2020/2/4

Security Level:

VRP8 Platform Introduction

www.huawei.com

Forewords

- Restricted by software and hardware technologies, earlier operating systems of network devices use monolithic models. The software is compiled into an executive file and executed by an embedded operating system. Only the single-CPU hardware is available, providing integrated control management and running all protocols and management data on one node. With the development and popularization of Internet technologies as well as IP orientation of carriers' networks, network devices are evolved from single boxes to single chassis and then to multichassis, with single core CPUs replaced with multi-core CPUs.
- Following the development trend, to provide higher reliability for carriers' networks and fully use hardware processing capabilities of the cluster, Huawei develops the VRP8 based on existing Versions.

Objectives

- Upon completion of this course, you will be able to:
  - Know history of Huawei VRP platform.
  - Know the features of VRP8.
  - Know the basic operation of VRP8.

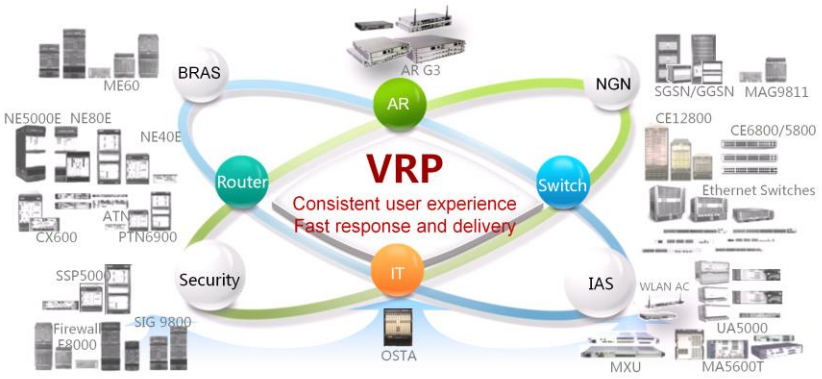
Contents

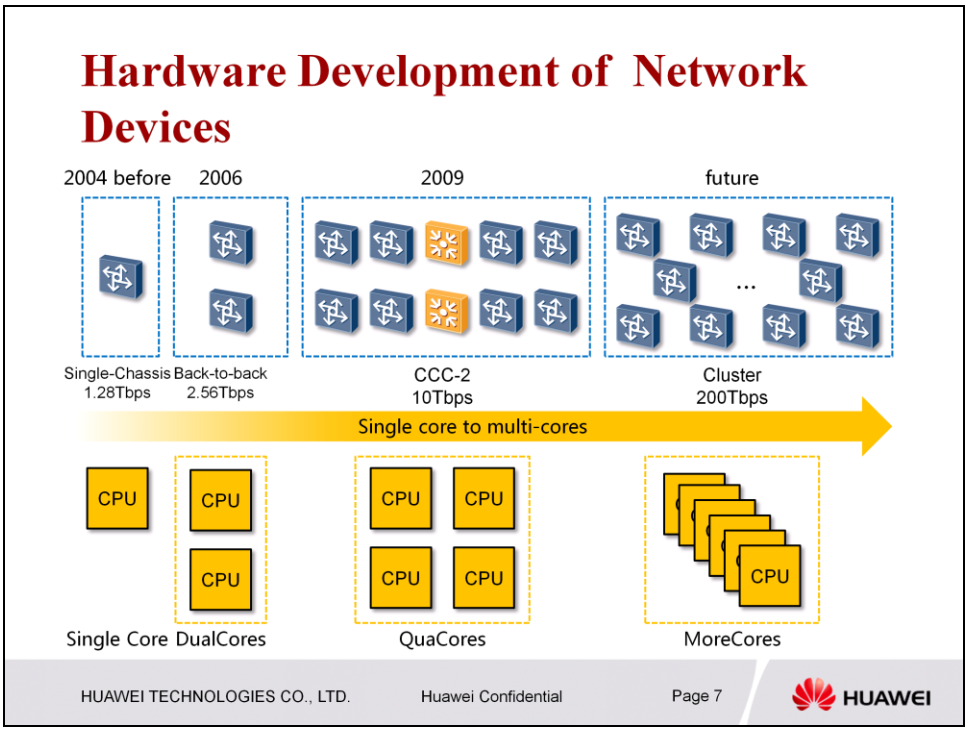
VRP Network Operation System

VRP Introduction

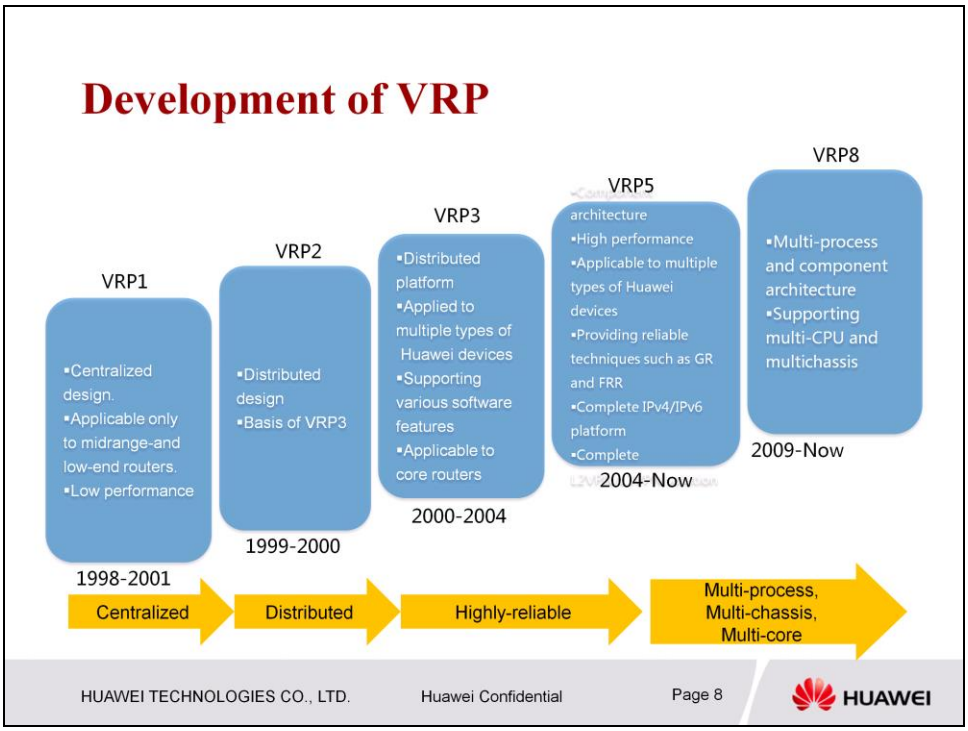
- VRP (Versatile Routing Platform) is a network OS running in IP devices, similar to iOS and Windows.
- VRP is the brain of IP devices which constructs the global network.
- VRP has high reliability which ensures IP network secure and stable operation.

VRP Serves Multi-product Families





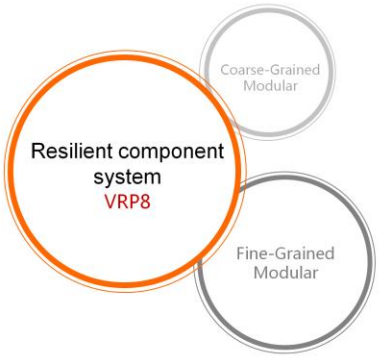
- Restricted by software and hardware technologies, earlier operating systems of network devices use monolithic models. The software is compiled into an executive file and executed by an embedded operating system. Only the single-CPU hardware is available, providing integrated control management and running all protocols and management data on one node. With the development and popularization of Internet technologies as well as IP orientation of carriers' networks, network devices are evolved from single boxes to single chassis and then to multichassis, with single core CPUs replaced with multi-core CPUs.



- The VRP5 is a distributed network operating system, and features high extensibility, reliability,and performance. Currently, network devices running VRP5 are serving more than 50 carriers worldwide. The VRP5 provides various features and its stability has been approved by the market.
- The VRP8 is a new-generation network operating system, which has a distributed, multi-process, and component architecture. The VRP8 supports distributed applications and virtualization techniques. It follows the hardware development trend and meets the exploding service requirements of carriers in five to ten years.

Contents

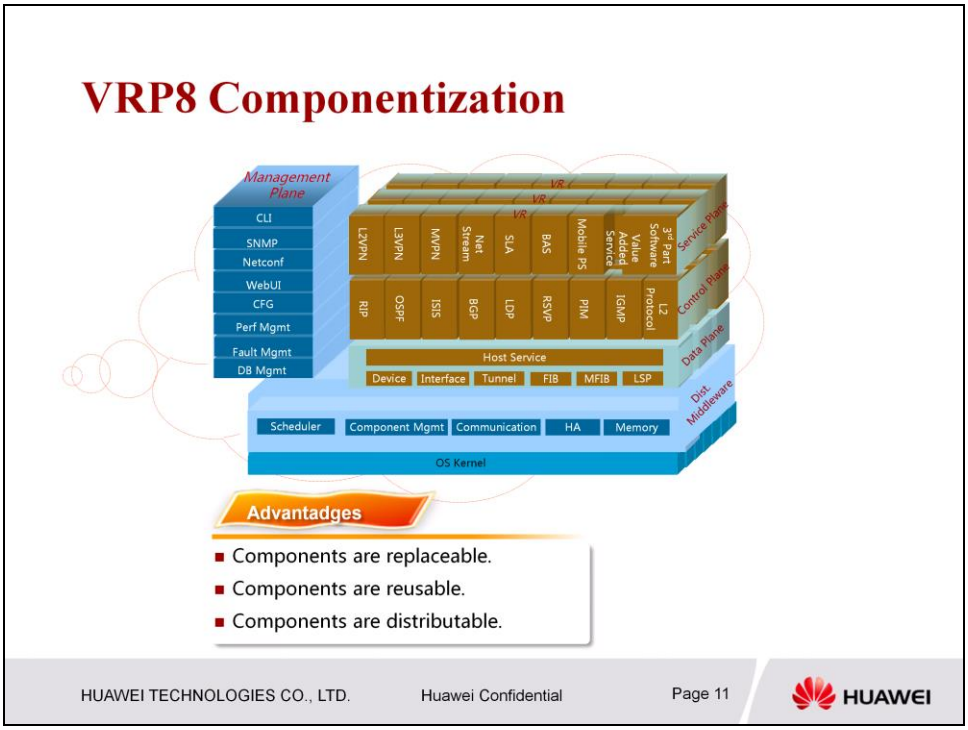
VRP8 Features



VRP8 Features

- Multi-core CPU or multi-process.
- Distributed applications.
- NSx (NSS, NSR, NSB and NSF) and In-Service Software Upgrade (ISSU).
- Netconf, and two-phase configuration commit and configuration rollback in the case of a commit failure to facilitate user management and enhanced carrier-class management and maintenance.

- Coarse-Grained Modular
  - Distributed process capability partially
  - Better reliability
  - Better scalability
- Resilient component system
  - Parallel and distributed architecture of full services
  - Non-Stop Anything
  - Cloud-routing capability
  - Flexible virtualization
  - High APSO

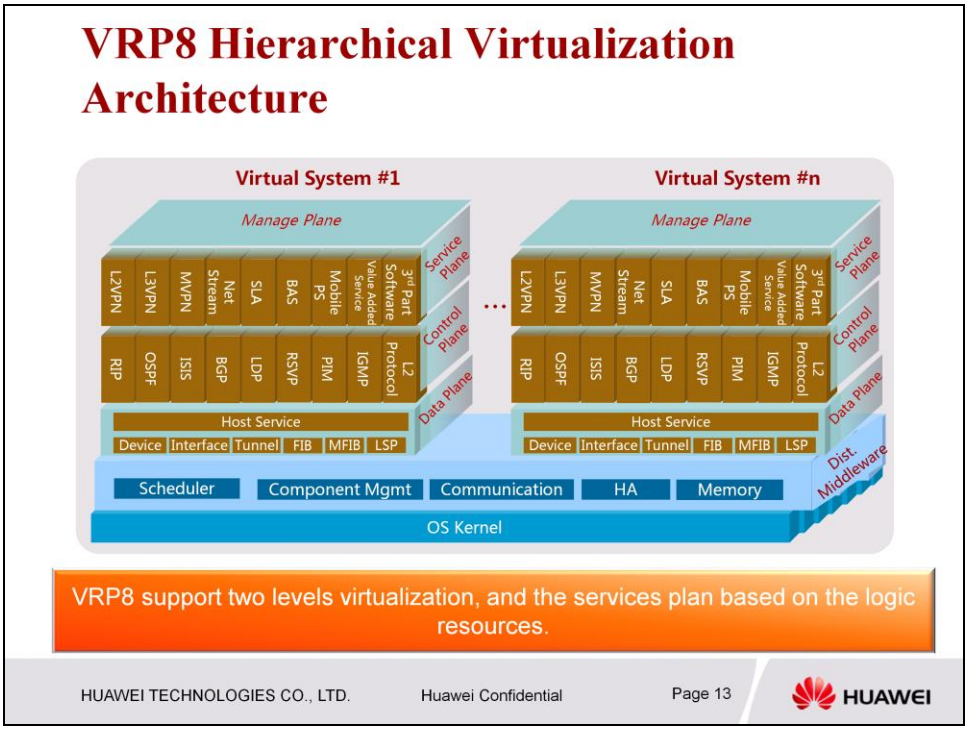


- Componentization refers to the method of encapsulating associated functions and data into a software module, which is instantiated to function as a basic unit of communication scheduling.
- The VRP8 architecture design is component-based. The entire system is divided into multiple independent components that communicate by using interfaces. One component provides services for another component by using an interface, and the served component does not need to know how the serving component provides services.
- The component-based architecture design has the following advantages:
  - Components are replaceable.
    - A component can be replaced by another component as long as the substitute provides the same functions and services (on interfaces) as those of the replaced component. The new component can even use a different programming language. This enables the upgrade of a single component of the VRP8, including replacing and upgrading a component and adding components.
  - Components are reusable.
    - High-quality software components serve for a long time,

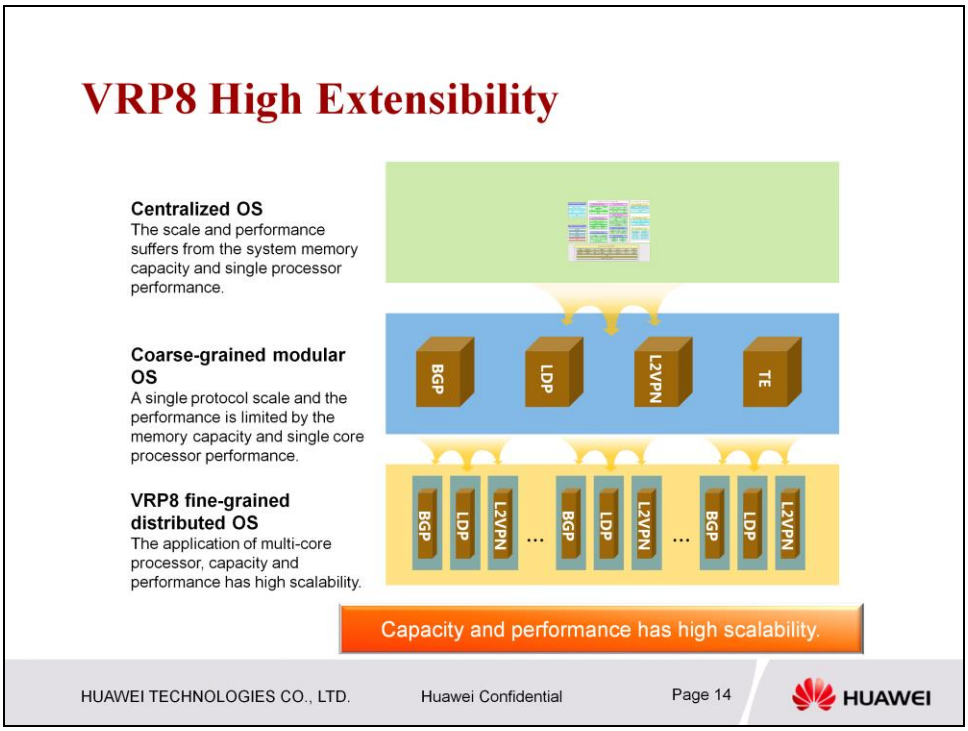
- and are stored in the software database. The VRP8 software model is able to be redefined and configured to customize a product architecture that is quite different from the hardware platform. In this manner, software features are customized.

❑ Components are distributable.

- VRP8 components are deployed in a distributed manner. Two relevant components are deployed on different nodes and they communicate with each other across networks. Component distribution is implemented without modifying components. Instead, only the data of related deployment policies needs to be modified.



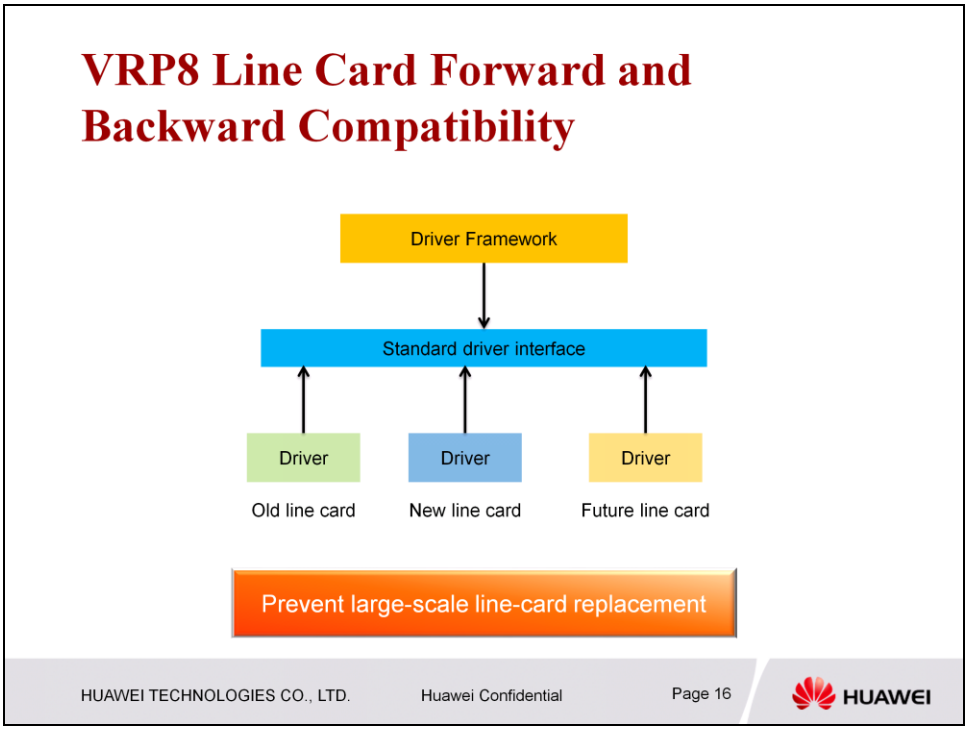
- A Physical Switch(PS) can be divided into multiple Virtual Switches(VS), each VS can be independently configured business, and multiple VSs shared line card physical interface resources, hardware resources, control plane processing capabilities.



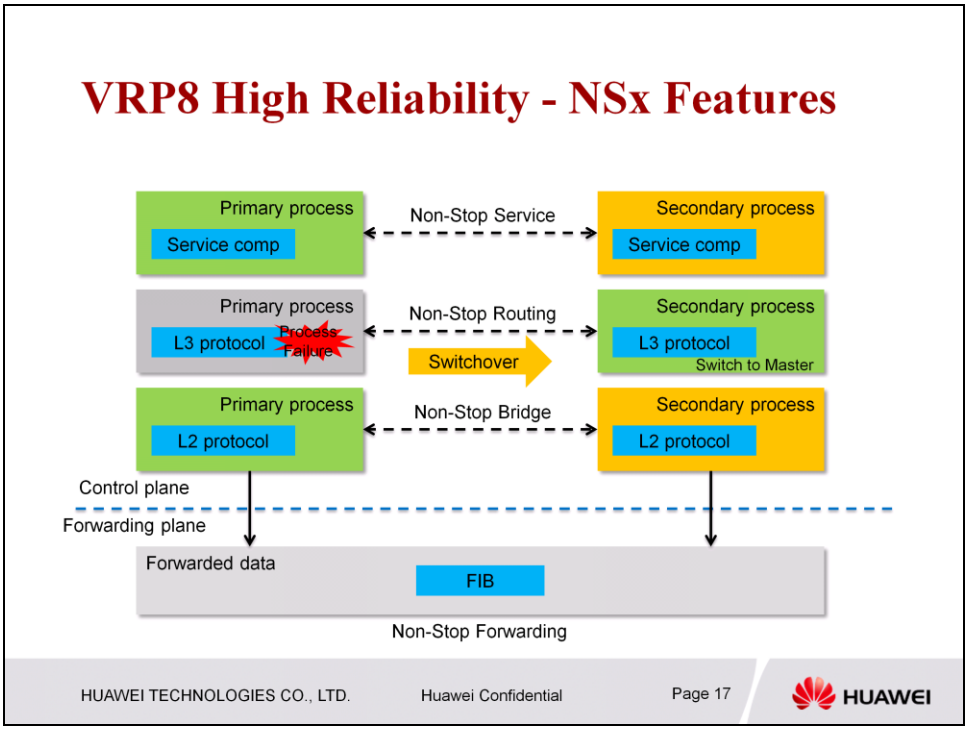
- To improve extensibility, the VRP8 supports backward compatibility and plug-and-play functionality on hardware line cards, allowing quick responses to users' demands. The VRP8 implements high expendability for the following items:
  - ❑ Line cards: Standard driver framework and plug-and-play are supported.
  - ❑ Software features: The data plane operates based on modules.
  - ❑ Capacity and performance: Services based on fine-granularity distribution are simultaneously processed.
  - ❑ l Operation and maintenance tools: The configuration plane is separate from the control plane.
- The trend is to utilize multi-chassis, multi-main control board, multi-CPU, and multi-core architectures in the development of the hardware on the existing core routers. The reason is that traditional integrated OS does not support modular service deployment or processing, and only depends on the processing capability of a single CPU with a single core. The second-generation OS supports coarse-granularity modules, allowing multiple protocol and service modules to simultaneously process services. These OSs, however, are incapable of supporting the

processing of protocol- and service-specific distributed instances and are still unable to take advantage of multi-CPU and multi-core processing capability. The VRP8 with the fine granularity distributed architecture, rotocol- and service-specific components allows a device to deploy services in distributed instances and to process services simultaneously. This helps a device to overcome the constraints of the single entity's processing capability and memory and to take advantage of integral hardware processing capability on the control plane, improving the sustainable extensibility of the device's performance and capacity.

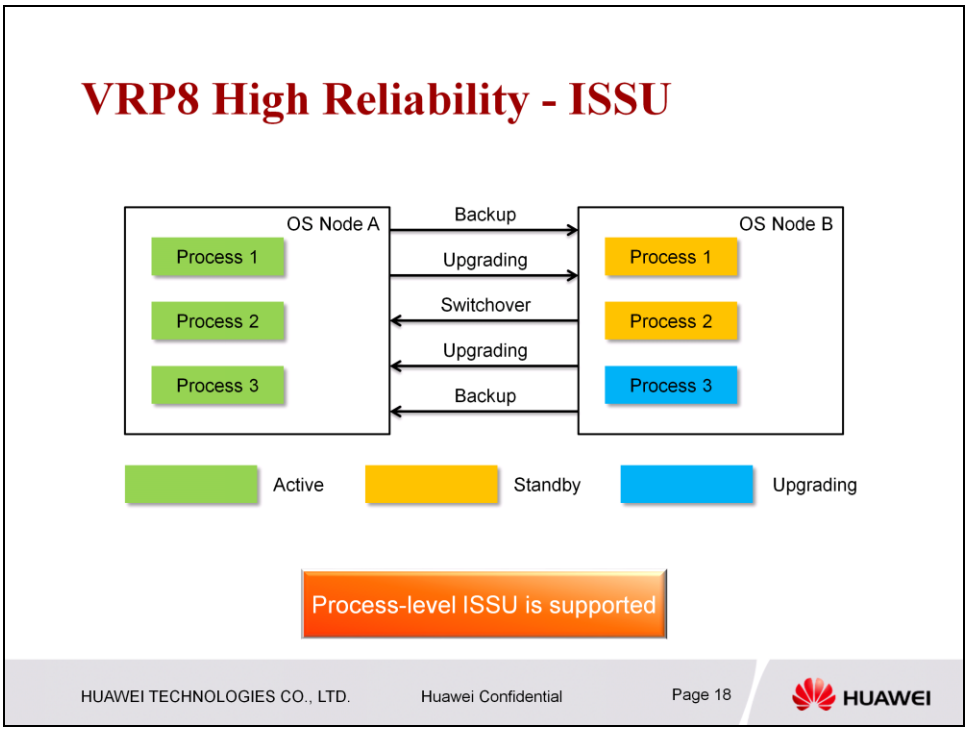
- On the VRP8, the data plane adopts a model-based forwarding technique. A mere change in the forwarding model, not in codes, allows a new function to be implemented or allows a function on the forwarding plane to be changed, enabling quick responses to carriers' demands.



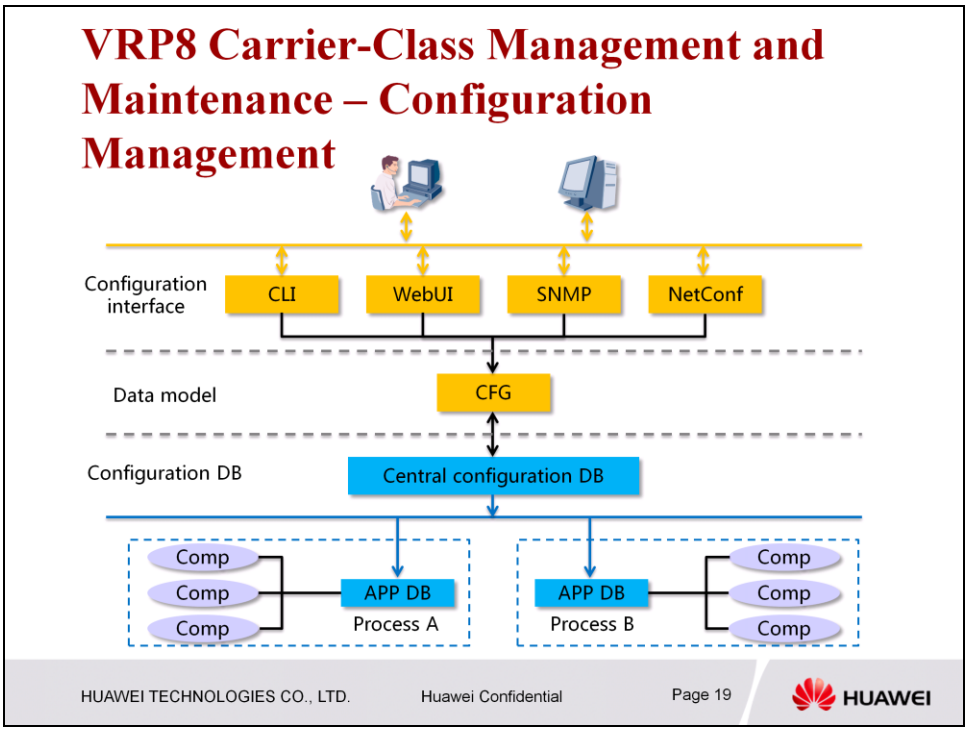
- To support various network interfaces, an IP network device usually supports a variety of line card types. The problem is that these cards historically needed to be replaced as technology progressed and chips were replaced and/or updated. To help carriers maximize the return on investment and prevent large-scale line-card replacement, software needs to support forward and backward compatibility of line cards. The VRP8 implements forward and backward compatibility on line cards over the standard driver framework with the help of software and hardware decoupling techniques.



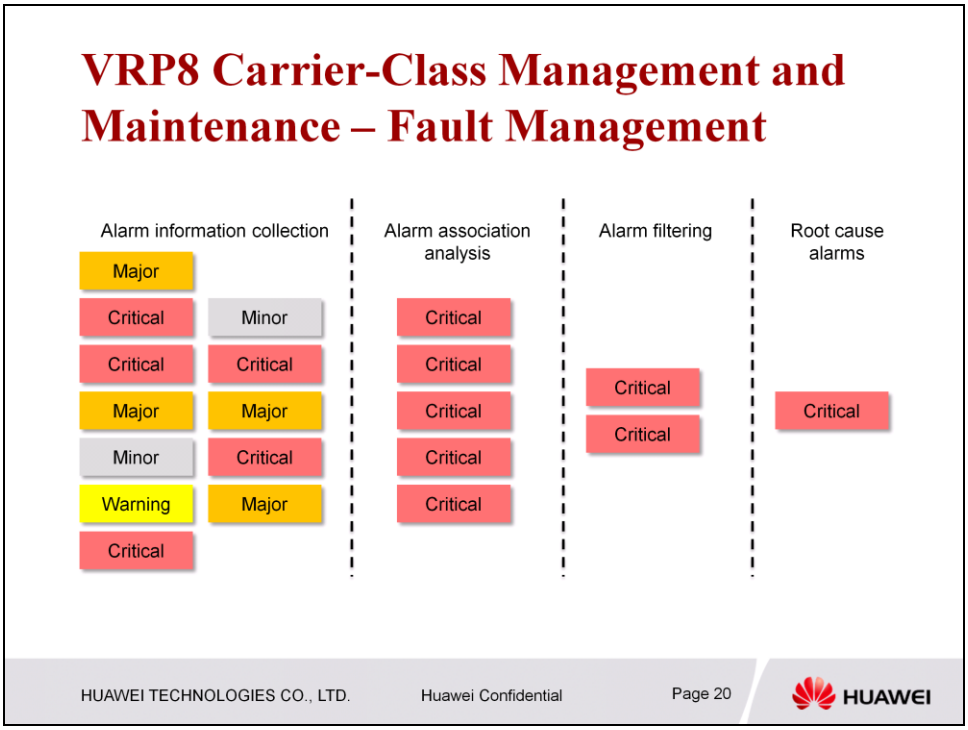
- On the VRP8, techniques such as Non-Stop x (NSx) and In-Service Software Upgrading (ISSU) are implemented to provide high reliability for services over networks, in line with carrier-class operation criteria.
- As shown in the figure, the unique VRP8 NSx features (NSS, NSR, NSB, and NSF) achieve non-stop service operation at all levels.



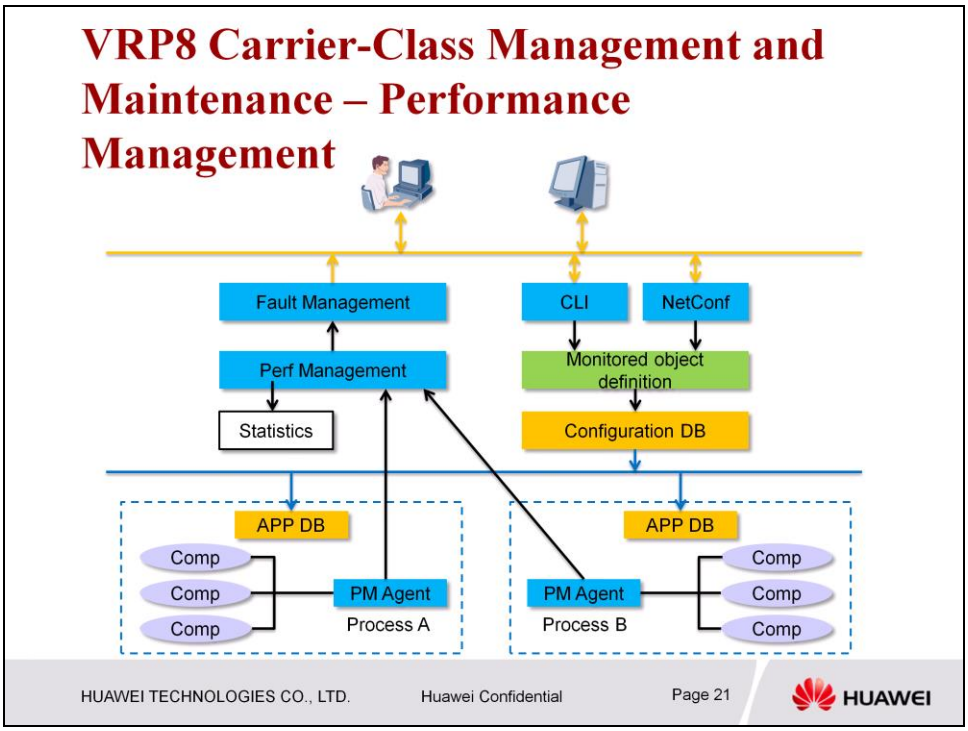
- Carrier-class devices continually provide non-stop and highly reliable services over networks. An essential point in the software architecture design, therefore, is how to maintain these services during a software upgrade. The VRP8 supports real-time process-level information hot backups and the forward compatibility of a backup interface. During a software upgrade, a slave node is upgrading the slave node, data and status information in the primary process running the old version on the master node are backed up to the slave node. After the backup, a primary/backup process switchover is performed, enabling the new primary process to take over services. Then, the slave node (previous master node) is upgraded. After the upgrade, a primary/backup process switchback is performed, achieving upgrade without interrupting services. The VRP8 architecture supports process-level ISSU. On a specific product running the VRP8, node-level (board-level) ISSU or process-level ISSU is supported. As shown in the figure.



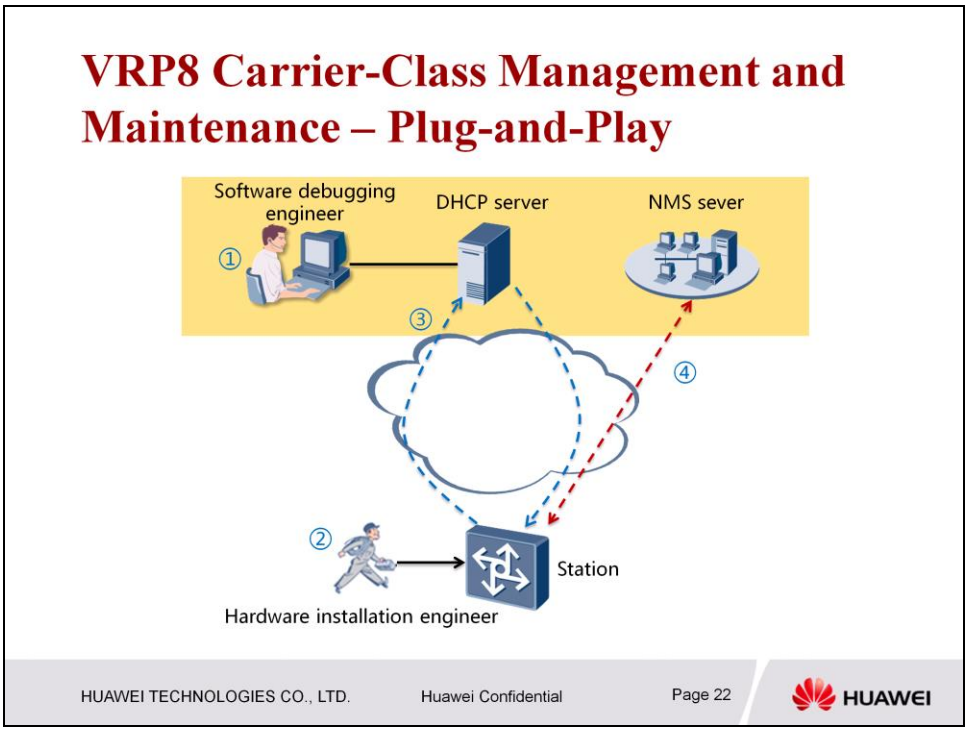
- A configuration interface layer provides a variety of configuration tools. A configuration tool parses a configuration request and then sends the request to a Configuration (CFG) component. The CFG component uses a pre-defined configuration information model to perform verification, association, and generation of configuration data. After a user commits a configuration and the configuration is successfully executed, configuration data is saved in a central database. A process-specific APP database obtains the configuration information from the central database.
- The VRP8 supports two-phase configuration commit and configuration rollback.
- The VRP8 management plane provides the following functions:
  - ❑ Support for a variety of existing configuration tools and more.
  - ❑ Implementation of model-based configuration.
  - ❑ Data verification and configuration rollback.
  - ❑ Database-assistant configuration data recovery.



- As shown in the figure, the VRP8 implements fault management based on service objects. The VRP8 creates a service object relationship model to analyze associations between alarms, filter out invalid alarms, and report root-cause-triggered alarms, speeding up fault identification.

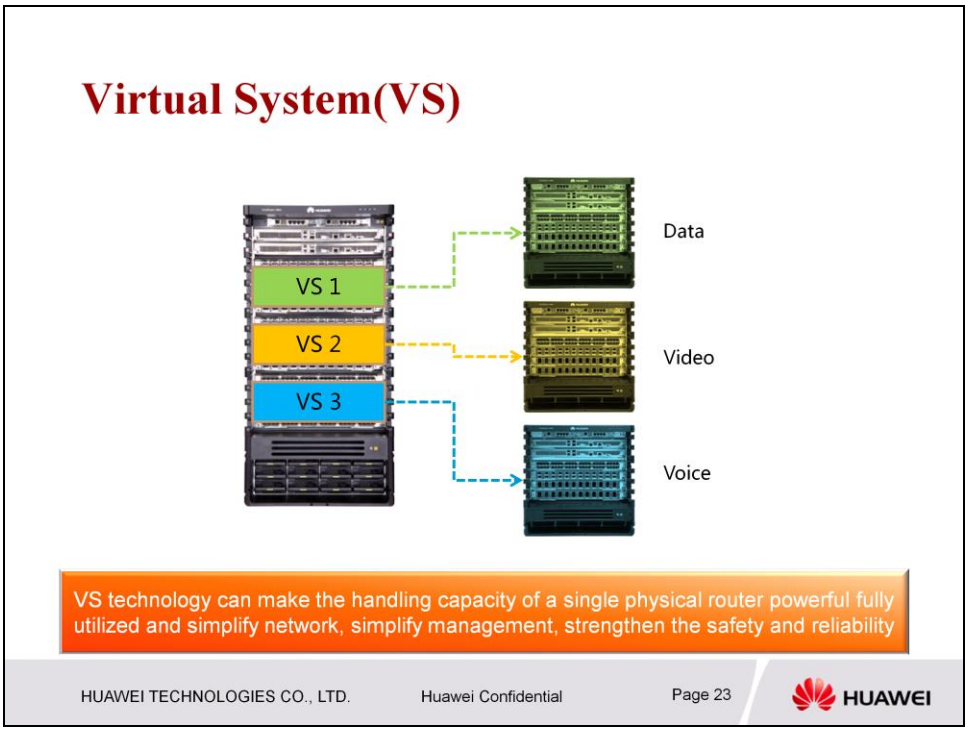


- As shown in the figure, the VRP8 provides a flexible performance management mechanism. Information about an object to be monitored, including a description of the object and a monitoring threshold, can be manually defined on a configuration interface and then configuration data is delivered by the center database. The APP component collects statistics about the configured object and sends them to a Perf Management server through a performance management (PM) agent. After receiving the statistics, the Perf Management server generates information about a fault based on the pre-defined object and monitoring threshold, and then sends information about the fault to the network management system (NMS) through a fault management center. Performance information can be viewed by running a command or through the NMS.

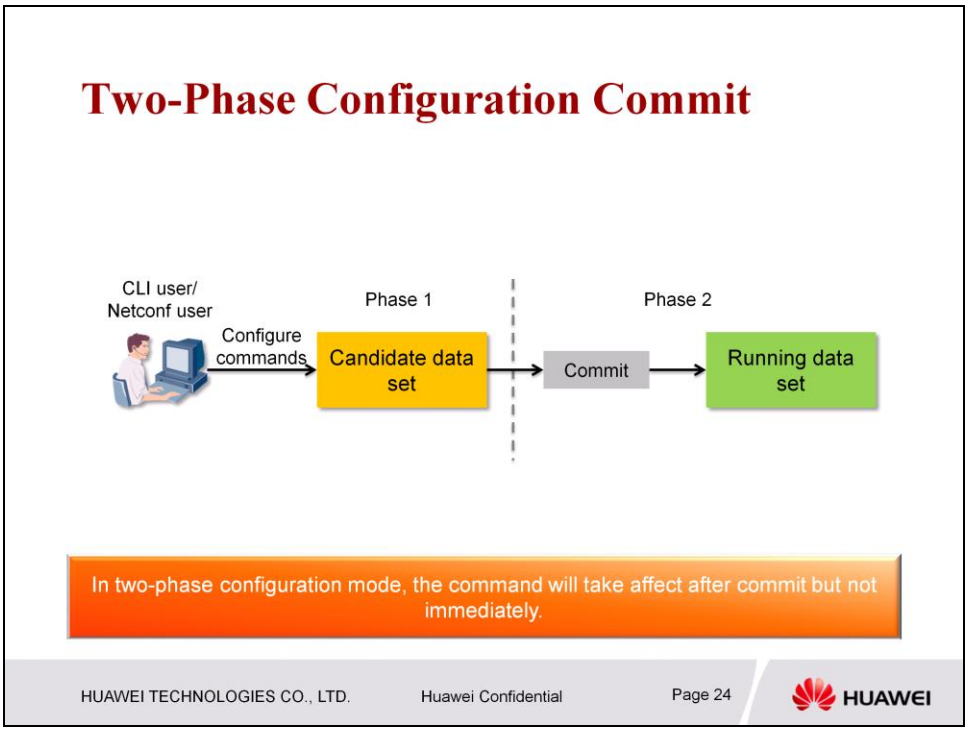


- As shown in the figure, VRP8 plug-and-play allows a great number of devices to be deployed on a site at a time and to be managed and maintained in remote mode, thus reducing OpEx.
- Devices supporting VRP8 plug-and-play are deployed as follows:

  1. Commissioning engineers import IP addresses and names of devices to be deployed to a DHCP server.
  2. Hardware installation engineers install devices and power them on.
  3. Devices automatically apply for IP addresses and initial configurations and the DHCP server assigns IP addresses and delivers initial configurations.
  4. The devices report their presence to the NMS and the NMS detects that the devices are online and then the commissioning engineers remotely commission the devices and configure services.



- 1:N virtualization divides a physical network facility into multiple isolated networks, to reduce the number of physical devices on a network and improve device usage efficiency. Huawei VRP V8's Virtual System (VS) feature can virtualizes a single physical device into multiple virtual systems. Each VS can be configured, managed, and maintained as an independent device. The VS's on a physical device are isolated from one another and can process services independently. In a data center, VS's on a physical device can carry different services or serve different user groups to improve network reliability and security. VS technology also improves the efficiency of network devices and reduces network construction costs. Because user groups are isolated and managed separately, user management becomes easier.

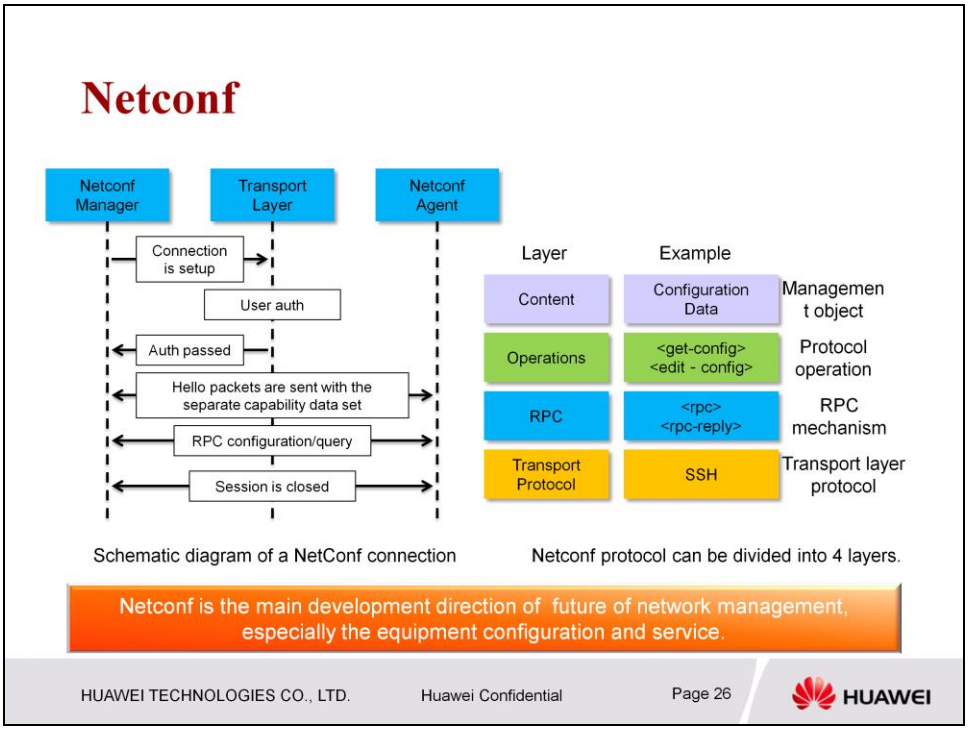


- In the two-phase configuration mode, the system configuration process is divided into two phases:
  - 1. The user enters a configuration command, and then the system performs syntax and semantics checks in the candidate database. If an incorrect clause is found, the system displays a message on the command line terminal, indicating the fault and cause.
  - 2. The user commits the configuration, and thus the system enters the configuration commit phase. The system delivers the configuration in the candidate database to the corresponding service module.
    - If the service takes effect, the system adds the configuration to the running database.
    - If the service fails, the system will prompt the user that the configuration is incorrect. After modifying the configuration, the user can recommit it.
- VRP8 configuration management uses a model-driven scheme on the basis of embedded databases. System configuration models are established by analyzing services supported by the delivered to the APP component after being committed. The configuration information model effectively addresses the impact of a configuration tool on the APP

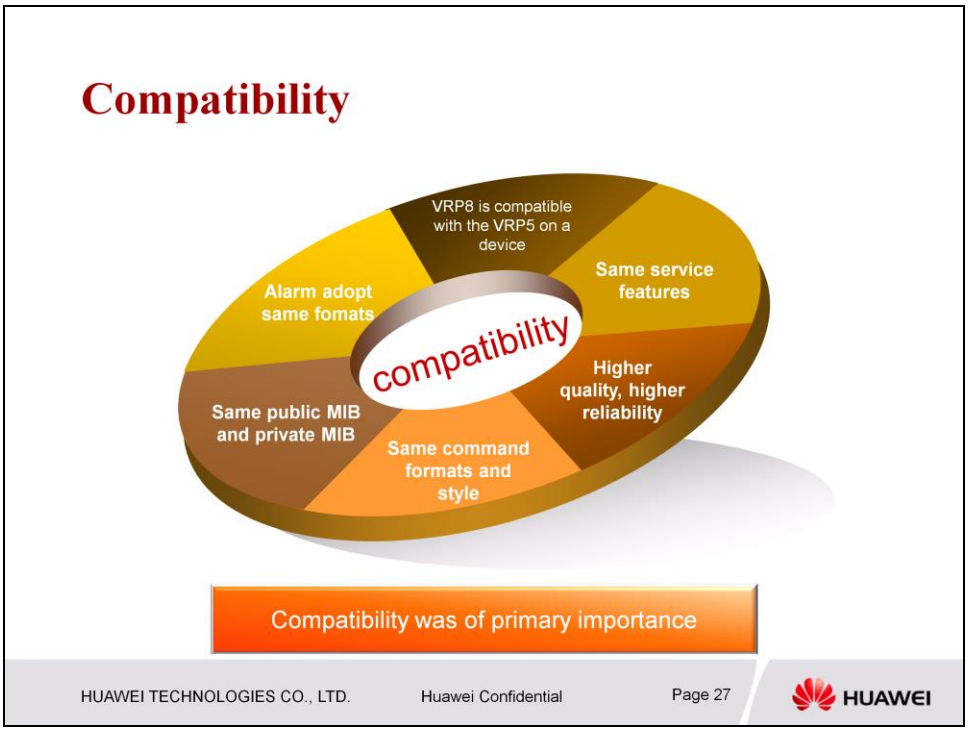
component so that the configuration tool is invisible to the APP component. In this manner, configuration tools, such as CLI, SNMP, Netconf, and WebUI can be added as needed.

- Phase configuration is implemented. With development of telecommunications services, requirements for configuration reliability of carrier-class devices become higher. One basic requirement is that devices support two-phase configuration.

- Concepts related to two-phase configuration are as follows:
    - Running database
        - A running database refers to the configuration set that is running in the system.
    - Candidate database
    - A candidate database refers to the mirrored running database created for a specific user in
        - the system during two-phase configuration. The user edits a configuration in the candidate database. After the configuration is complete, the user commits the configuration to the running database.
    - Rollback
        - Rollback refers to the process that the system cancels a configuration if it fails to be committed.
- In the first phase, the configuration is edited in the candidate database.
- In the candidate database, the system performs syntax and semantics checks for each command.
- After the configuration is committed, the system enters the second phase:
- The system commits the configuration in the candidate database to the running database, and generates a check point to record the committed configuration.
- Two-phase configuration commit enhances security and reliability of configurations and minimizes the impact of configurations on services.

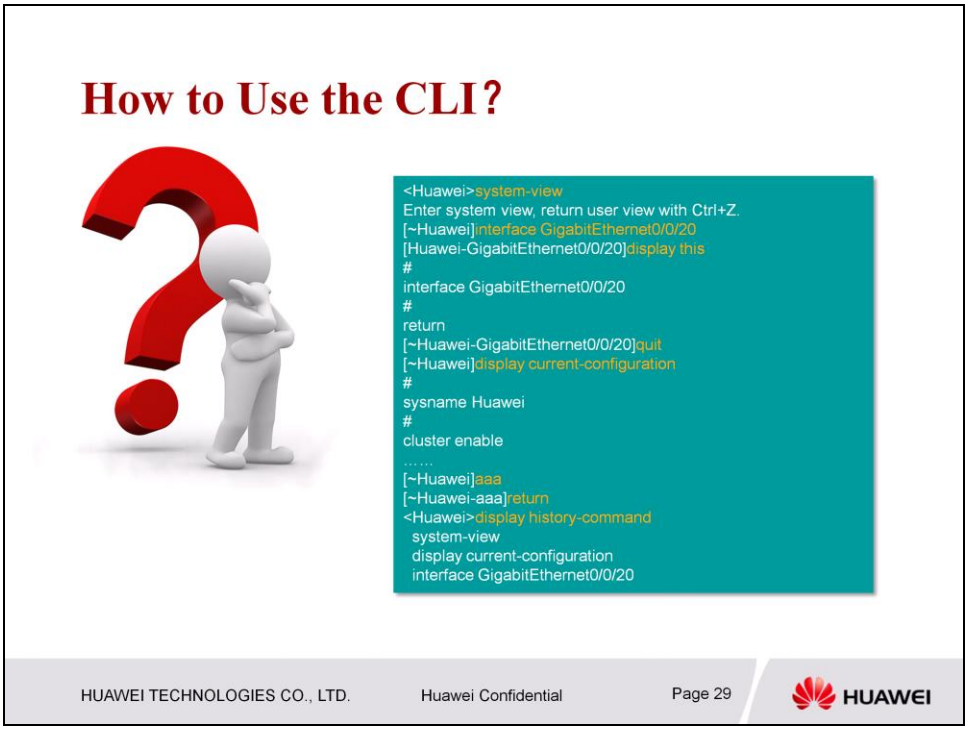


- NetConf is a mainstream trend for network management, especially in device configuration and service provision. The NetConf tool uses Extensible Markup Language (XML) to encode configuration data and protocol messages, and is operated and controlled by Remote Procedure Call (RPC). NetConf is more flexible and extensible than the CLI tool and SNMP tool. NetConf packets are exchanged by connection-oriented transport layer protocols, which provide NetConf packets with security and reliability and keep integrity. The NetConf protocol can be run over various eligible transport layer protocols.
- The NetConf protocol has the following characteristics:
  - ❑ It is connection-oriented and performs operations and control in RPC mode.
  - ❑ XML is used to encode configuration data and protocol messages.
  - ❑ It is extensible and allows a vendor to customize operations to implement specified management. This may result in different protocol-specific implementations on non-Huawei devices.
  - ❑ Configuration data is separated from status data.
  - ❑ It can be extended by defining a new function.



- During the design of the VRP8, compatibility was of primary importance.
  - ❑ To the system, an upgrade from VRP8 to VRP5 is seamless as both adopt the same syntax and configuration style. For example, during an upgrade, a VRP8 system can directly read configuration files associated with the VRP5, This means that the configuration files associated with the VRP5 do not need to be modified during the upgrade.
  - ❑ The maintenance commands in the VRP5 are still available in the VRP8. Therefore, users that are familiar with VRP5 commands do not require additional training.
- Public and private MIB files in the VRP8, in addition to commands, are compatible with those in the VRP5. This means that an NMS using CLIs and MIB interfaces can manage devices that are upgraded from the VRP5 to the VRP8 without being upgraded.
- Alarms in the VRP8 adopt the same formats as those in the VRP5. This allows VRP8 alarms to be processed in the same manner as VRP5 alarms.
- The VRP8 is compatible with the VRP5 on a device so that carriers can upgrade the VRP5 to the VRP8 to deploy new services without replacing existing devices.

Contents



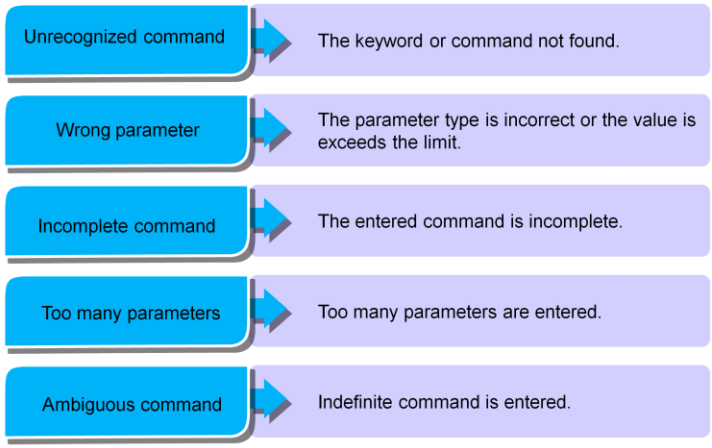
- Run the **system-view** command under the user view, can enter the system view.
- Run the **aaa** command under the system view, can enter the AAA view.
- Run the **diagnose** command can enter the diagnose view.
- run the **quit** command to return from the system view to the user view. If you run **quit** command under user view, you will logout the device.
- Run the **return** command, you can directly back to the user view from any current system view or sub view.
- Run the **display history-command** command can check the history commands in previous steps.
- Run the **display current-configuration** to check the running configuration information.
- Run the **display this** command the check the configuration information about the current view.
- Run the **display diagnostic-information [file-name]** to query the diagnose information of the system.

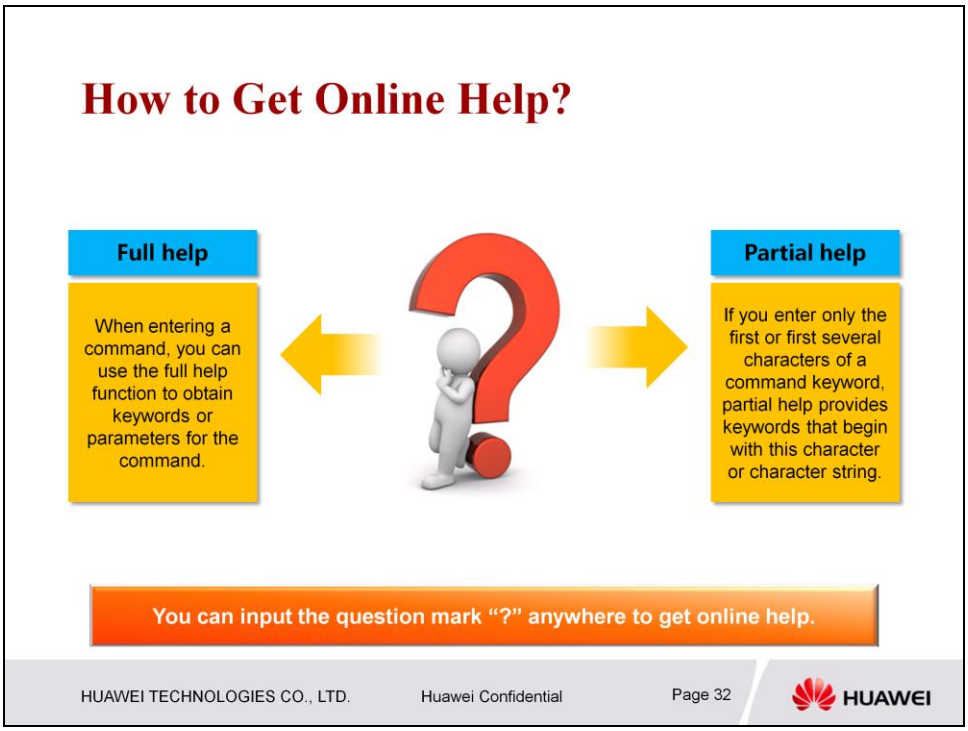
Common Function Keys

Common key	Inserts a character at the current location of the cursor if the editing buffer is not full, and the cursor moves to the right.
BackSpace	Deletes the character on the left of the cursor and the cursor moves to the left.
↑ or <Ctrl_P>	Access the previous history commands.
↓ or <Ctrl_N>	Access the next history command.
Tab	Enter an incomplete keyword and press Tab to complete the keyword.
<Ctrl_C>	Stops performing current functions.
Space	Continue to display the next screen information.
Enter	Continue to display the next line information.

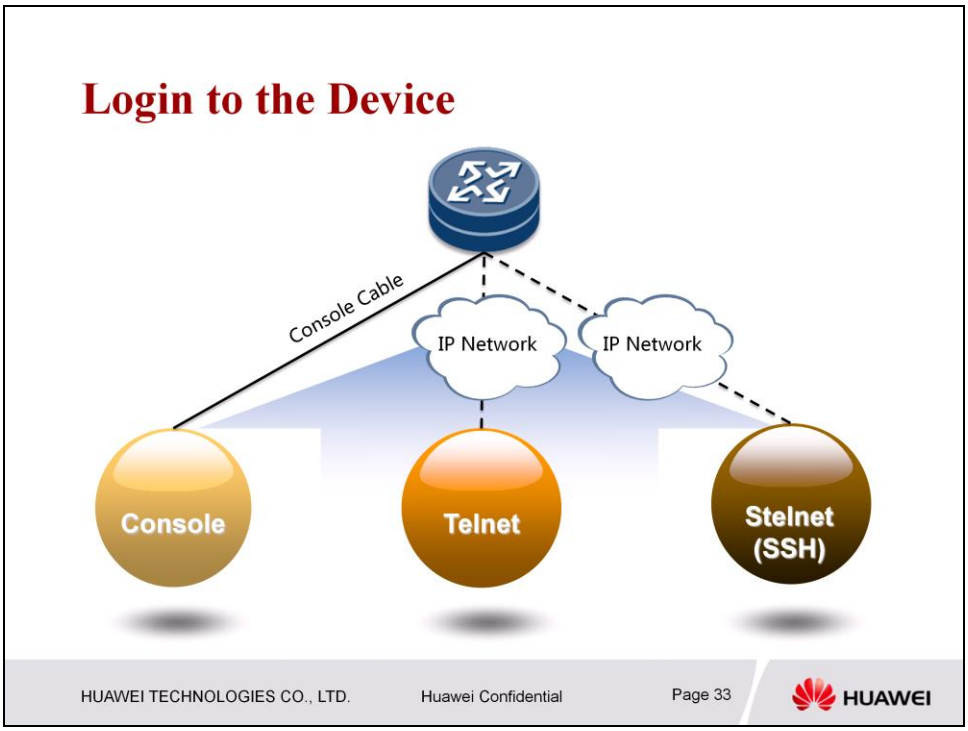
- Enter an incomplete keyword and press Tab to complete the keyword.
  - ❑ When a unique keyword matches the input, the system replaces the incomplete input with the unique keyword and displays it in a new line with the cursor leaving a space behind.
  - ❑ When the input has multiple matches, press Tab repeatedly to display the keywords beginning with the incomplete input in a circle until the desired keyword is displayed. In this case, the cursor closely follows the end of the keyword.
  - ❑ When an incorrect keyword is entered, press Tab and it is displayed in a new line without being changed.

Command Error Notice

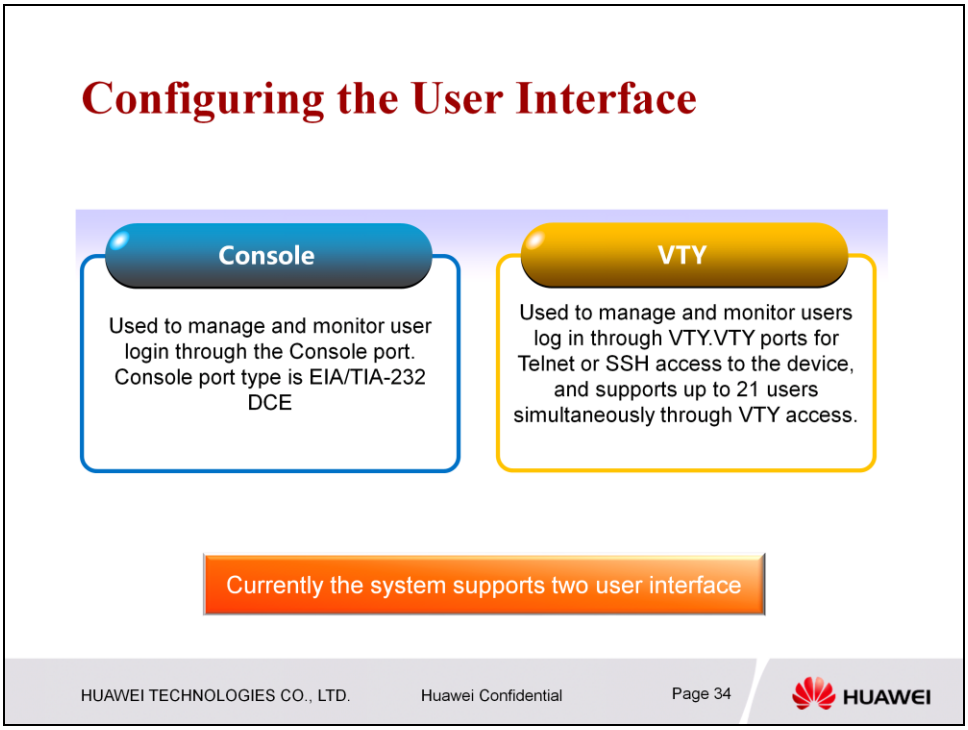




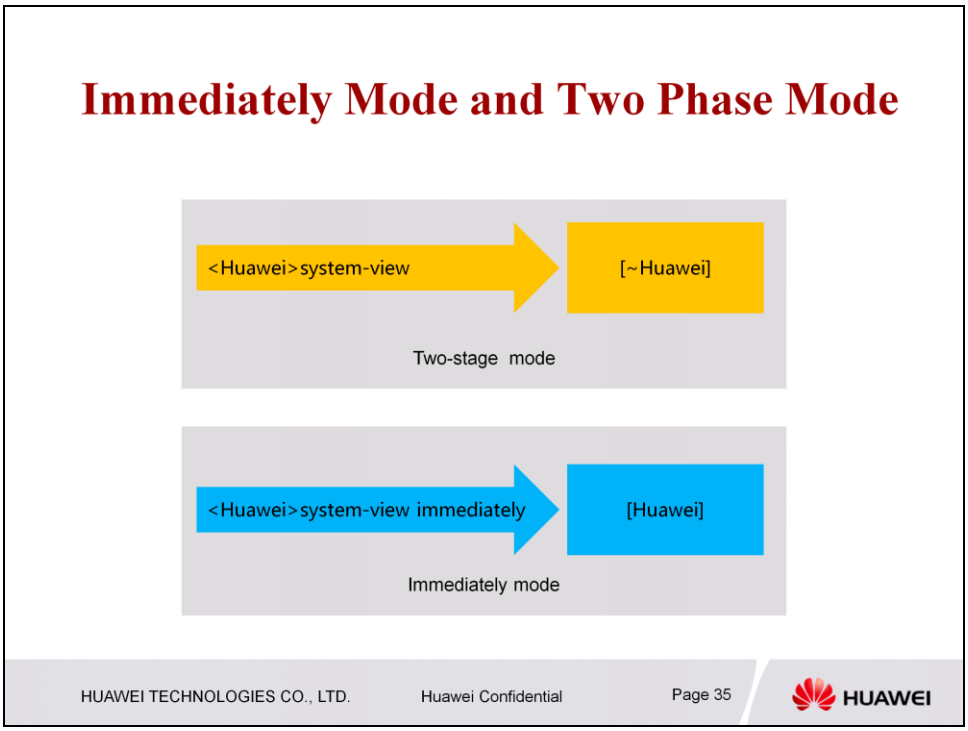
- Use any of the following methods to obtain full help from a command line.
  - ❑ Enter a question mark (?) in any command view to obtain all the commands and their simple descriptions.
  - ❑ Enter a keyword and a question mark (?) separated by a space. All keywords associated with this command, as well as simple descriptions, are displayed.
  - ❑ Enter a keyword and a question mark (?) separated by a space. All parameters associated with this keyword, as well as simple descriptions, are listed.
- Use any of the following methods to obtain partial help from a command line.
  - ❑ Enter a character string followed directly by a question mark (?) to display all keywords that begin with this character string.
  - ❑ Enter a command and a string followed directly by a question mark (?) to display all the keywords that begin with this string.
  - ❑ Enter the first several letters of a keyword in a command and press **Tab** to display a complete keyword. The first several letters, however, must uniquely identify the keyword. If they do not identify a specific keyword, press **Tab** continuously to display different keywords and you can select one as required.



- Console
  - ❑ User use a PC connect to the console port of the device, when the first time power on and configuration.
  - ❑ By default, user can directly login to the device by connect to the console port.
- Telnet
  - ❑ Through a terminal connected to the network using Telnet Login equipment for local or remote configuration, the target device to authenticate a user based on the configured parameters.
  - ❑ Commonly used for local or remote maintenance equipment, especially remote maintenance mode, brings great convenience for users, but security is not high.
  - ❑ By default, users can not through Telnet log in directly to the device.
- Stelnet
  - ❑ Through a terminal connected to the network, SSH (Secure Shell) security information security and powerful authentication function to protect equipment against IP spoofing and plain text password interception attacks.
  - ❑ By default, users can not through STelnet directly log in to the device.



- Configuring Console user interface
  - ❑ Run the system-view command to enter the system view.
  - ❑ Run the user-interface console ui-number to enter the console user interface.
- Configuring VTY (Virtual Type Terminal) user interface.
  - ❑ Run the system-view command to enter the system view.
  - ❑ Run the user-interface vty first-ui-number [last-ui-number] command to enter VTY user interface.
- A lots of contents can be configured by Console and VTY user interface, you can look at the product manual for details.



- The immediate mode is the traditional mode.
  - ❑ After you enter a command line and press **Enter**, the system performs the syntax check. The configuration takes effect as soon as it passes the syntax check.
- In the two-stage mode, the system configuration process is divided into two stages.
  - ❑ In the first stage, a user enters command lines and the system performs syntax and semantics checks in the candidate database. If syntax or semantics errors are found in the command lines, the system displays a message on the terminal to inform the user of the error and the cause.
  - ❑ In the second stage, a user commits the configuration, and the system enters the configuration commitment stage. The system delivers configurations in the candidate database to a service. If the configurations take effect, the system adds them to the running database of the current system. During the configuration commitment stage, the system checks the configuration validity and displays messages when configurations in the candidate databases are identical with those in the running databases.
- Run **system-view** to enable the two-stage mode;

- ❑ In the two-stage mode, if the user has modified configurations but has not submit the modification, the system prompt **~** is changed to *****, prompting the user that the configurations are not submitted. After the user runs the **commit** command to submit the configurations, the system prompt ***** is restored to **~**.
- ❑ The following is an example:

  <HUAWEI> **system-view**

  Enter system view, return user view with return command.

  [~HUAWEI] **sysname HUAWEIA**

  [*****HUAWEI] **commit**

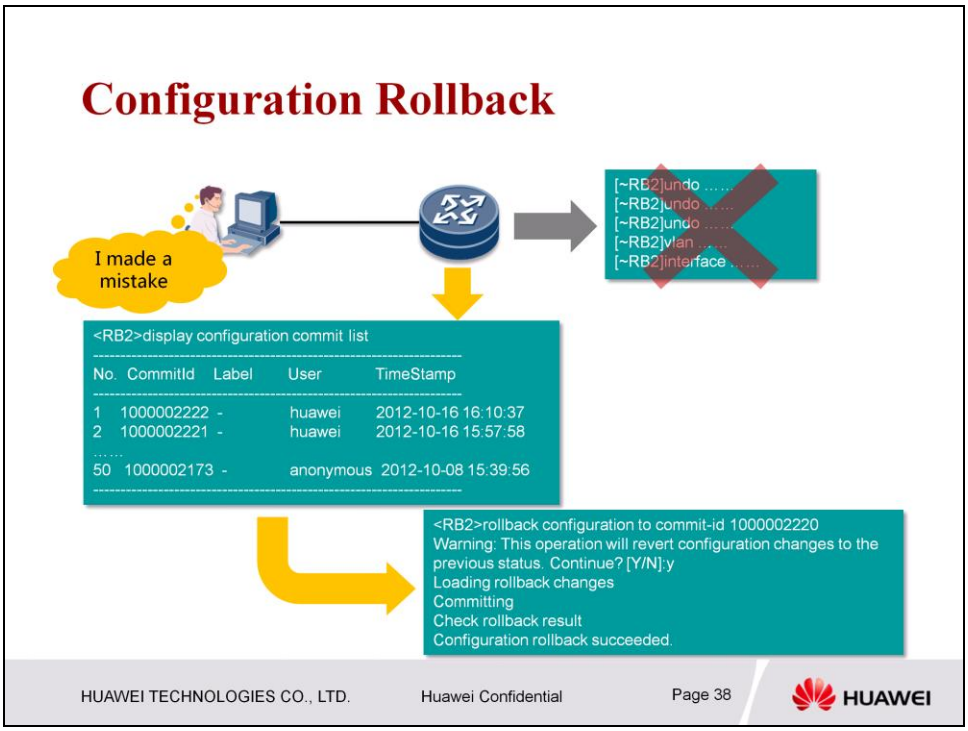
  [~HUAWEIA]
- ❑ NOTE: The symbol ***** has two meanings: When ***** is displayed in an interactive operation, it indicates the configurations that have not been submitted. After the **display current-configuration inactive** or **display current-configuration all** command is executed, ***** in the command output indicates offline configuration.

- (Optional) Run **display configuration candidate** [**merge**] to display all uncommitted configurations.
  - ❑ If you specify the **merge** keyword, the command displays uncommitted configurations and committed configurations. If you do not specify the **merge** keyword, the command displays uncommitted configurations.
  - ❑ You can edit the uncommitted configurations.
- (Optional) Run **clear configuration candidate** to delete all the uncommitted configurations.
  - ❑ If you do not need to execute the uncommitted configurations in the current configuration process, you can delete them.
- (Optional) Run **commit** **trial** [*time*] enable trial running of system configuration.
  - ❑ This configuration enables the trial running of new functions and services without interrupting the services running on the live network, which improves network reliability.
  - ❑ To disable the trial running of the configuration before the trial running times out, run the **abort trial** command to roll the system configuration back to the configuration state before the trial running. To view the trial running status of a system configuration, run the **display configuration trial status** command.

- Run **commit** to commit configurations.

Comparing Immediately Mode and Two Phase Mode

Immediately mode	Two phase mode
Advantages ❑The system can immediately detect the configuration impact on services.	**Advantages** ❑Allows several service configurations to take effect as a whole. ❑Allows users to preview configurations in the candidate database. ❑Clears a configuration that does not take effect if an error occurs or the configuration does not meet the expectation. ❑Minimizes the impact of configuration process on current services.
Disadvantages ❑Configuration errors impact services immediately because configurations take effect immediately. Besides, you cannot delete configurations from the services as a whole but have to delete commands one by one.	**Disadvantages** ❑Configurations take effect only after the commit command is run.



- If faults are caused by incorrect configurations or an unexpected configuration result occurs on the network after the configuration is submitted, you can roll back the configuration.
- Before you roll back the configuration, the device has used either of two modes to submit the configuration and generated a configuration rollback point.
- The modes for the configuration to take effect are as follows:
  - ❑ Immediate mode: After you enter the command and press **Enter**, the system checks whether the current configuration is the same as the historical configuration. If they are different, the system submits the configuration and generates the configuration rollback point. Therefore, multiple configuration rollback points may be generated when one feature is deployed.
  - ❑ Two-stage mode: When running a series of configuration commands, you must use the **commit** [**label** *label* | **description** *description*] command to enable the system to compare the current configuration with the historical configuration and generate a configuration rollback point. In this way, all configurations for a certain service using a series of commands can take effect. To quickly find the required configuration rollback point, specify **description** *description* to

configure the point

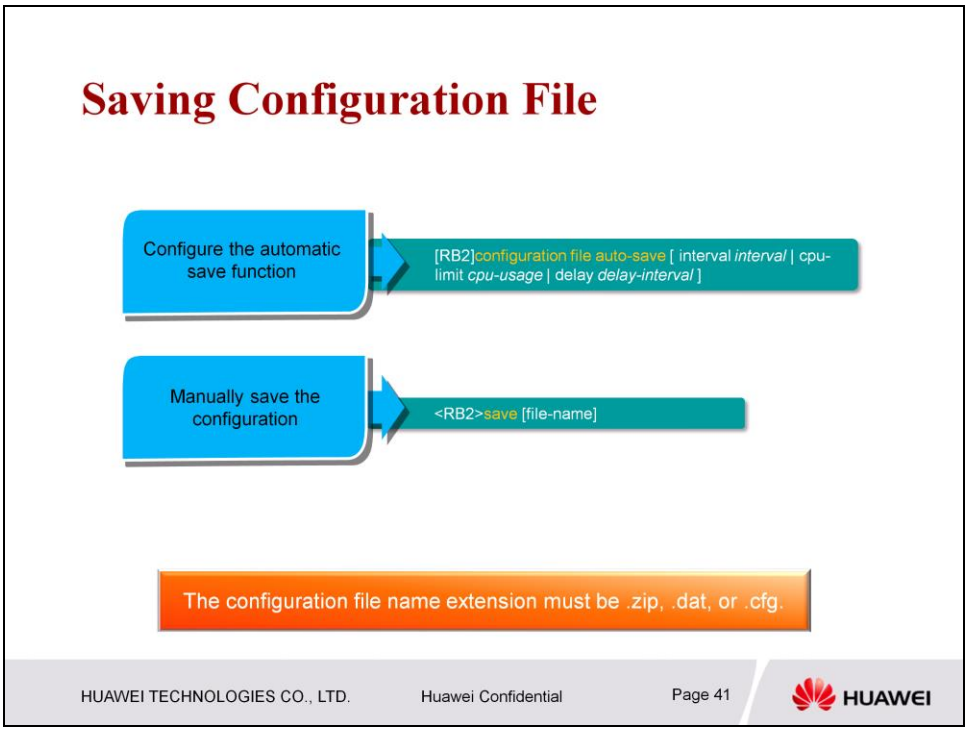
description. You are advised to use this mode to edit and submit the configuration.

- Check the configuration rollback points and the latest configuration changes.
  - ❑ Run **display configuration commit list** [**verbose**] [*number-of-commits*] All configuration rollback points and their details are displayed.
  - ❑ To check one or more configuration rollback points, specify *number-of-commits*.
  - ❑ Run **display configuration commit changes** [**at** *commit-id* | **since** *commit-id* | **last** *number-of-commits*] The configuration change for a specified configuration rollback point is displayed.
  - ❑ Based on the configuration change, you can determine whether to roll back the configuration and the possible effects on the system.
    - ■ To check the configuration changes at all configuration rollback points, do not specify any parameters in the command.
    - ■ To check the configuration change at a specified configuration rollback point, specify **at** *commit-id*.
    - ■ To check the configuration change from a specified configuration rollback point to the current state, specify **since** *commit-id*.
    - ■ To check one or multiple of the latest configuration changes, specify **last** *number-of-commits*.
- Roll back the configuration based on the required historical configuration state.
  - ❑ Run **return** The user view is displayed so that no data is submitted before rollback.
  - ❑ Run **rollback configuration** { **to commit-id** *commit-id* | **last** *number-of-commits* } The system is rolled back to the historical configuration state based on the specified configuration rollback point or the number of configuration changes.
  - ❑ Based on the configured rollback rule, run this command to delete the created configuration, re-create the deleted configuration, or recover the quondam configuration.

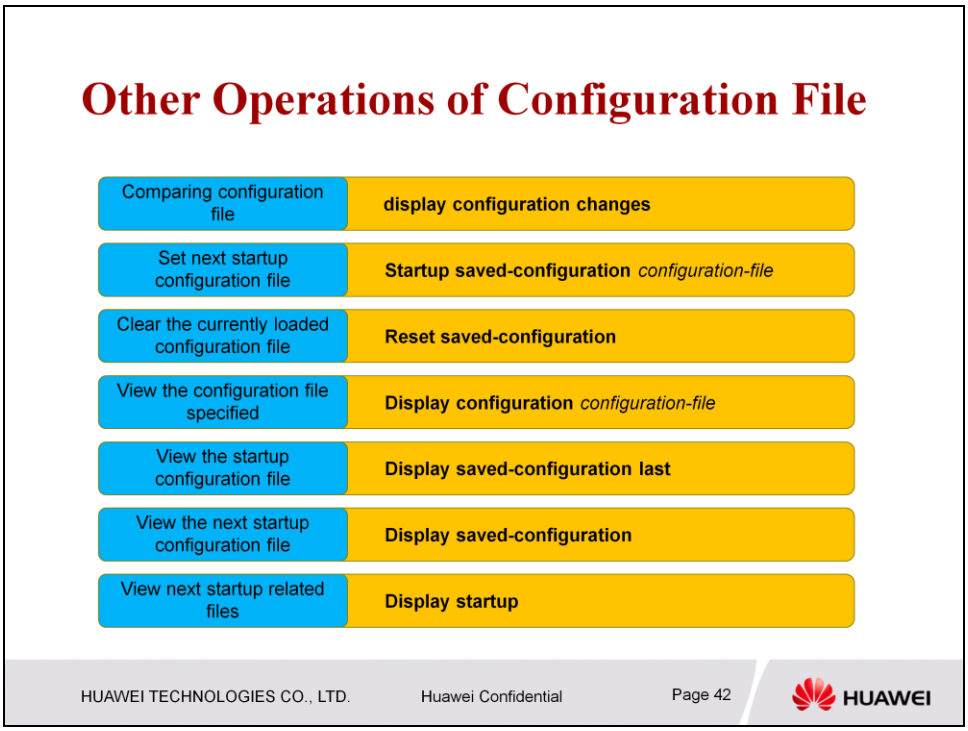
- To roll back the system to the historical configuration state at a certain configuration rollback point, specify **to commit-id** *commit-id*.
- To roll back the system to the historical configuration state before one or more latest configuration rollback points, specify **last** *number-of-commits*.

❑ NOTE: After the configuration rollback operation is complete, you can run the **display configuration rollback result** command to view the result of the latest configuration rollback operation.

- (Optional) Run **clear configuration commit** *commit-id***label** The earliest configuration rollback points are deleted.
- **Checking the Configuration**
- Run the **display configuration commit list** [**verbose**] [*number-of-commits*],command to view the configuration rollback point list.
- Run the **display configuration commit changes** [**at** *commit-id* | **since** *commit-id* | **last** *number-of-commits*],command to view information about configuration changes.
- Run the **display configuration rollback result**,command to view information about the latest configuration rollback operation, including prompt messages and failure messages.

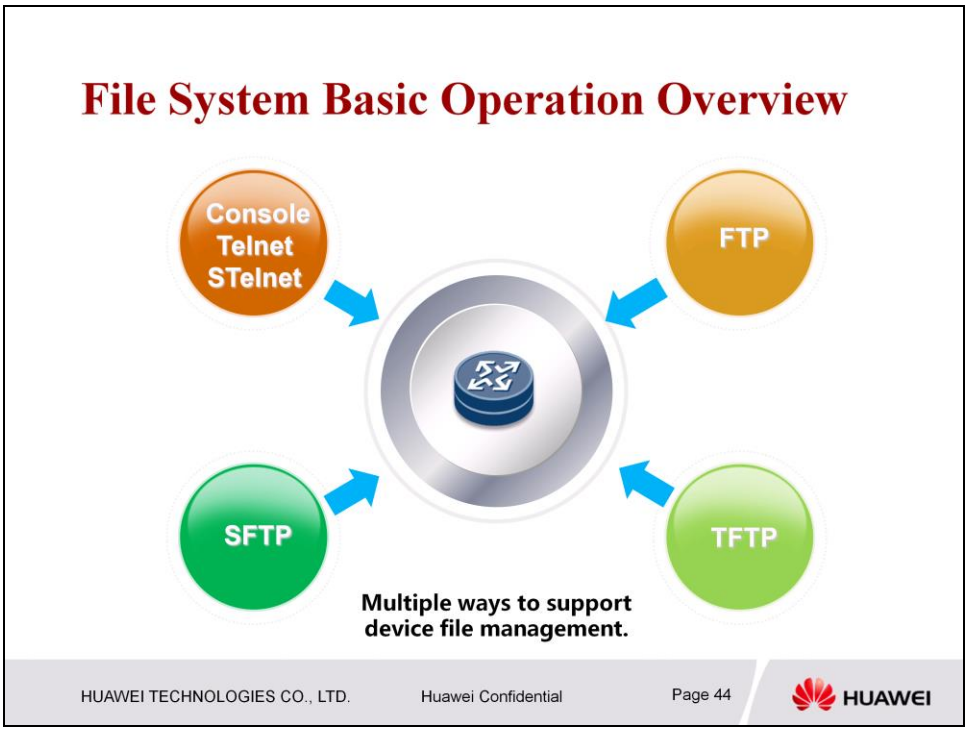


- Save the configurations automatically.
  - ❑ Run **system-view** The system view is displayed.
  - ❑ Run **configuration file auto-save** [**interval** *interval* | **cpu-limit** *cpu-usage* | **delay** *delay-interval*] * The system is configured to periodically save the configurations.
  - ❑ By default, system does not periodically save configurations.
  - ❑ The system cancels the automatic save operation when: Content is being written into the configuration file.
  - ❑ The interface board configurations are being recovered.
  - ❑ The CPU usage is excessively high.
- Save the configurations manually.
  - ❑ Run **save** [*configuration-file*] The current configuration is saved.
  - ❑ The configuration file name extension must be .zip, .dat, or .cfg. The system startup file must be stored in the root directory of the storage device.
  - ❑ NOTE: If you do not specify *configuration-file* when saving the configuration file for the first time, the system asks you whether to save the configuration file as **vrpcfg.zip**. The **vrpcfg.zip** file is the default system configuration file with empty configurations in initial state.



- **Comparing Configuration Files**
- Run **display configuration changes** [**running file** *configuration-file* | **file** *configuration-file* **running**] The system starts to check whether the current configurations are identical with the specified configuration file.
- The system compares the configuration files starting from the first line.
- **Set next startup configuration files**
- Run **startup system-software** *system-file* The system software to be loaded for next startup is specified.
- If the device has dual MPUs, run the **startup system-software** *system-file* **slave-board** command to specify the system software for the slave MPU to load during the next startup.
- NOTE: Specify the same system software for the master and slave MPUs.
- Run **startup saved-configuration** *configuration-file* The configuration file for next startup is specified.
- The device reads the configuration file from the root directory of the storage device for initialization when powered on.
- (Optional) Run **startup patch** *file-name* **all** The patch file for next startup is specified.

- **Clearing the Configuration File**
- You need to delete the configuration file when:
    - ❑ The software and configuration file do not match after the device software is upgraded.
    - ❑ The configuration file is damaged or an incorrect configuration file is loaded.
- CAUTION: Exercise caution when you run the **reset saved-configuration** command. You are advised to run this command under the guide of Huawei technical support personnel.
- **Procedure**
    - ❑ Run the **reset saved-configuration** command to clear the current configuration.
    - ❑ NOTE: Before clearing the configuration file, the system checks whether the current startup configuration file is consistent with the configuration file for the next startup.
    - ❑ If yes, run this command to clear the two configuration files. You can specify the next startup configuration file; otherwise, the file is empty.
    - ❑ If no, run this command to clear the current startup configuration file.
    - ❑ If the current startup configuration file is empty, the system displays a message indicating that the configuration file does not exist after you run this command.
    - ❑ If you do not run the **startup saved-configuration** command to specify a configuration file that contains correct configurations or the **save** command to save the configuration file, default parameters are used during system initialization when the device restarts next time.



- The user can manage the device directories, files by the following ways.
  - Console
  - Telnet
  - sTelnet
  - FTP
  - TFTP
  - SFTP

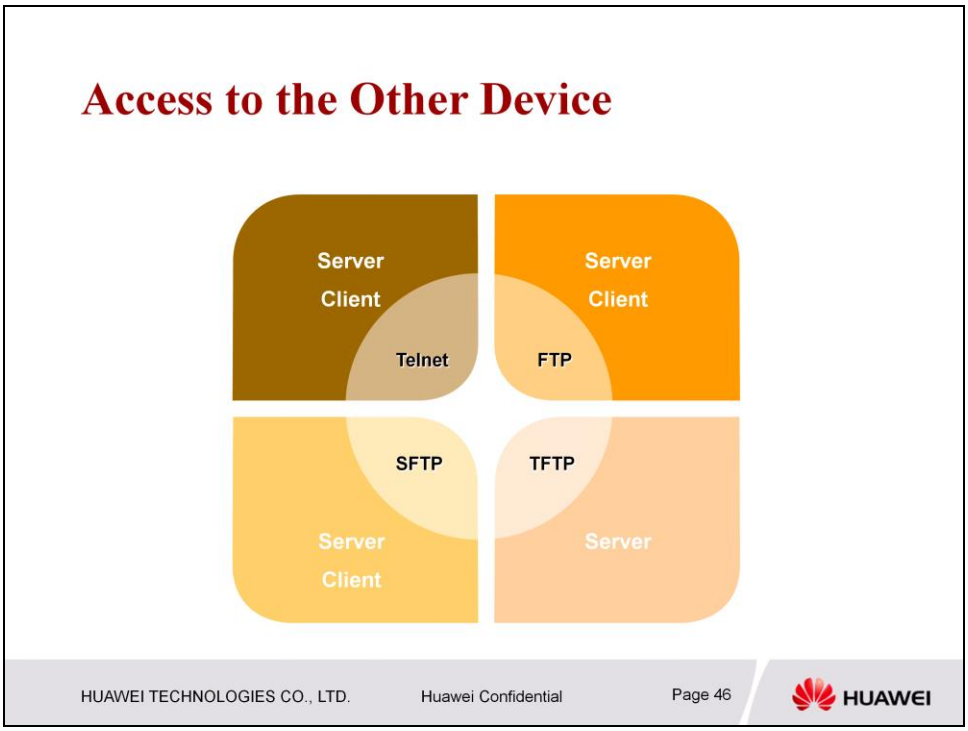
Basic Commands for the File System Operation

Directory management

```
----change the current directory----------
cd directory
----view the current directory------------
pwd
-----view the list of current directory---------
dir [/all][file-name]
-----create a directory-----------------------
mkdir directory
-----rename the directory---------------------
rename source-filename destination-filename
-----delete the directory---------------------
rmdir directory
```

File management

```
----view the contents of the file----------
more filename
-----copy the file-------------------------------
copy source-filename destination-filename
-----move the file-------------------------------
move source-filename destination-filename
-----rename the file----------------------------
rename source-filename destination-filename
------compress the file-----------------------
zip source-filename destination-filename
------delete the file-------------------------
delete [ /unreserved ] filename
-------recover the deleted file----------------
undelete filename
------clear the recycle-bin--------------------
reset recycle-bin [ /f | filename ]
------run a bat file--------------------------
execute filename
------configure system file prompts----------
file prompt { alert | quiet }
```



- Telnet:
  - ❑ If you logon to the remote device from a PC with a telnet terminal software, the remote device be a telnet server.
  - ❑ If you logon to the other device from the current device, the current devices be a client.
- FTP
  - ❑ If you run ftp command to logon to a remote device from a PC with a terminal software, the remote device be a ftp server.
  - ❑ If you run ftp command on current device to logon to a remote device, the current devices be a ftp client.
  - ❑ FTP support two file transport mode.
    - ■ Binary mode, this mode is used to transmit the program file, such as .app, .bin, .btm files.
    - ■ ASCII mode, this mode is used to transmit the text format file, such as .txt, .bat, .cfg files.
- TFTP
  - ❑ For security reasons, Huawei device can only support be a TFTP client, and can only support binary mode to transmit file.
- SFTP
  - ❑ SFTP protocol provide more security mechanisms than FTP

protocol, Huawei devices can support SFTP server or client.

Summary

- Support multi-cores and multi-process.
- Support virtual system.
- Support NSx（NSS、NSR、NSB、NSF） features and ISSU .
- Support Netconf protocol.
- Support two phase configuration and configuration rollback.
- Support auto-save configuration file.
- Compatible running on the VRP5 devices and same command style with VRP5.
- Support to logon to the device by Telnet, SSH, Console, FTP, TFTP, SFTP protocols.

Thank you

www.huawei.com