2/28/68

TO:

R. C. Daley	J. F. Ossanna
✓ J. H. Saltzer	P. A. Belmont
K. J. Martin	H. J. Hébert
T. H. Van Vleck	R. J. Sobecki
	P. Schicker

FROM: C. Marceau

Here is a first cut at specs for error handling in system processes. Any suggestions you can offer, on the basis of your experience or intended use of this module, would be greatly appreciated.

Maybe should try harder when retry | error gets an error —
e.g., Try to reset the operation.
Trouble is a last resort.

Identification

Error handling in System Processes

ring_1_error

C. Marceau

Purpose

In user working processes, system modules detecting an error can inform the user of the error by using the standard mechanism described in BY.11.00. Of course all hard core errors are handled in a special way (trouble) ~~(panic).~~ This section describes the way in which system processes executing in the administrative ring handle errors.

Usage

When an administrative ring procedure, which may be executing in a system process detects an error condition, it responds by making a call to one of the entries of the ring_1_error procedure.

1) call ring_1_error$non_fatal (status, status_type);

dcl status_type char (*);

status may be declared in any way desired.

If status_type="bit" then status is interpreted as a bit string. If status_type="fixed", status is assumed to be fixed bit (17). "float" and "char" are also possible values, although presumably rarely invoked.

Ring_1_error$non_fatal records in the trouble log the segment name of its caller and the location of the call, as well as its process id, process group id, and the status provided by its caller. After depositing this

information in the trouble log, in a "non-fatal error" entry, ring_1_error returns to its caller.

2) Ring_1_error$ring_1_error (status, status_type) takes the same action as its non-fatal entry, except that it doesn't return. After recording the problem in the trouble log as a "fatal" error ring_1_error finds out what process-group it is executing in. If it is in the overseer of a user process-group it calls the abort entry of the overseer procedure, which automatically logs out the user, saving his work if possible. In a user working process ring_1_error signals a disaster to the overseer, using the interprocess communication facility. Thus disaster in a user process group affects only that user.

If ring_1_error is called in a system process group, the system must be shut down, or at least cease operation for a while. So ring_1_error signals the disaster signal to system control. System control causes I/O to cease and causes all absentee jobs to be suspended. It then informs the system operator of the trouble. After a time the operator may decide to shut down the system. Or, when the trouble is repaired, the operator may tell system control to resume normal operations.

Handling of errors within ring_1_error

It may happen that ring_1_error itself discovers "impossible" conditions. Its reaction is to try everything 3 times. If the 3rd try is unsuccessful (e.g. in its attempt to signal the overseer), it calls the ring 0 procedure ~~panic~~ trouble to halt all operations immediately since a thorough clean up is not feasible.

protected entry 7-1