D R A F T:
5/29/67

Identification

System Definition

J. H. Saltzer

Purpose

This section describes the definition of, and naming conventions for, the Multics operating system.

Discussion

A certain amount of organization is essential in the maintenance of an operating system, so that one may identify, for accounting and debugging purposes, "which" system is or was in use at any given time. In Multics, this organization is formalized in a three-level system-naming scheme, and a library registry. By identifying the three system levels ~~and~~ versions in use of each of and also the name and version of any public (and registered private) libraries, one can establish precisely what "total operating system" he is or was dealing with. When a large system may be dynamically updated by any of a number of system programmers, this formal name-and-registry control is a prerequisite for sanity.

We first discuss the three system levels of. A later section ~~will~~ describes the library registry.

System Levels

The various procedures of Multics fell naturally into one of three systems: the hardcore system, the administrative system, and the user

interface system. In addition, there may be any number of command and subroutine libraries, described later under library registry. The three systems mentioned above will, of course, come in many variations, both major and minor. Each of the three systems has at least a two-component name (two character strings) with the major component differing only between systems of significantly differing characteristics ~~with the major component~~ and the minor component a version identifier, changed whenever any part of that system is replaced or updated.

The hardcore system consists of modules concerned with ~~main~~board management of shared hardware - principally the basic file system, the traffic controller, and the GIOC interface modules.

The Hardcore System

The hardcore system has the property that a single copy is shared among all users of a single shared-equipment complex. (Note that independent, though electically ~~connected~~ connected, equipment may simultaneously support another copy of the same or different hardcore system.) The version of the hardcore system in use is determined by the Multics System Tape which was used to initialize the system, and by the system operator who selected one of several systems appearing on the Multics System Tape.

The name of the hardcore system is stored in a PL/I structure, declared as follows:

```
dcl 1 hardcore_system_name
    2 installation_code  char(24)  /* Unique name of installation running
                                      this Multics. */
    2 tape_label         char(24)  /* Label from Multics System Tape used
                                      to generate system. */
    2 major_name         char(24)  /* Name of the hardcore system. */
    2 version_name       char(24)  /* Name of the version in use. */
    2 time_loaded        bit(72)   /* Calendar time system was loaded. */
```

Thus a typical hardcore system name, if printed in a ~~reasonable~~ readable format, might be:

```
install:   MITMAC
tape:      MST74
name:      Multics/B
version:   196
load time: 1400.3, 6/27/68
```

If the same system were in operation at another installation, the installation name, tape label, and load time would be different, but the major and "name" and "version", minor system names, would be the same. A single installation might normally have two series of hardcore system names, one (e.g., Multics/B) to denote the publicly available system and a second (e.g., Multics/debug) to denote a test system ~~run~~ normally run for short periods on a separately configured section of the hardware.

It is essential to the operation of the scheme that the two names "name" and "version" uniquely identify the system in use. ~~and~~ A pair of names once used must never be used again to identify a different system.

The Administrative System

The administrative system consists of those system modules which, in the standard version, define process groups, handle login and logout, do resource usage accounting and administration, and interprocess communication. Although it is possible for several distinct users to operate with distinct administrative systems simultaneously using a single hardcore and hardware system, it is likely that at any given installation some single version of the standard Multics administrative system, named Multics_admin, would be used. The flexibility to discard Multics_admin and replace it with a completely different administrative system for some (or all) users, while provided by Multics, implies a considerable design job. Such flexibility would probably only be ~~involved~~ invoked by a Multics installation dedicated to some special application such as ~~airline~~ airline reservations.

As indicated before, the administrative system has a two component name, a major component (e.g., "Multics_admin/A") which changes only when major specifications are altered, and a minor component (e.g., "147") which changes whenever any part of the system is modified.