Identification

System Operator Command to Delegate Responsibility
delegate
W. R. Strickler

Purpose

The System Operator delegates operator responsibility
to another operator by issuing the delegate command.
"Responsibility" may be one or more operator functions
(e.g., "media") or all those functions assigned to the
System Operator (in which case the System Operator is
relieved by another operator).

Usage

delegate name (function_1 -function_2- ...)

where

name is that of the operator to whom
responsibility is being delegated;

function_i is either "system" (new operator is
made responsible for all functions
assigned to present System Operator),
or the name of another function.

General Rules for Implementing System Operator Commands

Commands invoked in the System Operator's working process
for the performance of the system function communicate
with System Control. The System Operator's process-group
and System Control share several common data segments.

The segment "operator_comm" contains the structure

dcl 1 op_comm based (p),

2 sys_ctl_pid bit (36),           /*System Control
   process ID*/

2 op_rep_chn bit (70),          /*Name of event
   channel for signalling
   operator presence*/

2 op_req_chn bit (70);         /*Name of event
   channel for signalling
   a system function
   request*/
The segment "request_name" contains the structure:

```
  dcl 1 op_req based (rp),
    2 req_name char (32), /*Name of a request (command) belonging to the system function*/
    2 ref_chn bit (70);/*Name of channel over which System Control signals completion of the request*/
```

These segments contain information needed by a System Operator command to signal System Control and to receive a reflection signal from System Control. Any further information defining a particular request is placed in a shared data segment whose name is the same as that of the corresponding System Operator command; each such data segment is a branch in the request directory of System Control.

In general, all System Operator commands observe the following set of rules in communicating with System Control.

1. Place the name of the request (command) in `rp->op_reg->req_name`.
2. Create an event channel over which System Control can reflect completion of the request. Place the name of the channel in `rp->op_req->ref_chn`.
3. Place any information needed to further define the request in the data segment for the command in the request directory of System Control.
4. Signal the request to System Control over the channel named in `p->op_comm->op_req->chn`. Check status; if signal not sent, inform the operator and go to step 7.
5. Call the Wait Coordinator to wait for the reflection signal from System Control.
6. On wakeup, check status of the request. If any irregularities have occurred in fulfilling the request, System Control will have indicated them in the data segment for the command in the request directory of System Control reserved storage. In the case of any irregularity, the operator is informed; it is left up to him to determine the seriousness of irregularities and to decide how to correct them.
7. Delete the event channel created in step 2, and return.

Implementation of delegate

The procedure for the delegate command takes the steps outlined above. The segment "delegate" in the System Control request directory contains the structure:

```plaintext
dcl 1 deleg_args based (dp).

2 op_name char (24), /*operator to receive function(s)*/
2 n_fcns fixed bin (17), /*number of functions being delegated*/
2 function (dp deleg_args.n_fcns) char (32), /*name of function*/
2 state fixed bin (17), /*status returned by System Control*/
2 info char (64); /*descriptive info if error in delegation*/
```

Briefly, the procedure does the following:

1. Place "delegate" in rp→op_req.req_name.
2. Create an event channel and place its name in rp→op_req.ref_chn.
3. Place the argument name in dp→deleg_args.op_name.
4. Set dp→deleg_args.n_fcns, and, for each i, place the argument, function_i, in dp→deleg_args.function(i).
5. Signal System Control over the channel named in p→op_comm.op_req_chn, and then call the Wait Coordinator.
6. On wakeup, if dp→deleg_args.state = 0, go to step 7. Otherwise call write_out (BY.4.02) with the argument dp→deleg_args.info.
7. Delete the event channel created in step 2 and return.