

MCR10184

FAST Process Overseer Reports an Error at Logout

Revision 1.0

Eric Swenson

October 10, 2026

1 Problem Description

[Ticket #136](#) describes an issue where a user logged in with the FAST process overseer (`login USER -po >unb>fst_process_overseer_`) gets an error message on logging out:

```
Error: finish condition by logout$|623
(>system_library_standard>bound_as_requests_)
```

It does not happen when FAST is run from an ordinary process, nor with the DFAST process overseer.

1.1 Cause

`logout` signals `finish` so that programs can clean up before the process is destroyed; it is not an error. `fst_process_overseer_` establishes an `any_other` handler (`any_other_handler`, via `condition_`) that calls `condition_interpreter_` and prints the resulting message for every condition it receives, and only afterwards checks for `finish`:

```
    call condition_interpreter_ (addr (area), message_ptr, message_len, 1, mcptr,
                                cond_name, wcptr, info_ptr);
    ...
    call iox_$put_chars (iox_$user_output, message_ptr, message_len, code);
end;

if cond_name = "finish" then return;
```

So the `finish` signalled by `logout` is reported as an error. The DFAST overseer uses `standard.default_handler_`, which passes `finish` on, so it is not affected.

2 Proposed Fix

any_other_handler passes finish on at the start, without printing anything, as default_error_handler_ does:

```
cpa [lpn fst_process_overseer_.pl1] ==
```

Inserted in B:

```
B278          if cond_name = "finish" then do;          /* signalled by logout so
    programs can clean up; */
B279          cont = "1"b;          /* not an error, so print
    nothing and let the */
B280          return;          /* other finish handlers
    run (ticket 136) */
B281          end;
B282
```

Preceding:

```
A278          if cond_name = "command_error" |
```

```
A314          if cond_name = "finish" then return;
```

```
A315
```

Deleted by B, preceding:

```
B319          if info_ptr ^= null
```

Comparison finished: 2 differences, 7 lines.

The later `if cond_name = "finish" then return;` is removed. Passing the condition on (rather than returning as handled) lets any other finish handlers run; nothing else changes for other conditions.

3 Documentation

No documentation changes.

4 Test

1. Log in with `-po >unb>fst_process_overseer_`, use FAST, and log out (`logout` and `logout -hold`): no error message.
2. In FAST, run a program that takes a fault: the error message and the “Do you wish to continue the user program?” query appear as before.
3. Running `fast` from an ordinary process is unchanged.

5 Files Changed

bound_fast..s.archive	fst_process_overseer..pl1
bound_fast..archive	fst_process_overseer_ (recompiled)
>unb	bound_fast_ (rebound)

6 Version History

2026-10-10	1.0	Initial version of the MCR.
------------	-----	-----------------------------