MCR10215
exercise_disk -count and progress display
Revision 1.0

Eric Swenson

November 24, 2026

1 Problem Description

Ticket #246 requests enhancements to the `exercise_disk` program to allow shorter test runs to be made and to add a progress bar. Currently, the program's random test runs until 10,000,000 sectors have been written and read, which takes about six and a half hours. There is no way to run a shorter test and no indication of progress. (The ticket had an attached implementation, but the attachment is no longer on the Trac server: "Attachment 'exercise_disk.pl1' not found").

2 Proposed Fix

- New control argument `-count N` (`-ct N`): the random test stops after N sectors have been written and read, counted as before (each pass writes two sectors and reads them back, adding 4). N must be a positive decimal integer; the default stays 10,000,000. (The ticket proposed `-loops` but noted that the name is wrong, since the value counts reads and writes, not loops.)
- The counter `seek_counter` is now `fixed bin (35)`. It was declared `fixed bin` (17 bits) although it counted to 10,000,000. That worked only because a word holds 35 bits.
- Progress: internal procedures `start_progress` and `show_progress` print "0%" when the test starts, a dot for every 2% and the percentage at every 10%, as in the ticket's example, followed by a new line at the end. This is done for the random test (fraction of `-count`) and for the sequential test (fraction of the sectors on the pack). The range test (`-from`/`-to`) runs until stopped, so it has none.

A `compare_ascii` of the current and proposed `exercise_disk.pl1` follows:

```
Inserted in B:
B101            dcl     ioa_$nnl            entry options (variable);
Preceding:
A101
A102            dcl     cv_dec_check_       entry (char (*), fixed bin (35)) returns
    (fixed bin (35));
```

```
A124              dcl    (a, seek_counter)    fixed bin;       /* counter) for the write
    read pass */
Changed by B to:
B125              dcl    a                    fixed bin;
B126              dcl    seek_counter         fixed bin (35);  /* sectors read + written
    in the random test */
B127              dcl    max_count            fixed bin (35);  /* stop the random test at
    this many (-count) */
B128              dcl    progress_mark        fixed bin;       /* progress marks printed
    so far (0 to 50) */


Inserted in B:
B301              max_count = 10000000;                        /* a nominal number (the
    old fixed limit) */
Preceding:
A297
A298              if argcnt > 2 then /* process control args if they exist */


Inserted in B:
B415                        else if arg = "-count" | arg = "-ct" then do; /* size of the
    random test */
B416                                a = a + 1;
B417                                call cu_$arg_ptr (a, argptr, arglen, code);
B418                                if code ^= 0 then do;
B419                                         arg_sw = "0"b;
B420                                         go to bad_arg;
B421                                     end;
B422                                max_count = cv_dec_check_ (arg, code);
B423                                if code ^= 0 | max_count < 1 then do;
B424                                         call com_err_ (0, myname, "The -count value
    must be a positive decimal number: ^
\ca", arg);
B425                                         return;
B426                                     end;
B427                                end;
B428
Preceding:
A410                        else if arg = "-system" | arg = "-sys"


A563                        do while (seek_counter < 10000000); /* a nominal number */
Changed by B to:
B582                        call start_progress;
B583                        do while (seek_counter < max_count);

A581                        end;                                 /* seek_counter now max */
Changed by B to:
B601                             call show_progress (seek_counter, max_count);
B602                        end;                                 /* seek_counter now max */
B603                        call ioa_ ("");                      /* end the progress line */
```

```
A584              else if seq_sw then
Changed by B to:
B606              else if seq_sw then do;
B607                   call start_progress;


A600                   end;                                    /* end of a sequential pass
    */
Changed by B to:
B623                       call show_progress (x + 2, (highest_sector));
B624                   end;                                    /* end of a sequential pass
    */
B625                   call ioa_ ("");                         /* end the progress line */
B626              end;


Inserted in B:
B1042    /* Progress display: "0%" when the test starts, then a dot for every 2%
B1043       and the percentage at every 10%, so that one can estimate the time to
B1044       completion (ticket 246). */
B1045
B1046    start_progress: proc;
B1047
B1048             progress_mark = 0;
B1049             call ioa_$nnl ("0%");
B1050
B1051         end start_progress;
B1052
B1053
B1054    show_progress: proc (done, total);
B1055
B1056             dcl    (done, total)       fixed bin (35) parameter;
B1057             dcl    done_71             fixed bin (71);
B1058
B1059             done_71 = done;
B1060             do while (progress_mark < 50 & done_71 * 50 >= (progress_mark + 1) *
    total);
B1061                  progress_mark = progress_mark + 1;
B1062                  if mod (progress_mark, 5) = 0 then
B1063                       call ioa_$nnl ("^d%", 2 * progress_mark);
B1064                  else call ioa_$nnl (".");
B1065             end;
B1066
B1067         end show_progress;
B1068
Preceding:
A1016    EXIT: proc;

Comparison finished: 9 differences, 63 lines.
```

3 Documentation

The `exercise_disk.info` info segment, in >doc>`privileged` is updated to add a control argument of `-count N` (`-ct N`), and whose documentation says that the program ends after N sectors have been written and read. It also documents the 10,000,000 default value for N.

The comment about the test's taking several hours is augmented to reference the `-count N` control argument.

A new paragraph is added that documents the progress line.

A `compare_ascii` of the current and proposed `exercise_disk.info` follows:

```
cpa [wsp info exercise_disk.info] ==

A1       03/26/85 exercise_disk
Changed by B to:
B1       :Info: exercise_disk: 2026-11-24 exercise_disk
B2


Inserted in B:
B24      -count N, -ct N
B25         ends the random test after N sectors have been written and read
B26         (each pass writes two sectors and reads them back, counting 4).
B27         N is a positive decimal integer. The default is 10000000, which
B28         takes several hours.
Preceding:
A23      -device STR, -dv STR


A40         takes several hours. This is the default.
Changed by B to:
B46         takes several hours unless -count is given. This is the default.


Inserted in B:
B69      During a random or sequential test, a progress line is printed: "0%"
B70      when the test starts, a dot for every 2% of the test done, and the
B71      percentage at every 10%.
B72
Preceding:
A63      The assign_resource command must be used in conjunction with this

Comparison finished: 4 differences, 14 lines.
```

4 Test

1. `exercise_disk d501 PACK -dv dskb_01 -count 10000`: completes with the summary showing 5000 sectors written and 5000 read, and with a complete progress line.

2. `-sequential` on a small device: progress to 100%.
3. `-count`, `-count 0`, `-count x`: errors.
4. An error during the test (retry dialog) still works; the progress line resumes after the messages.

5 Files Changed

`bound_io_tools_.s.archive`	`exercise_disk.pl1`
`>tools`	`bound_io_tools_` (rebound)
`>doc>privileged`	`exercise_disk.info`

6 Version History

2026-11-24	1.0	Initial version of the MCR.