



---

# Junosphere

## Network Topology Guide

Release

2.7



---

Published: 2013-02-07

Juniper Networks, Inc.  
1194 North Mathilda Avenue  
Sunnyvale, California 94089  
USA  
408-745-2000  
[www.juniper.net](http://www.juniper.net)

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

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.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

*Junosphere Network Topology Guide*  
Copyright © 2013, Juniper Networks, Inc.  
All rights reserved.

Revision History  
2012-01-27—R1  
2012-03-05—R2  
2012-04-04—R3  
2012-07-25—R4  
2012-10-12—R5  
2012-11-29—R6  
2013-02-08—R7

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

	<b>About the Documentation . . . . .</b>	<b>ix</b>
	Documentation and Release Notes . . . . .	ix
	Documentation Conventions . . . . .	ix
	Documentation Feedback . . . . .	xi
	Requesting Technical Support . . . . .	xi
	Self-Help Online Tools and Resources . . . . .	xi
	Opening a Case with JTAC . . . . .	xii
<b>Part 1</b>	<b>Building and Using Junosphere Network Topologies</b>	
<b>Chapter 1</b>	<b>Junosphere Network Topologies Overview . . . . .</b>	<b>3</b>
	Overview . . . . .	3
<b>Chapter 2</b>	<b>Creating and Editing Topologies . . . . .</b>	<b>5</b>
	Creating Topologies with the Topology Wizard . . . . .	5
	Starting a Topology from the Topology Wizard . . . . .	9
	Editing a Topology . . . . .	11
	Copying a Topology . . . . .	13
<b>Chapter 3</b>	<b>Using Topologies in Public Libraries . . . . .</b>	<b>15</b>
	Topologies in Public Libraries Overview . . . . .	15
	Using Topologies in Public Libraries . . . . .	19
<b>Chapter 4</b>	<b>Building Topologies with VMM Scripting . . . . .</b>	<b>21</b>
	The topology.vmm File . . . . .	21
	.vmm File Syntax . . . . .	22
	Using Macros . . . . .	24
	VJX . . . . .	24
	BGP Service . . . . .	25
	Junos Space . . . . .	25
	CaridenMATE . . . . .	26
	CentOS Server . . . . .	27
	Mu Studio Performance . . . . .	27
	Mu Test Engine . . . . .	28
	Packet Design Route Explorer . . . . .	29
	Spirent TestCenter . . . . .	30
	VPTX . . . . .	31
	VSRX . . . . .	32
	VMX . . . . .	33
	WANDL . . . . .	34
	Advanced Parsing Capabilities . . . . .	34
	Parsing Concepts . . . . .	35

	Parser Syntax Rules . . . . .	35
<b>Chapter 5</b>	<b>Definition of Options in the VMM File . . . . .</b>	<b>37</b>
	Using the ipaddr Interface Option . . . . .	37
	VJX Virtual Machines Interface Options . . . . .	38
	Topology Configuration Example . . . . .	39
	Sample .vmm Configuration File . . . . .	39
	Creating the Virtual Machine Configuration . . . . .	40
	Configuring Junos OS Network Devices . . . . .	41
	Rules for Junos OS Configuration Files . . . . .	41
	Configuring Other Virtual Machines . . . . .	41
	Configuring CentOS Servers . . . . .	41
	Default Configuration File . . . . .	43
	Building and Testing Your Network Topology . . . . .	44
	Building a Topology File set . . . . .	45
	Connecting to the Network Topology . . . . .	45
<b>Part 2</b>	<b>Index</b>	
	Index . . . . .	49

# List of Figures

<b>Part 1</b>	<b>Building and Using Junosphere Network Topologies</b>	
<b>Chapter 2</b>	<b>Creating and Editing Topologies</b>	<b>5</b>
	Figure 1: Topology Wizard Interface	6
	Figure 2: Add Topology Dialog Box	7
	Figure 3: Image Information Bubble	8
	Figure 4: Edit Configuration Dialog Box	9
	Figure 5: Topology Saved Dialog Box—Sandbox	10
	Figure 6: Topology Saved Dialog Box—Bank	10
	Figure 7: Edit Topology Dialog Box	12
	Figure 8: Edit Topology Dialog Box for VMM Scripting	13
<b>Chapter 4</b>	<b>Building Topologies with VMM Scripting</b>	<b>21</b>
	Figure 9: Topology file set Structure	22
	Figure 10: Example topology.vmm File	23
<b>Chapter 5</b>	<b>Definition of Options in the VMM File</b>	<b>37</b>
	Figure 11: A Simple topology.vmm Configuration	39



# List of Tables

	<b>About the Documentation . . . . .</b>	<b>ix</b>
	Table 1: Notice Icons . . . . .	ix
	Table 2: Text and Syntax Conventions . . . . .	x
<b>Part 1</b>	<b>Building and Using Junosphere Network Topologies</b>	
<b>Chapter 2</b>	<b>Creating and Editing Topologies . . . . .</b>	<b>5</b>
	Table 3: Toolbox Icons . . . . .	6
<b>Chapter 3</b>	<b>Using Topologies in Public Libraries . . . . .</b>	<b>15</b>
	Table 4: Starter Topologies . . . . .	15
	Table 5: Ecosystem Partner Topologies . . . . .	16
	Table 6: Advanced Topologies . . . . .	18
<b>Chapter 4</b>	<b>Building Topologies with VMM Scripting . . . . .</b>	<b>21</b>
	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 xi
- Requesting Technical Support on page xi

## Documentation and Release Notes

---

For disclosure information on Junosphere Connector, refer to the files located at <http://www.juniper.net/support/products/junosphereconnector>.

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 x defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
<b>Bold text like this</b>	Represents text that you type.	To enter configuration mode, type the <b>configure</b> command:  user@host> <b>configure</b>
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> <b>show chassis alarms</b>  No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"> <li>Introduces important new terms.</li> <li>Identifies book names.</li> <li>Identifies RFC and Internet draft titles.</li> </ul>	<ul style="list-style-type: none"> <li>A policy <i>term</i> is a named structure that defines match conditions and actions.</li> <li><i>Junos OS System Basics Configuration Guide</i></li> <li>RFC 1997, <i>BGP Communities Attribute</i></li> </ul>
<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@# <b>set system domain-name</b> <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"> <li>To configure a stub area, include the <b>stub</b> statement at the [edit protocols ospf area area-id] hierarchy level.</li> <li>The console port is labeled <b>CONSOLE</b>.</li> </ul>
< > (angle brackets)	Enclose optional keywords or variables.	<b>stub</b> <default-metric <i>metric</i> >;
(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.	<b>broadcast</b>   <b>multicast</b>  ( <i>string1</i>   <i>string2</i>   <i>string3</i> )
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	<b>rsvp { # Required for dynamic MPLS only</b>
[ ] (square brackets)	Enclose a variable for which you can substitute one or more values.	<b>community name members</b> [ <i>community-ids</i> ]
Indentation and braces ( { } )	Identify a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; 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 send your comments to [techpubs-comments@juniper.net](mailto:techpubs-comments@juniper.net), or fill out the documentation feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable)

## Requesting Technical Support

---

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

## 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/>

## Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/> .
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <http://www.juniper.net/support/requesting-support.html> .

To open a case with JTAC for Junosphere, you must provide the bank serial number. To find the serial number, double-click the bank icon in Junosphere.

## PART 1

# Building and Using Junosphere Network Topologies

- [Junosphere 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 37](#)



## CHAPTER 1

# Junosphere 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

- Creating Topologies with the Topology Wizard on page 5
- Starting a Topology from the Topology Wizard on page 9
- Editing a Topology on page 11
- Copying a Topology on page 13

### Creating Topologies with the Topology Wizard

---



**NOTE:** When you access the Topology Wizard, you will get a message telling you to enable downloads from junosphere.net. To prevent this message from appearing again, click the Do not show again check box.

The Topology Wizard (see [Figure 1 on page 6](#)) 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

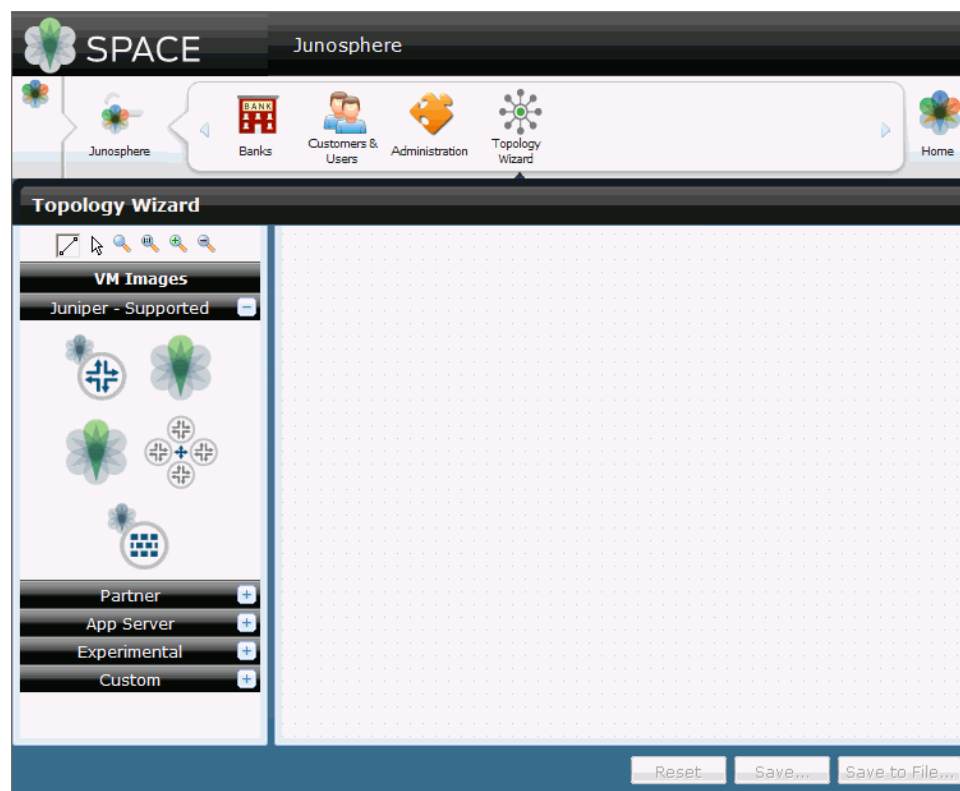








Table 3 on page 6 lists and describes the toolbox icons in the Topology Wizard.

Table 3: Toolbox Icons

Icon	Description
	Enables you to create a connection between elements on the canvas.
	Enables you to select an element on the canvas.
	Enables you to fit the topology to the canvas.
	Enables you to view the topology in its actual size.
	Enables you to zoom in on a part of the topology.
	Enables you to zoom out.

To use the Topology Wizard to create a new topology:

1. On the main menu, click the **Topology Wizard** icon. The Topology Wizard appears, with the **Add Topology** dialog box (see [Figure 2 on page 7](#)) displayed.

**Figure 2: Add Topology Dialog Box**

**Add Topology**

Name:

Description:

Downloadable: ☒

Editable: ☒

Select one or more libraries in which you want this topology to be added...

Bank	Sandbox	Library
<input type="checkbox"/> user_Bank	Default Sandbox - LAB	1vm-topology
<input type="checkbox"/> user_Bank	Default Sandbox - LAB	2VIX Centos
<input type="checkbox"/> user_Bank		Bank Library
<input type="checkbox"/> user_Bank		New Library

Libraries:

Add Cancel

2. Enter the name of the topology and a description, if desired.
3. Indicate whether the topology is to be downloadable, editable, or both.
4. Select the library to which you want to save the topology.
5. Click **Add**.

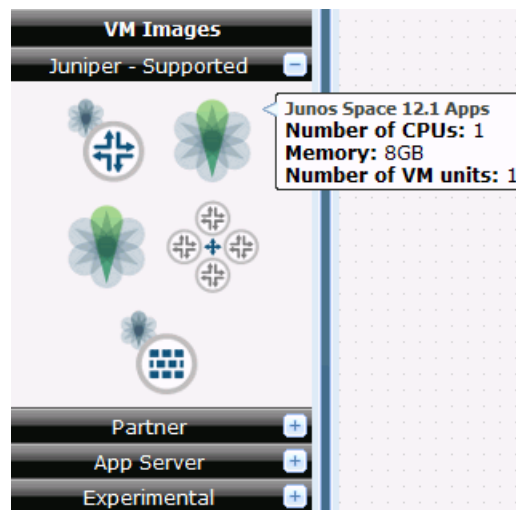
A new window opens in Junosphere with icons and tools on the left side see( [Figure 1 on page 6](#)).



**NOTE:** If you click **Cancel** without saving the file, you can still draw a topology and click **Save to File** to save the topology on your system for future use.

6. To see the different types of VM images, click the plus (+) sign for each type.
7. To see additional information about each image, place your mouse pointer on the image until an informational box, similar to that shown in [Figure 3 on page 8](#), appears.

Figure 3: Image Information Bubble



8. 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 drop the icon.
9. To create connections, click the **Connector** icon and hover the cursor over one of the objects that you want to connect until a hand icon appears.



**NOTE:** Do not connect the following four 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
- Cariden Mate

10. Enable “T” connections by creating a connection between the test or monitoring VM (such as Spirent, RouteExplorer, or Mu TestEngine) 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.
11. Click **Save** (if you clicked **Add**) or **Save to File** (if you clicked **Cancel** or you want to keep a copy of the topology for reference or for future use).

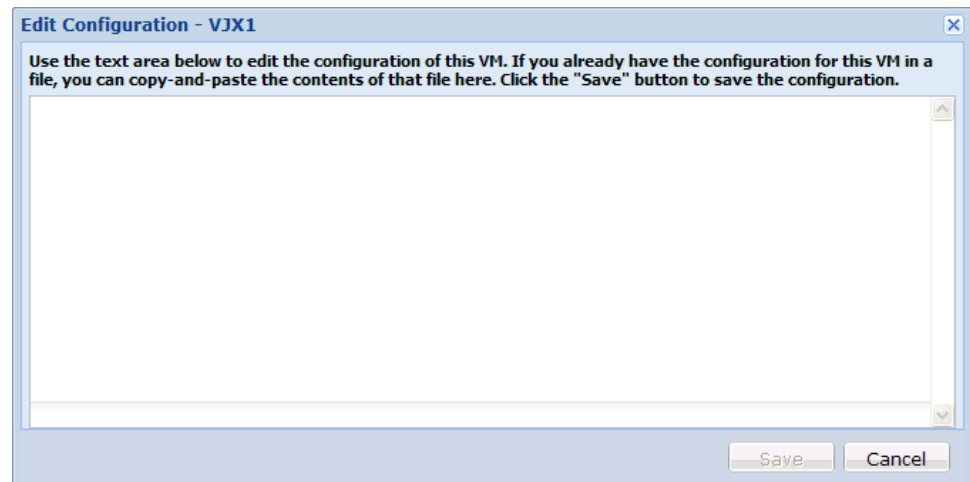


**NOTE:** In addition to saving the topology to a sandbox or file, the Topology Wizard has an auto-save feature that saves every five changes made to the topology diagram. These changes can include new VMs added, new links added, VMs removed, links removed, VMs moved, and other changes. The topology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

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 9](#).

12. 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. The following dialog box appears.

Figure 4: Edit Configuration Dialog Box



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



**NOTE:** When creating a topology, the Maximum Transmission Unit (MTU) in the VJX interface must be set to 1200. Otherwise, it cannot route BGP packets.

#### Related Documentation

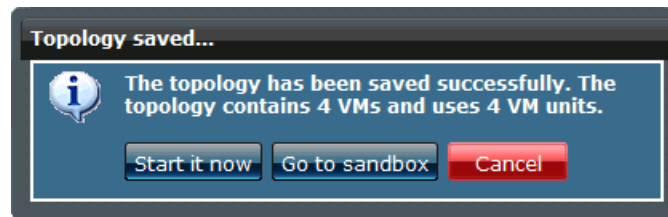
- [Starting a Topology from the Topology Wizard on page 9](#)
- [Editing a Topology on page 11](#)

## 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.

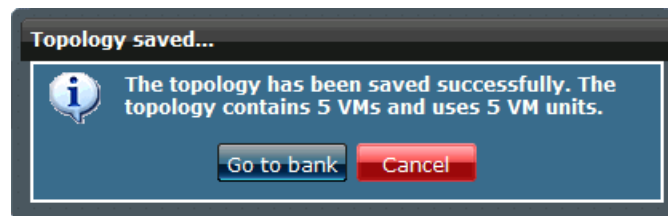
Once you have created a topology and click **Save** or **Save to File**, the following dialog box appears enabling you to start the topology now or to reserve time for future use.

Figure 5: Topology Saved Dialog Box—Sandbox



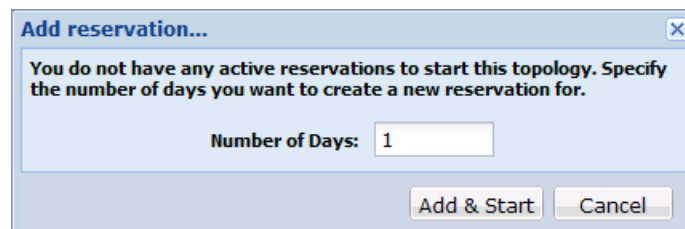
**NOTE:** If you select **Go to sandbox**, you are taken to the sandbox to launch the topology. If you saved the topology to a bank, the following dialog box appears, enabling you to go to the bank. Once the topology has been saved to the bank, it is visible to all sandboxes in that bank and can be copied to a sandbox and run. Remember that a topology can be started only in a sandbox, not in a bank.

Figure 6: Topology Saved Dialog Box—Bank



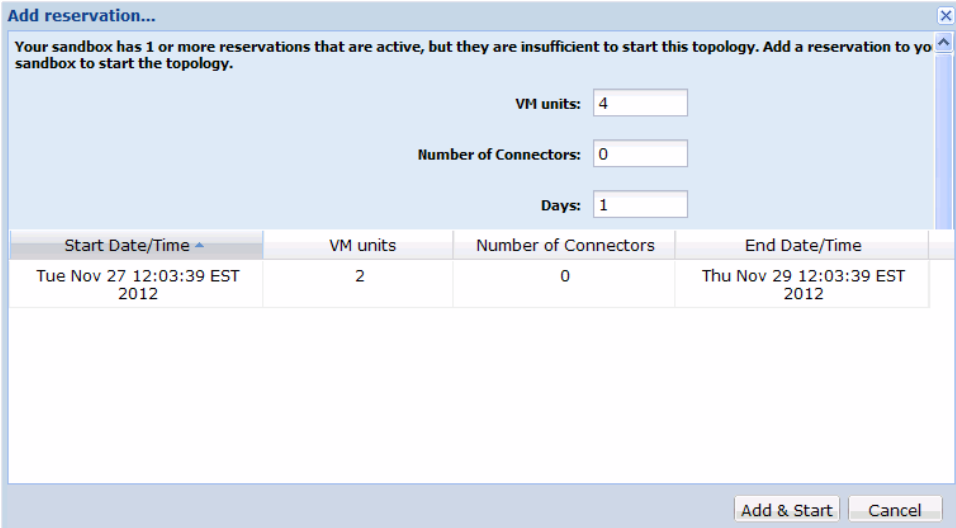
To launch the topology from the Topology Wizard:

1. Select **Start it now**. A dialog box (Figure 5 on page 10) appears that enables you to launch your topology directly from the Topology Wizard.
2. If there are no active reservations in your sandbox, the following dialog box appears, enabling you to create a reservation from the Topology Wizard.



- a. In the Add Reservation dialog box, enter the number of days you want for the reservation.
- b. 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 your 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.

VM units:

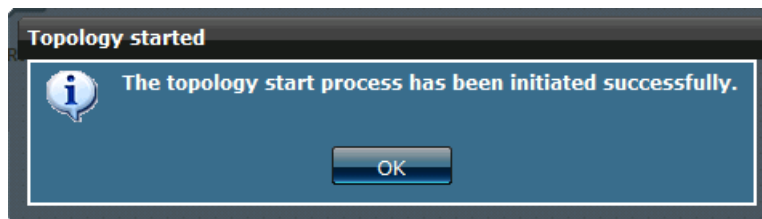
Number of Connectors:

Days:

Start Date/Time	VM units	Number of Connectors	End Date/Time
Tue Nov 27 12:03:39 EST 2012	2	0	Thu Nov 29 12:03:39 EST 2012

- 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 **Add & Start**.

The following dialog box appears.



- Click **OK**. The Active Topology Details tab appears.

#### Related Documentation

- [Creating Topologies with the Topology Wizard on page 5](#)
- [Editing a Topology on page 11](#)
- [Copying a Topology on page 13](#)

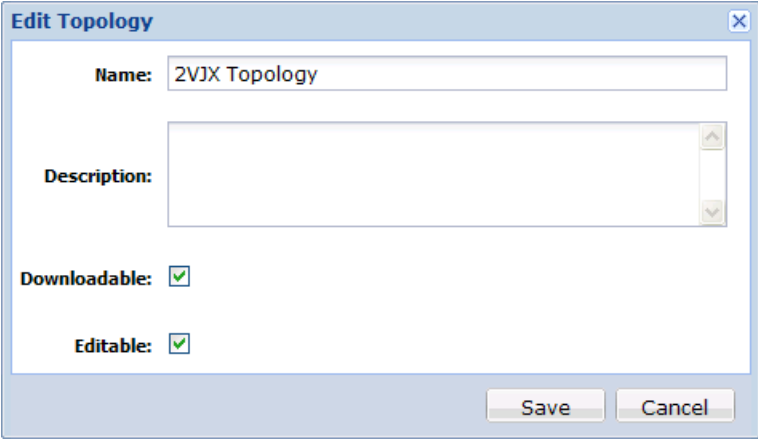
## Editing a Topology

You can modify any topology by editing its VMM file. You can also edit a topology that was created with the Topology Wizard within the Topology Wizard GUI.

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

1. Go to the topology in the **Libraries** tab.
2. Right-click the topology and select **Edit** from the menu. The following dialog box appears.

**Figure 7: Edit Topology Dialog Box**



The screenshot shows a standard Windows-style dialog box titled "Edit Topology". Inside the dialog, there is a "Name:" label followed by a text box containing "2VJX Topology". Below that is a "Description:" label followed by a larger, empty text box with vertical scrollbars. Further down are two checkboxes: "Downloadable:" and "Editable:", both of which are checked with green checkmarks. At the bottom right of the dialog are two buttons: "Save" and "Cancel".

3. Add information or change the existing information.  
For example, use the Description field to enter a description about the topology.
4. Click **Save**.  
The topology file opens in the Topology Wizard, enabling you to make graphical changes to the topology.

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 .vmm file in its place and edit the properties. To edit a topology that was created with scripting:

1. Go to the topology in the **Libraries** tab.
2. Right-click the topology and select **Edit** from the menu. The following dialog box appears.



Figure 8: Edit Topology Dialog Box for VMM Scripting

The screenshot shows the 'Edit Topology' dialog box. The 'Name' field is filled with '2VJX Topology'. The 'Description' field is an empty text area. The 'Downloadable' and 'Editable' checkboxes are both checked. The 'Fileset' field contains the text 'Select the fileset to upload...' and a 'Browse...' button. At the bottom of the dialog are 'Save' and 'Cancel' buttons.

3. Click the **Browse** button next to the File set field and browse to the .vmm file.
4. In the **Edit Topology** dialog box, add information or change the existing information.  
For example, use the Description field to enter a description about the topology.
5. Click **Save**.  
The .vmm file is uploaded.

**Related  
Documentation**

- [Creating Topologies with the Topology Wizard on page 5](#)
- [Copying a Topology on page 13](#)

## Copying a Topology

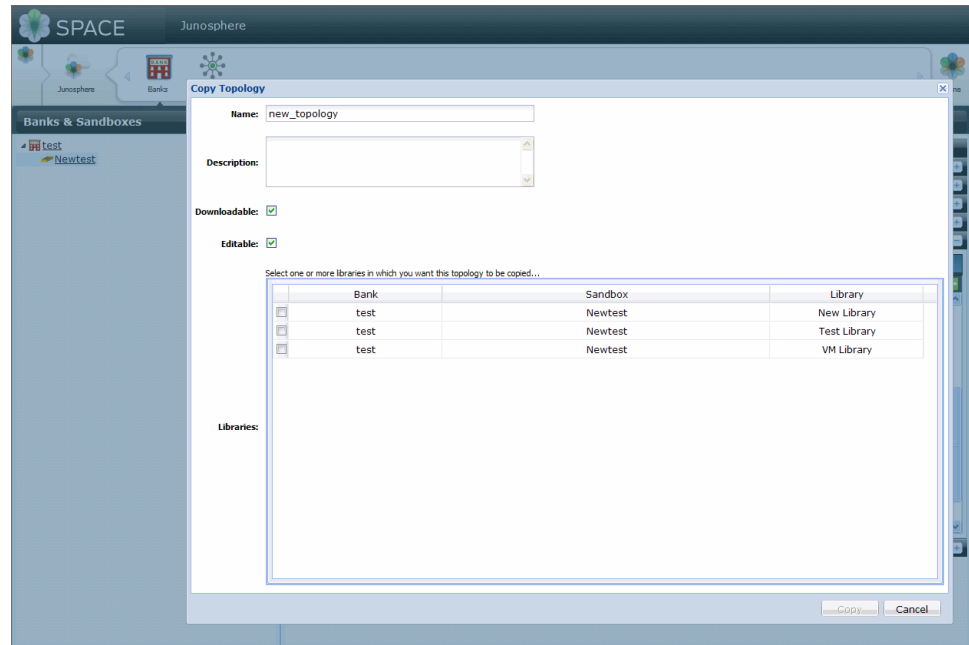
Junosphere's Copy feature enables users and bank administrators to copy a topology from one sandbox in a bank to other sandboxes in the same bank.

- 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 where the user has Library Management permission.

To copy a topology:

1. Navigate to the topology you want on the Libraries accordion tab.
2. Right-click on the topology and select **Copy** from the pop-up menu.

The following dialog box appears, listing all libraries in that sandbox.



3. Make any changes to the topology information, including name, description, downloadability, and editability.
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 5](#)
  - [Editing a Topology on page 11](#)

## CHAPTER 3

# Using Topologies in Public Libraries

- [Topologies in Public Libraries Overview on page 15](#)
- [Using Topologies in Public Libraries on page 19](#)

### Topologies in Public Libraries Overview

---

There are three different topology libraries:

1. Starter topologies, which provide simple, basic topology examples. [Table 4 on page 15](#) lists the current examples.

**Table 4: Starter Topologies**

Topology	Description	Diagram
1VJX Topology	Uses one VM. This is the most basic topology. It contains one VJX router with 16 ports. This topology enables you to familiarize yourself with Junosphere and to practice Junos OS CLI commands.	Not applicable.

---

Table 4: Starter Topologies (*continued*)

Topology	Description	Diagram
2VJX Topology	This topology uses two VMs and contains two VJX routers — vjx0 and vjx1 — that are connected to each other through port em1. The file set also contains diagrams and detailed explanations.	<p>Diagram showing two VJX routers connected via their em1 ports to a central private1 network. Each router has em0, em2, and em3 ports. A CentOS VM is connected to the left router's em0 port. The diagram is labeled g017840.</p>
2VJX_Space+Centos Topology	This topology is a variation of the 2VJX topology, but uses four VMs and includes an instance of Space for network management.  For information on launching Space in Junosphere, refer to the help files.	<p>Diagram showing two VJX routers connected via their em1 ports to a central private1 network. Each router has em0, em2, and em3 ports. A CentOS VM is connected to the left router's em0 port, and a Junos Space instance is connected to the right router's em0 port. The diagram is labeled g017839.</p>
2VJX-GRE_Centos Topology	This topology is a variation of the 2VJX topology, but uses three VMs. It contains two VJX routers — vjx0 and vjx1 — with 16 ports each and a Centos (Linux) server. Ports em1 and em8 in each router are connected to each other in a 1+1 configuration.	<p>Diagram showing two VJX routers connected via their em1 ports to a central private0 network. Each router has em0, em8, and em3 ports. A CentOS VM is connected to the left router's em0 port. The diagram is labeled g017838.</p>

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

Table 5: Ecosystem Partner Topologies

Topology	Description
Cariden-Mate	This topology uses one VM and runs Cariden Mate. Use this library to design new topologies and run simulations before loading the topologies into Junosphere.

Table 5: Ecosystem Partner Topologies (*continued*)

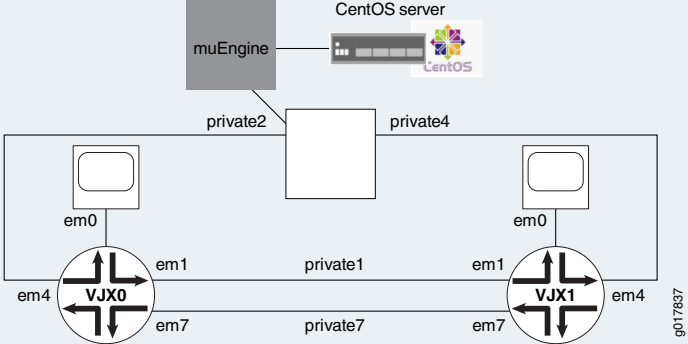
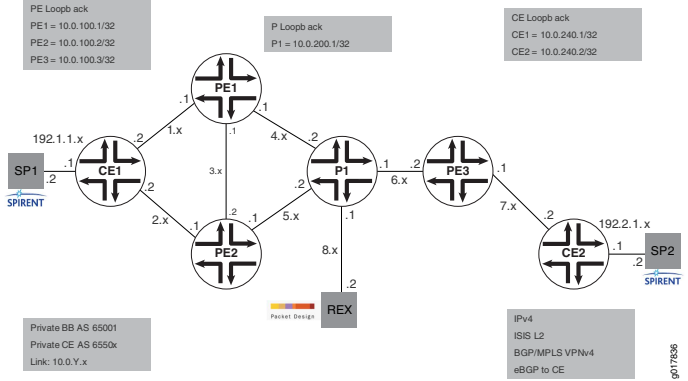
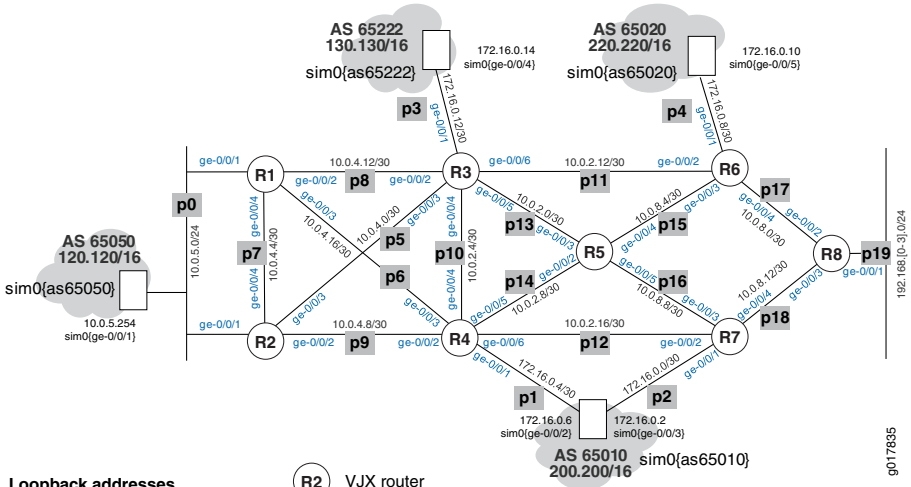
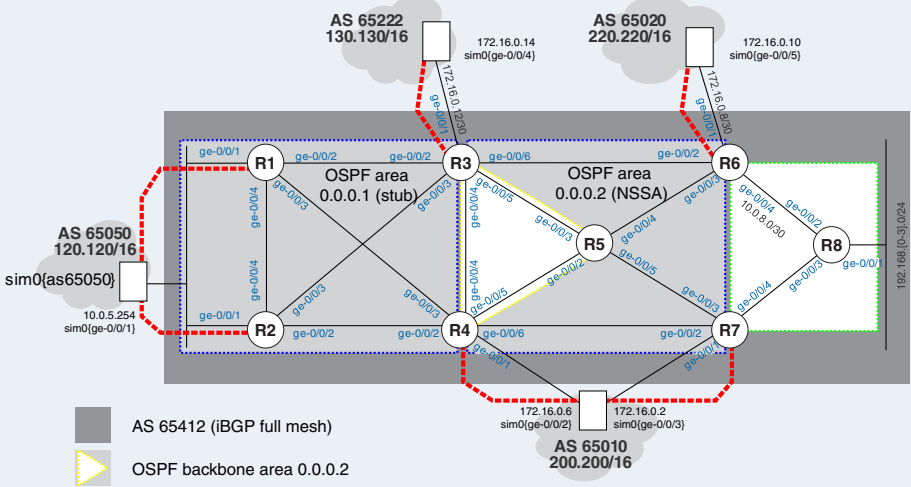
Topology	Description
Mu Dynamics+2VJX+Centos	<div></div> <p>This topology uses five VMs and illustrates the application of the Mu Dynamics analysis tool. This topology also contains two VJX routers and a Centos server that facilitates access to Mu Dynamics from within Junosphere.</p>
Spirent+Packet_Design_L3VPN	<div><p>This topology uses nine VMs and is a Layer 3 VPN lab with six routers, two Spirent TestCenters, and one Packet Design Route Explorer (REX). The Spirent TestCenter generates Layer 2/3 test signals and analyzes performance. The Packet Design REX identifies nodes in the network and produces a graphical representation of the network.</p></div> <div></div> <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. <a href="#">Table 6 on page 18</a> lists the examples.</p>

Table 6: Advanced Topologies

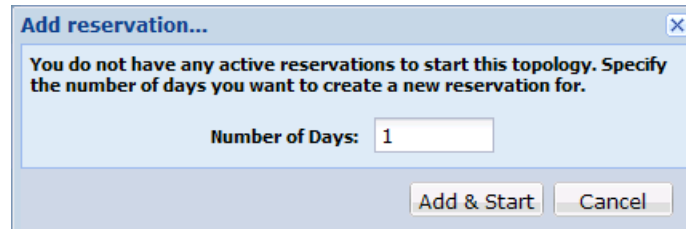
Topology	Description
Demo Baseline Topology	<p>This topology uses nine VMs and consists of eight VJX routers and an additional router connected to routers 1 through 7 that acts as a simulator. The topology is divided into four autonomous systems (ASs).</p>  <p><b>Loopback addresses</b> 10.0.255.x, x = router number</p> <p><b>Interface numbering</b> Lower number on the left If vertical, then top is lower number</p> <p><b>Legend:</b></p> <ul style="list-style-type: none"> <li><span style="border: 1px solid black; border-radius: 50%; padding: 2px;">R2</span> VJX router</li> <li><span style="border: 1px solid black; padding: 2px;">p17</span> Junosphere private bridge (num)</li> <li><span style="border: 1px solid black; width: 20px; height: 10px; display: inline-block;"></span> Simulated router (sim0 instance) routing-instance = {asnumber}</li> </ul>
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><b>Legend:</b></p> <ul style="list-style-type: none"> <li><span style="background-color: #cccccc; border: 1px solid black; display: inline-block; width: 15px; height: 10px;"></span> AS 65412 (iBGP full mesh)</li> <li><span style="border: 2px dashed yellow; display: inline-block; width: 15px; height: 10px;"></span> OSPF backbone area 0.0.0.2</li> <li><span style="border: 2px dashed blue; display: inline-block; width: 15px; height: 10px;"></span> OSPF area</li> <li><span style="border: 2px dashed green; display: inline-block; width: 15px; height: 10px;"></span> IS-IS Level 1 (Area 002)</li> <li><span style="color: red;">---</span> eBGP peer</li> </ul>

Topology	Description
----------	-------------

g017833



**NOTE:** If you do not have an existing reservation for the topology you want to start, the following dialog box appears after you select Start Topology from the pop-up menu.



Enter the number of days for the reservation and click Add & Start.



## 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 22](#)
- [Using Macros on page 24](#)
- [Advanced Parsing Capabilities on page 34](#)
- [Parsing Concepts on page 35](#)
- [Parser Syntax Rules on page 35](#)

### The topology.vmm File

---

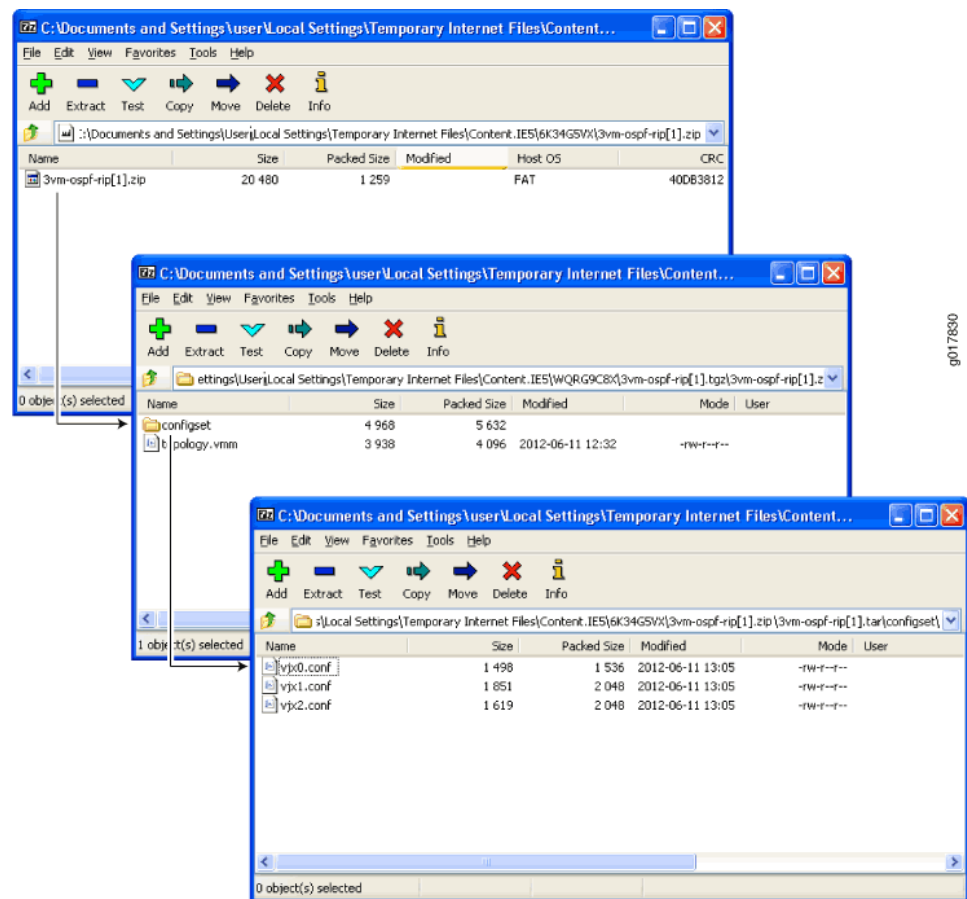
After understanding the customer's requirements, purpose, and goals for the network, you can define the virtual network topology using Virtual Machine Manager (VMM) language syntax.

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 or personality of the virtual machine. (Refer to the *Junosphere Release Notes 2.6* for a complete list of supported images.)
  - Virtual machine interfaces to map.
  - A link to a configuration file (optional).
- The connections between the 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 9 on page 22](#):

Figure 9: Topology file set Structure

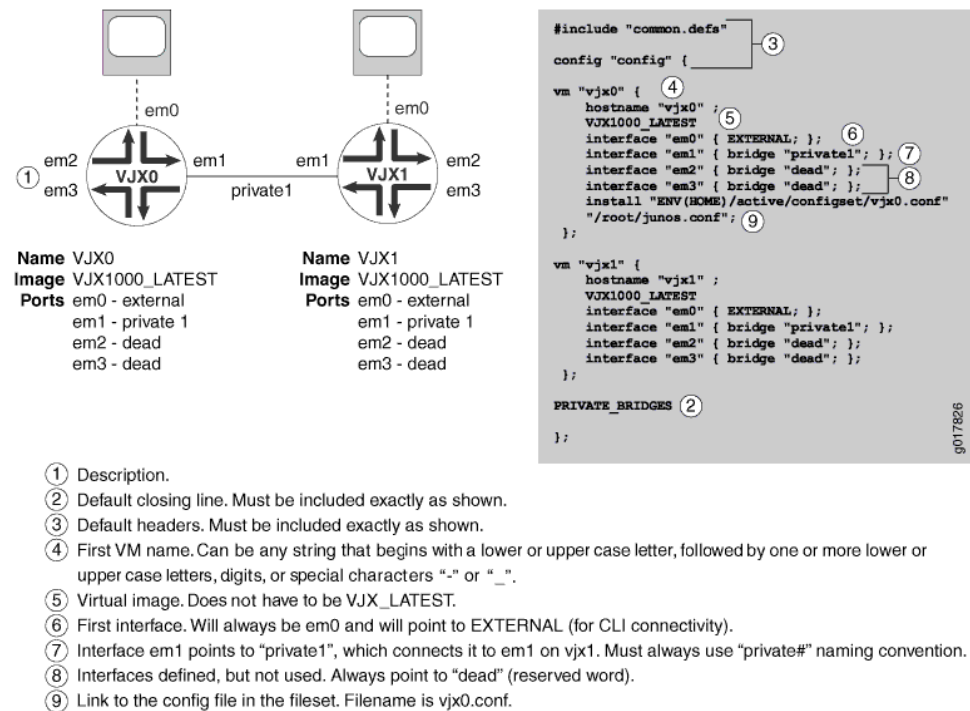


**NOTE:** To create the `topology.vmm` file, use any text editor. To integrate or to open files in your file set, use 7zip (for tar/gzip/tgz compression) or any Windows-compatible zip or compression tool (for zip compression).

## .vmm File Syntax

Consider the example of a `topology.vmm` file shown in [Figure 10](#) on page 23:

Figure 10: Example topology.vmm File



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 bootup 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 + };
vm_statement	install string_value string_value;
vm_statement	interface definition;
interface_definition	interface name_value { interface_statement + };

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

Configuration	Configuration Value
interface_statement	<b>bridge</b> <i>name_value</i> ;
interface_statement	<b>macaddr</b> <i>string_value</i> ;
interface_statement	<b>ipaddr</b> <i>string_value</i> ;
name_value	<i>string_value</i>

## Using Macros

This section lists and describes the macros that are available for the public topology files. Most macros are allowed for all banks; if a macro is not allowed for all banks, it must be configured.

### VJX

**Name:** VJX1000\_LATEST, VJX\_LATEST

**Category:** Juniper-supported

**VM Units:** 1

**RAM:** 2 GB

**CPU:** 1

**Permission:** All

**Description:** J-Series Router based on Junos12.1 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
};

```

## BGP Service

**Name:** BGP\_SERVICE

**Category:** Juniper-supported

**VM Units:** 3

**RAM:** 2 GB

**CPU:** 1

**Permission:** All

**Description:** Latest qualified BGP Service image. This is a special configuration of VJX that is used for BGP Service, based on the latest Junos OS release.

**Usage:**

```

#include "common.defs"

config "config" {

  vm "BGPO"{
    hostname "BGPO";
    BGP_SERVICE
  };

  PRIVATE_BRIDGES

};

```

## Junos Space

**Name:** SPACE\_LATEST

**Category:** Juniper-supported

**VM Units:** 1

**RAM:** 8 GB

**CPU:** 1

**Permission:** All

**Description:** Latest qualified Junos Space image. This currently maps to SPACE\_12\_2R1\_3.

**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

};
```

## CaridenMATE

**Name:** CaridenMATE\_LATEST

**Category:** Partner

**VM Units:** 1

**RAM:** 2 GB

**CPU:** 1

**Permission:** All

**Description:** Latest qualified Cariden MATE image. This currently maps to Cariden MATE 4.5.1.

**Usage:**

```
#include "common.defs"

config "config" {

    vm "MTE0"{
```

```

hostname "MTE0";
CaridenMATE_LATEST
};

PRIVATE_BRIDGES

};

```

## CentOS Server

**Name:** CENTOS\_5\_3, CENTOS64\_5\_4

**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_5_3
    interface "em0" { EXTERNAL; };

};

vm "cent01" {
    hostname "cent01";

    CENTOS64_5_4
    interface "em0" { EXTERNAL };
};

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. This currently maps to Mu Studio 6.1.

**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. This currently maps to Mu Studio 6.1.

**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

};

```

## Packet Design Route Explorer

**Name:** REX\_LATEST

**Category:** Partner

**VM Units:** 1

**RAM:** 2 GB

**CPU:** 1

**Permission:** All

**Description:** Latest qualified Virtual Route Explorer image This currently maps to REX\_9\_4.

**Usage:**

```

#include "common.defs"

config "config" {

vm "REX0"{
hostname "REX0";
REX_LATEST
REX1_MGMT

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

install "ENV(HOME)/active/configset/REX0.conf" "/system.conf";
};

vm "REX1"{
hostname "REX1";
REX_LATEST

```

## REX2\_MGMT

```
interface "em1" { bridge "private1"; };  
interface "em2" { bridge "private2"; };
```

```
install "ENV(HOME)/active/configset/REX1.conf" "/system.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 "private1"; }; /* ge-0/0/2 */
```

```
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 "private2"; }; /* ge-0/0/1 */  
interface "em2" { bridge "private3"; }; /* ge-0/0/2 */
```

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

```
vm "REX2"{  
  hostname "REX2";  
  REX_LATEST  
  REX3_MGMT
```

```
interface "em1" { bridge "private3"; };
```

```
install "ENV(HOME)/active/configset/REX2.conf" "/system.conf";  
};
```

## PRIVATE\_BRIDGES

```
};
```

## Spirent TestCenter

**Name:** SPIRENT\_LATEST

**Category:** Partner

**VM Units:** 1

**RAM:** 2 GB

**CPU:** 1

**Permission:** All

**Description:** Latest qualified Virtual Spirent TestCenter image. This currently maps to SPIRENT\_3\_90. You must upgrade your application to Virtual Spirent TestCenter 3.90.

**Usage:**

```
#include "common.defs"

config "config" {

  vm "spirent1" {
    hostname "spirent1";
    SPIRENT_LATEST
    interface "em0" { EXTERNAL; };
    interface "em1" { bridge "private2"; };
  };

  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:** 2

**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

};

```

## 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

};
```

## WANDL

**Name:**WANDL\_LATEST

**Category:** Partner

**VM Units:** 1

**RAM:** 4 GB

**CPU:** 1

**Permission:** All

**Description:** Latest qualified WANDL image.

**Usage:** Refer to the WANDL documentation for usage example.

---

## 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.

## Parsing Concepts

---

Parsing relies on the following concepts:

- VM (virtual machine) — Includes Juniper devices (such as VJX, Junos Space), partner tools (such as Spirent TestCenter, Cariden Mate) 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.

## 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.





## CHAPTER 5

# Definition of Options in the VMM File

- [Using the ipaddr Interface Option on page 37](#)
- [VJX Virtual Machines Interface Options on page 38](#)
- [Topology Configuration Example on page 39](#)
- [Sample .vmm Configuration File on page 39](#)
- [Creating the Virtual Machine Configuration on page 40](#)
- [Building and Testing Your Network Topology on page 44](#)

### 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; };  
};
```

---

## 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.

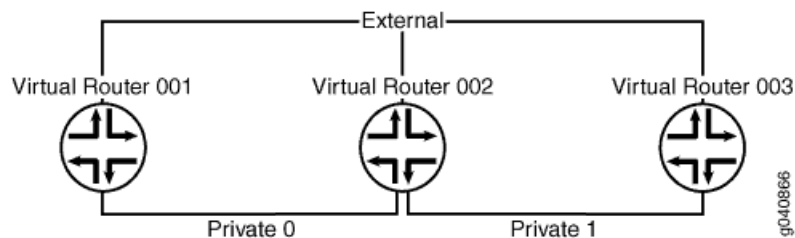
```
interface "<em#>" { bridge "private0"; };
```

- 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.

## Topology Configuration Example

Figure 11 on page 39 defines a simple **topology.vmm** configuration.

Figure 11: A Simple topology.vmm Configuration



The network devices vruter001, vruter002, and vruter003 are virtual machines representing virtual routers. Private0 and Private1 are bridges associated with network segments connecting vruter001 to vruter002 and vruter002 to vruter003, respectively. In order for the virtual machines to be able to communicate with each other, they must share a common bridge.

## 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 vrouter001 and vrouter002
interface "em1" { bridge "private0"; };

// description - configuration file to load on the router
install "ENV(HOME)/active/configset/vrouter001.conf" "/root/junos.conf";
};
vm "vrouter002" {
  hostname "vrouter002" ;

  VJX_LATEST

  interface "em0" { EXTERNAL;};

  // description - link between vrouter002 and vrouter001
  interface "em1" { bridge "private0"; };

  // description - link between vrouter002 and vrouter003
  interface "em2" { bridge "private1"; };

  // description - configuration file to load on the router
  install "ENV(HOME)/active/configset/vrouter002.conf" "/root/junos.conf";
};

vm "vrouter003" {
  hostname "vrouter003" ;

  VJX_LATEST

  interface "em0" { EXTERNAL;};

  // description - link between vrouter003 and vrouter002
  interface "em1" { bridge "private1"; };

  // description - configuration file to load on the router
  install "ENV(HOME)/active/configset/vrouter003.conf" "/root/junos.conf";
};

PRIVATE_BRIDGES

};
```

---

## 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 41](#)
- [Rules for Junos OS Configuration Files on page 41](#)

- [Configuring Other Virtual Machines on page 41](#)
- [Configuring CentOS Servers on page 41](#)
- [Default 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 using 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 Fileset” on page 45](#).

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 45](#).

## 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 43](#).
- 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 Cariden, 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.

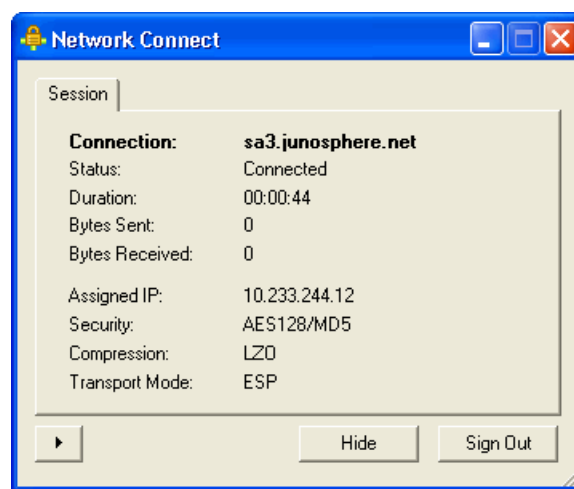
Make sure that you have an active topology and know the IP address. To configure CentOS servers:

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.

Double-click the lock icon. When the Network Connect dialog box appears, click **Sign Out**.





**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 is 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";
            }
        }
    }
}
}
```

---

## 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 45](#)
- [Connecting to the Network Topology on page 45](#)



## 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. On the Active Topology accordion tab, 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.

2. With a topology active, go to the Libraries accordion tab and click **Join**.

The Junosphere Access Portal appears.

3. Enter your username and password and click **Sign In**.
4. 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.

5. 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 Spirent, use the Spirent tool.

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

6. Log in to the Junos OS virtual device using the default username and password. The default root password is **Clouds**.
7. 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.

## 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.....	41
comments, in configuration statements.....	x
config file	
sample.....	39
config files	
rules.....	41
configuration file	
creating.....	9
curly braces, in configuration statements.....	x
customer support.....	xi
contacting JTAC.....	xi

## D

documentation	
comments on.....	xi

## I

ipaddr option	
assigning an IP address.....	37

## J

Junosphere Access Portal.....	45
-------------------------------	----

## M

macros.....	24
-------------	----

manuals	
comments on.....	xi

## N

network devices	
configuring.....	41
network topology	
connecting to.....	45
testing.....	44

## P

parentheses, in syntax descriptions.....	x
parsing	
advanced capabilities.....	34
concepts.....	35
syntax rules.....	35

## S

support, technical See technical support	
--	--

## T

technical support	
contacting JTAC.....	xi
topologies	
.vmm file syntax.....	22
advanced.....	17
building with vmm scripting.....	21
configuration example.....	39
creating.....	3
ecosystem partner.....	16
file sets.....	21
in public libraries.....	15, 19
modifying.....	3
starter.....	15
starting from Topology Wizard.....	9
topology	
auto-save feature.....	8
copying.....	13
editing.....	11
topology file sets	
building.....	45
Topology Wizard	
creating topologies with.....	5
making reservation from.....	10
starting topology from.....	9
toolbox icons.....	6

## V

virtual machine	
creating the configuration.....	40
virtual machines	
configuring.....	41
interface options.....	38
vmm scripting	
creating topologies with.....	3