Data Monitoring for PowerMAX OS

Version 5.1 Release Notes

May 2008

0890493-5.1

concurrent

Copyright

Disclaimer

The information contained in this document is subject to change without notice. Concurrent Computer Corporation has taken efforts to remove errors from this document, however, Concurrent Computer Corporation's only liability regarding errors that may still exist is to correct said errors upon their being made known to Concurrent Computer Corporation.

License

Trademark Acknowledgments

MAXAda, PowerMA OS, NightTrace, NightView, NightBench, and NightStar are trademarks of Concurrent Computer Corporation.

Contents

1.0. Introduction

Data Monitoring provides Ada, C, and Fortran callable services for non-intrusive monitoring of variables in executing processes.

Data monitoring is effected through the use of the **`usermap`** service, which employs the **`/proc`** file system (**`proc(4)`**) to read and write the address space of executing processes. Data monitoring services require symbolic information from the executable files of the processes to be monitored; thus, portions of the executable files must be built with the **`-g`** (debug) compilation option (C, Fortran, and Ada).

This version of Data Monitoring supports monitoring programs built with the following compilers:

- Concurrent MAXAda
- Concurrent Fortran
- Concurrent C/C++

Data monitoring includes the following general capabilities:

- Retrieve values of variables (or components of variables) in a target process
- Modify variables (or components of variables) in a target process
- Retrieve/modify user-specified memory locations in a target process
- Retrieve information about variables (or components of variables) in a target process (type, address, dimensions, constraints, etc.)
- List the components of a composite variable (e.g. record, structure, array)
- Scan an executable file for variables that can be monitored

The following variables are eligible for monitoring:

- Variables in library-level Ada packages (including nested packages)
- C variables whose storage class is `static` or `extern`
- Fortran variables within subroutines
- Fortran common block members

The following variables are **not** eligible for monitoring:

- Variables allocated on a program stack

  Examples include Ada variables within subprograms, C variables with storage class `auto`, and `procedure`, `function`, and `subroutine` parameters
- Variables whose base address cannot be statically determined
- Elements of array variables whose offsets are variable (for example, array[variable])

Values are expressed in symbolic formats appropriate for their respective variables. For example, Ada variables of enumerated types are expressed in terms of their enumeration image rather than their underlying integer representation. In addition to the services that return and expect values expressed in symbolic format, a low-level interface that reads and writes variables without symbolic formatting is provided.

The C and Fortran interface services are available in a single library, **`/usr/lib/libdm.a`**. The Ada interface is available in the MAXAda™ environment **`/usr/ada/default/rtdm`** and is shipped with the MAXAda product.

2.0. Documentation

Table 2-1 lists the Data Monitoring 5.1 documentation available from Concurrent.

Table 2-1. Data Monitoring Version 5.1 Documentation

Manual Name	Pub. Number
Data Monitoring Reference Manual	08900493-020
Data Monitoring for PowerMAX OS Version 5.1 Release Notes	0890493-5.1

Copies of the Concurrent documentation can be ordered by contacting the Concurrent Software Support Center. The toll-free number for calls within the continental United States is 1-800-245-6453. For calls outside the continental United States, the number is 1-954-283-1822.

Additionally, the manuals listed above are available:

- in PDF format under **`/usr/lib/datamon`** (Reference Manual only).
- at http://redhawk.ccur.com/docs.

Detailed man pages are also available, **`datamon(3x)`**, **`datamon(3f)`**.

3.0. Prerequisites

Prerequisites for Data Monitoring Version 5.1 for both the host system and target system are as follows:

3.1. Host System

3.1.1. Software

- PowerMAX OS version 4.3 or higher

3.1.2. Hardware

- any system supported by the host operating system

3.2. Target System

3.2.1. Software

- PowerMAX OS version 4.3 or higher

3.2.2. Hardware

- any system supported by the target operating system

4.0. System Installation

The Data Monitoring product is installed as a standard PowerMAX OS software package and utilizes the standard PowerMAX OS product installation mechanism, **pkgadd** (see **pkgadd(1)**).

The package name is **datamon**. This name is case-sensitive.

Please refer to the “Installing Add-on Software” chapter in the *System Administration Volume I* (0890429) manual for instructions on software installation.

4.1. Changes in this Release

Data Monitoring 5.1 is primarily a maintenance release. Additionally it includes the following new features:

- Allows monitoring of variables which have multiple locations described in their debug description (location lists) as long as there is only a single location referencing static memory. Some compilers temporarily operate on a copy of the memory location in machine registers before restoring the updated value to memory. All monitoring of such variables is done solely with the memory location.
- Array stride size is now available in `object_descriptor_t` structures in the `od_stride_bit_size` component. An array's stride is the size of the array component plus any padding required to reach the next component. This most often occurs in Ada records that have alignment requirements which exceed the minimal size of the structure. For example, on AMD64 systems, the following record minimally requires 9 bytes, but the distance between components of an array of such records is 16 bytes because the record requires 8-bytes alignment:

```
type t is
   record
      x : long_float;
      y : boolean;
   end record;
type vector is array (1..10) of t;
```

  An `object_descriptor_t` for a variable of type vector would have the following values:

```
desc.od_component_bit_size == 72
desc.od_stride_bit_size    == 128
```

- A new function, `dm_find`, has been added to the API. It operates essentially the same as `dm_list`, but allows association of an opaque user-defined context which is passed to the callback routine. See `/usr/include/datamon.h` for the specification of `dm_find`.

5.0. Direct Software Support

Software support is available from a central source. If you need assistance or information about your system, please contact the Concurrent Software Support Center at our toll free number 1-800-245-6453. For calls outside the continental United States, the number is 1-954-283-1822. The Software Support Center operates Monday through Friday from 8 a.m. to 5 p.m., Eastern Standard Time. You may submit a request for assistance at any time by using the Concurrent Computer Corporation web site at http://www.ccur.com/isd_support_contact.asp.

Calling the Software Support Center gives you immediate access to a broad range of skilled personnel and guarantees you a prompt response from the person most qualified to assist you. If you have a question requiring on-site assistance or consultation, the Software Support Center staff will arrange for a field analyst to return your call and schedule a visit.