doc/VMMInformer

Torsten Ebeling

COLLABORATORS

	TITLE : doc/VMMInformer		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY	Torsten Ebeling	February 12, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

doc/VMMInformer

1.1   doc/VMMInformer.guide

```
                                     VMMInformer
           (Graphical statistics program for use with VMM)
                        User's Guide
                        Version 1.0
                  $Date:  95/05/11 22:23:42 $
                    written by Torsten Ebeling
              email:  t_ebelin@informatik.uni-kl.de
```

CONTENTS

```
   0.
                 Copyright & Author
                  1.
                 Introduction
                  2.
                 Installation
                  3.
                 VMMInformer
                  4.
                 Preferences file - Syntax
                  5.
                 Known bugs
                  6.
                 Future plans
                  7.
                 Acknowledgments
```

1.2   doc/NerveEX.guide/COPYRIGHT

0. COPYRIGHT and AUTHOR

 IMPORTANT NOTICE: This program is copyrighted by Torsten Ebeling, but
 can be freely distributed, provided that the following rules are

respected.

- No change is made to the program nor to the accompanying documentation
- The package is always distributed in its complete form.
- Every form of distribution is allowed and encouraged, but no fee can
  be charged for this program except for, possibly, the cost of magnetic
  media and/or disk duplication and shipping.
- Inclusion in PD software libraries such as Fish Disks is allowed,
  provided the fees charged for these disks are comparable with those
  charged by Fred Fish.
- The program cannot be distributed in any commercial product without
  the written consent of the author.
- The author cannot be held liable for the suitability or accuracy of or
  any damage directly or indirectly caused by the use or misuse of this
  program. It is the sole responsibility of the user her/him self.

By copying, distributing and/or using the program you indicate your
acceptance of the above rules.

Author:

  email: t_ebelin@informatik.uni-kl.de

  snail-mail: Torsten Ebeling
              Kottenschanze 8
              67659 Kaiserslautern
              Germany

1.3 doc/VMMInformer.guide/INTRODUCTION

1. INTRODUCTION

I'm really fed up with writing manuals. This manual will be very
short.   VMMInformer is a utility which displays information about VMM
either as text or graphic. That's it.

VMMInformer requires VMM 2.1 or greater to be installed and running on
your system. VMM is a virtual memory manager written by Martin Apel.
VMM is available on AMINET and several PD/Shareware Disks or CDs.

1.4 doc/VMMInformer.guide/INSTALLATION

             2. INSTALLATION

VMMInformer requires at least OS2.04 to run. VMMInformer also
requires VMM 2.1 or greater to be installed and running on your
system.

Execute the following steps to install VMMInformer:

- Copy

                        VMMInformer
                         to your preferred path for utilities.
– Copy VMMInformer.prefs to ENVARC: or generate your own preferences file.

Now change the preferences file at will.

I   added   three example preferences files to this archive in the prefs
directory.   Use   these   preferences   files   to get a first impression
about the possibilities of VMMInformer.
VMMInformer tries to read the file specified as argument in the
command line first. If you didn't specify a prefs-file it will try to
read ENV:VMMInformer.prefs and after that ENVARC:VMMInformer.prefs.

See
                        Prefs-File syntax
                         for more details on that.

1.5   doc/VMMInformer.guide/VMMINFORMER

                        3. VMMINFORMER

There is not much to explain about VMMInformer:

   TEMPLATE: VMMInformer [prefs-file]

– At the top of the window all strings with the selected variables are
  displayed.
– Below the strings the graphical display begins. If you have defined
  any graphics you can change the size of the window with the
  sizing-gadget, otherwise there is no sizing-gadget visible.

See
                        Prefs-File syntax
                         for more details on that.

1.6   doc/VMMInformer.guide/PREFS_SYNTAX

4. PREFERENCES FILE – SYNTAX

IMPORTANT: DON'T edit the first line in the preferences file.
IMPORTANT: The declaration blocks must be defined in the following
          order: 1. "TASK/WINDOW"
                  2. "VARIABLES"
                  3. "STRINGS"
                  4. "GRAPHIC"
IMPORTANT: The syntax of the preferences file will be checked
          but not all errors will be recognized by the parser.

****

Possible errors during preferences file parsing:

Error no.:   Description:

    1   out of memory
    2   couldn't open config file
    3   SYNTAX error
    4   undefined identifier or use out of BEGIN ... END
    5   argument missing
    6   ULONG argument missing: ULONG required
    7   BOOLEAN argument missing: "TRUE" or "FALSE" required
    8   quote mark missing
    9   invalid argument
   10   total memory failure in 2 minutes ;-)
   11   "ARRAY" is not a valid variable name
   12   invalid use of variable
   13   variable name not found
   14   invalid array offset
   15   no graphic defined
****

Description of the preferences file syntax.

"//" starts a comment line

<BYTE>   :
<UBYTE>   :
<UWORD>   :
<ULONG>   :
<BOOLEAN>   : "TRUE" oder "FALSE"
<STRING>   : Must be enclosed by quotes. I.e.: "test test test"
              The string must be entered in printf-style format. A
              maximum of 6 variables is allowed. In title strings only
              one variable is allowed.

<VAR>   : One of the variables defined below (except d).
<FUNCTION> : For the definition of functions you must use RPN-format.
              You can use the following variables for your calculations:

              a) Variables defined in the VMStat-structure:
                 - VMSize   : Size of virtual memory
                 - VMFree   : Size of free virtual memory
                 - Faults   : Page faults up until now
                 - PagesWritten : Pages written up until now
                 - PagesRead   : Pages read up until now
                 - NumFrames   : Actual no. of used frames
                 - PagesUsed   : Pages used on device
                 - PageSize   : Pagesize
                 - TrapStructsFree: No. of free TrapStructs

                 See "VMM.guide" for more details.

              b) 4 more variables concerning free memory and the time:

```
                    - PUBLIC_FAST_Free: Free PUBLIC FAST memory
                    - CHIP_Free       : Free CHIP memory
                    - CurrentSeconds  : Current time (seconds)
                    - CurrentMicros   : Current time (micros)

          c) All key variables:
             Only SECONDS and MICROS may be of interest. (see below)

          d) Constants: i.e. 1, 45, 100000, ...  (no negative values)

          e) All "ARRAY"-variables defined.

          f) All variables defined with "DEFINE" (functions).

             IMPORTANT: variables in e) and f) must be defined
                        before they are used!!!

          SYNTAX of functions:

          IMPORTANT: Only   ULONG    variables    are    used   for
                     calculations.  In  complex  calculations round
                     off  errors  can occur.  This doesn't mean your
                     processor   is   faulty  (INTEL  outside).  ALL
                     variables are stored as ULONG integers.

          All functions MUST be entered in RPN-format!

          The following operators are allowed: +, -, *, /, %

          Examples:
            i)   Size of free virtual memory:
                 - DEFINE VMUsed = VMSize VMFree -

            ii)  No. of pagefaults during an update cycle:
                 - ARRAYFaults = Faults
                 - DEFINE Faults_per_update = ARRAYFaults(0) ARRAYFaults(-1) -

            iii) No. of faults during 1 second:
                 - SECONDS = 20
                 - MICROS  = 500000
                 - ARRAYFaults = Faults
                 - DEFINE Update_Micros = SECONDS 1000000 * MICROS +
                 - DEFINE Faults_per_update = ARRAYFaults(-1) ARRAYFaults(0) -
                 - DEFINE Faults_per_second = Faults_per_update 1000000 *  ↩
                    Update_Micros /


--> START OF PREFERENCES FILE
***********************************************************************
"BEGIN  TASK/WINDOW"  starts  the  definition  of  the task and window
                     specific variables.

"VERSION = <UBYTE>"   Version of preferences file format.   DON'T EDIT
"REVISION = <UBYTE>"  Revision of preferences file format.  DON'T EDIT

"PRIORITY = <BYTE>"  Priority of VMMInformer task (-128 bis 127).
```

```
"LEFT = <UWORD>"      Left edge of window.
"TOP  = <UWORD>"      Right edge of window.
"WIDTH  = <UWORD>"    Width of window.
                      (The height depends on the contents of the window.)

"ZOOMED = <BOOL>"          Open window zoomed (TRUE) or unzoomed (FALSE).
"ZOOMED_LEFT = <UWORD>"  Left edge of the zoomed window.
"ZOOMED_TOP  = <UWORD>"  Right edge of the zoomed window.

"BACKFILLCOLOR = <UBYTE>"  Background color (pen no.)
"PANELCOLOR    = <UBYTE>"  Panel color      (pen no.)
"TEXTCOLOR     = <UBYTE>"  Text color       (pen no.)

"END" ends this definition.

**********************************************************************
"BEGIN VARIABLES" starts the declaration of the variables.

"SECONDS = <ULONG>"
"MICROS  = <ULONG>" Update the window every x seconds and y micros.

"ARRAY<ident>  =  "  Declaration  of  a  variable which can be used in
                     calculations  later.  All values of this variable
                     are stored in an array.  (Up to the actual screen
                     size).   <ident>  is  the  name of this variable.
                     The  name  must follow directly after the keyword
                     "ARRAY".
                     I. e.: "ARRAY0", "ARRAYVMFree", "ARRAYStarTrek", ...

                     Alphanumeric  and numeric characters are allowed.
                     You can access the last value stored in the array
                     as follows:
                      - ARRAY<ident>(-1)
                     The  value  in  parentheses  can  be  between the
                     current screen width - 1 and 0 (a negative number).


"DEFINE  <var> = <FUNCTION>" This defines a new variable which will be
                            recalculated  every  update  cycle.  <var>
                            is  the name of this variable.  After you
                            have defined this variable you can access
                            it  in  the  following calculations.  You
                            can't  access  a  variable  which has not
                            been  defined  before  it is used for the
                            first time.
"END" ends this declaration.

**********************************************************************
"BEGIN STRINGS" starts the definition of strings.

"TITLE = <STRING>[, <VAR>[, <VAR>]...]"    Window title
"ZOOMTITLE = <STRING>[, <VAR>[, <VAR>]...]" Window title in zoomed state.

"TEXT  =  <STRING>[, <VAR>]" Strings defined in this way are displayed
                            at  the  top  of  the  window in order of
                            their definition.  <STRING> is the string
                            to  be  displayed in the window and <VAR>
```

```
                                   is the linked variable. <VAR> is optional.
"END" ends this definition.

**********************************************************************
"BEGIN GRAPHIC" starts the definition of graphical displays.

"AUTOSCALE = <BOOLEAN>" TRUE  enables autoscale option.
                       FALSE disables autoscale option.
                       (see "EXCEED_COLOR").

"SHIFT  =  <UWORD>"  Shift the graphic <SHIFT> pixels to the left when
                    the right window border is reached.

"COLOR        = <UBYTE>" Default color for the graphical display. (pen no.)
"EXCEED_COLOR = <UBYTE>" Color which is used to mark that the value
                        exceeds the graphical display.

"GRAPHIC  =  <VAR>"  Variable  which  is  displayed  as  graphic.  The
                    graphics  will  be  displayed  one underneath the
                    other  after  all  strings have been displayed in
                    the window.

"GRAPHIC_COLOR = <UBYTE>"    Overrides the default pen no..
"GRAPHIC_HEIGHT = <UWORD>"   Overrides the default height.
"GRAPHIC_AUTOSCALE = <BOOL>" Overrides AUTOSCALE option.

   "GRAPHIC_AUTOSCALE", "GRAPHIC_COLOR" und "GRAPHIC_HEIGHT" are always
   only in reference to the previously defined graphic.

"END" ends this definition.

**********************************************************************
--> END OF PREFERENCES FILE
```

1.7 doc/VMMInformer.guide/TROUBLESHOOTING

```
5. KNOWN BUGS

This is the first release and there are no known bugs in this version.
This doesn't mean there aren't any unknown bugs.  If you find any bugs
please mail them to me.

If  VMM isn't able to start correctly for some reason, VMMInformer may
hang.   This  is  not an error but is due to VMMInformer waiting for a
response from VMM which isn't able to respond.
```

1.8 doc/VMMInformer.guide/PLANS

```
6. FUTURE PLANS

- Rework the preferences file syntax.
```

1.9 doc/VMMInformer.guide/ACKNOWLEDGMENTS

7. ACKNOWLEDGMENTS

I would like to thank Martin Apel for providing me with ideas and
helping me with problems.

7. ACKNOWLEDGMENTS