To: Distribution

From: Noel I. Morris

Date: January 27, 1976

Subject: Plans for the Obsolescence, Unwiring, and Eventual Removal of the DST

Background

The Device Signal Table (DST) is a wired hardcore data base which aids several hardcore ring I/O subsystems in maintaining I/O device identity. It is indexed through the use of a device index (devx). This index is used to maintain attachment data about a particular I/O device in conjunction with the I/O Assignment Module (IOAM). It is also used to send wakeups to a user ring process on behalf of an I/O device. The device index is also used by the iom_manager to identify a particular I/O channel. Many devices do not operate through the iom_manager. Space is reserved by the iom_manager for device indices which it never "sees".

Proposal

It is proposed here to modify the various I/O subsystems (TTY software, IOI, etc.) to reduce dependence on the DST. These subsystems can send user ring wakeups by calling pxss directly. Some of them no longer need the functions provided by the IOAM since these functions are now adequately provided by RCP. This will allow the shrinking and unwiring of the DST.

It is also proposed here to modify the iom_manager to use a channel index which is totally unrelated to the device index. The use of a separate index will allow operation with a smaller iom_data segment.

The iom manager

Currently, when a channel is assigned in the iom_manager, a device index is assigned. This index is given to the iom_manager on all subsequent calls to identify the channel. It is also given by the iom_manager to an interrupt handling procedure to identify the channel which received an interrupt. This very same device index is also used as the index into the DST.

There are several problems with this current mechanism. First of all, many device indices are taken up by devices not assigned through the iom_manager (e.g., TTY channels). The iom_manager, therefore, must reserve room for 512 device indices even though the maximum number of channels that could ever be assigned in a dual IOM system is slightly less than 128. (In practice, fewer than 24 IOM channels are usually needed.)

The second problem arises in the case where a single interrupt handling program is used to handle interrupts from many different I/O channels. A good example of this case is the DIM which manages disk paging. There may be several disk subsystems, each of which may have up to 8 channels in operation. Yet, the disk DIM provides a single entry point for the iom_manager to notify it of an interrupt on any of those channels. The device index given to the interrupt handler must be mapped into the correct disk subsystem and channel index. In some cases, this mapping can only be done by a linear search of a data base (but not in the case of the disk DIM). It is clear that some efficiency could be gained if an I/O subsystem were given a better index by the iom_manager at interrupt time.

iom_manager Changes

In order to solve the problems discussed above, a simple change is being made to the iom_manager. An additional argument will be given to the iom_manager at channel assignment time. This argument must be nonzero. It, instead of the device index, will be passed to the interrupt handling program by the iom_manager. In addition, the index returned by the channel assignment entry of the iom_manager will have nothing whatever to do with a DST index. It will be a number which is only valid when a particular I/O subsystem is calling the iom_manager to perform some operation on a channel. This change will decouple the iom_manager channel index from the DST device index.

This change should be noticeable only in calls to one entry of the iom_manager. It affects <u>all</u> callers of iom_manager$iom_assign:

```
declare iom_manager$iom_assign entry (fixed bin(8),
          fixed bin(2), fixed bin(6), entry, fixed
          bin(12), fixed bin(35));
call iom_manager$iom_assign (chx, iomno, channo,
          int_proc, idx, code);
```

chx — is an index assigned by the iom_manager to be used on all subsequent calls. This index may <u>not</u> be used to reference the DST. (Output)

iomno — is the IOM to which the channel being assigned is connected. (Input) channo — is the number of the channel being assigned. (Input)

int_proc — is a procedure which will be called when an interrupt is received by the iom_manager for this channel. (Input)

idx — is an index, supplied by the caller, which will be given back to the caller when the interrupt procedure is called by the iom_manager. This index must be nonzero. (Input)

code — is an error code. If chx is zero and code is nonzero, an illegal iomno and/or channo is indicated. If chx and code are both nonzero, a prior assignment to the specified channel is indicated. (Output)

The calling sequence of the interrupt handling procedure does not change. However, the index given to this procedure by the iom_manager is no longer the index by which the iom_manager identifies the channel. Instead, it is a number which the interrupt handler may use to reference its own databases. All programs called by the iom_manager for channel interrupts will have to be modified to account for this change.

The I/O Interfacer

The I/O Interfacer (IOI) makes use of the DST in several places. The first of these occurs during device interrupt handling. The DST entry for a device is referenced by ioi_interrupt to retrieve the process ID and event ID needed to send a wakeup to the process running the interrupting device. The process ID in the DST entry for a device also referenced on each call into IOI in order to validate that particular call.

IOI uses the DST and the IOAM for maintaining information about devices which share an IOM channel (or group of channels). Such devices might be tape or disk drives. IOM channels for such devices are assigned during Multics system initialization. Also during initialization, these devices are assigned device indices, their device names are made known to the IOAM, and a pointer to the IOI channel information block is squirreled away in the DST entry for each such device. When a user makes a call to attach such a device, this information allows the I/O Interfacer to associate the device name with a channel (or group of channels).

The I/O Interfacer differentiates between devices which share an IOM channel and those which do not by the length of the device name. Devices whose names are 4 characters or less (e.g., RDRA, PUNA, PRTA) do not share IOM channels. Those whose names are longer than 4 characters (e.g., TAPE_01, DSKA_05) do share. When ioi_assign is called to attach a device, if the name is longer than 4 characters, it is looked up in the I/O Assignment Table (IOAT) in order to determine its device index. This index is then used to reference the DST entry for the device in order to retrieve the pointer to the information pertaining to the channel(s) used to run that device. The device number is obtained by parsing the device name into two components. For example, "TAPE_03" is parsed into "TAPE" and "03".

IOI Changes

The I/O Interfacer will be modified to avoid all use of the DST and IOAM. The data block maintained by IOI for each attached device (called the device table entry) will be expanded from 12 to 16 words. Two of these words will contain the event ID used to send wakeups to the process "owning" the device. Another word will contain the process ID of the "owning" process. The fourth word will be reserved for future expansion. This change will allow the I/O Interfacer to perform call validation and device interrupt notification without referencing the DST.

The module ioi_assign will require some changes to facilitate the association of a device with a channel at device assignment time. The device name need no longer be defined anywhere. Channels which share devices will still be assigned during system initialization. In addition, the channel name must be defined in IOI's databases. This will be accomplished by adding a word (4 characters) to the structure which maintains information about an associated group of channels and devices (called the group table entry). ioi_assign parses any device name longer than 4 characters. It will be changed to look up the channel name by examining all group table entries.

The changes described above remove the last vestige of IOI's dependence on the DST and IOAM. They also simplify the initialization software for tape and disk subsystems. The modifications described above are completely internal to the I/O Interfacer. No external calling sequences are changed.

The TTY DIM

The hardcore TTY software uses both the DST and IOAM. As in the I/O Interfacer, the DST is used for user ring notification of TTY interrupts. IOAM is used for maintaining device attachment information. The TTY DIM will be changed to maintain the process and event ID's in its own internal tables and to call pxss directly for interrupt notification. At this time, however, it will not be changed to remove dependence on IOAM. Thus, it will still maintain DST device indices. But, the DST will no longer be used by the TTY DIM at interrupt time, and thus, will not be required to be wired.

Other Hardcore I/O Subsystems

Any other I/O subsystems in the hardcore will be modified to avoid use of the DST wherever possible. All such subsystems (e.g., the operator's console DCM and the IMP DIM) will be required to maintain their own process ID and event ID. Use of IOAM may still be continued, if needed. When all such subsystems have been modified, the DST may be unwired.