Published: 03/04/67

Identification

Overview of dedicated resource management
J. H. Saltzer

Purpose

Some resources of a computer system are not easily shared among competing users on a minute-by-minute demand basis, but are more effectively dedicated to a specific user for some period of time despite the fact that the user may not achieve full utilization of the resource. Examples of resources which are typically dedicated are tape drives, tape reels, line printers, and console typewriters. Sometimes, resources which are normally shared may be also dedicated. For example, in some cases it may be appropriate to dedicate 20 per cent of the capacity of one processor to a single user.

Discussion

Dedicated resource management divides into two functions: The administrative function of reserving and allocating resources; and the security function of protecting resources allocated to one user from errors of other users.

Dedicated resource management is accomplished by utilizing the Multics ring structure. Three rings are assumed; the user (unprotected) ring, an administrative ring, and the hardcore ring. Located in the user ring are the user procedures attempting to utilize dedicated resources. In the administrative ring is a module known as the reserver and its data bases, the reservation tables. In the hardcore ring is the resource assignment manager and its data base, the resource assignment log.

The Reserver

The reserver handles two distinct administrative jobs: advance reservation and resource allocation at time of usage. We begin by considering advance reservation. Sometime in advance of the desired usage of a resource, the user (with the aid of a reservation command) calls entry point "reserve" in the administrative ring of one of his own working processes. He gives appropriate arguments to specify the nature of his request (desired usage interval, type of resource, etc.) and upon return receives an immediate reply confirming or denying the request.

The nature of the reservation algorithm is unimportant at this point. It may vary from device to device and from installation to installation. In general, the Reserver maintains a set of reservation files in the reservation directory described in BT.2. The reservation is made in the name of the user, and any process operating for the user may utilize the reservation.

The second job of the reserver is to check requests to use resources against previously made reservations. When the time for the usage of the resource occurs, the user performs an "allocate" call to the reserver specifying a previously reserved resource. The reserver, operating in the administrative ring of the working process, checks its reservation files to see if this user process-group has actually made an appropriate reservation. If the device is to come out of a pool, the reserver at this time selects a specific device from the pool. The reserver then performs a call to the "assign" entry of the resource assignment module, which operates in the hardcore ring. This call requests the resource assignment module to make the current assignment of the requested resource be this process of this process group. Finally, the reserver calls the dedicated resource metering module so that the starting time of the resource assignment can be noted for accounting purposes.

The reserver is described in detail in section BT.2.

The Resource Assignment Manager

The second major task of dedicated resource management is protecting the resources assigned to one user from accidental or intentional misuse by another user. This protection is accomplished, basically, by placing in the hardcore ring a module through which all resource assignment requests are directed and checked for authority and conflicts with other assignments. All requests of user procedures which imply using a resource (for example, a call to the GIOC interface module to post a list of channel commands) are then merely checked against the assignment initiated by the resource assignment manager. If a user process attempts, for example, to start output on a GIOC channel not assigned to it, the GIOC interface module can reject the call. The resource assignment manager may be called only from the administrative ring. This restriction insures that some administrative procedure (e.g., the reserver) has the opportunity to compare allocation requests against previous reservations and also that resource usage accounting can be done.

There is a distinct registry file for each assignable dedicated resource of the system. For example, there is a registry file for each tape handler, and a registry file for each tape reel. Access to the registry file is the control on access to the resource. The contents of the registry file, described in BT.1.00, depend on the particular resource and are not of interest here except to note that they are not accessible above the hardcore ring. The resource assignment module checks only for the existence of and access to the registry file.

Put another way, the data base of interest to the resource assignment module is the registry file directory, which lists access rights to the individual registry files. The registry file directories may not be modified except by certain administrative users; the names of the registry files and their access control lists are therefore secure and reliable information. The registry file directories are accessible for searching (execute access) in the administrative ring, so that the reserver has the option of checking access rights at the time a reservation is made.

The registry files are organized into directories of similar resources; within a directory, the name of the registry file for a cparticular device is the same as the standard name for that device. Four directories are currently used:

1. Input/Output devices
2. Major modules (system clocks, cpus, etc.)
3. Tape reels
4. Miscellaneous detachable media

When the resource assignment manager is asked (by the reserver) to allot a resource to a process it first looks in the registry files to determine whether or not the user has at this instant the privilege of using the resource.

If he does, and in addition the resource is not already assigned to another user, then the resource assignment module performs an assign call to some appropriate hardcore supervisor module, specifying the process identifier and the desired resource.

The module called to perform the actual assignment depends on which registry file directory contained the registry file, as shown below:

Registry File Directory	Module called	Entry point	where described
I/O devices	I/O device assignment module	assign_device	BF.
major modules	system assignment module	assign_module	BT.
tape reels other detachable media	media management module	assign_media	BT.

Although not currently worked out in detail, the planned ability for a user to operate with non-standard hard-core supervisor procedures will be validated by requiring the user to perform a call to assign himself each desired non-standard procedure. He then calls a special file system entry which actually performs the module replacements indicated by previous assign calls.

Figure one illustrates the relation between the user, the reserver, the resource assignment manager, and the hardcore supervisor. The resource assignment manager is described in detail in section BT.1.

Resource Assignment Log

Every resource assignment made by the resource assignment manager is entered in a logging segment twice: once at the beginning of the assignment and once at the end. A single entry in this log gives user process group identification, process identification, time and date, and resource name. Unsuccessful assignments are also logged. Whenever the logging segment "fills up"(32 words/entry allows 8000 entries in a 256K segment) the log is continued in another segment. The consecutive logging segments, stored in the resource assignment log directory, are intended to drop quickly to the lowest level of information storage since their use is for auditing and system checking, both infrequent operations.

I/O Device Assignment

Requests for I/O device assignment are forwarded, after validation, to the I/O device assignment module. This module, depending on the particular device being assigned, calls one of several hardcore-ring I/O interface modules. If the device being assigned is the only device on a particular GIOC channel, the GIOC interface module is called. The GIOC interface module records the process identification as the only one allowed to manipulate the channel in question. All future calls concerning a channel from outside the hardcore ring to the GIOC interface module are checked for validity by comparing the assignment previously recorded with the identification of the calling process. The identification of the calling process is reliably obtained by asking a traffic controller entry "who am I".

In the case of several devices which should be independently assigned but are all accessed by the same GIOC channel or channels, an additional interface module (e.g., the tape controller interface module) is interposed between the user and the GIOC interface module. The purpose of this extra module is to provide protection between users sharing the same channel. If the I/O device assignment module receives an assign-device call for a tape handler, it calls the tape controller interface module, which records the assignment. Again, all calls concerning a tape handler from outside the hardcore ring are validated by comparing the identification of the calling process obtained from the traffic controller with the identification stored in the tape handler assignment table. When the tape controller interface module performs a call to the GIOC interface module to actually initiate output on the shared channel, the GIOC interface module does not attempt to check the device assignment, because the call originated from within the hardcore ring.

The I/O device assignment module is described in detail in BT.

Media Assignment

Requests for media assignment are forwarded, after validation by the resource assignment module, to the media management module. This module merely adds the assignment to its list of all currently assigned media and returns. When later calls to the media management module are made by the user from outside the hardcore ring, it first checks to see if the medium requested is currently assigned to the calling process by comparing the identification of the calling process (obtained from the traffic controller) with that stored in the media assignment table. Only is the assignment is found, does the media management module initiate inter-process communication to the appropriate media operator requesting him to locate, load, unload, or return to storage the requested medium.

A full description of the media management module, including the meaning of "locate, load, unload, and return" will be found in BT.

Major Module Assignment

For some major modules (a major module is a central processor, system controller, system clock, GIOC, or drum controller) there is an appropriate meaning for assignment to a specific process. In the case of a system clock, it can be considered to be assigned to the alarm clock manager system process. In the case of a processor, a definition of assignment is provided by the ability of the traffic controller to allow a process to multiplex normally, but always "get" a specific processor for its quantum. This form of assignment is useful primarily to a product service engineer attempting to execute a processor test program on-line. At present, these are the only recognized assignments handled by the system assignment module.

Requests for major module assignments are forwarded, after validation by the resource assignment module, to the system assignment module. This module merely forwards the call to either the traffic controller entry assign_cpu (in the case of processors) or the hardcore clock interface module entry assign_clock. Calls to assign_clock are probably only appropriate at system initialization or reconfiguration time.

Example

As a example of the usage of the modules and ideas described above, consider the sequence of events which occur when a user wishes to utilize a magnetic tape. At some time in advance of the events considered here, the user has negotiated with his administrator to obtain the privilege of using tape drives and has also obtained permission to use one or more magnetic tape reels. (Section BO. is concerned with obtaining such privileges and permissions.) As a result of these negotiations, his login identification appears on the access control list of each tape drive and each tape reel he is permitted to use.

Perhaps a day or so before the intended usage, the user logs in and, using reservation commands described in BX., calls the reserver to find a time suitable to him in which a tape drive can be reserved. Depending on local installation policy and the tape reels involved he may or may not have to reserve his particular planned usage of the tape reels also. (A reservation is probably not needed for reels accessible to only one user.) Having made the reservation, he may log out. The memory of this transaction is kept in the reservation tables.

At the time scheduled for the reservation, the user logs in again. In anticipation of his planned usage of a tape reel, he issues a locate command. This command performs two calls of interest to dedicated resource management: allocate and locate. The allocate call, specifying the desired tape reel, is checked by the reserver against the reservation tables, if any are kept for the particular tape reel, and an assign call is made to the resource assignment manager. The resource assignment manager checks to see if this user is allowed access to this reel by looking at the registry file directory for tape reels. Since (we have assumed) the correct access control will be found, the resource assignment manager passes an assign_media call to the media management module, which notes the assignment in its tables. Control now passes by a series of returns to the locate command, which then issues the locate call to the media management module. As described in section BT., the meaning of this call is to locate and move the desired medium to the vicinity of the computer. The media management module checks its assignment table, finds the reel is assigned as expected, and issues an interprocess signal to the media operator, causing a message to appear at his typewriter console which suggests that he locate the needed tape reel in the tape vaults. Control now returns to the locate command which is now completed. The user may type other commands while the operator is locating the reel.

The user now calls the I/O system to attach an I/O stream consisting of a specified tape reel on a tape drive. The I/O system forwards this attach call to the tape device interface module (tape DIM) which then performs the following two calls:

1. Allocate, in the reserver, specifying a tape drive. The reserver checks and finds the previously made reservation. It chooses a particular tape drive from the available pool and calls assign in the resource assignment manager. The resource assignment manager looks up the tape drive in the registry files, and if the access is successful calls the assign_device entry of the I/O device assignment module. This module will look up the device in the configuration table, call the tape controller interface module to establish the assignment, and return the device index associated with the tape drive to the resource assignment manager. The resource assignment manager returns the device index to the reserver, which returns it to the tape DIM (in the user ring). This device index is the identifier which the tape DIM uses in future calls to the hardcore ring I/O modules.

2. Load, in the media management module, specifying the device index obtained in the previous step, and the reel name. The media management module checks its assignment table and sends an interuser signal to the operator to mount the reel on the drive.

The tape DIM waits (by calling the wait coordinator) for the signal that the load was successful, and then returns to its caller indicating that the original attach call was successful.

The user now calls read and/or write in the I/O system on his stream name. These calls are switched to the tape DIM, which transforms them into calls to the tape controller interface module specifying a device index. On each call, the tape controller interface module checks to see that the device index is one assigned to this process, and, if it is, completes the requested call by transforming it to a channel control list and passing it to the GIOC interface module.

Finally, if the detach arguments request it, the tape DIM calls return in the media management module to have the reel returned to storage. This final step may be skipped at the caller's option, implying that he will use the reel again soon or else explicitly call return by issuing a command. (If he forgets to issue a return command, logout will release all assigned resources including the reel.) The tape DIM now returns to its caller, indicating that the detach call has been completed.

At this point, no changes have been made in the user's original reservation. If he reserved a tape drive for two hours and issued the detach call after only 30 minutes of use, his reservation remains in effect, and the reserver will allow him to make another attach call to continue to use a drive. He may, of course, receive a different physical drive upon reattaching. If, instead, he is finished with the tape drive, he may release the reservation by an appropriate call to the reserver. The reserver may impose charging policies which encourage the user to release reservations he does not plan to use.

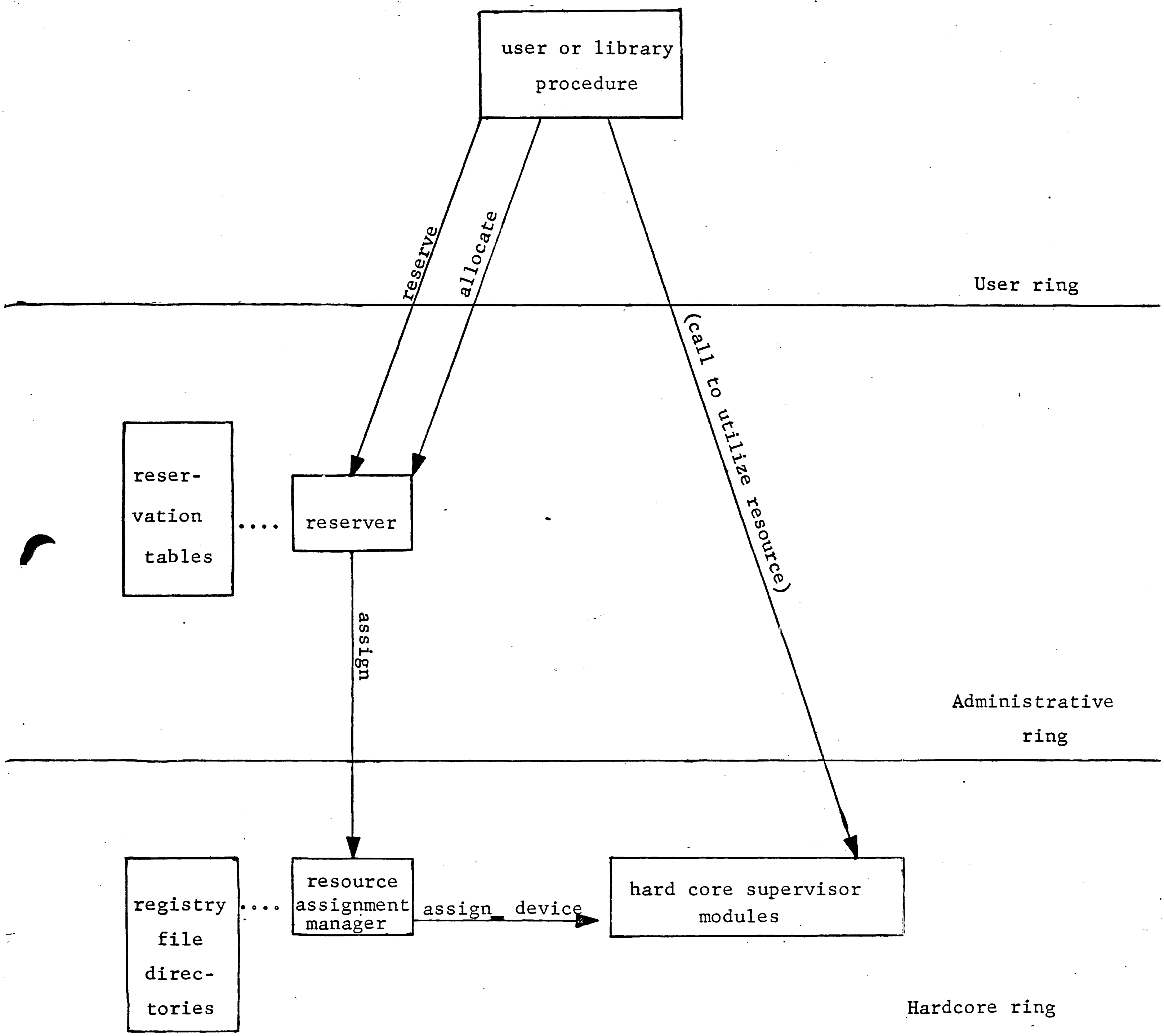

Figure 1 dedicated resource management