ZebOS-XP GMRP SMI Reference

IP Infusion Inc.

Generated by Doxygen 1.6.1

Wed Dec 16 12:33:23 2015

Contents

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 2

File Index

2.1 File List

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

Chapter 3

Data Structure Documentation

3.1 gmrp_msg_ Struct Reference

Data Fields

- smi_cindex_t **cindex_0**
- u_int32_t **vr_id**
- char **protocol** [255]
- char **bridge_name** [255]
- u_int16_t **vlan_id**
- char **ifname** [255]
- pal_time_t **timer_value**
- char **reg_type** [255]
- u_char **is_gmrp_en**
- int **timer_type**
- u_char **is_p2p**
- u_char **periodic_timer_status**
- int **regis_type**
- int **forward_all**
- int **extended_filtering_flag**
- int **smi_debug**
- struct xmrpTimers **timerData**
- struct xmrpVlanStats **statsData**
- struct xmrpCommonList **xmrpData**

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

- smi_gmrp_msg.h

3.2 xmrpBridgeConfig Struct Reference

Data Fields

- char **portName** [32]
- char **ifState**
- u_char **registrationType**
- u_char **forwardAll**
- pal_time_t **timerValues** [SMI_GMRP_GARP_MAX_TIMERS]

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

- smi_gmrp_msg.h

3.3 xmrpCommonList Struct Reference

Data Fields

- int **have_more**
- int **more_count**
- struct list * **xmrpList**

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

- smi_gmrp_msg.h

3.4 xmrpFsmState Struct Reference

Data Fields

- char **portName** [32]
- u_int16_t **vlanId**
- int **numState**
- char **attrIndex** [128]
- char **applicantState** [128][4]
- char **registrarState** [128][4]

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

- smi_gmrp_msg.h

3.5 xmrpTimers Struct Reference

Data Fields

- pal_time_t **timerValues** [SMI_GMRP_GARP_MAX_TIMERS]

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

- smi_gmrp_msg.h

3.6 xmrpVlanStats Struct Reference

Data Fields

- u_int32_t **rxCounters** [XMRP_TOTAL_ATTR_EVENTS]
- u_int32_t **txCounters** [XMRP_TOTAL_ATTR_EVENTS]

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

- smi_gmrp_msg.h

Chapter 4

File Documentation

4.1 smi_gmrp.h File Reference

Provides APIs for managing GARP Multicast Registration Protocol (GMRP) GMRP is a Generic Attribute Registration Protocol (GARP) application that provides a constrained multicast flooding facility similar to IGMP snooping. The APIs provided in this file forms the basis of ZebOS GMRP/MMRP management. These APIs are used by various north bound management interfaces like CLI, SNMP and SMI. `#include "smi_client.h"`

```
#include "smi_gmrp_msg.h"
```

Functions

- int **smi_client_create_n_send_gmrp_msg** (struct smi_client_handler *async, int vrid, gmrp_msg *msg, int optype)
- int **smi_client_read_sync_gmrp_msg** (struct smi_client_handler *ach, int msgtype, void *getmsg)
- s_int32_t smi_xmrp_enable_global (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)

  Enable the GMRP/MMRP feature globally for the given bridge instance. NOTE: GMRP cannot be enabled if IGMP snooping is enabled.

- s_int32_t **smi_xmrp_enable_global_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)
- s_int32_t smi_xmrp_disable_global (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)

  Disable the GMRP/MMRP feature globally for the given bridge instance.

- s_int32_t **smi_xmrp_disable_global_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)
- s_int32_t smi_xmrp_enable_per_vlan (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name, u_int16_t vlan_id)

Enable the GMRP/MMRP feature per VLAN for the given bridge instance. NOTE: GMRP cannot be enabled if IGMP snooping is enabled.

- s_int32_t **smi_xmrp_enable_per_vlan_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name, u_int16_t vlan_id)
- s_int32_t smi_xmrp_disable_per_vlan (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name, u_int16_t vlan_id)

  Disable the GMRP/MMRP feature per VLAN for the given bridge instance. NOTE: GMRP cannot be enabled if IGMP snooping is enabled.

- s_int32_t **smi_xmrp_disable_per_vlan_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name, u_int16_t vlan_id)

- s_int32_t smi_xmrp_enable_per_port_per_vlan (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, u_int16_t vlan_id)

  Enables the GMRP/MMRP for the particular port instance of particular VLAN and starts required timers. NOTE: GMRP cannot be enabled if IGMP snooping is enabled.

- s_int32_t **smi_xmrp_enable_per_port_per_vlan_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, u_int16_t vlan_id)

- s_int32_t smi_xmrp_disable_per_port_per_vlan (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, u_int16_t vlan_id)

  Disables the GMRP/MMRP for the particular port instance of particular VLAN and starts required timers. NOTE: GMRP cannot be enabled if IGMP snooping is enabled.

- s_int32_t **smi_xmrp_disable_per_port_per_vlan_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, u_int16_t vlan_id)

- s_int32_t smi_xmrp_set_join_timer (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, pal_time_t timer_value)

  Sets the join timer, used for joining the group, measured in centiseconds used in GMRP/MMRP protocol at interface level.
  The following types of timers are supported
  Join: Specify the timer for joining the group
  Leave: Specify the timer for leaving the group
  Join: LeaveAll Specify the timer for leaving all groups
  .

- s_int32_t **smi_xmrp_set_join_timer_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, pal_time_t timer_value)
- s_int32_t smi_xmrp_set_leave_timer (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, pal_time_t timer_value)

  Sets the leave timer, used for leaving the group, measured in centiseconds used in GMRP/MMRP protocol at interface level.
  The following types of timers are supported
  Join: Specify the timer for joining the group

Leave: Specify the timer for leaving the group
Join: LeaveAll Specify the timer for leaving all groups
.

- s_int32_t **smi_xmrp_set_leave_timer_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, pal_time_t timer_value)
- s_int32_t smi_xmrp_set_leave_all_timer (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, pal_time_t timer_value)

  Sets the leave timer, used for leaving from all groups, measured in centiseconds used in GMRP/MMRP protocol at interface level.
  The following types of timers are supported
  Join: Specify the timer for joining the group
  Leave: Specify the timer for leaving the group
  Join: LeaveAll Specify the timer for leaving all groups
  .

- s_int32_t **smi_xmrp_set_leave_all_timer_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, pal_time_t timer_value)

- s_int32_t smi_mmrp_set_if_mode_to_p2p (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname)

  Set the interface mode to point-to-point behavior option on an MMRP-enabled bridge.

- s_int32_t **smi_mmrp_set_if_mode_to_p2p_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname)
- s_int32_t smi_mmrp_unset_if_mode_to_p2p (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname)

  Unset the interface mode to point-to-point behavior option on an MMRP-enabled bridge.

- s_int32_t **smi_mmrp_unset_if_mode_to_p2p_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname)
- s_int32_t smi_mmrp_enable_periodic_timer_global (struct smiclient_globals *azg, u_int32_t vr_id, char *bridge_name)

  Enable the periodic timer option for all ports of an MMRP-enabled bridge.

- s_int32_t **smi_mmrp_enable_periodic_timer_global_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *bridge_name)
- s_int32_t smi_mmrp_disable_periodic_timer_global (struct smiclient_globals *azg, u_int32_t vr_id, char *bridge_name)

  Enable the periodic timer option for all ports of an MMRP-enabled bridge.

- s_int32_t **smi_mmrp_disable_periodic_timer_global_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *bridge_name)
- s_int32_t smi_mmrp_enable_periodic_timer_per_port (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname)

  Enable the periodic timer option for the given port of an MMRP-enabled bridge.

- s_int32_t **smi_mmrp_enable_periodic_timer_per_port_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname)
- s_int32_t smi_mmrp_disable_periodic_timer_per_port (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname)

  Disable the periodic timer option for the given port of an MMRP-enabled bridge.

- s_int32_t **smi_mmrp_disable_periodic_timer_per_port_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname)
- s_int32_t smi_xmrp_enable_per_port (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname)

  Enables the GMRP/MMRP on the particular port instance of all VLANs and starts required timers.

- s_int32_t **smi_xmrp_enable_per_port_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname)
- s_int32_t smi_xmrp_enable_all_port (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol)

  Enables the GMRP/MMRP on all the ports instance of all VLANs and starts required timers.

- s_int32_t **smi_xmrp_enable_all_port_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol)
- s_int32_t smi_xmrp_disable_per_port (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname)

  Disables the GMRP/MMRP on the particular port instance of all VLANs and starts required timers.

- s_int32_t **smi_xmrp_disable_per_port_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname)
- s_int32_t smi_xmrp_disable_all_port (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol)

  Disables the GMRP/MMRP on all the ports instance of all VLANs and starts required timers.

- s_int32_t **smi_xmrp_disable_all_port_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol)
- s_int32_t smi_xmrp_set_registration_type (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, char *reg_type)

  Sets the GMRP/MMRP port's multicast group registration type to one of the following.
  fixed: The multicast groups currently registered on the switch will remain on the port, but subsequent new registrations or de-registrations based on timers do not affect the port.
  forbidden: All registered multicast groups are de-registered, and prevents further multicast registration on the port.
  normal: Set multicast group registration and de-registration to dynamic.
  restricted: Set to restricted registration.
  NOTE: GMRP cannot be enabled if IGMP snooping is enabled.

- s_int32_t **smi_xmrp_set_registration_type_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, char *reg_type)
- s_int32_t smi_xmrp_enable_forward_all (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname)

  Enable GMRP/MMRP protocol's 'forwrd all' option on the interface.

- s_int32_t **smi_xmrp_enable_forward_all_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname)
- s_int32_t smi_xmrp_disable_forward_all (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname)

  Disable GMRP/MMRP protocol's 'forwrd all' option on the interface.

- s_int32_t **smi_xmrp_disable_forward_all_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname)
- s_int32_t smi_xmrp_enable_extended_filtering (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)

  Enable GMRP/MMRP to enable 'extended filtering option at bridge level.

- s_int32_t **smi_xmrp_enable_extended_filtering_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)
- s_int32_t smi_xmrp_disable_extended_filtering (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)

  Disable GMRP/MMRP to enable 'extended filtering option at bridge level.

- s_int32_t **smi_xmrp_disable_extended_filtering_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)
- s_int32_t smi_xmrp_clear_all_vlan_statistics (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)

  Clear GMRP/MMRP statistics of all VLANs.

- s_int32_t **smi_xmrp_clear_all_vlan_statistics_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name)
- s_int32_t smi_xmrp_clear_per_vlan_statistics (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, u_int16_t vlan_id, char *bridge_name)

  Clear GMRP/MMRP statistics of all VLANs.

- s_int32_t smi_xmrp_show_times (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *ifname, struct xmrpTimers *timerData)

  Show GMRP/MMRP configured timer values.

- s_int32_t smi_xmrp_show_bridge_configuration (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name, struct list *brConfList, int(*funPointer)(struct list *brConfList))

  Show GMRP/MMRP bridge configuration.

- s_int32_t smi_xmrp_show_finite_state_machine (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name, struct list *fsmStateList, int(*funPointer)(struct list *fsmStateList))

Show GMRP/MMRP finite state machine details.

- s_int32_t smi_xmrp_show_per_vlan_statistics (struct smiclient_globals *azg, u_int32_t vr_id, char *protocol, char *bridge_name, u_int16_t vlan_id, struct xmrpVlanStats *vlanStats)

  Show various GMRP/MMRP related statistics of particular VLAN.

- int smi_is_gmrp_enabled (struct smiclient_globals *azg, int vr_id, char *protocol, char *bridge_name, u_char *is_gmrp_en)

  Function to verify whether GMRP is enabled globally.

- int smi_xmrp_get_timer (struct smiclient_globals *azg, int vr_id, char *protocol, char *ifname, int timer_type, pal_time_t *timer_value)

  Function to get configured timer value for the specific timer.

- int smi_mmrp_verify_p2p_mode_set (struct smiclient_globals *azg, int vr_id, char *ifname, u_char *is_p2p)

  Function to verify whether p2p mode is enable on port.

- int smi_mmrp_get_periodic_timer_per_port_status (struct smiclient_globals *azg, int vr_id, char *ifname, u_char *periodic_timer_status)

  Function to verify periodic timer is set for the specific port.

- int smi_xmrp_get_registration_type (struct smiclient_globals *azg, int vr_id, char *protocol, char *ifname, int *regis_type)

  Function to get muticast group registration type.

- int smi_xmrp_get_forward_all_flag (struct smiclient_globals *azg, int vr_id, char *protocol, char *ifname, int *forward_all)

  Function to verify foward all flag is enabled.

- int smi_xmrp_get_extended_filtering_flag (struct smiclient_globals *azg, int vr_id, char *protocol, char *bridge_name, int *extended_filtering_flag)

  Function to get extended filtering flag.

- int smi_gmrp_debug_set (struct smiclient_globals *azg, int vr_id, int debug)

  sets for debug value

- int smi_gmrp_debug_unset (struct smiclient_globals *azg, int vr_id, int debug)

  unset for debug value

- int smi_gmrp_show_debugging (struct smiclient_globals *azg, int vr_id, u_int32_t *smi_debug)

  show deebug events

- int **smi_gmrp_debug_set_validate** (struct smiclient_globals *azg, u_int32_t debug)

- int **smi_gmrp_debug_unset_validate** (struct smiclient_globals *azg, u_int32_t debug)

4.1.1 Detailed Description

Provides APIs for managing GARP Multicast Registration Protocol (GMRP) GMRP is a Generic Attribute Registration Protocol (GARP) application that provides a constrained multicast flooding facility similar to IGMP snooping. The APIs provided in this file forms the basis of ZebOS GMRP/MMRP management. These APIs are used by various north bound management interfaces like CLI, SNMP and SMI.

4.1.2 Function Documentation

4.1.2.1 int smi_gmrp_debug_set (struct smiclient_globals * *azg*, int *vr_id*, int *debug*)

sets for debug value smi_gmrp_debug_set

Parameters:

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

← ***vr_id***

← ***debug***

Returns:

0 on success RESULT_OK on success

4.1.2.2 int smi_gmrp_debug_unset (struct smiclient_globals * *azg*, int *vr_id*, int *debug*)

unset for debug value smi_gmrp_debug_unset

Parameters:

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

← ***vr_id***

← ***debug***

Returns:

0 on success RESULT_OK on success

4.1.2.3 int smi_gmrp_show_debugging (struct smiclient_globals * *azg*, int *vr_id*, u_int32_t * *smi_debug*)

show deebug events smi_gmrp_show_debugging

Parameters:

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

← ***vr_id***

→ ***smi_debug***

Returns:

0 on success RESULT_OK on success

4.1.2.4 int smi_is_gmrp_enabled (struct smiclient_globals * *azg*, int *vr_id*, char * *protocol*, char * *bridge_name*, u_char * *is_gmrp_en*)

Function to verify whether GMRP is enabled globally. smi_is_gmrp_enabled

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name <1-32>

→ ***is_gmrp_en*** Flag to mention whether gmrp enabled or disabled
Enabled (1)
Disable (0)

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.5 s_int32_t smi_mmrp_disable_periodic_timer_global (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *bridge_name*)

Enable the periodic timer option for all ports of an MMRP-enabled bridge. smi_mmrp_disable_periodic_timer_global

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***bridge_name*** Bridge name

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_SPBD_TYPE
NSM_BRIDGE_NO_PORT_CFG
NSM_BRIDGE_ERR_NOT_BOUND
NSM_BRIDGE_ERR_MMRP_NOCONFIG
NSM_ERR_MMRP_NOCONFIG_ONPORT

4.1.2.6 s_int32_t smi_mmrp_disable_periodic_timer_per_port (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *ifname*)

Disable the periodic timer option for the given port of an MMRP-enabled bridge. smi_-mmrp_disable_periodic_timer_per_port

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***ifname*** Interface name

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_SPBD_TYPE
NSM_BRIDGE_NO_PORT_CFG
NSM_BRIDGE_ERR_NOT_BOUND
NSM_BRIDGE_ERR_MMRP_NOCONFIG
NSM_ERR_MMRP_NOCONFIG_ONPORT

4.1.2.7 s_int32_t smi_mmrp_enable_periodic_timer_global (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *bridge_name*)

Enable the periodic timer option for all ports of an MMRP-enabled bridge. smi_-mmrp_enable_periodic_timer_global

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***bridge_name*** Bridge name

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND

NSM_BRIDGE_ERR_SPBD_TYPE
NSM_BRIDGE_NO_PORT_CFG
NSM_BRIDGE_ERR_NOT_BOUND
NSM_BRIDGE_ERR_MMRP_NOCONFIG
NSM_ERR_MMRP_NOCONFIG_ONPORT

4.1.2.8 s_int32_t smi_mmrp_enable_periodic_timer_per_port (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *ifname*)

Enable the periodic timer option for the given port of an MMRP-enabled bridge. smi_-mmrp_enable_periodic_timer_per_port

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***ifname*** Interface name

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_SPBD_TYPE
NSM_BRIDGE_NO_PORT_CFG
NSM_BRIDGE_ERR_NOT_BOUND
NSM_BRIDGE_ERR_MMRP_NOCONFIG
NSM_ERR_MMRP_NOCONFIG_ONPORT

4.1.2.9 int smi_mmrp_get_periodic_timer_per_port_status (struct smiclient_globals * *azg*, int *vr_id*, char * *ifname*, u_char * *periodic_timer_status*)

Function to verify periodic timer is set for the specific port. smi_mmrp_get_periodic_-timer_per_port_status

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***ifname*** Interface Name

→ ***periodic_timer_status*** timer status

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.10 s_int32_t smi_mmrp_set_if_mode_to_p2p (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *ifname*)

Set the interface mode to point-to-point behavior option on an MMRP-enabled bridge. smi_mmrp_set_if_mode_to_p2p

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***ifname*** Interface name

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.11 s_int32_t smi_mmrp_unset_if_mode_to_p2p (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *ifname*)

Unset the interface mode to point-to-point behavior option on an MMRP-enabled bridge. smi_mmrp_unset_if_mode_to_p2p

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***ifname*** Interface name

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.12 int smi_mmrp_verify_p2p_mode_set (struct smiclient_globals * *azg*, int *vr_id*, char * *ifname*, u_char * *is_p2p*)

Function to verify whether p2p mode is enable on port. smi_mmrp_verify_p2p_-mode_set

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***ifname*** Interface Name

→ ***is_p2p*** 1 - mode p2p

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.13 s_int32_t smi_xmrp_clear_all_vlan_statistics (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*)

Clear GMRP/MMRP statistics of all VLANs. smi_xmrp_clear_all_vlan_statistics

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name <1-32>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.14 s_int32_t smi_xmrp_clear_per_vlan_statistics (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, u_int16_t *vlan_id*, char * *bridge_name*)

Clear GMRP/MMRP statistics of all VLANs. smi_xmrp_clear_per_vlan_statistics

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***vlan_id*** VLAN number <1-4094>

← ***bridge_name*** Bridge name <1-32>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.15 s_int32_t smi_xmrp_disable_all_port (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*)

Disables the GMRP/MMRP on all the ports instance of all VLANs and starts required timers. smi_xmrp_disable_all_port

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_NOT_CFG
NSM_BRIDGE_NO_PORT_CFG

4.1.2.16 s_int32_t smi_xmrp_disable_extended_filtering (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*)

Disable GMRP/MMRP to enable 'extended filtering option at bridge level. smi_-xmrp_disable_extended_filtering

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name <1-32>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE

4.1.2.17 s_int32_t smi_xmrp_disable_forward_all (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*)

Disable GMRP/MMRP protocol's 'forwrd all' option on the interface. smi_xmrp_-disable_forward_all

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

Returns:

0 on success, otherwise one of the following error codes
NSM_ERR_GMRP_NOCONFIG_ONPORT
NSM_BRIDGE_ERR_GENERAL

4.1.2.18 s_int32_t smi_xmrp_disable_global (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*)

Disable the GMRP/MMRP feature globally for the given bridge instance. smi_xmrp_-disable_global

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name <1-32>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND

4.1.2.19 s_int32_t smi_xmrp_disable_per_port (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*)

Disables the GMRP/MMRP on the particular port instance of all VLANs and starts required timers. smi_xmrp_disable_per_port

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL
NSM_BRIDGE_ERR_SPBD_TYPE

4.1.2.20 s_int32_t smi_xmrp_disable_per_port_per_vlan (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*, u_int16_t *vlan_id*)

Disables the GMRP/MMRP for the particular port instance of particular VLAN and starts required timers. NOTE: GMRP cannot be enabled if IGMP snooping is enabled. smi_xmrp_disable_per_port_per_vlan

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

← ***vlan_id*** VLAN number <1-4094>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL
NSM_BRIDGE_ERR_GMRP_NOCONFIG
NSM_GMRP_ERR_GMRP_NOT_CFG_ON_VLAN
NSM_GMRP_ERR_VLAN_NOT_CFG_ON_PORT
NSM_ERR_GMRP_NOCONFIG_ONPORT
NSM_GMRP_ERR_GMRP_NOT_CFG_ON_PORT_VLAN
NSM_GMRP_ERR_GMRP_GLOBAL_CFG_PORT

4.1.2.21 s_int32_t smi_xmrp_disable_per_vlan (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*, u_int16_t *vlan_id*)

Disable the GMRP/MMRP feature per VLAN for the given bridge instance. NOTE: GMRP cannot be enabled if IGMP snooping is enabled. smi_xmrp_disable_per_vlan

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name <1-32>

← ***vlan_id*** VLAN number <1-4094>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_MEM

4.1.2.22 s_int32_t smi_xmrp_enable_all_port (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*)

Enables the GMRP/MMRP on all the ports instance of all VLANs and starts required timers. smi_xmrp_enable_all_port

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_NOT_CFG
NSM_BRIDGE_NO_PORT_CFG

4.1.2.23 s_int32_t smi_xmrp_enable_extended_filtering (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*)

Enable GMRP/MMRP to enable 'extended filtering option at bridge level. smi_xmrp_-enable_extended_filtering

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name <1-32>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE

4.1.2.24 s_int32_t smi_xmrp_enable_forward_all (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*)

Enable GMRP/MMRP protocol's 'forwrd all' option on the interface. smi_xmrp_-enable_forward_all

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

Returns:

0 on success, otherwise one of the following error codes
NSM_ERR_GMRP_NOCONFIG_ONPORT
NSM_BRIDGE_ERR_GENERAL

4.1.2.25 s_int32_t smi_xmrp_enable_global (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*)

Enable the GMRP/MMRP feature globally for the given bridge instance. NOTE: GMRP cannot be enabled if IGMP snooping is enabled. smi_xmrp_enable_global

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name <1-32>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_MEM

4.1.2.26 s_int32_t smi_xmrp_enable_per_port (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*)

Enables the GMRP/MMRP on the particular port instance of all VLANs and starts required timers. smi_xmrp_enable_per_port

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL
NSM_BRIDGE_ERR_SPBD_TYPE

4.1.2.27 s_int32_t smi_xmrp_enable_per_port_per_vlan (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*, u_int16_t *vlan_id*)

Enables the GMRP/MMRP for the particular port instance of particular VLAN and starts required timers. NOTE: GMRP cannot be enabled if IGMP snooping is enabled. smi_xmrp_enable_per_port_per_vlan

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

← ***vlan_id*** VLAN number <1-4094>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL
NSM_BRIDGE_ERR_GMRP_NOCONFIG
NSM_GMRP_ERR_GMRP_NOT_CFG_ON_VLAN
NSM_GMRP_ERR_VLAN_NOT_CFG_ON_PORT
NSM_GMRP_ERR_GMRP_GLOBAL_CFG_PORT

4.1.2.28 s_int32_t smi_xmrp_enable_per_vlan (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*, u_int16_t *vlan_id*)

Enable the GMRP/MMRP feature per VLAN for the given bridge instance. NOTE: GMRP cannot be enabled if IGMP snooping is enabled. smi_xmrp_enable_per_vlan

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name <1-32>

← ***vlan_id*** VLAN number <1-4094>

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_MEM

4.1.2.29 int smi_xmrp_get_extended_filtering_flag (struct smiclient_globals * *azg*, int *vr_id*, char * *protocol*, char * *bridge_name*, int * *extended_filtering_flag*)

Function to get extended filtering flag. smi_xmrp_get_extended_filtering_flag

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** protocol

← ***ifname*** Interface Name

→ ***extended_filtering_flag*** One of the following values

enabled (1)

disabled (0)

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.30 int smi_xmrp_get_forward_all_flag (struct smiclient_globals * *azg*, int *vr_id*, char * *protocol*, char * *ifname*, int * *forward_all*)

Function to verify foward all flag is enabled. smi_xmrp_get_forward_all_flag

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** protocol

← ***ifname*** Interface Name

→ ***forward_all*** One of the following values

Forward all enabled (1)

Forward all disabled (0)

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.31 int smi_xmrp_get_registration_type (struct smiclient_globals * *azg*, int *vr_id*, char * *protocol*, char * *ifname*, int * *regis_type*)

Function to get muticast group registration type. smi_xmrp_get_registration_type

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** protocol

← ***ifname*** Interface Name

→ ***regis_type*** One of the following values
GID_EVENT_NORMAL_REGISTRATION (8)
GID_EVENT_FIXED_REGISTRATION (9)
GID_EVENT_FORBID_REGISTRATION (10)
GID_EVENT_RESTRICTED_GROUP_REGISTRATION (37)

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.32 int smi_xmrp_get_timer (struct smiclient_globals * *azg*, int *vr_id*, char * *protocol*, char * *ifname*, int *timer_type*, pal_time_t * *timer_value*)

Function to get configured timer value for the specific timer. smi_xmrp_get_timer

Parameters:

← ***azg*** Pointer to the SMI client global structure
← ***vr_id*** Virtual Router ID <0-255>
← ***protocol*** Protocol type string {gmrp|mmrp}
← ***ifname*** Interface Name
← ***timer_type*** Pass one of the following timer type
SMI_GARP_JOIN_TIMER
SMI_GARP_LEAVE_TIMER
SMI_GARP_LEAVE_ALL_TIMER
→ ***timer_value*** timer value

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL

4.1.2.33 s_int32_t smi_xmrp_set_join_timer (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*, pal_time_t *timer_value*)

Sets the join timer, used for joining the group, measured in centiseconds used in GMRP/MMRP protocol at interface level.

The following types of timers are supported

Join: Specify the timer for joining the group

Leave: Specify the timer for leaving the group

Join: LeaveAll Specify the timer for leaving all groups

. smi_xmrp_set_join_timer

Parameters:

← ***azg*** Pointer to the SMI client global structure
← ***vr_id*** Virtual Router ID <0-255>
← ***protocol*** Protocol type string {gmrp|mmrp}
← ***ifname*** Interface name
← ***timer_value*** Timer value in centiseconds <1-0xffffffff>

Returns:

0 on success, otherwise one of the following error codes
NSM_ERR_GMRP_NOCONFIG_ONPORT
NSM_BRIDGE_ERR_GENERAL

4.1.2.34 s_int32_t smi_xmrp_set_leave_all_timer (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*, pal_time_t *timer_value*)

Sets the leave timer, used for leaving from all groups, measured in centiseconds used in GMRP/MMRP protocol at interface level.

The following types of timers are supported

Join: Specify the timer for joining the group

Leave: Specify the timer for leaving the group

Join: LeaveAll Specify the timer for leaving all groups

. smi_xmrp_set_leave_all_timer

Parameters:

← ***azg*** Pointer to the SMI client global structure
← ***vr_id*** Virtual Router ID <0-255>
← ***protocol*** Protocol type string {gmrp|mmrp}
← ***ifname*** Interface name
← ***timer_value*** Timer value in centiseconds <1-0xffffffff>

Returns:

0 on success, otherwise one of the following error codes
NSM_ERR_GMRP_NOCONFIG_ONPORT
NSM_BRIDGE_ERR_GENERAL

4.1.2.35 s_int32_t smi_xmrp_set_leave_timer (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*, pal_time_t *timer_value*)

Sets the leave timer, used for leaving the group, measured in centiseconds used in GMRP/MMRP protocol at interface level.

The following types of timers are supported

Join: Specify the timer for joining the group

Leave: Specify the timer for leaving the group

Join: LeaveAll Specify the timer for leaving all groups

. smi_xmrp_set_leave_timer

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

← ***timer_value*** Timer value in centiseconds <1-0xffffffff>

Returns:

0 on success, otherwise one of the following error codes
NSM_ERR_GMRP_NOCONFIG_ONPORT
NSM_BRIDGE_ERR_GENERAL

4.1.2.36 s_int32_t smi_xmrp_set_registration_type (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*, char * *reg_type*)

Sets the GMRP/MMRP port's multicast group registration type to one of the following.

fixed: The multicast groups currently registered on the switch will remain on the port, but subsequent new registrations or de-registrations based on timers do not affect the port.

forbidden: All registered multicast groups are de-registered, and prevents further multicast registration on the port.

normal: Set multicast group registration and de-registration to dynamic.

restricted: Set to restricted registration.

NOTE: GMRP cannot be enabled if IGMP snooping is enabled. smi_xmrp_set_-registration_type

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

← ***reg_type*** Registration types {fixed|forbidden|normal|restricted}

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL
NSM_BRIDGE_ERR_GMRP_NOCONFIG
NSM_GMRP_ERR_GMRP_NOT_CFG_ON_VLAN
NSM_GMRP_ERR_VLAN_NOT_CFG_ON_PORT
NSM_ERR_GMRP_NOCONFIG_ONPORT
NSM_GMRP_ERR_GMRP_NOT_CFG_ON_PORT_VLAN
NSM_GMRP_ERR_GMRP_GLOBAL_CFG_PORT

4.1.2.37 s_int32_t smi_xmrp_show_bridge_configuration (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*, struct list * *brConfList*, int(*)(struct list *brConfList) *funPointer*)

Show GMRP/MMRP bridge configuration. smi_xmrp_show_bridge_configuration

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name

→ ***brConfList*** Pointer to linked list of structure xmrpBridgeConfig

← ***funPointer*** Callback function pointer

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_GMRP_NOCONFIG
CLI_ERROR

4.1.2.38 s_int32_t smi_xmrp_show_finite_state_machine (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*, struct list * *fsmStateList*, int(*)(struct list *fsmStateList) *funPointer*)

Show GMRP/MMRP finite state machine details. smi_xmrp_show_finite_state_machine_details

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***bridge_name*** Bridge name

→ ***fsmStateList*** Pointer to linked list of structure xmrpFsmState

← ***funPointer*** Callback function pointer

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_GMRP_NOCONFIG
CLI_ERROR

4.1.2.39 s_int32_t smi_xmrp_show_per_vlan_statistics (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *bridge_name*, u_int16_t *vlan_id*, struct xmrpVlanStats * *vlanStats*)

Show various GMRP/MMRP related statistics of particular VLAN. smi_xmrp_show_-per_vlan_statistics

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***vlan_id*** VLAN number <1-4094>

← ***bridge_name*** Bridge name <1-32>

→ ***vlanStats*** Pointer to structure xmrpVlanStats

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_GENERAL
RESULT_ERROR

4.1.2.40 s_int32_t smi_xmrp_show_times (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *protocol*, char * *ifname*, struct xmrpTimers * *timerData*)

Show GMRP/MMRP configured timer values. smi_xmrp_show_timers

Parameters:

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

← ***vr_id*** Virtual Router ID <0-255>

← ***protocol*** Protocol type string {gmrp|mmrp}

← ***ifname*** Interface name

→ ***timerData*** Pointer to xmrpTimers structure

Returns:

0 on success, otherwise one of the following error codes
NSM_BRIDGE_ERR_NOTFOUND
NSM_BRIDGE_ERR_GMRP_NOCONFIG
CLI_ERROR

4.2 smi_gmrp_msg.h File Reference

Defines data structures used by gmrp/mmrp SMI APIs. `#include "pal.h"`

```
#include "message.h"
#include "thread.h"
#include "network.h"
#include "log.h"
#include "tlv.h"
#include "syslog.h"
#include <sys/types.h>
#include "pal_types.h"
#include "pal_socket.h"
#include "prefix.h"
#include "smi_server.h"
```

Data Structures

- struct xmrpTimers
- struct xmrpBridgeConfig
- struct xmrpFsmState
- struct xmrpVlanStats
- struct xmrpCommonList
- struct gmrp_msg_

Defines

- #define **SMI_GMRP_EVENT** 1
- #define **SMI_GMRP_TIMER** 2
- #define **SMI_GMRP_CLI** 3
- #define **SMI_GMRP_PACKET** 4
- #define **SMI_GMRP_ALL** 5
- #define **GMRP_DEBUG_EVENT** 0x01
- #define **GMRP_DEBUG_CLI** 0x02
- #define **GMRP_DEBUG_TIMER** 0x04
- #define **GMRP_DEBUG_PACKET** 0x08
- #define **SMI_BRIDGE_MIN_VAL** 1
- #define **SMI_BRIDGE_MAX_VAL** 32
- #define **SMI_VLAN_ID_MIN_VAL** 2
- #define **SMI_VLAN_ID_MAX_VAL** 4094
- #define **SMI_XMRP_TIMER_MIN** 1
- #define **SMI_XMRP_TIMER_MAX** 0xffffffff

- #define **XMRP_TOTAL_ATTR_EVENTS** 7
- #define **SMI_GMRP_CTYPE_VR_ID** 0
- #define **SMI_GMRP_CTYPE_PROTOCOL** 1
- #define **SMI_GMRP_CTYPE_BRIDGE_NAME** 2
- #define **SMI_GMRP_CTYPE_VLAN_ID** 3
- #define **SMI_GMRP_CTYPE_IFNAME** 4
- #define **SMI_GMRP_CTYPE_TIMER_VALUE** 5
- #define **SMI_GMRP_CTYPE_REG_TYPE** 6
- #define **SMI_XMRP_CTYPE_SHOW_TIMERS** 7
- #define **SMI_XMRP_CTYPE_SHOW_BRIDGE** 8
- #define **SMI_XMRP_CTYPE_SHOW_FSM** 9
- #define **SMI_XMRP_CTYPE_SHOW_STATS** 10
- #define **SMI_GMRP_CTYPE_IS_GMRP_EN** 11
- #define **SMI_GMRP_CTYPE_TIMER_TYPE** 12
- #define **SMI_GMRP_CTYPE_IS_P2P** 13
- #define **SMI_GMRP_CTYPE_PERIODIC_TIMER_STATUS** 14
- #define **SMI_GMRP_CTYPE_REGIS_TYPE** 15
- #define **SMI_GMRP_CTYPE_FORWARD_ALL** 16
- #define **SMI_GMRP_CTYPE_EXTENDED_FILTERING_FLAG** 17
- #define **SMI_GMRP_CTYPE_SMI_DEBUG** 18
- #define **SMI_GMRP_CTYPE_EXTENDED_1** 31

Typedefs

- typedef struct gmrp_msg_ **gmrp_msg**

Enumerations

- enum **smi_gmrp_garp_timers** {

  SMI_GMRP_GARP_JOIN_TIMER = 0, **SMI_GMRP_GARP_LEAVE_-TIMER** = 1, **SMI_GMRP_GARP_LEAVE_ALL_TIMER** = 2, **SMI_-GMRP_GARP_LEAVE_CONF_TIMER** = 3,

  SMI_GMRP_GARP_LEAVEALL_CONF_TIMER = 4, **SMI_GMRP_-GARP_PERIODIC_TIMER** = 5, **SMI_GMRP_GARP_MAX_TIMERS** = 6 }

Functions

- int **smi_parse_gmrp** (u_char **pnt, u_int16_t *size, struct smi_msg_header *header, void *arg, SMI_CALLBACK callback)
- int **smi_encode_gmrp** (u_char **pnt, u_int16_t *size, gmrp_msg *msg)
- int **smi_decode_gmrp** (u_char **pnt, u_int16_t *size, gmrp_msg *msg)

4.2.1 Detailed Description

Defines data structures used by gmrp/mmrp SMI APIs.

Index