ZebOS-XP Static Route SMI Reference IP Infusion Inc.

Generated by Doxygen 1.6.1

Wed Dec 16 12:33:54 2015

Contents

1	File	Index			1	
	1.1	File Li	st		1	
2	2 File Documentation 3					
	2.1	smi_ri	b.h File Re	ference	3	
		2.1.1	Detailed	Description	7	
		2.1.2	Function	Documentation	7	
			2.1.2.1	smi_rib_debug	7	
			2.1.2.2	smi_rib_fib_retain_set	8	
			2.1.2.3	smi_rib_fib_retain_unset	8	
			2.1.2.4	smi_rib_get_vrf_details	9	
			2.1.2.5	smi_rib_get_vrf_given_gateway_sdkapi	9	
			2.1.2.6	smi_rib_ip_route_all_vrf_unset_sdkapi	10	
			2.1.2.7	smi_rib_ip_route_prefix_set_sdkapi	10	
			2.1.2.8	smi_rib_ip_route_prefix_unset_sdkapi	10	
			2.1.2.9	smi_rib_ip_route_unset_sdkapi	11	
			2.1.2.10	smi_rib_ipv4_route_set_sdkapi	11	
			2.1.2.11	smi_rib_ipv4_route_stale_clear	12	
			2.1.2.12	smi_rib_ipv4_route_unset_sdkapi	12	
			2.1.2.13	smi_rib_ipv4_route_vrf_ifname_set_sdkapi	13	
			2.1.2.14	smi_rib_ipv4_route_vrf_ifname_unset_sdkapi	13	
			2.1.2.15	smi_rib_ipv6_route_stale_clear	14	
			2.1.2.16	smi_rib_multipath_num_func_sdkapi	14	
			2.1.2.17	smi_rib_no_debug	14	
			2.1.2.18	smi_rib_set_maximum_fib_routes_sdkapi	15	
			2 1 2 19	smi rib set maximum static routes sdkapi	15	

ii CONTENTS

2.1.2.20	smi_rib_show_ipv4_route_details_sdkapi	16
2.1.2.21	smi_rib_show_ipv4_route_interface	16
2.1.2.22	smi_rib_show_ipv4_route_nhaddr	17
2.1.2.23	smi_rib_show_ipv6_route_details_sdkapi	17
2.1.2.24	smi_rib_show_ipv6_route_interface	18
2.1.2.25	smi_rib_show_ipv6_route_nhaddr	19
2.1.2.26	smi_rib_show_route_ipv4_sdkapi	19
2.1.2.27	smi_rib_show_route_ipv6_sdkapi	20
2.1.2.28	smi_rib_show_route_summary_ipv4	21
2.1.2.29	smi_rib_show_route_summary_ipv6	21
2.1.2.30	smi_rib_unset_maximum_fib_routes_sdkapi	22
2 1 2 31	smi rib unset maximum static routes sdkani	22

Chapter 1

File Index

1	1	1 '	Fil	ρ	T :	ict

Here is a list of all documented files with brief descriptions:	
smi_rib.h (Provides APIs for managing static_routes)	

2 File Index

Chapter 2

File Documentation

2.1 smi_rib.h File Reference

```
Provides APIs for managing static_routes. #include "smi_client.h"
#include "smi_rib_msg.h"
```

Functions

- int **smi_rib_multipath_num_func_sdkapi_validate** (struct smiclient_globals *azg, int set, int mutipathNum)
- int smi_rib_set_maximum_static_routes_sdkapi_validate (struct smiclient_globals *azg, int vrId, int num)
- int smi_rib_unset_maximum_static_routes_sdkapi_validate (struct smiclient_globals *azg, int vrId)
- int smi_rib_set_maximum_fib_routes_sdkapi_validate (struct smiclient_globals *azg, int vrId, int num)
- int smi_rib_unset_maximum_fib_routes_sdkapi_validate (struct smiclient_globals *azg, int vrId)
- int **smi_rib_fib_retain_set_validate** (struct smiclient_globals *azg, u_int32_t vrId, int retainTime)
- int **smi_rib_fib_retain_unset_validate** (struct smiclient_globals *azg, u_int32_t vrId)
- int **smi_rib_ipv4_route_stale_clear_validate** (struct smiclient_globals *azg, u int32 t vrId)
- int **smi_rib_ipv6_route_stale_clear_validate** (struct smiclient_globals *azg, u_int32_t vrId)
- int **smi_rib_ip_route_ifprefix_set_sdkapi_validate** (struct smiclient_globals *azg, int vrId, char *ipv4DestinationPrefix, char *ifname, int distanceValue, u_int32_t tagValue, char *staticRouteDescription)
- int **smi_rib_ip_route_ifprefix_unset_sdkapi_validate** (struct smiclient_globals *azg, int vrId, char *ipv4DestinationPrefix, char *ifname, int distance-Value, u_int32_t tagValue, char *staticRouteDescription)

• int **smi_rib_ip_route_prefix_set_sdkapi_validate** (struct smiclient_globals *azg, int vrId, char *ipv4DestinationPrefix, char *ipv4GatewayStr, int distance-Value, u_int32_t tagValue, char *staticRouteDescription)

- int **smi_rib_ip_route_prefix_unset_sdkapi_validate** (struct smiclient_globals *azg, int vrId, char *ipv4DestinationPrefix, char *ipv4GatewayStr, int distance-Value, u int32 t tagValue, char *staticRouteDescription)
- int **smi_rib_ip_route_unset_sdkapi_validate** (struct smiclient_globals *azg, int vrId, struct prefix_ipv4 *ipv4)
- int smi_rib_ip_route_all_vrf_unset_sdkapi_validate (struct smiclient_globals *azg, int vrId, struct prefix_ipv4 *ipv4)
- int smi_rib_ipv4_route_vrf_ifname_set_sdkapi_validate (struct smiclient_globals *azg, int vrId, char *vrfName, struct prefix_ipv4 *ipv4Prefix, struct pal_in4_addr *ipv4GateAddr, char *ifName, int distance, int metric, u_int32_t tag, char *desc)
- int smi_rib_ipv4_route_vrf_ifname_unset_sdkapi_validate (struct smiclient_globals *azg, int vrId, char *vrfName, struct prefix_ipv4 *ipv4Prefix, struct pal_in4_addr ipv4GateAddr, char *ifName, int distance, u_int32_t tag, char *desc)
- int smi_rib_ip_mroute_prefix_set_sdkapi_validate (struct smiclient_globals *azg, int vrId, char *vrfName, char *mrouteIpv4Prefix, char *mrouteGateStr, char *mrouteifname, char *routeType, int mrouteDistance)
- int **smi_rib_ip_mroute_prefix_unset_sdkapi_validate** (struct smiclient_globals *azg, int vr_id, char *vrfName, char *ipv4_prefix, char *route_type)
- int smi_rib_multipath_num_func_sdkapi (struct smiclient_globals *azg, int set, int mutipathNum)

Set multipath numbers installed to FIB.

int smi_rib_set_maximum_static_routes_sdkapi (struct smiclient_globals *azg, int vrId, int num)

Set maximum static routes number.

int smi_rib_unset_maximum_static_routes_sdkapi (struct smiclient_globals *azg, int vrId)

Set maximum static routes number to the default value 4294967294UL.

• int smi_rib_set_maximum_fib_routes_sdkapi (struct smiclient_globals *azg, int vrId, int num)

Set maximum fib routes number. Allowed number of fib routes excluding Kernel, Connect and Static.

int smi_rib_unset_maximum_fib_routes_sdkapi (struct smiclient_globals *azg, int vrId)

Set maximum fib routes number to the default value 4294967294.

• int smi_rib_fib_retain_set (struct smiclient_globals *azg, u_int32_t vrId, int retainTime)

Retain FIB for a specific time after RIB restarts.

- int smi_rib_fib_retain_unset (struct smiclient_globals *azg, u_int32_t vrId)

 Retain FIB retain time to default value.
- int smi_rib_ipv4_route_stale_clear (struct smiclient_globals *azg, u_int32_-t vrId)

Function to clear IPv4 stale kernel routes from NSM RIB and FIB.

int smi_rib_ipv6_route_stale_clear (struct smiclient_globals *azg, u_int32_-t vrId)

Function to clear IPv6 stale kernel routes from NSM RIB and FIB.

- int smi_rib_ip_route_prefix_set_sdkapi (struct smiclient_globals *azg, int vr_id, char *ipv4_prefix, char *gate_str, int distance, u_int32_t tag, char *desc)

 Function to configure ipv4 static route.
- int **smi_rib_ip_route_ifprefix_set_sdkapi** (struct smiclient_globals *azg, int vr_id, char *ipv4_prefix, char *ifname, int distance, u_int32_t tag, char *desc)
- int smi_rib_ip_route_prefix_unset_sdkapi (struct smiclient_globals *azg, int vr_id, char *ipv4_prefix, char *gate_str, int distance, u_int32_t tag, char *desc)

 Function to clear configured ipv4 static route.
- int **smi_rib_ip_route_ifprefix_unset_sdkapi** (struct smiclient_globals *azg, int vr_id, char *ipv4_prefix, char *ifname, int distance, u_int32_t tag, char *desc)
- int smi_rib_ipv4_route_set_sdkapi (struct smiclient_globals *azg, int vrId, char *vrfName, struct prefix_ipv4 *ipv4Prefix, struct pal_in4_addr *ipv4GateAddr, char *ifname, int distance, int metric, int snmpRouteType, u_int32_t tag, char *desc)

Establish ipv4 static routes into VRF.

• int smi_rib_ipv4_route_unset_sdkapi (struct smiclient_globals *azg, int vrId, char *vrfName, struct prefix_ipv4 *ipv4Prefix, struct pal_in4_addr *ipv4GateAddr, char *ifname, int distance, u_int32_t tag, char *desc)

clear ipv4 static routes into VRF

• int smi_rib_ipv4_route_vrf_ifname_set_sdkapi (struct smiclient_globals *azg, int vrld, char *vrfName, struct prefix_ipv4 *ipv4Prefix, struct pal_in4_addr *ipv4GateAddr, char *ifName, int distance, int metric, u_int32_t tag, char *desc)

Establish ipv4 static routes into VRF interfacename.

• int smi_rib_ipv4_route_vrf_ifname_unset_sdkapi (struct smiclient_globals *azg, int vrId, char *vrfName, struct prefix_ipv4 *ipv4Prefix, struct pal_in4_addr ipv4GateAddr, char *ifName, int distance, u_int32_t tag, char *desc)

clear ipv4 static routes into VRF interface name

• int smi_rib_ip_mroute_prefix_set_sdkapi (struct smiclient_globals *azg, int vrId, char *vrfName, char *mrouteIpv4Prefix, char *mrouteGateStr, char *mrouteifname, char *routeType, int mrouteDistance)

- int **smi_rib_ip_mroute_prefix_unset_sdkapi** (struct smiclient_globals *azg, int vr_id, char *vrfName, char *ipv4_prefix, char *route_type)
- int smi_rib_show_route_ipv6_sdkapi (struct smiclient_globals *azg, int startIndex, int endIndex, int vrId, char *vrfName, int database, u_char type, struct list *ribList, int(*callback)(struct list *ribList))

Function to establish IPV6 static routes into VRF.

• int smi_rib_show_route_ipv4_sdkapi (struct smiclient_globals *azg, int startIndex, int endIndex, int vrId, char *vrfName, int database, u_char type, struct list *ribList, int(*callback)(struct list *ribList))

Use this function to get all ipv4 routes from the routing table.

int smi_rib_show_ipv4_route_nhaddr (struct smiclient_globals *azg, int startIndex, int endIndex, char *vrfName, int vrId, char *ipAddr, struct list *ribList, int(*callback)(struct list *ribList))

Use this function to get all ipv4 routes from the routing table for specified next hop address.

• int smi_rib_show_ipv6_route_nhaddr (struct smiclient_globals *azg, int startIndex, int endIndex, char *vrfName, int vrId, char *ipAddr, struct list *ribList, int(*callback)(struct list *ribList))

Use this function to get all ipv6 routes from the routing table for specified next hop address.

• int smi_rib_show_ipv6_route_interface (struct smiclient_globals *azg, int startIndex, int endIndex, char *vrfName, int vrId, char *ifName, struct list *ribList, int(*callback)(struct list *ribList))

Use this function to get all ipv6 routes from the routing table for specified interface name.

• int smi_rib_show_ipv4_route_interface (struct smiclient_globals *azg, int startIndex, int endIndex, char *vrfName, int vrId, char *ifName, struct list *ribList, int(*callback)(struct list *ribList))

Use this function to get all routes from the routing table for specified interface name.

int smi_rib_get_vrf_details (struct smiclient_globals *azg, int startIndex, int endIndex, char *vrfName, int vrId, struct list *vrfList, int(*callback)(struct list *vrfList))

Use this function to get all VRF details.

• int smi_rib_show_ipv6_route_details_sdkapi (struct smiclient_globals *azg, int startIndex, int endIndex, char *vrfName, int vrId, char *ipAddr, struct list *ribList, int(*callback)(struct list *ribList))

Use this function to get ipv6 routing table for the specified ip address.

• int smi_rib_show_ipv4_route_details_sdkapi (struct smiclient_globals *azg, int startIndex, int endIndex, char *vrfName, int vrId, char *ipAddr, struct list *ribList, int(*callback)(struct list *ribList))

Use this function to get ipv4 routing table for the specified ip address.

• int smi_rib_show_route_summary_ipv4 (struct smiclient_globals *azg, char *vrfName, int vrId, struct smi_route_summ *rtSumm)

Function to get summary of all ipv4 routes.

• int smi_rib_show_route_summary_ipv6 (struct smiclient_globals *azg, char *vrfName, int vrId, struct smi_route_summ *rtSumm)

Function to get summary of all ipv6 routes.

• int smi_rib_get_vrf_given_gateway_sdkapi (struct smiclient_globals *azg, int vrId, char *str, vrf_id_t vrId, int family)

Get the vrfId given gateway(Gateway can be ifName or nh address).

• int smi_rib_ip_route_unset_sdkapi (struct smiclient_globals *azg, int vrId, char *vrfName, struct prefix_ipv4 *ipv4)

Function to clear configured ipv4 static route.

• int smi_rib_ip_route_all_vrf_unset_sdkapi (struct smiclient_globals *azg, int vrId, char *vrfName, struct prefix_ipv4 *ipv4)

Function to clear configured ipv4 static route.

- int smi_rib_debug (struct smiclient_globals *azg, int vrId, int debug)

 Use this function to specify the options for the displayed debugging information for RIP events, RIP packets and RIP NSM.
- int smi_rib_no_debug (struct smiclient_globals *azg, int vrId, int debug)

 Use this function to disable specific debugging.
- int **smi_rib_clear_ip_route_sdkapi** (struct smiclient_globals *azg, afi_t afi, int vrId, char *vrfName, char *prefixStr)

2.1.1 Detailed Description

Provides APIs for managing static_routes.

2.1.2 Function Documentation

2.1.2.1 int smi_rib_debug (struct smiclient_globals * azg, int vrId, int debug)

Use this function to specify the options for the displayed debugging information for RIP events, RIP packets and RIP NSM. smi_rib_debug

Parameters:

8

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← *debug* Pass debug flag as following:

SMI_RIB_DBG_ALL - Debug all RIB information

SMI_RIB_DBG_EVENTS - Debug RIB events

SMI_RIB_DBG_PACKET - Debug RIB and NSM communications

SMI_RIB_DBG_PACKET_SEND - Debug sent packets

SMI_RIB_DBG_PACKET_RECV - Debug received packets

SMI_RIB_DBG_PACKET_DETAIL - Display detailed information for the sent and received packet

SMI_RIB_DBG_PACKET_SEND_DETAIL - Display detailed information for the sent packet

SMI_RIB_DBG_PACKET_RECV_DETAIL - Display detailed information for the received packet

Returns:

0 on success, otherwise one of the following error codes RIB_API_SET_ERR_- $\mbox{VR_NOT_EXIST}$

2.1.2.2 int smi_rib_fib_retain_set (struct smiclient_globals * azg, u_int32_t vrId, int retainTime)

Retain FIB for a specific time after RIB restarts. smi_rib_fib_retain_set

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router ID
- ← *retainTime* Retain time range is <0-65535>. O means means FIB forever. In other case Specific time should be mentioned.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.3 int smi_rib_fib_retain_unset (struct smiclient_globals * azg, u_int32_t vrId)

Retain FIB retain time to default value. smi_rib_fib_retain_unset

Parameters:

← azg Pointer to the SMI client global structure

← vrId Virtual Router ID

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.4 int smi_rib_get_vrf_details (struct smiclient_globals * azg, int startIndex, int endIndex, char * vrfName, int vrId, struct list * vrfList, int(*)(struct list *vrfList) callback)

Use this function to get all VRF details. smi_rib_get_vrf_details

Parameters:

- ← azg Pointer to the SMI client global structure
- \leftarrow *startIndex* start index
- \leftarrow *endIndex* end index
- ← vrfName VRF Name. Pass NULL to get all VRF details in a list.
- ← vrId Virtual Router Id
- → *vrfList* Link list of structure smi_vrf_details. smi_vrf_details structure holds details of single VRF. List should be intialized by caller.
- → callback Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_vrf_details. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.5 int smi_rib_get_vrf_given_gateway_sdkapi (struct smiclient_globals * azg, int vrId, char * str, vrf_id_t vrfId, int family)

Get the vrfId given gateway(Gateway can be ifName or nh address). smi_rib_get_vrf_given_gateway_sdkapi

Parameters:

- \leftarrow azg Pointer to the SMI client global structure
- ← *vrId* virtual id [in] str Gateway string. (ifName or nh_adress) [out] vrfId Updated vrfId if found else not updated. [in] family AF_INET or AF_INET6

Returns:

RIB_API_SET_SUCCESS on success,otherwise one of the following error codes RIB_API_SET_ERR_VR_NOT_EXIST

2.1.2.6 int smi_rib_ip_route_all_vrf_unset_sdkapi (struct smiclient_globals * azg, int vrId, char * vrfName, struct prefix_ipv4 * ipv4)

Function to clear configured ipv4 static route. smi_rib_ip_route_all_vrf_unset_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← *ipv4* Ipv4 Address

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.7 int smi_rib_ip_route_prefix_set_sdkapi (struct smiclient_globals * azg, int vr_id, char * ipv4_prefix, char * gate_str, int distance, u_int32_t tag, char * desc)

Function to configure ipv4 static route. smi_rib_ip_route_prefix_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← ipv4Prefix Virtual Router Id
- ← *gateStr* Gateway address
- ← *distance* Distance
- $\leftarrow tag \text{ Tag}$
- $\leftarrow desc$ Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.8 int smi_rib_ip_route_prefix_unset_sdkapi (struct smiclient_globals * azg, int vr_id, char * ipv4_prefix, char * gate_str, int distance, u_int32_t tag, char * desc)

Function to clear configured ipv4 static route. smi_rib_ip_route_prefix_unset_sdkapi

Parameters:

← azg Pointer to the SMI client global structure

- ← vrId Virtual Router Id
- *← ipv4Prefix* Ipv4 Address
- ← *gateStr* Gateway address
- ← *distance* Distance
- $\leftarrow tag \text{ Tag}$
- \leftarrow desc Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.9 int smi_rib_ip_route_unset_sdkapi (struct smiclient_globals * azg, int vrId, char * vrfName, struct prefix_ipv4 * ipv4)

Function to clear configured ipv4 static route. smi_rib_ip_route_unset_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← *ipv4* Ipv4 Address

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.10 int smi_rib_ipv4_route_set_sdkapi (struct smiclient_globals * azg, int vrId, char * vrfName, struct prefix_ipv4 * ipv4Prefix, struct pal_in4_addr * ipv4GateAddr, char * ifname, int distance, int metric, int snmpRouteType, u_int32_t tag, char * desc)

Establish ipv4 static routes into VRF. smi_rib_ipv4_route_set_sdkapi

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← *vrfName* VRF name
- *← ipv4Prefix* Ipv4 Address
- ← *ipv4GateAddr* Gateway address
- *← ifName* Interface name
- ← *distance* Distance

- \leftarrow *metric* Metric
- $\leftarrow tag \text{ Tag}$
- ← desc Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.11 int smi_rib_ipv4_route_stale_clear (struct smiclient_globals * azg, u_int32_t vrld)

Function to clear IPv4 stale kernel routes from NSM RIB and FIB. smi_rib_ipv4_route_stale_clear

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.12 int smi_rib_ipv4_route_unset_sdkapi (struct smiclient_globals * azg, int vrId, char * vrfName, struct prefix_ipv4 * ipv4Prefix, struct pal_in4_addr * ipv4GateAddr, char * ifname, int distance, u_int32_t tag, char * desc)

clear ipv4 static routes into VRF smi_rib_ipv4_route_unset_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← vrfId VRF name
- *← ipv4Prefix* Ipv4 Address
- ← *ipv4GateAddr* Gateway address
- ← *ifName* Interface name
- \leftarrow *distance* Distance
- ← *metric* Metric
- *← tag* Tag
- \leftarrow desc Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.13 int smi_rib_ipv4_route_vrf_ifname_set_sdkapi (struct smiclient_globals * azg, int vrId, char * vrfName, struct prefix_ipv4 * ipv4Prefix, struct pal_in4_addr * ipv4GateAddr, char * ifName, int distance, int metric, u_int32_t tag, char * desc)

Establish ipv4 static routes into VRF interfacename. smi_rib_ipv4_route_vrf_ifname_set_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← *vrfName* VRF name
- *← ipv4Prefix* Ipv4 Address
- ← *ipv4GateAddr* Gateway address
- *← ifName* Interface name
- ← *distance* Distance
- ← *metric* Metric
- $\leftarrow tag \text{ Tag}$
- \leftarrow desc Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.14 int smi_rib_ipv4_route_vrf_ifname_unset_sdkapi (struct smiclient_globals * azg, int vrId, char * vrfName, struct prefix_ipv4 * ipv4Prefix, struct pal_in4_addr ipv4GateAddr, char * ifName, int distance, u_int32_t tag, char * desc)

clear ipv4 static routes into VRF interface name smi_rib_ipv4_route_vrf_ifname_unset_sdkapi

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← *vrfName* VRF name
- *← ipv4Prefix* Ipv4 Address
- ← *ipv4GateAddr* Gateway address (Pass null in case of no gateway address)
- ← *ifName* Interface name
- ← *distance* Distance
- ← *metric* Metric

```
\leftarrow tag Tag \leftarrow desc Description
```

Returns:

 $SMI_SUCCESS$ on success, otherwise one of the following error codes SMI_ERROR

2.1.2.15 int smi_rib_ipv6_route_stale_clear (struct smiclient_globals * azg, u_int32_t vrId)

Function to clear IPv6 stale kernel routes from NSM RIB and FIB. smi_rib_ipv6_route_stale_clear

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.16 int smi_rib_multipath_num_func_sdkapi (struct smiclient_globals * azg, int set, int mutipathNum)

Set multipath numbers installed to FIB. smi_rib_multipath_num_func_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← set Flag to determine set or unset. Pass O to set multipath number to default
- ← *mutipathNum* Number of multipath to be set

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.17 int smi_rib_no_debug (struct smiclient_globals * azg, int vrId, int debug)

Use this function to disable specific debugging. smi_rib_no_debug

Parameters:

← azg Pointer to the SMI client global structure

- ← vrId Virtual Router Id
- ← *debug* Pass debug flag as following:

SMI_RIB_DBG_ALL - Debug all RIB information

SMI_RIB_DBG_EVENTS - Debug RIB events

SMI_RIB_DBG_PACKET - Debug RIB and NSM communications

SMI RIB DBG PACKET SEND - Debug sent packets

SMI_RIB_DBG_PACKET_RECV - Debug received packets

SMI_RIB_DBG_PACKET_DETAIL - Display detailed information for the sent and received packet

SMI_RIB_DBG_PACKET_SEND_DETAIL - Display detailed information for the sent packet

SMI_RIB_DBG_PACKET_RECV_DETAIL - Display detailed information for the received packet

Returns:

0 on success, otherwise one of the following error codes RIB_API_SET_ERR_- VR_NOT_EXIST

2.1.2.18 int smi_rib_set_maximum_fib_routes_sdkapi (struct smiclient_globals * azg, int vrId, int num)

Set maximum fib routes number. Allowed number of fib routes excluding Kernel, Connect and Static. smi_rib_set_maximum_fib_routes_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router ID
- ← *num* Fib routes number to be set. Range <1-4294967294>

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.19 int smi_rib_set_maximum_static_routes_sdkapi (struct smiclient_globals * azg, int vrId, int num)

Set maximum static routes number. smi_rib_set_maximum_static_routes_sdkapi

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router ID

← *num* Number of the static route which can be configured <1-4294967294>

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.20 int smi_rib_show_ipv4_route_details_sdkapi (struct smiclient_globals * azg, int startIndex, int endIndex, char * vrfName, int vrId, char * ipAddr, struct list * ribList, int(*)(struct list * ribList) callback)

Use this function to get ipv4 routing table for the specified ip address. smi_rib_show_ipv4_route_details_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- \leftarrow *startIndex* start index
- \leftarrow *endIndex* end index
- ← *vrfName* VRF Name. Pass NULL for default VRF.
- ← vrId Virtual Router Id
- $\leftarrow ipAddr$ ipv4 address
- → *ribList* Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.
- → callback Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

 $SMI_SUCCESS$ on success, otherwise one of the following error codes SMI_ERROR

2.1.2.21 int smi_rib_show_ipv4_route_interface (struct smiclient_globals * azg, int startIndex, int endIndex, char * vrfName, int vrId, char * ifName, struct list * ribList, int(*)(struct list * ribList) callback)

Use this function to get all routes from the routing table for specified interface name. smi_rib_show_ipv4_route_interface

- ← azg Pointer to the SMI client global structure
- ← *startIndex* start index
- \leftarrow *endIndex* end index

- ← vrfName VRF Name. Pass NULL for default VRF.
- ← vrId Virtual Router Id
- *← ifName* Interface Name
- → ribList Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.
- → *callback* Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.22 int smi_rib_show_ipv4_route_nhaddr (struct smiclient_globals * azg, int startIndex, int endIndex, char * vrfName, int vrId, char * ipAddr, struct list * ribList, int(*)(struct list * ribList) callback)

Use this function to get all ipv4 routes from the routing table for specified next hop address. smi_rib_show_ipv4_route_nhaddr

Parameters:

- \leftarrow azg Pointer to the SMI client global structure
- ← startIndex start index
- \leftarrow *endIndex* end index
- ← vrfName VRF Name. Pass NULL for default VRF.
- ← vrId Virtual Router Id
- ← *ipAddr* Next hop in the IPV4 routing table to display
- → *ribList* Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.
- → *callback* Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.23 int smi_rib_show_ipv6_route_details_sdkapi (struct smiclient_globals * azg, int startIndex, int endIndex, char * vrfName, int vrId, char * ipAddr, struct list * ribList, int(*)(struct list * ribList) callback)

Use this function to get ipv6 routing table for the specified ip address. smi_rib_show_ipv6_route_details_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- \leftarrow *startIndex* start index
- \leftarrow *endIndex* end index
- ← *vrfName* VRF Name. Pass NULL for default VRF.
- ← vrId Virtual Router Id
- $\leftarrow ipAddr$ ipv6 address
- → *ribList* Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.
- → *callback* Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.24 int smi_rib_show_ipv6_route_interface (struct smiclient_globals * azg, int startIndex, int endIndex, char * vrfName, int vrId, char * ifName, struct list * ribList, int(*)(struct list * ribList) callback)

Use this function to get all ipv6 routes from the routing table for specified interface name. smi_rib_show_ipv6_route_interface

Parameters:

- ← azg Pointer to the SMI client global structure
- ← *startIndex* start index
- \leftarrow *endIndex* end index
- ← vrfName VRF Name. Pass NULL for deafult VRF
- ← vrId Virtual Router Id
- ← *ifname* Interface Name
- → *ribList* Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.
- → *callback* Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.25 int smi_rib_show_ipv6_route_nhaddr (struct smiclient_globals * azg, int startIndex, int endIndex, char * vrfName, int vrId, char * ipAddr, struct list * ribList, int(*)(struct list * ribList) callback)

Use this function to get all ipv6 routes from the routing table for specified next hop address. smi_rib_show_ipv6_route_nhaddr

Parameters:

- ← azg Pointer to the SMI client global structure
- \leftarrow *startIndex* start index
- \leftarrow *endIndex* end index
- ← vrfName VRF Name. Pass NULL for default VRF.
- ← vrId Virtual Router Id
- ← *ipAddr* Next hop in the IPV6 routing table to display
- → *ribList* Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.
- → callback Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.26 int smi_rib_show_route_ipv4_sdkapi (struct smiclient_globals * azg, int startIndex, int endIndex, int vrId, char * vrfName, int database, u_char type, struct list * ribList, int(*)(struct list * ribList) callback)

Use this function to get all ipv4 routes from the routing table. smi_rib_show_route_ipv4_sdkapi

- ← azg Pointer to the SMI client global structure
- ← startIndex start index
- \leftarrow *endIndex* end index
- ← vrId Virtual Router Id
- ← vrfName VRF Name. Pass NULL to get all VRF details in a list.
- ← *database* To Display IPv6 routing table database information pass '1', else '0'
- ← type Type of the route to be fetched <1-9> 1-IPI_ROUTE_KERNEL, 2-IPI_ROUTE CONNECT
 - 3-IPI_ROUTE_STATIC, 4-IPI_ROUTE_RIP
 - 5-IPI_ROUTE_RIPNG, 6-IPI_ROUTE_OSPF
 - 7-IPI_ROUTE_OSPF6, 8-IPI_ROUTE_BGP
 - 9-IPI_ROUTE_ISIS

→ *ribList* Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ *callback* Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

 $SMI_SUCCESS$ on success, otherwise one of the following error codes SMI_ERROR

2.1.2.27 int smi_rib_show_route_ipv6_sdkapi (struct smiclient_globals * azg, int startIndex, int endIndex, int vrId, char * vrfName, int database, u_char type, struct list * ribList, int(*)(struct list * ribList) callback)

Function to establish IPV6 static routes into VRF. smi_ipv6_route_vrf_ifname_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router Id
- ← vrfName VRF Name
- ← *ipv6Prefix* Ipv6 Address
- ← gateStr Gateway address
- ← *distance* Distance

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

smi_rib_show_route_ipv6_sdkapi

Use this function to get all ipv6 routes from the routing table.

- ← azg Pointer to the SMI client global structure
- \leftarrow *startIndex* start index
- \leftarrow *endIndex* end index
- ← vrId Virtual Router Id
- ← vrfName VRF Name. Pass NULL to get all VRF details in a list.
- \leftarrow *database* To Display IPv6 routing table database information pass '1', else pass '0'

- ← type Type of the route to be fetched <1-9> 1-IPI_ROUTE_KERNEL, 2-IPI_ROUTE_CONNECT
 - 3-IPI_ROUTE_STATIC, 4-IPI_ROUTE_RIP
 - 5-IPI ROUTE RIPNG, 6-IPI ROUTE OSPF
 - 7-IPI_ROUTE_OSPF6, 8-IPI_ROUTE_BGP
 - 9-IPI ROUTE ISIS
- → *ribList* Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.
- → *callback* Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.28 int smi_rib_show_route_summary_ipv4 (struct smiclient_globals * azg, char * vrfName, int vrId, struct smi_route_summ * rtSumm)

Function to get summary of all ipv4 routes. smi_rib_show_route_summary_ipv4

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrfName VRF Name. Pass NULL for default VRF.
- ← vrId Virtual Router Id
- → rtSumm Output parameter to hold the route summary smi route summ.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.29 int smi_rib_show_route_summary_ipv6 (struct smiclient_globals * azg, char * vrfName, int vrId, struct smi_route_summ * rtSumm)

Function to get summary of all ipv6 routes. smi_rib_show_route_summary_ipv6

- ← azg Pointer to the SMI client global structure
- ← vrfName VRF Name. Pass NULL for default VRF.
- ← *vrId* Virtual Router Id
- → *rtSumm* Output parameter to hold the route summary smi_route_summ.

Returns:

 $SMI_SUCCESS$ on success, otherwise one of the following error codes SMI_ERROR

2.1.2.30 int smi_rib_unset_maximum_fib_routes_sdkapi (struct smiclient_globals * azg, int vrId)

Set maximum fib routes number to the default value 4294967294. smi_rib_unset_maximum_fib_routes_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← vrId Virtual Router ID

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.31 int smi_rib_unset_maximum_static_routes_sdkapi (struct smiclient_globals * azg, int vrId)

Set maximum static routes number to the default value 4294967294UL. smi_rib_unset_maximum_static_routes_sdkapi

Parameters:

- ← azg Pointer to the SMI client global structure
- ← *vrId* Virtual Router ID

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

Index

mi_rib	.h, 3	smi_rib_show_ipv6_route_nhaddr
sm	i_rib_debug, 7	18
sm	ni_rib_fib_retain_set, 8	smi_rib_show_route_ipv4_sdkapi,
	i_rib_fib_retain_unset, 8	19
	ni_rib_get_vrf_details, 9	smi_rib_show_route_ipv6_sdkapi,
	ii_rib_get_vrf_given_gateway	20
	sdkapi, 9	smi_rib_show_route_summary
sm	ni_rib_ip_route_all_vrf_unset	ipv4, 21
	sdkapi, 9	smi_rib_show_route_summary
sm	ni_rib_ip_route_prefix_set_sdkapi,	ipv6, 21
	10	smi_rib_unset_maximum_fib
sm	ni_rib_ip_route_prefix_unset	routes_sdkapi, 22
	sdkapi, 10	smi_rib_unset_maximum_static
sm	ni_rib_ip_route_unset_sdkapi, 11	routes_sdkapi, 22
sm	ni_rib_ipv4_route_set_sdkapi, 11	smi_rib_debug
sm	ni_rib_ipv4_route_stale_clear, 12	smi_rib.h, 7
sm	ni_rib_ipv4_route_unset_sdkapi,	smi_rib_fib_retain_set
	12	smi_rib.h, 8
sm	ni_rib_ipv4_route_vrf_ifname	smi_rib_fib_retain_unset smi_rib.h, 8
	set_sdkapi, 12	smi_rib_get_vrf_details
sm	ni_rib_ipv4_route_vrf_ifname	smi_rib_get_vii_detans smi_rib.h, 9
	unset_sdkapi, 13	smi_rib_get_vrf_given_gateway_sdkap
sm	ni_rib_ipv6_route_stale_clear, 14	smi_rib.h, 9
sm	ni_rib_multipath_num_func	smi_rib_ip_route_all_vrf_unset_sdkapi
	sdkapi, 14	smi_rib.h, 9
	ni_rib_no_debug, 14	smi_rib_ip_route_prefix_set_sdkapi
sm	ni_rib_set_maximum_fib_routes	smi_rib.h, 10
	sdkapi, 15	smi_rib_ip_route_prefix_unset_sdkapi
sm	ni_rib_set_maximum_static	smi_rib.h, 10
	routes_sdkapi, 15	smi_rib_ip_route_unset_sdkapi
sm	ni_rib_show_ipv4_route_details	smi_rib.h, 11
	sdkapi, 16	smi_rib_ipv4_route_set_sdkapi
sm	ni_rib_show_ipv4_route_interface,	smi_rib.h, 11
	16	smi_rib_ipv4_route_stale_clear
sm	ni_rib_show_ipv4_route_nhaddr,	smi_rib.h, 12
		smi_rib_ipv4_route_unset_sdkapi
sm	ni_rib_show_ipv6_route_details sdkapi, 17	smi_rib.h, 12
0.000	±	smi_rib_ipv4_route_vrf_ifname_set
SIII	ii_rib_show_ipv6_route_interface, 18	sdkapi
	10	smi_rib.h, 12

24 INDEX

```
smi_rib_ipv4_route_vrf_ifname_unset_-
         sdkapi
    smi_rib.h, 13
smi_rib_ipv6_route_stale_clear
    smi_rib.h, 14
smi_rib_multipath_num_func_sdkapi
    smi_rib.h, 14
smi_rib_no_debug
    smi_rib.h, 14
smi_rib_set_maximum_fib_routes_-
         sdkapi
    smi_rib.h, 15
smi_rib_set_maximum_static_routes_-
         sdkapi
    smi_rib.h, 15
smi_rib_show_ipv4_route_details_sdkapi
    smi_rib.h, 16
smi_rib_show_ipv4_route_interface
    smi_rib.h, 16
smi_rib_show_ipv4_route_nhaddr
    smi rib.h, 17
smi_rib_show_ipv6_route_details_sdkapi
    smi_rib.h, 17
smi_rib_show_ipv6_route_interface
    smi_rib.h, 18
smi\_rib\_show\_ipv6\_route\_nhaddr
    smi_rib.h, 18
smi_rib_show_route_ipv4_sdkapi
    smi_rib.h, 19
smi_rib_show_route_ipv6_sdkapi
    smi_rib.h, 20
smi_rib_show_route_summary_ipv4
    smi_rib.h, 21
smi_rib_show_route_summary_ipv6
    smi_rib.h, 21
smi_rib_unset_maximum_fib_routes_-
         sdkapi
    smi_rib.h, 22
smi rib unset maximum static routes -
         sdkapi
    smi_rib.h, 22
```