-) to distinguish the search name from a control argument. There are no other differences between the search names described above and those given with the -search_name control argument. One or more -search_name control arguments may be given in the command.

-descriptor ref_name
gives a reference name which identifies the library descriptor describing the libraries to be searched. If no -descriptor control argument is given, then the default library descriptor is used.

-retain, -ret
indicates that library entries which are awaiting deletion from the library (as determined by the library search program) are to be copied.

-omit
indicates that library entries awaiting deletion from the library are to be omitted from the search, and are not to be copied. This is the default.

-output_file file,
-indicates that status information for each copied library entry is to be appended to a file. A relative or absolute pathname of the file may be given. If it does have a suffix of fetch, then one is assumed.
-all, -a indicates that all available status information for copied library entries is to be recorded in the output file.

-default, -dft indicates that only default status information is to be recorded in the output file. This is the default.

Notes

Any combination of the control arguments governing naming (-name, -primary, and -match) may be given in the command. However, the following groups of control arguments are mutually exclusive, and only one argument from each group may be given in the command: -chase and -no_chase; -long and -brief; -container, -components, and -entry; -retain and -omit; and -all and -default.

An -all or -default control argument may only be specified when the -output_file control argument is also given. The particular status information recorded in the output file for the -default control argument is under the control of the library search program. It includes the information deemed most important for the type of entry contained in the library.

If the file given in the -output_file control argument does not exist, it is created by library_fetch. If it does exist, new status information is appended to the end of the file preserving any previously recorded status. This feature allows the user to build a history of the entries copied out of a library.

When using the -into control argument, care must be taken to insure that the equal name included in the -into pathname can be applied to all names to be placed on each of the copied entries. Name duplications can easily result when more than one library entry matches the search names.

The -container and -components control arguments are provided to facilitate copying all of the library entries included in a given bound segment or related to a given subsystem. For example, by identifying a component of the source archive for a bound segment and using the -container control
argument, the entire source archive is copied into the user's directory. Similarly, by identifying a directory in the library containing all of the component entries of a subsystem and using the -components control argument, each component is copied into the user's directory.

When the -container, -components, or -chase control arguments are used, it may happen that none of the entrynames on a copied library entry matches any of the search names. Because the user may have requested that only matching names be placed on the copies, the library search program causes the first entryname to be placed on the copy when one of these three control arguments is used, in addition to any names requested by the user.

Examples

    library_fetch abbrev.pl1 -into >udd>Multics>user>new_=.=

copies the source segment abbrev.pl1 into the directory >udd>Multics>user, renaming the copy new_abbrev.pl1.

    library_fetch bound_runoff_.* -library online

copies all of the segments in the online libraries whose names begin with bound_runoff_ into the user's working directory. This might include the source archive, bindable object archive, bound object segment, and bind listing.

    if bound_runoff_.* -library online.source -components

copies all of the source components from the source archive for bound_runoff_ into the user's working directory.

    if qedx.pl1 -components

copies all of the source components in the archive containing qedx.pl1 into the user's working directory.
library_fetch

library_fetch *.alm -lb network.source -into new_*.alm

copies all ALM source segments from the network source library into the user's working directory, and adds new_ to the names placed on each segment.

library_fetch pl1_status.info -nm -lb info

copies the pl1_status.info segment from the info segment libraries into the user's working directory, copying all entrynames from the library entry onto the copy.

library_fetch **.ec -library online.???????

copies all exec_com segments from the online source and object libraries into the user's working directory.

library_fetch -lb supervisor.bc bound_sss_wired_.*?

copies the bind segment from the bindable object archive called bound_sss_wired_.archive. Note that although the object archive itself matches the search name which was given, only the matching archive component is copied because the -container control argument was not given.

library_fetch -lb include stack_frame.incl.*

copies the stack frame declaration include segments for all source languages from the include library into the user's working directory.

DRAFT: SUBJECT TO CHANGE
**TABLE:**

| TITLE: | Fix newline bug in abbrev |
| AUTHOR: | S. Herbst |

**Status:**
- **MCR 1159**
- Written: 05/05/75
- Expires: 11/12/75
- Status: A

**Category (Check One):**
- Lib. Maint. Tools
- Sys. Anal. Tools
- Sys. Prog. Tools

**Document:**
- BOS
- Salvager
- Ring Zero
- Ring One
- SysDaemon/Admin.
- Runtime
- User Cmnd/Subr.
- MPM (Vol, Sect.)
- PAMS (AN #)
- MOSN (Sect.)
- MPAM (Sect.)
- MSAM (Sect.)

**Objections/Comments:**
- Info Segs
- Other (Name): None (Reason) no change

**Use these headings:** SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional)

**SUMMARY:**

Fix abbrev to handle request lines (lines beginning with a period) that do not end in newline.

**REASON:**

If abbrev is given a request line whose last character is not a newline, it sometimes ignores the last character. Requiring that all command lines passed to the current command processor end in newline is an unnecessary restriction and, in fact, one that is not universally upheld today.
TITLE: Implement io_call move_attach

AUTHOR: S. Herbst

SUMMARY:

Implement the move_attach operation in the io_call command.

REASONS:

Users will be able to call the new entry iox_$move_attach from command level.

DETAILED PROPOSAL:

"io_call move_attach X Y" looks for an existing block X, aborts if not found, looks for and creates if necessary a block Y, calls iox_$move_attach with pointers to X and Y, and reports any errors.
(WPM Commands, under io_call)

move_attach OPERATION

The usage of move_attach is:

io_call move_attach switchname target_switchname

where:

1. switchname is the name of an attached I/O switch.
2. target_switchname is the name of a detached I/O switch.

This command moves the attachment of the first I/O switch to the target I/O switch and leaves the first switch in the detached state.

If a control block for the target switch does not exist, one is created.
TITLE: Add MPC dumper facility to BOS

AUTHOR: Noel I. Morris (for D. A. Kayden)

- Coded in [ ] PL/I [X] ALM [ ] other-
  explain in DETAILED PROPOSAL
- Planned for System MR
- Fixes Bug Number(s)
- Documented in MTB
- Category (Check One)
  Lib. Maint. Tools
  Sys. Anal. Tools
  Sys. Prog. Tools

- User/Operations-visible
  Interface change? [ ] yes [X] no
  Incompatible change? [ ] yes [X] no
  Performance: [ ] Better [X] Same
  Replaces MCR

Document Specify One or More

- Document
  MFM (Vol, Sect.)
  PLMS (AN #)
  MOSN (Sect.) 5
  MSAM (Sect.)

- Specified
  Info Segs
  Other (Name)
  None (Reason)

Use these headings: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional)

SUMMARY:

For the purposes of debugging and tracing hardware/firmware/software problems, it is useful to be able to dump MPC storage and trace tables. The new BOS command, MPCD, will provide this capability. An MOSN will be issued describing the use of the MPCD command.
TO: Operations
FROM: Noel I. Morris
DATE: 27 May 75
RE: Command to dump MPC

A new BOS command has been made available to assist programming staff in system development and system malfunction analysis. This command will dump MPC memory and the trace table kept by the MPC.

Usage:

MPCD -devname- -trace- -dump-

-devname- is the name of a device connected to the mpc. It must be described in the config deck by a PRPH card.
-trace- will, if present, dump the MPC trace table.
-dump- will, if present, dump MPC memory.

Note

When dumping the MPC trace table, switch 4 on the appropriate MPC should be placed in the UP position in order to inhibit further tracing.
TITLE: Fix bug in BOS card reading program

AUTHOR: Noel I. Morris

SUMMARY:
A coding bug in the BOS card reading module causes it to attempt to read more cards after an EOF card is detected.

Proposal:
Fix the bug.
SUMMARY:
AFDSC has complained that they are unable to load a BOS tape which was written on 7-track handler. The problem was traced to the BOS 5-card loader. The 5-card loader program attempts to set the bootload handler to 1600 bpi and then read the tape. If the read fails because of a PE/NRZI compatibility check, it tries again after setting the density to 800 bpi. If the handler is a 7-track handler, the SD16 (set density to 1600 bpi) command is rejected. The 5-card loader then uses whatever density that the drive was already set to. If this is not equal to 800 bpi, the tape will not be loaded successfully.

Proposal:
Change the 5-card loader to set density to 800 bpi first. If a compatibility check occurs, then try 1600 bpi.
TITILE: Install new record_io_ and bound_plio2_

AUTHOR: R. Schoeman

SUMMARY: Install a new record_io_ and a new bound_plio2_.

REASONS: record_io_ in bound_sss_wired_ has been changed to:

1) do fast record i/o, i.e. call iox_ directly, for keyed files and delete statements, cases which were excluded in previous version;
2) fix several bugs which were found in it which had the effect of excluding some unkeyed cases which could have been done directly from fast handling.

The speed improvement in the cases now handled directly is 15% - 60%. plio2_recio_ in bound_plio2_ required minor changes at its interface with record_io, and a minor bug was fixed in it.

IMPLICATIONS: There are no user-visible changes, outside of the decreased execution time.

DETAILED PROPOSAL: record_io_ is coded in alm, plio_recio_ is in PL/1.
**TITLE:** Install new user ring tty dim  
**AUTHOR:** S. Herbst

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Documentation Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td>Document Specify One or More</td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td></td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td></td>
</tr>
<tr>
<td>355</td>
<td></td>
</tr>
<tr>
<td>BOS</td>
<td></td>
</tr>
<tr>
<td>Salvager</td>
<td>MPM (Vol. Sect.)</td>
</tr>
<tr>
<td>Ring Zero</td>
<td></td>
</tr>
<tr>
<td>Ring One</td>
<td>PLSM (AN #) User I/O (57)</td>
</tr>
<tr>
<td>SysDaemon/Admin.</td>
<td>MOSN (Sect.)</td>
</tr>
<tr>
<td>Runtime</td>
<td></td>
</tr>
<tr>
<td>X User Cmd/Subr.</td>
<td>MSAM (Sect.)</td>
</tr>
</tbody>
</table>

**Objections/Comments:**  
- Use these headings: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional)

**SUMMARY:**

Replace the user ring portion of the old tty dim with the IOX dim tty.

**REASON:**

IOX conversion.

**DETAILED CONVERSION:**

tty operations are currently performed as write-arounds by the module tty_attach, which also contains write-arounds for a variety of other converted dims. This change takes out the tty entries and puts them in a separate module tty that has the entry tty_attach. The old module tty_attach with tty entries removed is renamed to ios_write_around._
# Multics Change Request

**TITLE:** Convert bound_command_loop to IOX

**AUTHOR:** S. Herbst

<table>
<thead>
<tr>
<th>Coded in</th>
<th>Category (Check One)</th>
<th>Status</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ ] FL/I</td>
<td>Lib. Maint. Tools</td>
<td>Written</td>
<td>05/02/75</td>
</tr>
<tr>
<td>[ ] ALM</td>
<td>Sys. Anal. Tools</td>
<td>Expires</td>
<td>11/31/75</td>
</tr>
<tr>
<td>[ ] other</td>
<td>Sys. Prog. Tools</td>
<td>DOCUMENTATION CHANGES</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Explain in DETAILED PROPOSAL</th>
<th>355</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Planned for System MR</th>
<th>Salvager</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Fixes Bug Number(s)</th>
<th>Ring Zero</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Documented in MTB</th>
<th>Ring One</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>User/Operations-visible</th>
<th>SysDaemon/Admin.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Interface change?</th>
<th>MPAMS (AN #) Commands</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Incompatible change?</th>
<th>MOGN (Sect.)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Performance:</th>
<th>Runtime</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Replaces MCR</th>
<th>User Cmmd/Subr.</th>
</tr>
</thead>
</table>

---

**Objections/Comments:**

- Info Segs
- Other (Name)
- None (Reason)

---

**SUMMARY:**

Convert bound_command_loop to use iox.

**REASON:** IOX conversion

**DETAILED PROPOSAL:**

user_real_init_admin will try an IOX attachment first and if the I/O module cannot be found will try an IOS attachment. This is done so that users can still specify an IOS outer module.

Naming conventions for IOS and IOX modules are different.
**TITLE:** Fix bug in copy_seg

**AUTHOR:** S. Herbst

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Document</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td>BOS</td>
<td>MPM (Vol, Sect.)</td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td>Salvager</td>
<td>PLMS (AN #)</td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td>Ring Zero</td>
<td>MOSN (Sect.)</td>
</tr>
<tr>
<td></td>
<td>Ring One</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SysDaemon/Admin.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Runtime</td>
<td></td>
</tr>
<tr>
<td></td>
<td>X User Cmmd/Subr.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MSAM (Sect.)</td>
</tr>
</tbody>
</table>

**DATE:** 05/06/75

**STATUS:**

<table>
<thead>
<tr>
<th>Written</th>
<th>Status</th>
<th>Expires</th>
</tr>
</thead>
<tbody>
<tr>
<td>Written</td>
<td>A 05/ 13/75</td>
<td>11/13/75</td>
</tr>
</tbody>
</table>

**Objections/Comments:**

- No change
- Use these headings: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional)

**SUMMARY:**

Change copy_seg to handle the case where the target is an existing link to the source segment as a name duplication on the link rather than an attempt to specify the same segment as source and target.

**REASONS:**

The copy command currently says:

"Attempt to specify the same segment as both old and new".

in the above case, whereas it should say:

"Name duplication. Do you want to unlink the old link etc.?"
<table>
<thead>
<tr>
<th>TITLE: Fix bug(s) in stu_</th>
<th>AUTHORS: J. M. Broughton</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATUS DATE</td>
<td></td>
</tr>
<tr>
<td>Written</td>
<td>05/06/75</td>
</tr>
<tr>
<td>Expires</td>
<td>11/3/75</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Lib. Maint. Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixes Bug Number(s)</td>
<td>Sys. Prog. Tools</td>
</tr>
<tr>
<td>Documented in MTB</td>
<td>355</td>
</tr>
<tr>
<td>User/Operations-visible</td>
<td>BOS</td>
</tr>
<tr>
<td>Interface change? yes</td>
<td>Salvager</td>
</tr>
<tr>
<td>Incompatible change? yes</td>
<td>Ring Zero</td>
</tr>
<tr>
<td>Performance: Same</td>
<td>SysDaemon/Admin.</td>
</tr>
<tr>
<td>Replaces MCR</td>
<td>X Runtime</td>
</tr>
<tr>
<td></td>
<td>User Cmm/Subr.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DOCUMENTATION CHANGES</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPM (Vol, Sect.)   AK92 SWG - Subrout</td>
<td>Info Segs</td>
</tr>
<tr>
<td>PLMS (AN #)</td>
<td>MPAM (Sect.)</td>
</tr>
<tr>
<td>MO6N (Sect.)</td>
<td>MSAM (Sect.)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Objections/Comments:</th>
<th></th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>REASONS:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>There is a problem in finding the address of members of PL/I structures and FORTRAN common blocks that contain several large arrays. The precision of the variables in question is not sufficient to adequately handle the large offsets that can be encountered.</td>
</tr>
</tbody>
</table>

The parameter/return value for decode_runtime_value has always been misdeclared. Correct operation has hinged on the fact that fixed bin and fixed bin(35) both occupy a full word. However, the small precision contributes to the above problem and must now be changed.

<table>
<thead>
<tr>
<th>IMPLICATIONS:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>User/system programs calling stu_$decode_runtime_value should be changed to reflect the new declaration of the entry. There is, however, no foreseeable change in PL/I which would prevent those programs which now work correctly from continuing to work.</td>
</tr>
</tbody>
</table>
TITLE: Change operation of delete_ on directories

AUTHOR: S. Herbst

Category (Check One)

- Lib. Maint. Tools
- Sys. Anal. Tools
- Sys. Prog. Tools

- Code in PL/I
- Coded in other
- Planned for System MR
- Fixes Bug Number(s)
- Documented in MTB

User/Operations-visible

Interface change? [ ] yes [X] no
- Incompatible change? [ ] yes [X] no
- Performance: [X] Better [ ] Same
- Replaces MCR 1149

- Document Specify One or More

Objections/Comments:

Use these headings: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional)

SUMMARY:

Change delete_ to delete a directory containing message segments and mailboxes.

REASONS:

It is too difficult and confusing to delete such a directory today using the delete_dir command and takes more than one invocation of the command.

DETAILED PROPOSAL:

Replace call to hcs_$del_dir_tree with some internal code that calls mailbox_$delete and message_segment_$delete when necessary. If the delete operation still fails, a message is printed naming the branch on which it failed.

IMPLICATIONS:

hcs_$del_dir_tree becomes an obsolete interface and can be deleted.

Additional feature: delete_dir will terminate and unsnap links to all branches that are deleted in a subtree.

NOTE: del_dir_tree must remain in ring zero.
** TITLE: ** Increase the size of IMP DIM link tables.

** AUTHOR: ** Raj Kanodia

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DOCUMENTATION CHANGES</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Document Specify One or More</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MPM (Vol, Sect.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLMS (AN #)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MO6N (Sect.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MPAM (Sect.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSAM (Sect.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

** STATUS **

<table>
<thead>
<tr>
<th>Status</th>
<th>Written</th>
<th>Expires</th>
</tr>
</thead>
<tbody>
<tr>
<td>A 05/13/75</td>
<td>11/13/75</td>
<td></td>
</tr>
</tbody>
</table>

** Use these headings: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional) **

** SUMMARY: **

At present, the IMP DIM link tables permit at most 84 simultaneous Network connections to foreign hosts. With recent increase in the number of network hosts that are on the Network and in constant communication with mit-multics, (approximately 72), the situation has reached a crisis proportion with only 12 connection being available for users (some of which get paralyzed due to an NCP bug.)

The proposed change will double the size of the tables.
MULTICS CHANGE REQUEST  MCR 1173

TITLE: Add commenting capability to reduction compiler language
AUTHOR: Gary C. Dixon

Planned for System: MR 3.0
Fixes Bug Number(s): not applicable
Documented in MTB: not applicable
Incompatible Change: no
User/Operations-visible Interface Change: yes
Coded In: ( ) PL/I ( ) ALM ( ) other-see below
Performance: ( ) better ( ) same ( ) worse

DOCUMENTATION CHANGES (specify one or more)
MPM (vol, sect) MPAM (sect)
MOSN (sect) MSAM (sect)
PLMs (AN#) AN51
Info Segs
Other

OBJECTIONS/COMMENTS:

HEADINGS ARE: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (optional)

SUMMARY:
Allow comments to be included in a set of reductions.

REASONS:
Comments would make it possible to write a more understandable set of reductions which can be more easily maintained. They would provide a mechanism for the author of the set of reductions to express his ideas and intentions for the reductions.

A comment mechanism was included in the original design for the reduction language, but was omitted from the final implementation because time pressures prevented a decision on a good set of comment delimiters. Subsequent thought on the matter and discussions with rdc users have lead to the proposal detailed below.

IMPLICATIONS:
It will be easier for the author to express the intended function of a set of reductions, thereby conveying additional information to subsequent maintainers of an rdc translator.

DETAILED PROPOSAL:
The comments are intended for use as a header for a group of reductions, or to clarify the meaning to the reader of a single reduction. It is desirable that the comments be used in such a way that they do not obscure the otherwise simple organization of a reduction statement. The following convention was chosen to promote this pattern of usage:

Page 1
The characters " begin a comment in the reduction language, which ends with the next newline character.

Examples of expected usage are:

```
\" Pass one: all tokens are scanned to identify labels and to count the tokens in the input string.
pass1 / <name> / set_label LEX / \ 
/ ( / LEX / skip \ 
```

and

```
label / <is_it> / [ptr=complicated] / \ 
\" when it is, make sure ptr is updated to \ 
\" reflect complicated relationship. 
/ <any> / ERROR(5) / \ 
```

The choice of delimiters was based upon three factors:

1. Maintaining upwards compatibility with existing rdc translators. It is likely that no translators besides the reduction_compiler use \ in their syntax specifications, and \ is not legal in any other field of a reduction statement except as the end-of-statement delimiter. An end-of-statement delimiter followed by a " is an impossible sequence in the current reduction language because the field following the end-of-statement delimiter is the label field of the next statement, and labels cannot contain " characters.

2. Providing a commenting convention similar to one of the existing conventions. The PL/I convention could not be used without changing the comment delimiters because the set of reductions is already contained as a comment in the PL/I source segment, and PL/I comments cannot contain nested PL/I comments. The ALM convention was chosen because of the beneficial restrictions it places on the format of comments. The ALM " comment delimiter had to be replaced by " to differentiate a comment from the beginning of a quoted string. In many ways, an escaped quote (\") is an appropriate comment delimiter.

3. Not introducing any new break characters into the reduction language. In general, adding break characters to a language is a non-upwards compatible change because it causes the compiler to translate differently any existing source segments coded in any language.
SUMMARY:
Make reduction_compiler_ (the translation subroutine for the reduction_compiler) properly translate any PL/I statement used as a semantic statement. Currently, it does not properly translate a statement which contains two PL/I atoms not separated by an operator. For example, it translates

```
(if able = 5 then go to quit)
```

as

```
if abe=5thengotoquit;
```

instead of as

```
if able = 5 then go to quit;
```

REASONS:
The documentation for the reduction_compiler states that any PL/I statement can be used as a semantic statement.

DETAILED PROPOSAL:
The problem will be solved by putting a space in the output before every token in a semantic statement, with the following exceptions.

1. No space will be placed between the last token of a semantic statement and its ending semi-colon statement delimiter.

2. No space will be placed between an argument in a subprogram call and any comma delimiter which follows it.
3. No space will be placed between a quoted string and any token which follows it, in order to handle bit string constants (e.g., "101"b).

4. No space will be placed between any of the following pairs of characters which have a special meaning in the PL/I language:

   -> >= <= -= -> <=

5. No space will be placed between any minus sign (-) and the token which follows, in order to handle signed numeric constants.

In addition, the semantic statements appearing in the same pair of brackets will be placed on different lines in the output segment.

To do this, four new break characters must be added to the reduction language:

   ^ - = ;

IMPLICATIONS:

Advantages: the reduction compiler will stop generating bad PL/I code for semantic statements. Code for semantic statements will be output in a more readable format.

Disadvantages: existing programs which use the reduction compiler may have to be changed if they use any of the new break characters in an absolute token specification with other characters. For example, "able-baker" is now one specification but would become three after the change unless enclosed in quotes. This is not deemed a serious drawback because no code generated by the reduction compiler has been installed yet.

The extra reductions required to properly translate a semantic statement add about 10% more virtual CPU time to the reduction compilation of a translator. However, since compiler response time is still noticeably good, and since the reduction compiler is a lightly-used tool, this performance degradation is not very significant.