

**ZebOS-XP MLD SMI Reference**  
IP Infusion Inc.

Generated by Doxygen 1.6.1

Wed Dec 16 12:33:59 2015



# Contents

<b>1</b>	<b>File Index</b>	<b>1</b>
1.1	File List . . . . .	1
<b>2</b>	<b>File Documentation</b>	<b>3</b>
2.1	smi_mld.h File Reference . . . . .	3
2.1.1	Detailed Description . . . . .	7
2.1.2	Function Documentation . . . . .	8
2.1.2.1	smi_debug_ipv6_mld_sdkapi . . . . .	8
2.1.2.2	smi_debug_no_ipv6_mld_sdkapi . . . . .	8
2.1.2.3	smi_mrib6_mld_clear_sdkapi . . . . .	9
2.1.2.4	smi_mrib6_mld_if_access_list_set_sdkapi . . . . .	9
2.1.2.5	smi_mrib6_mld_if_access_list_unset_sdkapi . . . . .	10
2.1.2.6	smi_mrib6_mld_if_immediate_leave_set_sdkapi . . . . .	10
2.1.2.7	smi_mrib6_mld_if_immediate_leave_unset_sdkapi . . . . .	11
2.1.2.8	smi_mrib6_mld_if_limit_set_sdkapi . . . . .	11
2.1.2.9	smi_mrib6_mld_if_limit_unset_sdkapi . . . . .	12
2.1.2.10	smi_mrib6_mld_if_lmqc_set_sdkapi . . . . .	12
2.1.2.11	smi_mrib6_mld_if_lmqc_unset_sdkapi . . . . .	13
2.1.2.12	smi_mrib6_mld_if_lmqi_set_sdkapi . . . . .	13
2.1.2.13	smi_mrib6_mld_if_lmqi_unset_sdkapi . . . . .	14
2.1.2.14	smi_mrib6_mld_if_mroute_pxy_set_sdkapi . . . . .	14
2.1.2.15	smi_mrib6_mld_if_mroute_pxy_unset_sdkapi . . . . .	15
2.1.2.16	smi_mrib6_mld_if_pxy_service_set_sdkapi . . . . .	15
2.1.2.17	smi_mrib6_mld_if_pxy_service_unset_sdkapi . . . . .	16
2.1.2.18	smi_mrib6_mld_if_querier_timeout_set_sdkapi . . . . .	16
2.1.2.19	smi_mrib6_mld_if_querier_timeout_unset_sdkapi . . . . .	17

2.1.2.20	<a href="#">smi_mrib6_mld_if_query_interval_set_sdkapi . . .</a>	18
2.1.2.21	<a href="#">smi_mrib6_mld_if_query_interval_unset_sdkapi . .</a>	18
2.1.2.22	<a href="#">smi_mrib6_mld_if_query_response_interval_set_sdkapi . . . . .</a>	19
2.1.2.23	<a href="#">smi_mrib6_mld_if_query_response_interval_unset_sdkapi . . . . .</a>	19
2.1.2.24	<a href="#">smi_mrib6_mld_if_robustness_var_set_sdkapi . . .</a>	20
2.1.2.25	<a href="#">smi_mrib6_mld_if_robustness_var_unset_sdkapi .</a>	20
2.1.2.26	<a href="#">smi_mrib6_mld_if_set_sdkapi . . . . .</a>	21
2.1.2.27	<a href="#">smi_mrib6_mld_if_startup_query_count_set_sdkapi</a>	21
2.1.2.28	<a href="#">smi_mrib6_mld_if_startup_query_count_unset_sdkapi . . . . .</a>	22
2.1.2.29	<a href="#">smi_mrib6_mld_if_startup_query_interval_set_sdkapi</a>	23
2.1.2.30	<a href="#">smi_mrib6_mld_if_startup_query_interval_unset_sdkapi . . . . .</a>	23
2.1.2.31	<a href="#">smi_mrib6_mld_if_static_group_source_set_sdkapi</a>	24
2.1.2.32	<a href="#">smi_mrib6_mld_if_static_group_source_unset_sdkapi</a>	24
2.1.2.33	<a href="#">smi_mrib6_mld_if_unset_sdkapi . . . . .</a>	25
2.1.2.34	<a href="#">smi_mrib6_mld_if_version_set_sdkapi . . . . .</a>	25
2.1.2.35	<a href="#">smi_mrib6_mld_if_version_unset_sdkapi . . . . .</a>	26
2.1.2.36	<a href="#">smi_mrib6_mld_limit_set_sdkapi . . . . .</a>	26
2.1.2.37	<a href="#">smi_mrib6_mld_limit_unset_sdkapi . . . . .</a>	27
2.1.2.38	<a href="#">smi_mrib6_mld_ssm_map_enable_set_sdkapi . . .</a>	27
2.1.2.39	<a href="#">smi_mrib6_mld_ssm_map_enable_unset_sdkapi . .</a>	28
2.1.2.40	<a href="#">smi_mrib6_mld_ssm_map_static_set_sdkapi . . . .</a>	28
2.1.2.41	<a href="#">smi_mrib6_mld_ssm_map_static_unset_sdkapi . .</a>	28
2.2	<a href="#">smi_mld_snooping.h File Reference . . . . .</a>	30
2.2.1	<a href="#">Detailed Description . . . . .</a>	32
2.2.2	<a href="#">Function Documentation . . . . .</a>	32
2.2.2.1	<a href="#">smi_debug_mld_snoop_sdkapi . . . . .</a>	32
2.2.2.2	<a href="#">smi_debug_no_mld_snoop_sdkapi . . . . .</a>	33
2.2.2.3	<a href="#">smi_mld_snoop_api_clear_all . . . . .</a>	33
2.2.2.4	<a href="#">smi_mld_snoop_api_clear_group . . . . .</a>	33
2.2.2.5	<a href="#">smi_mld_snoop_api_clear_group_if . . . . .</a>	34
2.2.2.6	<a href="#">smi_mld_snoop_api_clear_if . . . . .</a>	34

2.2.2.7	smi_mld_snoop_api_if_fast_leave_set . . . . .	35
2.2.2.8	smi_mld_snoop_api_if_fast_leave_unset . . . . .	35
2.2.2.9	smi_mld_snoop_api_if_mrouter_set . . . . .	36
2.2.2.10	smi_mld_snoop_api_if_mrouter_unset . . . . .	36
2.2.2.11	smi_mld_snoop_api_if_querier_set . . . . .	37
2.2.2.12	smi_mld_snoop_api_if_querier_unset . . . . .	37
2.2.2.13	smi_mld_snoop_api_if_report_suppress_set . . . . .	38
2.2.2.14	smi_mld_snoop_api_if_report_suppress_unset . . . . .	38
2.2.2.15	smi_mld_snoop_api_if_snooping_set . . . . .	39
2.2.2.16	smi_mld_snoop_api_if_snooping_unset . . . . .	39
2.2.2.17	smi_mld_snoop_api_set_snooping . . . . .	40
2.2.2.18	smi_mld_snoop_api_unset_snooping . . . . .	40



# Chapter 1

## File Index

### 1.1 File List

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

<a href="#">smi_mld.h</a> (Provides API for managing Multicast Listener Discovery protocol implementation in ZebOS ) . . . . .	3
<a href="#">smi_mld_snooping.h</a> (Describes the command API functions for managing the MLD Snooping ) . . . . .	30





# Chapter 2

## File Documentation

### 2.1 smi\_mld.h File Reference

Provides API for managing Multicast Listener Discovery protocol implementation in ZebOS. `#include "smi_client.h"`

`#include "smi_mld_msg.h"`

#### Functions

- int [smi\\_mrib6\\_mld\\_limit\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name, u\_int32\_t limit, char \*except\_alist)  
*This function sets the limit for group-record states across all interfaces in the specified MLD instance. Exception list is specified to exclude certain groups on which limit value will not be applied.*
- int [smi\\_mrib6\\_mld\\_limit\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name)  
*This function unsets the limit for group-record states across all interfaces in the specified MLD Instance.*
- int [smi\\_mrib6\\_mld\\_ssm\\_map\\_enable\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name)  
*This function enables Source-Specific-Mapping (SSM) at instance level.*
- int [smi\\_mrib6\\_mld\\_ssm\\_map\\_enable\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name)  
*This function disables Source-Specific-Mapping (SSM) at instance level.*
- int [smi\\_mrib6\\_mld\\_ssm\\_map\\_static\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name, char \*alist, char \*msrc\_arg)  
*This function sets Source-Specific-Mapping (SSM) static definition at instance level. The specified source address will be used to produce (G,S)SSM mapping for*

*the group address defined as the supplied access-list reference string. This function may be invoked multiple times to define multiple SSM mappings.*

- int [smi\\_mrib6\\_mld\\_ssm\\_map\\_static\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name, char \*alist, char \*msrc\_arg)

*This function unsets Source-Specific-Mapping (SSM) identified by the supplied access-list reference string and source-address string.*

- int [smi\\_mrib6\\_mld\\_if\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function enables MLD protocol feature on the specified interface.*

- int [smi\\_mrib6\\_mld\\_if\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function disables MLD protocol feature on the specified interface.*

- int [smi\\_mrib6\\_mld\\_if\\_access\\_list\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, char \*alist)

*This function sets the MLD access-list for the specified interface.*

- int [smi\\_mrib6\\_mld\\_if\\_access\\_list\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function unsets the MLD access-list.*

- int [smi\\_mrib6\\_mld\\_if\\_immediate\\_leave\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, char \*alist)

*This function minimizes the leave latency of MLD membership. Interface configured for MLD layer-3 protocols will be applied. alist contains group name for which immediate leave feature is enabled.*

- int [smi\\_mrib6\\_mld\\_if\\_immediate\\_leave\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function disables the minimization of leave latency feature of MLD membership.*

- int [smi\\_mrib6\\_mld\\_if\\_lmqc\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int32\_t lmqc)

*This function sets Last-Member-Query-Count(LMQC) value.*

- int [smi\\_mrib6\\_mld\\_if\\_lmqc\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function unsets Last-Member-Query-Count(LMQC) value.*

- int [smi\\_mrib6\\_mld\\_if\\_lmqi\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int32\_t lmqi)

*This function sets Last-Member-Query-Interval(LMQI) value. This API sets the frequency at which routers will send MLD group specific host query messages.*

- int [smi\\_mrib6\\_mld\\_if\\_lmqi\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function unsets Last-Member-Query-Interval(LMQI) value.*
- int [smi\\_mrib6\\_mld\\_if\\_limit\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int32\_t limit, char \*except\_alist)  
*This function sets the limit for group-record states on the specified interface and MLD Instance. An exception access list can be specified to exclude certain groups from being subject to this limit value.*
- int [smi\\_mrib6\\_mld\\_if\\_limit\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function unsets the limit and exception access-list for group-record states on the specified interface and MLD instance.*
- int [smi\\_mrib6\\_mld\\_if\\_mroute\\_pxy\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, char \*mrtr\_pxy\_ifname)  
*This function sets Proxy-serve interface association for the specified interface. It makes upstream Proxy service interface functional.*
- int [smi\\_mrib6\\_mld\\_if\\_mroute\\_pxy\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function unsets Proxy-serve interface association on the specified downstream interface.*
- int [smi\\_mrib6\\_mld\\_if\\_pxy\\_service\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function sets the specified interface for MLD proxy-service.*
- int [smi\\_mrib6\\_mld\\_if\\_pxy\\_service\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function unsets MLD proxy-service on the specified interface.*
- int [smi\\_mrib6\\_mld\\_if\\_querier\\_timeout\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int16\_t other\_querier\_interval)  
*This function sets the MLD other-querier timeout. It configures the timeout period before the router can become querier for the interface after the previous querier has stopped querying .*
- int [smi\\_mrib6\\_mld\\_if\\_querier\\_timeout\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function unsets the MLD other-querior timeout.*
- int [smi\\_mrib6\\_mld\\_if\\_query\\_interval\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int32\_t query\_interval)  
*This function sets the MLD query interval value. It configures the frequency at which MLD host query message is sent over the specified interface.*

- int [smi\\_mrib6\\_mld\\_if\\_query\\_interval\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function unsets the MLD query interval value.*

- int [smi\\_mrib6\\_mld\\_if\\_startup\\_query\\_interval\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int32\_t startup\_query\_interval)

*This function sets the MLD startup query interval value. It configures the frequency at which MLD host query message is sent over the specified interface on startup.*

- int [smi\\_mrib6\\_mld\\_if\\_startup\\_query\\_interval\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function unsets the MLD startup query interval value.*

- int [smi\\_mrib6\\_mld\\_if\\_startup\\_query\\_count\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int32\_t startup\_query\_count)

*This function sets the MLD startup query count value. It configures the number of times host query message is sent over the specified interface on startup.*

- int [smi\\_mrib6\\_mld\\_if\\_startup\\_query\\_count\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function unsets the MLD startup query count value.*

- int [smi\\_mrib6\\_mld\\_if\\_query\\_response\\_interval\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int32\_t response\_interval)

*This function sets the MLD query-max-response time advertised in MLD queries.*

- int [smi\\_mrib6\\_mld\\_if\\_query\\_response\\_interval\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function unsets the MLD query-response interval value.*

- int [smi\\_mrib6\\_mld\\_if\\_robustness\\_var\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int32\_t robustness\_var)

*This function sets the robustness variable on the specified interface.*

- int [smi\\_mrib6\\_mld\\_if\\_robustness\\_var\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function unsets the robustness variable on the specified interface.*

- int [smi\\_mrib6\\_mld\\_if\\_version\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, u\_int16\_t version)

*This function sets the MLD version.*

- int [smi\\_mrib6\\_mld\\_if\\_version\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname)

*This function unsets the MLD version.*

- int [smi\\_mrrib6\\_mld\\_if\\_static\\_group\\_source\\_set\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, char \*mcg\_addr, char \*src\_addr, bool\_t is\_ssm\_mapped)

*This function sets the MLD static group.*

- int [smi\\_mrrib6\\_mld\\_if\\_static\\_group\\_source\\_unset\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*ifname, char \*mcg\_addr, char \*src\_addr, bool\_t is\_ssm\_mapped)

*This function unsets the MLD static group.*

- int [smi\\_mrrib6\\_mld\\_clear\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name)

*This function clears MLD state information.*

- int [smi\\_debug\\_ipv6\\_mld\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name, int debug)

*Use this function to enable debugging of all MLD, or a specific component of MLD. This command applies to interfaces configured for MLD Layer-3 multicast protocols.*

- int [smi\\_debug\\_no\\_ipv6\\_mld\\_sdkapi](#) (struct smiclient\_globals \*azg, char \*vrf\_name, int debug)

*Use this function to disable all MLD debugging or debugging of a specific component of MLD.*

- int [smi\\_show\\_ipv6\\_mld\\_interface](#) (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, char \*vrf\_name, char \*ifname, struct list \*ipv6MldInfoList, int(\*funpointer)(struct list \*ipv6MldInfoList))
- int [smi\\_show\\_ipv6\\_mld\\_ssm\\_map](#) (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, char \*vrf\_name, u\_int16\_t mli\_cflags\_enable, u\_int16\_t mli\_cflags\_static)
- int [smi\\_show\\_ipv6\\_mld\\_group](#) (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, char \*vrf\_name, bool\_t detail\_flag, struct list \*ipv6MldInfoList, int(\*funpointer)(struct list \*ipv6MldInfoList))
- int [smi\\_show\\_ipv6\\_mld\\_groups\\_by\\_ifname](#) (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, char \*vrf\_name, char \*ifname, bool\_t detail\_flag, struct list \*ipv6MldInfoList, int(\*funpointer)(struct list \*ipv6MldInfoList))
- int [smi\\_show\\_ipv6\\_mld\\_groups\\_by\\_addr](#) (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, char \*vrf\_name, bool\_t detail\_flag, char \*addr, struct list \*ipv6MldInfoList, int(\*funpointer)(struct list \*ipv6MldInfoList))

### 2.1.1 Detailed Description

Provides API for managing Multicast Listener Discovery protocol implementation in ZebOS. IPv6 Multicast Listener Discovery (MLD) is used by IPv6 devices to discover multicast listeners on directly attached links.

## 2.1.2 Function Documentation

### 2.1.2.1 `int smi_debug_ipv6_mld_sdkapi (struct smiclient_globals * azg, char * vrf_name, int debug)`

Use this function to enable debugging of all MLD, or a specific component of MLD. This command applies to interfaces configured for MLD Layer-3 multicast protocols. `smi_debug_ipv6_mld_sdkapi`

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF NAME. Pass Null in case of default VRF
- ← *debug* Pass debug flag as following:
  - SMI\_MLD\_DECODE\_FLAG - Debug MLD decoding
  - SMI\_MLD\_ENCODE\_FLAG - Debug MLD encoding
  - SMI\_MLD\_EVENTS\_FLAG - Debug MLD events
  - SMI\_MLD\_FSM\_FLAG - Debug MLD Finite State Machine (FSM)
  - SMI\_MLD\_TIB\_FLAG - Debug MLD Tree Information Base (TIB)
  - SMI\_MLD\_ALL\_FLAG - Debug all MLD

#### Returns:

- 0 on success, otherwise one of the following error codes `IGMP_ERR_NO_CONTEXT_INFO`  
`SMI_ERROR`

### 2.1.2.2 `int smi_debug_no_ipv6_mld_sdkapi (struct smiclient_globals * azg, char * vrf_name, int debug)`

Use this function to disable all MLD debugging or debugging of a specific component of MLD. `smi_debug_no_ipv6_mld_sdkapi`

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF NAME. Pass Null in case of default VRF
- ← *debug* Pass debug flag as following:
  - SMI\_MLD\_DECODE\_FLAG - Debug MLD decoding
  - SMI\_MLD\_ENCODE\_FLAG - Debug MLD encoding
  - SMI\_MLD\_EVENTS\_FLAG - Debug MLD events
  - SMI\_MLD\_FSM\_FLAG - Debug MLD Finite State Machine (FSM)
  - SMI\_MLD\_TIB\_FLAG - Debug MLD Tree Information Base (TIB)
  - SMI\_MLD\_ALL\_FLAG - Debug all MLD

**Returns:**

0 on success, otherwise one of the following error codes IGMP\_ERR\_NO\_CONTEXT\_INFO  
SMI\_ERROR

**2.1.2.3 int smi\_mrib6\_mld\_clear\_sdkapi (struct smiclient\_globals \* azg, char \* vrf\_name)**

This function clears MLD state information. smi\_mrib6\_mld\_clear\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF name
- ← *ifname* Interface name string
- ← *pgrp* Pointer to the specified MLD group address to clear
- ← *psrc* Pointer to the specified source address to clear

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_DOOM

**2.1.2.4 int smi\_mrib6\_mld\_if\_access\_list\_set\_sdkapi (struct smiclient\_globals \* azg, char \* ifname, char \* alist)**

This function sets the MLD access-list for the specified interface. smi\_mrib6\_mld\_if\_access\_list\_set\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *alist* Pointer to access-list name

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF

```

MLD_ERR_NO_SUCH_IFF
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_DOOM
MLD_ERR_GENERIC
MLD_ERR_OOM

```

#### 2.1.2.5 int smi\_mrib6\_mld\_if\_access\_list\_unset\_sdkapi (struct smiclient\_globals \*azg, char \*ifname)

This function unsets the MLD access-list. smi\_mrib6\_mld\_if\_access\_list\_unset\_sdkapi

##### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

##### Returns:

```

MLD_ERROR_NONE on success, otherwise one of the following error codes
MLD_ERR_INVALID_VALUE
MLD_ERR_L2_PHYSICAL_IF
MLD_ERR_NO_SUCH_IFF
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_DOOM
MLD_ERR_GENERIC
MLD_ERR_OOM

```

#### 2.1.2.6 int smi\_mrib6\_mld\_if\_immediate\_leave\_set\_sdkapi (struct smiclient\_globals \*azg, char \*ifname, char \*alist)

This function minimizes the leave latency of MLD membership. Interface configured for MLD layer-3 protocols will be applied. alist contains group name for which immediate leave feature is enabled. smi\_mrib6\_mld\_if\_immediate\_leave\_set\_sdkapi

##### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *alist* Pointer to access-list name

##### Returns:

```

MLD_ERROR_NONE on success, otherwise one of the following error codes

```



```

MLD_ERR_INVALID_VALUE
MLD_ERR_L2_PHYSICAL_IF
MLD_ERR_NO_SUCH_IFF
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_GENERIC
MLD_ERR_OOM

```

### 2.1.2.7 int smi\_mrib6\_mld\_if\_immediate\_leave\_unset\_sdkapi (struct smiclient\_globals \* azg, char \* ifname)

This function disables the minimization of leave latency feature of MLD membership.  
smi\_mrib6\_mld\_if\_immediate\_leave\_unset\_sdkapi

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

#### Returns:

```

MLD_ERROR_NONE on success, otherwise one of the following error codes
MLD_ERR_INVALID_VALUE
MLD_ERR_L2_PHYSICAL_IF
MLD_ERR_NO_SUCH_IFF
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_GENERIC
MLD_ERR_OOM

```

### 2.1.2.8 int smi\_mrib6\_mld\_if\_limit\_set\_sdkapi (struct smiclient\_globals \* azg, char \* ifname, u\_int32\_t limit, char \* except\_alist)

This function sets the limit for group-record states on the specified interface and MLD Instance. An exception access list can be specified to exclude certain groups from being subject to this limit value. smi\_mrib6\_mld\_if\_limit\_set\_sdkapi

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *limit* Limit value numeric <1-2097152>
- ← *except\_alist* Pointer to an access-list name

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

**2.1.2.9 int smi\_mrib6\_mld\_if\_limit\_unset\_sdkapi (struct smiclient\_globals \*  
 azg, char \* ifname)**

This function unsets the limit and exception access-list for group-record states on the specified interface and MLD instance. smi\_mrib6\_mld\_if\_limit\_unset\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

**2.1.2.10 int smi\_mrib6\_mld\_if\_lmqc\_set\_sdkapi (struct smiclient\_globals \* azg,  
 char \* ifname, u\_int32\_t lmqc)**

This function sets Last-Member-Query-Count(LMQC) value. smi\_mrib6\_mld\_if\_lmqc\_set\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *lmqc* The LMQC value numeric <2-7>

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_GENERIC  
MLD\_ERR\_DOOM

**2.1.2.11 int smi\_mrib6\_mld\_if\_lmqc\_unset\_sdkapi (struct smiclient\_globals \*  
azg, char \* ifname)**

This function unsets Last-Member-Query-Count(LMQC) value. smi\_mrib6\_mld\_if\_lmqc\_unset\_sdkapi

**Parameters:**

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_GENERIC  
MLD\_ERR\_DOOM

**2.1.2.12 int smi\_mrib6\_mld\_if\_lmqi\_set\_sdkapi (struct smiclient\_globals \* azg,  
char \* ifname, u\_int32\_t lmqi)**

This function sets Last-Member-Query-Interval(LMQI) value. This API sets the frequency at which routers will send MLD group specific host query messages. smi\_mrib6\_mld\_if\_lmqi\_set\_sdkapi

**Parameters:**

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface name string

← *lmqi* The LMQI value numeric <1000-25500>

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

**2.1.2.13 int smi\_mrib6\_mld\_if\_lmqi\_unset\_sdkapi (struct smiclient\_globals \* azg, char \* ifname)**

This function unsets Last-Member-Query-Interval(LMQI) value. smi\_mrib6\_mld\_if\_lmqi\_unset\_sdkapi

**Parameters:**

← *azg* Pointer to the SMI client global structure  
 ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_DOOM

**2.1.2.14 int smi\_mrib6\_mld\_if\_mroute\_pty\_set\_sdkapi (struct smiclient\_globals \* azg, char \* ifname, char \* mrtr\_pty\_ifname)**

This function sets Proxy-service interface association for the specified interface. It makes upstream Proxy service interface functional. smi\_mrib6\_mld\_if\_mroute\_pty\_set\_sdkapi

**Parameters:**

← *azg* Pointer to the SMI client global structure

- ← *ifname* Interface name string
- ← *mrtr\_pxy\_ifname* Pointer to the interface name of the upstream proxy-service interface

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes

- MLD\_ERR\_INVALID\_VALUE
- MLD\_ERR\_L2\_PHYSICAL\_IF
- MLD\_ERR\_NO\_SUCH\_IFF
- MLD\_ERR\_MALFORMED\_ARG
- MLD\_ERR\_CFG\_FOR\_PROXY\_SERVICE
- MLD\_ERR\_NO\_SUCH\_GROUP\_REC
- MLD\_ERR\_NO\_SUCH\_SOURCE\_REC
- MLD\_ERR\_OOM
- MLD\_ERR\_GENERIC
- MLD\_ERR\_DOOM

**2.1.2.15 int smi\_mrib6\_mld\_if\_mroute\_pxy\_unset\_sdkapi (struct smiclient\_globals \* azg, char \* ifname)**

This function unsets Proxy-service interface association on the specified downstream interface. smi\_mrib6\_mld\_if\_mroute\_pxy\_unset\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes

- MLD\_ERR\_INVALID\_VALUE
- MLD\_ERR\_L2\_PHYSICAL\_IF
- MLD\_ERR\_NO\_SUCH\_IFF
- MLD\_ERR\_MALFORMED\_ARG
- MLD\_ERR\_CFG\_FOR\_PROXY\_SERVICE
- MLD\_ERR\_NO\_SUCH\_GROUP\_REC
- MLD\_ERR\_NO\_SUCH\_SOURCE\_REC
- MLD\_ERR\_OOM
- MLD\_ERR\_GENERIC
- MLD\_ERR\_DOOM

**2.1.2.16 int smi\_mrib6\_mld\_if\_pxy\_service\_set\_sdkapi (struct smiclient\_globals \* azg, char \* ifname)**

This function sets the specified interface for MLD proxy-service. smi\_mrib6\_mld\_if\_pxy\_service\_set\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_CFG\_WITH\_MROUTE\_PROXY  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

#### 2.1.2.17 int smi\_mrib6\_mld\_if\_pxy\_service\_unset\_sdkapi (struct smiclient\_globals \* azg, char \* ifname)

This function unsets MLD proxy-service on the specified interface. smi\_mrib6\_mld\_if\_pxy\_service\_unset\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_CFG\_WITH\_MROUTE\_PROXY  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

#### 2.1.2.18 int smi\_mrib6\_mld\_if\_querier\_timeout\_set\_sdkapi (struct smiclient\_globals \* azg, char \* ifname, u\_int16\_t other\_querier\_interval)

This function sets the MLD other-querier timeout. It configures the timeout period before the router can become querier for the interface after the previous querier has

stopped querying . smi\_mrib6\_mld\_if\_querier\_timeout\_set\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *other\_querier\_interval* MLD other-querier timeout value numeric <60-300>

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_GENERIC  
MLD\_ERR\_DOOM

**2.1.2.19 int smi\_mrib6\_mld\_if\_querier\_timeout\_unset\_sdkapi (struct smiclient\_globals \* azg, char \* ifname)**

This function unsets the MLD other-querier timeout. smi\_mrib6\_mld\_if\_querier\_timeout\_unset\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_GENERIC  
MLD\_ERR\_DOOM

### 2.1.2.20 `int smi_mrib6_mld_if_query_interval_set_sdkapi (struct smiclient_globals * azg, char * ifname, u_int32_t query_interval)`

This function sets the MLD query interval value. It configures the frequency at which MLD host query message is sent over the specified interface. `smi_mrib6_mld_if_query_interval_set_sdkapi`

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF name
- ← *ifname* Interface name string
- ← *query\_interval* The MLD query interval value numeric <1-18000>

#### Returns:

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_QI\_LE\_QRI  
 MLD\_ERR\_MALFORMED\_ARG

MLD\_ERR\_NO\_SUCH\_GROUP\_REC

MLD\_ERR\_NO\_SUCH\_SOURCE\_REC

MLD\_ERR\_OOM

MLD\_ERR\_GENERIC

MLD\_ERR\_DOOM

### 2.1.2.21 `int smi_mrib6_mld_if_query_interval_unset_sdkapi (struct smiclient_globals * azg, char * ifname)`

This function unsets the MLD query interval value. `smi_mrib6_mld_if_query_interval_unset_sdkapi`

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

#### Returns:

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_QI\_LE\_QRI



MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_GENERIC  
MLD\_ERR\_DOOM

#### 2.1.2.22 int smi\_mrib6\_mld\_if\_query\_response\_interval\_set\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*, u\_int32\_t *response\_interval*)

This function sets the MLD query-max-response time advertised in MLD queries. smi\_mrib6\_mld\_if\_query\_response\_interval\_set\_sdkapi

##### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *response\_interval* MLD query-response interval value numeric <1-240>

##### Returns:

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_QRI\_GT\_QI  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_GENERIC  
MLD\_ERR\_DOOM

#### 2.1.2.23 int smi\_mrib6\_mld\_if\_query\_response\_interval\_unset\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*)

This function unsets the MLD query-response interval value. smi\_mrib6\_mld\_if\_query\_response\_interval\_unset\_sdkapi

##### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

##### Returns:

MLD\_ERROR\_NONE on success, otherwise one of the following error codes

```

MLD_ERR_INVALID_VALUE
MLD_ERR_L2_PHYSICAL_IF
MLD_ERR_NO_SUCH_IFF
MLD_ERR_QRI_GT_QI
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_OOM
MLD_ERR_GENERIC
MLD_ERR_DOOM

```

#### 2.1.2.24 `int smi_mrib6_mld_if_robustness_var_set_sdkapi (struct smiclient_globals * azg, char * ifname, u_int32_t robustness_var)`

This function sets the robustness variable on the specified interface. `smi_mrib6_mld_if_robustness_var_set_sdkapi`

##### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *robustness\_var* Robustness variable value numeric <2-7>

##### Returns:

```

MLD_ERROR_NONE on success, otherwise one of the following error codes
MLD_ERR_INVALID_VALUE
MLD_ERR_L2_PHYSICAL_IF
MLD_ERR_NO_SUCH_IFF
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_OOM
MLD_ERR_GENERIC
MLD_ERR_DOOM

```

#### 2.1.2.25 `int smi_mrib6_mld_if_robustness_var_unset_sdkapi (struct smiclient_globals * azg, char * ifname)`

This function unsets the robustness variable on the specified interface. `smi_mrib6_mld_if_robustness_var_unset_sdkapi`

##### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_GENERIC  
MLD\_ERR\_DOOM

**2.1.2.26 int smi\_mrib6\_mld\_if\_set\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*)**

This function enables MLD protocol feature on the specified interface. smi\_mrib6\_mld\_if\_set\_sdkapi

**Parameters:**

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM  
MLD\_ERR\_GENERIC  
MLD\_ERR\_DOOM

**2.1.2.27 int smi\_mrib6\_mld\_if\_startup\_query\_count\_set\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*, u\_int32\_t *startup\_query\_count*)**

This function sets the MLD startup query count value. It configures the number of times host query message is sent over the specified interface on startup. smi\_mrib6\_mld\_if\_startup\_query\_count\_set\_sdkapi

**Parameters:**

← *azg* Pointer to the SMI client global structure  
← *vrf\_name* VRF name

- ← *ifname* Interface name string
- ← *startup\_query\_interval* The MLD query interval value numeric <1-18000>

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_QI\_LE\_QRI  
 MLD\_ERR\_MALFORMED\_ARG

MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

### 2.1.2.28 int smi\_mrib6\_mld\_if\_startup\_query\_count\_unset\_sdkapi (struct smiclient\_globals \* azg, char \* ifname)

This function unsets the MLD startup query count value. smi\_mrib6\_mld\_if\_startup\_query\_count\_unset\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_QI\_LE\_QRI  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

### 2.1.2.29 int smi\_mrib6\_mld\_if\_startup\_query\_interval\_set\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*, u\_int32\_t *startup\_query\_interval*)

This function sets the MLD startup query interval value. It configures the frequency at which MLD host query message is sent over the specified interface on startup. smi\_mrib6\_mld\_if\_startup\_query\_interval\_set\_sdkapi

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF name
- ← *ifname* Interface name string
- ← *startup\_query\_interval* The MLD query interval value numeric <1-18000>

#### Returns:

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_QI\_LE\_QRI  
 MLD\_ERR\_MALFORMED\_ARG

MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

### 2.1.2.30 int smi\_mrib6\_mld\_if\_startup\_query\_interval\_unset\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*)

This function unsets the MLD startup query interval value. smi\_mrib6\_mld\_if\_startup\_query\_interval\_unset\_sdkapi

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

#### Returns:

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF

```

MLD_ERR_QI_LE_QRI
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_OOM
MLD_ERR_GENERIC
MLD_ERR_DOOM

```

**2.1.2.31 int smi\_mrib6\_mld\_if\_static\_group\_source\_set\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*, char \* *mcg\_addr*, char \* *src\_addr*, bool\_t *is\_ssm\_mapped*)**

This function sets the MLD static group. smi\_mrib6\_mld\_if\_static\_group\_source\_set\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *pgrp* Pointer to specific MLD group address
- ← *psrc* Pointer to specific MLD source address
- ← *is\_ssm\_mapped* Whether the SSM-mapped entry is mapped

**Returns:**

```

MLD_ERROR_NONE on success, otherwise one of the following error codes
MLD_ERR_INVALID_VALUE
MLD_ERR_L2_PHYSICAL_IF
MLD_ERR_NO_SUCH_IFF
MLD_ERR_L3_NON_VLAN_IF
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_OOM

```

**2.1.2.32 int smi\_mrib6\_mld\_if\_static\_group\_source\_unset\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*, char \* *mcg\_addr*, char \* *src\_addr*, bool\_t *is\_ssm\_mapped*)**

This function unsets the MLD static group. smi\_mrib6\_mld\_if\_static\_group\_source\_unset\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string
- ← *pgrp* Pointer to specific MLD group address

- ← *psrc* Pointer to specific MLD source address
- ← *is\_ssm\_mapped* Whether the SSM-mapped entry is mapped

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes

- MLD\_ERR\_INVALID\_VALUE
- MLD\_ERR\_L2\_PHYSICAL\_IF
- MLD\_ERR\_NO\_SUCH\_IFF
- MLD\_ERR\_L3\_NON\_VLAN\_IF
- MLD\_ERR\_MALFORMED\_ARG
- MLD\_ERR\_NO\_SUCH\_GROUP\_REC
- MLD\_ERR\_OOM

### 2.1.2.33 int smi\_mrib6\_mld\_if\_unset\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*)

This function disables MLD protocol feature on the specified interface. smi\_mrib6\_mld\_if\_unset\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes

- MLD\_ERR\_INVALID\_VALUE
- MLD\_ERR\_L2\_PHYSICAL\_IF
- MLD\_ERR\_NO\_SUCH\_IFF
- MLD\_ERR\_MALFORMED\_ARG
- MLD\_ERR\_NO\_SUCH\_GROUP\_REC
- MLD\_ERR\_NO\_SUCH\_SOURCE\_REC
- MLD\_ERR\_OOM
- MLD\_ERR\_GENERIC
- MLD\_ERR\_DOOM

### 2.1.2.34 int smi\_mrib6\_mld\_if\_version\_set\_sdkapi (struct smiclient\_globals \* *azg*, char \* *ifname*, u\_int16\_t *version*)

This function sets the MLD version. smi\_mrib6\_mld\_if\_version\_set\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface name string

← *version* MLD version no numeric <1-2>

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

**2.1.2.35 int smi\_mrib6\_mld\_if\_version\_unset\_sdkapi (struct smiclient\_globals \* azg, char \* ifname)**

This function unsets the MLD version. smi\_mrib6\_mld\_if\_version\_unset\_sdkapi

**Parameters:**

← *azg* Pointer to the SMI client global structure  
 ← *ifname* Interface name string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM  
 MLD\_ERR\_GENERIC  
 MLD\_ERR\_DOOM

**2.1.2.36 int smi\_mrib6\_mld\_limit\_set\_sdkapi (struct smiclient\_globals \* azg, char \* vrf\_name, u\_int32\_t limit, char \* except\_alist)**

This function sets the limit for group-record states across all interfaces in the specified MLD instance. Exception list is specified to exclude certain groups on which limit value will not be applied. smi\_mrib6\_mld\_limit\_set\_sdkapi

**Parameters:**

← *azg* Pointer to the SMI client global structure



- ← *vrf\_name* VRF name
- ← *limit* Limit value numeric <1-2097152>
- ← *except\_alist* Pointer to an access-list name

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_OOM

**2.1.2.37 int smi\_mrib6\_mld\_limit\_unset\_sdkapi (struct smiclient\_globals \* azg, char \* vrf\_name)**

This function unsets the limit for group-record states across all interfaces in the specified MLD Instance. smi\_mrib6\_mld\_limit\_unset\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF name

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE

**2.1.2.38 int smi\_mrib6\_mld\_ssm\_map\_enable\_set\_sdkapi (struct smiclient\_globals \* azg, char \* vrf\_name)**

This function enables Source-Specific-Mapping (SSM) at instance level. smi\_mrib6\_mld\_ssm\_map\_enable\_set\_sdkapi

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF name

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE

### 2.1.2.39 `int smi_mrib6_mld_ssm_map_enable_unset_sdkapi (struct smiclient_globals * azg, char * vrf_name)`

This function disables Source-Specific-Mapping (SSM) at instance level. `smi_mrib6_mld_ssm_map_enable_unset_sdkapi`

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF name

#### Returns:

- MLD\_ERROR\_NONE on success, otherwise one of the following error codes
- MLD\_ERR\_INVALID\_VALUE

### 2.1.2.40 `int smi_mrib6_mld_ssm_map_static_set_sdkapi (struct smiclient_globals * azg, char * vrf_name, char * alist, char * msrc_arg)`

This function sets Source-Specific-Mapping (SSM) static definition at instance level. The specified source address will be used to produce (G,S)SSM mapping for the group address defined as the supplied access-list reference string. This function may be invoked multiple times to define multiple SSM mappings. `smi_mrib6_mld_ssm_map_static_set_sdkapi`

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF name
- ← *alist* Pointer to group address access-list name
- ← *msrc\_arg* Pointer to source address string

#### Returns:

- MLD\_ERROR\_NONE on success, otherwise one of the following error codes
- MLD\_ERR\_INVALID\_VALUE
- MLD\_ERR\_MALFORMED\_ARG
- MLD\_ERR\_OOM

### 2.1.2.41 `int smi_mrib6_mld_ssm_map_static_unset_sdkapi (struct smiclient_globals * azg, char * vrf_name, char * alist, char * msrc_arg)`

This function unsets Source-Specific-Mapping (SSM) identified by the supplied access-list reference string and source-address string. `smi_mrib6_mld_ssm_map_static_unset_sdkapi`

**Parameters:**

- ← *azg* Pointer to the SMI client global structure
- ← *vrf\_name* VRF name
- ← *alist* Pointer to group address access-list name
- ← *msrc\_arg* Pointer to source address string

**Returns:**

MLD\_ERROR\_NONE on success, otherwise one of the following error codes  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_VALUE

## 2.2 smi\_mld\_snooping.h File Reference

Describes the command API functions for managing the MLD Snooping. `#include "smi_client.h"`

`#include "smi_mld_snooping_msg.h"`

### Functions

- `s_int32_t smi_mld_snoop_api_set_snooping` (struct `smiclient_globals *azg`)  
*This function enables MLD Snooping on all interfaces of this instance not explicitly (individually) disabled for MLD Snooping. MLD Snooping is globally enabled by default.*
- `s_int32_t smi_mld_snoop_api_unset_snooping` (struct `smiclient_globals *azg`)  
*This function disables MLD Snooping on all interfaces of this instance not explicitly (individually) enabled for MLD Snooping.*
- `s_int32_t smi_mld_snoop_api_if_snooping_set` (struct `smiclient_globals *azg`, `char *ifname`)  
*This function explicitly enables MLD Snooping on the specified VLAN interface.*
- `s_int32_t smi_mld_snoop_api_if_snooping_unset` (struct `smiclient_globals *azg`, `char *ifname`)  
*This function explicitly disables MLD Snooping on the specified VLAN interface. It also unsets all configuration associated with MLD Snooping on the VLAN interface.*
- `s_int32_t smi_mld_snoop_api_if_fast_leave_set` (struct `smiclient_globals *azg`, `char *ifname`)  
*This function enables fast-leave processing on the specified VLAN interface. Fast-leave processing is analogous to immediate-leave processing.*
- `s_int32_t smi_mld_snoop_api_if_fast_leave_unset` (struct `smiclient_globals *azg`, `char *ifname`)  
*This function disables fast-leave processing on the specified VLAN interface.*
- `s_int32_t smi_mld_snoop_api_if_mrouter_set` (struct `smiclient_globals *azg`, `char *ifname`, `char *mrouter_ifname`)  
*This function statically identifies a particular VLAN constituent interface as a multi-cast router (mrouter) interface for MLD Snooping on the specified VLAN interface. This function may be invoked multiple times to configure multiple VLAN constituent interfaces as mrouter interfaces.*
- `s_int32_t smi_mld_snoop_api_if_mrouter_unset` (struct `smiclient_globals *azg`, `char *ifname`, `char *mrouter_ifname`)  
*This function unsets the static configuration of a VLAN constituent interface as an mrouter interface on the specified VLAN interface.*

- s\_int32\_t [smi\\_mld\\_snoop\\_api\\_if\\_querier\\_set](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function enables MLD Snooping Querier functionality on the specified VLAN interface.*
- s\_int32\_t [smi\\_mld\\_snoop\\_api\\_if\\_querier\\_unset](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function disables MLD Snooping Querier functionality on the specified VLAN interface.*
- s\_int32\_t [smi\\_mld\\_snoop\\_api\\_if\\_report\\_suppress\\_set](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function enables MLD Snooping report-suppression on all constituent interfaces of the specified VLAN interface.*
- s\_int32\_t [smi\\_mld\\_snoop\\_api\\_if\\_report\\_suppress\\_unset](#) (struct smiclient\_globals \*azg, char \*ifname)  
*This function disables MLD Snooping report-suppression on all constituent interfaces of the specified VLAN interface.*
- s\_int32\_t [smi\\_mld\\_snoop\\_api\\_clear\\_if](#) (struct smiclient\_globals \*azg, char \*ifname)  
*Delete the Interface MLD entries of All groups of given interface.*
- s\_int32\_t [smi\\_mld\\_snoop\\_api\\_clear\\_group\\_if](#) (struct smiclient\_globals \*azg, char \*ip\_addr, char \*ifname)  
*Delete the MLD group cache entries of Given group on given interface.*
- s\_int32\_t [smi\\_mld\\_snoop\\_api\\_clear\\_all](#) (struct smiclient\_globals \*azg)  
*Delete the MLD groups cache entries of All groups from all interfaces.*
- s\_int32\_t [smi\\_mld\\_snoop\\_api\\_clear\\_group](#) (struct smiclient\_globals \*azg, char \*ip\_addr)  
*Delete the MLD group cache entries of Given group from all interfaces.*
- int [smi\\_debug\\_mld\\_snoop\\_sdkapi](#) (struct smiclient\_globals \*azg, int debug)  
*Use this function to enable all MLD SNOOPING debugging.*
- int [smi\\_debug\\_no\\_mld\\_snoop\\_sdkapi](#) (struct smiclient\_globals \*azg, int debug)  
*Use this function to disable all MLD SNOOPING debugging.*
- int [smi\\_show\\_mld\\_snooping\\_interface](#) (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, char \*ifname, struct list \*mldSnoopInfoList, int(\*funpointer)(struct list \*mldSnoopInfoList))
- int [smi\\_show\\_mld\\_snooping\\_mrouter\\_ifname](#) (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, char \*ifname, struct list \*ipMldSnoopMrouterList, int(\*funpointer)(struct list \*ipMldSnoopMrouterList))

- **int smi\_show\_mld\_snooping\_statistics\_ifname** (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, char \*ifname, struct list \*ipMldSnoopStatList, int(\*funpointer)(struct list \*ipMldSnoopStatList))
- **int smi\_show\_mld\_snooping\_groups\_detail** (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, bool\_t detail\_flag, struct list \*ipMldSnoopGroupList, int(\*funpointer)(struct list \*ipMldSnoopGroupList))
- **int smi\_show\_mld\_snooping\_groups\_by\_ifname** (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, bool\_t detail\_flag, char \*ifname, struct list \*ipMldSnoopGroupList, int(\*funpointer)(struct list \*ipMldSnoopGroupList))
- **int smi\_show\_mld\_snooping\_groups\_by\_addr** (struct smiclient\_globals \*azg, u\_int32\_t vr\_id, bool\_t detail\_flag, char \*grpAddr, struct list \*ipMldSnoopGroupList, int(\*funpointer)(struct list \*ipMldSnoopGroupList))

### 2.2.1 Detailed Description

Describes the command API functions for managing the MLD Snooping. These APIs are used by various north bound management interfaces like CLI, SNMP and SMI

### 2.2.2 Function Documentation

#### 2.2.2.1 int smi\_debug\_mld\_snoop\_sdkapi (struct smiclient\_globals \* azg, int debug)

Use this function to enable all MLD SNOOPING debugging. smi\_debug\_mld\_snoop\_sdkapi

##### Parameters:

- ← **azg** Pointer to the SMI client global structure
- ← **debug** Pass debug flag as following:
  - SMI\_MLD\_SNOOP\_DECODE\_FLAG - Debug MLD decoding
  - SMI\_MLD\_SNOOP\_ENCODE\_FLAG - Debug MLD encoding
  - SMI\_MLD\_SNOOP\_EVENTS\_FLAG - Debug MLD events
  - SMI\_MLD\_SNOOP\_FSM\_FLAG - Debug MLD Finite State Machine (FSM)
  - SMI\_MLD\_SNOOP\_TIB\_FLAG - Debug MLD Tree Information Base (TIB)
  - SMI\_MLD\_SNOOP\_ALL\_FLAG - Debug all MLD

##### Returns:

- 0 on success, otherwise one of the following error codes MLD\_ERR\_NO\_CONTEXT\_INFO
- SMI\_ERROR

### 2.2.2.2 int smi\_debug\_no\_mld\_snoop\_sdkapi (struct smiclient\_globals \* *azg*, int *debug*)

Use this function to disable all MLD SNOOPING debugging. smi\_debug\_no\_mld\_snoop\_sdkapi

#### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *debug* Pass debug flag as following:
  - SMI\_MLD\_SNOOP\_DECODE\_FLAG - Debug MLD decoding
  - SMI\_MLD\_SNOOP\_ENCODE\_FLAG - Debug MLD encoding
  - SMI\_MLD\_SNOOP\_EVENTS\_FLAG - Debug MLD events
  - SMI\_MLD\_SNOOP\_FSM\_FLAG - Debug MLD Finite State Machine (FSM)
  - SMI\_MLD\_SNOOP\_TIB\_FLAG - Debug MLD Tree Information Base (TIB)
  - SMI\_MLD\_SNOOP\_ALL\_FLAG - Debug all MLD

#### Returns:

0 on success, otherwise one of the following error codes MLD\_ERR\_NO\_CONTEXT\_INFO  
SMI\_ERROR

### 2.2.2.3 s\_int32\_t smi\_mld\_snoop\_api\_clear\_all (struct smiclient\_globals \* *azg*)

Delete the MLD groups cache entries of All groups from all interfaces. smi\_mld\_snoop\_api\_clear\_all

#### Parameters:

- ← *azg* Pointer to the SMI client global structure

#### Returns:

Return Success (i.e Zero), otherwise one of the following errors  
MLD\_ERR\_NO\_CONTEXT\_INFO  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_OOM

### 2.2.2.4 s\_int32\_t smi\_mld\_snoop\_api\_clear\_group (struct smiclient\_globals \* *azg*, char \* *ip\_addr*)

Delete the MLD group cache entries of Given group from all interfaces. smi\_mld\_snoop\_api\_clear\_group

**Parameters:**

← *azg* Pointer to the SMI client global structure

← *ip\_addr* Multicast Group Address

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_NO\_CONTEXT\_INFO

MLD\_ERR\_INVALID\_VALUE

MLD\_ERR\_NO\_SUCH\_IFF

MLD\_ERR\_NO\_SUCH\_GROUP\_REC

MLD\_ERR\_OOM

### 2.2.2.5 `s_int32_t smi_mld_snoop_api_clear_group_if (struct smiclient_globals *azg, char *ip_addr, char *ifname)`

Delete the MLD group cache entries of Given group on given interface. `smi_mld_snoop_api_clear_group_if`

**Parameters:**

← *azg* Pointer to the SMI client global structure

← *ip\_addr* Multicast Group Address

← *ifname* Interface Name

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_NO\_CONTEXT\_INFO

MLD\_ERR\_INVALID\_VALUE

MLD\_ERR\_NO\_SUCH\_IFF

MLD\_ERR\_NO\_SUCH\_GROUP\_REC

MLD\_ERR\_OOM

### 2.2.2.6 `s_int32_t smi_mld_snoop_api_clear_if (struct smiclient_globals *azg, char *ifname)`

Delete the Interface MLD entries of All groups of given interface. `smi_mld_snoop_api_clear_if`

**Parameters:**

← *azg* Pointer to the SMI client global structure

← *ifname* Interface Name

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors



MLD\_ERR\_NO\_CONTEXT\_INFO  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_OOM

#### 2.2.2.7 s\_int32\_t smi\_mld\_snoop\_api\_if\_fast\_leave\_set (struct smiclient\_globals \* *azg*, char \* *ifname*)

This function enables fast-leave processing on the specified VLAN interface. Fast-leave processing is analogous to immediate-leave processing. smi\_mld\_snoop\_api\_if\_fast\_leave\_set

##### Parameters:

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface Name

##### Returns:

Return Success (i.e Zero), otherwise one of the following errors  
MLD\_ERR\_NONE  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_L3\_NON\_VLAN\_IF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM

#### 2.2.2.8 s\_int32\_t smi\_mld\_snoop\_api\_if\_fast\_leave\_unset (struct smiclient\_globals \* *azg*, char \* *ifname*)

This function disables fast-leave processing on the specified VLAN interface. smi\_mld\_snoop\_api\_if\_fast\_leave\_unset

##### Parameters:

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface Name

##### Returns:

Return Success (i.e Zero), otherwise one of the following errors  
MLD\_ERR\_NONE  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF

```

MLD_ERR_NO_SUCH_IFF
MLD_ERR_L3_NON_VLAN_IF
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_OOM

```

#### 2.2.2.9 `s_int32_t smi_mld_snoop_api_if_mrouter_set (struct smiclient_globals *azg, char *ifname, char *mrouter_ifname)`

This function statically identifies a particular VLAN constituent interface as a multi-cast router (mrouter) interface for MLD Snooping on the specified VLAN interface. This function may be invoked multiple times to configure multiple VLAN constituent interfaces as mrouter interfaces. `smi_mld_snoop_api_if_mrouter_set`

##### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface Name
- ← *mrouter\_ifname* Pointer to the interface-name string of the VLAN constituent interface to be identified as the mrouter interface

##### Returns:

Return Success (i.e Zero), otherwise one of the following errors

```

MLD_ERR_NONE
MLD_ERR_INVALID_VALUE
MLD_ERR_L2_PHYSICAL_IF
MLD_ERR_NO_SUCH_IFF
MLD_ERR_L3_NON_VLAN_IF
MLD_ERR_MALFORMED_ARG
MLD_ERR_NO_SUCH_GROUP_REC
MLD_ERR_NO_SUCH_SOURCE_REC
MLD_ERR_OOM

```

#### 2.2.2.10 `s_int32_t smi_mld_snoop_api_if_mrouter_unset (struct smiclient_globals *azg, char *ifname, char *mrouter_ifname)`

This function unsets the static configuration of a VLAN constituent interface as an mrouter interface on the specified VLAN interface. `smi_mld_snoop_api_if_mrouter_unset`

##### Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ifname* Interface Name
- ← *mrouter\_ifname* Pointer to the interface-name string of the VLAN constituent interface to be identified as the mrouter interface

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_NONE  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_L3\_NON\_VLAN\_IF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM

**2.2.2.11 s\_int32\_t smi\_mld\_snoop\_api\_if\_querier\_set (struct smiclient\_globals \* azg, char \* ifname)**

This function enables MLD Snooping Querier functionality on the specified VLAN interface. smi\_mld\_snoop\_api\_if\_querier\_set

**Parameters:**

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface Name

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_NONE  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_L3\_NON\_VLAN\_IF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM

**2.2.2.12 s\_int32\_t smi\_mld\_snoop\_api\_if\_querier\_unset (struct smiclient\_globals \* azg, char \* ifname)**

This function disables MLD Snooping Querier functionality on the specified VLAN interface. smi\_mld\_snoop\_api\_if\_querier\_unset

**Parameters:**

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface Name

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_NONE  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_L3\_NON\_VLAN\_IF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM

### 2.2.2.13 s\_int32\_t smi\_mld\_snoop\_api\_if\_report\_suppress\_set (struct smiclient\_globals \* *azg*, char \* *ifname*)

This function enables MLD Snooping report-suppression on all constituent interfaces of the specified VLAN interface. smi\_mld\_snoop\_api\_if\_report\_suppress\_set

**Parameters:**

← *azg* Pointer to the SMI client global structure  
 ← *ifname* Interface Name

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_NONE  
 MLD\_ERR\_INVALID\_VALUE  
 MLD\_ERR\_L2\_PHYSICAL\_IF  
 MLD\_ERR\_NO\_SUCH\_IFF  
 MLD\_ERR\_L3\_NON\_VLAN\_IF  
 MLD\_ERR\_MALFORMED\_ARG  
 MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
 MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
 MLD\_ERR\_OOM

### 2.2.2.14 s\_int32\_t smi\_mld\_snoop\_api\_if\_report\_suppress\_unset (struct smiclient\_globals \* *azg*, char \* *ifname*)

This function disables MLD Snooping report-suppression on all constituent interfaces of the specified VLAN interface. smi\_mld\_snoop\_api\_if\_report\_suppress\_unset

**Parameters:**

← *azg* Pointer to the SMI client global structure  
 ← *ifname* Interface Name

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_NONE  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_L3\_NON\_VLAN\_IF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM

**2.2.2.15 s\_int32\_t smi\_mld\_snoop\_api\_if\_snooping\_set (struct smiclient\_globals \* azg, char \* ifname)**

This function explicitly enables MLD Snooping on the specified VLAN interface.  
smi\_mld\_snoop\_api\_if\_snooping\_set

**Parameters:**

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface Name

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_VLAN\_IF\_NOT\_RUNNING  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_L3\_NON\_VLAN\_IF  
MLD\_ERR\_NONE

**2.2.2.16 s\_int32\_t smi\_mld\_snoop\_api\_if\_snooping\_unset (struct smiclient\_globals \* azg, char \* ifname)**

This function explicitly disables MLD Snooping on the specified VLAN interface. It also unsets all configuration associated with MLD Snooping on the VLAN interface.  
smi\_mld\_snoop\_api\_if\_snooping\_unset

**Parameters:**

← *azg* Pointer to the SMI client global structure  
← *ifname* Interface Name

**Returns:**

Return Success (i.e Zero), otherwise one of the following errors

MLD\_ERR\_NONE  
MLD\_ERR\_INVALID\_VALUE  
MLD\_ERR\_L2\_PHYSICAL\_IF  
MLD\_ERR\_NO\_SUCH\_IFF  
MLD\_ERR\_L3\_NON\_VLAN\_IF  
MLD\_ERR\_MALFORMED\_ARG  
MLD\_ERR\_NO\_SUCH\_GROUP\_REC  
MLD\_ERR\_NO\_SUCH\_SOURCE\_REC  
MLD\_ERR\_OOM

#### 2.2.2.17 `s_int32_t smi_mld_snoop_api_set_snooping (struct smiclient_globals * azg)`

This function enables MLD Snooping on all interfaces of this instance not explicitly (individually) disabled for MLD Snooping. MLD Snooping is globally enabled by default. `smi_mld_snoop_api_set_snooping`

##### Parameters:

← *azg* Pointer to the SMI client global structure

##### Returns:

Return Success (i.e Zero), otherwise one of the following errors  
MLD\_ERR\_NONE

#### 2.2.2.18 `s_int32_t smi_mld_snoop_api_unset_snooping (struct smiclient_globals * azg)`

This function disables MLD Snooping on all interfaces of this instance not explicitly (individually) enabled for MLD Snooping. `smi_mld_snoop_api_unset_snooping`

##### Parameters:

← *azg* Pointer to the SMI client global structure

##### Returns:

Return Success (i.e Zero), otherwise one of the following errors  
MLD\_ERR\_NONE

# Index

smi\_debug\_ipv6\_mld\_sdkapi  
    smi\_mld.h, 8

smi\_debug\_mld\_snoop\_sdkapi  
    smi\_mld\_snooping.h, 32

smi\_debug\_no\_ipv6\_mld\_sdkapi  
    smi\_mld.h, 8

smi\_debug\_no\_mld\_snoop\_sdkapi  
    smi\_mld\_snooping.h, 32

smi\_mld.h, 3

    smi\_debug\_ipv6\_mld\_sdkapi, 8

    smi\_debug\_no\_ipv6\_mld\_sdkapi, 8

    smi\_mrrib6\_mld\_clear\_sdkapi, 9

    smi\_mrrib6\_mld\_if\_access\_list\_set\_-  
        sdkapi, 9

    smi\_mrrib6\_mld\_if\_access\_list\_-  
        unset\_sdkapi, 10

    smi\_mrrib6\_mld\_if\_immediate\_-  
        leave\_set\_sdkapi, 10

    smi\_mrrib6\_mld\_if\_immediate\_-  
        leave\_unset\_sdkapi, 11

    smi\_mrrib6\_mld\_if\_limit\_set\_-  
        sdkapi, 11

    smi\_mrrib6\_mld\_if\_limit\_unset\_-  
        sdkapi, 12

    smi\_mrrib6\_mld\_if\_lmqc\_set\_-  
        sdkapi, 12

    smi\_mrrib6\_mld\_if\_lmqc\_unset\_-  
        sdkapi, 13

    smi\_mrrib6\_mld\_if\_lmqi\_set\_-  
        sdkapi, 13

    smi\_mrrib6\_mld\_if\_lmqi\_unset\_-  
        sdkapi, 14

    smi\_mrrib6\_mld\_if\_mrroute\_pxy\_-  
        set\_sdkapi, 14

    smi\_mrrib6\_mld\_if\_mrroute\_pxy\_-  
        unset\_sdkapi, 15

    smi\_mrrib6\_mld\_if\_pxy\_service\_-  
        set\_sdkapi, 15

    smi\_mrrib6\_mld\_if\_pxy\_service\_-  
        unset\_sdkapi, 16

    smi\_mrrib6\_mld\_if\_querier\_-  
        timeout\_set\_sdkapi, 16

    smi\_mrrib6\_mld\_if\_querier\_-  
        timeout\_unset\_sdkapi, 17

    smi\_mrrib6\_mld\_if\_query\_interval\_-  
        set\_sdkapi, 17

    smi\_mrrib6\_mld\_if\_query\_interval\_-  
        unset\_sdkapi, 18

    smi\_mrrib6\_mld\_if\_query\_-  
        response\_interval\_set\_sdkapi,  
        19

    smi\_mrrib6\_mld\_if\_query\_-  
        response\_interval\_unset\_-  
        sdkapi, 19

    smi\_mrrib6\_mld\_if\_robustness\_-  
        var\_set\_sdkapi, 20

    smi\_mrrib6\_mld\_if\_robustness\_-  
        var\_unset\_sdkapi, 20

    smi\_mrrib6\_mld\_if\_set\_sdkapi, 21

    smi\_mrrib6\_mld\_if\_startup\_query\_-  
        count\_set\_sdkapi, 21

    smi\_mrrib6\_mld\_if\_startup\_query\_-  
        count\_unset\_sdkapi, 22

    smi\_mrrib6\_mld\_if\_startup\_query\_-  
        interval\_set\_sdkapi, 22

    smi\_mrrib6\_mld\_if\_startup\_query\_-  
        interval\_unset\_sdkapi, 23

    smi\_mrrib6\_mld\_if\_static\_group\_-  
        source\_set\_sdkapi, 24

    smi\_mrrib6\_mld\_if\_static\_group\_-  
        source\_unset\_sdkapi, 24

    smi\_mrrib6\_mld\_if\_unset\_sdkapi, 25

    smi\_mrrib6\_mld\_if\_version\_set\_-  
        sdkapi, 25

    smi\_mrrib6\_mld\_if\_version\_unset\_-  
        sdkapi, 26

    smi\_mrrib6\_mld\_limit\_set\_sdkapi,  
        26

    smi\_mrrib6\_mld\_limit\_unset\_-  
        sdkapi, 27

- smi\_mrib6\_mld\_ssm\_map\_enable\_-  
set\_sdkapi, [27](#)
- smi\_mrib6\_mld\_ssm\_map\_enable\_-  
unset\_sdkapi, [27](#)
- smi\_mrib6\_mld\_ssm\_map\_static\_-  
set\_sdkapi, [28](#)
- smi\_mrib6\_mld\_ssm\_map\_static\_-  
unset\_sdkapi, [28](#)
- smi\_mld\_snoop\_api\_clear\_all  
smi\_mld\_snooping.h, [33](#)
- smi\_mld\_snoop\_api\_clear\_group  
smi\_mld\_snooping.h, [33](#)
- smi\_mld\_snoop\_api\_clear\_group\_if  
smi\_mld\_snooping.h, [34](#)
- smi\_mld\_snoop\_api\_clear\_if  
smi\_mld\_snooping.h, [34](#)
- smi\_mld\_snoop\_api\_if\_fast\_leave\_set  
smi\_mld\_snooping.h, [35](#)
- smi\_mld\_snoop\_api\_if\_fast\_leave\_unset  
smi\_mld\_snooping.h, [35](#)
- smi\_mld\_snoop\_api\_if\_mrouterset  
smi\_mld\_snooping.h, [36](#)
- smi\_mld\_snoop\_api\_if\_mrouterset\_unset  
smi\_mld\_snooping.h, [36](#)
- smi\_mld\_snoop\_api\_if\_querier\_set  
smi\_mld\_snooping.h, [37](#)
- smi\_mld\_snoop\_api\_if\_querier\_unset  
smi\_mld\_snooping.h, [37](#)
- smi\_mld\_snoop\_api\_if\_report\_-  
suppress\_set  
smi\_mld\_snooping.h, [38](#)
- smi\_mld\_snoop\_api\_if\_report\_-  
suppress\_unset  
smi\_mld\_snooping.h, [38](#)
- smi\_mld\_snoop\_api\_if\_snooping\_set  
smi\_mld\_snooping.h, [39](#)
- smi\_mld\_snoop\_api\_if\_snooping\_unset  
smi\_mld\_snooping.h, [39](#)
- smi\_mld\_snoop\_api\_set\_snooping  
smi\_mld\_snooping.h, [40](#)
- smi\_mld\_snoop\_api\_unset\_snooping  
smi\_mld\_snooping.h, [40](#)
- smi\_mld\_snooping.h, [30](#)
- smi\_debug\_mld\_snoop\_sdkapi, [32](#)
- smi\_debug\_no\_mld\_snoop\_sdkapi,  
[32](#)
- smi\_mld\_snoop\_api\_clear\_all, [33](#)
- smi\_mld\_snoop\_api\_clear\_group,  
[33](#)
- smi\_mld\_snoop\_api\_clear\_group\_if,  
[34](#)
- smi\_mld\_snoop\_api\_clear\_if, [34](#)
- smi\_mld\_snoop\_api\_if\_fast\_leave\_-  
set, [35](#)
- smi\_mld\_snoop\_api\_if\_fast\_leave\_-  
unset, [35](#)
- smi\_mld\_snoop\_api\_if\_mrouterset,  
[36](#)
- smi\_mld\_snoop\_api\_if\_mrouterset\_-  
unset, [36](#)
- smi\_mld\_snoop\_api\_if\_querier\_set,  
[37](#)
- smi\_mld\_snoop\_api\_if\_querier\_-  
unset, [37](#)
- smi\_mld\_snoop\_api\_if\_report\_-  
suppress\_set, [38](#)
- smi\_mld\_snoop\_api\_if\_report\_-  
suppress\_unset, [38](#)
- smi\_mld\_snoop\_api\_if\_snooping\_-  
set, [39](#)
- smi\_mld\_snoop\_api\_if\_snooping\_-  
unset, [39](#)
- smi\_mld\_snoop\_api\_set\_snooping,  
[40](#)
- smi\_mld\_snoop\_api\_unset\_-  
snooping, [40](#)
- smi\_mrib6\_mld\_clear\_sdkapi  
smi\_mld.h, [9](#)
- smi\_mrib6\_mld\_if\_access\_list\_set\_-  
sdkapi  
smi\_mld.h, [9](#)
- smi\_mrib6\_mld\_if\_access\_list\_unset\_-  
sdkapi  
smi\_mld.h, [10](#)
- smi\_mrib6\_mld\_if\_immediate\_leave\_-  
set\_sdkapi  
smi\_mld.h, [10](#)
- smi\_mrib6\_mld\_if\_immediate\_leave\_-  
unset\_sdkapi  
smi\_mld.h, [11](#)
- smi\_mrib6\_mld\_if\_limit\_set\_sdkapi  
smi\_mld.h, [11](#)
- smi\_mrib6\_mld\_if\_limit\_unset\_sdkapi  
smi\_mld.h, [12](#)
- smi\_mrib6\_mld\_if\_lmqc\_set\_sdkapi  
smi\_mld.h, [12](#)
- smi\_mrib6\_mld\_if\_lmqc\_unset\_sdkapi  
smi\_mld.h, [13](#)
- smi\_mrib6\_mld\_if\_lmqi\_set\_sdkapi



- smi\_mld.h, [13](#)
- smi\_mrib6\_mld\_if\_lmqi\_unset\_sdkapi  
smi\_mld.h, [14](#)
- smi\_mrib6\_mld\_if\_mroute\_pxy\_set\_-  
sdkapi  
smi\_mld.h, [14](#)
- smi\_mrib6\_mld\_if\_mroute\_pxy\_unset\_-  
sdkapi  
smi\_mld.h, [15](#)
- smi\_mrib6\_mld\_if\_pxy\_service\_set\_-  
sdkapi  
smi\_mld.h, [15](#)
- smi\_mrib6\_mld\_if\_pxy\_service\_unset\_-  
sdkapi  
smi\_mld.h, [16](#)
- smi\_mrib6\_mld\_if\_querier\_timeout\_-  
set\_sdkapi  
smi\_mld.h, [16](#)
- smi\_mrib6\_mld\_if\_querier\_timeout\_-  
unset\_sdkapi  
smi\_mld.h, [17](#)
- smi\_mrib6\_mld\_if\_query\_interval\_set\_-  
sdkapi  
smi\_mld.h, [17](#)
- smi\_mrib6\_mld\_if\_query\_interval\_-  
unset\_sdkapi  
smi\_mld.h, [18](#)
- smi\_mrib6\_mld\_if\_query\_response\_-  
interval\_set\_sdkapi  
smi\_mld.h, [19](#)
- smi\_mrib6\_mld\_if\_query\_response\_-  
interval\_unset\_sdkapi  
smi\_mld.h, [19](#)
- smi\_mrib6\_mld\_if\_robustness\_var\_set\_-  
sdkapi  
smi\_mld.h, [20](#)
- smi\_mrib6\_mld\_if\_robustness\_var\_-  
unset\_sdkapi  
smi\_mld.h, [20](#)
- smi\_mrib6\_mld\_if\_set\_sdkapi  
smi\_mld.h, [21](#)
- smi\_mrib6\_mld\_if\_startup\_query\_-  
count\_set\_sdkapi  
smi\_mld.h, [21](#)
- smi\_mrib6\_mld\_if\_startup\_query\_-  
count\_unset\_sdkapi  
smi\_mld.h, [22](#)
- smi\_mrib6\_mld\_if\_startup\_query\_-  
interval\_set\_sdkapi  
smi\_mld.h, [22](#)
- smi\_mrib6\_mld\_if\_startup\_query\_-  
interval\_unset\_sdkapi  
smi\_mld.h, [23](#)
- smi\_mrib6\_mld\_if\_static\_group\_-  
source\_set\_sdkapi  
smi\_mld.h, [24](#)
- smi\_mrib6\_mld\_if\_static\_group\_-  
source\_unset\_sdkapi  
smi\_mld.h, [24](#)
- smi\_mrib6\_mld\_if\_unset\_sdkapi  
smi\_mld.h, [25](#)
- smi\_mrib6\_mld\_if\_version\_set\_sdkapi  
smi\_mld.h, [25](#)
- smi\_mrib6\_mld\_if\_version\_unset\_sdkapi  
smi\_mld.h, [26](#)
- smi\_mrib6\_mld\_limit\_set\_sdkapi  
smi\_mld.h, [26](#)
- smi\_mrib6\_mld\_limit\_unset\_sdkapi  
smi\_mld.h, [27](#)
- smi\_mrib6\_mld\_ssm\_map\_enable\_set\_-  
sdkapi  
smi\_mld.h, [27](#)
- smi\_mrib6\_mld\_ssm\_map\_enable\_-  
unset\_sdkapi  
smi\_mld.h, [27](#)
- smi\_mrib6\_mld\_ssm\_map\_static\_set\_-  
sdkapi  
smi\_mld.h, [28](#)
- smi\_mrib6\_mld\_ssm\_map\_static\_unset\_-  
sdkapi  
smi\_mld.h, [28](#)