

Interface

Created by Takashi Higashimura, last modified just a moment ago

- [Referenced YANG models](#)
 - [IETF](#)
 - [OpenConfig](#)
 - [XE](#)
 - [XR](#)
 - [NX](#)
- [Structure Hierarchy](#)
- [Interface Conf Structure](#)
- [Interface Ops structure](#)
 - [show commands](#)
 - [Ops structure](#)

Referenced YANG models

IETF

<https://tools.ietf.org/html/rfc7223>

OpenConfig

<https://github.com/openconfig/public/blob/master/release/models/interfaces/openconfig-interfaces.yang>

XE

<https://github.com/YangModels/yang/blob/master/vendor/cisco/xe/1651/Cisco-IOS-XE-interfaces.yang>

XR

<https://github.com/YangModels/yang/blob/master/vendor/cisco/xr/621/cisco-xr-openconfig-interfaces-deviations.yang>

<https://github.com/YangModels/yang/blob/master/vendor/cisco/xr/621/cisco-xr-openconfig-interfaces-types.yang>

NX

<https://github.com/YangModels/yang/blob/master/vendor/cisco/nx/7.0-3-l6-1/ietf-interfaces.yang>

Structure Hierarchy

```

Interface
  +-- PysicalInterface
  |   +-- EthernetInterface
  |       +-- ServiceInstance
  |       +-- LagMemberInterface
  +-- VirtualInterface
      +-- SubInterface
      |   +-- ServiceInstance
      +-- LoopbackInterface
      +-- VlanInterface
      +-- Nve
      |   +-- Vni
      +-- LagInterface

```

NOTE: Some type of interfaces such as PortChannel, Nve will be covered in the future when related future's structure is created.

Interface Conf Structure

	XE	XR	NX
Interface	<pre> <interface> (config)# interface <interface> (config-if)# </pre>	<pre> <interface>,<l2transport> : Bool (config)# interface <interface> [l2transport] (config-if)# </pre>	<pre> <interface> (config)# interface <interface> (config-if)# </pre>

	XE	XR	NX
description -Str	-if)#description <description> #show interfaces Description: desc	-if)#description <description> #show interfaces detail Description: desc	-if)# description <description> # show interface Description: desc
enabled -True False	True -if)#no shutdown False -if)#shutdown #show interfaces GigabitEthernet1 is administratively down , line protocol is down	True -if)#no shutdown False -if)#shutdown #show interfaces detail GigabitEthernet0/0/0/0 is administratively down , line protocol is administratively down	True -if)#no shutdown False -if)#shutdown # show interface thernet2/1 is down (Administratively down)
link_up_down_trap_enable -True False	-if)#snmp trap link-status No show cmd	N/A (XR can configure this only in global config mode)	-if)# snmp trap link-status No show cmd
mtu -Int	-if)#mtu <mtu> #show interfaces MTU 1600 bytes, BW 768 Kbit/sec, DLY 100 usec,	-if)#mtu <mtu> #show interfaces detail MTU 1600 bytes, BW 768 Kbit (Max: 1000000 Kbit)	-if)# mtu <mtu> # show interface MTU 1600 bytes, BW 768 Kbit, DLY 10 usec
vrf -Str vrf_downstream (XE only) -Str	-if)#vrf forwarding <vrf> [downstream <vrf_downstream>] #show vrf detail VRF VRF1 (VRF Id = 1); default RD <not set>; default VPNID <not set> Interfaces: Gi1	-if)#vrf <vrf> #show vrf all detail VRF VRF1 ; RD not set; VPN ID not set Interfaces: GigabitEthernet0/0/0/0	-if)# vrf member <vrf> # show vrf all interface Interface VRF-Name VRF-ID Site-of-Origin Ethernet2/1 VRF1 3 --

	XE	XR	NX
mac_address -Str ('aaaa.bbbb.cccc')	<pre>-if)#mac-address <mac_address> #show interfaces Hardware is CSR vNIC, address is aaaa.bbbb.cccc (bia 5254.0043.7ea5)</pre>	<pre>-if)#mac-address <mac_address> #show interfaces detail Hardware is GigabitEthernet, address is aaaa.bbbb.cccc (bia 5254.0019.877b)</pre>	<pre>-if)# mac-address <mac_address> # show interface Hardware: 10/100/1000 Ethernet, address: aaaa.bbbb.cccc (bia 5254.00dc.b48d)</pre>
bandwidth -Int (Kbit/sec)	<pre>-if)#bandwidth <bandwidth> #show interfaces MTU 1600 bytes, BW 768 Kbit/sec, DLY 100 usec,</pre>	<pre>-if)#bandwidth <bandwidth> #show interfaces detail MTU 1600 bytes, BW 768 Kbit (Max: 1000000 Kbit)</pre>	<pre>-if)#bandwidth <bandwidth> # show interface MTU 1600 bytes, BW 768 Kbit, DLY 10 usec</pre>
link_status -True False	<pre>-if)#logging event link-status No show cmd</pre>	N/A (XR can configure this only in global config mode)	<pre>-if)# logging event port link-status No show cmd</pre>
load_interval -Int (seconds)	<pre>-if)#load-interval <load-interval> Router#show interfaces 30 second input rate 0 bits/sec, 0 packets/sec 30 second output rate 0 bits/sec, 0 packets/sec</pre>	<pre>-if)#load-interval 30 #show interfaces detail 30 second input rate 0 bits/sec, 0 packets/sec 30 second output rate 0 bits/sec, 0 packets/sec</pre>	<pre>-if)# load-interval 30 # show interface 30 seconds input rate 0 bits/sec, 0 packets/sec 30 seconds output rate 0 bits/sec, 0 packets/sec</pre>

	XE	XR	NX
encapsulation -Enum('dot1q') first_dot1q -Str second_dot1q -Str native_vlan_dot1q -True False	<pre>-if)#encapsulation <encapsulation> <first_dot1q> [[native] second-dot1q <second_dot1q>] #show interfaces gi1.10 Encapsulation 802.1Q Virtual LAN, Vlan ID 20.</pre>	<pre>-subif)#encapsulation <encapsulation> <first_dot1q> [second-dot1q <second_dot1q>] -if)#dot1q native vlan <native_vlan> #show vlan interface Interface Encapsulation Outer 2nd Service MTU LineP VLAN VLAN State Gi0/0/0/0.10 Double 802.1Q 10 10 L3 1608 admin-down</pre>	<pre>-if)#encapsulation <encapsulation> <first_dot1q> [native] # show interface Encapsulation 802.1Q Virtual LAN, Vlan ID 10, medium is broadcast</pre>
<p>#This config can be multiple lines</p> ipv4prefix -Str # '192.168.1.1/24' ipv4_secondary -Boolean route_tag -Str secondary_vrf -Str	<pre>-if)# ip address <ipv4> <prefix-length> [sedondary [vrf <secondary_vrf>]] <ipv4> : from <ipv4prefix> <prefix-length> : from <ipv4prefix> <prefix-length> : 1-32 (needs to be converted to 255.255.255.255 format) #show ip interface Internet address is 10.1.1.1/24 Secondary address 10.2.2.2/24</pre>	<pre>-if)# ipv4 address <ipv4prefix> [[sedondary] [route-tag <route_tag>]] [secondary route-tag <route_tag>] ipv4_secondary = '10.2.2.2/24': -if)# ipv4 address 10.2.2.2/24 secondary #show ipv4 vrf all interface Internet address is 10.1.1.1/24 Secondary address 10.2.2.2/24</pre>	<pre>-if)# ip address <ipv4prefix> [secondary] [tag <route-tag>] # show ip interface vrf all IP address: 10.4.4.4, IP subnet: 10.4.4.0/24 IP address: 10.2.2.2, IP subnet: 10.2.2.0/24 secondary IP address: 10.3.3.3, IP subnet: 10.3.3.0/24 secondary R1# show routing vrf all 192.168.1.0/24, ubest/mbest: 1/0, attached *via 192.168.1.2, Eth1/1, [0/0], 00:08:44, direct, tag 222</pre>

	XE	XR	NX
dhcp -True False dhcp_client_id -Str dhcp_hostname	-if)# ip address dhcp [client-id <dhcp_client_id> [hostname <dhcp_hostname>] #show ip interface Internet address will be negotiated using DHCP	N/A	-if)# ip address dhcp # show ip interface vrf all
unnumbered_intf_ref -Str	-if)#ip unnumbered <unnumbered_intf_ref> #show interfaces gi1.20 Interface is unnumbered . Using address of Loopback0 (11.11.11.11)	-if)#ipv4 unnumbered <unnumbered_intf_ref> #show ipv4 vrf all interface Interface is unnumbered . Using address of Loopback0 (2.2.2.2/32)	-if)# ip unnumbered <unnumbered_intf_ref> # show ip interface vrf all IP unnumbered interface (loopback0)

	XE	XR	NX
<p>#This config can be multiple lines</p> <p>ipv6prefix</p> <p>-Str # '2001:db8:1::1/64'</p> <p>ipv6_anycast</p> <p>-True False</p> <p>ipv6_eui_64</p> <p>-True False</p> <p>ipv6_route_tag</p> <p>-Str</p>	<pre>-if)#ipv6 address <ipv6prefix> [anycast eui-64] #show ipv6 interface 2001:DB8:1:1::1, subnet is 2001:DB8:1:1::/64 [TEN] 2001:DB8:3:3::3, subnet is 2001:DB8:3:3::/64 [ANY/TEN] 2001:DB8:4:4:A8AA:BBFF:FEB B:CCCC, subnet is 2001:DB8:4:4::/64 [EUI/TEN]</pre>	<pre>if)#ipv6 address <ipv6prefix> [eui-64] [route- tag <ipv6_route_tag> #show ipv6 vrf all interface 2001:db8:1:1::1, subnet is 2001:db8:1:1::/64 [TENTATIVE] 2001:db8:4:4::4, subnet is 2001:db8:4:4::/64 [TENTATIVE] with route-tag 10 2001:db8:3:3:a8aa:bbff:fe bb:cccc, subnet is 2001:db8:3:3::/64 [TENTATIVE]</pre>	<pre>-if)# ipv6 address <ipv6prefix> [anycast eui64] [tag <ipv6_route_tag>] # show ipv6 interface vrf all IPv6 address: 2001:db8:1:1::1/64 [VALID] 2001:db8:3:3:3/64 [VALID] 2001:db8:4:4:a8aa:bbff: febb:cccc/64 [VALID] 2001:db8:2:2:2/64 [VALID] IPv6 subnet: 2001:db8:1:1::/64 Anycast configured addresses: 2001:db8:2:2:2/64 [VALID] IPv6 link-local address: fe80::a8aa:bbff:febb:cccc (default) [VALID] #show routing ipv6 vrf all 2001:db8:2:2::/64, ubest/mbest: 1/0, attached *via 2001:db8:2:2::2, Eth1/1, [0/0], 00:00:22, direct, , tag 222</pre>
<p>ipv6_unnumbered_intf_ref</p> <p>-Str</p>	<pre>-if)#ipv6 unnumbered <ipv6_unnumbered_intf_ref> #show ipv6 interface Interface is unnumbered. Using address of Loopback0</pre>	N/A	N/A

	XE	XR	NX
ipv6_enabled -True False	True -if)#ipv6 enable #show ipv6 interface GigabitEthernet1 is administratively down, line protocol is down	True -if)#ipv6 enable #show ipv6 vrf all interface Loopback0 is Up, ipv6 protocol is Up, Vrfid is default (0x60000000)	N/A # show ipv6 interface vrf all Ethernet2/1 , Interface status: protocol-up/link- up/admin-up, iod: 36
ipv6_autoconf -True False ipv6_autoconf_default -True False	-if)#ipv6 address autoconfig [default] #show ipv6 interface Stateless address autoconfig enabled	-if)# ipv6 address autoconfig No show cmd	-if)# ipv6 address autoconfig [default] No show cmd
switchport_enable -Boolean	switchport #show interfaces switchport Switchport: Enabled	N/A	switchport # show interface switchport Switchport: Enabled
switchport_mode -Enum('access','dot1q-tunnel','fex- fabric','private-vlan','trunk')	switchport mode <switchport_mode> #show interfaces switchport Switchport: Enabled Administrative Mode: access	N/A	switchport mode <switchport_mode> # show interface switchport Switchport: Enabled Operational Mode: trunk
medium -Enum('p2p','broadcast')	-if)#medium p2p No show cmd	if)#ipv4 point-to-point No show cmd	-if)# medium p2p # show interface Encapsulation ARPA, medium is broadcast
delay -Int	-if)#delay 10 #show interfaces MTU 1600 bytes, BW 768 Kbit/sec, DLY 100 usec,	N/A	-if)# delay 10 # show interface MTU 1600 bytes, BW 768 Kbit, DLY 3330 usec

	XE	XR	NX
(Vxlan) evpn_multisite_fabric_tracking - Bool			-if)# evpn multisite fabric-tracking # show nve multisite fabric-links Interface State ----- Ethernet1/53 Up
(Vxlan) evpn_multisite_dci_tracking - Bool			-if)# evpn multisite dci-tracking # show nve multisite dci-links Interface State ----- Ethernet1/50 Up
(Vxlan) ip_forward - Bool			-if)# ip forward No show cmd (show run only)
Interface +-- PhysicalInterface			
port_speed -Str ((Enum('10','100','1000','10000','100000','40000','auto')))	-if)#speed <port_speed> #show interfaces Full Duplex, 1000 Mbps, link type is force-up, media type is RJ45	-if)#speed <port_speed> #show interfaces detail Full-duplex, 1000 Mb/s, unknown, link type is force-up	-if)# speed <port_speed> # show interface full-duplex, 1000 Mb/s
Interface +-- PhysicalInterface +-- EthernetInterface			

	XE	XR	NX
access_vlan -Str	-if)# switchport access vlan <access_vlan> #show interfaces switchport Administrative Mode: access Access Mode VLAN: 100	N/A	-if)# switchport access vlan <access_vlan> # show interface switchport Operational Mode: access Access Mode VLAN: 100 (Vlan not created)
trunk_vlans -Str	-if)# switchport trunk allowed vlan <trunk_vlans> #show interfaces switchport Administrative Mode: trunk Trunking VLANs Enabled: ALL	N/A	-if)# switchport trunk allowed vlan <trunk_vlans> # show interface switchport Operational Mode: trunk Trunking VLANs Allowed: 100,200,300
trunk_add_vlans -Str	-if)# switchport trunk allowed vlan add <trunk_add_vlans> #show interfaces switchport Administrative Mode: trunk Trunking VLANs Enabled: ALL	N/A	-if)# switchport trunk allowed vlan add <trunk_add_vlans> # show interface switchport Operational Mode: trunk Trunking VLANs Allowed: 100,200,300
trunk_remove_vlans	-if)# switchport trunk allowed vlan remove <trunk_remove_vlans> #show interfaces switchport Administrative Mode: trunk Trunking VLANs Enabled: ALL	N/A	-if)# switchport trunk allowed vlan remove <trunk_remove_vlans> # show interface switchport Operational Mode: trunk Trunking VLANs Allowed: 100,200,300

	XE	XR	NX
native_vlan -Str	-if)#switchport trunk native vlan <native_vlan> #show interfaces switchport Trunking Native Mode VLAN: 1 (default)	N/A	-if)#switchport trunk native vlan <native_vlan> # show interface switchport Trunking Native Mode VLAN: 1 (default)
auto_negotiate -True False	-if)#negotiation auto #show interfaces Full Duplex, 1000Mbps, link type is force-up , media type is RJ45 #show interfaces Full Duplex, 1000Mbps, link type is auto , media type is RJ45	-if)#negotiation auto #show interfaces Full-duplex, 1000Mb/s, unknown, link type is autonegotiation	-if)# speed auto -if)# duplex auto # show interface e4/1 Auto-Negotiation is turned off OR Auto-Negotiation is turned on
duplex_mode -'full' 'half'	-if)# duplex <duplex_mode> #show interfaces Full Duplex, 1000Mbps, link type is force-up, media type is RJ45	-if)#duplex <duplex_mode> #show interfaces Full -duplex, 1000Mb/s, unknown, link type is autonegotiation	-if)# duplex <duplex_mode> # show interface full -duplex, 1000 Mb/s
flow_control_receive -True False	-if)# flowcontrol receive [on off] #show interfaces output flow-control is unsupported, input flow-control is unsupported	-if)#flow-control ingress	-if)# flowcontrol receive [on off] # show interface Input flow-control is off , output flow-control is off
flow_control_send -True False	-if)# flowcontrol send [on off] #show interfaces output flow-control is unsupported , input flow-control is unsupported	-if)#flow-control egress	-if)# flowcontrol send [on off] # show interface Input flow-control is off, output flow-control is off

	XE	XR	NX
<p>(Lag) lag_bundle_id -Int</p> <p>lag_activity -Enum('active','passive','on','auto','desirable')</p> <p>lag_non_silent -Bool</p> <p>lag_force -Bool</p>	<pre>(config-if)# channel-group <lag_bundle_id> mode {auto [non-silent]]desirable [non- silent]]on}}{active passive} #show lacp internal A - Device is in Active mode P - Device is in Passive mode Channel group 1 Port Flags State Priority Key Key Number State Gi2 SA bndl 32768 0x1 0x1 0x1 0x3D Gi3 SA bndl 32768 0x1 0x1 0x1 0x3D</pre>	<pre>(config-if)# bundle id <lag_bundle_id> [mode {active on passive}] #show bundle Bundle-Ether1</pre>	<pre>(config-if)# channel-group <lag_bundle_id> [force] [mode {on active passive}] # show port-channel database port-channel1 Ports: Ethernet1/1 [active] [up] *</pre>
<p>(Lag) lag_lacp_port_priority -Int</p>	<pre>(config-if)# lacp port-priority <lag_lacp_port_priority> #show lacp internal Gi4 SP bndl 10 0x2 0x2 0x1 0x3C</pre>	<pre>(config-if)# bundle port- priority <lag_lacp_port_priority> #show bundle Gi0/0/0/0 Local Active 0x000a, 0x0001 1000000 (NOTE: if not 0x8000)</pre>	<pre>(config-if)# lacp port- priority <Int> No show cmd (only show run)</pre>
<p>(Lag) lag_pagp_port_priority -Int</p>	<pre>(config-if)# pagp port-priority <lag_pagp_port_priority> #show pagp internal Gi0/3 SC U6/S7 H 30s 1 10 Any 11</pre>	N/A	N/A
<pre>Interface +-- PhysicalInterface +-- EthernetInterface +-- ServiceInstance</pre>	<si_id>		<si_id>

	XE	XR	NX
si_id -Int si_type -Enum('ethernet')	-if)# service instance <si_id> <si_type>	N/A	-if)# service instance <si_id> <si_type>
si_encap -Enum('dot1q','default') si_encap_vlan_id -Int	'dot1q': -if-srv)# encapsulation <si_encap> <si_encap_vlan_id> 'default': -if-srv)# encapsulation <si_encap>	N/A	'dot1q': -if-srv)# encapsulation <si_encap> <si_encap_vlan_id> 'default': -if-srv)# encapsulation <si_encap>
si_rewrite_ingress_tag_pop -Int si_rewrite_ingress_tag_pop_symmetric -Bool	-if-srv)# rewrite ingress tag pop <si_rewrite_ingress_tag_pop> [symmetric]	N/A	-if-srv)# rewrite ingress tag pop <si_rewrite_ingress_tag_ pop> [symmetric]
Interface +-- VirtualInterface			
(None)			
Interface +-- VirtualInterface +-- SubInterface			
(None)			
Interface +-- VirtualInterface +-- SubInterface +-- ServiceInstance			

	XE	XR	NX
si_id -Int si_type -Enum('ethernet')	-if)# service instance <si_id> <si_type>	N/A	-if)# service instance <si_id> <si_type>
si_encap -Enum('dot1q') si_encap_vlan_id -Int	-if-srv)# encapsulation <si_encap> <si_encap_vlan_id>	N/A	-if-srv)# encapsulation <si_encap> <si_encap_vlan_id>
si_rewrite_ingress_tag_pop -Int si_rewrite_ingress_tag_pop_symmetric -Bool	-if-srv)# rewrite ingress tag pop <si_rewrite_ingress_tag_pop> [symmetric]	N/A	N/A
Interface +-- VirtualInterface +-- LoopbackInterface			
(None)			
Interface +-- VirtualInterface +-- VlanInterface			
(None)			
Interface +-- VirtualInterface +-- Nve			(config)# interface <nve_name>
(Vxlan) nve_name - Str 'nve1'			(config)# interface <nve_name>

	XE	XR	NX
(Vxlan) nve_host_reachability_protocol - Enum ('bgp')			(config-if-nve)# host-reachability protocol <nve_host_reachability_protocol> No show cmd (show run only)
(Vxlan) nve_adv_virtual_rmac - Bool			(config-if-nve)# advertise virtual-rmac # show nve interface nve 1 detail i Adver Virtual RMAC Advertisement: Yes
(Vxlan) nve_src_intf_loopback - Str			(config-if-nve)# source-interface <nve_src_intf_loopback> # show nve interface nve 1 detail i Source-Interface: Source-Interface: loopback1 (primary: 201.11.11.11, secondary: 201.12.11.22)
(Vxlan) nve_multisite_bgw_intf - Str			(config-if-nve)# multisite border-gateway interface <nve_multisite_bgw_intf> # show nve interface nve 1 detail Multisite bgw-if: loopback2 (ip: 101.101.101.101, admin: Down, oper: Down)

	XE	XR	NX
Interface +-- VirtualInterface +-- Nve +-- Vni			(config-if-nve)# member vni <nve_vni> [associate-vrf] (config-if-nve-vni)#
(Vxlan) nve_vni - Int nve_vni_associate_vrf - Bool			(config-if-nve)# member vni <nve_vni> [associate-vrf] (config-if-nve-vni)# No show cmd (show run only)
(Vxlan) nve_vni_suppress_arp - Bool			(config-if-nve-vni)# suppress-arp No show cmd (show run only)
(Vxlan) nve_vni_multisite_ingress_replication - Bool			(config-if-nve-vni)# multisite ingress-replication No show cmd (show run only)
(Vxlan) nve_vni_mcast_group - Str			(config-if-nve-vni)# mcast-group <nve_vni_mcast_group> # show nve vni Interface VNI Multicast-group State Mode Type [BD/VRF] Flags ----- - ----- -- nve1 5001 234.1.1.1 Up CP L2 [1001]

	XE	XR	NX
Interface +-- VirtualInterface +-- LagInterface			
(Lag) enabled_lacp -Bool	N/A	N/A	Global Configuration Mode (config)# feature lacp # show feature i lacp lacp 1 enabled
(Lag) lag_lacp_system_priority -Int	Global Configuration Mode (config)# lacp system-priority <lag_lacp_system_priority> #show lacp sys-id 100 , 001e.49e6.bc00	Global Configuration Mode (config)# lacp system priority <lag_lacp_system_priority> #show lacp system-id Priority MAC Address ----- 0x0064 00-1b-0c-10-5a-26	Global Configuration Mode (config)# lacp system- priority <lag_lacp_system_priorit y> # show lacp system- identifier 100 ,5e-2-0-0-7
(Lag) lag_lacp_max_bundle -Int	(config-if)# lacp max-bundle <lag_lacp_max_bundle> No show cmd (only show run)	(config-if)# bundle maximum-active links <lag_lacp_max_bundle> #show bundle Maximum active links: 2	(config-if)# lacp max- bundle <lag_lacp_max_bundle> No show cmd (only show run)
(Lag) lag_lacp_min_bundle -Int	(config-if)# lacp min-bundle <lag_lacp_min_bundle> No show cmd (only show run)	(config-if)# bundle minimum- active-links <lag_lacp_min_bundle> #show bundle Minimum active links / bandwidth: 2 / 1 kbps	(config-if)# lacp min-links <lag_lacp_min_bundle> No show cmd (only show run)

	XE	XR	NX
(Bfd) lag_bfd_v4_destination -Str	N/A	(config-if)# bfd address-family ipv4 destination <lag_bfd_v4_destination>	N/A
(Bfd) lag_bfd_v4_fast_detect -Bool	N/A	(config-if)# bfd address-family ipv4 fast-detect	N/A
(Bfd) lag_bfd_v4_min_interval -Int	N/A	(config-if)# bfd address-family ipv4 minimum-interval <lag_bfd_v4_min_interval>	N/A
(Bfd) lag_bfd_v6_destination -Str	N/A	(config-if)# bfd address-family ipv6 destination <lag_bfd_v6_destination>	N/A
(Bfd) lag_bfd_v6_fast_detect -Bool	N/A	(config-if)# bfd address-family ipv6 fast-detect	N/A
(Bfd) lag_bfd_v6_min_interval -Int	N/A	(config-if)# bfd address-family ipv6 minimum-interval <lag_bfd_v6_min_interval>	N/A
Show Commands:	show interfaces show vrf detail show ip interface show ipv6 interface show interfaces switchport show etherchannel summary show interfaces [intf] accounting	show interfaces detail show vlan interface show vrf all detail show ipv4 vrf all interface show ipv6 vrf all interface show bundle show interfaces [intf] accounting	show interface show vrf all interface show ip interface vrf all show ipv6 interface vrf all show interface switchport show routing ipv6 vrf all show routing vrf all
Sample config:	<pre>interface GigabitEthe description desc mac-address aaaa.bbb</pre>	<pre>interface GigabitE description desc bandwidth 768</pre>	<pre>interface Etherne description desc no switchport</pre>

```

mtu 1600
bandwidth 768
bandwidth inherit 768
vrf forwarding VRF1
ip address 10.2.2.2
ip address 10.1.1.1
logging event link-s
load-interval 30
delay 333
shutdown
speed 1000
no negotiation auto
medium p2p
ipv6 address 2001:DE
ipv6 address 2001:DE
ipv6 address 2001:DE
ipv6 address 2001:DE

```

```

mtu 1600
vrf VRF1
ipv4 point-to-poi
ipv4 address 10.1
ipv4 address 10.2
ipv6 address 2001
ipv6 address 2001
ipv6 address 2001
negotiation auto
carrier-delay up
mac-address aaaa.
load-interval 30
dot1q native vlan
!

```

```

interface GigabitE
encapsulation dot
!
interface GigabitE
ipv4 point-to-poi
ipv4 unnumbered L
encapsulation dot
!

```

```

mtu 1600
delay 333
logging event po
bandwidth 768
bandwidth inher
mac-address aaaa
vrf member VRF1
no ip redirects
ip address 10.4
ip address 10.2
ip address 10.3
ipv6 address 200
ipv6 address 200
ipv6 address 200
no shutdown

```

```

interface Etherne
shutdown
mtu 1600
encapsulation do
interface Etherne
mtu 1600
encapsulation do
medium p2p
ip unnumbered lo
no shutdown

```

```

interface Etherne
switchport

```

	XE	XR	
			<pre> switchport mode switchport trunk no shutdown interface Ethernet shutdown switchport switchport access </pre>

Interface Ops structure

show commands

IOS-XE	IOS-XR	NX-OS
<pre> show interfaces show vrf detail show ip interface show ipv6 interface show interface switchport show etherchannel summary show interfaces [intf] accounting </pre>	<pre> show interfaces detail show vlan interface show vrf all detail show ipv4 vrf all interface show ipv6 vrf all interface show bundle show interfaces [intf] accounting </pre>	<pre> show interface show vrf all interface show ip interface vrf all show ipv6 interface vrf all show interface switchport show routing ipv6 vrf all show routing vrf all </pre>

Ops structure

NOTE: where self represents the instance object of the ops object

```
self.info = {
    interface: {
        # Ops/Conf 'Ethernet1/1'|'Vlan10'|'Ethernet1/1.10'
        'description': description, # Ops/Conf 'To R2'
        'type': type, # Ops/Conf '10/100/1000 Ethernet'
        'oper_status': oper_status, # Ops 'up'|'down'
        'last_change': last_change, # Ops '00:00:04'|'never'
        'phys_address': phys_address, # Ops '5254.009c.f2e6'
        'mtu': mtu, # Ops/Conf '1500 bytes'
        'enabled': enabled, # Ops/Conf True(no shut)|False(shut)
        'vlan_id': vlan_id, # Ops '10'
        'access_vlan': access_vlan, # Ops/Conf '100'
        'trunk_vlans': trunk_vlans, # Ops/Conf '100-200'
        'mac_address': mac_address, # Ops/Conf 'AAAA.BBBB.CCCC'
        'auto_negotiate': auto_negotiate, # Ops/Conf True|False
        'duplex_mode': duplex_mode, # Ops/Conf 'full'|'half'
        'port_speed': port_speed, # Ops/Conf '100'
        'switchport_enable': switchport_enable, # Ops/Conf Boolean
        'switchport_mode': switchport_mode, # Ops/Conf 'trunk'|'access'
        'medium': medium, # Ops/Conf Enum('p2p','broadcast')
        'delay': delay, # Ops/Conf '300'
        'port_channel': {
            'port_channel_member': port_channel_member, # Ops Boolean
            'port_channel_int': port_channel_int, # Ops 'Po1'
            'port_channel_member_intfs': port_channel_member_intfs, # Ops list['Eth1/1','Eth1/2']
        },
        'flow_control': {
            'receive': flow_control_receive, # Ops/Conf Boolean
            'send': flow_control_send, # Ops/Conf Boolean
        }
        'bandwidth': bandwidth, # Ops/Conf '1000000' Kbit
    }
}
```

```
'link_status': link_status,          # Ops/Conf True|False (logging)
'vrf': vrf,                          # Ops/Conf 'VRF1'
'vrf_downstream': vrf_downstream,    # Conf 'VRF2'
'accounting': {
  protocol: { # Ops Str 'arp'
    'pkts_in': pkts_in, # Ops Int 9
    'pkts_out': pkts_out, # Ops Int 9
    'chars_in': chars_in, # Ops Int 378
    'chars_out': chars_out, # Ops Int 378
  },
},
'counters': {
  'rate': {
    'load_interval': load_interval, # Ops/Conf '5 minutes'
    'in_rate': in_rate,              # Ops '125 bits/sec'
    'in_rate_pkts': in_rate_pkts,    # Ops '55 packets/sec'
    'out_rate': out_rate,            # Ops '125 bits/sec'
    'out_rate_pkts': out_rate_pkts,  # Ops '55 packets/sec'
  }
  'in_pkts': in_pkts,                # Ops '8'
  'in_octets': in_octets,            # Ops '888'
  'in_unicast_pkts': in_unicast_pkts, # Ops '5'
  'in_broadcast_pkts': in_broadcast_pkts, # Ops '0'
  'in_multicast_pkts': in_multicast_pkts, # Ops '0'
  'in_discards': in_discards,        # Ops '0'
  'in_errors': in_errors,            # Ops '0'
  'in_unknown_protos': in_unknown_protos, # Ops '0'
  'in_mac_control_frames': in_mac_control_frames, # Ops '0'
  'in_mac_pause_frames': in_mac_pause_frames, # Ops '0'
  'in_oversize_frames': in_oversize_frames, # Ops '0'
  'in_jabber_frames': in_jabber_frames, # Ops '0'
  'in_fragment_frames': in_fragment_frames, # Ops '0'
```

```
'in_8021q_frames': in-8021q-frames,          # Ops '0'
'in_crc_errors': in_crc_errors,             # Ops '0'
'out_pkts': out_pkts,                       # Ops '33'
'out_octets': out_octets,                   # Ops '358'
'out_unicast_pkts': out_unicast_pkts,       # Ops '1'
'out_broadcast_pkts': out_broadcast_pkts,   # Ops '2'
'out_multicast_pkts': out_multicast_pkts,   # Ops '2'
'out_discard': out_discard,                 # Ops '0'
'out_errors': out_errors,                   # Ops '0'
'out_mac_control_frames': out_mac_control_frames, # Ops '0'
'out_mac_pause_frames': out_mac_pause_frames, # Ops '0'
'out_8021q_frames': out_8021q_frames,       # Ops '0'
'last_clear': last_clear,                   # Ops 'never'|'00:00:02'
},
'encapsulation': {
  'enacapsulation': encapsulation, # Ops/Conf 'dot1q'|'arpa'|'ppp'|'hdlc'
  'first_dot1q': first_dot1q,       # Ops/Conf '100'
  'second_dot1q': second_dot1q,     # Ops/Conf 'any'|'100,200-300'
  'native_vlan': native_vlan,       # Ops/Conf '1'
},
'ipv4': {
  {ipv4|ipv4_secondary}/prefix_length: { # Ops/Conf '192.168.1.1/24'
    'ip': ipv4|ipv4_secondary,         # Ops/Conf '192.168.1.1'
    'prefix_length': prefix_length,    # Ops/Conf '24'
    'origin': origin,                  # Ops 'other'|'static'|'dhcp'|'link_layer'(v6)|'random'
    'sedondary': secondary,            # Ops/Conf True|False
    'route_tag': route_tag,           # Ops/Conf '0'
    'secondary_vrf': secondary_vrf,   # Ops/Conf 'VRF1'
  }
  'unnumbered': {
    'interface_ref': unnumbered_intf_ref, # Ops/Conf 'Ethernet2'
  },
},
```

```
    },
    'ipv6': {
      ipv6[/prefix_length]: {
        # Ops/Conf '2001:db8:1:1::1/64' = ipv6pre
        'ip': ipv6, # Ops/Conf '2001:db8:1:1::1'
        'prefix_length': ipv6_prefix_length, # Ops/Conf '64'
        'anycast': ipv6_anycast, # Ops/Conf True|False
        'eui_64': ipv6_eui_64, # Ops/Conf True|False
        'route_tag': ipv6_route_tag, # Ops/Conf '10'
        'origin': ipv6_origin, # Ops 'other' | 'static' | 'dhcp' | 'link_layer' (v6) | 'r
        'status': ipv6_status, # Ops 'valid' | 'preferred' | 'deprecated' | 'invalid' |
        'autoconf': { # Ops/Conf (Conf:v6_auto_conf)
          'valid_lifetime': ipv6_valid_lifetime, # Ops '1535'
          'preferred_lifetime': ipv6_preferred_lifetime, # Ops '335'
        }
      },
      'unnumbered': {
        'interface_ref': ipv6_unnumbered_intf_ref, # Ops/Conf 'Ethernet2'
      },
      'enabled': ipv6_enabled, # Ops/Conf True|False
    },
  },
}

# port_channel is developed only under NXOS Interface Ops.
```

No labels