



*TOMORROW
starts here.*

Cisco *live!*



Implementing Network Automation - Power Tools for Enterprise Switching

BRKCRS-3090

James Gilarte – Systems Engineer

#clmel

Cisco *live!*

Hidden Features of IOS/IOS-XE

IOS/
IOS-XE

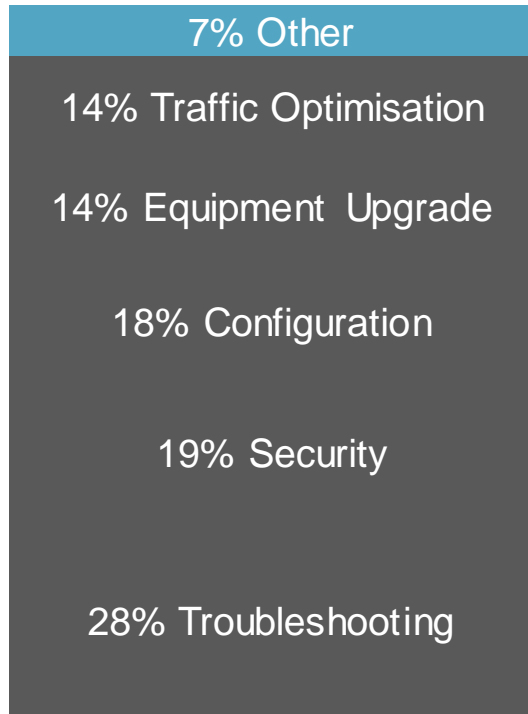


Today's IT Model – Complex, Not Fast Enough

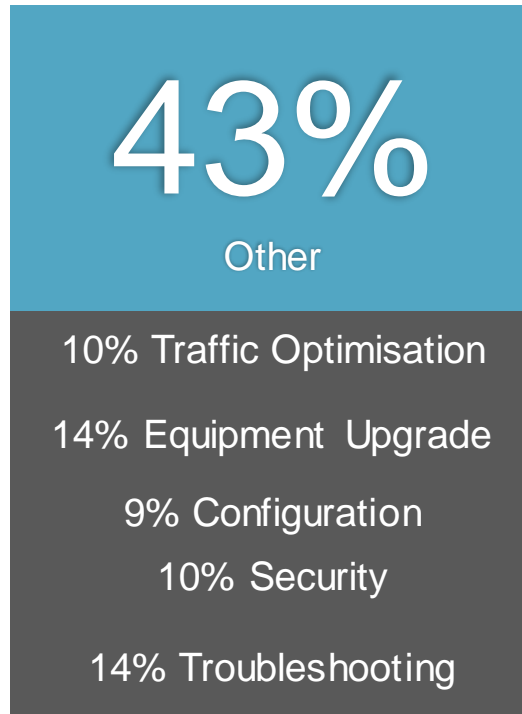
Box by Box
Manual Configuration



Where Is The Time Going?



CURRENT IT*



FAST IT

36%

More Time Available for Business Innovation

36%

Total Network Operations Time Savings

Agenda

- Automated Deployment
 - Smart Install
 - Next Gen PnP
- Automated Port Profiling
 - Auto Smartports
 - AutoConf
 - Migration From ASP to AutoConf
- Automated Virtual Network
 - Easy VSS
- Q&A
- Conclusion

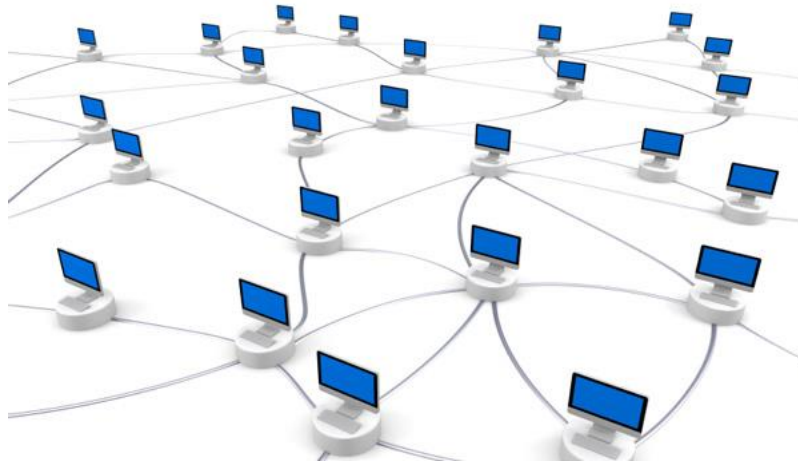




Automated Deployment

The Issues We Are Trying To Solve

Simple...Deploying network devices in the easiest manner





Smart Install

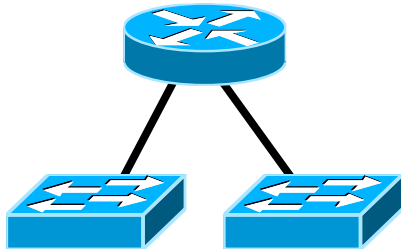
Cisco *live!*

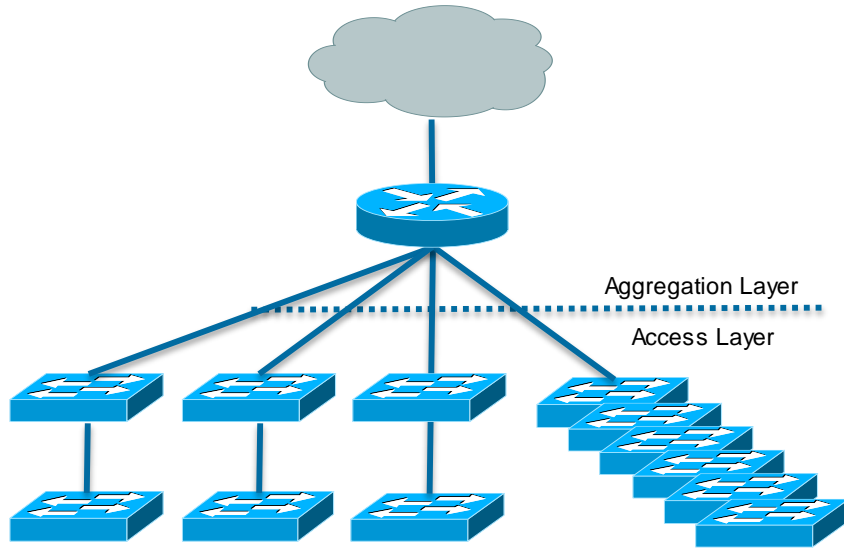
Really Quick Notes

Smart Install

Available since IOS 12.2(55) SE or later

CLI configuration





Switch Deployment

Switch Replacement

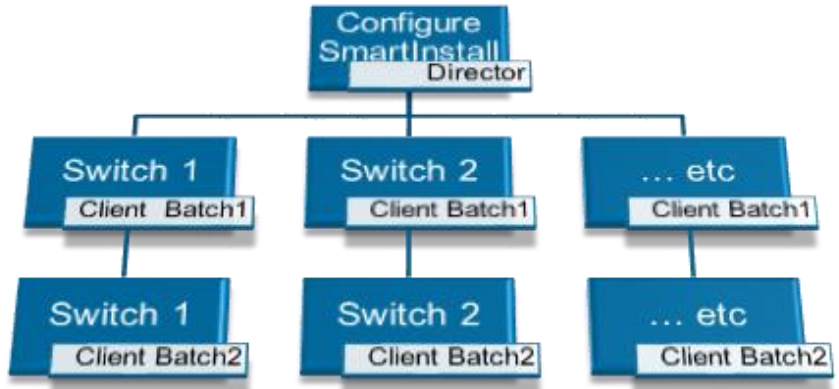


Rack and Stack

Typical Process—no SmartInstall



SmartInstall Process – Multi-Tasking!



Remove the human bottleneck up 20x or more

Smart Install Components and Terminology

Client

Receives image and configuration from Director

Groups

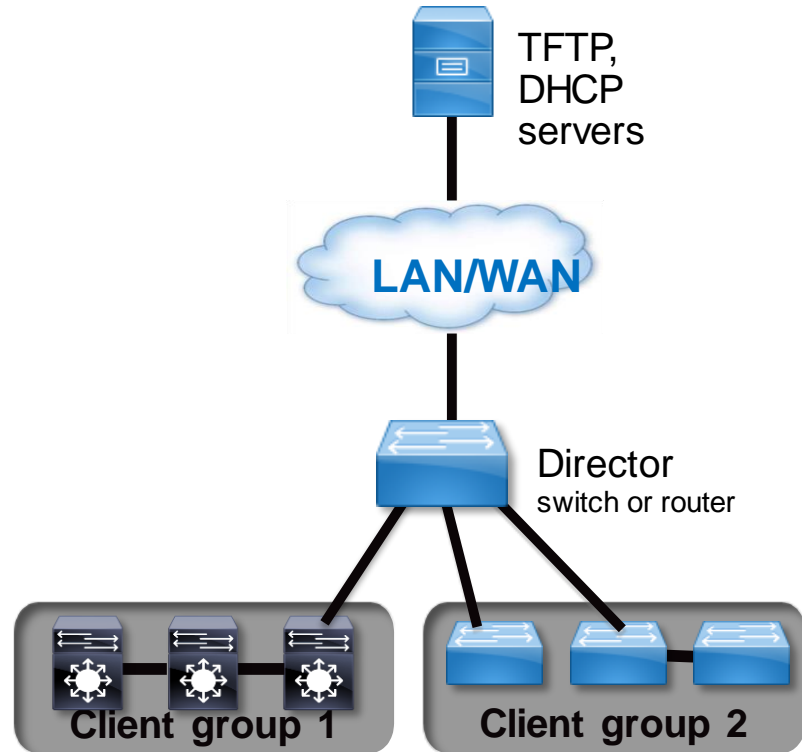
Collection of Clients (same image and config)

Director

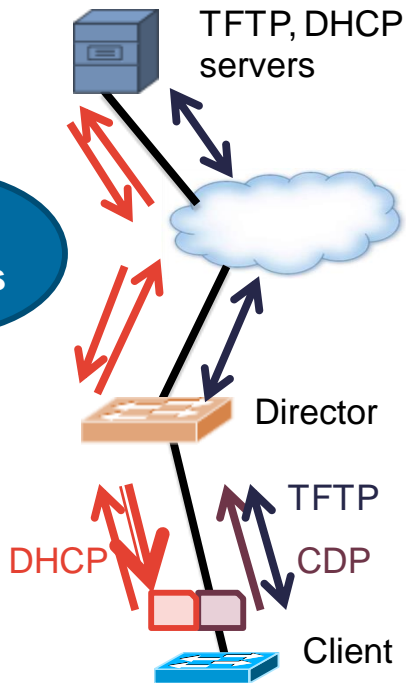
Manages client image and configuration

DHCP and TFTP Servers

Centrally located and shared across network



Smart Install Process



1. New switch connected
2. Director discovers client via CDP
3. New switch issues DHCP discover
4. Director adds options to DHCP offer
(Director MUST be first L3 hop between client and DHCP server)
5. Client retrieves image, config via TFTP
6. Client reboots with new configuration and image

Compatible Products (Reference)

Smart Install Directors

ISR Branch Router

G1: 1841, 2801, 2811, 2821, 2851,
3825, 3845
G2: 1921, 1941, 2901, 2911, 2921,
2951, 3925, 3945, 3925E, 3945E,

Catalyst 3K

3750, 3750G, 3750v2, 3750E, 3560,
3560v2, 3560E, 3560G, 3750X, 3560X,
3650, 3850

Catalyst 4500 64 clients
Catalyst 6500 32 clients

Smart Install Clients

Catalyst 3K

3750, 3750v2, 3750E, 3750G, 3750X,
3560, 3560v2, 3560E, 3560G, 3560X, 3850,
3650

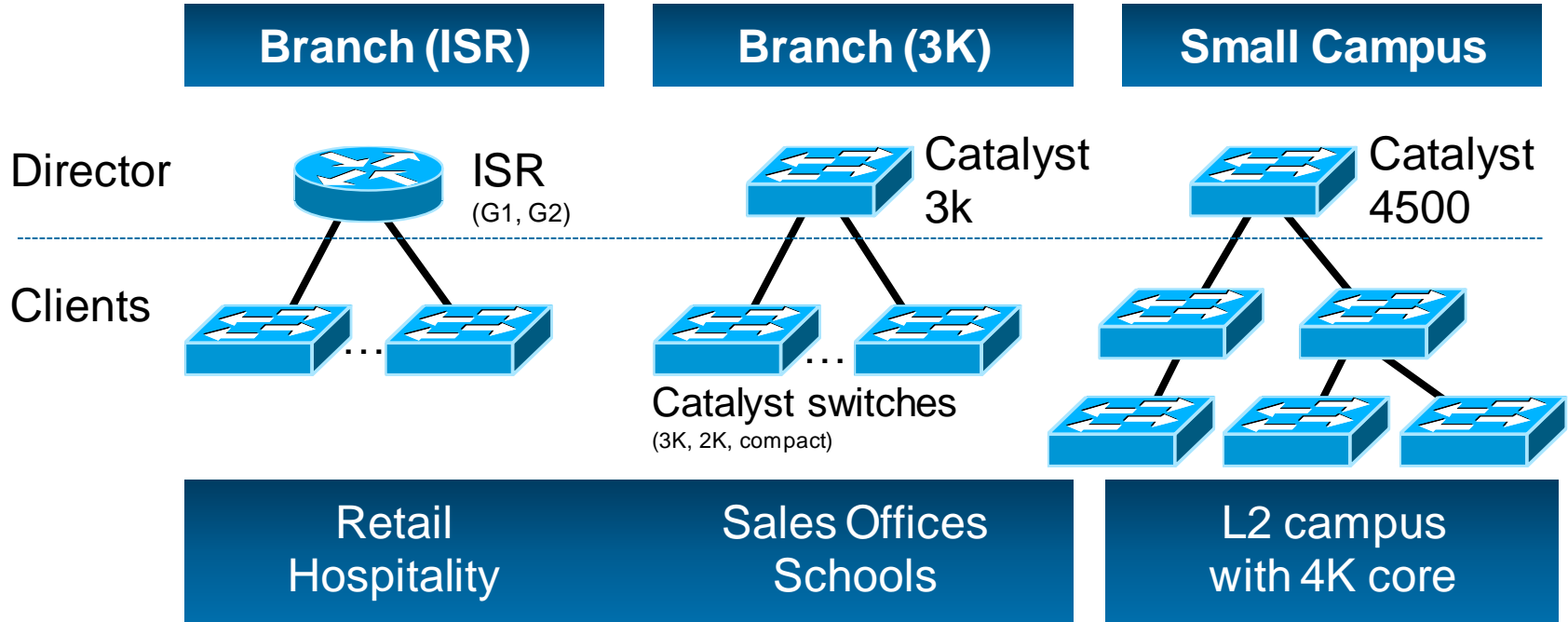
Catalyst 2K

2960, 2960S, 2960G, 2960P, 2975

Catalyst 2K/3K Compact

2960C, 3560C

Common Deployment Scenarios



Enable Smart Install On Director

- **Step #1 – execute the ‘director’ ‘vstack basic’ commands**

```
Switch(config)#vstack director 10.10.0.1  
Switch(config)#vstack basic
```

Enable Smart Install On Director

- **Step #2 (DHCP Optional) – execute the ‘vstack dhcp local-server’ command**

```
Switch(config)#vstack dhcp local-server smart-install-switches
Switch(config-vstack-dhcp)#address-pool 10.10.0.0 255.255.255.0
Switch(config-vstack-dhcp)#default-router 10.10.0.1
Switch(config-vstack-dhcp)#file-server 10.10.0.1
Switch(config-vstack-dhcp)#exit
Switch(config)#ip dhcp remember
```

Enable Smart Install On Director

- **Step #3 (Optional) – copy image and config file to director with ‘copy tftp flash’**

```
Switch#copy tftp flash
Address or name of remote host []? 10.10.0.100
Source filename []? c3750-image.tar
Destination filename [c3750-image.tar]?
Do you want to over write? [confirm]
Accessing tftp://10.10.0.100/c3750-image.tar...
```

```
Switch#copy tftp flash
Address or name of remote host []? 10.10.0.100
Source filename []? smart-install.txt
Destination filename [smart-install.txt]?
Do you want to over write? [confirm]
Accessing tftp://10.10.0.100/smart-install.txt...
```

Setup Director as TFTP Server w/Image and Config

- **Step #4 – assign default image and config with ‘vstack’ command**

```
Switch(config)#vstack image flash:c3750-image.tar  
Switch(config)#vstack config flash:smart-install.txt
```

TFTP Server Guidelines for Director

- Total flash memory space (used and free) must be large enough for Clients
- Flash must be large enough to contain Director configuration and image also
- IOS images vary in size depending on Client type, flash memory is limited
- If more than one product ID on the network, best to use external TFTP server

Use Built-In Groups to Upgrade New Client Switches

- **Step #1 – execute ‘vstack group built-in’ command to use Built-In groups**

```
Switch(config)#vstack group built-in 3560g 48
Switch(config-vstack-group)#image tftp://10.10.0.100/3560_48_imagelist.tar
Switch(config-vstack-group)#config tftp://10.10.0.100/3560_48_config.txt

Switch(config)#vstack group built-in 3750g 48poe
Switch(config-vstack-group)#image tftp://10.10.0.100/3750_48poe_imagelist.tar
Switch(config-vstack-group)#config tftp://10.10.0.100/3750_48poe_config.txt
```

Configure Custom Group Based on Stack

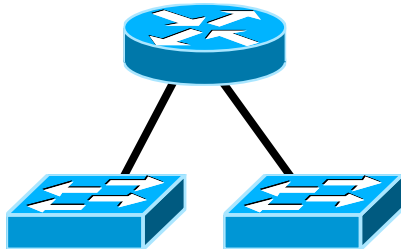
- **Step #1 – execute ‘vstack group custom’ to create custom group**

```
Switch(config)#vstack group custom 3750_24poe-stack-group stack
Switch(config-vstack-group)#image tftp://10.10.0.100/3750stack_24poeimagelist.txt
Switch(config-vstack-group)#config tftp://10.10.0.100/3750stack_24poe_config.txt
Switch(config-vstack-group)#match 1 3750 24poe
Switch(config-vstack-group)#match 2 3750 24poe
Switch(config-vstack-group)#match 3 3750 24poe
Switch(config-vstack-group)#match 4 3750 24poe
```

Really Quick Notes

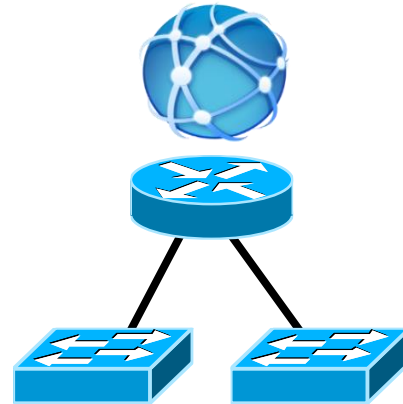
Smart Install

- Available since IOS 12.2(55) SE or later
- CLI configuration



Next Gen PnP (APIC-EM)

- Available since IOS 15.2(2)/3.6
- GUI operational model
- APIC-EM Based



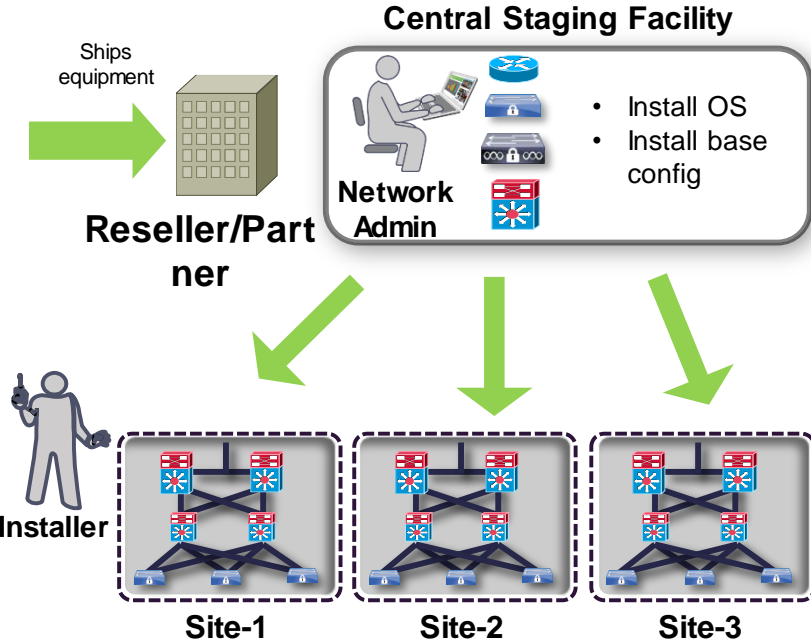


Next Generation Plug-n-Play

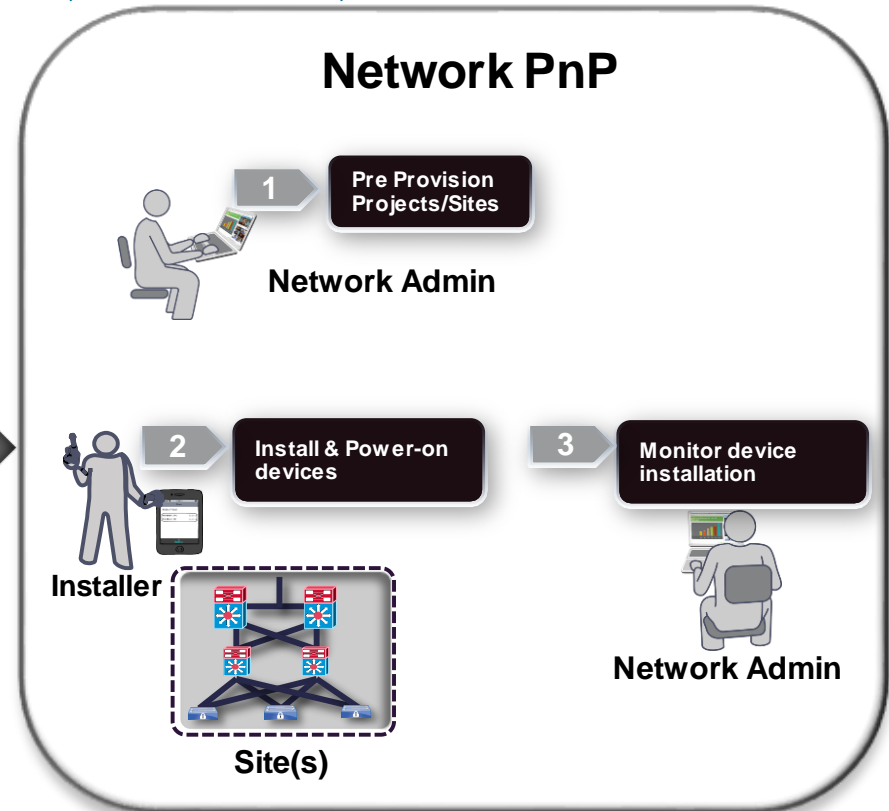
Cisco *live!*

Network Plug-N-Play – Simple, Secure, Scalable

Today's Process



Network PnP



Unskilled
Installer

GUI Based

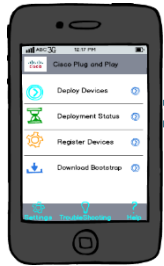
Consistent for devices &
PIN(Campus/Branch)

Secure

Zero-touch
RMA

Greenfield
& Brownfield

Network PnP – Components



PnP Helper Applications: Applications on smart phones that facilitate deployment. Deliver Boot Strap config when needed and Status/Troubleshooting checks

Cisco Cloud Redirection Service
<https://devicehelper.cisco.com/device-helper>



PnP Protocol: Protocol between the Agent and the PnP server. This is an open schema allowing third-party development of PnP servers



PnP Agent: An embedded agent on the ISR and Catalyst to automate deployment process



PnP Server: A central server that manages Sites, Site devices and its images, configurations, files and licenses for the deployment. APIC EM has Cisco built PnP Server and also provides north bound REST API for third party/custom applications integration. PnP Server communicates with the Agents using an open PnP protocol.

Use Case : Branch Deployment

Step 1

Pre Provision Site in APIC EM

- Serial Number based match rule
- Config and/or Image
- Installer ID

Network Admin



Step 2

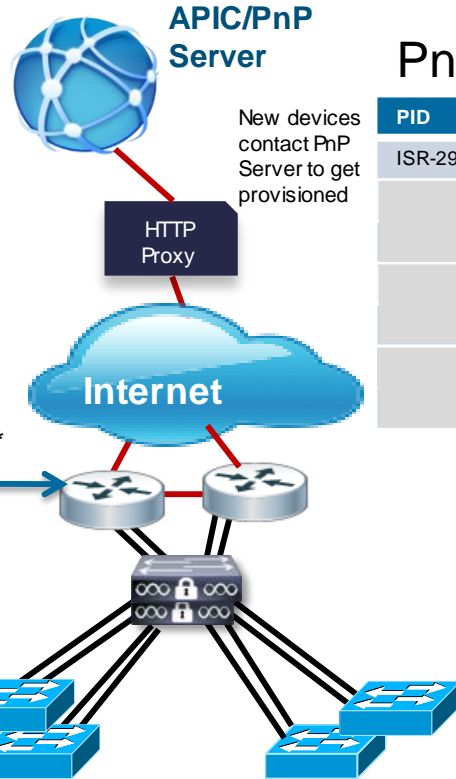
Installer on site with PnP mobile application

- Rack and Stack devices
- Power-on
- Start Deployment
- Check Status and/or troubleshoot (optional)



Installer

Deliver bootstrap*



PnP Server/Site Updates

PID	Serial #	Hostname	IP address
ISR-2951	FOX23zxd	ISR-main	192.168.15.1

Network Admin

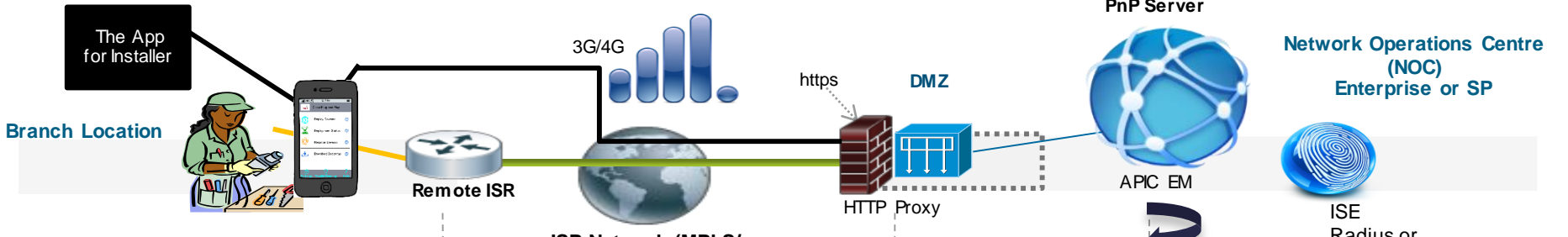
Step 3



IT Admin can remotely monitor status of install

Installer App based Automated Install with PnP Server (Reference)

zero-touch config for the installer



0) Installer powers on ISR

- 1) Installer connects the ISR console cable with the App
- 2) Installer clicks "Deploy Device" option and the Installer App connects to PnP Server over 3G/4G/Wifi
- 3) PnP Server validates the credentials of the installer
- 4) PnP Server registers Serial number of that device and sends the ISR bootstrap config
- 5) The App receives the bootstrap config (WAN config + Agent config) from PnP Server.
- 6) App installs server certificate (via console) and applies bootstrap config. to ISR
- 7) ISR connects to Server using https and request its image and full config.



APIC-EM: Network PnP Dashboard

The screenshot displays the APIC-EM Network Plug and Play (PnP) dashboard. The interface includes a left-hand navigation menu with options like Home, Discovery, Device Inventory, Host Inventory, Topology, IWAN, MACsec, Policy, Quality of Service, and Policy Analysis. The main content area is titled "APIC - Enterprise Module" and features a navigation bar with tabs for Status, Sites, Image Management, Unclaimed Devices, and RMA. A red dashed box highlights the "Sites" tab. Below the navigation bar, there is a search input field labeled "Find Site..." and a "Load Site" button. The dashboard displays five key metrics in large, rounded boxes: Pre-Provisioned (3), In Progress (1), Completed (0), Unclaimed Devices (0), and Errors (0). The top right corner shows a "Sign Out" button and a notification badge with the number 61. The bottom of the page includes a footer with the text "Page loaded in 15660ms" and a "I wish this page would..." prompt.

Pre-Provisioned	In Progress	Completed	Unclaimed Devices	Errors
3	1	0	0	0

APIC-EM ZTD: Pre-provision Site

'Site' management

Site : grouping of devices in a network for pre-provisioning automation

Site = Branch or campus location

The screenshot displays the APIC-EM ZTD interface. The browser address bar shows the URL <https://sdn-controller.cisco.com/ztd/sites>. The page title is "APIC - Enterprise Module". The navigation menu on the left includes "Home", "BETA", "Discovery", "Device Inventory", "Host Inventory", "Topology", "Policy", "Pre-BETA", "Quality of Service", "Policy Analysis", and "Zero Touch Deployment". The main content area has tabs for "Status", "Sites", "Image Management", "Unclaimed Devices", and "RMA". A red box highlights the "Location:" input field and the "Load", "Create", "Clone", and "Delete" buttons. Below this, there is a checkbox for "Deploy Images that Do not Support Cisco PnP Protocol (Unsecure)". Another red box highlights the "Add Device" form, which includes input fields for "Serial Number", "* Device Name", and "* Product ID", along with an "Add Device" button. At the bottom, there is a table header for "ZTD Site Devices" with columns for "Serial Number", "Device Name", "Product ID", "Config", "Bootstrap", "Image", "Status", and "Delete". The table content shows "No site location selected." and a "Next" button. The footer indicates "Page loaded in 000ms" and "I wish this page would..."

APIC-EM ZTD: Site – Add Devices

Site Workflow

- Serial # and PID based device matching on server
- Operational Config and/or IOS image for each device
- Bootstrap config optional
- Import/Export to use table driven data entry

The screenshot displays the APIC-EM ZTD interface for 'Branch A'. The left sidebar shows navigation options: Home, BETA (Discovery, Device Inventory, Host Inventory, Topology, Policy), Pre-BETA (Quality of Service, Policy Analysis), and Zero Touch Deployment (highlighted). The main content area shows the 'Add Devices' form with a 'Location' dropdown set to 'Branch A' and buttons for 'Load', 'Create', 'Clone', and 'Delete'. A checkbox option is present: 'Deploy Images that Do not Support Cisco PnP Protocol (Unsecure)'. Below the form are input fields for 'Serial Number', '* Device Name', and '* Product ID', followed by an 'Add Device' button. A table titled 'Branch A Devices' lists eight devices with columns for Serial Number, Device Name, Product ID, Config, Bootstrap, Image, Status, and Delete. A red box highlights the 'Add Device' button and the table. Another red box highlights a secondary set of buttons: 'Topology', 'Table', 'Export', 'Import', and 'Refresh'. The table footer indicates 'Displaying Eight of Eight Devices' and a 'Next' button is visible at the bottom right.

Serial Number	Device Name	Product ID	Config	Bootstrap	Image	Status	Delete
FOC12314AA44	Access_switch 1	WS-C3560C-8PC				UNKNOWN	
FOC12314AA34	Access_switch 2	WS-C3560C-8PC				UNKNOWN	
FOC12314AAF1	Access_switch 3	WS-C3560C-8PC				UNKNOWN	
FOC12314AA12	Access_switch 4	WS-C3560C-8PC				UNKNOWN	
FOCX23423DD1	3850stack1	WS-C3850-48P				UNKNOWN	
FOCX23423234	3850stack2	WS-C385048P				UNKNOWN	
FOC98798CC2	ISR 1	CISCO29551				UNKNOWN	
FOC98798CD6	ISR 2	CISCO2951				UNKNOWN	

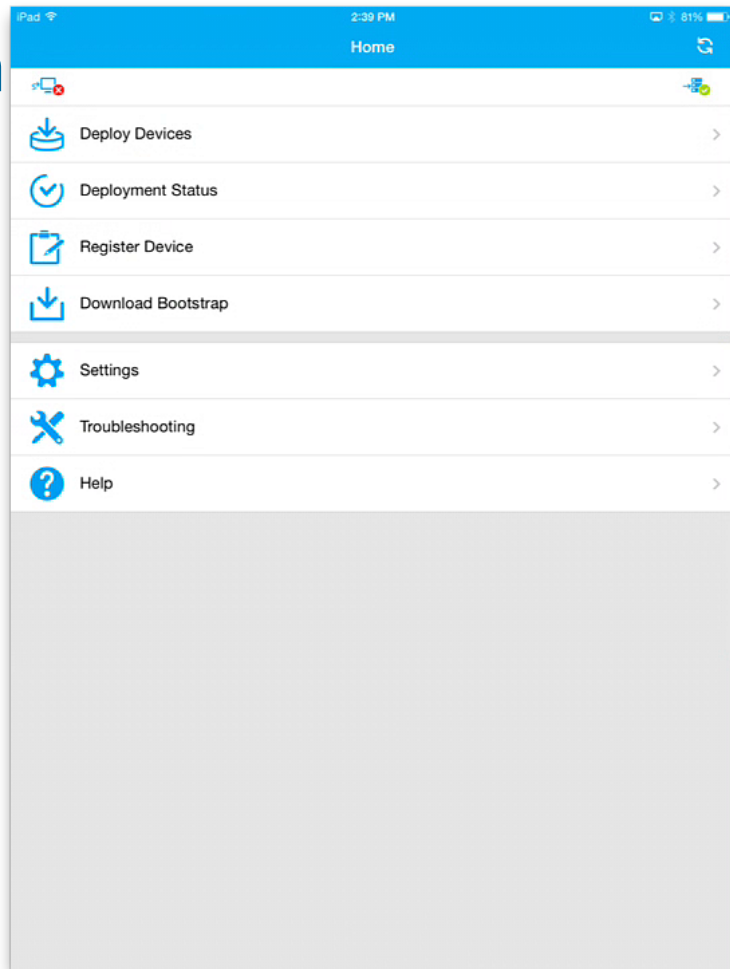
Installer App: Home Screen

App used by remote installer

Runs on iPad/iPhone

App provides -

- ◆ Deliver bootstrap configuration
- ◆ Status of PnP devices
- ◆ Notes for installer
- ◆ Register a device for a site
- ◆ Troubleshooting device install





Automated Port Profiling

The Issues We Are Trying To Solve

- Manual Configuration of every port
 - Device Moves – What happens?
- Wasted Ports
- Unsure how to mix multiple features together
- Not knowing what is connected
 - Which interface is the printer connected to?

Really Quick Notes

Auto Smart Ports

- Available since IOS 12.2(55) SE or later
- Configure ports on a switch based on the end device connected to it
- Macro based solution





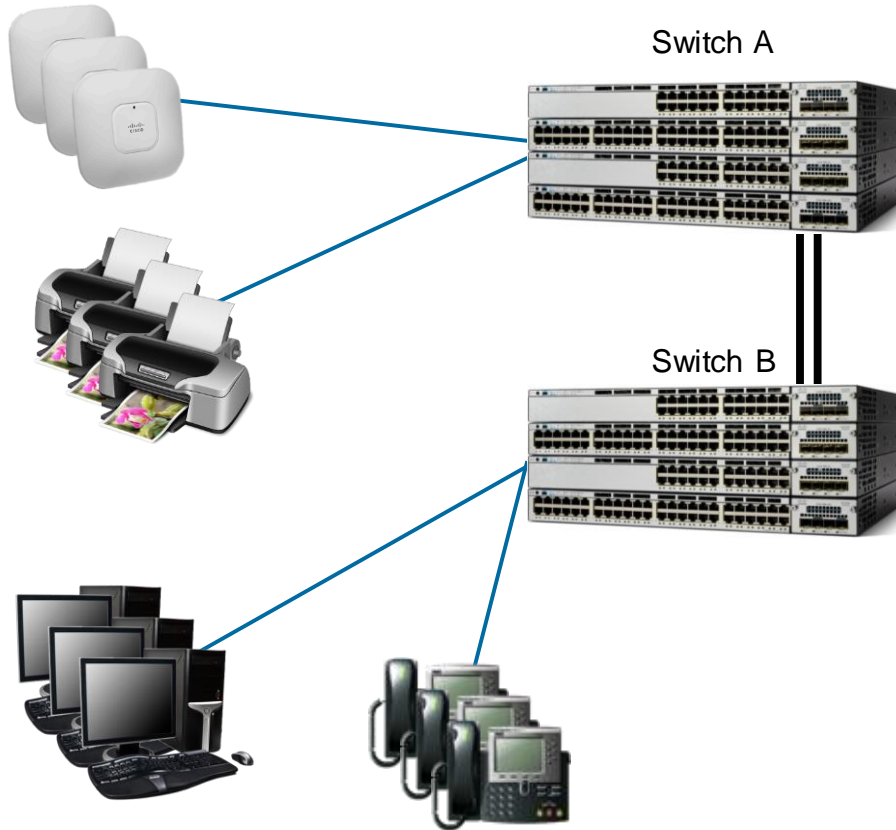
Auto Smartports

Poll Question

Who is using Auto Smartports today?

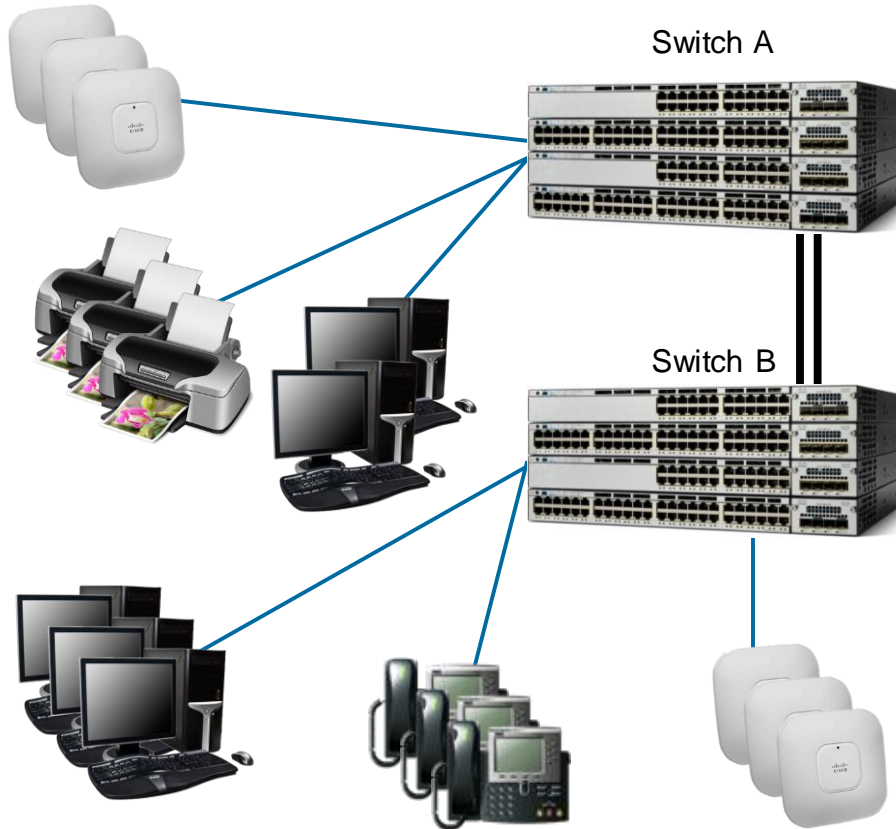


Auto Smartports – Deployment Examples



- Today
 - Interface configuration is static
 - Devices bound to specific interfaces because of config
 - Access Points connect to switch A
 - Everything else connects to Switch B
 - Available ports in Switch A
- Be careful which cables connect to which switch interfaces!

Auto Smartports – Using Auto Smartports



- With Auto Smartports, No hard binding between device and Interface
- Devices connect anywhere
- IOS applies the configuration dynamically
- Configuration matches with type of device (Consistency 😊)
- Over time, devices balance on switches in IDF
- Balance Access Points across physical switches

Auto Smartports – How it Works

1. ASP snoops incoming packets for
 - Source MAC Address
 - CDP – Cisco Discovery Protocol
 - LLDP – Link Layer Discovery Protocol
 - DHCP Discover from end device
2. Uses Above to classify Device Type
3. Device Type triggers the macro to an interface
 - Macro = set of interface level CLI commands.
 - Built-in Macro's for well known devices

Auto Smartports – How it Works (Example)

Order of events for IP Phone attachment and configuration applied



- Attach Phone
- Power up via POE
- CDP/LLDP Exchange
- Get Voice VLAN Config
- Register with CUCM

- Phone is Attached
- Provide PoE as requested
- CDP/LLDP Exchange
- Classifies Device as IP Phone
- Apply Macro

Contents of Macro:

- Voice and Data VLAN plus QoS
- Cisco best practice for security

Auto Smartports – Built in Device Macros

Built in Devices with Macros

Switch# show macro auto device ?

access-point	Display auto configuration information for the autonomous access point
ip-camera	Display auto configuration information for the video surveillance camera
media-player	Display auto configuration information for the digital media player
phone	Display auto configuration information for the phone device
router	Display auto configuration information for the router device
switch	Display auto configuration information for the switch device

Macro Contents – IP Phone

Interface Configuration of Cisco_Phone_Auto_SmartPort

Switch# show run interface Gig 1/0/6

```
interface GigabitEthernet1/0/6
  switchport access vlan 10
  switchport mode access
  switchport block unicast
  switchport voice vlan 11
  switchport port-security maximum 3
  switchport port-security maximum 2 vlan access
  switchport port-security
  switchport port-security aging time 1
  switchport port-security violation restrict
  switchport port-security aging type inactivity
  load-interval 30
  srr-queue bandwidth share 10 10 60 20
  queue-set 2
  priority-queue out
  mls qos trust device cisco-phone
  mls qos trust cos
  macro description CISCO_PHONE_EVENT
  auto qos voip cisco-phone
```

Cisco Best Practices for IP Phone

... Continued

```
storm-control broadcast level pps 1k
storm-control multicast level pps 2k
storm-control action trap
spanning-tree portfast
spanning-tree bpduguard enable
service-policy input AutoQoS-Police-
  CiscoPhone
ip dhcp snooping limit rate 15
!
```

Auto Smart Ports – The Basics

- Built-in Macros have default vlan id.

- Change vlan id for built-in macros

```
Switch(config)#macro auto execute CISCO_PHONE_EVENT builtin \  
    CISCO_PHONE_AUTO_SMARTPORT VOICE_VLAN=10 ACCESS_VLAN=3  
(repeat for all devices or builtin macros)
```

- Use LAST_RESORT MACRO for Unclassified Devices

- Applied to interface that has no matches (eg: laptops)

```
Switch(config)#macro auto global control trigger last-resort  
Switch(config)#macro auto execute CISCO_LAST_RESORT_EVENT builtin \  
    CISCO_LAST_RESORT_SMARTPORT ACCESS_VLAN=data_vlan
```

- Enable Auto Smart Ports – Last step

```
Switch(config)# macro auto global processing
```

Auto Smart Ports – What Macro Has Been Applied

```
Switch# show macro auto interface
```

```
Global Auto Smart Port Status
```

```
Auto Smart Ports Enabled
```

```
Fallback : CDP Disabled
```

Interface	Auto Smart Port	Fallback	Macro Description(s)
Vl1	TRUE	None	No Macro Applied
Vl10	TRUE	None	No Macro Applied
Fa0	TRUE	None	No Macro Applied
Gi1/0/1	TRUE	None	No Macro Applied
Gi1/0/2	TRUE	None	CISCO_WIRELESS_AP_EVENT
Gi1/0/3	TRUE	None	No Macro Applied
Gi1/0/4	TRUE	None	CISCO_LAST_RESORT_EVENT
Gi1/0/5	TRUE	None	HP_printer_OUI macro
Gi1/0/6	TRUE	None	CISCO_CUSTOM_EVENT
Gi1/0/7	TRUE	None	CISCO_PHONE_EVENT

```
.  
. .  
. .
```



laptop

Really Quick Notes

Auto Smart Ports

- Available since IOS 12.2(55) SE or later
- Configure ports on a switch based on the end device connected to it
- Macro based solution



AutoConf

- Available since IOS 15.2(2)E and IOS-XE 3.6 or later
- Automatic configuration of physical switch interface based on device
- Session templates for access session
- Template based solution



Cisco live!

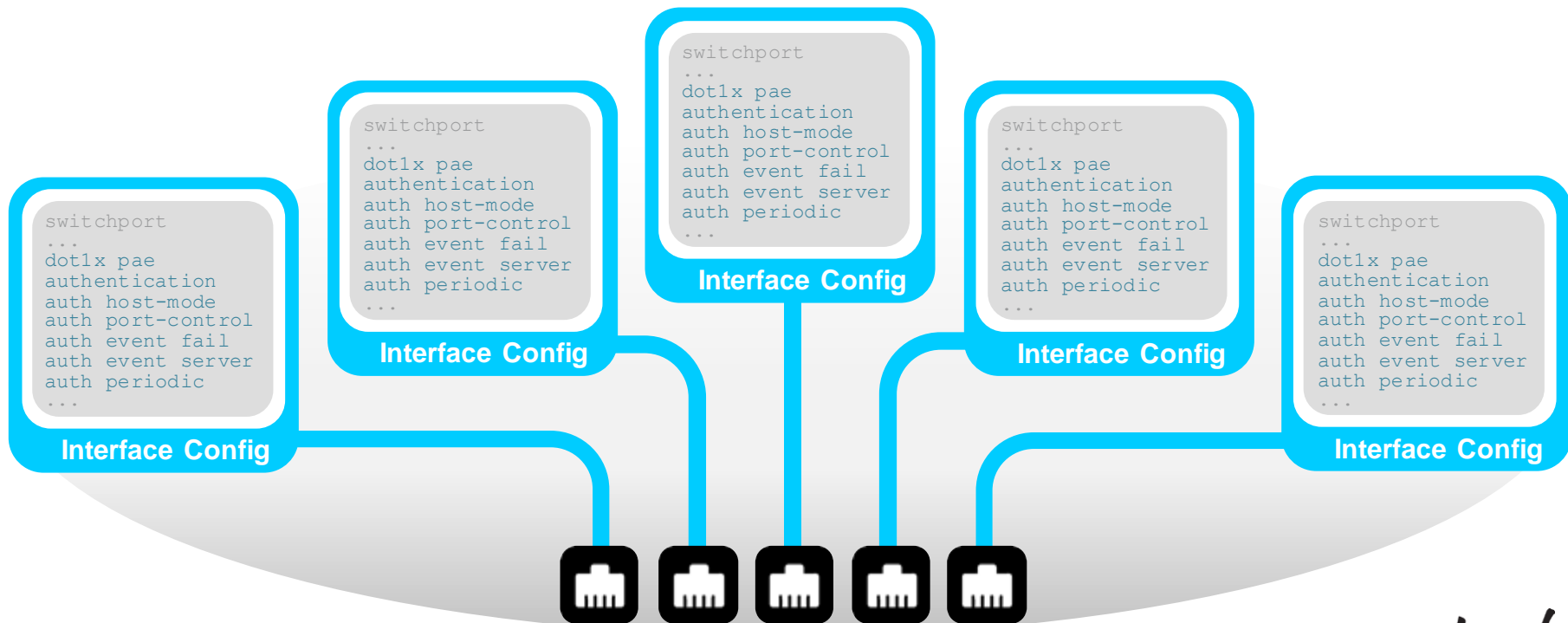


AutoConf

Cisco *live!*

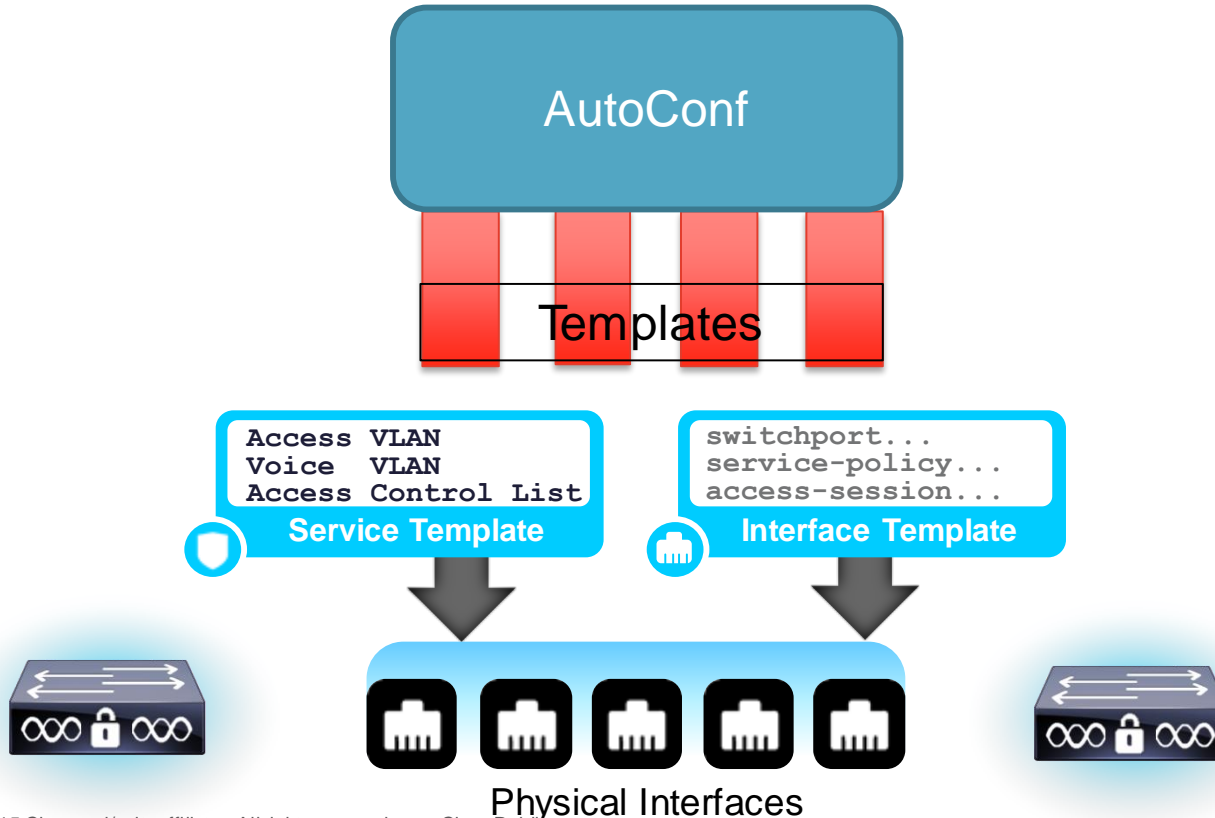
AutoConf – Taking Automation A Step Further

Per port configurations – No More!

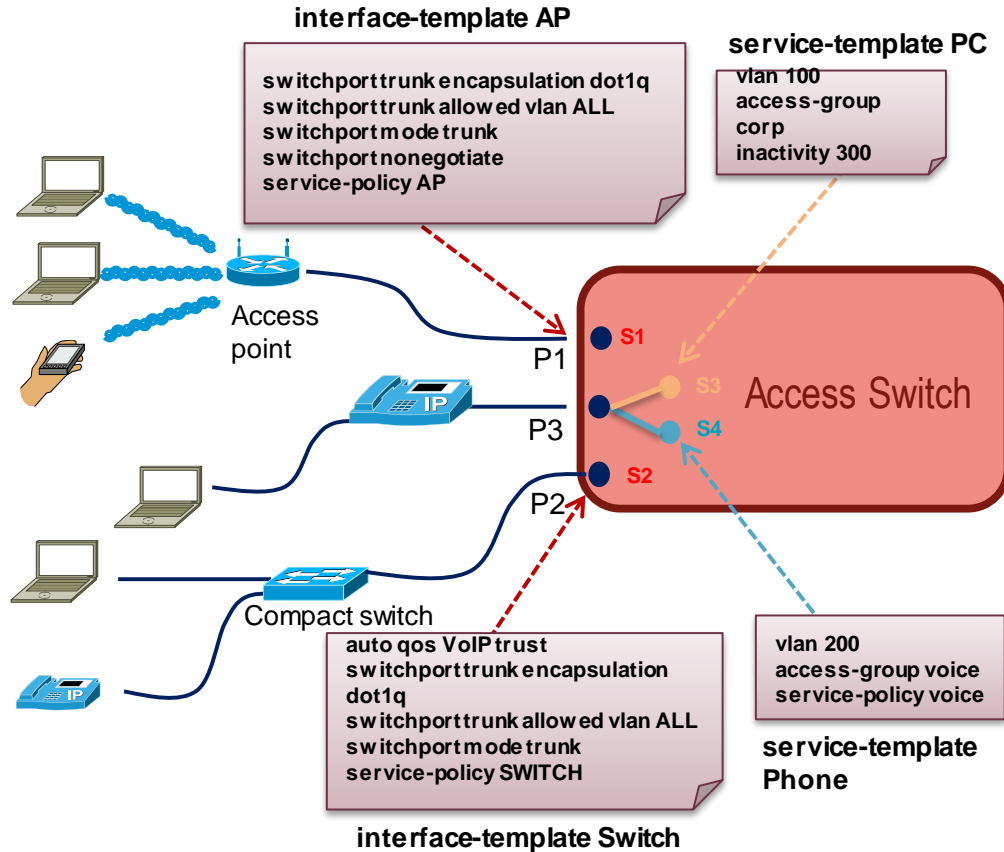


AutoConf – Interface Templates Relationship

Templates and Policies



Autoconf – Campus Use Case



Interface Templates

- Activated on **INTERFACES**
- Auto-conf the network device (one per port) e.g. Switch or AP
- Template impacts all the traffic via that interface
- Stays ON as long as activated

Service Templates

- Activated on **NETWORK SESSIONS**
- Template impacts only the control or data packets to the session
- No impact on other sessions sharing port
- Stays ON as long as the session exists

AutoConf: Default Hierarchy

- **BUILTIN_AUTOCONF_POLICY** - AutoConf policy that identifies parameter map

All builtin by default

Container relationship

```
3750X# show policy-map type control subscriber BUILTIN_AUTOCONF_POLICY
```

```
BUILTIN_AUTOCONF_POLICY
  event identity-update match-all
  10 class always do-until-failure
  10 map attribute-to-service table BUILTIN_DEVICE_TO_TEMPLATE
```

```
3750X# show parameter-map type subscriber attribute-to-service all
```

```
Parameter-map name: BUILTIN_DEVICE_TO_TEMPLATE
Map: 10 map device-type regex "Cisco-IP-Phone"
Action(s):
  20 interface-template IP_PHONE_INTERFACE_TEMPLATE
Map: 20 map device-type regex "Cisco-IP-Camera"
Action(s):
  20 interface-template IP_CAMERA_INTERFACE_TEMPLATE
Map: 30 map device-type regex "Cisco-DMP"
Action(s):
  20 interface-template DMP_INTERFACE_TEMPLATE
```

AutoConf Policy

Parameter Map

Mapping Device type A to
interface template X

Mapping Device type B to
interface template Y

Mapping Device type C to
interface template Z

Interface Templates: Built-in Templates

- Built-in Templates based on common end devices

```
3750X# show template interface brief
```

Template-Name	Source	Bound-to-Interface
-----	-----	-----
AP_INTERFACE_TEMPLATE	Built-in	No
DMP_INTERFACE_TEMPLATE	Built-in	No
IP_CAMERA_INTERFACE_TEMPLATE	Built-in	No
IP_PHONE_INTERFACE_TEMPLATE	Built-in	No
LAP_INTERFACE_TEMPLATE	Built-in	No
MSP_CAMERA_INTERFACE_TEMPLATE	Built-in	No
MSP_VC_INTERFACE_TEMPLATE	Built-in	No
PRINTER_INTERFACE_TEMPLATE	Built-in	No
ROUTER_INTERFACE_TEMPLATE	Built-in	No
SWITCH_INTERFACE_TEMPLATE	Built-in	No
TP_INTERFACE_TEMPLATE	Built-in	No

Interface Templates: Static Apply an Interface Template with “Source”

- Statically apply Interface template with “source <templatename>” on interface
- Full interface configuration use “show derived-config interface <intf>”
- Template name appears in “show running interface <intf>”
- By default, access vlan is 1.
 - Modify built-in to change

```
3750X(config-if)#source template DMP_INTERFACE_TEMPLATE
3750X(config-if)# end
```

```
3750X# show derived-config interface Gig 1/0/10
!
interface GigabitEthernet1/0/10
  switchport mode access
  switchport access vlan 10
  switchport block unicast
  switchport port-security
  srr-queue bandwidth share 1 30 35 5
  priority-queue out
  mls qos trust dscp
  spanning-tree portfast
  spanning-tree bpduguard enable
end
```

```
3750X# show run interface Gig 1/0/10
Building configuration...
```

```
Current configuration : 79 bytes
!
interface GigabitEthernet1/0/10
  source template DMP_INTERFACE_TEMPLATE
```

Interface Templates: Modified Built-in Templates

- Modified templates distinguished from original built-in
- Easy to determine template in use

```
3750X# show template interface brief
```

Template-Name	Source	Bound-to-Interface
-----	-----	-----
AP_INTERFACE_TEMPLATE	Built-in	No
DMP_INTERFACE_TEMPLATE	Modified-Built-in	Yes
IP_CAMERA_INTERFACE_TEMPLATE	Built-in	No
IP_PHONE_INTERFACE_TEMPLATE	Built-in	No
LAP_INTERFACE_TEMPLATE	Built-in	No
MSP_CAMERA_INTERFACE_TEMPLATE	Built-in	No
MSP_VC_INTERFACE_TEMPLATE	Built-in	No
PRINTER_INTERFACE_TEMPLATE	Built-in	No
ROUTER_INTERFACE_TEMPLATE	Built-in	No
SWITCH_INTERFACE_TEMPLATE	Built-in	No
TP_INTERFACE_TEMPLATE	Built-in	No



Modified
template

Interface Templates: Create Your Own Template

- Easy to create your own template and apply.
- Non builtin called “user”
- Apply “user” is same as builtin

```
3750X# configure term
3750X(config)# template APPLE_TV_INTF_TEMPLATE
3750X(config-template)# switchport access vlan 33
3750X(config-template)# spanning-tree portfast
3750X(config-template)# switchport mode access
3750X(config-template)# mls qos trust dscp
3750X(config-template)# description Apple TV
3750X(config-template)# exit
```

```
3750X#
3750X# show template brief
Interface Templates
=====
```



New template

Template-Name	Source	Bound-to-Interface
-----	-----	-----
APPLE_TV_INTF_TEMPLATE	User	No
AP_INTERFACE_TEMPLATE	Built-in	No
DMP_INTERFACE_TEMPLATE	Modified-Built-in	Yes
IP_CAMERA_INTERFACE_TEMPLATE	Built-in	No

AutoConf In Action: Dynamic Binding to Interface

- After IP Phone connected to Interface Gi1/0/2
- No change to running configuration
 - Show run int <intf>



```
3750X# show run interface gi1/0/2
Current configuration : 38 bytes
!
interface GigabitEthernet1/0/2
end
```


AutoConf In Action: Dynamic Binding to Interface

- After IP Phone connected to Interface Gi1/0/2
 - Full Configuration displayed with derived command
 - `show derived int <intf>`



```
3750X# show derived int gi1/0/2
Derived configuration : 616 bytes
!
interface GigabitEthernet1/0/2
  switchport access vlan 10
  switchport mode access
  switchport block unicast
  switchport voice vlan 11
  switchport mode access
  switchport port-security maximum 3
  switchport port-security maximum 2 vlan access
  switchport port-security aging time 1
  switchport port-security aging type inactivity
  switchport port-security violation restrict
  switchport port-security
  load-interval 30
  srr-queue bandwidth share 1 30 35 5
  priority-queue out
  mls qos trust cos
  storm-control broadcast level pps 1k
  storm-control multicast level pps 2k
  storm-control action trap
  spanning-tree portfast
  spanning-tree bpduguard enable
  ip dhcp snooping limit rate 15
end
```

AutoConf In Action: Dynamic Binding to Interface

- What template is bound to interface?
 - Show template interface binding
 - show template binding

```
3750X# show template interface binding all
```

Template-Name	Source	Method	Interface
IP_PHONE_INTERFACE_TEMPLATE	Built-in	dynamic	Gi1/0/2

```
3750X# show template binding target gi1/0/2
```

```
Interface Templates
```

```
=====
```

```
Interface: Gi1/0/2
```

Method	Source	Template-Name
dynamic	Built-in	IP_PHONE_INTERFACE_TEMPLATE

```
Service Templates
```

```
=====
```

```
Interface: Gi1/0/2
```

Session	Source	Template-Name
-----	-----	-----



Templates: Highlights

- Service Templates applied to Access Sessions
 - Interface Templates applied to physical ports
- Service Template configuration only impacts session traffic
 - No impact to other sessions on same physical port.
- Use Service Template on non physical interfaces
 - For WLAN
 - SVI
 - Authenticated Sessions (eg: User Auth, MAC Auth)
 - Wired and wireless

Things to Remember

- Built-in templates must be modified for vlan config
 - All templates default to access vlan 1 ☹️
 - Switchport access vlan X
 - Switchport voice vlan Y
 - Switchport trunk native vlan Z
- Once modified, built-in templates show in running and startup config
- Show-derived interface to see applied template configuration
- Interface Templates are not supported on EtherChannels, so not AutoConf
- AutoConf enabled on all interfaces by default
 - Explicitly disable on interface “**access-session inherit disable autoconf**”

Benefits of AutoConf

- Config File Readability and Manageability
- Smaller configuration files
- Built-in Interface Templates for ease of use
- All Interface Templates are customisable.
- Advantages over Auto Smart Ports
 - Templates updates immediately ripple to interfaces
 - Per session or per port templates
 - No change to running-config
 - Full rollback and precedence management
 - Compatible with Session Networking/AutoConf

A long-exposure photograph of a city street at night. The foreground is dominated by vibrant, multi-colored light trails from moving vehicles, creating a sense of motion and energy. In the background, a modern urban landscape is visible, featuring a prominent pedestrian bridge with blue lighting and several illuminated buildings. The overall scene is a blend of dynamic light patterns and static architectural elements.

Migration From ASP to AutoConf

Things to Know

- Only ASP or Autoconf should be running on an interface
- If user defined ASP macros are in use
 - new templates will have to be created to match.

Scenario 1: This simple use case describes a user migrating from ASP to Autoconf on a global level, with fully built-in implementation.

1. Verify ASP is running

```
Switch# show running-config | include macro auto global  
"macro auto global processing"
```

2. Disable ASP on a global level

```
Switch(config)# no macro auto global processing
```


Scenario 1: Continued.

3. To ensure removal of all ASP macros, administratively enable/disable interfaces where allowed
Switch(config)#interface range gigabit 0/1-24
Switch(config)#shutdown
Switch(config)#no shutdown
4. Enable Autoconf on a global level
Switch# autoconf enable

Scenario 2: This simple use case describes a user adding the Autoconf solution while still running ASP on specific interfaces.

1. Verify ASP is running
Switch# show running-config | include macro auto global
“macro auto global processing”
2. Disable AutoConf on interfaces where current macro is desired
Switch(config)# interface gigabit 0/5
Switch(config-if)#access-session inherit disable AutoConf

Scenario 2: Continued

3. Disable ASP on all interfaces where AutoConf is desired

```
Switch(config)#interface range gigabit 0/1-4, gigabit 0/6-24
```

```
Switch(config-if-range)#no macro auto processing
```

```
Switch(config-if-range)#shutdown
```

```
Switch(config-if-range)#no shutdown
```

4. Enable Autoconf on a global level

```
Switch(config)# autoconf enable
```

5. Check running and derived configurations to ensure proper configs are applied

```
Switch# show running-config interface gigabit 0/5
```

```
Switch# show derived-config
```



Automated Virtual Network



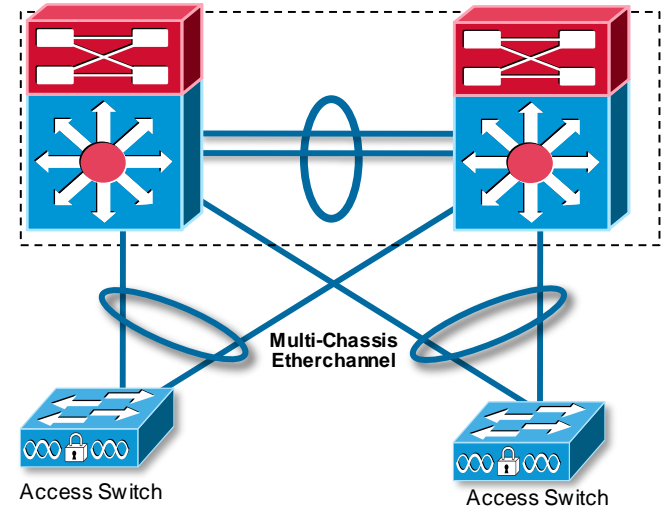
Easy VSS

Cisco *live!*

Easy VSS

Problem with Traditional VSS

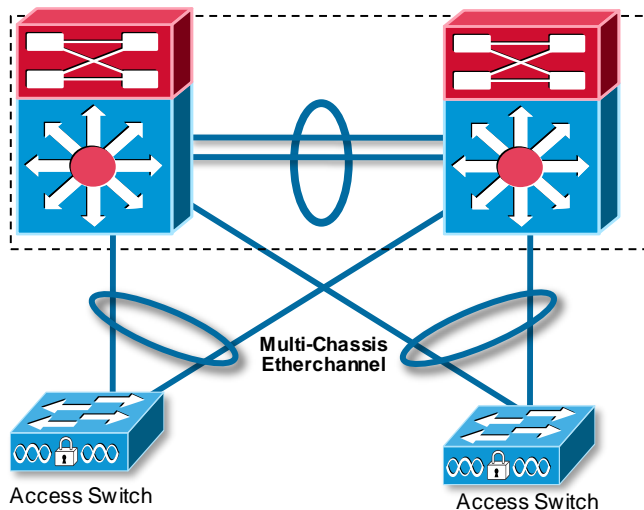
- Up to 30 lines of configuration
- Configuration on both Active & Standby
- Error Prone
- Version Mismatch – More manual tasks



Easy VSS

Easy VSS Configuration

- 1 Line – ‘switch convert mode easy-vss’
- Zero touch on Standby (No Config Needed)
- Mismatch Discovery & Fix
- Needs an L3 Reachability to the pair for communication
- Option to choose VSL Link



```
#(easy-vss)#VSL ?  
Local Interface      Remote Interface      Hostname      Standby-IP  
GigabitEthernet3/5  TenGigabitEthernet1/1  4K-DEMO      2.2.2.4  
GigabitEthernet3/6  TenGigabitEthernet1/2  4K-DEMO      2.2.2.4  
GigabitEthernet3/7  TenGigabitEthernet1/1  4K-DEMO2     2.2.2.5
```

Easy VSS

Traditional VSS Config

Switch 1

```
Switch-1(config)# switch virtual domain 100
Switch-1(config-vs-domain)# switch 1
Switch-1(config-vs-domain)# exit

Switch-1(config)# interface port-channel 10
Switch-1(config)# switchport
Switch-1(config-if)# switch virtual link 1
Switch-1(config-if)# no shutdown
Switch-1(config-if)# exit

Switch-1(config)# interface range tengigabitethernet 3/1-2
Switch-1(config-if)# channel-group 10 mode on

Switch-1# switch convert mode virtual
```

Switch 2

Switch 1

```
Switch-1# switch convert mode virtual
#(easy-vss)#VSL Te3/1 Te3/2
```

```
Switch-2(config)# switch virtual domain 100
```

```
Switch-2(config-vs-domain)# switch 2
Switch-2(config-vs-domain)# exit
```

```
Switch-2(config)# interface port-channel 20
Switch-2(config)# switchport
Switch-2(config-if)# switch virtual link 2
Switch-2(config-if)# no shutdown
Switch-2(config-if)# exit
```

```
Switch-2(config)# interface range tengigabitethernet 5/2-3
Switch-2(config-if)# channel-group 20 mode on
```

```
Switch-2# switch convert mode virtual
```

Switch 2

```
Switch-2(config)#
```




Conclusion

In Summary...

- Smart Operation Tools – Available today in IOS
- Smart Install – automates the process of installing switches (traditional and Next Gen)
- Auto Smartports – Device based automated configuration
- Templates are easy and there to save time with AutoConf
- Migration scenarios from ASP to AutoConf

Final Thoughts

Where can you apply more automation in your networks?

“If you spend too much time thinking about a thing, you’ll never get it done.”

- Bruce Lee



Q&A

Cisco *live!*

Complete Your Online Session Evaluation

Give us your feedback and receive a Cisco Live 2015 T-Shirt!

Complete your Overall Event Survey and 5 Session Evaluations.

- Directly from your mobile device on the Cisco Live Mobile App
- By visiting the Cisco Live Mobile Site
<http://showcase.genie-connect.com/clmelbourne2015>
- Visit any Cisco Live Internet Station located throughout the venue

T-Shirts can be collected in the World of Solutions on Friday 20 March 12:00pm - 2:00pm



Learn online with Cisco Live!

Visit us online after the conference for full access to session videos and presentations. www.CiscoLiveAPAC.com



Thank you.

Cisco *live!*



CISCO