



Junosphere

Network Topology Guide

Release

3.0



Modified: 2015-07-16

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Junosphere Network Topology Guide
Copyright © 2015, Juniper Networks, Inc.
All rights reserved.

Revision History
2015-07-15—R3

The information in this document is current as of the date on the title page.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

By downloading, installing or using such software, you agree to the posted [Terms of Use for Junosphere](#).

Table of Contents

	About the Documentation	ix
	Documentation and Release Notes	ix
	Documentation Conventions	ix
	Documentation Feedback	x
	Requesting Technical Support	xi
	Self-Help Online Tools and Resources	xi
Part 1	Building and Using Junosphere Network Topologies	
Chapter 1	Network Topologies Overview	3
	Overview	3
Chapter 2	Creating and Editing Topologies	5
	Topology Wizard Overview	5
	Creating Topologies with the Topology Wizard	7
	Starting a Topology from the Topology Wizard	10
	Editing a Topology	13
	Editing Topology Metadata	13
	Editing a Topology Using the Topology Wizard	13
	Copying a Topology	13
Chapter 3	Using Topologies in Public Libraries	15
	Topologies in Public Libraries Overview	15
	Using Topologies in Public Libraries	18
Chapter 4	Building Topologies with VMM Scripting	21
	The topology.vmm File	21
	.vmm File Syntax	23
	Editing a Topology with .VMM Scripting	24
	Using Macros	25
	CentOS Server	25
	Junos Space	25
	Mu Studio Performance	26
	Mu Test Engine	27
	VJX	28
	VMX	28
	VPTX	29
	VSRX	30
	WANDL	31
	Advanced Parsing Capabilities	32
	Parsing Concepts	32

	Parser Syntax Rules	33
	Viewing VM Images	33
Chapter 5	Definition of Options in the VMM File	35
	Using the ipaddr Interface Option	35
	VJX Virtual Machines Interface Options	36
	Topology Configuration Example	37
	Sample .vmm Configuration File	37
	Creating the Virtual Machine Configuration	39
	Configuring Junos OS Network Devices	39
	Rules for Junos OS Configuration Files	39
	Configuring Other Virtual Machines	40
	Configuring CentOS Servers	40
	Default Configuration File	41
	Loading a Junos OS Configuration File	43
	Building and Testing Your Network Topology	43
	Building a Topology File Set	44
	Connecting to the Network Topology	44
Part 2	Index	
	Index	49

List of Figures

Part 1	Building and Using Junosphere Network Topologies	
Chapter 2	Creating and Editing Topologies	5
	Figure 1: Topology Wizard Interface	5
	Figure 2: Create Topology Dialog Box	7
	Figure 3: Image Information Bubble	8
	Figure 4: Edit Configuration Dialog Box	10
	Figure 5: Topology Saved Dialog Box—Sandbox	11
	Figure 6: Topology Saved Dialog Box—Bank	12
	Figure 7: Copy Topology Dialog Box	14
Chapter 4	Building Topologies with VMM Scripting	21
	Figure 8: Topology file set Structure	22
	Figure 9: Example topology.vmm File	23
	Figure 10: View VM Images Window	34
Chapter 5	Definition of Options in the VMM File	35
	Figure 11: Simple topology.vmm Configuration	37
	Figure 12: Network Connect Dialog Box	41

List of Tables

	About the Documentation	ix
	Table 1: Notice Icons	ix
	Table 2: Text and Syntax Conventions	ix
Part 1	Building and Using Junosphere Network Topologies	
Chapter 2	Creating and Editing Topologies	5
	Table 3: Toolbox Icons	6
Chapter 3	Using Topologies in Public Libraries	15
	Table 4: Starter Topologies	15
	Table 5: Ecosystem Partner Topologies	16
	Table 6: Advanced Topologies	16
Chapter 4	Building Topologies with VMM Scripting	21
	Table 7: .vmm File Syntax	23

About the Documentation

- [Documentation and Release Notes on page ix](#)
- [Documentation Conventions on page ix](#)
- [Documentation Feedback on page x](#)
- [Requesting Technical Support on page xi](#)

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.


If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

Documentation Conventions

[Table 1 on page ix](#) defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.

[Table 2 on page ix](#) defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the configure command: user@host> configure

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
Fixed-width text like this	Represents output that appears on the terminal screen.	<code>user@host> show chassis alarms</code> <code>No alarms currently active</code>
<i>Italic text like this</i>	<ul style="list-style-type: none"> Introduces important new terms. Identifies book names. Identifies RFC and Internet draft titles. 	<ul style="list-style-type: none"> A policy <i>term</i> is a named structure that defines match conditions and actions. <i>Junos OS System Basics Configuration Guide</i> RFC 1997, <i>BGP Communities Attribute</i>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; interface names; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"> To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level. The console port is labeled CONSOLE.
< > (angle brackets)	Enclose optional keywords or variables.	<code>stub <default-metric metric>;</code>
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	<code>broadcast multicast</code> <code>(string1 string2 string3)</code>
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	<code>rsvp { # Required for dynamic MPLS only</code>
[] (square brackets)	Enclose a variable for which you can substitute one or more values.	<code>community name members [</code> <code>community-ids]</code>
Indentation and braces ({ })	Identify a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop address; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

Requesting Technical Support

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>
- Download the latest versions of software and review release notes: <http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <https://www.juniper.net/alerts/>
- Join and participate in the Juniper Networks Community Forum: <http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

PART 1

Building and Using Junosphere Network Topologies

- [Network Topologies Overview on page 3](#)
- [Creating and Editing Topologies on page 5](#)
- [Using Topologies in Public Libraries on page 15](#)
- [Building Topologies with VMM Scripting on page 21](#)
- [Definition of Options in the VMM File on page 35](#)

CHAPTER 1

Network Topologies Overview

- [Overview on page 3](#)

Overview

This document describes the process for creating and using the network topologies that are required for building virtual networks on Junosphere.

There are three ways to build or use topologies:

- Using the Topology Wizard to create a new topology
- Using existing topologies supplied in the public libraries in Junosphere
- Using Virtual Machine Manager (VMM) scripting to create a new topology or modify an existing topology

Each method produces a file set that includes a topology.vmm file that defines the virtual devices, such as routers, virtual distributed Ethernet (VDEs), and related connections between the devices within a single topology. The file set also contains a configuration file for each Junos OS virtual network element that is defined in the topology.vmm file.

CHAPTER 2

Creating and Editing Topologies

- [Topology Wizard Overview on page 5](#)
- [Creating Topologies with the Topology Wizard on page 7](#)
- [Starting a Topology from the Topology Wizard on page 10](#)
- [Editing a Topology on page 13](#)
- [Copying a Topology on page 13](#)

Topology Wizard Overview







The Topology Wizard (see [Figure 1 on page 5](#)) enables you to design and create your topologies quickly and easily. The Wizard has drag-and-drop functionality and automatically creates the **topology.vmm** file.

Figure 1: Topology Wizard Interface



[Table 3 on page 6](#) lists and describes the toolbox icons in the Topology Wizard.

Table 3: Toolbox Icons

Icon	Description
Connector Icon 	Enables you to create a connection between elements on the canvas.
Selector Icon 	Enables you to select an element on the canvas.
Fit to Page Icon 	Enables you to fit the topology to the canvas.
Actual Size Icon 	Enables you to view the topology in its actual size.
Zoom In Icon 	Enables you to zoom in on a part of the topology.
Zoom Out Icon 	Enables you to zoom out.

Related Documentation

- [Starting a Topology from the Topology Wizard on page 10](#)
- [Creating Topologies with the Topology Wizard on page 7](#)
- [Editing a Topology on page 13](#)

Creating Topologies with the Topology Wizard

To use the Topology Wizard to create a new topology:

1. From the navigation tree, select **Topologies > Create Topology in Wizard**. The Topology Wizard appears, with the **Create Topology** dialog box displayed.

Figure 2: Create Topology Dialog Box

Create Topology

Name: *

Description:

Downloadable: * ☒

Savable: * ☒

Select a destination library below... *

Sandbox			
	Bank	Sandbox	Name
<input type="checkbox"/>	B1	S1	Library23
<input type="checkbox"/>	B1	S1	testforwrongcustomer
<input type="checkbox"/>	B1	S2	Libr2

Create Cancel

2. Enter the name of the topology and a description, if desired.
3. Indicate whether the topology is to be downloadable, savable, or both.
4. Select the library in which you want to create the topology. Use the filtering fields to narrow your search, if necessary.
5. Click **Create**.

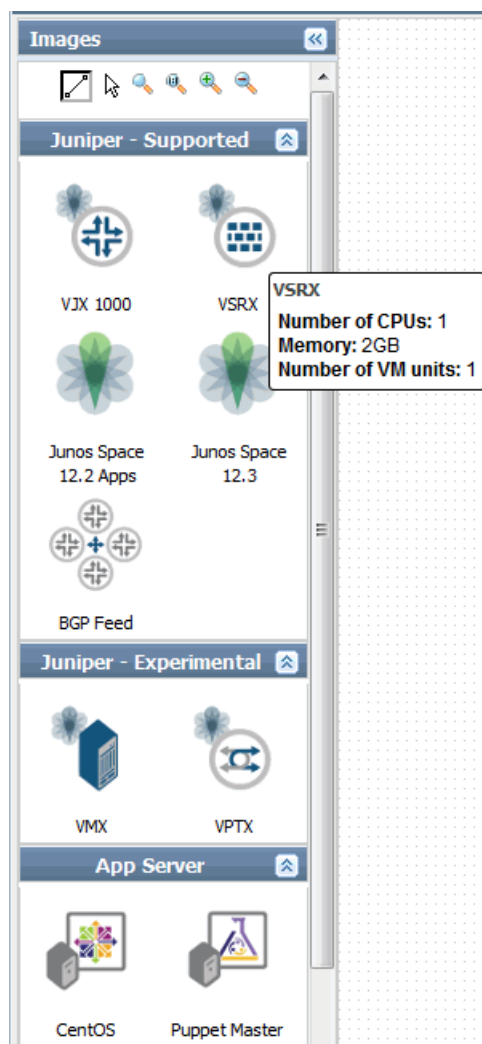
The Topology Wizard window opens in Junosphere with icons and tools on the left side.



NOTE: If this is your first time creating a topology, a **Required Settings** dialog box appears asking you to enable downloads from junosphere.net.

- To see the different types of virtual machine (VM) images, click the double arrow for each type.
- To see additional information about each image, place your mouse pointer on the image until a tooltip, similar to that shown in [Figure 3 on page 8](#), appears.

Figure 3: Image Information Bubble



6. Drag and drop icons for specific types of nodes from the icon gallery onto the canvas, or double-click an icon and then click anywhere on the canvas to place the icon.
7. To create connections, click the **Connector** icon and hover the cursor over the center of one of the objects that you want to connect until a hand icon appears and the selected element is surrounded by a green box.



NOTE: Do not connect the following VMs to other VMs because they are not network elements. These VM types are used only for network management and planning purposes:

- Junos Space
- CentOS
- Mu Studio

8. Enable “T” connections by creating a connection between a network element and the bridge linking the two VMs. The bridge is represented graphically as a circle with a label such as p0, p1, p2, and so on. There is no limit to the number of elements that can be connected to a bridge.
9. Click **Save** to save the topology in the library. You can also click **Save to File** to download a copy of the topology for future use.



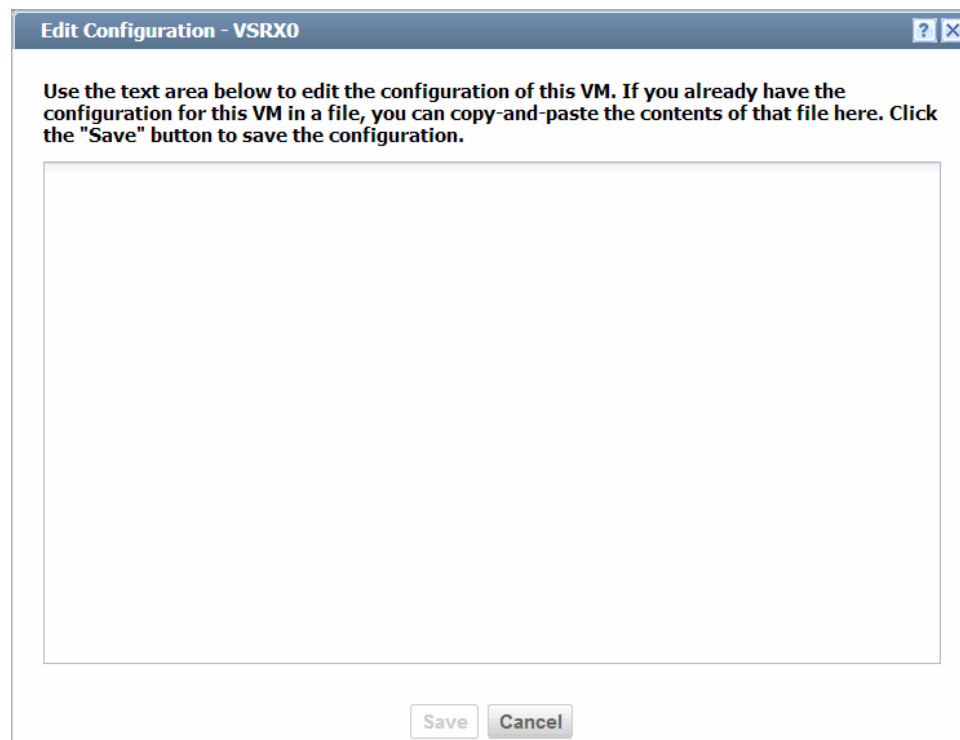
NOTE: In addition to saving the topology to a sandbox or file, the Topology Wizard has an auto-save feature that saves every five changes made to the topology diagram. These changes can include new VMs added, new links added, VMs removed, links removed, VMs moved, and other changes. The topology is automatically saved to the selected library.

Junosphere automatically logs out of the Topology Wizard after 30 minutes of being idle. If you are working in the Topology Wizard, you might lose work if you have not saved your changes. Although Junosphere automatically saves work in the topology wizard after every five changes made while creating the topology, it is best practice to manually save your changes as frequently as possible, especially if you are making many changes with a single action.

If you click **Save**, a dialog box appears, enabling you to launch your topology directly from the Topology Wizard. See [“Starting a Topology from the Topology Wizard” on page 10](#).

10. Create a **.config** file for any configurable network elements in the topology by right-clicking a VM in the topology and selecting **Edit VM Configuration** from the pop-up menu. A dialog box similar to the following appears.

Figure 4: Edit Configuration Dialog Box



11. Either type the configuration information in the dialog box or copy and paste configuration information from another source.
12. Click **Save**.



NOTE: When creating a topology, the Maximum Transmission Unit (MTU) in the VJX interface must be set to 1500. Otherwise, it cannot route BGP packets.

Related Documentation

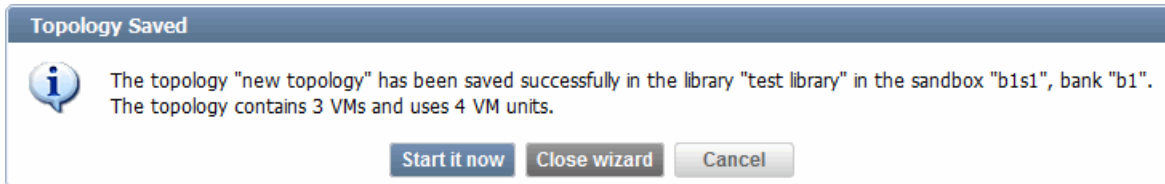
- [Viewing VM Images on page 33](#)
- [Starting a Topology from the Topology Wizard on page 10](#)
- [Editing a Topology Using the Topology Wizard on page 13](#)
- [Topology Wizard Overview on page 5](#)

Starting a Topology from the Topology Wizard

In addition to saving a topology to a sandbox, the Topology Wizard also enables you to launch the topology without having to return to the sandbox, even if there is no reservation for it.

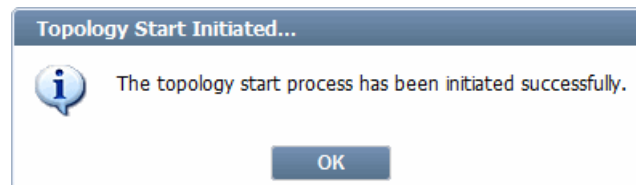
Once you have created a topology and clicked **Save** to save the topology to a sandbox library, the following dialog box appears, enabling you to start the topology.

Figure 5: Topology Saved Dialog Box—Sandbox

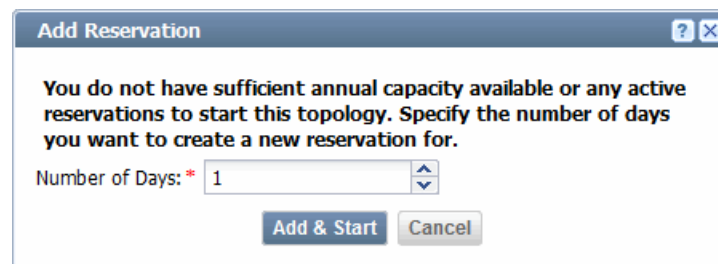


To launch the topology from the Topology Wizard:

1. Select **Start it now** (see [Figure 5 on page 11](#)).
2. If you have already made a reservation with sufficient capacity, the topology starts and the following dialog box appears, indicating that the topology start process has been initiated successfully.

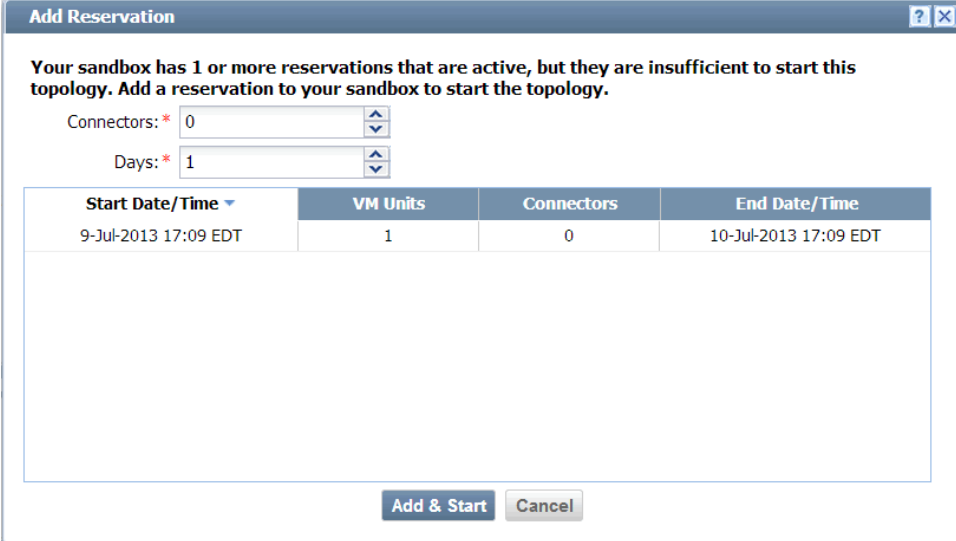


3. If there are no active reservations in your sandbox, the following dialog box appears, enabling you to create a reservation from the Topology Wizard.



- In the Add Reservation dialog box, enter the number of days you want for the reservation.
- Click **Add & Start**. If there is sufficient capacity available in the sandbox, the reservation is created for the exact number of VM units created in the topology and the topology is started.

If there are active reservations, but they are insufficient to start a topology, then the following dialog box appears.



Add Reservation

Your sandbox has 1 or more reservations that are active, but they are insufficient to start this topology. Add a reservation to your sandbox to start the topology.

Connectors: * 0

Days: * 1

Start Date/Time	VM Units	Connectors	End Date/Time
9-Jul-2013 17:09 EDT	1	0	10-Jul-2013 17:09 EDT

Add & Start **Cancel**

- Enter the number of VM units in the topology.
- Enter the number of connectors you want to reserve.
- Enter the number of days for the reservation.
- Click **Create**.



NOTE: If there is sufficient capacity available in the sandbox, a reservation is created.

- Click **OK**. The **Access Active Topology** window appears.

If you saved the topology to a bank library, the following dialog box appears. Once the topology has been saved to the bank, it is visible to all sandboxes in that bank and can be launched directly from the bank library. You must copy the topology to a sandbox before it is run.



NOTE: Only a bank administrator can create a topology in a bank library.

Figure 6: Topology Saved Dialog Box—Bank



Topology Saved

The topology has been saved successfully. The topology contains 1 VMs and uses 1 VM units.

Close wizard **Cancel**

- Related Documentation**
- [Creating Topologies with the Topology Wizard on page 7](#)
 - [Editing a Topology Using the Topology Wizard on page 13](#)

- [Copying a Topology on page 13](#)
- [Topology Wizard Overview on page 5](#)

Editing a Topology

There are two parts to editing a topology:

- [Editing Topology Metadata on page 13](#)
- [Editing a Topology Using the Topology Wizard on page 13](#)

Editing Topology Metadata

To edit a topology's metadata (name, description, and downloadable and savable check boxes), go to **Topologies > Manage Topologies** in the navigation tree and edit the data directly in the **Topologies** section. Then, click the **Save** button located in the lower right corner. This procedure is the same for topologies that were created using the Topology Wizard and for topologies that were uploaded.



NOTE: You cannot edit any fields that appear in bold, blue font.

Editing a Topology Using the Topology Wizard

To edit a topology that was created with the Topology Wizard:

1. Select **Topologies > Manage Topologies** from the navigation tree.
2. Select the topology you want to edit and click the Edit icon from the top left. The Topology Wizard appears with the selected topology displayed.
3. Add information or change the existing information.
4. Click **Save**.

Related Documentation

- [Editing a Topology with .VMM Scripting on page 24](#)
- [Starting a Topology from the Topology Wizard on page 10](#)

Copying a Topology

Junosphere's Copy feature enables users and bank administrators to copy a topology from one sandbox in a bank to other sandboxes in the same bank.

- Bank administrators can copy:
 - Any non-public topologies in their banks to any sandbox or library within their bank(s).
 - Any public, downloadable topologies to any sandbox or library within their bank(s).
- Users can copy:

- Any topology in one sandbox to any other sandbox, regardless of whether the topology is downloadable, as long as the user has Library Management permission.
- Any downloadable topologies in a sandbox in which the user does not have Library Management permission to any sandbox in which the user has Library Management permission.

To copy a topology:

1. Select **Topologies > Manage Topologies** from the navigation tree.
2. Select the topology you want to copy and click the **Copy** icon in the upper left portion of the **Topologies** section.

The following dialog box appears, listing all libraries in which the user has permission to copy the topology.

Figure 7: Copy Topology Dialog Box

Copy Topology

Name: * 1t

Description:

Downloadable: * ☒

Savable: * ☒

Select one or more destination libraries below... *

Sandbox		Bank		
Bank	Sandbox			Name
<input type="checkbox"/> B1	S1			Library23
<input type="checkbox"/> B1	S1			testforwrongcustomer
<input type="checkbox"/> B1	S2			Libr2

Copy Cancel

3. Make any changes to the topology information, including name, description, downloadability, and savability.
4. Select the library or libraries to which you want to copy the topology.
5. Click **Copy**.
6. When the Topology copied message appears, click **OK**.

Related Documentation

- [Creating Topologies with the Topology Wizard on page 7](#)
- [Editing a Topology Using the Topology Wizard on page 13](#)
- [Topology Wizard Overview on page 5](#)

CHAPTER 3

Using Topologies in Public Libraries

- [Topologies in Public Libraries Overview on page 15](#)
- [Using Topologies in Public Libraries on page 18](#)

Topologies in Public Libraries Overview

Select **Topologies > Manage Topologies** and click the **Public** tab to access topologies in public libraries. These topologies cannot be edited or saved. To use one, copy it and save it in another location and make edits to the copy.

There are three different types of topology libraries:

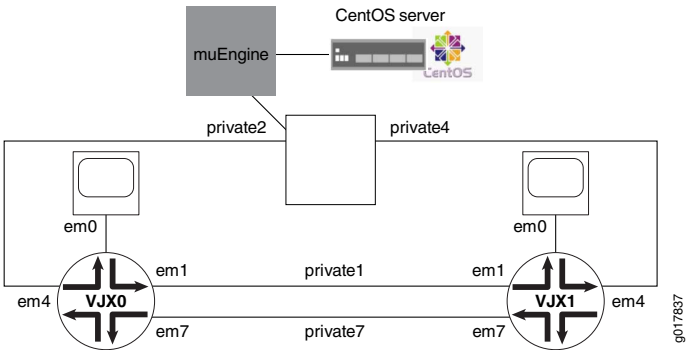
1. Starter topologies, which provide simple, basic topology examples.

Table 4: Starter Topologies

Topology	Description
2VJX+Space+Centos	This topology uses two VJX routers, an instance of Junos Space for network management, and a Centos server to access Junos Space from inside the cloud. Additional documentation for this public topology is included with the topology file set when it is downloaded from www.junosphere.net .
2VJX	This topology uses two VJX routers. The names of connectors are always called "private" and contain a number. Additional documentation for this public topology is included with the topology file set when it is downloaded from www.junosphere.net .
2VJX+Mu Test Engine	This topology uses two VJX routers. The Mu Test Engine is a separate virtual machine (VM) that runs tests as directed by the Mu Studio VM. You must configure the interfaces on the Mu Test engine to work inside your Junosphere network. Additional documentation for this public topology is included with the topology file set when it is downloaded from www.junosphere.net .
VSRX+Mu Studio	This topology uses one VSRX router and Mu Studio. Additional documentation for this public topology is included with the topology file set when it is downloaded from www.junosphere.net .

2. Ecosystem Partner topologies, which demonstrate the use of tools that are available in the Junosphere ecosystem. Partner solutions include Mu Dynamics and WANDL. [Table 5 on page 16](#) lists the examples.

Table 5: Ecosystem Partner Topologies

Topology	Description
Mu Dynamics+2VJX+Centos	 <p>This topology uses five VMs and illustrates the application of the Mu Dynamics analysis tool. This topology also contains two VJX routers and a Centos server that facilitates access to Mu Dynamics from within Junosphere.</p>

3. Advanced topologies, which contain more complex topologies with complete configuration files. These topologies have a larger number of nodes and diverse combinations of protocols. [Table 6 on page 16](#) lists the examples.

Table 6: Advanced Topologies

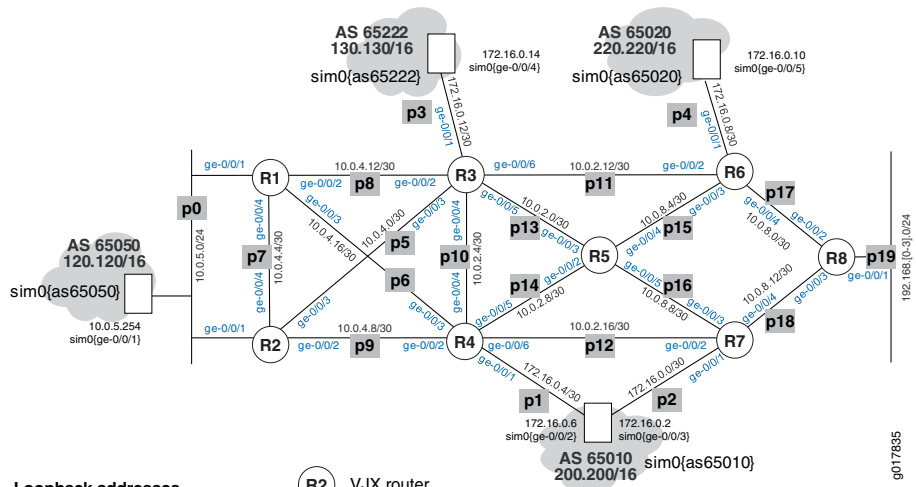
Topology	Description
Demo Baseline Topology	<p>This topology uses nine VMs and consists of eight VJX routers and an additional router connected to routers 1 through 7 that acts as a simulator. The topology is divided into four autonomous systems (ASs).</p>  <p>Loopback addresses 10.0.255.x, x = router number</p> <p>Interface numbering Lower number on the left If vertical, then top is lower number</p> <p>Legend: R2 VJX router p17 Junosphere private bridge (num) [] Simulated router (sim0 instance) routing-instance = {asnumber} </p>

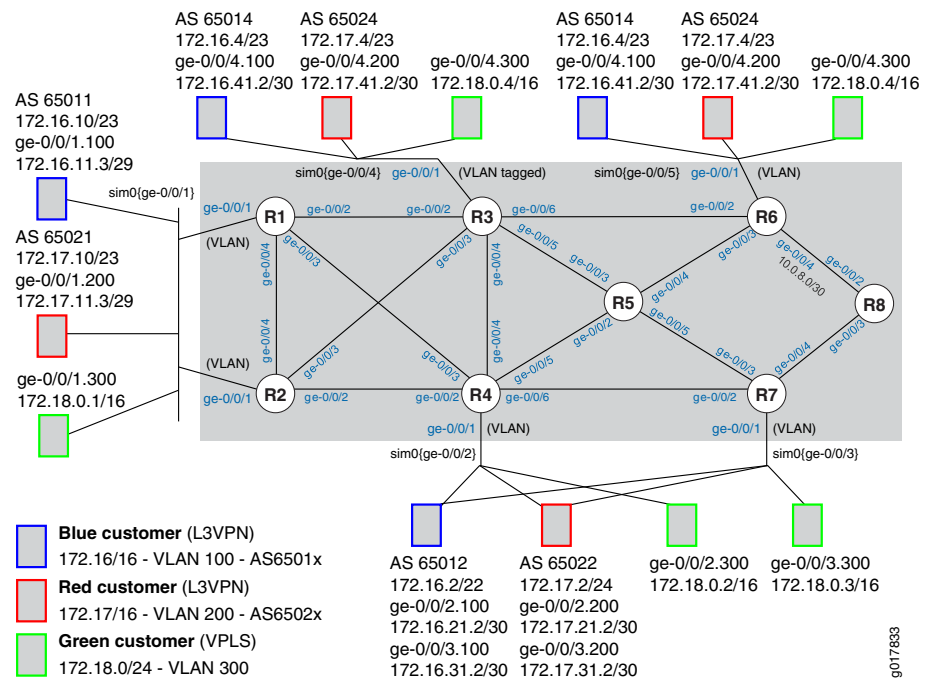
Table 6: Advanced Topologies (continued)

Topology	Description
Demo Enterprise Topology	<p>This topology is similar to the baseline topology in that it uses nine VMs and consists of eight VJX routers and a simulator. Like the baseline topology, it is divided into five autonomous systems (ASs). It also contains three OSPF areas (one for the core) and one IS-IS area.</p> <p>The diagram illustrates a network topology with eight routers (R1-R8) and five Autonomous Systems (ASes). The routers are interconnected in a mesh-like structure. The ASes are: AS 65050 (120.120/16), AS 65222 (130.130/16), AS 65020 (220.220/16), AS 65412 (iBGP full mesh), and AS 65010 (200.200/16). The topology is divided into three OSPF areas: OSPF area 0.0.0.1 (stub), OSPF area 0.0.0.2 (NSSA), and OSPF area 0.0.0.2 (NSSA). It also includes an IS-IS Level 1 area (Area 002). A legend identifies the components: AS 65412 (iBGP full mesh), OSPF backbone area 0.0.0.2, OSPF area, IS-IS Level 1 (Area 002), and eBGP peer.</p>

g017534

Table 6: Advanced Topologies (*continued*)

Topology	Description
Demo MPLS Topology	This topology consists of eight VJX routers with an additional VJX router used as a simulator. This topology is used to demonstrate an MPLS application. The network supports three customers with VPNs, two of which are Layer 3 VPNs; the third works as a VPLS.



Using Topologies in Public Libraries

The public libraries section of Junosphere contains a series of existing topology examples that can serve as a base from which you can create your own topologies.

Most files contain a PowerPoint presentation that graphically depicts the topology, as well as configuration files (.conf) that contain sample scripts.

To start a topology from these public libraries:

1. Determine the number of VM units required for the topology.
2. Select **Topologies > Manage Topologies**.
3. Click the **Public** tab. A list of topologies appears.
4. Check the box next to the topology that you want. A graphical representation of that topology appears in the Preview section.
5. Click the **Start** button at the bottom right of the **Topologies** section.
6. A dialog box appears asking you to select the sandbox in which you want to start the topology. After you select the sandbox, click the **Start** button.

If you do not have an existing reservation for the topology you want to start, a dialog box will appear and prompt you to add a reservation. If there is an active reservation, but it has insufficient capacity, a different dialog box appears, notifying you that you do not have sufficient capacity.

**Related
Documentation**

- [Creating Topologies with the Topology Wizard on page 7](#)
- [Editing a Topology Using the Topology Wizard on page 13](#)
- [Topology Wizard Overview on page 5](#)

CHAPTER 4

Building Topologies with VMM Scripting

This chapter provides information on defining the virtual network topology using Virtual Machine Manager (VMM) language syntax.

- [The topology.vmm File on page 21](#)
- [.vmm File Syntax on page 23](#)
- [Editing a Topology with .VMM Scripting on page 24](#)
- [Using Macros on page 25](#)
- [Advanced Parsing Capabilities on page 32](#)
- [Parsing Concepts on page 32](#)
- [Parser Syntax Rules on page 33](#)
- [Viewing VM Images on page 33](#)

The topology.vmm File

After understanding the customer's requirements, purpose, and goals for the network, you can define the virtual network topology using Virtual Machine Manager (VMM) language syntax.

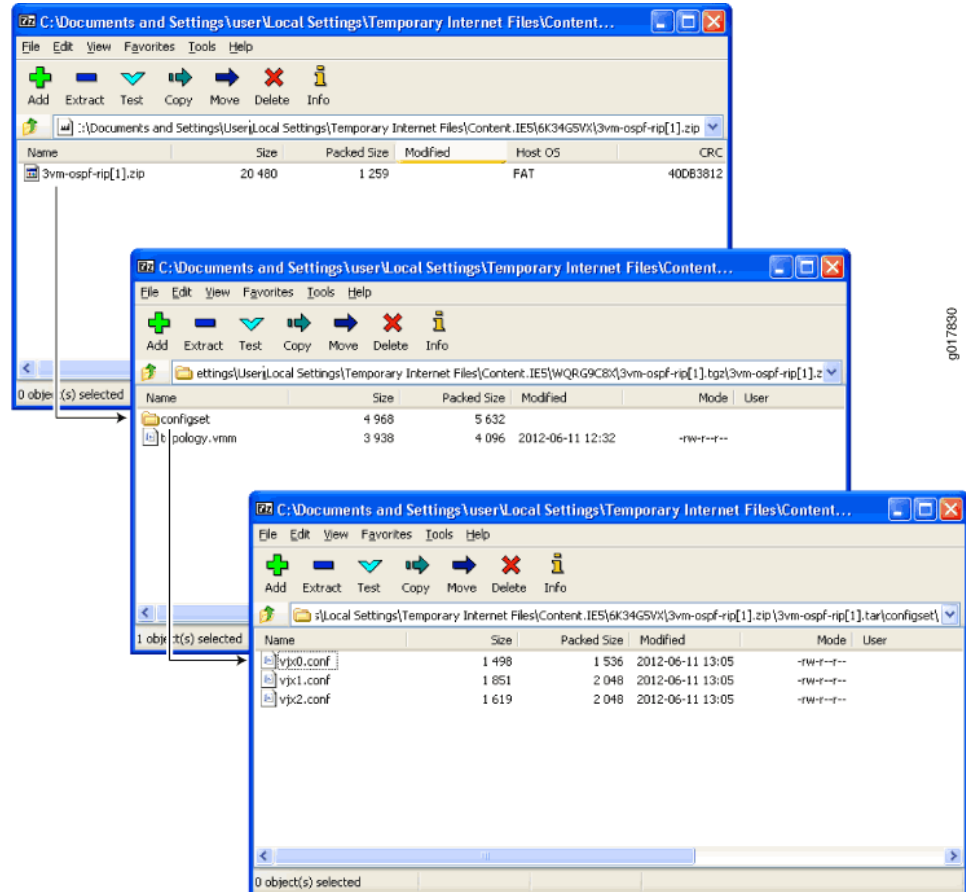
To create a virtual network using the VMM syntax, you need to define the following:

- .vmm file default headers and syntax components.
- Definition of each virtual machine, including:
 - The virtual machine name, such as `vrouter#`, where `#` is a number.
 - The virtual image to boot that defines the type of virtual machine. (Refer to the *Junosphere Release Notes 3.0* for a complete list of supported images.) You can also view a list of supported images in Junosphere by selecting **Configuration > View VM Images** in the navigation tree.
 - Virtual machine interfaces to map.
 - A link to a configuration file (optional).
- The connections between the virtual machines (VMs).

- To connect interfaces between two virtual machines, define them to be on the same bridge.
- Optional comments.

A topology file set is a compressed file (.zip or .tgz) that has the structure shown in [Figure 8 on page 22](#).

Figure 8: Topology file set Structure



NOTE: To create the `topology.vmm` file, use any text editor. To integrate or to open files in your file set, use 7zip (for zip compression) or any compression tool to create a zip/tgz compression).

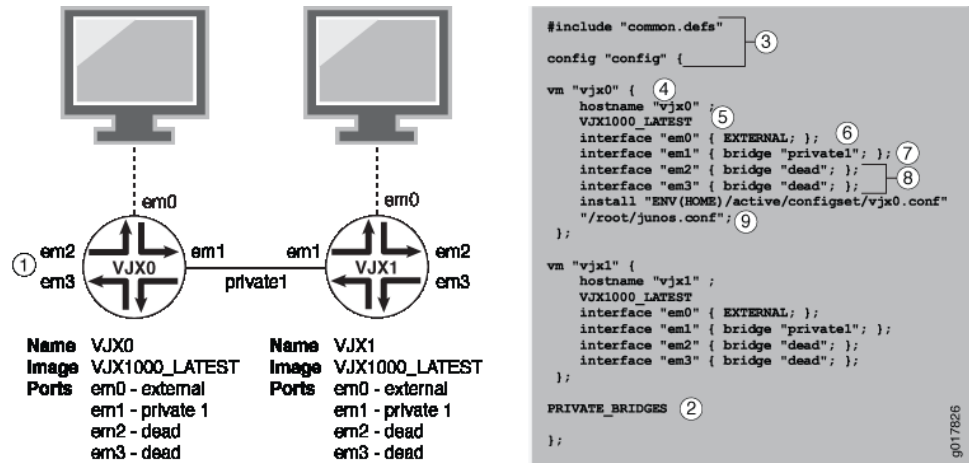
Related Documentation

- [.vmm File Syntax on page 23](#)
- [Editing a Topology with .VMM Scripting on page 24](#)
- [.vmm File Syntax on page 23](#)

.vmm File Syntax

Consider the example of a `topology.vmm` file shown in Figure 9 on page 23.

Figure 9: Example `topology.vmm` File



- ① Description.
- ② Default closing line. Must be included exactly as shown.
- ③ Default headers. Must be included exactly as shown.
- ④ First VM name. Can be any string that begins with a lower or uppercase letter, followed by one or more lower or upper case letters, digits, or special characters "-" or ".".
- ⑤ Virtual Image. Does not have to be VJX_LATEST.
- ⑥ First interface. Will always be em0 and will point to EXTERNAL (for CLI connectivity).
- ⑦ Interface em1 points to "private1", which connects it to em1 on vjx1. Must always use "private#" naming convention.
- ⑧ Interfaces defined, but not used. Always point to "dead" (reserved word).
- ⑨ Link to the config file in the fileset. Filename is vx0.conf.

Other objects are defined within context:

- Virtual machines and bridges can only be defined within the `config` object.
- Interfaces can only be defined within a virtual machine.



NOTE: A topology file set must be less than 10 MB; each file inside of the file set must be less than 6 MB.

The configuration file is applied as part of the virtual machine's bootstrap process.

Table 7 on page 23 shows the syntax of the `vmm` configuration file.

Table 7: .vmm File Syntax

Configuration	Configuration Value
config_statement	vm_definition
vm_definition	vm name_value { vm_statement + };

Table 7: .vmm File Syntax (*continued*)

Configuration	Configuration Value
vm_statement	install <i>string_value</i> <i>string_value</i> ;
vm_statement	interface <i>definition</i> ;
interface_definition	interface <i>name_value</i> { <i>interface_statement</i> + };
interface_statement	bridge <i>name_value</i> ;
interface_statement	macaddr <i>string_value</i> ;
interface_statement	ipaddr <i>string_value</i> ;
name_value	<i>string_value</i>

- Related Documentation**
- [The topology.vmm File on page 21](#)
 - [.vmm File Syntax on page 23](#)
 - [.vmm File Syntax on page 23](#)

Editing a Topology with .VMM Scripting

A topology file that was created with scripting cannot be edited in the Topology Wizard, so the only way to edit it is to upload another topology file set in its place.

To edit a topology with .vmm scripting:

1. After you have edited a .vmm file, you can upload it to Junosphere. From the navigation tree, select **Topologies > Manage Topologies**.
2. Click the **Upload topology** icon located in the top left side of the **Topologies** section.
3. In the **Upload Topology** dialog box, select a bank and library and choose a name for the topology. Add a description, if desired, and select whether the topology is downloadable or savable.
4. Then, select the file set for the topology that you edited by clicking the **Browse** button.



NOTE: You must upload the file set in the .zip or .tgz format.

5. Click the **Upload** icon.

The .vmm file is uploaded.

- Related Documentation**
- [The topology.vmm File on page 21](#)
 - [.vmm File Syntax on page 23](#)

- [Using Macros on page 25](#)

Using Macros

This section lists and describes some common macros that are available to all users.

CentOS Server

Name: CENTOS_6_3

Category: Application Server

VM Units: 1

RAM: 2 GB

CPU: 1

Permission: All

Description: CentOS 32-bit and 64-bit images.

Usage:

```
#include "common.defs"

config "config" {

  vm "centos01" {
    hostname "centos01" ;

    CENTOS_6_3
    interface "em0" { EXTERNAL; };
  };
  PRIVATE_BRIDGES

};
```

Junos Space

Name: SPACE_LATEST

Category:Juniper Networks-supported

VM Units: 1

RAM: 8 GB

CPU: 1

Permission: All

Description: Latest qualified Junos Space image. This currently maps to SPACE_12_3P_8.

Usage:

```
#include "common.defs"

config "config" {

    SPACE_LATEST

    vm "CTS0"{
        hostname "CTS0";
        CENTOS_5_3

        interface "em0" { EXTERNAL; };
    };

    vm "VJX0"{
        hostname "VJX0";
        VJX_LATEST

        interface "em0" { EXTERNAL; }; /* ge-0/0/0 */

        install "ENV(HOME)/active/configset/VJX0.conf" "/root/junos.conf";
    };

    PRIVATE_BRIDGES

};
```

Mu Studio Performance

Name: MU_STUDIO_LATEST

Category: Partner

VM Units: 1

RAM: 2 GB

CPU: 1

Permission: All

Description: Latest qualified Mu Studio image.

Usage:

```
#include "common.defs"

config "config" {

    vm "MUS0" {
        hostname "MUS0" ;

        MU_STUDIO_LATEST
    };
    PRIVATE_BRIDGES

};
```

Mu Test Engine

Name: MU_TESTENGINE_LATEST

Category: Partner

VM Units: 1

RAM: 2 GB

CPU: 1

Permission: All

Description: Latest qualified Test Engine image.

Usage:

```
#include "common.defs"

config "config" {

    vm "MUT0"{
        hostname "MUT0";
        MU_TESTENGINE_LATEST

        interface "em2" { bridge "private0"; macaddr MU_ENGINE_MAC1; };
        interface "em3" { bridge "private1"; macaddr MU_ENGINE_MAC2; };
    };

    vm "VJX0"{
        hostname "VJX0";
        VJX_LATEST

        interface "em0" { EXTERNAL; }; /* ge-0/0/0 */
        interface "em1" { bridge "private0"; }; /* ge-0/0/1 */

        install "ENV(HOME)/active/configset/VJX0.conf" "/root/junos.conf";
    };

    vm "VJX1"{
        hostname "VJX1";
        VJX_LATEST

        interface "em0" { EXTERNAL; }; /* ge-0/0/0 */
        interface "em1" { bridge "private1"; }; /* ge-0/0/1 */

        install "ENV(HOME)/active/configset/VJX1.conf" "/root/junos.conf";
    };

    PRIVATE_BRIDGES

};
```

VJX

Name: VJX1000_LATEST, VJX_LATEST

Category: Juniper Networks-supported

VM Units: 1

RAM: 2 GB

CPU: 1

Permission: All

Description: J-Series Router based on Junos12.3 CentOS Virtual Machine configurable up to 64 Ethernet Interfaces.

Usage:

```
#include "common.defs"

config "config" {

    vm "P1" {
        hostname "P1";
        VJX_LATEST
        interface "em0" { EXTERNAL; }; /* ge-0/0/0 – Management interface to P2 */
        interface "em1" { bridge "private1"; }; /* ge-0/0/1 - P1 connected to P2 */
        interface "em2" { bridge "private2"; }; /* ge-0/0/2 */
        interface "em3" { bridge "private3"; }; /* ge-0/0/3 */
        interface "em4" { bridge "private4"; }; /* ge-0/0/4 */
        interface "em5" { bridge "private5"; }; /* ge-0/0/5 */

        install "ENV(HOME)/active/configset/P1.conf" "/root/junos.conf";
    };

    vm "P2" {
        hostname "P2";
        VJX1000_LATEST
        interface "em0" { EXTERNAL; }; /* ge-0/0/0 - Management interface to P2 */
        interface "em1" { bridge "private1"; }; /* ge-0/0/1 – P2 connected to P1 */

        install "ENV(HOME)/active/configset/P2.conf" "/root/junos.conf";
    };

    PRIVATE_BRIDGES
};
```

VMX

Name: VMX_LATEST

Category: Experimental

VM Units: 2

RAM: 4 GB

CPU: 1

Permission: All

Description: Latest qualified VMX image.

Usage:

```
#include "common.defs"

config "config" {

    VMX_EXP(VMX0, VMX0_cosim, private_VMX0)

    interface "em2" { bridge "private0"; }; /* ge-0/0/1 */
    interface "em3" { bridge "private1"; }; /* ge-0/0/2 */
    interface "em4" { bridge "private2"; }; /* ge-0/0/3 */

    install "ENV(HOME)/active/configset/VMX0.conf" "/root/junos.conf";
};

vm "VJX0"{
    hostname "VJX0";
    VJX_LATEST

    interface "em0" { EXTERNAL; }; /* ge-0/0/0 */
    interface "em1" { bridge "private0"; }; /* ge-0/0/1 */
    interface "em2" { bridge "private2"; }; /* ge-0/0/2 */

    install "ENV(HOME)/active/configset/VJX0.conf" "/root/junos.conf";
};

VMX_EXP(VMX1, VMX1_cosim, private_VMX1)

    interface "em2" { bridge "private1"; }; /* ge-0/0/1 */
    interface "em3" { bridge "private2"; }; /* ge-0/0/2 */
    interface "em4" { bridge "private0"; }; /* ge-0/0/3 */

    install "ENV(HOME)/active/configset/VMX1.conf" "/root/junos.conf";
};

PRIVATE_BRIDGES

};
```

VPTX

Name: VPTX_LATEST, VPTX_EXP

Category: Experimental

VM Units: 2

RAM: 2 GB

CPU:1

Permission: All

Description: Latest qualified VPTX image.

Usage:

```
#include "common.defs"

config "config" {

    vm "VPTX0"{
        hostname "VPTX0";
        VPTX_LATEST

        interface "em0" { EXTERNAL; };
        interface "em1" { bridge "private0"; };
    };

    vm "VPTX1"{
        hostname "VPTX1";
        VPTX_LATEST

        interface "em0" { EXTERNAL; };
        interface "em1" { bridge "private1"; };
    };

    vm "VJX0"{
        hostname "VJX0";
        VJX_LATEST

        interface "em0" { EXTERNAL; }; /* ge-0/0/0 */
        interface "em1" { bridge "private0"; }; /* ge-0/0/1 */
        interface "em2" { bridge "private1"; }; /* ge-0/0/2 */

        install "ENV(HOME)/active/configset/VJX0.conf" "/root/junos.conf";
    };

    PRIVATE_BRIDGES

};
```

VSRX

Name:VSRX_LATEST, VSRX_LATEST_EXP, VSRX_EXP

Category: Experimental

VM Units: 1

RAM: 2 GB

CPU: 1

Permission: All

Description: Latest qualified VSRX image.

Usage:

```
#include "common.defs"

config "config" {

  vm "VSRX0"{
    hostname "VSRX0";
    VSRX_LATEST

    interface "em0" { EXTERNAL; };
    interface "em1" { bridge "private0"; };
  };

  vm "VJX0"{
    hostname "VJX0";
    VJX_LATEST

    interface "em0" { EXTERNAL; }; /* ge-0/0/0 */
    interface "em1" { bridge "private0"; }; /* ge-0/0/1 */
    interface "em2" { bridge "private1"; }; /* ge-0/0/2 */
    interface "em3" { bridge "private2"; }; /* ge-0/0/3 */

    install "ENV(HOME)/active/configset/VJX0.conf" "/root/junos.conf";
  };

  vm "VSRX1"{
    hostname "VSRX1";
    VSRX_LATEST

    interface "em0" { EXTERNAL; };
    interface "em1" { bridge "private1"; };
  };

  vm "VSRX2"{
    hostname "VSRX2";
    VSRX_LATEST

    interface "em0" { EXTERNAL; };
    interface "em1" { bridge "private2"; };
  };

  PRIVATE_BRIDGES

};
```

WANDL

Name: WANDL_LATEST

Category: Partner

VM Units: 1

RAM: 4 GB

CPU:1

Permission: All

Description: Latest qualified WANDL image.

Usage:

```
#include "common.defs"

config "config" {

    WANDL_LATEST

    PRIVATE_BRIDGES

};
```

**Related
Documentation**

- [The topology.vmm File on page 21](#)
- [.vmm File Syntax on page 23](#)
- [.vmm File Syntax on page 23](#)

Advanced Parsing Capabilities

To reduce the number of user errors and enhance usability, the Junosphere parsing feature provides the following capabilities:

- Upon upload of topology, the parser will:
 - Parse the **topology.vmm** file syntax before it is loaded into a library. This process ensures that all topologies in the libraries are correct.
 - Identify the number and type of network elements required for a topology.
- Upon start of topology, the parser will verify that image types invoked in the topology exist in the system at start time.

**Related
Documentation**

- [The topology.vmm File on page 21](#)
- [.vmm File Syntax on page 23](#)
- [.vmm File Syntax on page 23](#)

Parsing Concepts

Parsing relies on the following concepts:

- VM (virtual machine) — Includes Juniper Networks devices (such as VJX, Junos Space), partner tools, and an application server (CentOS).

- `em#` — Nomenclature that indicates the interface ports of the virtual network elements. `em0` is reserved for the managed interface.
- `private#` — Each `private#` refers to a bridge that enables a connection between interfaces of virtual machines.

**Related
Documentation**

- [The topology.vmm File on page 21](#)
- [.vmm File Syntax on page 23](#)
- [.vmm File Syntax on page 23](#)

Parser Syntax Rules

The following rules are enforced in the Junosphere parser:

- Valid VM names—A valid name begins with an alphabetic character (either `a – z` or `A – Z`), followed by an alphanumeric character (`a – z`, `A – Z`, `0 – 9`) or a special character that is either an underscore or a hyphen. Minimum length is one character; maximum length is 20 alphanumeric or special characters. Any character that is not listed here is not compatible.
- Bridge names—`Private0` to `private123` only. Custom definitions are not allowed.
- VM macro names—Refer to the *Junosphere Release Notes 2.6* for valid names; pathnames are no longer accepted.
- Commands—Commands such as **install** for a partner's VM are not compatible.

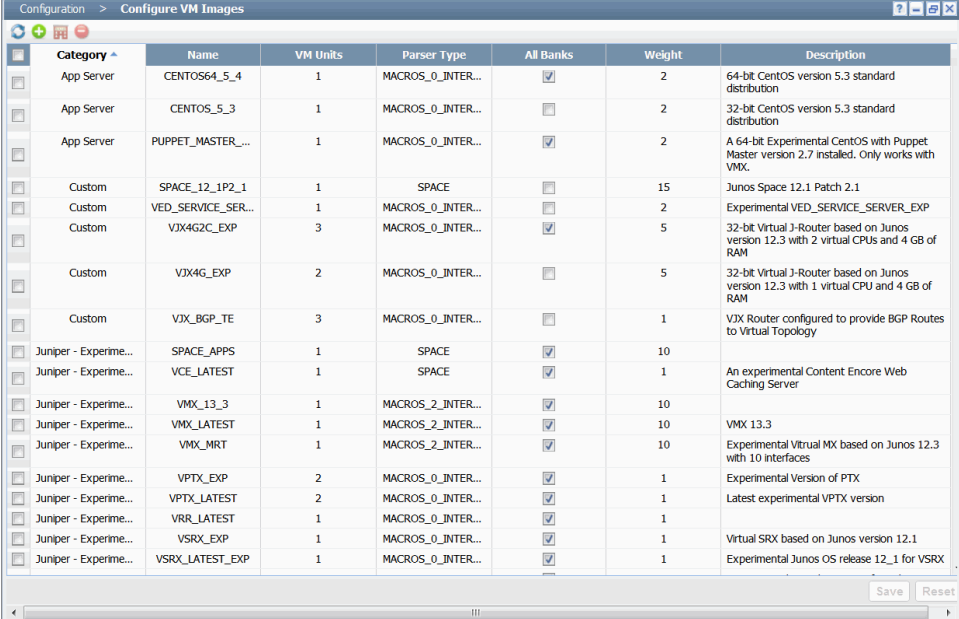
**Related
Documentation**

- [The topology.vmm File on page 21](#)
- [.vmm File Syntax on page 23](#)
- [.vmm File Syntax on page 23](#)

Viewing VM Images

To view information—such as name, category, the number of VM units, about a VM image—select **Configuration > View VM Images**. The following screen appears.

Figure 10: View VM Images Window



Category	Name	VM Units	Parser Type	All Banks	Weight	Description
App Server	CENTOS64_5_4	1	MACROS_0_INTER...	<input checked="" type="checkbox"/>	2	64-bit CentOS version 5.3 standard distribution
App Server	CENTOS_5_3	1	MACROS_0_INTER...	<input type="checkbox"/>	2	32-bit CentOS version 5.3 standard distribution
App Server	PUPPET_MASTER_...	1	MACROS_0_INTER...	<input checked="" type="checkbox"/>	2	A 64-bit Experimental CentOS with Puppet Master version 2.7 installed. Only works with VMX.
Custom	SPACE_12_1P2_1	1	SPACE	<input type="checkbox"/>	15	Junos Space 12.1 Patch 2.1
Custom	VED_SERVICE_SER...	1	MACROS_0_INTER...	<input type="checkbox"/>	2	Experimental VED_SERVICE_SERVER_EXP
Custom	VJX4GZC_EXP	3	MACROS_0_INTER...	<input checked="" type="checkbox"/>	5	32-bit Virtual J-Router based on Junos version 12.3 with 2 virtual CPUs and 4 GB of RAM
Custom	VJX4G_EXP	2	MACROS_0_INTER...	<input type="checkbox"/>	5	32-bit Virtual J-Router based on Junos version 12.3 with 1 virtual CPU and 4 GB of RAM
Custom	VJX_BGP_TE	3	MACROS_0_INTER...	<input type="checkbox"/>	1	VJX Router configured to provide BGP Routes to Virtual Topology
Juniper - Experime...	SPACE_APPS	1	SPACE	<input checked="" type="checkbox"/>	10	
Juniper - Experime...	VCE_LATEST	1	SPACE	<input checked="" type="checkbox"/>	1	An experimental Content Encore Web Caching Server
Juniper - Experime...	VMX_13_3	1	MACROS_2_INTER...	<input checked="" type="checkbox"/>	10	
Juniper - Experime...	VMX_LATEST	1	MACROS_2_INTER...	<input checked="" type="checkbox"/>	10	VMX 13.3
Juniper - Experime...	VMX_MRT	1	MACROS_2_INTER...	<input checked="" type="checkbox"/>	10	Experimental Virtual MX based on Junos 12.3 with 10 interfaces
Juniper - Experime...	VPTX_EXP	2	MACROS_0_INTER...	<input checked="" type="checkbox"/>	1	Experimental Version of PTX
Juniper - Experime...	VPTX_LATEST	2	MACROS_0_INTER...	<input checked="" type="checkbox"/>	1	Latest experimental VPTX version
Juniper - Experime...	VRR_LATEST	1	MACROS_0_INTER...	<input checked="" type="checkbox"/>	1	
Juniper - Experime...	VSRX_EXP	1	MACROS_0_INTER...	<input checked="" type="checkbox"/>	1	Virtual SRX based on Junos version 12.1
Juniper - Experime...	VSRX_LATEST_EXP	1	MACROS_0_INTER...	<input checked="" type="checkbox"/>	1	Experimental Junos OS release 12.1 for VSRX

Click the down arrow in any column to customize the display.

Related Documentation

- [Creating Topologies with the Topology Wizard on page 7](#)
- [Starting a Topology from the Topology Wizard on page 10](#)
- [Editing a Topology Using the Topology Wizard on page 13](#)

CHAPTER 5

Definition of Options in the VMM File

- [Using the ipaddr Interface Option on page 35](#)
- [VJX Virtual Machines Interface Options on page 36](#)
- [Topology Configuration Example on page 37](#)
- [Sample .vmm Configuration File on page 37](#)
- [Creating the Virtual Machine Configuration on page 39](#)
- [Building and Testing Your Network Topology on page 43](#)

Using the ipaddr Interface Option

Use the `ipaddr` option to assign an IP address to an interface from the `topology.vmm` file. Use this option with care, because while you can use your existing hostnames and IP addresses in the Junosphere topology models, and while Junosphere is architected based on secure virtual private network (VPN) paradigms, we recommend that you use alternative names and addresses in your topology files for additional security.

The following example shows the `ipaddr` interface option:

```
interface "emx" { bridge "<private#>"; ipaddr "aaa.bbb.ccc.ddd"; };
```

- The `ipaddr` function should *not* be used if a full router configuration is being loaded from the library.
 - A VMM `ipaddr`-assigned address will override an IP address contained in a Junos OS router configuration.
- The `ipaddr` function in VMM can be used for the quick creation of configured and addressed interfaces.
- By default, a /20 subnet mask will be applied to the interface; this cannot be changed.
- Take care to ensure that the assigned address of interface X is not in the same subnet as an address assigned to interface Y, otherwise the configuration will be rejected when the VJX virtual machine boots up. In the example below, R0 has two interfaces that are being configured with family inet addresses. The addresses (192.85.1.1 and 192.86.2.1) do not reside in the same subnet when the /20 subnet mask is applied.

In the example below, R0 has two interfaces that are being configured with family inet addresses:

```
vm "vrouter001" {  
  hostname "vrouter001" ;  
  VJX1000_LATEST  
  interface "em0" { EXTERNAL;};  
  interface "em1" { bridge "private0"; ipaddr "192.85.1.1"; };  
  interface "em2" { bridge "private2"; ipaddr "192.86.2.1"; };  
}
```

For CentOS VM users, this code should look like the following example:

```
vm "centos" {  
  hostname "centos" ;  
  CENTOS_5_3  
  interface "em0" { EXTERNAL; };  
};
```

Related Documentation

- [Creating Topologies with the Topology Wizard on page 7](#)
- [Starting a Topology from the Topology Wizard on page 10](#)
- [Editing a Topology Using the Topology Wizard on page 13](#)

VJX Virtual Machines Interface Options

VMM interfaces for VJX virtual machines are mapped as follows:

- VMM 'em0' = VJX 'ge-0/0/0'
- VMM 'em1' = VJX 'ge-0/0/1'
- VMM 'em2' = VJX 'ge-0/0/2'
- VMM 'em15' = VJX 'ge-0/0/15'

The VMM interfaces must always be sequential.

The following lists the options to consider when defining interfaces in a .vmm file:

- The *em0* / *ge-0/0/0* interface is reserved for out-of-band (OOB) management access. You cannot configure an IP address, features, or routing protocols against this interface.
- Additional interfaces can be configured with VLANs using the Junos OS router configuration file to create more logical interfaces.
- Interfaces must be listed sequentially, without gaps.
- Virtual machines must share a common segment (bridge) to communicate with each other.

```
interface "<em#>" { bridge "private0"; };
```


- EXTERNAL indicates a bridge to the management LAN outside the cloud. The EXTERNAL segment is created by the Junosphere environment as the LAN to which both the user (via the Secure Access service) and the management Ethernet port of every virtual machine in the user's topology can be connected.
- Only IPv4 addresses are currently supported in the Virtual Machine Manager configuration syntax. Other address families must be configured within the virtual machine.
- Refer to the *Junosphere Release Notes 2.6* for the current number of supported interfaces.

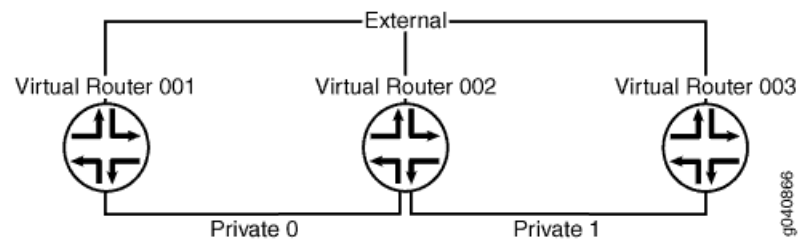
Related Documentation

- [Creating Topologies with the Topology Wizard on page 7](#)
- [Starting a Topology from the Topology Wizard on page 10](#)
- [Editing a Topology Using the Topology Wizard on page 13](#)

Topology Configuration Example

Figure 11 on page 37 defines a simple **topology.vmm** configuration.

Figure 11: Simple topology.vmm Configuration



The network devices `vrouter001`, `vrouter002`, and `vrouter003` are virtual machines representing virtual routers. `Private0` and `Private1` are bridges associated with network segments connecting `vrouter001` to `vrouter002` and `vrouter002` to `vrouter003`, respectively. In order for the virtual machines to be able to communicate with each other, they must share a common bridge.

Related Documentation

- [Creating Topologies with the Topology Wizard on page 7](#)
- [Starting a Topology from the Topology Wizard on page 10](#)
- [Editing a Topology Using the Topology Wizard on page 13](#)

Sample .vmm Configuration File

The following is an example of a `.vmm` configuration file that you can use as a starting configuration. To use this as code, copy and paste the text into an ASCII text editor such as Notepad and copy it again to eliminate any extra non-ASCII characters.

```
// description - global definitions.
//
```

```
#include "common.defs"
config "config" {

    vm "vruter001" {
        // description - hostname of set on VM
        hostname "vruter001";

        // description Operating system image to load
        VJX_LATEST

        // description - ge 0/0/0 management interface
        interface "em0" { EXTERNAL;};

        // description - ge 0/0/1 link between vruter001 and vruter002
        interface "em1" { bridge "private0"; };

        // description - configuration file to load on the router
        install "ENV(HOME)/active/configset/vruter001.conf" "/root/junos.conf";
    };
    vm "vruter002" {
        hostname "vruter002" ;

        VJX_LATEST

        interface "em0" { EXTERNAL;};

        // description - link between vruter002 and vruter001
        interface "em1" { bridge "private0"; };

        // description - link between vruter002 and vruter003
        interface "em2" { bridge "private1"; };

        // description - configuration file to load on the router
        install "ENV(HOME)/active/configset/vruter002.conf" "/root/junos.conf";
    };

    vm "vruter003" {
        hostname "vruter003" ;

        VJX_LATEST

        interface "em0" { EXTERNAL;};

        // description - link between vruter003 and vruter002
        interface "em1" { bridge "private1"; };

        // description - configuration file to load on the router
        install "ENV(HOME)/active/configset/vruter003.conf" "/root/junos.conf";
    };

    PRIVATE_BRIDGES

};
```

- Related Documentation**
- [Creating the Virtual Machine Configuration on page 39](#)
 - [Starting a Topology from the Topology Wizard on page 10](#)
 - [Editing a Topology Using the Topology Wizard on page 13](#)

Creating the Virtual Machine Configuration

To configure the virtual machines:

- For Junos OS network devices such as the VJX1000, write, build, and edit the Junos OS configurations.
- For other virtual machines, configure and boot those virtual machines.

This topic contains the following sections:

- [Configuring Junos OS Network Devices on page 39](#)
- [Rules for Junos OS Configuration Files on page 39](#)
- [Configuring Other Virtual Machines on page 40](#)
- [Configuring CentOS Servers on page 40](#)
- [Default Configuration File on page 41](#)
- [Loading a Junos OS Configuration File on page 43](#)

Configuring Junos OS Network Devices

To configure a Junos OS network device:

1. Manually compose a new Junos OS configuration file or edit an existing one for each virtual machine that uses a text editor.
2. Submit the virtual machine configuration files with the **topology.vmm** file as part of the topology file set, as listed in [“Building a Topology File Set” on page 44](#).

Subsequent to the initial submission, log in to the network devices and use the CLI to update the configurations, as discussed in [“Connecting to the Network Topology” on page 44](#).

Rules for Junos OS Configuration Files

The following rules apply to the Junos OS configuration files:

- The **topology.vmm** file must be coded to install each virtual machine configuration file from its library location, as noted in the following coding example:

```
install "ENV(HOME)/active/configset/vrouter001.conf" "/root/junos.conf";
```

The *install* command defines the active configuration. As part of the virtual machine's bootup process, the configuration file will be applied.
- Junosphere applies a default configuration to each Junos OS network device:

- A default configuration file is prepended to the configuration file you supply.
- Your configuration takes precedence over the default file.
- The default configuration will set up a root name and password, as well as basic services such as *telnet* and *ssh*.
- The default configuration is found in [“Default Configuration File” on page 41](#).
- Interface *ge-0/0/0* gets an assigned management IP address:
 - Do not otherwise configure this management interface.
 - Do not enable the interior gateway protocol (IGP) on this interface.

Configuring Other Virtual Machines

For other virtual machines, such as WANDL and Junos Space, configure and boot those virtual machines. Refer to the device user documentation.

Configuring CentOS Servers

To configure CentOS servers, you must have the following software installed on your local machine:

- Virtual network computing client, such as RealVNC.
- A file transfer application, such as FileZilla.

Make sure that you have an active topology and know the IP address. To configure CentOS servers:

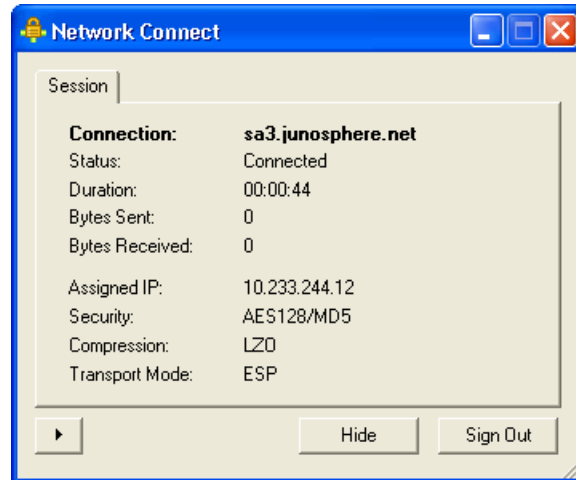
1. On the Active Topology tab, click **Join**. The Junos Space Access Portal appears.
2. Enter your Junosphere credentials to log in to the portal.
3. When the Network Connect screen appears, click **Start**. When connected, a “lock” icon appears in the bottom right part of your screen.



4. Launch your VPN client.
5. Enter the IP address of the CentOS machine(s) to which you want to connect and add **:1** to the address. This gives you access to port 1.
6. Click **OK**.
7. When prompted to enter your password, enter **Clouds** and click **OK**. The CentOS desktop appears.
8. Click the browser icon in the menu bar.

9. When the browser appears, enter the Space IP address in the address field.
10. Double-click the lock icon. When the Network Connect dialog box appears (Figure 12 on page 41), click **Sign Out**.

Figure 12: Network Connect Dialog Box



NOTE: If you do not sign out from your session, any unused secure access sessions will time out and the topology will be inaccessible until you re-establish the secure access session.

Default Configuration File

The following example shows the default Junos OS configuration file that is prepended to the configuration files that you submit.

```
groups {
  member0 {
    system {
      host-name XXXhostnameXXX;
    }
  }
  global {
    system {
      root-authentication {
        encrypted-password "$1$SGUyJfYE$r5hIy2IU4Iam01ye3u70v0"; ##
SECRET-DATA
      }
    }
    login {
      message "Welcome to the cloud\npassword is Clouds\n";
      time-zone America/Los_Angeles;
      debugger-on-panic;
      debugger-on-break;
      dump-on-panic;
      name-server {
        8.8.8.8;
      }
      services {
        finger;
      }
    }
  }
}
```

```
        ftp;
        rlogin;
        rsh;
        ssh;
        telnet;
        xnm-clear-text;
    }
    security {
        forwarding-options {
            family {
                inet6 {
                    mode packet-based;
                }
                mpls {
                    mode packet-based;
                }
                iso {
                    mode packet-based;
                }
            }
        }
    }
    syslog {
        host log {
            kernel info;
            any notice;
            pfe info;
            interactive-commands any;
        }
        file messages {
            kernel info;
            any notice;
            authorization info;
            pfe info;
            archive world-readable;
        }
        file security {
            interactive-commands any;
            archive world-readable;
        }
    }
    processes {
        routing enable;
        management enable;
        watchdog enable;
        snmp enable;
        inet-process enable;
        mib-process enable;
    }
}
chassis {
    dump-on-panic;
}
}
apply-groups [ global member0 ];
system {
    archival {
        configuration {
            transfer-on-commit;
            archive-sites {
```

```

        "ftp://tftp:tftp@XXXgatewayXXX/active/configset";
    }
}
}

```

Loading a Junos OS Configuration File

For all Juniper Networks virtual routers, you can load an existing interface configuration file from the terminal by using the **load** configuration mode command at the **[edit interfaces]** hierarchy level.

1. Copy an existing interface configuration file from the terminal, making sure the syntax of the configuration file is correct.
2. At the console of the Juniper Networks virtual router, enter configuration mode:

```

user@host> cli
[edit]
user@host#

```

3. From configuration mode, issue the **load merge terminal relative** command at the **[edit interfaces]** hierarchy level.

```

[edit interfaces]
user@host# load merge terminal relative

```

4. At the prompt, paste the contents of the clipboard and press **Enter**.
5. Press **Ctrl D**.
6. Commit the configuration to activate it on the virtual router.

For more information about the **load** command, see the *Junos OS CLI User Guide*. To learn more about loading a configuration from a file for Junos OS routers, see *Loading a Configuration from a File* and *Examples: Loading a Configuration from a File*.

Building and Testing Your Network Topology

After you have completed the configuration of your virtual network, perform the following tasks:

- [Building a Topology File Set on page 44](#)
- [Connecting to the Network Topology on page 44](#)

Building a Topology File Set

The process for building a topology file set depends on your operating system. Windows users can use WinZip or another Windows-based compression tool to create **.zip** files.

UNIX and Linux users can build a topology file set with a **.tgz** extension by performing the following steps:

1. Create the directory structure specified below:
 - `<home>/active/`
 - `<home>/active/configset/`
2. Add **topology.vmm**, the VMM configuration file that defines the topology, to the active directory.
3. Add the Junos OS configuration files for each network device referenced by **topology.vmm** to the `<home>/active/configset/` directory.
4. Create a topology file set, suitable for submission, by doing the following for **.tgz** file types:
 - a. Change to the `<home>/active/` directory.

```
cd <home>/active
```

- b. Tar the files.

```
tar -cvf mytopology.tar *
```

- c. Zip the files.

```
gzip mytopology.tar
```

This creates the *mytopology.tar.gz* file.

5. Change the extension on the file from *mytopology.tar.gz* to *mytopology.tgz*.

Topology file sets must have a **.zip** or **.tgz** extension to be uploaded.

Connecting to the Network Topology

To connect to the virtual devices in the network topology and work with them:

1. Select **Topologies > Access Active Topologies**.
2. Click the **Virtual Machines** tab to view the IP address and connection information for the Console port or management Ethernet of the virtual device.



NOTE: This information can be copied and pasted into an Excel spreadsheet.

3. Click **Join** in the Active Topology section at the top of the screen.

The Junosphere Access Portal appears.

4. Enter your username and password and click **Sign In**.
5. Click the Network Connect **Start** button.

Network Connect establishes a secure access SSL VPN to the internal management Ethernet of the topology. Traffic will be directed only to the local management Ethernet over that tunnel. Network Connect works best without web proxies, but works fine with a static proxy configuration. It will not work, however, if the browser is configured with a PAC (proxy auto-configuration) file.



NOTE: You need to have administration rights on your computer to install the Network Connect software.

You cannot run two simultaneous instances of Network Connect. If you already have one instance of Network Connect running for a corporate VPN, you must stop it before you begin another instance.

6. Connect to a virtual device using an appropriate communications program. For example:
 - For VJX, use telnet or CMD shell.
 - For CentOS, use VNC. Enter the IP address (with :1 at the end) of the virtual device; enter **Clouds** as the password.
 - For Junos Space or Mu Studio, open a browser from a virtual CentOS in Junosphere and browse to the IP address.

For more information, refer to the manual for each VM type.

7. Log in to the Junos OS virtual device using the default username and password. The default root password is **Clouds**.
8. If you are connecting to a Junos OS network device, enter **cli** to start using the Junos OS environment.

Just as with a physical Junos OS device, configuration changes are made in edit mode and then committed to implement the changes on the router. To save your changes to the library, click **Save** on the Junosphere Libraries menu.

Related Documentation

- *Loading a Configuration from a File*
- *Examples: Loading a Configuration from a File*
- [Connecting to the Network Topology on page 44](#)

PART 2

Index

- [Index on page 49](#)

Index

Symbols

#, comments in configuration statements.....	x
(), in syntax descriptions.....	x
< >, in syntax descriptions.....	x
[], in configuration statements.....	x
{ }, in configuration statements.....	x
(pipe), in syntax descriptions.....	x

B

braces, in configuration statements.....	x
brackets	
angle, in syntax descriptions.....	x
square, in configuration statements.....	x

C

CentOS servers	
configuring.....	40
comments, in configuration statements.....	x
config file	
sample.....	37
config files	
rules.....	39
configuration file	
creating.....	9
curly braces, in configuration statements.....	x

D

documentation	
comments on.....	x

I

ipaddr option	
assigning an IP address.....	35

J

Junosphere Access Portal.....	44
-------------------------------	----

M

macros.....	25
manuals	
comments on.....	x

N

network devices	
configuring.....	39
network topology	
connecting to.....	44
testing.....	43

P

parentheses, in syntax descriptions.....	x
parsing	
advanced capabilities.....	32
concepts.....	32
syntax rules.....	33

T

topologies	
.vmm file syntax.....	23
advanced.....	16
building with vmm scripting.....	21
configuration example.....	37
creating.....	3
ecosystem partner.....	15
file sets.....	22
in public libraries.....	15, 18
modifying.....	3
starter.....	15
starting from Topology Wizard.....	10
topology	
auto-save feature.....	9
copying.....	13
editing.....	13
topology file sets	
building.....	44
Topology Wizard	
creating topologies with.....	5, 7
making reservation from.....	11
starting topology from.....	10
toolbox icons.....	5

V

virtual machine	
creating the configuration.....	39
virtual machines	
configuring.....	40
interface options.....	36
vmm scripting	
creating topologies with.....	3

