

linux-proc-apm.scm

Linux /proc/apm APM Data Access in Scheme

Version 0.2
2005-04-08

Neil W. Van Dyke
neil@neilvandyke.org

<http://www.neilvandyke.org/linux-proc-apm-scm/>

Copyright © 2004 - 2005 Neil W. Van Dyke. This program is Free Software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This program is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose. See the GNU Lesser General Public License [LGPL] for details. For other license options and consulting, contact the author.

1 Introduction

This Scheme library is used to access Linux APM (Advanced Power Management) power information. It can be used for reporting battery status information, for monitoring battery charge and taking action when the charge is low, for ensuring that a laptop is on line power before performing a disk-intensive batch job, etc.

This library works by parsing the string format of the `/proc/apm` file interface. Information about the format was gleaned from the Linux kernel source files `[apm.c]` and `[apm_bios.h]`, and from the usermode programs of `[apmd]`. It does not support ACPI, nor is it a more generalized power data interface.

This library is currently slightly specific to PLT Scheme, but was written in such a manner as to make easy the porting to other Scheme implementations.

2 Data Format

The `linux-proc-apm-data` abstract data type can be thought of as having nine attributes, with the accessors described in this section. Unless specified otherwise in the examples, `d` is sample APM data, such as might be yielded by `(define d (linux-proc-apm-data))`.

`linux-proc-apm-data:driver-version data` [Procedure]
`linux-proc-apm-data:bios-version data` [Procedure]

Yield the APM kernel driver version number and the APM BIOS version number, respectively, as a strings.

```
(linux-proc-apm-data:driver-version d) ⇒ "1.16"  
(linux-proc-apm-data:bios-version d) ⇒ "1.2"
```

`linux-proc-apm-data:apm-flags` *data* [Procedure]
Yields the APM flags as a list of any subset of the symbols `bits16`, `bits32`, `idle-slows-clock`, `disabled`, and `disengaged`. For example:

```
(linux-proc-apm-data:apm-flags d) => (bits16 bits32)
```

`linux-proc-apm-data:ac-line-status` *data* [Procedure]
Yields the AC line power status as the symbol `off`, `on`, or `backup`, or `#f` if unknown. For example:

```
(linux-proc-apm-data:ac-line-status d) => off
```

`linux-proc-apm-data:battery-status` *data* [Procedure]
Yields the battery status as the symbol `high`, `low`, `critical`, `charging`, or `absent`, or `#f` if unknown. For example:

```
(linux-proc-apm-data:battery-status d) => high
```

`linux-proc-apm-data:battery-flags` *data* [Procedure]
Yields APM battery flags as a list of any subset of the symbols `high`, `low`, `critical`, `charging`, and `absent`. For example:

```
(linux-proc-apm-data:battery-flags d) => (high charging)
```

`linux-proc-apm-data:battery-percent` *data* [Procedure]
Yields the estimated battery charge percentage as an integer, or `#f` if unknown.

```
(linux-proc-apm-data:battery-percent d) => 99
```

`linux-proc-apm-data:battery-time` *data* [Procedure]
`linux-proc-apm-data:battery-time-units` *data* [Procedure]
This pair of procedures yield, respectively, the estimated remaining battery charge as an integer and a string representing the units. The units string is likely to be `"min"`. Either or both value can be `#f` if unknown.

```
(linux-proc-apm-data:battery-time d) => 335  
(linux-proc-apm-data:battery-time-units d) => "min"
```

`linux-proc-apm-data:kludged-battery-percent` *data* [Procedure]
Yields the estimated battery charge percentage as an integer, or `#f` if all fails. This works by first attempting to use APM's estimated percentage, but if that is unavailable, falling back to to a very crude fake percentage based on the `battery-status` or `battery-flags` attribute. This procedure is of questionable utility, yet may still find use in, say, a noncritical display of approximate battery charge.

```
(define d (parse-linux-proc-apm-string  
          "1.16 1.2 0x03 0x01 0x03 0x09 -1% -1 ?"))  
(linux-proc-apm-data:battery-percent d) => #f  
(linux-proc-apm-data:kludged-battery-percent d) => 90
```

3 Parsing

These procedures are concerned with parsing the data, and are not normally used directly.

`parse-linux-proc-apm-string` *str* [Procedure]

Yields the APM data parsed from string *str*, or `#f` if the string could not be parsed.

`parse-linux-proc-apm-file` *filename* [Procedure]

Yields the APM data from file *filename*, or `#f` if the data is unavailable (e.g., the file could not be accessed, or the data could not be parsed).

4 Data Access

The normal procedure for acquiring APM data is `linux-proc-apm-data`.

`current-linux-proc-apm-file` [Parameter]

Parameter for the file name of the default APM data file, defaulting to `"/proc/apm"`, surprisingly enough.

`linux-proc-apm-data` [Procedure]

Yields the APM data from the file given by the `current-linux-proc-apm-file` parameter, or `#f` if the data is unavailable.

5 Tests

The `linux-proc-apm.scm` test suite can be enabled by editing the source code file and loading `[Testeez]`; the test suite is disabled by default.

History

Version 0.2 — 2005-04-08

Added Testeez-based test suite. Minor documentation changes. Changed to `not-break-exn?` use to `PLT 3xx exn:fail?`.

Version 0.1 — 2004-08-03

Initial version.

References

[apm.c] <http://lxr.linux.no/source/arch/i386/kernel/apm.c?v=2.4.26>

[apm_bios.h]

http://lxr.linux.no/source/include/linux/apm_bios.h?v=2.4.26

[apmd] <http://www.worldvisions.ca/~apenwarr/apmd/>

[LGPL] Free Software Foundation, “GNU Lesser General Public License,” Version 2.1, 1999-02, 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

<http://www.gnu.org/copyleft/lesser.html>

[Testeez] Neil W. Van Dyke, “Testeez: Simple Test Mechanism for Scheme,” Version 0.1.

<http://www.neilvandyke.org/testeez/>