NAME

conky − A system monitor for X originally based on the torsmo code, but more kickass. It just keeps on given’er. Yeah.

SYNOPSIS

conky [*options*]

DESCRIPTION

Conky is a system monitor for X originally based on torsmo. Since its inception, Conky has changed significantly from its predecessor, while maintaining simplicity and configurability. Conky can display just about anything, either on your root desktop or in its own window. Not only does Conky have many built-in objects, it can also display just about any piece of information by using scripts and other external programs.

Conky has more than 250 built in objects, including support for a plethora of OS stats (uname, uptime, CPU usage, mem usage, disk usage, "top" like process stats, and network monitoring, just to name a few), built in IMAP and POP3 support, built in support for many popular music players (MPD, XMMS2, BMPx, Audacious), and much much more. Conky can display this info either as text, or using simple progress bars and graph widgets, with different fonts and colours.

We are always looking for help, whether its reporting bugs, writing patches, or writing docs. Please use the facilities at SourceForge to make bug reports, feature requests, and submit patches, or stop by #conky on irc.freenode.net if you have questions or want to contribute.

Thanks for your interest in Conky.

COMPILING

For users compiling from source on a binary distro, make sure you have the X development libraries installed (Unless you configure your build without X11). This should be a package along the lines of "libx11-dev" or "xorg-x11-dev" for X11 libs, and similar "-dev" format for the other libs required (depending on your build options). You should be able to see which extra packages you need to install by reading errors that you get from running ‘cmake’. The easiest way to view the available build options is to run ‘ccmake’ or ‘cmake-gui’ from the source tree, but be careful when disabling certain features as you may lose desired functionality. E.g., with BUILD_MATH disabled you won’t get errors but logarithmic graphs will be normal graphs and gauges will miss their line.

Conky has (for some time) been available in the repositories of most popular distributions. Here are some installation instructions for a few:

Gentoo users -- Conky is in Gentoo’s Portage... simply use "emerge app-admin/conky" for installation.

Debian, etc. users -- Conky should be in your repositories, and can be installed by doing "aptitude install conky".

Example to compile and run Conky with default components (note that some build options may differ for your system):

```
cmake −D CMAKE_INSTALL_PREFIX:string=/usr .

make

make install # Optional

src/conky
```

Conky has been tested to be compatible with C99 C and C++0x C++, however it has not been tested with anything other than gcc, and is not guaranteed to work with other compilers.

TIP: Try configuring Conky with ‘ccmake’ or ‘cmake-gui’ instead of just ‘cmake’.

YOU SHOULD KNOW

Conky is generally very good on resources. That said, the more you try to make Conky do, the more resources it is going to consume.

An easy way to force Conky to reload your ˜/.config/conky/conky.conf: "killall -SIGUSR1 conky". Saves you the trouble of having to kill and then restart. You can now also do the same with SIGHUP.

OPTIONS

Command line options override configurations defined in configuration file.

-a | --alignment= ALIGNMENT
Text alignment on screen, {top,bottom,middle}_{left,right,middle} or none. Can also be abbreviated with first chars of position, ie. tr for top_right. Only available with build flag BUILD_X11 enabled.

-b | --double-buffer
Use double buffering (eliminates "flicker"). Only available with build flag BUILD_X11 enabled.

-c | --config= FILE
Config file to load instead of $HOME/.config/conky/conky.conf

-C | --print-config
Print builtin default config to stdout. See also the section EXAMPLES for more information. Only available with build flag BUILD_BUILTIN_CONFIG enabled.

-d | --daemonize
Daemonize Conky, aka fork to background.

-D | --debug
Increase debugging output, ie. -DD for more debugging.

-f | --font= FONT
Font to use. Only available with build flag BUILD_X11 enabled.

-h | --help
Prints command line help and exits.

-i COUNT
Number of times to update Conky (and quit).

-o | --own-window
Create own window to draw. Only available with build flag BUILD_X11 enabled.

-p | --pause= SECONDS
Time to pause/wait before actually starting Conky.

-q | --quiet
Run Conky in ’quiet mode’ (ie. no output).

-t | --text= TEXT
Text to render, remember single quotes, like -t ’ $uptime ’

-u | --interval= SECONDS
Update interval.

−v | −V | −−version
Prints version, build information and general info. Exits after printing.

−w | −−window−id= WIN_ID
Window id to draw. Only available with build flag BUILD_X11 enabled.

−x X_COORDINATE
X position

−x X_COORDINATE
X position

−X | −−display= DISPLAY
X11 display to use. Only available with build flag BUILD_X11 enabled.

−y Y_COORDINATE
Y position

CONFIGURATION SETTINGS

Default configuration file location is $HOME/.config/conky/conky.conf or ${sysconfdir}/conky/conky.conf. On most systems, sysconfdir is /etc, and you can find the sample config file there (/etc/conky/conky.conf).

You might want to copy it to $HOME/.config/conky/conky.conf and then start modifying it. Other configs can be found at http://conky.sf.net/

alignment
Aligned position on screen, may be top_left, top_right, top_middle, bottom_left, bottom_right, bottom_middle, middle_left, middle_middle, middle_right, or none (also can be abreviated as tl, tr, tm, bl, br, bm, ml, mm, mr). See also gap_x and gap_y.

append_file
Append the file given as argument.

background
Boolean value, if true, Conky will be forked to background when started.

border_inner_margin
Inner border margin in pixels (the margin between the border and text).

border_outer_margin
Outer border margin in pixels (the margin between the border and the edge of the window).

border_width
Border width in pixels.

colorN
Predefine a color for use inside conky.text segments. Substitute N by a digit between 0 and 9, inclusively. When specifying the color value in hex, omit the leading hash (#).

console_graph_ticks

A comma-separated list of strings to use as the bars of a graph output to console/shell. The first list item is used for the minimum bar height and the last item is used for the maximum. Example: " ,_,=,#".

cpu_avg_samples

The number of samples to average for CPU monitoring.

default_bar_height

Specify a default height for bars. If not specified, the default value is 6.

default_bar_width

Specify a default width for bars. If not specified, the default value is 0, which causes the bar to expand to fit the width of your Conky window. If you set out_to_console = true, the text version of the bar will actually have no width and you will need to set a sensible default or set the height and width of each bar individually.

default_color

Default color and border color

default_gauge_height

Specify a default height for gauges. If not specified, the default value is 25.

default_gauge_width

Specify a default width for gauges. If not specified, the default value is 40.

default_graph_height

Specify a default height for graphs. If not specified, the default value is 25.

default_graph_width

Specify a default width for graphs. If not specified, the default value is 0, which causes the graph to expand to fit the width of your Conky window. If you set out_to_console = true, the text version of the graph will actually have no width and you will need to set a sensible default or set the height and width of each graph individually.

default_outline_color

Default outline color

default_shade_color

Default shading color and border's shading color

disable_auto_reload

Enable to disable the inotify-based auto config reload feature.

diskio_avg_samples

The number of samples to average for disk I/O monitoring.

display
Specify an X display to connect to.

xinerama_head
Specify a Xinerama head.

double_buffer
Use the Xdbe extension? (eliminates flicker) It is highly recommended to use own window with this one so double buffer won't be so big.

draw_borders
Draw borders around text?

draw_graph_borders
Draw borders around graphs?

draw_outline
Draw outlines?

draw_shades
Draw shades?

extra_newline
Put an extra newline at the end when writing to stdout, useful for writing to awesome's wiboxes.

font Font name in X, xfontsel can be used to get a nice font

format_human_readable
If enabled, values which are in bytes will be printed in human readable format (i.e., KiB, MiB, etc). If disabled, the number of bytes is printed instead.

gap_x Gap, in pixels, between right or left border of screen, same as passing -x at command line, e.g. gap_x 10. For other position related stuff, see 'alignment'.

gap_y Gap, in pixels, between top or bottom border of screen, same as passing -y at command line, e.g. gap_y 10. For other position related stuff, see 'alignment'.

hddtemp_host
Hostname to connect to for hddtemp objects. Defaults to "127.0.0.1".

hddtemp_port
Port to use for hddtemp connections. Defaults to 7634.

http_refresh
When this is set the page generated with out_to_http will automatically refresh each interval. Default value is no.

if_up_strictness

How strict should if_up be when testing an interface for being up? The value is one of up, link or address, to check for the interface being solely up, being up and having link or being up, having link and an assigned IP address.

imap Default global IMAP server. Arguments are: "host user pass [-i interval (in seconds)] [-f ’folder’] [-p port] [-e ’command’] [-r retries]". Default port is 143, default folder is ’INBOX’, default interval is 5 minutes, and default number of retries before giving up is 5. If the password is supplied as ’*’, you will be prompted to enter the password when Conky starts.

imlib_cache_flush_interval

Interval (in seconds) to flush Imlib2 cache.

imlib_cache_size

Imlib2 image cache size, in bytes. Defaults to 4MiB. Increase this value if you use $image lots. Set to 0 to disable the image cache.

lua_draw_hook_post function_name [function arguments]

This function, if defined, will be called by Conky through each iteration after drawing to the window. Requires X support. Takes any number of optional arguments. Use this hook for drawing things on top of what Conky draws. Conky puts ’conky_’ in front of function_name to prevent accidental calls to the wrong function unless you place ’conky_’ in front of it yourself.

lua_draw_hook_pre function_name [function arguments]

This function, if defined, will be called by Conky through each iteration before drawing to the window. Requires X support. Takes any number of optional arguments. Use this hook for drawing things on top of what Conky draws. Conky puts ’conky_’ in front of function_name to prevent accidental calls to the wrong function unless you place ’conky_’ in front of it yourself.

lua_load

Loads the Lua scripts separated by spaces.

lua_shutdown_hook function_name [function arguments]

This function, if defined, will be called by Conky at shutdown or when the configuration is reloaded. Use this hook to clean up after yourself, such as freeing memory which has been allocated by external libraries via Lua. Conky puts ’conky_’ in front of function_name to prevent accidental calls to the wrong function unless you place ’conky_’ in front of it yourself.

lua_startup_hook function_name [function arguments]

This function, if defined, will be called by Conky at startup or when the configuration is reloaded. Use this hook to initialize values, or for any run-once applications. Conky puts ’conky_’ in front of function_name to prevent accidental calls to the wrong function unless you place ’conky_’ in front of it yourself.

mail_spool

Mail spool for mail checking

max_port_monitor_connections

Allow each port monitor to track at most this many connections (if 0 or not set, default is 256)

max_text_width width
When a line in the output contains ’width’ chars and the end isn’t reached, the next char will start on a new line. If you want to make sure that lines don’t get broken, set ’width’ to 0

max_user_text bytes
Maximum size of user text buffer, i.e. text inside conky.text section in config file (default is 16384 bytes)

maximum_width pixels
Maximum width of window

minimum_height height
Minimum height of the window

minimum_width width
Minimum width of window

mpd_host
Host of MPD server

mpd_password
MPD server password

mpd_port
Port of MPD server

mysql_host
Host of MySQL server. Defaults to localhost

mysql_port
Port of MySQL server. Defaults to the default mysql port

mysql_user
MySQL user name to use when connecting to the server. Defaults to your username

mysql_password
Password of the MySQL user. Place it between "-chars. When this is not set there is no password used

mysql_db
MySQL database to use. Defaults to mysql

music_player_interval
Music player thread update interval (defaults to Conky’s update interval)

net_avg_samples
The number of samples to average for net data

no_buffers
Subtract (file system) buffers from used memory?

nvidia_display
The display that the nvidia variable will use (defaults to the value of the display variable)

out_to_console
Print text to stdout.

out_to_http
Let conky act as a small http-server serving it's text.

out_to_ncurses
Print text in the console, but use ncurses so that conky can print the text of a new update over the old text. (In the future this will provide more useful things)

out_to_stderr
Print text to stderr.

out_to_x
When set to no, there will be no output in X (useful when you also use things like out_to_console). If you set it to no, make sure that it's placed before all other X-related setting (take the first line of your configfile to be sure). Default value is yes

override_utf8_locale
Force UTF8? requires XFT

overwrite_file
Overwrite the file given as argument.

own_window
Boolean, create own window to draw?

own_window_class
Manually set the WM_CLASS name. Defaults to "Conky".

own_window_colour colour
If own_window_transparent no, set a specified background colour (defaults to black). Takes either a hex value (e.g. ffffff, note the lack of '#') or a valid RGB name (see /usr/lib/X11/rgb.txt)

own_window_hints undecorated,below,above,sticky,skip_taskbar,skip_pager
If own_window is yes, you may use these window manager hints to affect the way Conky displays. Notes: Use own_window_type desktop as another way to implement many of these hints implicitly. If you use own_window_type override, window manager hints have no meaning and are ignored.

own_window_title
Manually set the window name. Defaults to "conky (<hostname>)".

own_window_argb_visual
Boolean, use ARGB visual? ARGB can be used for real transparency, note that a composite manager is required for real transparency. This option will not work as desired (in most cases) in conjunction with ’own_window_type override’.

own_window_argb_value
When ARGB visuals are enabled, this use this to modify the alpha value used. Valid range is 0-255, where 0 is 0% opacity, and 255 is 100% opacity.

own_window_transparent
Boolean, set transparency? If ARGB visual is enabled, sets background opacity to 0%.

own_window_type
if own_window is yes, you may specify type normal, desktop, dock, panel or override (default: normal). Desktop windows are special windows that have no window decorations; are always visible on your desktop; do not appear in your pager or taskbar; and are sticky across all workspaces. Panel windows reserve space along a desktop edge, just like panels and taskbars, preventing maximized windows from overlapping them. The edge is chosen based on the alignment option. Override windows are not under the control of the window manager. Hints are ignored. This type of window can be useful for certain situations.

pad_percents
Pad percentages to this many decimals (0 = no padding)

pop3 Default global POP3 server. Arguments are: "host user pass [-i interval (in seconds)] [-p port] [-e ’command’] [-r retries]". Default port is 110, default interval is 5 minutes, and default number of retries before giving up is 5. If the password is supplied as ’*’, you will be prompted to enter the password when Conky starts.

short_units
Shortens units to a single character (kiB->k, GiB->G, etc.). Default is off.

show_graph_range
Shows the time range covered by a graph.

show_graph_scale
Shows the maximum value in scaled graphs.

stippled_borders
Border stippling (dashing) in pixels

temperature_unit
Desired output unit of all objects displaying a temperature. Parameters are either "fahrenheit" or "celsius". The default unit is degree Celsius.

templateN
Define a template for later use inside conky.text segments. Substitute N by a digit between 0 and 9, inclusively. The value of the variable is being inserted into the stuff inside conky.text at the corresponding position, but before some substitutions are applied:

’\n’ -> newline
’\\’ -> backslash
’\ ’ -> space
’\N’ -> template argument N (starting from 1)

text_buffer_size bytes
Size of the standard text buffer (default is 256 bytes). This buffer is used for intermediary text, such as individual lines, output from $exec vars, and various other variables. Increasing the size of this buffer can drastically reduce Conky’s performance, but will allow for more text display per variable. The size of this buffer cannot be smaller than the default value of 256 bytes.

times_in_seconds
If true, variables that output times output a number that represents seconds. This doesn’t affect $time, $tztime and $utime

top_cpu_separate
If true, cpu in top will show usage of one processor’s power. If false, cpu in top will show the usage of all processors’ power combined.

top_name_verbose
If true, top name shows the full command line of each process, including arguments (whenever possible). Otherwise, only the basename is displayed. Default value is false.

top_name_width
Width for $top name value (defaults to 15 characters).

total_run_times
Total number of times for Conky to update before quitting. Zero makes Conky run forever

update_interval seconds
Update interval

update_interval_on_battery seconds
Update interval when running on batterypower

uppercase
Boolean value, if true, text is rendered in upper case

use_spacer
Adds spaces around certain objects to stop them from moving other things around. Arguments are left, right, and none (default). The old true/false values are deprecated and default to right/none respectively. Note that this only helps if you are using a mono font, such as Bitstream Vera Sans Mono.

use_xft
Use Xft (anti-aliased font and stuff)

xftalpha
Alpha of Xft font. Must be a value at or between 1 and 0.

OBJECTS/VARIABLES

Colours are parsed using XParsecolor(), there might be a list of them: /usr/share/X11/rgb.txt. Colour can be also in #rrggbb format (hex).

Some objects may create threads, and sometimes these threads will not be destroyed until Conky terminates. There is no way to destroy or clean up threads while Conky is running. For example, if you use an MPD variable, the MPD thread will keep running until Conky dies. Some threaded objects will use one of the parameters as a ’key’, so that you only have 1 relevant thread running (for example, the $curl, $rss and $weather objects launch one thread per URI).

acpiacadapter (adapter)

ACPI ac adapter state. On linux, the adapter option specifies the subfolder of /sys/class/power_supply containing the state information (tries "AC" and "ADP1" if there is no argument given). Non-linux systems ignore it.

acpifan

ACPI fan state

acpitemp

ACPI temperature in C.

addr (interface)

IP address for an interface, or "No Address" if no address is assigned.

addrs (interface)

IP addresses for an interface (if one - works like addr). Linux only.

adt746xcpu

CPU temperature from therm_adt746x

adt746xfan

Fan speed from therm_adt746x

alignc (num)

Align text to centre

alignr (num)

Right-justify text, with space of N

apcupsd host port

Sets up the connection to apcupsd daemon. Prints nothing, defaults to localhost:3551

apcupsd_cable

Prints the UPS connection type.

apcupsd_charge

Current battery capacity in percent.

apcupsd_lastxfer

Reason for last transfer from line to battery.

apcupsd_linev
Nominal input voltage.

apcupsd_load
Current load in percent.

apcupsd_loadbar
Bar showing current load.

apcupsd_loadgauge (height),(width)
Gauge that shows current load.

apcupsd_loadgraph (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)
History graph of current load.

apcupsd_model
Prints the model of the UPS.

apcupsd_name
Prints the UPS user-defined name.

apcupsd_status
Prints current status (on-line, on-battery).

apcupsd_temp
Current internal temperature.

apcupsd_timeleft
Time left to run on battery.

apcupsd_upsmode
Prints the UPS mode (e.g. standalone).

apm_adapter
Display APM AC adapter status (FreeBSD, OpenBSD only)

apm_battery_life
Display APM battery life in percent (FreeBSD, OpenBSD only)

apm_battery_time
Display remaining APM battery life in hh:mm:ss or "unknown" if AC adapterstatus is on-line or charging (FreeBSD, OpenBSD only)

audacious_bar (height),(width)
Progress bar

audacious_bitrate
Bitrate of current tune

audacious_channels
Number of audio channels of current tune

audacious_filename
Full path and filename of current tune

audacious_frequency
Sampling frequency of current tune

audacious_length
Total length of current tune as MM:SS

audacious_length_seconds
Total length of current tune in seconds

audacious_main_volume
The current volume fetched from Audacious

audacious_playlist_length
Number of tunes in playlist

audacious_playlist_position
Playlist position of current tune

audacious_position
Position of current tune (MM:SS)

audacious_position_seconds
Position of current tune in seconds

audacious_status
Player status (Playing/Paused/Stopped/Not running)

audacious_title (max length)
Title of current tune with optional maximum length specifier

battery (num)
Battery status and remaining percentage capacity of ACPI or APM battery. ACPI battery number can be given as argument (default is BAT0).

battery_bar (height),(width) (num)
Battery percentage remaining of ACPI battery in a bar. ACPI battery number can be given as argument (default is BAT0, use all to get the mean percentage remaining for all batteries).

battery_percent (num)
Battery percentage remaining for ACPI battery. ACPI battery number can be given as argument (default is BAT0, use all to get the mean percentage remaining for all batteries).

battery_short (num)
Battery status and remaining percentage capacity of ACPI or APM battery. ACPI battery number can be given as argument (default is BAT0). This mode display a short status, which means that C is displayed instead of charging, D for discharging, F for full, N for not present, E for empty and U for unknown.

battery_time (num)
Battery charge/discharge time remaining of ACPI battery. ACPI battery number can be given as argument (default is BAT0).

blink text_and_other_conky_vars
Let 'text_and_other_conky_vars' blink on and off.

bmpx_album
Album in current BMPx track

bmpx_artist
Artist in current BMPx track

bmpx_bitrate
Bitrate of the current BMPx track

bmpx_title
Title of the current BMPx track

bmpx_track
Track number of the current BMPx track

bmpx_uri
URI of the current BMPx track

buffers
Amount of memory buffered

cached
Amount of memory cached

cmdline_to_pid string
PID of the first process that has string in it's commandline

cmus_aaa
Print aaa status of cmus (all/artist/album).

cmus_album
Prints the album of the current cmus song.

cmus_artist
Prints the artist of the current cmus song.

cmus_curtime
Current time of the current cmus song.

cmus_file
Print the file name of the current cmus song

cmus_date
Print the date of the current cmus song

cmus_genre
Print the genre name of the current cmus song

cmus_percent
Percent of song's progress.

cmus_progress (height),(width)
cmus' progress bar.

cmus_random
Random status of cmus (on/off).

cmus_repeat
Repeat status of cmus (song/all/off).

cmus_state
Current state of cmus (playing, paused, stopped etc).

cmus_timeleft
Time left of the current cmus song.

cmus_title
Prints the title of the current cmus song.

cmus_totaltime
Total length of the current cmus song.

cmus_track
Print track number of current cmus song.

color (color)
Change drawing color to 'color' which is a name of a color or a hexcode preceded with # (for example #0A1B2C). If you use ncurses only the following colors are supported: red,green,yel-

low,blue,magenta,cyan,black,white.

colorN

Change drawing color to colorN configuration option, where N is a digit between 0 and 9, inclusively.

combine var1 var2

Places the lines of var2 to the right of the lines of var1 separated by the chars that are put between var1 and var2. For example: ${combine ${head /proc/cpuinfo 2} - ${head /proc/meminfo 1}} gives as output "cpuinfo_line1 - meminfo_line1" on line 1 and "cpuinfo_line2 -" on line 2. $combine vars can also be nested to place more vars next to each other.

conky_build_arch

CPU architecture Conky was built for

conky_build_date

Date Conky was built

conky_version

Conky version

cpu (cpuN)

CPU usage in percents. For SMP machines, the CPU number can be provided as an argument. ${cpu cpu0} is the total usage, and ${cpu cpuX} (X >= 1) are individual CPUs.

cpubar (cpuN) (height),(width)

Bar that shows CPU usage, height is bar's height in pixels. See $cpu for more info on SMP.

cpugauge (cpuN) (height),(width)

Elliptical gauge that shows CPU usage, height and width are gauge's vertical and horizontal axis respectively. See $cpu for more info on SMP.

cpugraph (cpuN) (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)

CPU usage graph, with optional colours in hex, minus the #. See $cpu for more info on SMP. Uses a logarithmic scale (to see small numbers) when you use the -l switch. Takes the switch '-t' to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see).

curl url (interval_in_minutes)

Download data from URI using Curl at the specified interval. The interval may be a positive floating point value (0 is allowed), otherwise defaults to 15 minutes. Most useful when used in conjunction with Lua and the Lua API. This object is threaded, and once a thread is created it can't be explicitly destroyed. One thread will run for each URI specified. You can use any protocol that Curl supports.

desktop

Number of the desktop on which conky is running or the message "Not running in X" if this is the case.

desktop_name

Name of the desktop on which conky is running or the message "Not running in X" if this is the case.

desktop_number

Number of desktops or the message "Not running in X" if this is the case.

disk_protect device

Disk protection status, if supported (needs kernel-patch). Prints either "frozen" or "free " (note the padding).

diskio (device)

Displays current disk IO. Device is optional, and takes the form of sda for /dev/sda. A block device label can be specified with label:foo. Individual partitions are also allowed.

diskio_read (device)

Displays current disk IO for reads. Device as in diskio.

diskio_write (device)

Displays current disk IO for writes. Device as in diskio.

diskiograph (device) (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)

Disk IO graph, colours defined in hex, minus the #. If scale is non-zero, it becomes the scale for the graph. Uses a logarithmic scale (to see small numbers) when you use -l switch. Takes the switch ’-t’ to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see).

diskiograph_read (device) (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)

Disk IO graph for reads, colours defined in hex, minus the #. If scale is non-zero, it becomes the scale for the graph. Device as in diskio. Uses a logarithmic scale (to see small numbers) when you use -l switch. Takes the switch ’-t’ to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see).

diskiograph_write (device) (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)

Disk IO graph for writes, colours defined in hex, minus the #. If scale is non-zero, it becomes the scale for the graph. Device as in diskio. Uses a logarithmic scale (to see small numbers) when you use -l switch. Takes the switch ’-t’ to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see).

distribution

The name of the distribution. It could be that some of the untested distributions will show up wrong or as "unknown", if that’s the case post a bug on sourceforge, make sure it contains the name of your distribution, the contents of /proc/version and if there is a file that only exists on your distribution, also add the path of that file in the bug. If there is no such file, please add another way which we can use to identify your distribution.

downspeed (net)
Download speed in suitable IEC units

downspeedf (net)
Download speed in KiB with one decimal

downspeedgraph (netdev) (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)
Download speed graph, colours defined in hex, minus the #. If scale is non-zero, it becomes the scale for the graph. Uses a logarithmic scale (to see small numbers) when you use -l switch. Takes the switch '-t' to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see).

draft_mails (maildir) (interval)
Number of mails marked as draft in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

else Text to show if any of the above are not true

endif Ends an $if block.

entropy_avail
Current entropy available for crypto freaks

entropy_bar (height),(width)
Normalized bar of available entropy for crypto freaks

entropy_perc
Percentage of entropy available in comparison to the poolsize

entropy_poolsize
Total size of system entropy pool for crypto freaks

eval string
Evaluates given string according to the rules of conky.text interpretation, i.e. parsing any contained text object specifications into their output, any occuring '$$' into a single '$' and so on. The output is then being parsed again.

eve api_keyID api_vCode character_id
Fetches a character's currently training skill from the Eve Online API servers (http://www.eveonline.com/) and displays the skill along with the remaining training time. If the character is not actively training a skill then returns the empty string (for use with $if_empty).

exec command
Executes a shell command and displays the output in conky. Warning: this takes a lot more resources than other variables. I'd recommend coding wanted behaviour in C/C++ and posting a patch.

execbar (height),(width) command

Same as exec, except if the first value returned is a value between 0-100, it will use that number to draw a horizontal bar. The height and width parameters are optional, and default to the default_bar_height and default_bar_width config settings, respectively.

execgauge (height),(width) command

Same as exec, except if the first value returned is a value between 0-100, it will use that number to draw a round gauge (much like a vehicle speedometer). The height and width parameters are optional, and default to the default_gauge_height and default_gauge_width config settings, respectively.

execgraph command (height),(width) (gradient color 1) (gradient color 2) (scale) (−t) (−l)

Draws a horizontally scrolling graph with values from 0-100 plotted on the vertical axis. All parameters following the command are optional. Gradient colors can be specified as hexadecimal values with no 0x or # prefix. Use the -t switch to enable a temperature gradient, so that small values are "cold" with color 1 and large values are "hot" with color 2. Without the -t switch, the colors produce a horizontal gradient spanning the width of the graph. The scale parameter defines the maximum value of the graph. Use the -l switch to enable a logarithmic scale, which helps to see small values. The default size for graphs can be controlled via the default_graph_height and default_graph_width config settings.

If you need to execute a command with spaces, you have a couple options: 1) wrap your command in double-quotes, or 2) put your command into a separate file, such as ˜/bin/myscript.sh, and use that as your execgraph command. Remember to make your script executable!

In the following example, we set up execgraph to display seconds (0-59) on a graph that is 50px high and 200px wide, using a temperature gradient with colors ranging from red for small values (FF0000) to yellow for large values (FFFF00). We set the scale to 60.

${execgraph ˜/seconds.sh 50,200 FF0000 FFFF00 60 -t}

execi interval command

Same as exec, but with a specific interval in seconds. The interval can't be less than the update_interval in your configuration. See also $texeci.

execibar interval (height),(width) command

Same as execbar, but with an interval.

execigauge interval (height),(width) command

Same as execgauge, but with an interval.

execigraph interval command (height),(width) (gradient color 1) (gradient color 2) (scale) (−t) (−l)

Same as execgraph, but with an interval.

execp command

Executes a shell command and displays the output in conky. Warning: this takes a lot more resources than other variables. I'd recommend coding wanted behaviour in C/C++ and posting a patch. This differs from $exec in that it parses the output of the command, so you can insert things like ${color red}hi!${color} in your script and have it correctly parsed by Conky. Caveats: Conky parses and evaluates the output of $execp every time Conky loops, and then destroys all the objects. If you try to use anything like $execi within an $execp statement, it will functionally run at the same interval that the $execp statement runs, as it is created and destroyed at every interval.

execpi interval command
Same as execp, but with an interval. Note that the output from the $execpi command is still parsed and evaluated at every interval.

flagged_mails (maildir) (interval)
Number of mails marked as flagged in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

font (font)
Specify a different font. This new font will apply to the current line and everything following. You can use a $font with no arguments to change back to the default font (much like with $color)

format_time seconds format
Format time given in seconds. This var only works when the times_in_seconds configuration setting is on. Format is a string that should start and end with a "-char. The "-chars are not part of the output, \w,\d,\h,\m,\s,\(,\) and \\ are replaced by weeks,days,hours,minutes,seconds,(,) and \. If you leave out a unit, it’s value will be expressed in the highest unite lower then the one left out. Text between ()-chars will not be visible if a replaced unit in this text is 0. If seconds is a decimal number then you can see the numbers behind the point by using \S followed by a number that specifies the amount of digits behind the point that you want to see (maximum 9). You can also place a ’x’ behind \S so you have all digits behind the point and no trailing zero’s. (also maximum 9)

forwarded_mails (maildir) (interval)
Number of mails marked as forwarded in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

freq (n)
Returns CPU #n’s frequency in MHz. CPUs are counted from 1. If omitted, the parameter defaults to 1.

freq_g (n)
Returns CPU #n’s frequency in GHz. CPUs are counted from 1. If omitted, the parameter defaults to 1.

fs_bar (height),(width) fs
Bar that shows how much space is used on a file system. height is the height in pixels. fs is any file on that file system.

fs_bar_free (height),(width) fs
Bar that shows how much space is free on a file system. height is the height in pixels. fs is any file on that file system.

fs_free (fs)
Free space on a file system available for users.

fs_free_perc (fs)
Free percentage of space on a file system available for users.

fs_size (fs)
File system size.

fs_type (fs)
File system type.

fs_used (fs)
File system used space.

fs_used_perc (fs)
Percent of file system used space.

goto x
The next element will be printed at position ’x’.

gw_iface
Displays the default route’s interface or "multiple"/"none" accordingly.

gw_ip Displays the default gateway’s IP or "multiple"/"none" accordingly.

hddtemp (dev)
Displays temperature of a selected hard disk drive as reported by the hddtemp daemon. Use hddtemp_host and hddtemp_port to specify a host and port for all hddtemp objects. If no dev parameter is given, the first disk returned by the hddtemp daemon is used.

head logfile lines (next_check)
Displays first N lines of supplied text file. The file is checked every ’next_check’ update. If next_check is not supplied, Conky defaults to 2. Max of 30 lines can be displayed, or until the text buffer is filled.

hr (height)
Horizontal line, height is the height in pixels

hwmon (dev) type n (factor offset)
Hwmon sensor from sysfs (Linux 2.6). Parameter dev may be omitted if you have only one hwmon device. Parameter type is either ’in’ or ’vol’ meaning voltage; ’fan’ meaning fan; ’temp’ meaning temperature. Parameter n is number of the sensor. See /sys/class/hwmon/ on your local computer. The optional arguments ’factor’ and ’offset’ allow precalculation of the raw input, which is being modified as follows: ’input = input * factor + offset’. Note that they have to be given as decimal values (i.e. contain at least one decimal place).

i2c (dev) type n (factor offset)
I2C sensor from sysfs (Linux 2.6). Parameter dev may be omitted if you have only one I2C device. Parameter type is either ’in’ or ’vol’ meaning voltage; ’fan’ meaning fan; ’temp’ meaning temperature. Parameter n is number of the sensor. See /sys/bus/i2c/devices/ on your local computer. The optional arguments ’factor’ and ’offset’ allow precalculation of the raw input, which is being modified as follows: ’input = input * factor + offset’. Note that they have to be given as decimal values (i.e. contain at least one decimal place).

i8k_ac_status
If running the i8k kernel driver for Inspiron laptops, displays whether ac power is on, as listed in /proc/i8k (translated to human-readable). Beware that this is by default not enabled by i8k itself.

i8k_bios
If running the i8k kernel driver for Inspiron laptops, displays the bios version as listed in /proc/i8k.

i8k_buttons_status
If running the i8k kernel driver for Inspiron laptops, displays the volume buttons status as listed in /proc/i8k.

i8k_cpu_temp
If running the i8k kernel driver for Inspiron laptops, displays the cpu temperature in Celsius, as reported by /proc/i8k.

i8k_left_fan_rpm
If running the i8k kernel driver for Inspiron laptops, displays the left fan's rate of rotation, in revolutions per minute as listed in /proc/i8k. Beware, some laptops i8k reports these fans in reverse order.

i8k_left_fan_status
If running the i8k kernel driver for Inspiron laptops, displays the left fan status as listed in /proc/i8k (translated to human-readable). Beware, some laptops i8k reports these fans in reverse order.

i8k_right_fan_rpm
If running the i8k kernel driver for Inspiron laptops, displays the right fan's rate of rotation, in revolutions per minute as listed in /proc/i8k. Beware, some laptops i8k reports these fans in reverse order.

i8k_right_fan_status
If running the i8k kernel driver for Inspiron laptops, displays the right fan status as listed in /proc/i8k (translated to human-readable). Beware, some laptops i8k reports these fans in reverse order.

i8k_serial
If running the i8k kernel driver for Inspiron laptops, displays your laptop serial number as listed in /proc/i8k.

i8k_version
If running the i8k kernel driver for Inspiron laptops, displays the version formatting of /proc/i8k.

ibm_brightness
If running the IBM ACPI, displays the brigtness of the laptops's LCD (0-7).

ibm_fan
If running the IBM ACPI, displays the fan speed.

`ibm_temps N`
If running the IBM ACPI, displays the temperatures from the IBM temperature sensors (N=0..7) Sensor 0 is on the CPU, 3 is on the GPU.

`ibm_thinklight`
If running the IBM ACPI, displays the status of your ThinkLight™. Value is either ’on’, ’off’ or ’unknown’.

`ibm_volume`
If running the IBM ACPI, displays the "master" volume, controlled by the volume keys (0-14).

`ical number file`
Shows title of event number ’number’ in the ical (RFC 5545) file ’file’. The events are first ordered by starting time, events that started in the past are ignored. The events that are shown are the VEVENTS, the title that is shown is the SUMMARY and the starting time used for sorting is DT-START .

`irc server(:port) #channel (max_msg_lines)`
Shows everything that’s being told in #channel on IRCserver ’server’. TCP-port 6667 is used for the connection unless ’port’ is specified. Shows everything since the last time or the last ’max_msg_lines’ entries if specified.

`iconv_start codeset_from codeset_to`
Convert text from one codeset to another using GNU iconv. Needs to be stopped with iconv_stop.

`iconv_stop`
Stop iconv codeset conversion.

`if_empty (var)`
if conky variable VAR is empty, display everything between $if_empty and the matching $endif

`if_existing file (string)`
if FILE exists, display everything between if_existing and the matching $endif. The optional second parameter checks for FILE containing the specified string and prints everything between $if_existing and the matching $endif.

`if_gw` if there is at least one default gateway, display everything between $if_gw and the matching $endif

`if_match expression`
Evaluates the given boolean expression, printing everything between $if_match and the matching $endif depending on whether the evaluation returns true or not. Valid expressions consist of a left side, an operator and a right side. Left and right sides are being parsed for contained text objects before evaluation. Recognised left and right side types are:

doubleArgument consists of only digits and a single dot.
longArgument consists of only digits.
stringArgument is enclosed in quotation marks (")

Valid operands are: ’>’, ’<’, ’>=’, ’<=’, ’==’, ’!=’.

`if_mixer_mute (mixer)`

If mixer exists, display everything between $if_mixer_mute and the matching $endif. If no mixer is specified, "Vol" is used.

`if_mounted (mountpoint)`

if MOUNTPOINT is mounted, display everything between $if_mounted and the matching $endif

`if_mpd_playing`

if mpd is playing or paused, display everything between $if_mpd_playing and the matching $endif

`if_pa_sink_muted`

If Pulseaudio's default sink is muted, display everything between $if_pa_sink_muted and the corresponding $else or $endif.

`if_running (process)`

If PROCESS is running, display everything between $if_running and the corresponding $else or $endif. Note that PROCESS may be either a full command line with arguments (without the directory prefix), or simply the name of an executable. For example, either of the following will be true if there is a running process with the command line **/usr/bin/conky -u 5**:

${if_running conky -u 5}** or **${if_running conky}

It is important not to include trailing spaces. For example, **${if_running conky }** will be false.

`if_smapi_bat_installed (INDEX)`

when using smapi, if the battery with index INDEX is installed, display everything between $if_smapi_bat_installed and the matching $endif

`if_up (interface)`

if INTERFACE exists and is up, display everything between $if_up and the matching $endif

`if_updatenr (updatenr)`

If it's the UPDATENR-th time that conky updates, display everything between $if_updatenr and the matching $endif. The counter resets when the highest UPDATENR is reached. Example : "{$if_updatenr 1}foo$endif{$if_updatenr 2}bar$endif{$if_updatenr 4}$endif" shows foo 25% of the time followed by bar 25% of the time followed by nothing the other half of the time.

`if_xmms2_connected`

Display everything between $if_xmms2_connected and the matching $endif if xmms2 is running.

`image <path to image> (-p x,y) (-s WxH) (-n) (-f interval)`

Renders an image from the path specified using Imlib2. Takes 4 optional arguments: a position, a size, a no-cache switch, and a cache flush interval. Changing the x,y position will move the position of the image, and changing the WxH will scale the image. If you specify the no-cache flag (-n), the image will not be cached. Alternately, you can specify the -f int switch to specify a cache flush interval for a particular image. Example: ${image /home/brenden/cheeseburger.jpg -p 20,20 -s 200x200} will render 'cheeseburger.jpg' at (20,20) scaled to 200x200 pixels. Conky does not make any attempt to adjust the position (or any other formatting) of images, they are just rendered as per the arguments passed. The only reason $image is part of the conky.text section, is to allow for runtime modifications, through $execp $lua_parse, or some other method.

imap_messages (args)

Displays the number of messages in your global IMAP inbox by default. You can define individual IMAP inboxes separately by passing arguments to this object. Arguments are: "host user pass [-i interval (in seconds)] [-f 'folder'] [-p port] [-e 'command'] [-r retries]". Default port is 143, default folder is 'INBOX', default interval is 5 minutes, and default number of retries before giving up is 5. If the password is supplied as '*', you will be prompted to enter the password when Conky starts.

imap_unseen (args)

Displays the number of unseen messages in your global IMAP inbox by default. You can define individual IMAP inboxes separately by passing arguments to this object. Arguments are: "host user pass [-i interval (in seconds)] [-f 'folder'] [-p port] [-e 'command'] [-r retries]". Default port is 143, default folder is 'INBOX', default interval is 5 minutes, and default number of retries before giving up is 5. If the password is supplied as '*', you will be prompted to enter the password when Conky starts.

ioscheduler disk

Prints the current ioscheduler used for the given disk name (i.e. e.g. "hda" or "sdb")

journal lines (type)

Displays last N lines of the systemd journal. The optional type can be 'user' or 'system' which will show only the user or system journal respectively. By default, all journal lines visible to the user are shown. A maximum of 200 lines can be displayed, or until the text buffer is filled.

kernel

Kernel version

version

Git version numer (DragonFly only)

laptop_mode

The value of /proc/sys/vm/laptop_mode

lines textfile

Displays the number of lines in the given file

loadavg (1|2|3)

System load average, 1 is for past 1 minute, 2 for past 5 minutes and 3 for past 15 minutes. Without argument, prints all three values separated by whitespace.

loadgraph (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)

Load1 average graph, similar to xload, with optional colours in hex, minus the #. Uses a logarithmic scale (to see small numbers) when you use the -l switch. Takes the switch '-t' to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see).

lua function_name (function parameters)

Executes a Lua function with given parameters, then prints the returned string. See also 'lua_load' on how to load scripts. Conky puts 'conky_' in front of function_name to prevent accidental calls to the wrong function unless you put you place ’conky_’ in front of it yourself.

lua_bar (height, width) function_name (function parameters)

Executes a Lua function with given parameters and draws a bar. Expects result value to be an integer between 0 and 100. See also ’lua_load’ on how to load scripts. Conky puts ’conky_’ in front of function_name to prevent accidental calls to the wrong function unless you put you place ’conky_’ in front of it yourself.

lua_gauge (height, width) function_name (function parameters)

Executes a Lua function with given parameters and draws a gauge. Expects result value to be an integer between 0 and 100. See also ’lua_load’ on how to load scripts. Conky puts ’conky_’ in front of function_name to prevent accidental calls to the wrong function unless you put you place ’conky_’ in front of it yourself.

lua_graph function_name (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)

Executes a Lua function with and draws a graph. Expects result value to be any number, and by default will scale to show the full range. See also ’lua_load’ on how to load scripts. Takes the switch ’-t’ to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see). Conky puts ’conky_’ in front of function_name to prevent accidental calls to the wrong function unless you put you place ’conky_’ in front of it yourself.

lua_parse function_name (function parameters)

Executes a Lua function with given parameters as per $lua, then parses and prints the result value as per the syntax for the conky.text section. See also ’lua_load’ on how to load scripts. Conky puts ’conky_’ in front of function_name to prevent accidental calls to the wrong function unless you put you place ’conky_’ in front of it yourself.

machine

Machine, i686 for example

mails (mailbox) (interval)

Mail count in the specified mailbox or your mail spool if not. Both mbox and maildir type mailboxes are supported. You can use a program like fetchmail to get mails from some server using your favourite protocol. See also new_mails.

mboxscan (−n number of messages to print) (−fw from width) (−sw subject width) mbox

Print a summary of recent messages in an mbox format mailbox. mbox parameter is the filename of the mailbox (can be encapsulated using ’"’, ie. ${mboxscan -n 10 "/home/brenden/some box"}

mem Amount of memory in use

memwithbuffers

Amount of memory in use, including that used by system buffers and caches

membar (height),(width)

Bar that shows amount of memory in use

memwithbuffersbar (height),(width)

Bar that shows amount of memory in use (including memory used by system buffers and caches)

memdirty

Amount of "dirty" memory (linux only)

memeasyfree

Amount of free memory including the memory that is very easily freed (buffers/cache)

memfree

Amount of free memory

memgauge (height),(width)

Gauge that shows amount of memory in use (see cpugauge)

memgraph (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)

Memory usage graph. Uses a logarithmic scale (to see small numbers) when you use the -l switch. Takes the switch ’-t’ to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see).

memmax

Total amount of memory

memperc

Percentage of memory in use

mixer (device)

Prints the mixer value as reported by the OS. On Linux, this variable uses the OSS emulation, so you need the proper kernel module loaded. Default mixer is "Vol", but you can specify one of the available OSS controls: "Vol", "Bass", "Trebl", "Synth", "Pcm", "Spkr", "Line", "Mic", "CD", "Mix", "Pcm2 ", "Rec", "IGain", "OGain", "Line1", "Line2", "Line3", "Digital1", "Digital2", "Digital3", "PhoneIn", "PhoneOut", "Video", "Radio" and "Monitor".

mixerbar (device)

Displays mixer value in a bar as reported by the OS. See docs for $mixer for details on arguments.

mixerl (device)

Prints the left channel mixer value as reported by the OS. See docs for $mixer for details on arguments.

mixerlbar (device)

Displays the left channel mixer value in a bar as reported by the OS. See docs for $mixer for details on arguments.

mixerr (device)

Prints the right channel mixer value as reported by the OS. See docs for $mixer for details on arguments.

mixerrbar (device)
Displays the right channel mixer value in a bar as reported by the OS. See docs for $mixer for details on arguments.

moc_album
Album of the current MOC song

moc_artist
Artist of the current MOC song

moc_bitrate
Bitrate in the current MOC song

moc_curtime
Current time of the current MOC song

moc_file
File name of the current MOC song

moc_rate
Rate of the current MOC song

moc_song
The current song name being played in MOC.

moc_state
Current state of MOC; playing, stopped etc.

moc_timeleft
Time left in the current MOC song

moc_title
Title of the current MOC song

moc_totaltime
Total length of the current MOC song

monitor
Number of the monitor on which conky is running or the message "Not running in X" if this is the case.

monitor_number
Number of monitors or the message "Not running in X" if this is the case.

mpd_album
Album in current MPD song

mpd_artist
Artist in current MPD song must be enabled at compile

mpd_albumartist
Artist of the album of the current MPD song.

mpd_bar (height),(width)
Bar of mpd's progress

mpd_bitrate
Bitrate of current song

mpd_date
Date of current song

mpd_elapsed
Song's elapsed time

mpd_file
Prints the file name of the current MPD song

mpd_length
Song's length

mpd_name
Prints the MPD name field

mpd_percent
Percent of song's progress

mpd_random
Random status (On/Off)

mpd_repeat
Repeat status (On/Off)

mpd_smart (max length)
Prints the song name in either the form "artist - title" or file name, depending on whats available

mpd_status
Playing, stopped, et cetera.

mpd_title (max length)
Title of current MPD song

mpd_track
Prints the MPD track field

mpd_vol
MPD's volume

mysql query
Shows the first field of the first row of the result of the query.

nameserver (index)
Print a nameserver from /etc/resolv.conf. Index starts at and defaults to 0.

new_mails (mailbox) (interval)
Unread mail count in the specified mailbox or mail spool if not. Both mbox and maildir type mailboxes are supported.

nodename
Hostname

nodename_short
Short hostname (same as 'hostname -s' shell command).

no_update text
Shows text and parses the vars in it, but doesn't update them. Use this for things that do not change while conky is running, like $machine, $conky_version,... By not updating this you can save some resources.

nvidia argument (GPU_ID)
Nvidia graphics card information via the XNVCtrl library.

GPU_ID: Optional parameter to choose the GPU to be used as 0,1,2,3,.. Default parameter is 0

Possible arguments: (Temperatures are printed as float, all other values as integer. Bracketed arguments are aliases)

gputemp (**temp**) **GPU temperature**
gputempthreshold (**threshold**) **Temperature threshold where the GPU will reduce it's clock speed**
ambienttemp (**ambient**) **Ambient temperature outside the graphics card**
gpufreqcur (**gpufreq**) **Current GPU clock speed**
gpufreqmin **Minimum GPU clock speed**
gpufreqmax **Maximum GPU clock speed**
memfreqcur (**memfreq**) **Current memory clock speed**
memfreqmin **Minimum memory clock speed**
memfreqmax **Maximum memory clock speed**
mtrfreqcur (**mtrfreq**) **Current memory transfer rate clock speed**
mtrfreqmin **Minimum memory transfer rate clock speed**
mtrfreqmax **Maximum memory transfer rate clock speed**
perflevelcur (**perflevel**) **Current performance level**
perflevelmin **Lowest performance level**
perflevelmax **Highest performance level**
perfmode **Performance mode**
gpuutil **GPU utilization %**
membwutil **Memory bandwidth utilization %**
videoutil **Video engine utilization %**

pcieutil `PCIe bandwidth utilization %`
memused (**mem**) `Amount of used memory`
memfree (**memavail**) `Amount of free memory`
memmax (**memtotal**) `Total amount of memory`
memutil (**memperc**) `Memory utilization %`
fanspeed `Fan speed`
fanlevel `Fan level %`
imagequality `Image quality`
modelname `name of the GPU card`

`nvidiabar (height),(width) argument (GPU_ID)`

Same as nvidia, except it draws its output in a horizontal bar. The height and width parameters are optional, and default to the default_bar_height and default_bar_width config settings, respectively.

GPU_ID: Optional parameter to choose the GPU to be used as 0,1,2,3,.. Default parameter is 0

Note the following arguments are incompatible: **gputempthreshold** (**threshold**)
gpufreqmin
gpufreqmax
memfreqmin
memfreqmax
mtrfreqmin
mtrfreqmax
perflevelmin
perflevelmax
perfmode
memtotal (**memmax**)
fanspeed

`nvidiagauge (height),(width) argument (GPU_ID)`

Same as nvidiabar, except a round gauge (much like a vehicle speedometer). The height and width parameters are optional, and default to the default_gauge_height and default_gauge_width config settings, respectively.

GPU_ID: Optional parameter to choose the GPU to be used as 0,1,2,3,.. Default parameter is 0

For possible arguments see nvidia and nvidiabar.

`nvidiagraph argument (height),(width) (gradient color 1) (gradient color 2) (scale) (-t) (-l) GPU_ID`

Same as nvidiabar, except a horizontally scrolling graph with values from 0-100 plotted on the vertical axis. The height and width parameters are optional, and default to the default_graph_height and default_graph_width config settings, respectively.

GPU_ID: NOT optional. This parameter allows to choose the GPU to be used as 0,1,2,3,..

For possible arguments see nvidia and nvidiabar. To learn more about the -t -l and gradient color options, see execgraph.

`offset (pixels)`

Move text over by N pixels. See also $voffset.

`outlinecolor (color)`

Change outline color

pa_sink_volume

Pulseaudio's default sink volume percentage.

pa_sink_volumebar

Pulseaudio's default sink volume bar.

pa_sink_description

Pulseaudio's default sink description.

pa_sink_active_port_name

Pulseaudio's default sink active port name.

pa_sink_active_port_description

Pulseaudio's default sink active port description.

pa_card_name

Pulseaudio's default card name.

pa_card_active_profile

Pulseaudio's default card active profile.

pb_battery item

If running on Apple powerbook/ibook, display information on battery status. The item parameter specifies, what information to display. Exactly one item must be specified. Valid items are:

status Display if battery is fully charged, charging, discharging or absent (running on AC)
percent Display charge of battery in percent, if charging or discharging. Nothing will be displayed, if battery is fully charged or absent.
time Display the time remaining until the battery will be fully charged or discharged at current rate. Nothing is displayed, if battery is absent or if it's present but fully charged and not discharging.

pid_chroot pid

Directory used as rootdirectory by the process (this will be "/" unless the process did a chroot syscall)

pid_cmdline pid

Command line this process was invoked with

pid_cwd pid

Current working directory of the process

pid_environ pid varname

Contents of a environment-var of the process

pid_environ_list pid
List of environment-vars that the process can see

pid_exe pid
Path to executed command that started the process

pid_nice pid
The nice value of the process

pid_openfiles pid
List of files that the process has open

pid_parent pid
The pid of the parent of the process

pid_priority pid
The priority of the process (see ’priority’ in "man 5 proc")

pid_read pid
Total number of bytes read by the process

pid_state pid
State of the process

pid_state_short pid
One of the chars in "RSDZTW" representing the state of the process where R is running, S is sleeping in an interruptible wait, D is waiting in uninterruptible disk sleep, Z is zombie, T is traced or stopped (on a signal), and W is paging

pid_stderr pid
Filedescriptor binded to the STDERR of the process

pid_stdin pid
Filedescriptor binded to the STDIN of the process

pid_stdout pid
Filedescriptor binded to the STDOUT of the process

pid_threads pid
Number of threads in process containing this thread

pid_thread_list pid
List with pid’s from threads from this process

pid_time_kernelmode pid
Amount of time that the process has been scheduled in kernel mode in seconds

pid_time_usermode pid
Amount of time that the process has been scheduled in user mode in seconds

pid_time pid
Sum of $pid_time_kernelmode and $pid_time_usermode

pid_uid pid
The real uid of the process

pid_euid pid
The effective uid of the process

pid_suid pid
The saved set uid of the process

pid_fsuid pid
The file system uid of the process

pid_gid pid
The real gid of the process

pid_egid pid
The effective gid of the process

pid_sgid pid
The saved set gid of the process

pid_fsgid pid
The file system gid of the process

pid_vmpeak pid
Peak virtual memory size of the process

pid_vmsize pid
Virtual memory size of the process

pid_vmlck pid
Locked memory size of the process

pid_vmhwm pid
Peak resident set size ("high water mark") of the process

pid_vmrss pid
Resident set size of the process

pid_vmdata pid
Data segment size of the process

pid_vmstk pid

Stack segment size of the process

pid_vmexe pid

Text segment size of the process

pid_vmlib pid

Shared library code size of the process

pid_vmpte pid

Page table entries size of the process

pid_write pid

Total number of bytes written by the process

platform (dev) type n (factor offset)

Platform sensor from sysfs (Linux 2.6). Parameter dev may be omitted if you have only one platform device. Platform type is either 'in' or 'vol' meaning voltage; 'fan' meaning fan; 'temp' meaning temperature. Parameter n is number of the sensor. See /sys/bus/platform/devices/ on your local computer. The optional arguments 'factor' and 'offset' allow precalculation of the raw input, which is being modified as follows: 'input = input * factor + offset'. Note that they have to be given as decimal values (i.e. contain at least one decimal place).

pop3_unseen (args)

Displays the number of unseen messages in your global POP3 inbox by default. You can define individual POP3 inboxes separately by passing arguments to this object. Arguments are: "host user pass [-i interval (in seconds)] [-p port] [-e 'command'] [-r retries]". Default port is 110, default interval is 5 minutes, and default number of retries before giving up is 5. If the password is supplied as '*', you will be prompted to enter the password when Conky starts.

pop3_used (args)

Displays the amount of space (in MiB, 2ˆ20) used in your global POP3 inbox by default. You can define individual POP3 inboxes separately by passing arguments to this object. Arguments are: "host user pass [-i interval (in seconds)] [-p port] [-e 'command'] [-r retries]". Default port is 110, default interval is 5 minutes, and default number of retries before giving up is 5. If the password is supplied as '*', you will be prompted to enter the password when Conky starts.

processes

Total processes (sleeping and running)

read_tcp (host) port

Connects to a tcp port on a host (default is localhost), reads every char available at the moment and shows them.

read_udp (host) port

Connects to a udp port on a host (default is localhost), reads every char available at the moment and shows them.

replied_mails (maildir) (interval)
Number of mails marked as replied in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

rss uri interval_in_seconds action (num_par (spaces_in_front))
Download and parse RSS feeds. The interval may be a (floating point) value greater than 0. Action may be one of the following: feed_title, item_title (with num par), item_desc (with num par) and item_titles (when using this action and spaces_in_front is given conky places that many spaces in front of each item). This object is threaded, and once a thread is created it can't be explicitly destroyed. One thread will run for each URI specified. You can use any protocol that Curl supports.

running_processes
Running processes (not sleeping), requires Linux 2.6

running_threads
Number of running (runnable) threads. Linux only.

scroll (direction) length (step) (interval) text
Scroll 'text' by 'step' characters to the left or right (set 'direction' to 'left' or 'right' or 'wait') showing 'length' number of characters at the same time. The text may also contain variables. 'step' is optional and defaults to 1 if not set. 'direction' is optional and defaults to left if not set. When direction is 'wait' then text will scroll left and wait for 'interval' itertations at the beginning and end of the text. If a var creates output on multiple lines then the lines are placed behind each other separated with a '|'-sign. If you change the textcolor inside $scroll it will automatically have it's old value back at the end of $scroll. The end and the start of text will be separated by 'length' number of spaces unless direction is 'wait'.

seen_mails (maildir) (interval)
Number of mails marked as seen in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

shadecolor (color)
Change shading color

smapi (ARGS)
when using smapi, display contents of the /sys/devices/platform/smapi directory. ARGS are either '(FILENAME)' or 'bat (INDEX) (FILENAME)' to display the corresponding files' content. This is a very raw method of accessing the smapi values. When available, better use one of the smapi_* variables instead.

smapi_bat_bar (INDEX),(height),(width)
when using smapi, display the remaining capacity of the battery with index INDEX as a bar.

smapi_bat_perc (INDEX)
when using smapi, display the remaining capacity in percent of the battery with index INDEX. This is a separate variable because it supports the 'use_spacer' configuration option.

smapi_bat_power INDEX
when using smapi, display the current power of the battery with index INDEX in watt. This is a separate variable because the original read out value is being converted from mW. The sign of the

output reflects charging (positive) or discharging (negative) state.

smapi_bat_temp INDEX
when using smapi, display the current temperature of the battery with index INDEX in degree Celsius. This is a separate variable because the original read out value is being converted from milli degree Celsius.

sony_fanspeed
Displays the Sony VAIO fanspeed information if sony-laptop kernel support is enabled. Linux only.

stippled_hr (space)
Stippled (dashed) horizontal line

stock symbol data
Displays the data of a stock symbol. The following data is supported: adv(Average Daily Volume), ask, asksize, bid, askrt(ask realtime), bidrt(bid realtime), bookvalue, bidsize, change, commission, changert(change realtime), ahcrt(After Hours Change realtime), ds(dividend/share), ltd(Last Trade Date), tradedate, es(earnings/share), ei(error indication), epsecy(EPS Estimate Current Year), epseny(EPS Estimate Next Year), epsenq(EPS Estimate Next Quarter), floatshares, dayslow, dayshigh, 52weeklow, 52weekhigh, hgp(Holdings Gain Percent), ag(Annualized Gain), hg(Holdings Gain), hgprt(Holdings Gain Percent realtime), hgrt(Holdings Gain realtime), moreinfo, obrt(Order Book realtime), mc(Market Capitalization), mcrt(Market Cap realtime), ebitda, c52wlow(Change From 52-week Low), pc52wlow(Percent Change From 52-week Low), cprt(change percent realtime), lts(Last Trade Size), c52whigh(Change from 52-week high), pc52whigh(percent change from 52-week high), ltp(last trade price), hl(high limit), ll(low limit), dr(day’s range), drrt(day’s range realtime), 50ma(50-day Moving Average), 200ma(200-day Moving Average), c200ma(Change From 200-day Moving Average), pc200ma(Percent Change From 200-day Moving Average), c50ma(Change From 50-day Moving Average), pc50ma(Percent Change From 50-day Moving Average), name, notes, open, pc(previous close), pricepaid, cip(change in percent), ps(price/sales), pb(price/book), edv(Ex-Dividend Date), per(P/E Ratio), dpd(Dividend Pay Date), perrt(P/E Ratio realtime), pegr(PEG Ratio), pepsecy(Price/EPS Estimate Current Year), pepseny(Price/EPS Estimate Next Year), symbol, sharesowned, shortratio, ltt(Last Trade Time), tradelinks, tt(Ticker Trend), 1ytp(1 yr Target Price), volume, hv(Holdings Value), hvrt(Holdings Value realtime), 52weekrange, dvc(Day’s Value Change), dvcrt(Day’s Value Change realtime), se(Stock Exchange), dy(Dividend Yield)

swap Amount of swap in use

swapbar (height),(width)
Bar that shows amount of swap in use

swapfree
Amount of free swap

swapmax
Total amount of swap

swapperc
Percentage of swap in use

sysname

System name, Linux for example

tab (width, (start))

Puts a tab of the specified width, starting from column ’start’. The unit is pixels for both arguments.

tail logfile lines (next_check)

Displays last N lines of supplied text file. The file is checked every ’next_check’ update. If next_check is not supplied, Conky defaults to 2. Max of 30 lines can be displayed, or until the text buffer is filled.

tcp_ping host (port)

Displays the number of microseconds it takes to get a reply on a ping to to tcp ’port’ on ’host’. ’port’ is optional and has 80 as default. This works on both open and closed ports, just make sure that the port is not behind a firewall or you will get ’down’ as answer. It’s best to test a closed port instead of an open port, you will get a quicker response.

tcp_portmon port_begin port_end item (index)

TCP port (both IPv6 and IPv4) monitor for specified local ports. Port numbers must be in the range 1 to 65535. Valid items are:

count Total number of connections in the range
rip Remote ip address
rhost Remote host name
rport Remote port number
rservice Remote service name from /etc/services
lip Local ip address
lhost Local host name
lport Local port number
lservice Local service name from /etc/services

The connection index provides you with access to each connection in the port monitor. The monitor will return information for index values from 0 to n-1 connections. Values higher than n-1 are simply ignored. For the "count" item, the connection index must be omitted. It is required for all other items.

Examples:

${tcp_portmon 6881 6999 count} Displays the number of connections in the bittorrent port range
${tcp_portmon 22 22 rip 0} Displays the remote host ip of the first sshd connection
${tcp_portmon 22 22 rip 9} Displays the remote host ip of the tenth sshd connection
${tcp_portmon 1 1024 rhost 0} Displays the remote host name of the first connection on a privileged port
${tcp_portmon 1 1024 rport 4} Displays the remote host port of the fifth connection on a privileged port
${tcp_portmon 1 65535 lservice 14} Displays the local service name of the fifteenth connection in the range of all ports

Note that port monitor variables which share the same port range actually refer to the same monitor, so many references to a single port range for different items and different indexes all use the same monitor internally. In other words, the program avoids creating redundant monitors.

templateN (arg1) (arg2) (arg3 ...)

Evaluate the content of the templateN configuration variable (where N is a value between 0 and 9, inclusively), applying substitutions as described in the documentation of the corresponding configuration variable. The number of arguments is optional, but must match the highest referred index in the template. You can use the same special sequences in each argument as the ones valid for a template definition, e.g. to allow an argument to contain a whitespace. Also simple nesting of templates is possible this way.

Here are some examples of template definitions, note they are placed between [[...]] instead of ’ ... ’:

template0 = [[$\1\2]]
template1 = [[\1: ${fs_used \2} / ${fs_size \2}]]
template2 = [[\1 \2]]

The following list shows sample usage of the templates defined above, with the equivalent syntax when not using any template at all:

using template	same without template
${template0 node name}	$nodename
${template1 root /}	root: ${fs_free /} / ${fs_size /}
${template1 ${template2\ disk\ root} /}	disk root: ${fs_free /} / ${fs_size /}

texeci interval command

Runs a command at an interval inside a thread and displays the output. Same as $execi, except the command is run inside a thread. Use this if you have a slow script to keep Conky updating. You should make the interval slightly longer than the time it takes your script to execute. For example, if you have a script that take 5 seconds to execute, you should make the interval at least 6 seconds. See also $execi. This object will clean up the thread when it is destroyed, so it can safely be used in a nested fashion, though it may not produce the desired behaviour if used this way.

texecpi interval command

Same as execpi, except the command is run inside a thread.

threads

Total threads

time (format)

Local time, see man strftime to get more information about format

to_bytes size

If ’size’ is a number followed by a size-unit (kilobyte,mb,GiB,...) then it converts the size to bytes and shows it without unit, otherwise it just shows ’size’.

top type num

This takes arguments in the form:top (name) (number) Basically, processes are ranked from highest to lowest in terms of cpu usage, which is what (num) represents. The types are: "name", "pid", "cpu", "mem", "mem_res", "mem_vsize", "time", "uid", "user", "io_perc", "io_read" and "io_write". There can be a max of 10 processes listed.

top_io type num
Same as top, except sorted by the amount of I/O the process has done during the update interval

top_mem type num
Same as top, except sorted by mem usage instead of cpu

top_time type num
Same as top, except sorted by total CPU time instead of current CPU usage

totaldown (net)
Total download, overflows at 4 GB on Linux with 32-bit arch and there doesn't seem to be a way to know how many times it has already done that before conky has started.

totalup (net)
Total upload, this one too, may overflow

trashed_mails (maildir) (interval)
Number of mails marked as trashed in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

tztime (timezone (format))
Local time for specified timezone, see man strftime to get more information about format. The timezone argument is specified in similar fashion as TZ environment variable. For hints, look in /usr/share/zoneinfo. e.g. US/Pacific, Europe/Zurich, etc.

gid_name gid
Name of group with this gid

uid_name uid
Username of user with this uid

unflagged_mails (maildir) (interval)
Number of mails not marked as flagged in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

unforwarded_mails (maildir) (interval)
Number of mails not marked as forwarded in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

unreplied_mails (maildir) (interval)
Number of mails not marked as replied in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

unseen_mails (maildir) (interval)
Number of new or unseen mails in the specified mailbox or mail spool if not. Only maildir type mailboxes are supported, mbox type will return -1.

updates Number of updates
for debugging

upspeed (net)
Upload speed in suitable IEC units

upspeedf (net)
Upload speed in KiB with one decimal

upspeedgraph (netdev) (height),(width) (gradient colour 1) (gradient colour 2) (scale) (−t) (−l)
Upload speed graph, colours defined in hex, minus the #. If scale is non-zero, it becomes the scale for the graph. Uses a logarithmic scale (to see small numbers) when you use the -l switch. Takes the switch ’-t’ to use a temperature gradient, which makes the gradient values change depending on the amplitude of a particular graph value (try it and see).

uptime
Uptime

uptime_short
Uptime in a shorter format

user_names
Lists the names of the users logged in

user_number
Number of users logged in

user_terms
Lists the consoles in use

user_times
Lists how long users have been logged in for

user_time console
Lists how long the user for the given console has been logged in for

utime (format)
Display time in UTC (universal coordinate time).

v6addrs (−n) (−s) (interface)
IPv6 addresses for an interface, followed by netmask if -n is specified and scope with -s. Scopes are Global(G), Host-local(H), Link-local(L), Site-local(S), Compat(C) and Unspecified(/). Linux only.

voffset (pixels)
Change vertical offset by N pixels. Negative values will cause text to overlap. See also $offset.

`voltage_mv (n)`

Returns CPU #n’s voltage in mV. CPUs are counted from 1. If omitted, the parameter defaults to 1.

`voltage_v (n)`

Returns CPU #n’s voltage in V. CPUs are counted from 1. If omitted, the parameter defaults to 1.

`weather URI locID data_type (interval_in_minutes)`

Download, parse and display METAR data.

For the ’URI’, there are two possibilities:

http://weather.noaa.gov/pub/data/observations/metar/stations/
http://xoap.weather.com/weather/local/

The first one is free to use but the second requires you to register and obtain your partner ID and license key. These two must be written, separated by a space, into a file called .xoaprc which needs to be placed into your home directory.

’locID’ must be a valid location identifier for the required uri. For the NOAA site this must be a valid ICAO (see for instance https://pilotweb.nas.faa.gov/qryhtml/icao/). For the weather.com site this must be a valid location ID (see for instance http://aspnetresources.com/tools/locid.aspx).

’data_type’ must be one of the following:

last_update The date and time stamp of the data. The result depends on the URI used. For the NOAA site it is date (yyyy/mm/dd) and UTC time. For the weather.com one it is date ([m]m/[d]d/yy) and Local Time of the station.
temperature

Air temperature (you can use the ’temperature_unit’ config setting to change units)
cloud_cover

The highest cloud cover status
pressure

Air pressure in millibar
wind_speed

Wind speed in km/h
wind_dir

Wind direction
wind_dir_DEG

Compass wind direction
humidity

Relative humidity in %
weather

Any relevant weather event (rain, snow, etc.). This is not used if you are querying the weather.com site since this data is aggregated into the cloud_cover one
icon

Weather icon (only for www.weather.com). Can be used together with the icon kit provided upon registering to their service.

’delay_in_minutes’ (optional, default 30) cannot be less than 30 minutes.

This object is threaded, and once a thread is created it can’t be explicitly destroyed. One thread will run for each URI specified.

Note that these variables are still EXPERIMENTAL and can be subject to many future changes.

weather_forecast URI locID day data_type (interval_in_minutes)

Download, parse and display weather forecast data for a given day (daytime only).

For the ’URI’, for the time being only http://xoap.weather.com/weather/local/ is supported. See ’weather’ above for details of usage

’locID’, see ’weather’ above.

’day’ is a number from 0 (today) to 4 (3 days after tomorrow).

’data_type’ must be one of the following:

day **Day of the week**
date **Date, in the form MMM DD (ie. Jul 14)**
low **Minimun temperature (you can use the ’temperature_unit’ config setting to change units)**
hi **Maximum temperature (you can use the ’temperature_unit’ config setting to change units)**
icon **Weather icon. Can be used together with the icon kit provided upon registering to the weather.com service**
forecast **Weather forecast (sunny, rainy, etc.)**
wind_speed **Wind speed in km/h**
wind_dir **Wind direction**
wind_dir_DEG **Compass wind direction**
humidity **Relative humidity in %**
precipitation **Probability of having a precipitation (in %)**

’delay_in_minutes’ (optional, default 210) cannot be lower than 210 min.

This object is threaded, and once a thread is created it can’t be explicitly destroyed. One thread will run for each URI specified. You can use any protocol that Curl supports.

Note that these variables are still EXPERIMENTAL and can be subject to many future changes.

wireless_ap (net)

Wireless access point MAC address (Linux only)

wireless_bitrate (net)

Wireless bitrate (ie 11 Mb/s) (Linux only)

wireless_channel (net)

WLAN channel on which device ’net’ is listening (Linux only)

wireless_essid (net)
Wireless access point ESSID (Linux only)

wireless_freq (net)
Frequency on which device ’net’ is listening (Linux only)

wireless_link_bar (height),(width) (net)
Wireless link quality bar (Linux only)

wireless_link_qual (net)
Wireless link quality (Linux only)

wireless_link_qual_max (net)
Wireless link quality maximum value (Linux only)

wireless_link_qual_perc (net)
Wireless link quality in percents (Linux only)

wireless_mode (net)
Wireless mode (Managed/Ad-Hoc/Master) (Linux only)

words textfile
Displays the number of words in the given file

xmms2_album
Album in current XMMS2 song

xmms2_artist
Artist in current XMMS2 song

xmms2_bar (height),(width)
Bar of XMMS2’s progress

xmms2_bitrate
Bitrate of current song

xmms2_comment
Comment in current XMMS2 song

xmms2_date
Returns song’s date.

xmms2_duration
Duration of current song

xmms2_elapsed
Song’s elapsed time

xmms2_genre
Genre in current XMMS2 song

xmms2_id
XMMS2 id of current song

xmms2_percent
Percent of song’s progress

xmms2_playlist
Returns the XMMS2 playlist.

xmms2_size
Size of current song

xmms2_smart
Prints the song name in either the form "artist - title" or file name, depending on whats available

xmms2_status
XMMS2 status (Playing, Paused, Stopped, or Disconnected)

xmms2_timesplayed
Number of times a song was played (presumably).

xmms2_title
Title in current XMMS2 song

xmms2_tracknr
Track number in current XMMS2 song

xmms2_url
Full path to current song

LUA API

Conky features a Lua Programming API, and also ships with Lua bindings for some useful libraries. Note that the bindings require tolua++, which currently only compiles against Lua 5.1.

To use Lua Conky, you first need to make sure you have a version of Conky with Lua support enabled (‘‘conky -v’’ will report this). Conky defines certain global functions and variables which can be accessed from Lua code running in Conky. Scripts must first be loaded using the lua_load configuration option. You then call functions in Lua via Conky’s $lua, $lua_read, and Lua hooks.

Be careful when creating threaded objects through the Lua API. You could wind up with a whole bunch of threads running if a thread is created with each iteration.

At this time, the Lua API should not be considered stable and may change drastically from one release to another as it matures.

NOTE: In order to accommodate certain features in the cairo library’s API, Conky will export a few additional functions for the creation of certain structures. These are documented below.

`conky_parse(string) function`

This function takes a string that is evaluated as per Conky's TEXT section, and then returns a string with the result.

`conky_set_update_interval(number) function`

Sets Conky's update interval (in seconds) to 'number'.

`conky_window table`

This table contains some information about Conky's window. The following table describes the values contained:

drawable **`Window's drawable (Xlib Drawable), requires Lua extras enabled at compile time.`**
visual **`Window's visual (Xlib Visual), requires Lua extras enabled at compile time.`**
display **`Window's display (Xlib Display), requires Lua extras enabled at compile time.`**
width **`Window width (in pixels).`**
height **`Window height (in pixels).`**
border_inner_margin **`Window's inner border margin (in pixels).`**
border_outer_margin **`Window's outer border margin (in pixels).`**
border_width **`Window's border width (in pixels).`**
text_start_x **`The x component of the starting coordinate of text drawing.`**
text_start_y **`The y component of the starting coordinate of text drawing.`**
text_width **`The width of the text drawing region.`**
text_height **`The height of the text drawing region.`**

NOTE: This table is only defined when X support is enabled.

`conky_info table`

This table contains some information about Conky's internal data. The following table describes the values contained:

update_interval **`Conky's update interval (in seconds).`**
uptime **`System uptime, in seconds.`**

`conky_build_info string`

A string containing the build info for this particular instance of Conky, including the version, build date, and architecture.

`conky_build_date string`

A string containing the build date for this particular instance of Conky.

`conky_build_arch string`

A string containing the build architecture for this particular instance of Conky.

`conky_version string`

A string containing the version of the current instance of Conky.

`conky_config string`

A string containing the path of the current Conky configuration file.

`cairo_text_extents_t:create() function`

Call this function to return a new cairo_text_extents_t structure. A creation function for this structure is not provided by the cairo API. After calling this, you should use tolua.takeownership() on the return value to ensure ownership is passed properly.

`cairo_font_extents_t:create()` function

Call this function to return a new cairo_font_extents_t structure. A creation function for this structure is not provided by the cairo API. After calling this, you should use tolua.takeownership() on the return value to ensure ownership is passed properly.

`cairo_matrix_t:create()` function

Call this function to return a new cairo_matrix_t structure. A creation function for this structure is not provided by the cairo API. After calling this, you should use tolua.takeownership() on the return value to ensure ownership is passed properly.

EXAMPLES

`conky` **`−t ’${time %D %H:%M}’ −o −u 30`**

Start Conky in its own window with date and clock as text and 30 sec update interval.

`conky` **`−a top_left −x 5 −y 500 −d`**

Start Conky to background at coordinates (5, 500).

`conky` **`−C > ˜/.config/conky/conky.conf`**

Do not start Conky, but have it output the builtin default config file to ˜/.config/conky/conky.conf for later customising.

FILES

`${sysconfdir}/conky/conky.conf`

Default system-wide configuration file. The value of ${sysconfdir} depends on the compile-time options (most likely /etc).

`˜/.config/conky/conky.conf`

Default personal configuration file.

BUGS

Drawing to root or some other desktop window directly doesn’t work with all window managers. Especially doesn’t work well with Gnome and it has been reported that it doesn’t work with KDE either. Nautilus can be disabled from drawing to desktop with program gconf-editor. Uncheck show_desktop in /apps/nautilus/preferences/. There is -w switch in Conky to set some specific window id. You might find xwininfo -tree useful to find the window to draw to. You can also use -o argument which makes Conky to create its own window. If you do try running Conky in its own window, be sure to read up on the own_window_type settings and experiment.

SEE ALSO

http://conky.sourceforge.net/

http://www.sourceforge.net/projects/conky

http://wiki.conky.be

#conky on irc.freenode.net

COPYING

Copyright (c) 2005-2012 Brenden Matthews, Philip Kovacs, et. al. Any original torsmo code is licensed under the BSD license (see LICENSE.BSD for a copy). All code written since the fork of torsmo is licensed under the GPL (see LICENSE.GPL for a copy), except where noted differently (such as in portmon and audacious code which are LGPL, and prss which is an MIT-style license).

AUTHORS

The Conky dev team (see AUTHORS for a full list of contributors).