

C

Generated by Doxygen 1.9.8



<b>1 Table of contents</b>	<b>1</b>
1.1 System calls	1
1.2 Basic types	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 File Index</b>	<b>5</b>
3.1 File List	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 file_handler Struct Reference	7
4.2 known_types Struct Reference	7
4.3 list_handler Struct Reference	8
4.4 memory_info_data Struct Reference	8
4.5 pointer Struct Reference	9
4.6 stack_element Struct Reference	9
4.7 stack_handler Struct Reference	10
4.8 string_reference Struct Reference	10
4.9 type_methods_definitions Struct Reference	10
4.10 virtual_pointer Struct Reference	11
4.11 virtual_pointer_enumerator Struct Reference	11
4.12 vm Struct Reference	12
4.13 vm_pointer Struct Reference	13
4.14 vm_state Struct Reference	14
<b>5 File Documentation</b>	<b>15</b>
5.1 api.h	15
5.2 compile.h	16
5.3 data.h	16
5.4 errors.h	17
5.5 headers.h	18
5.6 macros.h	18
5.7 time.h	19
5.8 version.h	20
5.9 info_v1.h	20
5.10 src/sys/list/list_v1.h File Reference	20
5.10.1 Detailed Description	21
5.11 list_v1.h	21
5.12 memory_v1.h	22
5.13 options_v1.h	23
5.14 src/sys/os/os_v1.h File Reference	23
5.14.1 Detailed Description	24
5.15 os_v1.h	24

5.16 src/vm/v1/pointer/pointer_v1.h File Reference . . . . .	25
5.16.1 Detailed Description . . . . .	26
5.17 pointer_v1.h . . . . .	26
5.18 src/vm/v1/types/data/data_v1.h File Reference . . . . .	26
5.18.1 Detailed Description . . . . .	27
5.19 data_v1.h . . . . .	27
5.20 src/vm/v1/types/file/file_v1.h File Reference . . . . .	28
5.20.1 Detailed Description . . . . .	29
5.21 file_v1.h . . . . .	29
5.22 src/vm/v1/types/object/object_v1.h File Reference . . . . .	30
5.22.1 Detailed Description . . . . .	30
5.23 object_v1.h . . . . .	31
5.24 stack_v1.h . . . . .	31
5.25 src/vm/v1/types/string/string_v1.h File Reference . . . . .	32
5.25.1 Detailed Description . . . . .	33
5.26 string_v1.h . . . . .	33
5.27 src/vm/v1/types/string_pointer/string_pointer_v1.h File Reference . . . . .	34
5.27.1 Detailed Description . . . . .	34
5.28 string_pointer_v1.h . . . . .	34
5.29 src/vm/v1/types/user/user_v1.h File Reference . . . . .	35
5.29.1 Detailed Description . . . . .	36
5.30 user_v1.h . . . . .	36
5.31 virtual_v1.h . . . . .	36
<b>Index</b>	<b>39</b>

# Chapter 1

## Table of contents

### 1.1 System calls

- info
- list
- memory
- options
- os

### 1.2 Basic types

- data
- file
- object
- stack
- string
- string\_pointer
- user



## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">file_handler</a>	7
<a href="#">known_types</a>	7
<a href="#">list_handler</a>	8
<a href="#">memory_info_data</a>	8
<a href="#">pointer</a>	9
<a href="#">stack_element</a>	9
<a href="#">stack_handler</a>	10
<a href="#">string_reference</a>	10
<a href="#">type_methods_definitions</a>	10
<a href="#">virtual_pointer</a>	11
<a href="#">virtual_pointer_enumerator</a>	11
<a href="#">vm</a>	12
<a href="#">vm_pointer</a>	13
<a href="#">vm_state</a>	14





## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

src/std/ <a href="#">api.h</a> . . . . .	15
src/std/ <a href="#">compile.h</a> . . . . .	16
src/std/ <a href="#">data.h</a> . . . . .	16
src/std/ <a href="#">errors.h</a> . . . . .	17
src/std/ <a href="#">headers.h</a> . . . . .	18
src/std/ <a href="#">macros.h</a> . . . . .	18
src/std/ <a href="#">time.h</a> . . . . .	19
src/std/ <a href="#">version.h</a> . . . . .	20
src/sys/info/ <a href="#">info_v1.h</a> . . . . .	20
src/sys/list/ <a href="#">list_v1.h</a> . . . . .	
C API / list . . . . .	20
src/sys/memory/ <a href="#">memory_v1.h</a> . . . . .	22
src/sys/options/ <a href="#">options_v1.h</a> . . . . .	23
src/sys/os/ <a href="#">os_v1.h</a> . . . . .	
C API / os . . . . .	23
src/vm/v1/pointer/ <a href="#">pointer_v1.h</a> . . . . .	
C API / pointer . . . . .	25
src/vm/v1/types/data/ <a href="#">data_v1.h</a> . . . . .	
C API / data . . . . .	26
src/vm/v1/types/file/ <a href="#">file_v1.h</a> . . . . .	
C API / file . . . . .	28
src/vm/v1/types/object/ <a href="#">object_v1.h</a> . . . . .	
C API / object . . . . .	30
src/vm/v1/types/stack/ <a href="#">stack_v1.h</a> . . . . .	31
src/vm/v1/types/string/ <a href="#">string_v1.h</a> . . . . .	
C API / string . . . . .	32
src/vm/v1/types/string_pointer/ <a href="#">string_pointer_v1.h</a> . . . . .	
C API / string pointer . . . . .	34
src/vm/v1/types/user/ <a href="#">user_v1.h</a> . . . . .	
C API / user . . . . .	35
src/vm/v1/virtual/ <a href="#">virtual_v1.h</a> . . . . .	36



## Chapter 4

# Class Documentation

### 4.1 file\_handler Struct Reference

#### Public Attributes

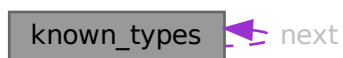
- FILE \* **file**

The documentation for this struct was generated from the following files:

- src/vm/v1/pointer/pointer\_v1.c
- src/vm/v1/types/file/file\_v1.c

### 4.2 known\_types Struct Reference

Collaboration diagram for known\_types:



#### Public Attributes

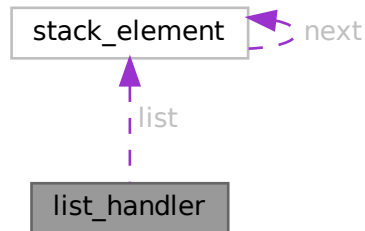
- u64 **id**
- [known\\_types\\_ptr](#) **next**
- void(\* **free** )([pointer\\_ptr](#) ptr)

The documentation for this struct was generated from the following file:

- src/vm/v1/pointer/pointer\_v1.c

### 4.3 list\_handler Struct Reference

Collaboration diagram for list\_handler:



#### Public Attributes

- [stack\\_ptr](#) `list`

The documentation for this struct was generated from the following files:

- `src/vm/v1/pointer/pointer_v1.c`
- `src/vm/v1/types/string/string_v1.c`

### 4.4 memory\_info\_data Struct Reference

#### Public Attributes

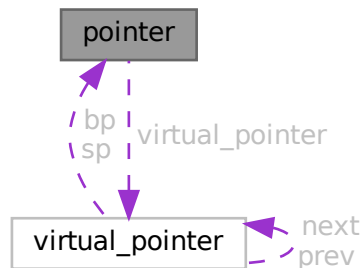
- `u64 alloc`
- `u64 used`
- `u64 free`

The documentation for this struct was generated from the following file:

- `src/sys/memory/memory_v1.c`

## 4.5 pointer Struct Reference

Collaboration diagram for pointer:



### Public Attributes

- [virtual\\_pointer\\_ptr](#) `virtual_pointer`
- `void * data`
- `u64 size`
- `u64 address`
- `u64 id`

The documentation for this struct was generated from the following file:

- `src/vm/v1/pointer/pointer_v1.c`

## 4.6 stack\_element Struct Reference

Collaboration diagram for `stack_element`:



### Public Attributes

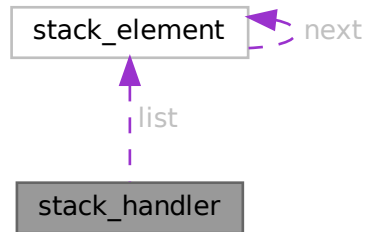
- [stack\\_ptr](#) `next`
- `void * data`

The documentation for this struct was generated from the following file:

- `src/std/data.h`

## 4.7 `stack_handler` Struct Reference

Collaboration diagram for `stack_handler`:



### Public Attributes

- u64 **size**
- [stack\\_ptr](#) **list**

The documentation for this struct was generated from the following file:

- `src/vm/v1/types/stack/stack_v1.c`

## 4.8 `string_reference` Struct Reference

### Public Attributes

- u64 **address**
- u64 **offset**

The documentation for this struct was generated from the following file:

- `src/vm/v1/types/string/string_v1.c`

## 4.9 `type_methods_definitions` Struct Reference

### Public Attributes

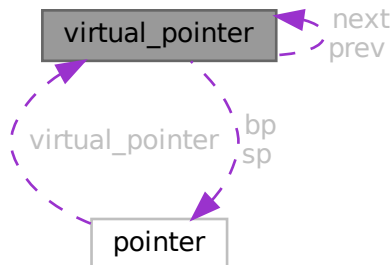
- void(\* **free** )(pointer\_ptr ptr)

The documentation for this struct was generated from the following file:

- `src/std/data.h`

## 4.10 virtual\_pointer Struct Reference

Collaboration diagram for virtual\_pointer:



### Public Attributes

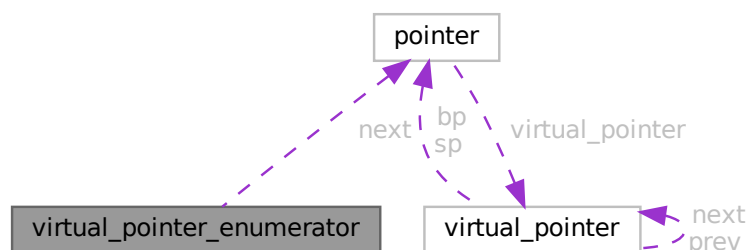
- `pointer_ptr * sp`
- `pointer_ptr * bp`
- `virtual_pointer_ptr prev`
- `virtual_pointer_ptr next`
- `u64 offset`
- `u64 size`

The documentation for this struct was generated from the following file:

- `src/vm/v1/virtual/virtual_v1.c`

## 4.11 virtual\_pointer\_enumerator Struct Reference

Collaboration diagram for virtual\_pointer\_enumerator:



### Public Attributes

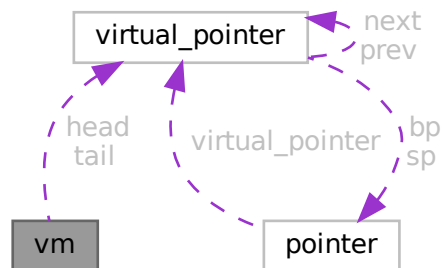
- [pointer\\_ptr](#)(\* [next](#) )(void)

The documentation for this struct was generated from the following file:

- `src/vm/v1/virtual/virtual_v1.c`

## 4.12 vm Struct Reference

Collaboration diagram for vm:



### Public Attributes

- [virtual\\_pointer\\_ptr](#) **head**
- [virtual\\_pointer\\_ptr](#) **tail**

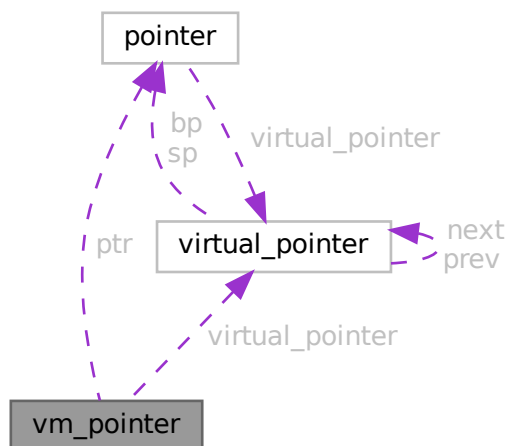
The documentation for this struct was generated from the following file:

- `src/vm/v1/virtual/virtual_v1.c`



## 4.13 vm\_pointer Struct Reference

Collaboration diagram for vm\_pointer:



### Public Attributes

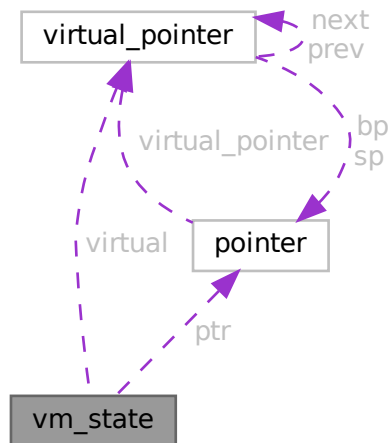
- `pointer_ptr * ptr`
- `virtual_pointer_ptr virtual_pointer`

The documentation for this struct was generated from the following file:

- `src/vm/v1/virtual/virtual_v1.c`

## 4.14 vm\_state Struct Reference

Collaboration diagram for vm\_state:



### Public Attributes

- `virtual_pointer_ptr` **virtual**
- `pointer_ptr * ptr`

The documentation for this struct was generated from the following file:

- `src/vm/v1/virtual/virtual_v1.c`

## Chapter 5

# File Documentation

### 5.1 api.h

```
00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 3, 2025 at 6:17:28 PM GMT+3
00008  *
00009  */
00010 /*
00011      Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013      This program is free software: you can redistribute it and/or modify
00014      it under the terms of the GNU General Public License as published by
00015      the Free Software Foundation, either version 3 of the License, or
00016      (at your option) any later version.
00017
00018      This program is distributed in the hope that it will be useful,
00019      but WITHOUT ANY WARRANTY; without even the implied warranty of
00020      MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021      GNU General Public License for more details.
00022
00023      You should have received a copy of the GNU General Public License
00024      along with this program. If not, see <https://www.gnu.org/licenses/>.
00025  */
00026
00027 #ifndef _STD_API_H_
00028 #define _STD_API_H_
00029
00030 #define USING_API
00031
00032 #define USING_DATA
00033 #define USING_HEADERS
00034 #define USING_MACROS
00035
00036 #include "std/data.h"
00037 #include "std/headers.h"
00038 #include "std/macros.h"
00039 #include "std/version.h"
00040
00041 #define VERSION 1
00042 #define MAJOR 0
00043 #define MINOR 0
00044 #define REVISION 0
00045
00046 #define VM_EXPAND(X) X
00047 #define VM_CONCAT_(A, B) A##B
00048 #define VM_CONCAT(A, B) VM_CONCAT_(A, B)
00049 #define VM_STRINGIFY_(X) #X
00050 #define VM_STRINGIFY(X) VM_STRINGIFY_(X)
00051
00052 #define PRIVATE_API(x) VM_CONCAT(API(x), VM_CONCAT(_, GIT_COMMIT_HASH_VALUE))
00053 #define API_VERSION VM_EXPAND(VM_CONCAT(, VM_STRINGIFY(VM_CONCAT(VERSION, VM_CONCAT(.,
VM_CONCAT(MAJOR, VM_CONCAT(., VM_CONCAT(MINOR, VM_CONCAT(., REVISION))))))))
00054 #define API(x) VM_CONCAT(x, VM_CONCAT(, VM_CONCAT(v, VM_CONCAT(VERSION, VM_CONCAT(, VM_CONCAT(MAJOR,
VM_CONCAT(, VM_CONCAT(MINOR, VM_CONCAT(, REVISION))))))))
00055
00056 #endif /* _STD_API_H_ */
```

## 5.2 compile.h

```

00001 /**--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 9, 2025 at 12:49:35 PM GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program.  If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _STD_COMPILE_H_
00028 #define _STD_COMPILE_H_
00029
00030 #ifdef CONFIG_MEMORY_DEBUG_INFO
00031 #if CONFIG_MEMORY_DEBUG_INFO == 1
00032 #define USE_MEMORY_DEBUG_INFO
00033 #endif
00034 #endif
00035
00036 #ifdef CONFIG_MEMORY_CLEANUP
00037 #if CONFIG_MEMORY_CLEANUP == 1
00038 #define USE_MEMORY_CLEANUP
00039 #endif
00040 #endif
00041
00042 #ifdef CONFIG_MEMORY_ALLOC
00043 #if CONFIG_MEMORY_ALLOC == 1
00044 #define USE_MEMORY_ALLOC
00045 #endif
00046 #endif
00047
00048 #ifdef CONFIG_GC
00049 #if CONFIG_GC == 1
00050 #define USE_GC
00051 #endif
00052 #endif
00053
00054 #ifdef CONFIG MOCKS
00055 #if CONFIG MOCKS == 1
00056 #define USE MOCKS
00057 #endif
00058 #endif
00059
00060 #ifdef CONFIG_MEMCPY
00061 #if CONFIG_MEMCPY == 1
00062 #define USE_MEMCPY
00063 #endif
00064 #endif
00065
00066 #endif /* _STD_COMPILE_H_ */

```

## 5.3 data.h

```

00001 /**--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 9, 2025 at 12:09:41 PM GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012

```

```

00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program. If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _STD_DATA_H_
00028 #define _STD_DATA_H_
00029
00030 #define USING_DATA
00031
00032 typedef unsigned long long u64;
00033 typedef unsigned long int u32;
00034 typedef unsigned short int u16;
00035 typedef unsigned char u8;
00036
00037 typedef signed long long s64;
00038 typedef signed long int s32;
00039 typedef signed short int s16;
00040 typedef signed char s8;
00041
00042 typedef struct stack_element* stack_ptr;
00043 typedef struct stack_element {
00044     /* points to next node */
00045     stack_ptr next;
00046     /* data */
00047     void* data;
00048 } stack_element;
00049
00050 typedef struct pointer* pointer_ptr;
00051 typedef const struct pointer* const_pointer_ptr;
00052 typedef struct virtual_pointer* virtual_pointer_ptr;
00053 typedef const struct virtual_pointer* const_virtual_pointer_ptr;
00054
00055 typedef struct type_methods_definitions {
00056     void (*free)(pointer_ptr ptr);
00057 } type_methods_definitions;
00058
00059 enum type {
00060     /* value used for ephemeral type - null */
00061     TYPE_NULL = 0,
00062     /* value used for pointer type - ref */
00063     TYPE_DATA = 1,
00064     /* value used for string type - string */
00065     TYPE_STRING = 2,
00066     /* value used for string ref type - string ref */
00067     TYPE_STRING_POINTER = 3,
00068     /* value used for file type - file */
00069     TYPE_FILE = 4,
00070     /* value used for stack type - stack */
00071     TYPE_STACK = 5,
00072     /* value used for object type - object */
00073     TYPE_OBJECT = 6,
00074     /* value used for user type - user (id: +0, +1, +2, +3, ...) */
00075     TYPE_USER = 7,
00076 };
00077
00078 #endif /* _STD_DATA_H_ */

```

## 5.4 errors.h

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   11 December 2023 at 9:16:31 GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by

```

```

00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program. If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _STD_ERRORS_H_
00028 #define _STD_ERRORS_H_
00029
00030 #define ERROR_MSG_INDEX(n) (-n)
00031
00032 #define ERROR_MSG_ID_INVALID_ARGUMENT ERROR_MSG_INDEX((u64)1)
00033
00034 #define ERROR(x) ERROR_MSG_ID_##x
00035 #define ERROR_INVALID_ARGUMENT ERROR(INVALID_ARGUMENT)
00036
00037 #endif /* _STD_ERRORS_H_ */

```

## 5.5 headers.h

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 3, 2025 at 10:36:19 PM GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program. If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _STD_HEADERS_H_
00028 #define _STD_HEADERS_H_
00029
00030 #define USING_HEADERS
00031
00032 #include <stdint.h>
00033
00034 #ifdef _WIN32
00035 #include <windows.h>
00036 #define PATH_MAX MAX_PATH
00037 #else
00038 #include <sys/time.h>
00039 #include <unistd.h>
00040 #endif
00041
00042 #include <limits.h>
00043
00044 #endif /* _STD_HEADERS_H_ */

```

## 5.6 macros.h

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:

```

```

00007  *    February 3, 2025 at 5:06:59 PM GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program. If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _STD_MACROS_H_
00028 #define _STD_MACROS_H_
00029
00030 #define USING_MACROS
00031
00032 #define CLEAN(ptr) \
00033     if (ptr != 0) { \
00034         ptr = 0; \
00035     }
00036
00037 #ifdef __GNUC__
00038 #define FALL_THROUGH __attribute__((fallthrough))
00039 #define ATTRIBUTE
00040 #define INIT __attribute__((constructor))
00041 #define DESTROY __attribute__((destructor))
00042 #else
00043 #define INIT
00044 #define DESTROY
00045 #define FALL_THROUGH ((void)0)
00046 #endif /* __GNUC__ >= 7 */
00047
00048 #define RXP_DEBUG_TESTS
00049
00050 #endif /* _STD_MACROS_H_ */

```

## 5.7 time.h

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  * Yes
00004  * Created:
00005  * 11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  * 12 December 2023 at 9:49:00 GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program. If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _STD_TIME_H_
00028 #define _STD_TIME_H_
00029
00030 #define TIME_INIT(s) \
00031     struct timeval s##t1; \
00032     struct timeval s##t2; \
00033     struct timeval s##diff;
00034
00035 #define TIME_START(s) \
00036     gettimeofday(&s##t1, NULL);

```

```

00037
00038 #define TIME_END(s) \
00039     gettimeofday(&s##t2, NULL); \
00040     TIME_CALC(s);
00041
00042 #define TIME_CALC(s) \
00043     s##diff.tv_sec = s##t2.tv_sec - s##t1.tv_sec; \
00044     s##diff.tv_usec = s##t2.tv_usec - s##t1.tv_usec;
00045
00046 #define TIME_VALUE(s) \
00047     (u64)(s##diff.tv_sec * 1000000 + s##diff.tv_usec)
00048
00049 #define TIME_PRINT(s) printf("%0.3Lf\n", (long double)(TIME_VALUE(s)) / 1000.0);
00050
00051 #endif /* _STD_TIME_H_ */

```

## 5.8 version.h

```

00001 #define GIT_COMMIT_HASH_VALUE a69a6c80c68bfc91350e87190790022a0ac8b2ed
00002 #define GIT_COMMIT_HASH "a69a6c80c68bfc91350e87190790022a0ac8b2ed"
00003 #define UNIX_TIMESTAMP 1739099470

```

## 5.9 info\_v1.h

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 5:55:55 PM GMT+3
00008  *
00009  */
00010 /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013  This program is free software: you can redistribute it and/or modify
00014  it under the terms of the GNU General Public License as published by
00015  the Free Software Foundation, either version 3 of the License, or
00016  (at your option) any later version.
00017
00018  This program is distributed in the hope that it will be useful,
00019  but WITHOUT ANY WARRANTY; without even the implied warranty of
00020  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021  GNU General Public License for more details.
00022
00023  You should have received a copy of the GNU General Public License
00024  along with this program. If not, see <https://www.gnu.org/licenses/>.
00025  */
00026 #ifndef _POINTER_INFO_V1_H_
00027 #define _POINTER_INFO_V1_H_
00028
00029 #include "std/api.h"
00030
00031 typedef struct PRIVATE_API(info_methods) {
00032     const u64 timestamp;
00033     const char* commit;
00034     const char* version;
00035 } info_methods;
00036
00037 /* definition */
00038 extern const info_methods PRIVATE_API(info_methods_definitions);
00039
00040 /* definition */
00041 #ifndef INLINE
00042 const info_methods* info = &PRIVATE_API(info_methods_definitions);
00043 #else
00044 /* definition */
00045 static const info_methods* sys_info = &PRIVATE_API(info_methods_definitions);
00046 #endif
00047
00048 #endif /* _POINTER_INFO_V1_H_ */

```

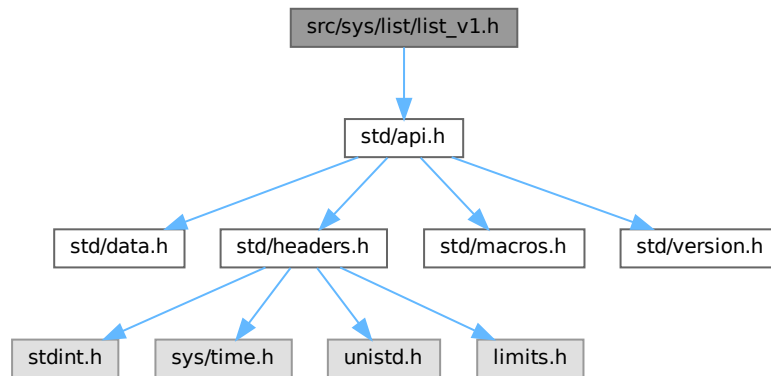
## 5.10 src/sys/list/list\_v1.h File Reference

C API / list.



```
#include "std/api.h"
```

Include dependency graph for list\_v1.h:



## Functions

- struct **PRIVATE\_API** (list\_methods)
- const list\_methods **PRIVATE\_API** (list\_methods\_definitions)

## Variables

- list\_methods

### 5.10.1 Detailed Description

C API / list.

structure for organizing stack functions

## 5.11 list\_v1.h

[Go to the documentation of this file.](#)

```

00001  /**--coding:utf-8 --
00002  * Auto updated?
00003  * Yes
00004  * Created:
00005  * 11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  * February 9, 2025 at 1:15:21 PM GMT+3
00008  *
00009  */
00010  /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013  This program is free software: you can redistribute it and/or modify
00014  it under the terms of the GNU General Public License as published by
00015  the Free Software Foundation, either version 3 of the License, or
00016  (at your option) any later version.
00017
00018  This program is distributed in the hope that it will be useful,

```

```

00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program.  If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _LIST_MICRO_DATA_H_
00028 #define _LIST_MICRO_DATA_H_
00029
00030 #include "std/api.h"
00031
00032 typedef struct PRIVATE_API(list_methods) {
00033     /* initialize context */
00034     void (*init)(stack_ptr* current);
00035     /* destroy context */
00036     void (*destroy)(stack_ptr* current);
00037     /* push item on current context (stack) */
00038     void (*push)(stack_ptr* current, void* item);
00039     /* pop item on current context (stack) */
00040     void* (*pop)(stack_ptr* current);
00041     /* peek item on current context (stack) */
00042     void* (*peek)(stack_ptr* current);
00043 #ifdef USE_MEMORY_DEBUG_INFO
00044     /* print head */
00045     void (*print_head)(stack_ptr* current);
00046     /* print */
00047     void (*print)(stack_ptr* current);
00048 #endif
00049 } list_methods;
00050
00051 /* definition */
00052 extern const list_methods PRIVATE_API(list_methods_definitions);
00053
00054 /* definition */
00055 #ifdef INLINE
00056 const list_v1* list_methods = &PRIVATE_API(list_methods_definitions);
00057 #else
00058 /* definition */
00059 static const list_methods* sys_list = &PRIVATE_API(list_methods_definitions);
00060 #endif
00061
00062 #endif /* _LIST_MICRO_DATA_H_ */

```

## 5.12 memory\_v1.h

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 9, 2025 at 1:15:19 PM GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program.  If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _GENERIC_MEMORY_H_
00028 #define _GENERIC_MEMORY_H_
00029
00030 #include "std/api.h"
00031
00032 #ifdef USE_MEMORY_DEBUG_INFO
00033 extern void global_statistics(void);
00034 #endif
00035
00036 typedef struct PRIVATE_API(memory_methods) {

```

```

00037     void* (*alloc)(u64 size);
00038     void (*free)(void* ptr, u64 size);
00039     void* (*realloc)(void* old_ptr, u64 size, u64 new_size);
00040 #ifdef USE_MEMORY_DEBUG_INFO
00041     void (*set)(void* dest, u8 c, u64 count);
00042 #endif
00043 } memory_methods;
00044
00045 /* definition */
00046 extern const memory_methods PRIVATE_API(memory_methods_definitions);
00047
00048 /* definition */
00049 #ifdef INLINE
00050 const memory_methods* memory = &PRIVATE_API(memory_methods_definitions);
00051 #else
00052 /* definition */
00053 static const memory_methods* sys_memory = &PRIVATE_API(memory_methods_definitions);
00054 #endif
00055
00056 #endif /* _GENERIC_MEMORY_H_ */

```

## 5.13 options\_v1.h

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 9, 2025 at 1:15:01 PM GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program. If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _VM_V1_OPTIONS_H_
00028 #define _VM_V1_OPTIONS_H_
00029
00030 #include "std/api.h"
00031
00032 typedef struct PRIVATE_API(options_methods) {
00033     const u64 size;
00034 } options_methods;
00035
00036 /* definition */
00037 extern const options_methods PRIVATE_API(options_methods_definitions);
00038
00039 /* definition */
00040 #ifdef INLINE
00041 const options_methods* options = &PRIVATE_API(options_methods_definitions);
00042 #else
00043 /* definition */
00044 static const options_methods* sys_options = &PRIVATE_API(options_methods_definitions);
00045 #endif
00046
00047 #endif /* _VM_V1_OPTIONS_H_ */

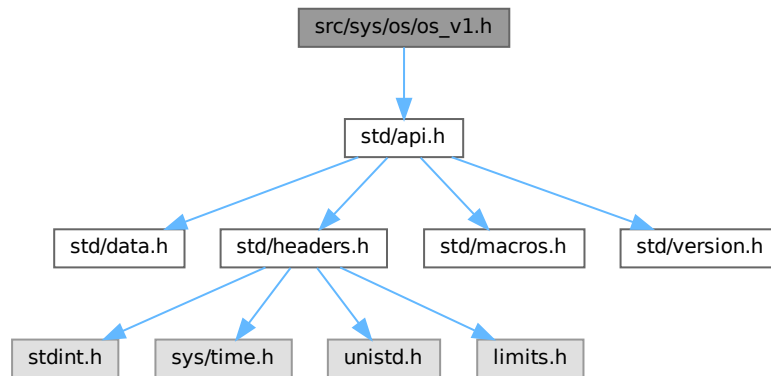
```

## 5.14 src/sys/os/os\_v1.h File Reference

C API / os.

```
#include "std/api.h"
```

Include dependency graph for os\_v1.h:



## Functions

- struct **PRIVATE\_API** (os\_methods)
- const os\_methods **PRIVATE\_API** (os\_methods\_definitions)

## Variables

- **os\_methods**

### 5.14.1 Detailed Description

C API / os.

## 5.15 os\_v1.h

[Go to the documentation of this file.](#)

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 7:05:39 PM GMT+3
00008  *
00009 */
00010 /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013  This program is free software: you can redistribute it and/or modify
00014  it under the terms of the GNU General Public License as published by
00015  the Free Software Foundation, either version 3 of the License, or
00016  (at your option) any later version.
00017
00018  This program is distributed in the hope that it will be useful,
00019  but WITHOUT ANY WARRANTY; without even the implied warranty of
00020  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021  GNU General Public License for more details.

```

```

00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program.  If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _POINTER_TYPES_OS_H_
00028 #define _POINTER_TYPES_OS_H_
00029
00030 #include "std/api.h"
00031
00032 typedef struct PRIVATE_API(os_methods) {
00033     u64 (*getenv) (u64 name);
00034     u64 (*getcwd) (void);
00035     void (*putc) (u64 ptr);
00036 } os_methods;
00037
00038 /* definition */
00039 extern const os_methods PRIVATE_API(os_methods_definitions);
00040
00041 /* definition */
00042 #ifndef INLINE
00043 const os_methods* os = &PRIVATE_API(os_methods_definitions);
00044 #else
00045 /* definition */
00046 static const os_methods* os = &PRIVATE_API(os_methods_definitions);
00047 #endif
00048 #endif /* _POINTER_TYPES_OS_H_ */

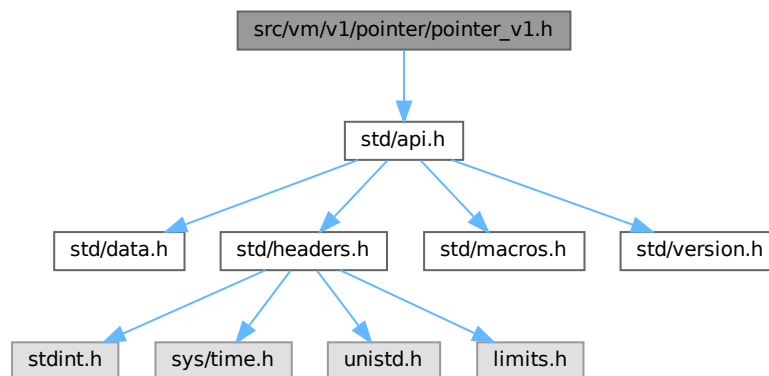
```

## 5.16 src/vm/v1/pointer/pointer\_v1.h File Reference

C API / pointer.

```
#include "std/api.h"
```

Include dependency graph for pointer\_v1.h:



### Functions

- struct **PRIVATE\_API** (pointer\_methods)
- const pointer\_methods **PRIVATE\_API** (pointer\_methods\_definitions)

### Variables

- **pointer\_methods**

### 5.16.1 Detailed Description

C API / pointer.

## 5.17 pointer\_v1.h

[Go to the documentation of this file.](#)

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  * Yes
00004  * Created:
00005  * 11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  * February 9, 2025 at 2:21:36 PM GMT+3
00008  *
00009  */
00010 /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013  This program is free software: you can redistribute it and/or modify
00014  it under the terms of the GNU General Public License as published by
00015  the Free Software Foundation, either version 3 of the License, or
00016  (at your option) any later version.
00017
00018  This program is distributed in the hope that it will be useful,
00019  but WITHOUT ANY WARRANTY; without even the implied warranty of
00020  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021  GNU General Public License for more details.
00022
00023  You should have received a copy of the GNU General Public License
00024  along with this program. If not, see <https://www.gnu.org/licenses/>.
00025  */
00026
00027 #ifndef _POINTER_V1_H_
00028 #define _POINTER_V1_H_
00029
00030 #include "std/api.h"
00031
00032 typedef struct PRIVATE_API(pointer_methods) {
00033     void (*init)(u64 size);
00034     void (*destroy)(void);
00035     void (*gc)(void);
00036     pointer_ptr (*alloc)(u64 size, u64 id);
00037     void (*realloc)(pointer_ptr ptr, u64 size);
00038     u64 (*register_type)(u64 id, const struct type_methods_definitions* data_type);
00039     void (*free)(u64 ptr);
00040     u64 (*address)(const_pointer_ptr ptr);
00041     virtual_pointer_ptr (*virtual)(const_pointer_ptr ptr);
00042     void (*release)(pointer_ptr ptr);
00043     u64 (*size)(const_pointer_ptr ptr);
00044     void* (*read)(const_pointer_ptr ptr);
00045     u64 (*read_type)(const_pointer_ptr ptr, u64 id);
00046     void (*write)(pointer_ptr ptr, virtual_pointer_ptr m, u64 address);
00047 #ifdef USE_MEMORY_DEBUG_INFO
00048     void (*dump)(pointer_ptr ptr);
00049     void (*dump_ref)(void** ptr);
00050 #endif
00051 } pointer_methods;
00052
00053 /* definition */
00054 extern const pointer_methods PRIVATE_API(pointer_methods_definitions);
00055
00056 /* definition */
00057 #ifdef INLINE
00058 const spointer_methods* pointer = &PRIVATE_API(pointer_methods_definitions);
00059 #else
00060 /* definition */
00061 static const pointer_methods* pointer = &PRIVATE_API(pointer_methods_definitions);
00062 #endif
00063 #endif /* _POINTER_V1_H_ */

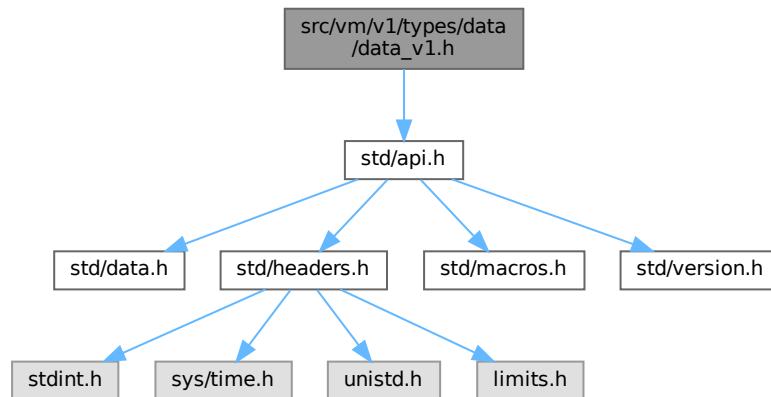
```

## 5.18 src/vm/v1/types/data/data\_v1.h File Reference

C API / data.

```
#include "std/api.h"
```

Include dependency graph for data\_v1.h:



## Functions

- struct **PRIVATE\_API** (data\_methods)
- const data\_methods **PRIVATE\_API** (data\_methods\_definitions)

## Variables

- data\_methods

### 5.18.1 Detailed Description

C API / data.

## 5.19 data\_v1.h

[Go to the documentation of this file.](#)

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 6:56:25 PM GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
  
```

```

00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program.  If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #define USING_API
00028
00029 #ifndef _POINTER_TYPES_DATA_H_
00030 #define _POINTER_TYPES_DATA_H_
00031
00032 #include "std/api.h"
00033
00038 typedef struct PRIVATE_API(data_methods) {
00039     u64 (*alloc)(u64 size);
00040     void (*free)(u64 ptr);
00041     void* (*unsafe)(u64 ptr);
00042     u64 (*size)(u64 ptr);
00043 } data_methods;
00044
00045 /* definition */
00046 extern const data_methods PRIVATE_API(data_methods_definitions);
00047
00048 /* definition */
00049 #ifdef INLINE
00050 const struct data_methods* data = &PRIVATE_API(data_methods_definitions);
00051 #else
00052 /* definition */
00053 static const data_methods* data = &PRIVATE_API(data_methods_definitions);
00054 #endif
00055
00056 #endif /* _POINTER_TYPES_DATA_H_ */

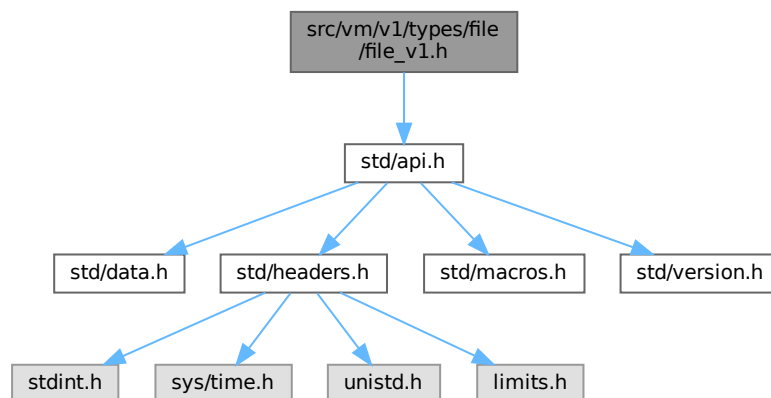
```

## 5.20 src/vm/v1/types/file/file\_v1.h File Reference

C API / file.

```
#include "std/api.h"
```

Include dependency graph for file\_v1.h:



### Functions

- struct **PRIVATE\_API** (file\_methods)
- const file\_methods **PRIVATE\_API** (file\_methods\_definitions)



## Variables

- `file_methods`

### 5.20.1 Detailed Description

C API / file.



```
struct file_methods {
    u64 (*alloc)(u64 file_path_ptr, u64 mode_ptr);
    void (*free)(u64 ptr);
    u64 (*data)(u64 ptr);
};
```

## 5.21 file\_v1.h

[Go to the documentation of this file.](#)

```
00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 5:55:58 PM GMT+3
00008  *
00009  */
00010 /*
00011     Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013     This program is free software: you can redistribute it and/or modify
00014     it under the terms of the GNU General Public License as published by
00015     the Free Software Foundation, either version 3 of the License, or
00016     (at your option) any later version.
00017
00018     This program is distributed in the hope that it will be useful,
00019     but WITHOUT ANY WARRANTY; without even the implied warranty of
00020     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021     GNU General Public License for more details.
00022
00023     You should have received a copy of the GNU General Public License
00024     along with this program. If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _POINTER_TYPES_FILE_H_
00028 #define _POINTER_TYPES_FILE_H_
00029
00030 #define USING_API
00031
00032 #include "std/api.h"
00033
00048 typedef struct PRIVATE_API(file_methods) {
00049     u64 (*alloc)(u64 file_path_ptr, u64 mode_ptr);
00050     void (*free)(u64 ptr);
00051     u64 (*data)(u64 ptr);
00052 } file_methods;
```

```

00053
00054 /* definition */
00055 extern const file_methods PRIVATE_API(file_methods_definitions);
00056
00057 /* definition */
00058 #ifndef INLINE
00059 const file_methods* file = &PRIVATE_API(file_methods_definitions);
00060 #else
00061 /* definition */
00062 static const file_methods* file = &PRIVATE_API(file_methods_definitions);
00063 #endif
00064
00065 #endif /* _POINTER_TYPES_FILE_H_ */

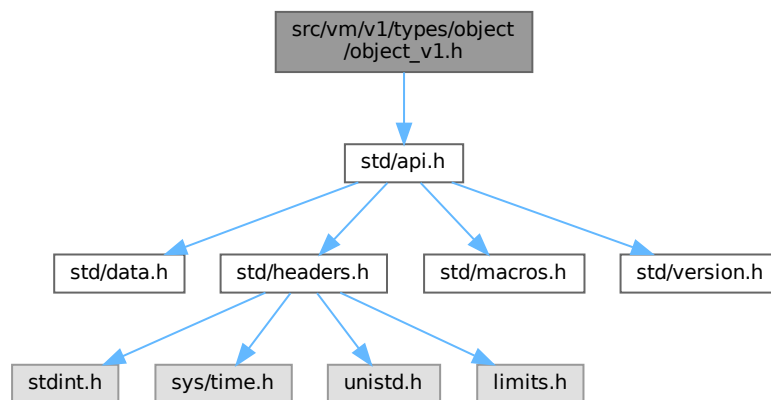
```

## 5.22 src/vm/v1/types/object/object\_v1.h File Reference

C API / object.

```
#include "std/api.h"
```

Include dependency graph for object\_v1.h:



### Functions

- struct **API** (object\_methods)
- const object\_methods **PRIVATE\_API** (object\_methods\_definitions)

### Variables

- **object\_methods**

### 5.22.1 Detailed Description

C API / object.

## 5.23 object\_v1.h

[Go to the documentation of this file.](#)

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 5:55:57 PM GMT+3
00008  *
00009  */
00010 /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013  This program is free software: you can redistribute it and/or modify
00014  it under the terms of the GNU General Public License as published by
00015  the Free Software Foundation, either version 3 of the License, or
00016  (at your option) any later version.
00017
00018  This program is distributed in the hope that it will be useful,
00019  but WITHOUT ANY WARRANTY; without even the implied warranty of
00020  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021  GNU General Public License for more details.
00022
00023  You should have received a copy of the GNU General Public License
00024  along with this program. If not, see <https://www.gnu.org/licenses/>.
00025  */
00026
00027 #ifndef _POINTER_TYPES_OBJECT_H_
00028 #define _POINTER_TYPES_OBJECT_H_
00029
00030 #include "std/api.h"
00031
00032 typedef struct API(object_methods) {
00033     u64 (*alloc)(u64 size);
00034     void (*free)(u64 ptr);
00035     void* (*unsafe)(u64 ptr);
00036     u64 (*load)(const void* data, u64 size);
00037     u64 (*size)(u64 ptr);
00038 } object_methods;
00039
00040 /* definition */
00041 extern const object_methods PRIVATE_API(object_methods_definitions);
00042
00043 /* definition */
00044 #ifdef INLINE
00045 const object_methods* string_pointer = &PRIVATE_API(object_methods_definitions);
00046 #else
00047 /* definition */
00048 static const object_methods* object = &PRIVATE_API(object_methods_definitions);
00049 #endif
00050
00051 #endif /* _POINTER_TYPES_OBJECT_H_ */

```

## 5.24 stack\_v1.h

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 5:55:58 PM GMT+3
00008  *
00009  */
00010 /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013  This program is free software: you can redistribute it and/or modify
00014  it under the terms of the GNU General Public License as published by
00015  the Free Software Foundation, either version 3 of the License, or
00016  (at your option) any later version.
00017
00018  This program is distributed in the hope that it will be useful,
00019  but WITHOUT ANY WARRANTY; without even the implied warranty of
00020  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021  GNU General Public License for more details.
00022
00023  You should have received a copy of the GNU General Public License
00024  along with this program. If not, see <https://www.gnu.org/licenses/>.

```

```

00025 */
00026
00027 #ifndef _POINTER_TYPES_LIST_H_
00028 #define _POINTER_TYPES_LIST_H_
00029
00030 #include "std/api.h"
00031
00032 typedef struct PRIVATE_API(stack_methods) {
00033     u64 (*alloc)(void);
00034     void (*free)(u64 ptr);
00035     u64 (*peek)(u64 list_ptr);
00036     u64 (*peekn)(u64 list_ptr, u64 nelements);
00037     u64 (*pop)(u64 list_ptr);
00038     u64 (*popn)(u64 list_ptr, u64 nelements);
00039     void (*push)(u64 list_ptr, u64 ptr);
00040     u64 (*size)(u64 ptr);
00041     void (*release)(u64 ptr);
00042 } stack_methods;
00043
00044 /* definition */
00045 extern const stack_methods PRIVATE_API(stack_methods_definitions);
00046
00047 /* definition */
00048 #ifndef INLINE
00049 const stack_methods* stack = &PRIVATE_API(stack_methods_definitions);
00050 #else
00051 /* definition */
00052 static const stack_methods* stack = &PRIVATE_API(stack_methods_definitions);
00053 #endif
00054
00055 #endif /* _POINTER_TYPES_LIST_H_ */

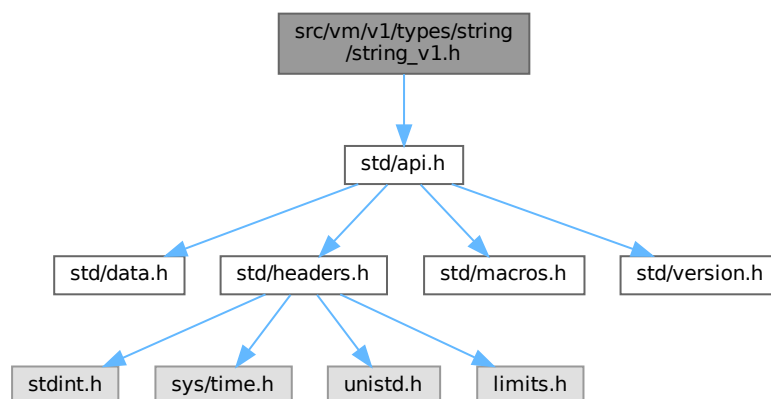
```

## 5.25 src/vm/v1/types/string/string\_v1.h File Reference

C API / string.

```
#include "std/api.h"
```

Include dependency graph for string\_v1.h:



### Functions

- struct **API** (string\_methods)
- const string\_methods **PRIVATE\_API** (string\_methods\_definitions)

**Variables**

- **string\_methods**

**5.25.1 Detailed Description**

C API / string.

**5.26 string\_v1.h**

[Go to the documentation of this file.](#)

```

00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 8:05:38 PM GMT+3
00008  *
00009  */
00010 /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013  This program is free software: you can redistribute it and/or modify
00014  it under the terms of the GNU General Public License as published by
00015  the Free Software Foundation, either version 3 of the License, or
00016  (at your option) any later version.
00017
00018  This program is distributed in the hope that it will be useful,
00019  but WITHOUT ANY WARRANTY; without even the implied warranty of
00020  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021  GNU General Public License for more details.
00022
00023  You should have received a copy of the GNU General Public License
00024  along with this program. If not, see <https://www.gnu.org/licenses/>.
00025  */
00026
00027 #ifndef _POINTER_TYPES_STRING_H_
00028 #define _POINTER_TYPES_STRING_H_
00029
00030 #include "std/api.h"
00031
00032 typedef struct API(string_methods) {
00033     void (*free)(u64 ptr);
00034     u64 (*copy)(u64 ptr);
00035     void (*strcpy)(u64 dest, u64 src);
00036     void (*strcat)(u64 dest, u64 src);
00037     u64 (*strrchr)(u64 src_ptr, u64 match_ptr);
00038     u64 (*strchr)(u64 src_ptr, u64 match_ptr);
00039     u64 (*match)(u64 src_ptr, u64 match_ptr);
00040     u64 (*offset)(u64 src_ptr, u64 match_ptr);
00041     u64 (*load)(const char* data);
00042     void (*put_char)(u64 ptr, char value);
00043     char* (*unsafe)(u64 ptr);
00044     u64 (*size)(u64 ptr);
00045     u64 (*lessthan)(u64 src, u64 dest);
00046     u64 (*greaterthan)(u64 src, u64 dest);
00047     u64 (*equals)(u64 src, u64 dest);
00048     u64 (*compare)(u64 src, u64 dest);
00049     u64 (*left)(u64 src, u64 offset);
00050     u64 (*strncpy)(u64 src, u64 nbytes);
00051     u64 (*left_strncpy)(u64 src, u64 offset);
00052     u64 (*right)(u64 src, u64 offset);
00053     u64 (*move_left)(u64 src, u64 nbytes);
00054     u64 (*move_right)(u64 src, u64 nbytes);
00055     u64 (*strcmp)(u64 src, u64 dest);
00056 } string_methods;
00057
00058 /* definition */
00059 extern const string_methods PRIVATE_API(string_methods_definitions);
00060
00061 /* definition */
00062 #ifdef INLINE
00063 const string_methods* string = &PRIVATE_API(string_methods_definitions);
00064 #else
00065 /* definition */
00066 static const string_methods* string = &PRIVATE_API(string_methods_definitions);
00067 #endif
00068 #endif /* _POINTER_TYPES_STRING_H_ */

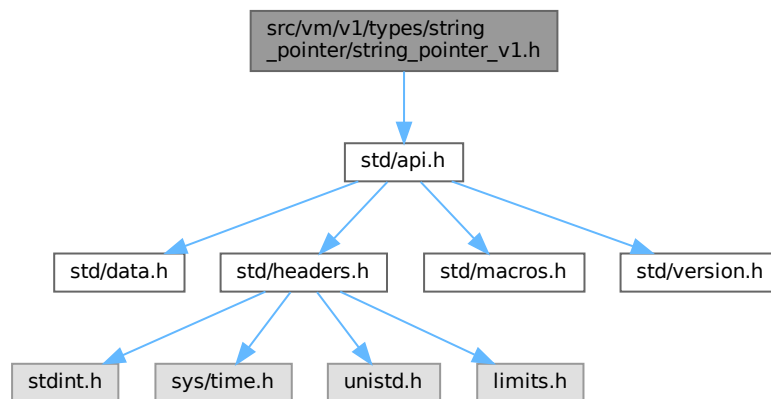
```

## 5.27 src/vm/v1/types/string\_pointer/string\_pointer\_v1.h File Reference

C API / string pointer.

```
#include "std/api.h"
```

Include dependency graph for string\_pointer\_v1.h:



### Functions

- struct **PRIVATE\_API** (string\_pointer\_methods)
- const string\_pointer\_methods **PRIVATE\_API** (string\_pointer\_methods\_definitions)

### Variables

- string\_pointer\_methods

### 5.27.1 Detailed Description

C API / string pointer.

## 5.28 string\_pointer\_v1.h

[Go to the documentation of this file.](#)

```

00001  /**--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 5:55:56 PM GMT+3
00008  *
00009  */
00010  /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012  */

```

```

00013      This program is free software: you can redistribute it and/or modify
00014      it under the terms of the GNU General Public License as published by
00015      the Free Software Foundation, either version 3 of the License, or
00016      (at your option) any later version.
00017
00018      This program is distributed in the hope that it will be useful,
00019      but WITHOUT ANY WARRANTY; without even the implied warranty of
00020      MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021      GNU General Public License for more details.
00022
00023      You should have received a copy of the GNU General Public License
00024      along with this program. If not, see <https://www.gnu.org/licenses/>.
00025 */
00026
00027 #ifndef _POINTER_TYPES_STRING_POINTER_H_
00028 #define _POINTER_TYPES_STRING_POINTER_H_
00029
00030 #define USING_API
00031
00032 #include "std/api.h"
00033
00034 typedef struct PRIVATE_API(string_pointer_methods) {
00035     void (*free)(u64 ptr);
00036 } string_pointer_methods;
00037
00038 /* definition */
00039 extern const string_pointer_methods PRIVATE_API(string_pointer_methods_definitions);
00040
00041 /* definition */
00042 #ifndef INLINE
00043 const string_pointer_methods* string_pointer = &PRIVATE_API(string_pointer_methods_definitions);
00044 #else
00045 /* definition */
00046 static const string_pointer_methods* string_pointer =
00047     &PRIVATE_API(string_pointer_methods_definitions);
00048 #endif
00049 #endif /* _POINTER_TYPES_STRING_POINTER_H_ */

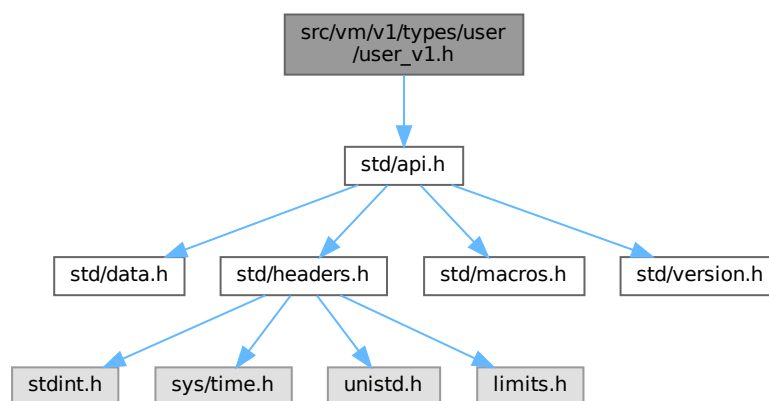
```

## 5.29 src/vm/v1/types/user/user\_v1.h File Reference

C API / user.

```
#include "std/api.h"
```

Include dependency graph for user\_v1.h:



### Functions

- struct **PRIVATE\_API** (user\_methods)
- const user\_methods **PRIVATE\_API** (user\_methods\_definitions)

## Variables

- `user_methods`

### 5.29.1 Detailed Description

C API / user.

## 5.30 user\_v1.h

[Go to the documentation of this file.](#)

```
00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 8, 2025 at 5:55:56 PM GMT+3
00008  *
00009  */
00010 /*
00011      Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013      This program is free software: you can redistribute it and/or modify
00014      it under the terms of the GNU General Public License as published by
00015      the Free Software Foundation, either version 3 of the License, or
00016      (at your option) any later version.
00017
00018      This program is distributed in the hope that it will be useful,
00019      but WITHOUT ANY WARRANTY; without even the implied warranty of
00020      MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021      GNU General Public License for more details.
00022
00023      You should have received a copy of the GNU General Public License
00024      along with this program. If not, see <https://www.gnu.org/licenses/>.
00025  */
00026
00027 #ifndef _POINTER_TYPES_USER_TYPE_H_
00028 #define _POINTER_TYPES_USER_TYPE_H_
00029
00030 #define USING_API
00031
00032 #include "std/api.h"
00033
00034 typedef struct PRIVATE_API(user_methods) {
00035     u64 (*alloc)(void);
00036     void (*free)(u64 ptr);
00037 } user_methods;
00038
00039 /* definition */
00040 extern const user_methods PRIVATE_API(user_methods_definitions);
00041
00042 /* definition */
00043 #ifdef INLINE
00044 const user_methods* user = &PRIVATE_API(user_methods_definitions);
00045 #else
00046 /* definition */
00047 static const user_methods* user = &PRIVATE_API(user_methods_definitions);
00048 #endif
00049
00050 #endif /* _POINTER_TYPES_USER_TYPE_H_ */
```

## 5.31 virtual\_v1.h

```
00001 /*--coding:utf-8 --
00002  * Auto updated?
00003  *   Yes
00004  * Created:
00005  *   11 December 2023 at 9:06:14 GMT+3
00006  * Modified:
00007  *   February 9, 2025 at 12:22:59 PM GMT+3
```



```
00008  *
00009  */
00010  /*
00011  Copyright (C) 2022-2047 Artur Mustafin (artur.mustafin@gmail.com)
00012
00013  This program is free software: you can redistribute it and/or modify
00014  it under the terms of the GNU General Public License as published by
00015  the Free Software Foundation, either version 3 of the License, or
00016  (at your option) any later version.
00017
00018  This program is distributed in the hope that it will be useful,
00019  but WITHOUT ANY WARRANTY; without even the implied warranty of
00020  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00021  GNU General Public License for more details.
00022
00023  You should have received a copy of the GNU General Public License
00024  along with this program. If not, see <https://www.gnu.org/licenses/>.
00025  */
00026
00027  #ifndef _VIRTUAL_H_
00028  #define _VIRTUAL_H_
00029
00030  #include "std/api.h"
00031
00032  struct vm;
00033  struct pointer;
00034
00035  typedef struct PRIVATE_API(virtual_methods) {
00036      void (*init)(struct vm**, u64 size);
00037      void (*destroy)(struct vm**);
00038      u64 (*alloc)(pointer_ptr ptr);
00039      void (*free)(const_pointer_ptr ptr);
00040      pointer_ptr (*read)(u64 address);
00041      pointer_ptr (*read_type)(u64 address, u64 id);
00042      void (*enumerator_init)(void);
00043      void (*enumerator_destroy)(void);
00044      u64 (*enumerator_next)(void);
00045      #ifdef USE_MEMORY_DEBUG_INFO
00046          void (*dump)(void);
00047          void (*dump_ref)(void);
00048      #endif
00049  } virtual_methods;
00050
00051  /* definition */
00052  extern const virtual_methods PRIVATE_API(virtual_methods_definitions);
00053
00054  /* definition */
00055  #ifdef INLINE
00056  const virtual_methods* virtual = &PRIVATE_API(virtual_methods_definitions);
00057  #else
00058  /* definition */
00059  static const virtual_methods* virtual = &PRIVATE_API(virtual_methods_definitions);
00060  #endif
00061
00062  #endif /* _VIRTUAL_H_ */
```



# Index

[file\\_handler](#), [7](#)

[known\\_types](#), [7](#)

[list\\_handler](#), [8](#)

[memory\\_info\\_data](#), [8](#)

[pointer](#), [9](#)

[src/std/api.h](#), [15](#)

[src/std/compile.h](#), [16](#)

[src/std/data.h](#), [16](#)

[src/std/errors.h](#), [17](#)

[src/std/headers.h](#), [18](#)

[src/std/macros.h](#), [18](#)

[src/std/time.h](#), [19](#)

[src/std/version.h](#), [20](#)

[src/sys/info/info\\_v1.h](#), [20](#)

[src/sys/list/list\\_v1.h](#), [20](#), [21](#)

[src/sys/memory/memory\\_v1.h](#), [22](#)

[src/sys/options/options\\_v1.h](#), [23](#)

[src/sys/os/os\\_v1.h](#), [23](#), [24](#)

[src/vm/v1/pointer/pointer\\_v1.h](#), [25](#), [26](#)

[src/vm/v1/types/data/data\\_v1.h](#), [26](#), [27](#)

[src/vm/v1/types/file/file\\_v1.h](#), [28](#), [29](#)

[src/vm/v1/types/object/object\\_v1.h](#), [30](#), [31](#)

[src/vm/v1/types/stack/stack\\_v1.h](#), [31](#)

[src/vm/v1/types/string/string\\_v1.h](#), [32](#), [33](#)

[src/vm/v1/types/string\\_pointer/string\\_pointer\\_v1.h](#), [34](#)

[src/vm/v1/types/user/user\\_v1.h](#), [35](#), [36](#)

[src/vm/v1/virtual/virtual\\_v1.h](#), [36](#)

[stack\\_element](#), [9](#)

[stack\\_handler](#), [10](#)

[string\\_reference](#), [10](#)

[Table of contents](#), [1](#)

[type\\_methods\\_definitions](#), [10](#)

[virtual\\_pointer](#), [11](#)

[virtual\\_pointer\\_enumerator](#), [11](#)

[vm](#), [12](#)

[vm\\_pointer](#), [13](#)

[vm\\_state](#), [14](#)