



## Variables

| Value              | Description  |
|--------------------|--|
| allowscriptcontrol | If player can control the game throughout the script's execution (no pockets or further scripts). 0 = no. 1 = yes.   |
| autoobjectcull(no) | If autocull is enabled on an object. 1 = yes. 0 = no.  |
| autoplayercull(no) | IF autocull is enabled on a player. 1 = yes. 0 = no.   |
| autotxtsfx         | If engine generates automatic tones for dialogue boxes. 1= yes with PC speaker emu. 0 = no. 2 = yes (with sfx named "talk"). 3 = yes (with individual player sfx named "talk-player-name\$(no)") |
| autoupdate         | If game automatically checks for updates. 1 = yes. 0 = no. 2 = yes and also it will display a readme file once updated.  |
| awardbannerlocx    | X location of award banner   |
| awardbannerlocy    | Y location of award banner   |
| awardbannerresx    | X resolution of award banner   |
| awardbannerresy    | Y resolution of award banner   |
| awarddisplay       | Number if award to be displayed.   |
| awardgracetime     | Amount of time award is displayed.   |
| awarditemlocx      | X location of award item   |
| awarditemlocy      | Y location of award item   |
| awarditemresx      | X resolution of award item   |
| awarditemresy      | Y resolution of award item   |
| awardmenuno        | Which number award the award menu is displaying.   |
| awardspeed         | Number of pixels per frame the award notification scrolls in by.   |
| awardtextlocx      | X location of award text   |
| awardtextlocy      | Y location of award text   |
| awardvalue(no)     | If award has been completed or not, 0 = no. 1 = yes.   |
| bannerlettercount  | Counts number of letters produced by a text banner.  |

|                        |   |
|------------------------|---|
| bcontrol               | If back control is enabled. 1 = yes. 0 = no.                        |
| bcontrolcode1          | Hardware code for BACK/PAUSE key (option 1)                         |
| bcontrolcode2          | Hardware code for BACK/PAUSE key (option 2)                         |
| bcontrolcode3          | Hardware code for BACK/PAUSE key (option 3)                         |
| bcontrolcode4          | Hardware code for BACK/PAUSE key (option 4)                         |
| bgcurrencycoloura      | Alpha transparency of letter backgrounds in currency.               |
| bgcurrencycolourb      | Blue colour of letter backgrounds in currency.                      |
| bgcurrencycolourg      | Green colour of letter backgrounds in currency.                     |
| bgcurrencycolourr      | Red colour of letter backgrounds in currency.                       |
| bgmenudefaultcoloura   | Alpha transparency of default letter backgrounds in menus.          |
| bgmenudefaultcolourb   | Blue colour of default letter backgrounds in menus.                 |
| bgmenudefaultcolourg   | Green colour of default letter backgrounds in menus.                |
| bgmenudefaultcolourr   | Red colour of default letter backgrounds in menus.                  |
| bgmenuselectcoloura    | Alpha transparency of selected letter backgrounds in menus.         |
| bgmenuselectcolourb    | Blue colour of selected letter backgrounds in menus.                |
| bgmenuselectcolourg    | Green colour of selected letter backgrounds in menus.               |
| bgmenuselectcolourr    | Red colour of selected letter backgrounds in menus.                 |
| bgpocketdefaultcoloura | Alpha transparency colour of default letter backgrounds in pockets. |
| bgpocketdefaultcolourb | Blue colour of default letter backgrounds in pockets.               |
| bgpocketdefaultcolourg | Green colour of default letter backgrounds in pockets.              |
| bgpocketdefaultcolourr | Red colour of default letter backgrounds in pockets.                |
| bgpocketselectcoloura  | Alpha transparency of selected letter backgrounds in pockets.       |
| bgpocketselectcolourb  | Blue colour of selected letter backgrounds in pockets.              |
| bgpocketselectcolourg  | Green colour of selected letter backgrounds in pockets.             |
| bgpocketselectcolourr  | Red colour of selected letter backgrounds in                        |

|                        |   |
|------------------------|---|
|                        | pockets.  |
| bgpromptcoloura        | Alpha transparency of letter backgrounds in developer prompt.   |
| bgpromptcolourb        | Blue colour of letter backgrounds in developer prompt.  |
| bgpromptcolourg        | Green colour of letter backgrounds in developer prompt.   |
| bgpromptcolourr        | Red colour of letter backgrounds in developer prompt.   |
| bgselectobjectcoloura  | Alpha colour of background in select object banner.   |
| bgselectobjectcolourb  | Blue colour of background in select object banner.  |
| bgselectobjectcolourg  | Green colour of background in select object banner.   |
| bgselectobjectcolourr  | Red colour of background in select object banner.   |
| bgspeechcoloura        | Alpha transparency of letter backgrounds in speech.   |
| bgspeechcolourb        | Blue colour of letter backgrounds in speech.  |
| bgspeechcolourg        | Green colour of letter backgrounds in speech.   |
| bgspeechcolourr        | Red colour of letter backgrounds in speech.   |
| bgterminalcoloura      | Alpha transparency of letter backgrounds in terminals.  |
| bgterminalcolourb      | Blue colour of letter backgrounds in terminals.   |
| bgterminalcolourg      | Green colour of letter backgrounds in terminals.  |
| bgterminalcolourr      | Red colour of letter backgrounds in terminals.  |
| carryplayerd(no)       | Carries NPC direction value into memory for next map.   |
| carryplayerjourney(no) | Carries NPC journey progress value into memory for next map.  |
| carryplayerx(no)       | Carries NPC x location into memory for next map.  |
| carryplayery(no)       | Carries NPC y location into memory for next map.  |
| carryvalues            | Tells engine if new map has any values to be carried over from previous map. 1 = carry values. 0 = do not carry values. |
| centreno               | Number of characters in string to be centred  |

|                  |   |
|------------------|---|
| checkpoint(no)   | Value of checkpoint (1-10). 0 = off. 1 = on.  |
| choicearrowl     | X location value of left arrow in choice banner.  |
| choicearrowr     | X location of right arrow in choice banner.   |
| choiceno         | Number of choice selected by player   |
| choicescriptloop | Used to tell spoof trigger that the next script is from a backchoice script command and that it is okay to loop the same script. 0 = off. 1 = on. |
| choicetotal      | Number of choices presented to player in total  |
| clearscreen      | Value used to send drawing sub to clear full screen before drawing. 1 = clearscreen. 0 = no.  |
| collisionstep    | Step for objects marked as collision only. Default is 13.   |
| colpass1         | Used to tell engine if object or players X coordinate is on screen and collision needs applying.  |
| colpass2         | Used to tell engine if object or players Y coordinate is on screen and collision needs applying.  |
| combinex         | X location of "COMBINE" text in pockets.  |
| consolelogging   | If engine logs to console or not. 1 = yes. 0 = no.  |
| ct(no)           | Type of terminal item. 0 = none. 1 = file. 2 = directory. 3 = app.  |
| ctime            | Current length of seconds game has been running as calculated by the itime and TIMER value  |
| currency         | Amount of currency player has   |
| currencychange   | Amount currency is proposed to change   |
| dcontrol         | If walk down control is enabled. 1 = yes. 0 = no.   |
| dcontrolcode1    | Hardware code for DOWN key (option 1)   |
| dcontrolcode2    | Hardware code for DOWN key (option 2)   |
| dcontrolcode3    | Hardware code for DOWN key (option 3)   |
| dcontrolcode4    | Hardware code for DOWN key (option 4)   |
| demofilesamount  | Amount of demo files detected for the demo files menu.  |
| devlogomode      | Tells engine when music on developer logo   |

|                     |  |
|---------------------|--|
|                     | should be played. 1 = before/during fade. 2 = after fade.  |
| devlogono           | Number of developer logos to be displayed.   |
| devmode             | Developer Mode. Enables in-game command prompt. 1 = on. 0 = off.   |
| diagonalmove        | Switch for detecting when player attempts to move diagonally. 1 = diagonal movement. 0 = no diagonal movement.                           |
| direction           | Direction of player character. 1 = UP. 2 = DOWN. 3 = RIGHT. 4 = LEFT.  |
| disablefade         | Disables fade effects for dev purposes. 0 = fade on. 1 = fade off.   |
| displayconsole      | If console messages are displayed or not. 0 = no. 1 = yes.   |
| downloadiconresx    | X resolution of download icon.   |
| downloadiconresy    | Y resolution of download icon.   |
| dplayerx(no)        | Default X position of NPC  |
| dplayery(no)        | Default Y position of NPC  |
| drawpass1           | Used to tell engine if object or players X coordinate is on screen and needs to be drawn.  |
| drawpass2           | Used to tell engine if object or players Y coordinate is on screen and needs to be drawn.  |
| effectani           | Lets effect and displayspeedrun sub know if animation is running to change draw order. 0 = default draw order. 1 = animation draw order. |
| enablemapoffsets    | If map animation offsets are activated. 0 = no. 1 = yes.   |
| enableobjectoffsets | If object animation offsets are activated. 0 = no. 1 = yes.  |
| enableplayeroffsets | If player animation offsets are activated. 0 = no. 1 = yes.  |
| endframes           | Total number of frames in a speedrun.  |
| endofspeedrunfile   | Tells engine if a speedrun file has reached the end. 1 = yes. 0 = no.  |
| erasesaveonly       | Tells engine if a save erase only load has been requested by parameter. 1 = yes. 0 = no.   |
| errorcrash          | Set to 1 if engine has crashed beyond recovery.  |
| eventnumber         | Extra console event data   |
| exitsave            | Tells engine if to save game on exit. 0 = no. 1 = yes.   |
| extrahertz          | Loop limiter for parts of the game that don't require frames to be pushed.   |
| fadeinmusic         | Number of music file currently being faded in.   |
| fadeoutmusic        | Number of music file currently being faded out.  |
| fadespeed           | How much the fade increases or decreases by per cycle.   |

|                    |   |
|--------------------|---|
| fadestatus         | Status of fade effect. 0 = off. 1 = fade out on. 2 = slight fade out on.                |
| fading             | If a fade effect is currently occurring. 1 = yes. 0 = no.                               |
| fixvame            | Used to tell engine to force an update.   |
| fontbuffer         | Number of pixels given at the end of a letter in text banners.                          |
| fontmode           | Used to tell engine what font is in use. 1 = header. 2 = default. 3 = small.            |
| fontsize           | Size of font  |
| footpace           | Pace of player foot changing whilst walking.  |
| forcefullscreen    | Forces VaME to launch in fullscreen mode. 1 = yes. 0 = no.                              |
| forcewindowed      | Forces VaME to launch in windowed mode. 1 = yes. 0 = no.                                |
| fps                | Calculated frames per second  |
| framecount         | Number of frames displayed in speedrun info.  |
| frames             | Counts number of frames pushed to screen  |
| gametime           | Amount of time save file has been played.   |
| givex              | X location of "GIVE" text in pockets.   |
| hertz              | Loops per second / fps engine performs. Set to 0 if uncapped.                           |
| hud                | Developer HUD type.   |
| ifaward            | Number of award being checked   |
| ifawardno          | Number of times in script chain the 'ifaward' command has been run                      |
| ifawardresult      | Result of award checking. 1 = award granted. 0 = award not granted.                     |
| ifcheckpointno     | Number of times a 'ifcheckpoint' command has been run during a script                   |
| ifcheckpointresult | Result of ifcheckpoint enquiry. 1 = checkpoint active. 0 = checkpoint inactive.         |
| ifcurrencyamount   | Amount of currency ifcurrency should check.   |
| ifcurrencyno       | Number of times a 'ifcurrency' command has been run during a script                     |
| ifcurrencyresult   | Result of ifcurrency enquiry. 1 = enough currency found. 0 = not enough currency found. |
| ifdirectionno      | Number of times a 'ifdirection' command has been run during a script                    |
| ifgone             | Tels engine if requested pocket item has been marked "gone forever" or not. 2 = item    |

|                 |   |
|-----------------|---|
|                 | gone. (anything else) = item available.   |
| ifgoneno        | Number of times a 'ifgone' command has been run during a script   |
| ifholding       | Result of ifholding enquiry. 1 = item found. 0 = item not found.  |
| ifholdingno     | Number of times a 'ifholding' command has been run during a script  |
| ifmapno         | Tells engine what map number to query.  |
| ifmapnono       | Number of times a 'ifmapno' command has been run during a script  |
| ifmapnoresult   | Tells engine if the mapno request is successful or not. 1 = on requested map. 0 = not on requested map.   |
| ifmodel         | Tells engine if the 'ifmodel' request is successful or not. 0 = no. 1 = yes.  |
| ifmodelno       | Number of times a 'ifmodel' command has been run during a script  |
| ifpocket        | Result of ifpocket enquiry. 1 = item found. 0 = item not found.   |
| ifpocketno      | Number of times a 'ifpocket' command has been run during a script   |
| ifrandomno      | Number of times a 'ifrandom' command has been run during a script   |
| ifrandomresult  | Result of ifrandom enquiry. 1 = conditions met. 0 = conditions not met.   |
| ifvalueno       | Number of times ifvalue command has been run in script chain  |
| ifvalueresult   | Result of ifvalue command. 0 = no. 1 = yes.   |
| igametime       | Initial amount of time played in save file when loaded.   |
| imode           | What to do with the letter I in text boxes. 0 = off, letter I is not adjusted. 1 = adjustment for lowercase I. 2 = adjustment for both lowercase and uppercase I. |
| installtype     | What type of install the game has. 1 = multi platform. 2 = platform specific.   |
| iscripttimer    | Initial ctime when script timer is set.   |
| itime           | Initial TIMER value when game launches  |
| keypresseddown  | If DOWN key is registered as being pressed. 0 = not pressed. 1 = pressed.   |
| keypressedleft  | If LEFT key is registered as being pressed. 0 = not pressed. 1 = pressed.   |
| keypressedright | If RIGHT key is registered as being pressed. 0 = not pressed. 1 = pressed.  |
| keypressedup    | If UP key is registered as being pressed. 0 =   |



|                         |  |
|-------------------------|--|
|                         | not pressed. 1 = pressed.                                |
| lcontrol                | If walk left control is enabled. 1 = yes. 0 = no.        |
| lcontrolcode1           | Hardware code for LEFT key (option 1)                    |
| lcontrolcode2           | Hardware code for LEFT key (option 2)                    |
| lcontrolcode3           | Hardware code for LEFT key (option 3)                    |
| lcontrolcode4           | Hardware code for LEFT key (option 4)                    |
| letcurrencycoloura      | Alpha transparency of letters in currency.               |
| letcurrencycolourb      | Blue colour of letters in currency.                      |
| letcurrencycolourg      | Green colour of letters in currency.                     |
| letcurrencycolourr      | Red colour of letters in currency.                       |
| letmenudefaultcoloura   | Alpha transparency of default letters in menus.          |
| letmenudefaultcolourb   | Blue colour of default letters in menus.                 |
| letmenudefaultcolourg   | Green colour of default letters in menus.                |
| letmenudefaultcolourr   | Red colour of default letters in menus.                  |
| letmenuselectcoloura    | Alpha transparency of selected letters in menus.         |
| letmenuselectcolourb    | Blue colour of selected letters in menus.                |
| letmenuselectcolourg    | Green colour of selected letters in menus.               |
| letmenuselectcolourr    | Red colour of selected letters in menus.                 |
| letpocketdefaultcoloura | Alpha transparency colour of default letters in pockets. |
| letpocketdefaultcolourb | Blue colour of default letters in pockets.               |
| letpocketdefaultcolourg | Green colour of default letters in pockets.              |
| letpocketdefaultcolourr | Red colour of default letters in pockets.                |
| letpocketselectcoloura  | Alpha transparency of selected letters in pockets.       |
| letpocketselectcolourb  | Blue colour of selected letters in pockets.              |
| letpocketselectcolourg  | Green colour of selected letters in pockets.             |
| letpocketselectcolourr  | Red colour of selected letters in pockets.               |

|                        |  |
|------------------------|--|
| letpromptcoloura       | Alpha transparency of letters in developer prompt.   |
| letpromptcolourb       | Blue colour of letters in developer prompt.  |
| letpromptcolourg       | Green colour of letters in developer prompt.   |
| letpromptcolourr       | Red colour of letters in developer prompt.   |
| letselectobjectcoloura | Alpha colour of letters in select object banner.   |
| letselectobjectcolourb | Blue colour of letters in select object banner.  |
| letselectobjectcolourg | Green colour of letters in select object banner.   |
| letselectobjectcolourr | Red colour of letters in select object banner.   |
| letspeechcoloura       | Alpha transparency of letters in speech.   |
| letspeechcolourb       | Blue colour of letters in speech.  |
| letspeechcolourg       | Green colour of letters in speech.   |
| letspeechcolourr       | Red colour of letters in speech.   |
| letterminalcoloura     | Alpha transparency of letters in terminals.  |
| letterminalcolourb     | Blue colour of letters in terminals.   |
| letterminalcolourg     | Green colour of letters in terminals.  |
| letterminalcolourr     | Red colour of letters in terminals.  |
| liteload               | Tells engine if a lighter load has been requested by parameter. 1 = yes. 0 = no.   |
| loadassets             | Number of assets to be loaded in (for load bar use)  |
| loadbarsize            | Amount load bar should increase with every loaded asset  |
| loadiconresx           | X resolution of load icon  |
| loadiconresy           | Y resolution of load icon  |
| lookx                  | X location of "LOOK" text in pockets.  |
| mainmenu               | Used to tell engine if main menu is active. 1 = yes. 0 = no.   |
| mapaniooffset          | Number of ms player animation is offset by.  |
| mapeffect              | Effects (if any) used on map. 0 = none. 1 = dark. 2 = rain. 3 = stormy. 4 = torch. 5 = dark fade (right). 6 = dark fade (left). 7 = dark fade (down). 8 = dark fade (up). 9 = pitch black. |
| mapno                  | Map metadata number. Tells engine which map data to load   |
| mapobjectno            | Number of objects on map, stored in map metadata.  |

|                 |   |
|-----------------|---|
| mapplayerno     | Number of NPC's on map, stored in map metadata.   |
| mapscript       | Whether script is assigned to map or loose or combination. 1 = map object/player script. 2 = pocket combination. 3 = pocket look script. 4 = timer script. 5 = system script. |
| maptriggerno    | Number of triggers on map, stored in map metadata.  |
| mapx            | Width of current map spirte   |
| mapy            | Hieght of current map sprite  |
| mcy(no)         | Y location of menu choice   |
| menunos         | Number of items in menu   |
| menuposx        | X position of text on menus. Set to -1 for centralised text.  |
| modamount       | Amount of mods found for mod menu.  |
| modrunning      | If mod is enabled. 1 = yes. 0 = no.   |
| mouse           | Mouse input.  |
| mpfoot          | Current foot of player walking. Set at either 1 or 2.   |
| mpfootloop      | Keeps main player foot changing.  |
| mpidle          | Number of seconds engine must wait before main player begins idle animation   |
| mplayerx(no)    | End X location for NPC  |
| mplayery(no)    | End Y location for NPC  |
| mpnote(no)      | Frequency of main player talk noise.  |
| mpposx          | Main player X position  |
| mpposy          | Main player Y position  |
| mps             | Main player "step"  |
| mpwalking       | Value switch if main player is walking. 1 = walking. 0 = not walking.   |
| mpx             | Width of main player character  |
| mpy             | Height of main player character   |
| musicfadechange | Amount music fades per frame  |
| musicfadein     | If music is being faded in. 0 = no. 1 = yes.  |
| musicfadeinvol  | Current volume of music being faded in.   |
| musicfadeout    | If music is being faded out. 1 = fading out. 0 = not.   |
| musicfadeoutvol | Current volume of music being faded out.  |

|                     |  |
|---------------------|--|
| musicpause          | Used to tell engine if music is paused. 1 = paused. 0 = playing.   |
| musictransitionmode | What the music does automatically when a track change is requested. 1 = immediate cut. 2 = fade.                             |
| musicvol            | Tells engine what volume music is at.  |
| nextmapscript       | Map script value of any scripts triggered by an already running script.  |
| noclip              | Developer value to turn off object collision. 1 = on. 0 = off.   |
| nodraw              | Disables screen drawing for one frame. 0 = screen draws. 1 = screen drawing disabled.  |
| nosave              | Tells engine if a no save load has been requested by parameter. 1 = yes. 0 = no.   |
| noupdate            | Tells engine if update check should be skipped or not. 1 = yes. 0 = no.  |
| objectcollision(no) | Custom collision for object. 1 = Collision on. 0 = Collision off.  |
| objecthighlight(no) | If an object is being highlighted by the selectobject banner. 1 = yes. 0 = no.   |
| objectl             | Used to tell draw system which layer to draw. 1 = before mainplayer. 2 = aftermainplayer.                                    |
| objectl(no)         | Assigned to tell draw system which layer object is on. 1 = before mainplayer. 2 = aftermainplayer. 3 = don't draw.           |
| objectlayer(no)     | Used if an object needs forcing onto a layer. 0 = no force / automatic layering. 1 = force layer 1. 2 = force layer 2.       |
| objectoffset(no)    | Number of ms object animation is offset by.  |
| objectresx(no)      | X Resolution for object  |
| objectresy(no)      | Y Resolution for object  |
| objects(no)         | Used to tell object collision how many Y pixels above centre screen to stop player. This allows for different object depths. |
| objectspeed(no)     | Speed multiplier for object. Default is 1.   |
| objectstep          | Used to tell object collision how many X pixels to allow "wriggle room" for. Usually set at 2.                               |
| objectx(no)         | X Location for object  |
| objecty(no)         | Y Location for object  |
| oldmapno            | Old map number   |
| oldpocketdisplay    | Old pocketdisplay value to check if pocket item displayed has changed.   |
| oldsaveamount       | Amount of erased saves counted in save folder.   |

|                     |   |
|---------------------|---|
| oposx               | Previous X axis position of player/background   |
| oposy               | Previous Y axis position of player/background   |
| pace                | Speed of scrolling game world (or character speed)  |
| parallaxmode        | What parallax mode the map loads. 0 = off. 1 = background. 2 = foreground.  |
| parallaxoffset      | Randomly generated animation offset   |
| pcontrol            | If pockets control is enabled. 1 = yes. 0 = no.   |
| pcontrolcode1       | Hardware code for POCKETS key (option 1)  |
| pcontrolcode2       | Hardware code for POCKETS key (option 2)  |
| pcontrolcode3       | Hardware code for POCKETS key (option 3)  |
| pcontrolcode4       | Hardware code for POCKETS key (option 4)  |
| pfoot(no)           | Tells engine which foot NPC is on when walking.   |
| pfootloop(no)       | Tells engine when to switch NPC foot whilst walking.  |
| playercollision(no) | Custom collision for player. 1 = Collision on. 0 = Collision off.   |
| playerd(no)         | Direction of NPC. 1 = UP. 2 = DOWN. 3 = RIGHT. 4 = LEFT   |
| playerdefault(no)   | Default direction of NPC. 1 = UP. 2 = DOWN. 3 = RIGHT. 4 = LEFT. -1 = FOLLOWS MAIN PLAYER.  |
| playergrace(no)     | Number of seconds engine must wait before moving NPC. Can be set to -1 to bypass return journey or -2 to bypass journey to destination. |
| playerhighlight(no) | If a player is being highlighted by the select-object banner. 1 = yes. 0 = no.  |
| playeridle          | Number of seconds engine must wait before players begin idle animation (effects all players including NPCs)                             |
| playerjourney(no)   | Tells engine what stage of idle walk NPC is at. 1 = first stage (walk away). 2 = second stage (walk back).                              |
| playerlayer(no)     | Assigned to tell draw system which layer player is on. 1 = before mainplayer. 2 = after-mainplayer.                                     |
| playerlayer2(no)    | Used if a player needs forcing onto a layer. 0 = no force / automatic layering. 1 = force layer 1. 2 = force layer 2.                   |
| playernote(no)(no)  | Frequency of NPC talk noise.  |
| playeroffset(no)    | Number of ms player animation is offset by.   |
| playerperiod(no)    | Number of seconds left before player moves.   |
| players(no)         | Used to tell object collision how many Y  |

|                   |  |
|-------------------|--|
|                   | pixels above centre screen to stop player. This allows for different object depths.                      |
| playerscript(no)  | If NPC is being controlled by a script or not. 1 = YES. 0 = NO.  |
| playerspeed(no)   | Speed multiplier for player. Default is 1.   |
| playerwalkdivide  | Tells engine to divide all NPC default walking speed (pace) by an amount.                                |
| playerwalking(no) | If NPC is walking or not.  |
| playerx(no)       | Starting X location for NPC  |
| playery(no)       | Starting Y location for NPC  |
| pocketarrowllocx  | X location of left pocket arrow  |
| pocketarrowllocy  | Y location of left pocket arrow  |
| pocketarrowresx   | X resolution of pocket arrow   |
| pocketarrowresy   | Y resolution of pocket arrow   |
| pocketarrowrlocx  | X location of right pocket arrow   |
| pocketarrowrlocy  | Y location of right pocket arrow   |
| pocketbannerresx  | X resolution of pocket banner  |
| pocketbannerresy  | Y resolution of pocket banner  |
| pocketcarry       | Number of pocket items player is carrying.   |
| pocketdisplay     | ID number of pocket value being displayed in inventory   |
| pocketdivert      | If pocket has been diverted to a script. 1 = yes. 0 = no.  |
| pockethudanispeed | Speed of pocket hud animation  |
| pockethudresx     | X resolution of pocket hud   |
| pockethudresy     | Y resolution of pocket hud   |
| pocketitem(no)    | If mainplayer has an item in their pocket or not. 0 = not picked up. 1 = picked up. 2 = given away/used. |
| pocketline        | What line in pocketfile pocket sub is reading  |
| pocketnos         | Number of pocket items loaded in game  |
| pocketon          | If pocket sub is active or not. 1 = yes. 0 = no.   |
| pocketredraw      | If pockets need redrawing. 0 = no redraw. 1 = redraw.  |

|                   |   |
|-------------------|---|
| pocketselectresx  | X resolution of pocket select   |
| pocketselectresy  | Y resolution of pocket select   |
| pocketselectx     | X position of pocket select   |
| pocketselecty     | Y position of pocket select   |
| pocketspriteresx  | X resolution of pocket item sprite  |
| pocketspriteresy  | Y resolution of pocket item sprite  |
| pocketspritex     | X location of pocket item sprite  |
| pocketspritey     | Y location of pocket item sprite  |
| pocketvisible(no) | Used to tell engine if a pocket item is visible or not. 1 = visible. 0 = not visible. |
| posx              | Current X axis position of player/background  |
| posy              | Current Y axis position of player/background  |
| promptquit        | Set to 1 if command prompt requests game to be quit.                                  |
| rainspread        | Range of rain.  |
| rainx(no)         | X co-ordinate of rain   |
| rainy(no)         | Y co-ordinate of rain   |
| randomscriptvalue | Current random script value   |
| rcontrol          | If walk right control is enabled. 1 = yes. 0 = no.                                    |
| rcontrolcode1     | Hardware code for RIGHT key (option 1)  |
| rcontrolcode2     | Hardware code for RIGHT key (option 2)  |
| rcontrolcode3     | Hardware code for RIGHT key (option 3)  |
| rcontrolcode4     | Hardware code for RIGHT key (option 4)  |
| resx              | X value of resolution   |
| resy              | Y value of resolution   |
| runterminal       | If terminal is to run. 1 = terminal running. 0 = no terminal.                         |
| runupdate         | Lets engine know if an update is scheduled. 0 = no. 1 = yes.                          |

|                       |   |
|-----------------------|---|
| saveiconresx          | X resolution of save icon   |
| saveiconresy          | Y resolution of save icon   |
| savetimehour          | Amount of hours the current save file has been played   |
| savetimemin           | Amount of mins the current save file has been played  |
| savetimesec           | Amount of seconds the current save file has been played   |
| scontrol              | If select control is enabled. 1 = yes. 0 = no.  |
| scontrolcode1         | Hardware code for SELECT key (option 1)   |
| scontrolcode2         | Hardware code for SELECT key (option 2)   |
| scontrolcode3         | Hardware code for SELECT key (option 3)   |
| scontrolcode4         | Hardware code for SELECT key (option 4)   |
| screenmode            | Fullscreen toggle. 1 = fullscreen, 2 = window   |
| scriptimageresx       | X resolution for script image   |
| scriptimageresy       | Y resolution for script image   |
| scriptrun             | Used to tell engine if script is running. 1 = yes. 0 = no.  |
| scriptskip            | IF script should be put into skip mode. 0 = np, run script normally. 1 = yes, skip was requested. |
| scripttime            | Current amount of time the script timer has been counting.  |
| scripttimer           | Amount of time set before a script will run.  |
| scriptvalue(no)       | Custom script values.   |
| scriptwaittime        | Amount of time that the script must wait until it can continue.                                   |
| scriptwalkdivide      | Tells engine to divide all script walking speeds by an amount.                                    |
| selectobjecthighlight | If selected world object highlighter is on. 1 = on. 0 = off.                                      |
| setupboot             | Used to determine if engine is setting up. 1 = yes. 0 = no.                                       |
| setuprecord           | If a demo recording is being set up. 0 = no. 1 = yes.   |
| sfxvol                | Tells engine what volume sfx is at.   |
| silentgive            | Tells engine if giving of items/currency is silent or not. 0 = not silent. 1 = silent.            |



|                |  |
|----------------|--|
| silenttake     | Tells engine if taking of items/currency is silent or not. 0 = not silent. 1 = silent.                   |
| sitime         | itime for save time  |
| skipallowed    | If the script is permitted to be skipped. 1 = yes. 0 = no.   |
| soundmode      | Used to tell engine what sounds/music to play. 1 = off. 2 = music and sfx. 3 = music only. 4 = sfx only. |
| speedrun       | Used to tell what mode the speedrun is on. 0 = off. 1 = on. 2 = complete.                                |
| speedrunhour   | Number of hours speedrun has been going.   |
| speedrunmin    | Number of minutes speedrun has been going.   |
| speedrunsec    | Number of seconds speedrun has been going.   |
| stime          | ctime for save time  |
| stposx         | X position of terminal selector  |
| stposy         | Y position of terminal selector  |
| sysstat        | What the terminal face in terminal text files displays. 1 = ok. 2 = busy. 3 = err.                       |
| tanidelay      | Delay amount for animation in terminal   |
| tdelay         | Delay amount for terminal  |
| temp(number)   | Temporary values, often used for calculation. Scrubbed at end of sub or at end of use.                   |
| tempmusiccut   | Switches the musictransitionmode on a temp basis for instances of stopping and starting music instantly  |
| tempmusicfade  | Switches the musictransitionmode on a temp basis for instances of fading music out and in                |
| terminalcol1   | Location of first column in terminal   |
| terminalcol2   | Location of second column in terminal  |
| terminalcol3   | Location of third column in terminal   |
| terminaldir    | If terminal is in a directory or not. 1 = yes. 2 = just exited directory. 0 = no.                        |
| terminalfacex  | X location of terminal face/status logo  |
| terminalfacey  | Y location of terminal face/status logo  |
| terminalnoboot | If terminal launches with boot GUI. 0 = launch normally. 1 = launch with no boot gui.                    |

|                    |  |
|--------------------|--|
| terminalrow1       | Location of first row in terminal  |
| terminalrow2       | Location of second row in terminal   |
| textbannerfaceresx | X resolution of text banner face   |
| textbannerfaceresy | Y resolution of text banner face   |
| textbannerfacey    | Y location of text banner face   |
| textbannerresx     | X resolution of text banner  |
| textbannerresy     | Y resolution of text banner  |
| textbannersound    | If engine plays “select” sound when using text dialogues. 0 = no. 1 = yes.               |
| totalcheckpoints   | Total number of checkpoints supported in game.   |
| totalframes        | Total number of animation frames supported per animation.                                |
| totalmusics        | Total number of music files supported in game.   |
| totalobjects       | Total number of objects supported per map.   |
| totalplayers       | Total number of players supported per map.   |
| totalpockets       | Total number of pocket items supported in game.  |
| totalscriptvalues  | Total number of script values  |
| totalsfxs          | Total number of sfx files supported in game.   |
| totaltriggers      | Total number of triggers supported per map.  |
| triggera(no)       | Used to tell engine if trigger has been collided with. 1 = triggered. 0 = not triggered. |
| triggerd(no)       | Used to tell engine if trigger is considered active. 1 = active. 0 = not active.         |
| triggerx(no)1      | First X value of trigger   |
| triggerx(no)2      | Second X value of trigger  |
| triggery(no)1      | First Y value of trigger   |
| triggery(no)2      | Second Y value of trigger  |
| ttype              | Terminal item type (selected) 0 = none. 1 = file. 2 = directory. 3 = app.                |
| ucontrol           | If walk up control is enabled. 1 = yes. 0 = no.  |

|                |   |
|----------------|---|
| ucontrolcode1  | Hardware code for UP key (option 1)   |
| ucontrolcode2  | Hardware code for UP key (option 2)   |
| ucontrolcode3  | Hardware code for UP key (option 3)   |
| ucontrolcode4  | Hardware code for UP key (option 4)   |
| userquit       | Used to tell engine if user has requested quit.<br>1 = user quit. 0 = none. |
| usex           | X location of "USE" text in pockets.  |
| valuesaveno    | Number of in-game value to be saved.  |
| variablelength | Length of variable command.   |
| xxit           | Triggers end game sequence  |
| restoremenu    | If save restore menu is currently open or not.<br>1 = yes. 0 = no.          |
| seperateval1   | Used to separate script values  |
| seperateval2   | Used to separate script values  |

---

## String Values

| String                  | Description   |
|-------------------------|---|
| action\$                | Prompt Command (see prompt appendix)                                  |
| aloc\$                  | Location of animation frames folder                                   |
| anifile\$               | Name of animation metadata file.                                      |
| anisprite\$             | Sprite being replaced by animation.                                   |
| awardarrowleft\$        | Name of award left arrow file.  |
| awardarrowright\$       | Name of award right arrow file.                                       |
| awardarrowselectleft\$  | Name of award selected left arrow file.                               |
| awardarrowselectright\$ | Name of award selected right arrow file.                              |
| awardbanner\$           | Name of award banner file   |
| awarddescription(no)\$  | Description of award.   |
| awardname(no)\$         | Name of award.  |
| awardnone\$             | Name of missing award file.   |
| awardnotification\$     | The text that award notifications say                                 |
| awardtitle\$            | Title given to awards   |
| bannercharacter\$       | Current character being printed to screen in text and choice banners. |
| bannertemp\$            | Used to scour text banner values for any system variable requests.    |
| centretext\$            | Text to be centered   |
| changelog\$             | Name of game change log file for updater use.                         |
| choicebanner\$          | Filename of choice banner for player choices                          |
| choicename\$(no)        | Name of choice for player choices.                                    |
| cn(number)\$            | Name of terminal item.  |
| consolelog\$            | Location of console/error log... "data/consolelog.txt"                |
| currencyname\$          | Name of the currency used in the game                                 |
| currentmusic\$          | Name of music file currently playing                                  |
| currentpocketshort\$    | Short identifier name for current pocket item                         |
| datafolder\$            | Name of game data folder for updater use.                             |
| devlogo\$               | Name of developer logo graphic.                                       |
| dloc\$                  | Location of main metadata folder                                      |
| downloadicon\$          | Filename of download icon   |

|                      |  |
|----------------------|--|
| downloadlink\$       | Download link for update files.  |
| emptypockets\$       | Speech the mainplayer says when you have nothing in your pockets.        |
| engineversionno\$    | Build version number for engine.   |
| errdescription\$     | Description of error that occurred.                                      |
| eventdata\$          | Extra data for console event   |
| eventtitle\$         | Title of console event   |
| filename\$           | Name of executable file.   |
| floc\$               | Location of fonts metadata folder  |
| fontname\$           | Name of font (including ext)   |
| fontstyle\$          | Defines font style can equal "BOLD" "ITALICS" or "UNDERLINED" or nothing |
| giveaction\$         | Text displayed for "GIVE" action in pockets.                             |
| giveitem\$           | Name of item to be given to main player                                  |
| hardbuild\$          | Hard coded engine version number to compare with metadata.               |
| hideitem\$           | Name of proposed pocket item to hide.                                    |
| ifgone\$             | Pocket item engine is to check if marked "gone forever".                 |
| ifholding\$          | Checks is mainplayer is holding an item                                  |
| ifmodel\$            | Character model to check.  |
| ifpocket\$           | Checks pocket for item   |
| lastconsoleline\$    | Last line sent to the console printer.                                   |
| lnxexe\$             | Name of game Linux executable for updater use.                           |
| loadbar\$            | Name of loading bar graphic file.  |
| loadicon\$           | Filename of load icon.   |
| lookaction\$         | Text displayed for "LOOK" action in pockets.                             |
| mainplayerlongname\$ | Long name of main player   |
| manual\$             | Name of game manual for updater use.                                     |
| mapdir\$             | Name of current map directory  |
| mapfile\$            | Name of current map file   |
| mapname\$            | Name of current map  |
| mapscriptdir\$       | Name of script folder  |
| menu\$               | Name of menu file to be read   |
| menubackdrop\$       | Name of image used for main menu background.                             |
| menuchoice(no)\$     | Name of choice in menu.  |
| menucommand(no)\$    | Name of command in menu  |

|                         |   |
|-------------------------|---|
| menumusic\$             | Name of music slot for menu   |
| mloc\$                  | Location of map metadata folder   |
| moddingname\$           | What the engine refers to mods as. (ie. could be set as something like DLC or expansions) |
| modloadprompt\$         | Text that displays before loading a mod.  |
| modmenubackdrop\$       | Filename of mod menu backdrop   |
| modname\$               | Name of mod that is running.  |
| mplayermodel\$          | Name of character that player controls  |
| museloc\$               | Location of music metadata folder   |
| musicfile\$             | Name of music file being loaded   |
| newversionno\$          | Version number of potential new update.   |
| nospeedrunfilesprompt\$ | Tells player that no speedrun files were found.   |
| objectlongname(no)\$    | Used to tell engine what the name of the object is, no shortening or weird code names.    |
| objectname(no)\$        | Name of object  |
| objecttype\$            | Tells engine if object is NPC or Object or neither. Equals either "NPC" or "OBJ" or "NON" |
| oldmapname\$            | Name of old map name  |
| oldmplayermodel\$       | Name of old main player model   |
| oldmusic\$              | Name of old music file previously played  |
| oldsavename\$           | Display name for old saves.   |
| oloc\$                  | Location of map objects folder  |
| parentdir\$             | Previous terminal metadata file (for if you're in a directory)                            |
| playerlongname\$(no)    | Long name of player or NPC  |
| playmusic\$             | Name of proposed music file to be played  |
| playsfx\$               | Name of proposed sound effect to be played  |

|                                |   |
|--------------------------------|---|
| ploc\$                         | Location of player metadata folder                              |
| pocketarrowleft\$              | Name of left arrow pocket sprite (dormant)                      |
| pocketarrowright\$             | Name of right arrow pocket sprite (dormant)                     |
| pocketarrowselectleft\$        | Name of left arrow pocket sprite (selected)                     |
| pocketarrowselectright\$       | Name of right arrow pocket sprite (selected)                    |
| pocketarrowunavailableleft\$   | Name of left pocket arrow sprite (unavailable)                  |
| pocketarrowunavailablelright\$ | Name of right pocket arrow sprite (unavailable)                 |
| pocketbanner\$                 | Name of pocket banner sprite                                    |
| pocketdisplaydescription\$     | Description of item being displayed in pockets                  |
| pocketdisplayname\$            | Name of item being displayed in pockets                         |
| pocketfile\$                   | Name of pocket sprite to be loaded or unloaded                  |
| pockethudimage\$               | Name used for image used for pockets.                           |
| pocketloc\$                    | Location of pocket metadata folder                              |
| pocketname(no)\$               | Name of pocket item   |
| pocketselect\$                 | Name of pocket select sprite                                    |
| pocketshort(no)\$              | Short identifier name for pocket item                           |
| prompt\$                       | Prompt Command (player input) (see prompt appendix)             |
| proposedobject\$               | Name of object or player that engine is checking collision for. |
| readme\$                       | Name of game readme file for updater use.                       |
| restoremenubackdrop\$          | Filename of restore save menu backdrop                          |
| ros\$                          | Client Operating System. Either “win” or “lnx”                  |
| runterminal\$                  | Name of terminal file to be run                                 |
| saveicon\$                     | Filename of save icon   |
| scriptimage\$                  | Name for image used to notify player script is running.         |

|                     |  |
|---------------------|--|
| scriptline\$        | Line of script currently being executed.   |
| scriptloc\$         | Location of Stage Director Script Utility scripts folder   |
| scriptname\$        | Used to tell engine what the filename of the script is.  |
| scripttimername\$   | Name of script to be run once script timer depletes.   |
| selectobject\$      | Name of current selected object within the world   |
| selectobject\$      | Name of object or player that has collided with mainplayer                                       |
| selectobjectlong\$  | Long name of the selected object.  |
| sfxfile\$           | Name of sound effect being loaded  |
| sfxloc\$            | Location of sound effects files  |
| showimage\$         | Name of image to be displayed on screen.   |
| showitem\$          | Name of proposed pocket item to show.  |
| sloc\$              | Location of player saves metadata folder   |
| source\$            | Name of game source code for updater use.  |
| spofoption1\$       | Spoof option 1   |
| spofoption1result\$ | Text result of selecting spoof option 1  |
| spofoption2\$       | Spoof option 2   |
| spofoption2result\$ | Text result of selecting spoof option 2  |
| spofoptiontitle\$   | Question asked when spoof option selected  |
| takeitem\$          | Name of item to be taken from main player  |
| temp(number)\$      | Temporary value. Wiped blank at end of sub   |
| terminalhold\$      | Temp value container for transferring item holding information between pocket and terminal subs. |
| textbanner\$        | Name of text banner sprite   |
| textbannername\$    | Image used for text banner name background.  |
| textline\$          | Line of text used in text banner   |



|                   |  |
|-------------------|--|
| textspeech\$      | Text used in text banner   |
| texttemp1\$       | Used to split text banner values so words can be replaced.   |
| texttemp2\$       | Used to split text banner values so words can be replaced.   |
| title\$           | Title bar text   |
| tloc\$            | Location of terminal metadata folder   |
| torcheffectfile\$ | Filename of torch effect overlay.  |
| tos\$             | Name of terminal OS  |
| triggername(no)\$ | Used to tell engine what script to run when trigger is collided with.  |
| tselect\$         | Name of terminal item (selected).  |
| txtfile(no)\$     | Line to displayed in terminal text files.  |
| uiloc\$           | Location of UI metadata folder   |
| unziplink\$       | Download link for windows unzip tool.  |
| updatelink\$      | Link to check for updates.   |
| updaterlinklnx\$  | Download link for linux updater.   |
| updaterlinkwin\$  | Download link for windows updater.   |
| updatesource\$    | Name of game updater source code for up-dater use.   |
| updatezip\$       | Name of game updater zip file, minus zip extension. For updater use only.  |
| useaction\$       | Text displayed for “USE” action in pockets.  |
| value\$           | Prompt Command Argument (see prompt appendix)  |
| variablevalue\$   | Command for variable   |
| varworkingtemp\$  | Temporary value for working out of variable commands   |
| versionno\$       | Build version number. Include “DEV” (anywhere) to mark as a developer build to block updates. Include “3RDPARTY” (anywhere) to mark as a build intended for 3 <sup>rd</sup> party stores to block updates. |
| winexe\$          | Name of game Windows executable for up-dater use.  |

|             |  |
|-------------|--|
| seperate1\$ | Used to separate script commands               |
| seperate2\$ | Used to separate script commands               |
| seperate3\$ | Used to separate script commands               |
| temps\$(no) | Current script command separated into an array |

---

## Metadata Index

**Filename** - engine.ddf

**Description** - Engine setup values.

**Location** - dloc\$

**Mode** - Input Only

**Values** - devmode, consolelogging, displayconsole, autoupdate, installtype, devlogono, selectobjecthighlight, musictransitionmode, musicfadechange, title\$, filename\$, totalobjects, totalplayers, totaltriggers, totalpockets, totalcheckpoints, totalframes, totalsfxs, totalmusics, totalscriptvalues, totalawards, resx, resy, hertz, extrahertz, exitsave, autotxtsfx, ucontrol, dcontrol, lcontrol, rcontrol, scontrol, pcontrol, bcontrol, ucontrolcode1, ucontrolcode2, ucontrolcode3, ucontrolcode4, dcontrolcode1, dcontrolcode2, dcontrolcode3, dcontrolcode4, lcontrolcode1, lcontrolcode2, lcontrolcode3, locontrolcode4, rcontrolcode1, rcontrolcode2, rcontrolcode3, rcontrolcode4, scontrolcode1, scontrolcode2, scontrolcode3, scontrolcode4, pcontrolcode1, pcontrolcode2, pcontrolcode3, pcontrolcode4, bcontrolcode1, bcontrolcode2, bcontrolcode3, bcontrolcode4, enableobjectoffsets, enableplayeroffsets, enablemapoffsets, fadespeed, pace, objectstep, collisionstep, playeridle, footpace, fontname\$, fontsize, fontstyle\$, fontbuffer, imode, playerwalkdivide, scriptwalkdivide, scriptimage\$, scriptimageresx, scriptimageresy, pockethudimage\$, pockethudresx, pockethudresy, pocketarrowright\$, pocketarrowleft\$, pocketarrowselectright\$, pocketarrowselectleft\$, pocketarrowunavailableright\$, pocketarrowunavailableleft\$, pocketarrowresx, pocketarrowresy, pockethudanispeed, pocketarrowrlocx, pocketarrowrlocy, pocketarrowllocx, pocketarrowllocy, pocketspritex, pocketspritey, pocketspriteresx, pocketspriteresy, pocketbanner\$, pocketbannerresx, pocketbannerresy, textbannersound, textbanner\$, textbannername\$, textbannerresx, textbannerresy, pocketselect\$, pocketselectx, pocketselecty, pocketselectresx, pocketselectresy, lookaction\$, lookx, useaction\$, giveaction\$, combineaction\$, usex, givex, combinex, textbannerfacey, textbannerfaceresx, textbannerfaceresy, choicebanner\$, choicearrowl, choicearrowr, tos\$, tdelay, stposx, stposy, tanidelay, terminalcol1, terminalcol2, terminalcol3, terminalrow1, terminalrow2, terminalfacex, terminalfacey, currencyname\$, loadicon\$, loadiconresx, loadiconresy, saveicon\$, saveiconresx, saveiconresy, downloadicon\$, downloadiconresx, downloadiconresy, torcheffectfile\$, loadbar\$, devlogo\$, devlogomode, awardbanner\$, awardbannerresx, awardbannerresy, awardbannerlocx, awardbannerlocy, awarditemresx, awarditemresy, awarditemlocx, awarditemlocy, awardtextlocx, awardtextlocy, awardgracetime, awardtitle\$, awardnotification\$, awardspeed, awardnone\$, awardarrowleft\$, awardarrowright\$, awardarrowselectleft\$, awardarrowselectright\$, versionno\$, engineversionno\$, updatelink\$, letmenuselectcolourr, letmenuselectcolourg, letmenuselectcolourb, letmenuselectcoloura, bgmenuselectcolourr, bgmenuselectcolourg, bgmenuselectcolourb, bgmenuselectcoloura, letmenudefaultcolourr, letmenudefaultcolourg, letmenudefaultcolourb, letmenudefaultcoloura, bgmenudefaultcolourr, bgmenudefaultcolourg, bgmenudefaultcolourb, bgmenudefaultcoloura, letpromptcolourr, letpromptcolourg, letpromptcolourb, letpromptcoloura, bgpromptcolourr, bgpromptcolourg, bgpromptcolourb, bgpromptcoloura, letpocketselectcolourr, letpocketselectcolourg, letpocketselectcolourb, letpocketselectcoloura, bgpocketselectcolourr, bgpocketselectcolourg, bgpocketselectcolourb, bgpocketselectcoloura, letpocketdefaultcolourr, letpocketdefaultcolourg, letpocketdefaultcolourb, letpocketdefaultcoloura, bgpocketdefaultcolourr, bgpocketdefaultcolourg, bgpocketdefaultcolourb, bgpocketdefault-

coloura, letcurrencycolourr, letcurrencycolourg, letcurrencycolourb, letcurrency-coloura, bgcurrencycolourr, bgcurrencycolourg, bgcurrencycolourb, bgcurrency-coloura, letspeechcolourr, letspeechcolourg, letspeechcolourb, letspeechcol-oura, bgspeechcolourr, bgspeechcolourg, bgspeechcolourb, bgspeechcoloura, letterminalcolourr, letterminalcolourg, letterminalcolourb, letterminalcoloura, bgterminalcolourr, bgterminalcolourg, bgterminalcolourb, bgterminalcoloura, letselectbannercolourr, letselectbannercolourg, letselectbannercolourb, letse-lectbannercoloura, bgselectbannercolourr, bgselectbannercolourg, bgselect-bannercolourb, bgselectbannercoloura, spoofoptiontitle\$, spoofoption1\$, spoof-option2\$, spoofoption1result\$, spoofoption2result\$, moddingname\$

**Filename** - options.ddf

**Description** - Options data.

**Location** - sloc\$

**Mode** - Input and Output

**Values** - screenmode, soundmode, musicvol, sfxvol

**Filename** - defaultoptions.ddf

**Description** - Default options data.

**Location** - sloc\$

**Mode** - None (file replaces options.ddf when no option config is detected)

**Values** - (see options.ddf)

**Filename** - savedata.ddf

**Description** - Player save data.

**Location** - sloc\$

**Mode** - Input and Output

**Values** - mapno, currency, posx, posy, direction, igametime, pocketcarry, pock-etitem(totalpockets), (**repeat pocketitem until totalpockets is reached**), pocketvisible(totalpockets), (**repeat pocketvisible until totalpockets is reached**), checkpoint(totalcheckpoints), (**repeat checkpoint until total-checkpoints is reached**), mplayermodel\$

**Filename** - defaultsave.ddf

**Description** - Default player save data.

**Location** - sloc\$

**Mode** - None (file replaces savedata.ddf when player erases save)

**Values** - (see savedata.ddf)

**Filename** - fileloc.ddf

**Description** - Directory locations in Windows and then Linux format.

**Location** - dloc\$

**Mode** - Input Only

**Values** - dloc\$, mloc\$, ploc\$, floc\$, sloc\$, oloc\$, scriptloc\$, museloc\$, sfxloc\$, pocketloc\$, uiloc\$, tloc\$, aloc\$, menuloc\$, dloc\$, mloc\$, ploc\$, floc\$, sloc\$, oloc\$, scriptloc\$, museloc\$, sfxloc\$, pocketloc\$, uiloc\$, tloc\$, aloc\$, menuloc\$

**Filename** - map(mapno).ddf

**Description** - Map data.

**Location** - mloc\$/mapdir\$

**Mode** - Input Only

**Values** - mapname\$, playmusic\$, mapeffect, parallaxmode, mapx, mapy, mapobjectno, mapplayerno, maptriggerno, objectname\$(totalobjects), objectx(totalobjects), objecty(totalobjects), **(repeat objectname\$ to objecty until totalframes is reached)**, playername\$(totalplayers), playerx(totalplayers), playery(totalplayers), mplayerx(totalplayers), mplayery(totalplayers), playergrace(totalplayers), playerdefault(totalplayers), **(repeat playername\$ to playerdefault until totalplayers is reached)**, triggername\$(totaltriggers), triggerx1(totaltriggers), triggerx1(totaltriggers), triggerx1(totaltriggers), triggerx1(totaltriggers), triggerx1(totaltriggers), **(repeat triggername\$ to triggerx1 until totalframes is reached)**

**Notes** - objectname(no)\$ to equal "[COLLISIONONLY]" if you'd like an object to act as collision only. First X and Y co-ordinates are stored in objectx(no). Second X and Y co-ordinates are stored in objecty(no). X and Y values are separated by a decimal point, ie. '42.36'. All numbers used are to be two digits or above, one digit numbers use a '0' before the number ie '01.02'. The values used are 'objectx(no).objecty(no)' and then 'objectresx(no).objectresy(no)'. Uses 'collision-step' as the object step.

**Filename** - (playername).ddf

**Description** - Player data for main player and NPCs.

**Location** - ploc\$/(playername)/

**Mode** - Input Only

**Values** - playername(totalplayers)\$, playerresx(totalplayers), playerresy(totalplayers), players(totalplayers), playernote1(totalplayers), playernote2(totalplayers), playerlayer2(totalplayers), playerspeed(totalplayers), playercollision(totalplayers)

OR

temp\$, mpx, mpy, mps, mpnote1, mpnote2, temp

**Filename** - consolelog.txt

**Description** - Console log for development purposes.

**Location** - dloc\$

**Mode** - Output Only

**Values** - date\$, time\$, err, errdescription\$

OR

date\$, time\$, eventtitle\$, eventdata\$, eventnumber

**Filename** - (objectname).ddf

**Description** - Map object data.

**Location** - oloc\$/(objectname)/

**Mode** - Input Only

**Values** - objectname\$(totalobjects), objectresx, objectresy, objects(totalobjects), objectlayer(totalobjects), objectspeed(totalobjects), objectcollision(totalobjects)

**Filename** - musicfiles.ddf

**Description** - List of music files to be loaded into memory.

**Location** - museloc\$

**Mode** - Input Only

**Values** - (list of music files in quotes, no extension).

**Filename** - sfxfiles.ddf

**Description** - List of sound effect files to be loaded into memory.

**Location** - sfxloc\$

**Mode** - Input Only

**Values** - (list of sound effect files in quotes, no extension).

**Filename** - pocketfiles.ddf

**Description** - List of pocket files to be loaded into memory.

**Location** - pocketloc\$

**Mode** - Input Only

**Values** - (list of pocket files in quotes, no extension).

**Filename** - (itemname).ddf

**Description** - Pocket item metadata.

**Location** - pocketloc\$/(itemname)/

**Mode** - Input Only

**Values** - pocketname\$(totalpockets)\$, pocketdescription\$(totalpockets)

**Notes** - pocketdescription\$(totalpockets) to equal "[RUNSCRIPT] {script name}" if you'd like a pocket item to run a script when looked at.

**Filename** - (scriptname).vsf

**Description** - Script file.

**Location** - scriptloc\$

**Mode** - Input Only

**Values** - (list of script commands in quotes).

**Filename** - (menuname).ddf

**Description** - Menu metadata

**Location** - uiloc\$

**Mode** - Input Only

**Values** - menuchoice1\$, menuchoice2\$, menuchoice3\$, menuchoice4\$, menuchoice5\$, menuchoice6\$, menucommand1\$, menucommand2\$, menucommand3\$, menucommand4\$, menucommand5\$, menucommand6\$, mcy1, mcy2, mcy3, mcy4, mcy5, mcx6, menunox, menuposx, menubackdrop\$, menu-music\$

**Filename** - (terminal data name).ddf

**Description** - Terminal Metadata

**Location** - tloc\$/(terminal data name)/

**Mode** - Input Only

**Values** - ct1, cn1\$, ct2, cn2\$, ct3, cn3\$, ct4, cn4\$, ct5, cn5\$, ct6, cn6\$, par-entdir\$

**Filename** - (animation data name).ddf

**Description** - Animation Metadata.

**Location** - aloc\$/(animation data name)/

**Mode** - Input Only

**Values** - aniframes, frame(totalframes), (**repeat frame until totalframes is reached**)

**Filename** - checkupdate.ddf

**Description** - Latest update information. Will only exist whilst updates are being checked or installed.

**Location** - (executable folder)

**Mode** - Input Only

**Values** - newversionno\$, updaterlinklnx\$, updaterlinkwin\$, downloadlink\$, un-  
ziplink\$, datafolder\$, winexe\$, lnexe\$, readme\$, changelog\$, manual\$,  
source\$, updatesource\$, updatezip\$, savestatus, minsavever\$

## Subs

| Sub Name            | Sub Function  |
|---------------------|---|
| animation:          | Loads and Draws animation                                     |
| awarddraw:          | Draws granted awards onto the screen.                         |
| awardload:          | Loads award assets and data                                   |
| awardmenu:          | Loads award menu  |
| awardunload:        | Unloads awards from memory                                    |
| carryplayervalues:  | Copies player values to memory to be used on another map.     |
| centretext:         | Centralises text for display use                              |
| choicebannercalc:   | Puts choice text on player choice menu.                       |
| choicebannerdraw:   | Draws banner for player choice menu.                          |
| collision:          | Manages collision subsSubs                                    |
| collisionconverter: | Converts map co-ordinate values into collision data.          |
| collisionprinter:   | Sets values for collision conflict event for console printer. |
| consoleboot:        | Dumps boot info to errorlog.txt                               |
| consoleprinter:     | Dumps any additional info to errorlog.txt                     |
| controlgenerator:   | Generates control dialogue text                               |
| deleteupdaters:     | Deletes any remaining updater files that aren't needed.       |
| devlogo:            | Displays developer logo.                                      |
| dimmer:             | Assigns array values.   |
| displayconsole:     | Displays developer console                                    |
| effectdraw:         | Draws special map effects.                                    |
| endgame:            | Saves any additional savedata then quits game                 |
| endgamemenu:        | Asks for user confirmation of quit.                           |
| endscriptcmd:       | Ends the script command and moves onto the next.              |
| erasesave:          | Erases save file and replaces it with default                 |
| errorduringerror:   | Failsafe if error handler encounters an unexpected error.     |
| errorhandler:       | Calls redirects or rescue code upon engine errors             |
| errorprinter:       | Dumps error information to errorlog.txt                       |
| fadein:             | Fade in utility   |
| fadeout:            | Fade out utility  |
| findcontrol:        | Matches unicode with letter keys for control instructions.    |
| fontload:           | Loads engine font into memory                                 |
| fontunload:         | Unloads fonts   |



|                   |   |
|-------------------|---|
| footchanger:      | Calculates when player foot should change when walking                  |
| game:             | Main engine loop, calls upon subs required                              |
| gameloop:         | Quick version of game loop for script purposes.                         |
| gamereboots:      | Checks to see if the game requires a reboot.                            |
| generateoffsets:  | Generates random map animation offsets for objects and players.         |
| givecurrency:     | Gives currency to player  |
| hideitem:         | Changes a pocket item visibility to hide.                               |
| ifaward:          | Checks status of award  |
| ifcurrency:       | Checks if player has enough currency                                    |
| ifholding:        | Checks if player is holding an item in hand                             |
| ifmapno:          | Checks to see if player is on requested map.                            |
| ifmodel:          | Checks to see if player character is using a requested character model. |
| ifpocket:         | Checks pocket for item.   |
| inputload:        | Informs engine and console of enabled game controls.                    |
| inputter:         | Input sub. handles all player input from the keyboard                   |
| layercalc:        | Calculates which layer objects and players are drawn on                 |
| loadbar:          | Displays load bar.  |
| loadbarsetup:     | Counts amount of assets to be loaded for the loading bar                |
| loadgame:         | Arranges loading sequence for loading previous saves during gameplay.   |
| mainplayerdraw:   | Draws mainplayer to screen  |
| mainplayerload:   | Loads data and sprites on player character                              |
| mainplayerunload: | Unloads main player sprites   |
| mapcollision:     | Calculates and implements map sprite boundaries                         |
| mapload:          | Loads map data and sprites  |
| mapmusicsetter:   | Sets current music the music attached to current map                    |
| mapunload:        | Unloads map sprites   |
| markgone:         | Marks inventory items as 'gone forever'                                 |
| menugenerator:    | Menu generator sub intended for use before and during gameplay          |
| missingasset:     | Replaces art assets with error texture                                  |
| modload:          | Checks launch parameters for any mods to load.                          |
| modmenu:          | Menu for launching mods.  |
| musicfadein:      | Fades in music  |

|                         |   |
|-------------------------|---|
| musicfadeout:           | Fades out music   |
| musicfadeoutstart:      | Begins process of fading out music  |
| musicfadestopper:       | Double checks that no unwanted music is being played whilst the fade is on. |
| musicload:              | Loads music files into memory   |
| musicpause:             | Pauses music  |
| musicplay:              | Plays requested music file  |
| musicstop:              | Stops playing requested music   |
| musictoggle:            | Toggles music modes   |
| musicunload:            | Unloads music files from memory   |
| musicvol:               | Sets music volume.  |
| objectcollision:        | Handles object collision  |
| objectcollisionchecker: | Checks to see if mainplayer is colliding with any objects.                  |
| objectdraw:             | Draws objects to screen   |
| objectload:             | Loads object metadata and sprites   |
| objectunload:           | Unloads loaded object data  |
| optionload:             | Loads option values into memory   |
| parallaxdraw:           | Draws parallax layers to screen.  |
| playdemomenu:           | Menu for starting playback of demo or speedrun files                        |
| playercollision:        | Handles NPC collision   |
| playercollisionchanger: | Handles NPC collision   |
| playerdraw:             | Determines order of NPCs to be drawn  |
| playerload:             | Loads NPC's into memory   |
| playermove:             | Calculates NPC movement   |
| playerunload:           | Unloads NPC's from memory   |
| pocketarrowleft:        | Flashes left pocket arrow   |
| pocketarrowright:       | Flashes right pocket arrow  |
| pocketcalcdown:         | Works out what pocket item to display (going down)                          |
| pocketcalcup:           | Works out what pocket item to display (going up)                            |
| pocketcombine:          | Combines pocket items   |
| pocketdraw:             | Draws pockets to screen and displays in-game pocket controls                |
| pockettext:             | Draws pocket function controls.   |
| pocketload:             | Loads pocket items into memory  |
| pocketunload:           | Unloads pocket items from memory  |
| prompt:                 | Engine command prompt for development purposes                              |
| promptquit:             | Dumps quit info to errorlog.txt   |
| readtxt:                | Terminal file opener  |

|                        |  |
|------------------------|--|
| savedefault:           | Overwrites default save with current settings and game progress.             |
| saveload:              | Loads previous save  |
| saverestore:           | Menu for restoring old saves.  |
| savesave:              | Saves game   |
| savetime:              | Saves only the gametime and award values to save                             |
| savevalue:             | Saves a specific value to save file.   |
| screendraw:            | Draws game on screen when required   |
| screenload:            | Sets screen mode   |
| screentoggle:          | Toggles fullscreen and windowed modes  |
| scriptanimatecmd:      | Processes script 'animate' command.  |
| scriptcarryvaluescmd:  | Processes script 'carryvalues' command.                                      |
| scriptcheckpointcmd:   | Processes script 'checkpoint' command.                                       |
| scriptchoicecmd:       | Processes script 'choice' command.   |
| scriptcontrolcmd:      | Processes script 'control' command.  |
| scriptdirectioncmd:    | Processes script 'direction' command.  |
| scriptgiveawardcmd:    | Processes script 'giveaward' command.  |
| scriptgivecmd:         | Processes script 'give' and 'silentgive' command.                            |
| scriptgivecurrencycmd: | Processes script 'givecurrency' and 'silent-givecurrency' command.           |
| scripthaltcmd:         | Processes script 'halt' command.   |
| scripthalttimedcmd:    | Processes script 'halttimed' command.  |
| scripthidecmd:         | Processes script 'hide' command.   |
| scriptifawardcmd:      | Processes script 'ifaward' command.  |
| scriptifcheckpointcmd: | Processes script 'ifcheckpoint' command.                                     |
| scriptifcurrencycmd:   | Processes script 'ifcurrency' command.                                       |
| scriptifdirectioncmd:  | Processes script 'ifdirection' command.                                      |
| scriptifgonecmd:       | Processes script 'ifgone' command.   |
| scriptifholdingcmd:    | Processes script 'ifholding' command.  |
| scriptifmapnocmd:      | Processes script 'ifmapno' command.  |
| scriptifmodelcmd:      | Processes script 'ifmodel' command.  |
| scriptifpocketcmd:     | Processes script 'ifpocket' command.   |
| scriptifrandomcmd:     | Processes script 'ifrandom' command.   |
| scriptiftimedcmd:      | Processes script 'iftimed' command.  |
| scriptifvaluecmd:      | Processes script 'ifvalue' command.  |
| scriptmakevaluecmd:    | Processes script 'makevalue' command.  |
| scriptmapcmd:          | Processes script 'map' command.  |
| scriptmapeffectcmd:    | Processes script 'mapeffect' command.  |
| scriptmovecmd:         | Processes script 'move' and 'sprint' command.                                |
| scriptmusiccmd:        | Processes script 'music' command.  |
| scriptnametrim:        | Trims the script name according to what is required by script 'if' commands. |
| scriptpilotcmd:        | Processes script 'pilot' command.  |

|                        |  |
|------------------------|--|
| scriptruncmd:          | Processes script 'run' command.                                      |
| scriptsaycmd:          | Processes script 'say' command.                                      |
| scriptscriptcmd:       | Processes script 'script' command.                                   |
| scriptshellnxcmd:      | Processes script 'shellnx' command.                                  |
| scriptshellwincmd:     | Processes script 'shellwin' command.                                 |
| scriptshowcmd:         | Processes script 'show' command.                                     |
| scriptspeakercmd:      | Processes script 'speaker' command.                                  |
| scripttakecmd:         | Processes script 'take' command.                                     |
| scripttakecurrencycmd: | Processes script 'takecurrency' and 'silent-takecurrency' command.   |
| scriptterminaltextcmd: | Processes script 'terminaltext' command.                             |
| scripttimedscriptcmd:  | Processes script 'timedscript' command.                              |
| scripttriggercmd:      | Processes script 'trigger' command.                                  |
| scriptvaluefetcher:    | Fetches script value   |
| scriptwaitcmd:         | Processes script 'wait' command.                                     |
| scriptwarpcmd:         | Processes script 'warp' command.                                     |
| selectobjectbanner:    | Draws selected object name onto screen                               |
| seperatecommands:      | Unified way of separating a script command into separate values.     |
| setup:                 | Game launch setup, assigns engine values from metadata               |
| sfxload:               | Loads sound effect files into memory                                 |
| sfxplay:               | Plays requested sound effect   |
| sfxstop:               | Stops all sound effects that are playing                             |
| sfxunload:             | Unloads sound effect files from memory                               |
| sfxvol:                | Sets sfx volume.   |
| showimage:             | Displays an image on screen.   |
| showitem:              | Changes a pocket item visibility to show.                            |
| slightfadein:          | Undims screen  |
| slightfadeout:         | Dims screen  |
| takecurrency:          | Takes currency from player   |
| talksfx:               | Generates player talking sound effects.                              |
| terminaldraw:          | Draws terminal screens.  |
| terminalload:          | Loads terminal data  |
| terminalunload:        | Unloads terminal data  |
| textbannercalc:        | Calculates text length from string in text banner                    |
| textbannerdraw:        | Draws text banner  |
| timeframecounter:      | Counts timer and frame stats   |
| triggercollision:      | Checks to see if mainplayer is colliding with any invisible triggers |
| uiloader:              | Loads UI items (user interface stuff)                                |
| uiunload:              | Unloads UI items (user interface stuff)                              |
| updatechecker:         | Checks internet for available updates.                               |
| useobject:             | Interacts real world object or player                                |

|                        |   |
|------------------------|---|
| usepocket:             | Uses pocket item on map object.                         |
| usepocketpocket:       | Combines pocket items                                   |
| variablevalueinjector: | Injects the value of engine variables into text banners |
| whitefadein:           | Fades in (using white)                                  |
| whitefadeout:          | Fades out (using white)                                 |
| worlddraw:             | Draws map to screen                                     |

---

## Menu Commands

The following is a list of available commands on main menu.

| Menu Command    | Command Description                 |
|-----------------|-------------------------------------|
| displayawards   | Shows award menu                    |
| displayconsole  | Displays developer console          |
| displaycontrols | Displays controls                   |
| endgame         | Ends Game                           |
| erasesave       | Erases Save                         |
| loadmod         | Opens mod loading menu              |
| menu (menuname) | Opens different menu                |
| musictoggle     | Toggles Music                       |
| musicvol        | Sets music volume                   |
| playgame        | Plays Game                          |
| reloadsave      | Reloads the current save.           |
| restoresave     | Opens save restore menu             |
| screentoggle    | Toggles fullscreen                  |
| sfxtoggle       | Toggles Sfx                         |
| sfxvol          | Sets sfx volume                     |
| spooftoggle     | A fake toggle option                |
| updategame      | Checks for update then runs updater |

## Prompt Commands

**The VaME command prompt is designed to be a flexible command based tool that allows access to engine values and can manipulate them upon request. Similar to most command terminals, the VaME console works on a "> COMMAND ARGUMENT" basis in that it uses a series of basic commands followed by an instruction of which the command works from. For instance, "> whatis direction" will display the value of the direction value.**

### **command name**

Description of command and what its functions are. (usage example)

*Possible arguments - all possible command arguments are listed here.*

**All Command Prompt commands are listed below.**

### **whatis**

Displays the value of a requested variable or string value. (whatis mapno)

*Possible arguments - noclip, resx, resy, mapno, frames, ros, mplayermodel, pace, ctime, itime, location, oldlocation, direction, posx, posy, oposx, oposy, mapname, mpx, mpy, mpwalking, mpfoot, mpfootloop, footpace, hud, clearscreen, errdescription, err, errorline, date, time, fps, oldmapno, oldmapname, oldmplayermodel, fontname, fontsize, fontstyle, devmode, mapobjectno, maptriggerno, mapplayerno, playmusic, currentmusic, oldmusic, soundmode, mpidle, playeridle, triggername, playername, objectname, pocketitem1, pocketitem2, pocketitem3, pocketitem4, pocketitem5, pocketitem6, disablefade, currency, mapeffect, versionno, timer, scriptline, musicvol, sfxvol, engineversionno, exitsave, checkpoint, collisionstep, checkpoint, mainmenu, fade status, random*

### **display**

Displays certain values on a developer HUD within the game. (display location)

*Possible arguments -*

*off (removes any developer hud currently displayed)*

*location (location co-ordinates of top left corner of map)*

*olocation (previous co-ordinates of top left corner of map)*

*time (local time as well as engine time (ctime\$))*

*frames (FPS and number of frames processed)*

*maplocation (location co-ordinates of top left corner and bottom right corner of map)*

*fadein (fade in effect)*

*fadeout (fade out effect)*

*techspecs (tonne of great info for the tech nerds x)*

*layers (draw layer data)*

*mplayerlocation (location of player sprite on map)*

*selectobject (name of selected map object)*

*console (opens console - cannot be removed with "display off" command)*

*playerlocation (location of npc sprites on map)*

*pocketnos (number of inventory items loaded into game)*

*pocketcarry (number of inventory items carried by player)*

*miniconsole (last console line)*

*checkpoint (active checkpoints)*

*gametime (amount of time save file has been used)*

### **system**

Quits or resets game or game function. (system quit)

*Possible arguments -*

*now (closes engine immediately, no endgame procedure)*

*hang (closes engine immediately, no endgame procedure, window remains open until closed)*

*map (reloads all map data)*

*mainplayer (reloads all mainplayer data)*

*screen (reloads all screen setting data)*

*font (reloads all font data)*

*quit (triggers endgame procedure)*

*music (toggles music)*

*ui (reloads all gui data)*

*pockets (reloads all pocket data)*

*sfx (reloads all sound effect data)*

*terminal (reloads all terminal data)*

*update (checks for update online)*

*fix (attempts to repair game)*

*consolelog (erases consolelog.txt)*

*colour (reloads text colour data)*

*speedrun (begins speedrun mode. cannot be disabled until game quit.)*

*restart (restarts engine)*

### **script**

Runs a requested script or ends the script (script demo)

*Possible arguments - any script file in script (scriptloc\$) folder (no ext).*

### **error**

Sends a requested error signal to the engine. (error 420)

*Possible arguments - any error code.*

### **sfx**

Plays a requested sound effect. (sfx pickup)

*Possible arguments - any sound effect file in sfx (sloc\$) folder.*

### **shell**

Runs a requested command to host OS. (shell ls)

*Possible arguments - any command that can be run in host OS terminal/ prompt.*

### **change**

Changes value of requested variable or string value. (change mapno)

*Possible arguments - pace, mapno, direction, mainplayer, posx, posy, resx, resy, fontsize, fontname, fontstyle, noclip, devmode, playmusic, soundmode, ros, disablefade, currency, mapeffect, versionno, timer, musicvol, sfxvol, consolelogging, displayconsole, checkpoint, trigger*

### **giveitem**

Gives player an item. (give floppy)

*Possible arguments- any pocket item loaded into memory.*



**takeitem**

Takes an item off player. (take floppy)

*Possible arguments - any pocket item loaded into memory.*

**hideitem**

Hides an item. (hide floppy)

*Possible arguments - any pocket item loaded into memory.*

**showitem**

Shows an item. (show floppy)

*Possible arguments - any pocket item loaded into memory.*

**say**

Displays text of choice in text banner. (say hello world!)

*Possible arguments - any text.*

**givecurrency**

Gives currency to player. (givecurrency 10)

*Possible arguments - any positive integer.*

**takecurrency**

Takes currency from player. (takecurrency 10)

*Possible arguments - any positive integer.*

**markgone**

Takes pocket item off player forever. (markgone floppy)

*Possible arguments - any pocket item loaded into memory except currency.*

**markback**

Resets pocket item after a fullytake command. (fullygive floppy)

*Possible arguments - any pocket item loaded into memory except currency.*

**save**

Executes save functions. (save game)

*Possible arguments -*

*game (saves game)*

*erase (erases savegame - will reset engine)*

*default (rewrites default save)*

*load (loads last save)*

## Stage Director Commands

| Command Name (possible arguments)   | Example  | Command Description  |
|---|--|--|
| allowcontrol  | "allowcontrol"   | Allows player to control the game whilst the script runs (intended for short scripts that change statues of checkpoints or triggers).                        |
| allowskip   | "allowskip"  | Player can now skip the script if they hold the back/pause key after this command is run.  |
| animate (object/player then objectname/playername then animation file name) | "animate mainplayer anitest"<br>"animate player ivan anitest"<br>"animate object desk anitest"         | Executes a custom animation on player or object.   |
| backchoice (choice 1, choice 2, choice 3)                                   | "backchoice choice1, choice2, choice3"   | Displays a choice menu for the player to pick. Once picked the script diverts to previous script in the script chain (scriptname\$ + "-ifchoice(choiceno)"). |
| backmove (playername then X or Y then co-ordinate value or userandom)       | "backmove mainplayer X 100"<br>"backmove player ivan X 100"<br>"backmove mainplayer X userandom"       | Moves player/NPC X or Y values (walking backwards)   |
| backsprint (playername then X or Y then co-ordinate value or userandom)     | "backsprint mainplayer X 100"<br>"backsprint player ivan X 100"<br>"backsprint mainplayer X userandom" | Moves player/NPC X or Y values (sprinting backwards)   |
| carryvalues   | "carryvalues"  | Copies NPC location values into memory to be copied into the next map. This maintains location of NPCs in duplicate maps.                                    |
| checkpoint (number or all or userandom then on or off)                      | "checkpoint 1 on"<br>"checkpoint 1 off"<br>"checkpoint all off"<br>"checkpoint userandom off"          | Sets checkpoints to be either on or off.   |
| choice (choice 1, choice 2, choice 3)                                       | "choice choice1, choice2, choice3"   | Displays a choice menu for the player to pick. Once picked the script diverts to alternate script (scriptname\$ + "-ifchoice(choiceno)").                    |

|  |   |  |
|--|---|--|
| clear  | "clear"   | Tells screen drawing function to clear the screen.   |
| collision (on or off)  | "collision off"   | Turns collision on or off.   |
| control (up, down, left, right, select, pockets or back then on or off)                  | "control pockets off"<br>"control back on"  | Enables or disables player inputs.   |
| dim (on or off)  | "dim on"  | Controls dim   |
| direction (playername then direction number or userandom (or "faceplayer" if using NPC.) | "direction mainplayer 2"<br>"direction player ivan 2"<br>"direction player ivan faceplayer"<br>"direction mainplayer userandom" | Changes direction of player or NPC   |
| effects (on or off)  | "effects on"  | Enables or disables effects such as fading and dimming.  |
| fade (in or out)   | "fade in"   | Controls fade  |
| give (item name)   | "give floppy"   | Gives player character an item. Does not work with currency.                                       |
| giveaward (award number)   | "giveaward 1"   | Gives award to player and displays a notification about it if the play hasnt had the award before. |
| givecurrency (positive integer or userandom)   | "givecurrency 10"<br>"givecurrency userandom"   | Gives set amount of currency to player.  |
| halt (number of seconds or userandom)  | "halt 10"<br>"halt userandom"   | Halts engine for a number of resconds.   |

|   |  |   |
|---|--|---|
| halttimed                                     | "halttimed"                                  | Cancels any timed scripts that are active.  |
| hide (item name or all)                       | "hide floppy"<br>"hide all"                  | Changes visibility of pocket item to hidden.  |
| ifaward (award number)                        | "ifaward 1"<br>"ifaward userandom"           | Checks to see if award has been granted to the player. If so script diverts to alternate script (scriptname\$ + "-ifaward"). If not then script continues.                          |
| ifcheckpoint (checkpoint number or userandom) | "ifcheckpoint 1"<br>"ifcheckpoint userandom" | Checks to see if checkpoint is active. If so then the script diverts to alternate script (scriptname\$ + "-ifcheckpoint"). If not then the script continues.                        |
| ifcurrency (currency amount or userandom)     | "ifcurrency 10"<br>"ifcurrency userandom"    | Checks to see if mainplayer has a requested amount of currency. If so script diverts to alternate script (scriptname\$ + "-ifcurrency"). If not then script continues.              |
| ifdirection (direction number or userandom)   | "ifdirection 1"<br>"ifdirection userandom"   | Checks to see if direction of mainplayer is facing a certain direction. If so script diverts to alternate script (scriptname\$ + "-ifdirection"). If not then script continues.     |
| ifgone (pocket item)                          | "ifgone floppy"                              | Checks to see if pocket item has been marked "gone forever". If so script diverts to alternate script (scriptname\$ + "-ifgone"). If not then script continues.                     |
| ifholding (pocket item)                       | "ifholding floppy"                           | Checks to see if mainplayer is holding a requested item in hand. If so then the script diverts to alternate script (scriptname\$ + "-ifholding"). If not then the script continues. |
| ifmapno (mapno or userandom)                  | "ifmapno 1"<br>"ifmapno userandom"           | Checks to see player is on a certain map. If so script diverts to alternate script  |

|  |   |  |
|--|---|--|
|  |   | (scriptname\$ + "-ifmapno"). If not then script continues.   |
| ifmodel  | "ifmodel ivan"  | Checks to see if player character is using a certain sprite model. If so the script diverts to alternate script (scriptname\$ + "-ifmodel"). If not then the script continues. |
| ifpocket (pocket item)   | "ifpocket floppy"   | Checks to see if mainplayer has requested pocket item. If so then the script diverts to alternate script (scriptname\$ + "-ifpocket"). If not then the script continues.       |
| ifrandom (above, below or equal then number)                   | "ifrandom above 5"<br>"ifrandom below 5"<br>"ifrandom equal 5"    | Checks to see if last generated random number matches conditions. If so script diverts to alternate script (scriptname\$ + "-ifrandom"). If not then script continues.         |
| iftimed  | "iftimed"   | Checks to see if a timed script is active. If so then the script diverts to alternate script (scriptname\$ + "-iftimed"). If not then the script continues.                    |
| ifvalue (handle number then above, below or equal then number) | "ifvalue 1 above 5"<br>"ifvalue 1 below 5"<br>"ifvalue 1 equal 5" | Checks to see if last generated random number matches conditions. If so script diverts to alternate script (scriptname\$ + "-ifrandom"). If not then script continues.         |
| loadgame   | "loadgame"  | Loads previous save  |
| loading  | "loading"   | Displays load icon. Will only work when script is not drawing anything. Icon automatically removed if any screen draw or clear occurs.   |
| makerandom (range starting small to large)                     | "makerandom 1 50"   | Generates a random number for the script to use between the defined range.   |
| makevalue (handle number then starting value or userandom)     | "makevalue 1 100"<br>"makevalue 1 userandom"                      | Makes a value to be stored in memory and saved.  |
| map (mapno or userandom)                                       | "map 1"<br>"map userandom"  | Changes to map number placed in argument or random map defined in argument.  |

|   |  |   |
|---|--|---|
| mapeffect (dark / rain / storm / torch / sunsetleft / sunsetright / sunsetdown / sunsetup / pitchblack / off) | "mapeffect dark"   | Displays an effect over the map.  |
| markgone (pocket item)  | "markgone floppy"  | Sets pocket item as "gone forever" and unable to be got by the mainplayer again. Does not work with currency.                                   |
| model (playername then player model name)   | "model mainplayer ivan"  | Changes mainplayer model  |
| modvalue (handle number then add, minus, divide, times then value or userandom)                               | "modvalue 1 add 100"<br>"modvalue 1 minus userandom"   | Manipulates the value of a value.   |
| move (playername or object-name then X or Y then co-ordinate value or userandom)                              | "move mainplayer X 100"<br>"move player ivan X 100"<br>"move object desk X 100"<br>"move mainplayer X userandom" | Moves object/NPC X or Y values (walking)  |
| music control (play or stop / fade or cut then on or off)   | "music control play"<br>"music control fade on"<br>"music control cut on"  | Either plays or stops current music.  |
| music file (music file name)  | "music file menu"  | Plays requested music file (and stops previous music if playing)  |
| pilot (playername then on or off)   | "pilot ivan on"  | Sets NPC's to halt their idle walkabouts and assume script control.   |
| remark (remark)   | "remark test script"   | Does nothing. Used for commenting in script files.  |
| run (application)   | "run dummyapp"   | Runs an external application from utility folder and pauses VaME. Windows executables end with "_win.exe". Linux executables end with "_linux". |
| savegame  | "savegame"   | Saves the game.   |
| savevalue (handle number)   | "savevalue 1"  | Saves a specific value to the save file.  |
| saving  | "saving"   | Displays save icon. Will only work when script is not drawing anything. Icon automatically removed if any screen draw or clear occurs.          |

|                                  |   |  |
|----------------------------------|---|--|
| say (words)                      | "say hello world!"  | Displays words in dialogue box.                            |
| script (script name)             | "script testscript01"                                     | Runs a script (ends current script)                        |
| sfx (sound effect file name)     | "sfx pickup"  | Plays requested sound file                                 |
| Sfxloop (sound effect file name) | "sfxloop pickup"  | Loops a sound effect until sfxstop is run.                 |
| sfxstop                          | "sfxstop"   | Halts all sfx including looping ones.                      |
| shelllnx (shell command)         | "shelllnx rm test.txt"                                    | Runs a shell command (linux only).                         |
| shellwin (shell command)         | "shellwin del text.txt"                                   | Runs a shell command (windows only, cmd not powershell)    |
| show (item name or all)          | "show floppy"<br>"show all"                               | Changes visibility of pocket item to visible.              |
| showimage (image name)           | "showimage map"   | Displays an image on screen.                               |
| silentgive (pocket item)         | "silentgive floppy"                                       | Silently gives a pocket item without displaying animation. |
| silentgivecurrency               | "silentgivecurrency 10"<br>"silentgivecurrency userandom" | Silently gives currency without displaying animation.      |
| silenttake (pocket item)         | "silenttake floppy"                                       | Silently takes a pocket item without displaying animation. |

|   |  |  |
|---|--|--|
| silenttakecurrency  | "silenttakecurrency 10"<br>"silenttakecurrency useran-<br>dom"   | Silently takes currency<br>without displaying animation.   |
| slowfade (in or out)  | "slowfade in"  | Slowly fades in or out.  |
| speaker (playername)  | "speaker ivan"<br>"speaker mainplayer"<br>"speaker nobody"   | Sets speaker for text banner<br>words. Set to equal "main-<br>player" for the main player.<br>Set to equal "nobody" for no<br>speaker.   |
| sprint (playername or object-<br>name then X or Y then co-or-<br>dinate value or userandom) | "sprint mainplayer X 100"<br>"sprint player ivan X 100"<br>"sprint object desk X 100"<br>"sprint object desk X useran-<br>dom" | Sprints object or NPC X or Y<br>values (sprinting)   |
| take (item name)  | "take floppy"  | Takes an item off player<br>character. Does not work with<br>currency.   |
| takecurrency (positive integer<br>or userandom)   | "takecurrency 10"<br>"takecurrency userandom"  | Takes set amount of currency<br>off player.  |
| terminal (filename)   | "terminal test"  | Launches specified terminal<br>file. Will end the script.  |
| terminalnoboot (filename)   | "terminalnoboot test"  | Launches a specified ter-<br>minal file without the boot an-<br>imation. Will end the script.  |
| terminaltext (line 1, line 2,<br>line 3, line 4, line 5, sysstat)                           | "terminaltext this is line 1,<br>this is line 2, this is line 3,<br>this is line 4, this is line 5, 1"                         | Displays a terminal text file.   |
| timedscript (scriptname then<br>time in seconds or useran-<br>dom)                          | "timedscript testscript 60"<br>"timedscript testscript<br>userandom"   | Launches a specified script<br>after a number of seconds<br>have passed provided no<br>other scripts are running. If<br>another script is running at<br>the time of launch the launch<br>will occur after the current<br>script has ended. |



|   |  |   |
|---|--|---|
| trigger (triggerno or userandom then on or off)   | "trigger 1 on"<br>"trigger 1 off"<br>"trigger userandom on"  | Changes if trigger is active or not.                    |
| wait (number of seconds or userandom)   | "wait 5"<br>"wait userandom"   | Waits number of seconds placed in argument              |
| warp (player or object then playername or objectname then X or Y then co-ordinate value or userandom) | "warp mainplayer X 100"<br>"warp player ivan X 100"<br>"warp object desk X 100"<br>"warp mainplayer X userandom" | Changes X or Y value of object, player or NPC instantly |
| white (in or out)   | "white in"<br>"white out"  | Fades but uses white instead of black.                  |
| resetsavetime   | "resetsavetime"  | Resets save timer back to 0.                            |

---

## Stage Director Script Say Variables

| Name           | Description  |
|----------------|--|
| [selectobject] | Current world object                               |
| [helditem1]    | Current held pocket item                           |
| [helditem2]    | Second held pocket item (for pocket mergers)       |
| [ros]          | Client operating system                            |
| [tos]          | In game terminal OS name                           |
| [gametitle]    | Title of the game                                  |
| [musicvol]     | Volume the music is set at                         |
| [sfxvol]       | Volume the sfx are set at                          |
| [proposedmod]  | Proposed mod to load                               |
| [moddingname]  | What mods are referred to in game                  |
| [gametime]     | How long the save file has been played in HH:MM:SS |
| [value]-(no)   | A script value                                     |

## Error Codes

| Error Code | Description (and Potential Fix)   | Result                      |
|------------|---|-----------------------------|
| 5          | ILLEGAL FUNCTION CALL   | No Crash                    |
| 6          | OVERFLOW  | No Crash                    |
| 7          | OUT OF MEMORY (check to see if host machine running VaME has enough memory.)  | Guru Meditation             |
| 9          | Subscript out of range. (An array's upper or lower dimensioned boundary has been exceeded.)                           | No Crash                    |
| 11         | DIVISION BY ZERO (check any metadata values that are processed by any engine algorithms)                              | Crash (not Guru Meditation) |
| 13         | TYPE MISMATCH   | No Crash                    |
| 19         | NO RESUME - RESUME REQUIRED   | No Crash                    |
| 20         | RESUME WITHOUT ERROR  | No Crash                    |
| 26         | FOR WITHOUT NEXT  | No Crash                    |
| 51         | INTERNAL MALFUNCTION  | No Crash                    |
| 52         | METADATA FILE DATA NOT FOUND (check if metadata file engine is reading has the expected values within)                | No Crash                    |
| 53         | METADATA FILE NOT FOUND (check if metadata file engine is trying to read exists in expected location)                 | No Crash                    |
| 54         | BAD FILE MODE   | No Crash                    |
| 55         | FILE ALREADY OPEN   | No Crash                    |
| 61         | DISK FULL (check to see if hard disk drive VaME is running on isn't full. Seeing this error may cause savedata loss.) | No Crash                    |
| 62         | INPUT PAST END OF FILE (check to see if metadata engine is reading has all the required values within)                | No Crash                    |
| 64         | BAD FILE NAME (file name is illegal. Avoid symbols and log filenames.)  | No Crash                    |
| 67         | TOO MANY FILES (over 15 files are loaded using OPEN.)   | Guru Meditation             |
| 73         | FEATURE UNAVAILABLE (command used is incompatible).   | No Crash                    |

|     |  |                 |
|-----|--|-----------------|
| 76  | INVALID METADATA FILE PATH   | No Crash        |
| 97  | FALSE FLAG ERROR (for development purposes)  | No Crash        |
| 258 | INVALID HANDLE   | No Crash        |
| 420 | MISSING ENGINE METADATA (required engine metadata file missing. Check dloc\$/engine.ddf)                             | Guru Meditation |
| 421 | MISSING METADATA DIRECTORY (required metadata directory is missing. Check if dloc\$ has all the needed directories.) | Guru Meditation |
| 422 | MISSING DEFAULT SAVE FILE (required default save file is missing. Check sloc\$/defaultsave.ddf)                      | Guru Meditation |
| 423 | MISSING SCRIPT (required script file is missing. Check scriptloc\$/ contents for missing or misplaced scripts.       | No Crash        |
| 424 | MISSING ANIMATION FILE (required animation file is missing. Check aloc\$/anifile\$/)                                 | No Crash        |
| 425 | MISSING TERMINAL FILE (required terminal file is missing. Check tloc\$/runterminal\$/)                               | No Crash        |
| 426 | CONFLICTING LAUNCH PARAMETERS (launch parameters -fix and -noupdate used together)                                   | Guru Meditation |
| 427 | GAME REQUIRES A DIFFERENT ENGINE VERSION   | Guru Meditation |
| 428 | NO CONTROLS DEFINED (required input codes are not present)   | Guru Meditation |
| 429 | NO FONT DEFINED (required font file or metadata not present)   | Guru Meditation |
| 430 | Unsupported/unknown OS   | Guru Meditation |
| 431 | CANNOT LOAD MORE THAN ONE MOD AT ONCE  | Guru Meditation |
| 432 | REQUESTED MOD NOT FOUND  | Guru Meditation |
| 433 | CONFLICTING LAUNCH PARAMETERS (launch parameters -windowed and -full-  | Guru Meditation |

|                                  |                               |                 |
|----------------------------------|-------------------------------|-----------------|
|                                  | screen used together)         |                 |
| 434                              | DEFAULT OPTIONS FILE MISSING! | Guru Meditation |
| 435                              | Missing full screen image!    | No Crash        |
| 436                              | Division by Zero!             | No Crash        |
| 666                              | Demonic Error                 | Guru Meditation |
| any other number not listed here | UNKNOWN ERROR                 | Guru Meditation |

## **Mandatory Music**

The following is a list of the mandatory music files that must exist.

| <b>Filename</b> | <b>Description</b>   |
|-----------------|----------------------|
| menu.ogg        | Music for main menu. |

## Mandatory Sound Effects

The following is a list of the mandatory music files that must exist.

| Filename         | Description   |
|------------------|---|
| move.ogg         | Move menu/pocket/terminal item.                       |
| select.ogg       | Select menu/pocket/terminal item.                     |
| terminalon.ogg   | Sound played when terminal is switched on.            |
| terminaloff.ogg  | Sound played when terminal is switched off.           |
| openpocket.ogg   | Sound played when pocket is opened.                   |
| closepocket.ogg  | Sound played when pocket is closed.                   |
| pickup.ogg       | Sound played when pocket item/currency is picked up.  |
| drop.ogg         | Sound played when pocket item/currency is given away. |
| lightning.ogg    | Sound played for lightning during storm map effect.   |
| awardgranted.ogg | Sound played when an award is granted to the player.  |
| devlogo.ogg      | Sound played when developer logo is displayed.        |

## Launch Parameters

**Windows:** `vame_win.exe` [parameters]

**Linux:** `./vame_linux` [parameters]

| Parameter              | Description  |
|------------------------|--|
| -console               | Forces engine to display engine console (only if dev-mode is switched on.)   |
| -devmode               | Forces engine to run in developer mode.  |
| -erasesave             | Erases save then closes game.  |
| -fix                   | Attempts to fix any damaged files by forcing an update install. Cannot be used with -noupdate parameter.   |
| -fullscreen            | Launches in fullscreen mode overriding user settings. Cannot be used with -windowed parameter.   |
| -lite                  | Skips loading of music and sound effects during launch. This will benefit slower machines that take a long time to decode audio and/or have low RAM. All audio is disabled when this mode is active. |
| -mod=[mod folder name] | Launches a mod in the requested folder. Brackets not required. Cannot be used twice or with -fix parameter. Automatically disables updates. Must be the last parameter or it will fail.              |
| -nofx                  | Disables any effects the engine uses. Good at reducing flicker on slower machines.   |
| -nosave                | Prevents game from saving automatically if autosave is enabled.  |
| -noupdate              | Skips update checker. Cannot be used with -fix parameter.  |
| -windowed              | Launches in windowed mode overriding user settings. Cannot be used with -fullscreen parameter.   |



## Input Values

| Value | Description            |
|-------|------------------------|
| a     | Walking controls / map |
| b     | Pockets                |
| c     | Pockets                |
| d     | Pockets + Text Banner  |
| dd    | Text Banner            |
| ddd   | Text Banner            |
| t     | Terminal               |
| tt    | Terminal / Readtxt     |
| ss    | Fullscreen Image       |
| ci    | Choice Banner          |