

TO: J. F. Ossanna (BTL)  
V. A. Vyssotsky  
FROM: Robert R. Fenichel  
SUBJECTS: (1) Reserver--I/O System Interface  
(2) Registry Files  
DATE: November 17, 1966

I have attached my draft documents on the subjects in question. As you suggested, I have called the second one BF.3. However, I notice that my BTABLE (10/20/66) has BF.3 full of DIM's. I trust that you will sort this out.

I also trust that you will let me know if I have distorted our discussion of these matters.

cc: F. J. Corbató  
E. L. Glaser  
R. M. Graham  
J. H. Saltzer ✓  
D. R. Widrig  
L. J. Lambert (GE)

*Access control  
Printed output gadgets  
Metering  
Meters  
+ Prices, etc.*

BT-1.00

Identification

## Overview of Registry Files

Robert R. Fenichel, V. A. Vyssotsky, J. F. Ossanna

Purpose

One directory in the file system is devoted to Registry Files. Each of these files is associated with a single physical frame. For an explanation of frames, see BF.1.10.

The file is the single internal repository of information regarding the disposition and, indeed, the persistence of the frame.

Summary of documentation

All manipulation of the Registry Files is accomplished by the Registry File Maintainer (RFM) program. The format of the Registry Files, consequently, is of no interest to non-implementers of the RFM.

For purposes related to prevention of inter-user conflict, the Reserver and I/O System make a number of calls to the RFM. These calls are described briefly in BT.3.1, and a complete description appears in BF.3.1.

The RFM will tell users of the physical dispositions of frames. Whose office is console #155 in? Which vault is reel #2603 in? Or is this reel away being stripped, or mounted on some drive? Calls related to physical disposition are described in BF.3.2.

A given frame may be described by several different names around the system. For example, a console may be known by its office location and by its extension number, and one RFM call (desc\_under) serves to tie these miscellaneous names to a unique frame index. While the frame indices will enable discovery of blunders into synonymy, their main function is as pointers to hardware routing information (GIOG channel numbers, etc.). RFM calls hinted at in this paragraph are discussed in BF.3.3.

Certain frames may be restricted to use by subsets of the user population. RFM calls concerned with access privileges are discussed in BF.3.4.

Hardware malfunctions will be logged in the Registry Files of

**DRAFT**  
date 11/17/66

electrical  
address?

the participating frame(s). For example, tape problems will each result in at least two entries: one for the reel and one for the drive. Intermittently, engineers may use special entries to scan the Registry Files for frames which have been present at too many accidents. RFM calls concerned with malfunction logging are described in BF.3.5.

**DRAFT**  
date \_\_\_\_\_

IdentificationReserver-I/O System Interface

Robert R. Fenichel, V. A. Vyssotsky, J. F. Ossanna

Purpose

In response to user requests the Reserver fills in its schedule of anticipated use of I/O frames. When reservations become due, the reserver must help the user to capture the appropriate frames.

Frames are distributed by the Registry File Maintainer, which is generally accountable for all use of these items. The RFM is the bulletin board of communication between the Reserver and the I/O System, and this document is consequently much concerned with the RFM. A more complete description of the RFM appears in BF.3.

The relations described in the text are also shown in the diagram which forms the last page of this section.

Registry File Contents

Each physical frame is represented in its Registry File with at least the following information:

- (a) Process i.d. of overseer of current attacher, if any
- (b) Beginning (calendar time) of next reservation, if any is foreseen
- (c) If frame now attached to some user, is it reserved, or only attached at the attacher's risk?

General-purpose entries to RFM

The Reserver and I/O system deal with frames in terms of types and descriptions. A single frame may have several different descriptions, and the RFM must allow distant parts of the system to learn of synonymy which may obtain between descriptions. The RFM accepts

desc\_under(type, description, response, index)

and returns response and index. Unless response is zero, index is

DRAFT  
date 11/17/66

But you have  
RFM  
entries  
described here

Why not only in  
reserver in files?

Poling not  
intend to  
RFM.

What is a  
type  
What is a  
description

What kind of  
index is index?  
is it a file-use  
non-index?

the unique internal identification of the frame described.

An administrator may wish to force his own ideas upon the Reserver. It is easy enough for him to insure special entries into the Reserver's private data, but unless frames are released these new reservations will not be honored.

*Why?* Because frames may be under the control of idiosyncratic DIM's, there is no easy way to pry a frame from an unhelpful user. The administrator will be forced to arrange for automatic logout of the users who are attaching what he wants. Transmission of the automatic logout signal is not a problem of this section.

But how does the administrator know whom to log out? The RFM will accept the call

*specified?*

whohas(type, description, total, n, processid)

This call provides the caller with

- (a) The total number of owned instances of the specified frame, and
- (b) The processid of the overseer of nth frame's owner, where n is given in the call.

Using this processid, other information (e.g., user name) is obtainable.

#### Administrative Entry to Reserver

In order to prevent futile brief attachments, the administrator may set a time period of guaranteed minimum unpreempted time. He does this with the call

graceperiod(type, time)

Following this call, an attach for a device of this type will always result in a reservation of length time, if no reservation already exists.

*program or  
pattern was  
clear...*

Relevant calls, IOS to Reserver

An attach call to the I/O System will generally result in an iwant call to the Reserver. In response to

iwant(type, description, index, response)

the Reserver will use whohas to see if a frame filling this description is already owned by the caller.

(A) If there is such a frame, the response is set accordingly and the frame's index is returned.

(B) If there is no such frame:

- (1) ~~If the graceperiod~~ for devices of this type is non-zero, the Reserver simulates an attempted reservation for the period starting immediately and extending for the graceperiod. Following successful establishment of such a reservation, the iwant call is effectively restarted; the philosophy here is similar to that of implicit-reference fault-removal (BG.0).

- (2) If the graceperiod is zero, the iwant call is transformed into

allot(type, description, 0, index, response)

This call is explained later in this document.

A detach call to the I/O system results in an

imdonewith(index, response)

call which, for reasons of symmetry, is directed to the Reserver. The Reserver actually does absolutely nothing to this call, and it drops through to the RFM immediately. If the frame was attached or reserved by some other user, or reserved by this user, an error response is given. If the frame was attached but unreserved by this

*imdonewith everything  
needed for QWT + sm.*

user, his i.d. is unposted from it.

#### Relevant Calls, Reserver to RFM

When a user asks the Reserver for a reservation, he mentions a bracketing calendar period (like "This Thursday"). The main task of the Reserver is conversion of such data as "one hour Thursday" into two data, the reservation start time ("1100 Thursday") and the reservation stop time ("1200 Thursday"). In between the reservation start time and the reservation stop time the Reserver must see to it that the facilities being reserved do not get attached by someone other than the reserver.

This latter task ~~is~~ made difficult by two independent problems. In the first place, many or all reservable facilities may also be subject to use without reservation. The Reserver must be able to retrieve facilities which are presently in use by other users. These users may have their own DIMs, and only these DIMs will know how to release the devices in question. In short, the Reserver can retrieve devices only by an automatic logout of the process group now holding these devices.

Secondly, the Reserver must face the problem which arises because specific physical frames ("drive J") are subject to specific reservation (say, by engineers). Suppose, for example, that this drive is reserved for a period to begin at 1400. If an "any-drive" reservation starts at 1350, it probably should be honored with some drive other than drive J.

In the unusual case that a specific frame is to be reserved, the Reserver calls the RFM with

`iwillallot(type, description, time)`

Each of these calls supersedes all previous ones. If time is given as zero, the frame is listed in the Registry Files as not being threatened by specific reservation. When the RFM is asked to supply a frame from some pool, it chooses that frame whose next specific

*supersed*

reservation, if any, is most distant.

At the reservation start time, the RFM is called with an allot call. The general appearance of this call is

```
allot(type, description, reservebit, index, response)
```

Roughly speaking, the purpose and effect of this call is to post the current process group as the owner of some frame of the type and description shown. In addition, the bit in the Registry File is set to show that this frame is (1) or is not (0) owned by means of a reservation. The index is returned.

Actually, there are three cases, as described by the response returned.

- (a) No frame of this type and description is known.
- (b) No frame of this type and description is available. That is, all have been allotted with this or higher reservebit value.
- (c) Normal case. This breaks into
  - (1) Frame was free
  - (2) Frame was attached, but not reserved. The reservebit parameter was 1.

The previous attacher has been logged out, and the frame has been captured.

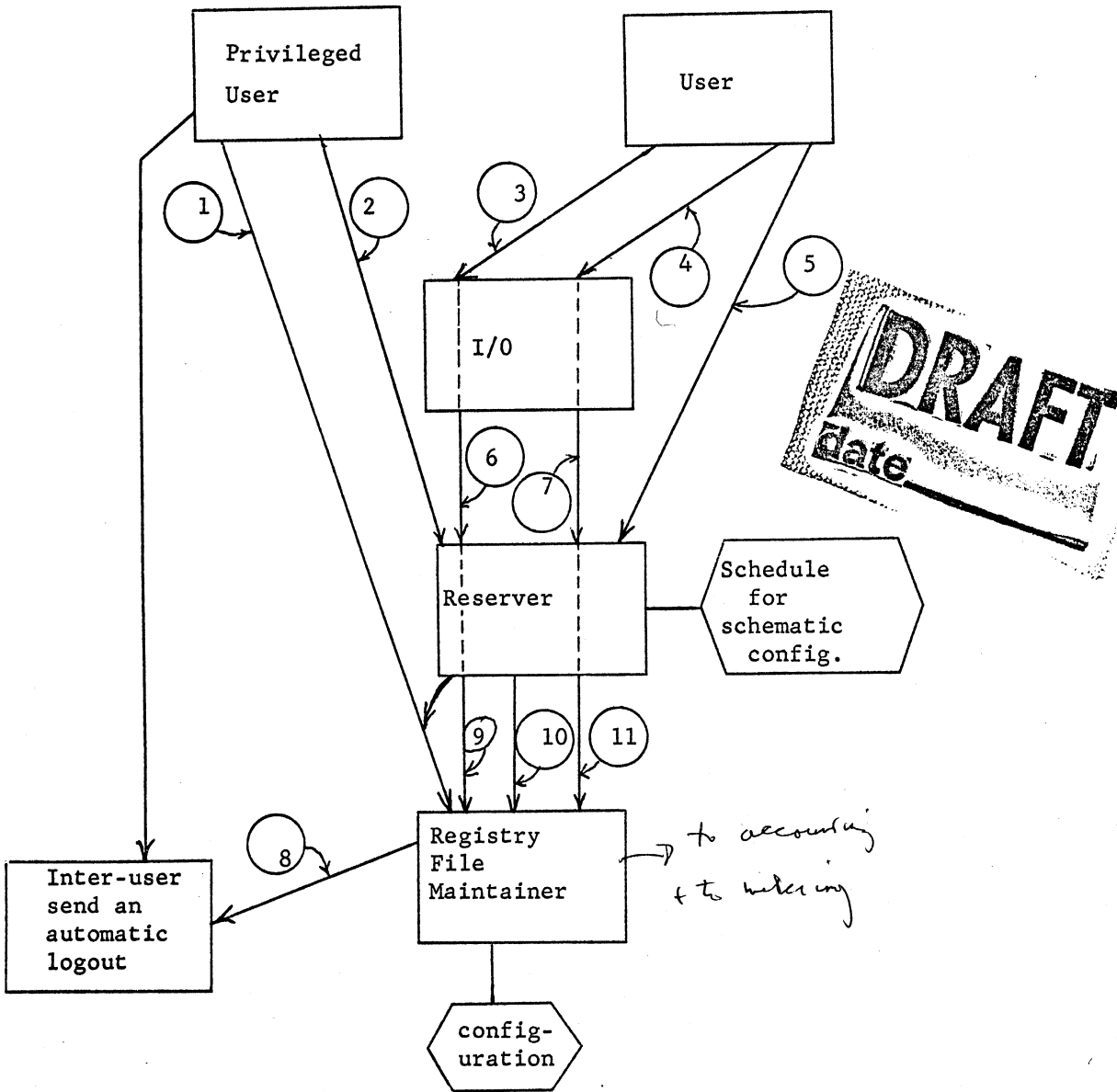
At the reservation stop time, the reserver makes one or more calls like

```
unallot(index)
```

The result of this call is to reset the indexed frame's status to "in use, but not reserved."

Does not take  
into account  
the G/M  
validation  
problem





- 1. Whohas
- 2. Graceperiod, privileged calls to force stuff into schedule
- 3. Attach
- 4. Detach
- 5. Pleaseallot, pleaseunallot
- 6. Iwant

- 7. Imdonewith
- 8. [Preemption in allot] ← by RFM?
- 9. Allot
- 10. Iwillallot, unallot
- 11. Imdonewith