BQ.0.3

DRAFT
12.8.69

Identification

Account Group Administration

Michael J. Spier

An account represents the resources, in terms of dollars and secondary storage records, allocated to some user or perhaps to a group of users. An account is an entry in an Account Group Table (AGT) which describes the allocation distribution among members of an account group (normally members of a single project).

Associated with the AGT is the account administrator's Account Group Master File (AGMF) which is an ASCII fiel containing a section per account. An AGMF may be modified through the use of a context editor. A dedicated command named "cv_agmf" is available for converting the symbolic AGMF into a binary AGT. Conversion takes place in the account administrator's user-process; the source AGMF is checked for possible errors (syntactical, logical or omission). If any error was found, it prints out adequate diagnostics (all errors are detected in a single pass), then prints out "FATAL ERROR. CONVERSION UNSUCCESSFUL". If no error was found, the command prints "CONVERSION SUCCESSFUL.",

The Multics account group is a collection of Multics accounts which share some common denominator, where each individual account normally represents the financial resources allocated to some user (or group of users). Two possible approaches for account grouping are a) financially oriented, i.e., all accounts which draw on the same financial source (budget), and b) administratively oriented, i.e., all accounts of a specific group of users. Other approaches are conceptually possible, however only the two methods stated above may be meaningfully applied under the present Multics system control implementation. The normal Multics administrative usage is to establish an account group for all users of a single project (b), but as projects are normally independently financed, the same account group also corresponds to accounts drawing on a single financial source (a). Normally, then, the terms "project" and "account group" are in some way synonymous; the account group table (AGT) normally resides in the project directory, and the project administrator and account group administrator are normally one and the same person.

Associated with the account group is a master account representing the total resources alloted to the account group; the sum of the resources of all component accounts within the group may not exceed the size of the master account. From the implementation point of view, a master account structurally resembles a normal account, the only difference is in the size of allocated resources. The resource allocated to an account is an amount of dollars which the user spends through system usage. System usage falls into two major categories,

a) charges for permanently allocated resources (registration fee, secondary storage etc.) which are time-dependent and not directly related to any specific user session, and b) charges for resources which are temporarily allocated to a specific user session (cpu usage, core usage, I/O device usage, terminal connect time etc.). It follows that accounting for type (a) charges should be done periodically whereas accounting for type (b) charges must be done during the user session itself, and consequently both types of accounting need be handled separately. Therefore, an account consists of two major items, the "quota" which is the money alloted for type (a) charges, and the "resource" which is the money allotted for type (b) charges. Furthermore, type (b) charges vary, depending upon the time at which the user session is held, in order to entice people to use the system less during "rush hours" (e.g., workdays from 8am to 6pm) and more during the less popular "slack hours" (e.g., nights, weekends, holidays). The price discrepancy may be in the order of several hundred percent, and in addition the absolute price of execution (in terms of cpu time) may be lower during "slack hours" due to less competition among users (less page/segment faults, less process unloading/deactivating etc.). As a result of this pricing policy, a given task (in the sense of "user-project") may cost n dollars if carried out during the most expensive shift, and only n/k dollars (where k may be non-trivial) if carried out during a less expensive shift. The thrifty account manager may well wish to capitalize on this and allocate to that task only the smaller amount of money; such allocation is, however, meaningless unless the user is restricted to using only the less expensive shift (if he were admitted into the most expensive shift, he would only accomplish 1/k of his task before his money ran out). Therefore, in the implementation, type (b) charge allocations are expressed in terms of per-shift sub-allocations.

To summarize, an account (be is a master-account or a regular account) represents a certain amount of allocated money (say $1000) which is paid for by some financial source. The account's administrator first divides the money into (a) and (b) charge allocations (say $500 for permanently-allocated resources and $500 for temporarily-allocated resources), then proceeds to subdivide the (b) charge allocation into per-shift sub-allocations (say $200 for shift-1, $200 for shift-2, and $50 each for shifts 3 and 4). In the present accounting control implementation, an account consists of 5 allottments: a quota and four resource allotments corresponding to shifts 1-4 as indicated by the example above.

Temporarily-allocated (type (b)) resources are dynamically managed by the system, on line, and awarded to users on a first-come first-served basis. Permanently-allocated resources (type (a)) must be ditributed among users in advance, regardless of actual usage. Of these resources, secondary storage record capacity is finite, and the amount distributed must not exceed the system's capacity. Therefore, associated with an account's quota (dollars) there is a record quota (number of records) allocated to that account. Record usage is charged against the money quota, and a special charge is made for the record quota reservation, regardless of whether or not the entire quota is actually used.

Accounts are further divided into two parts: a) the allotment, and b) metered usage. All system usage is metered and charged, as explained above, against the account's allotment. When usage equals or exceeds the allotment, an error condition is established which results in one of the two following actions: a) if the excess was in the record quota, i.e., user is still financed and is only of trying to access more than his allocated share of the system' resources, then an error condition is signalled within his process, but b) if the excess occured in an actual money allotment, then the user is no longer financed and an "out-of-funds" condition is signalled to the user overseer which terminates the user session. The out-of-funds condition applies only to the actually depletes allotment, i.e., if a user's shift-1 allotment ran out and his session has been automatically logged-out because of it, he may still login later in the day on his valid shift-2 allotment.