

Boson NetSim: Can it be used in a classroom?

Can [Boson NetSim](#) be used in a classroom? You bet! You can create the components needed in lab packs so that you can deliver customized labs in the classroom or as homework for students.

One of the nice things about NetSim is the Lab Compiler feature, which allows you to create custom labs and custom lab packs. NetSim allows instructors to deliver customized labs that can be used in the classroom or as homework for students. Let's explore the five steps needed to create a custom lab and lab pack.

The five steps used to create a lab pack are the following:

1. [Create the Lab Document](#) – a required document that describes the steps that should be performed in the lab
2. [Create the Topology File](#) – a required file that defines the devices that will be used in the simulated network in the lab
3. [Create the Loading Configuration Files*](#) – optional running configuration files that can be as detailed or as sparse as you wish
4. [Create the Completed Configuration Files](#) – completed configuration files that contain the commands that the student is required to enter in order to successfully complete the lab
5. [Combine the files with the Lab Compiler](#) – used to combine all the pieces you created in the first four steps and export the compiled lab pack

* The loading configuration files are optional. You do not need to configure loading files. If you choose to not configure the loading files, all devices in the topology will load with the default configurations.

In the following steps, we will create the files used in a lab. You can download sample completed files [here](#).

You will need the following components before your custom lab pack is complete:

1. [Lab Document](#) – describes the steps that should be performed in the lab. The document must be in XPS format for use in NetSim. You can create this document in any format you wish and then “print” it using an XPS printer to get the correct format.
2. [Topology File](#) – defines the devices that will be used in the simulated network in the lab. You can create a custom topology in NetSim using the tools in the Network Designer. When you have completed adding the devices and connections you desire, you should save the topology to a folder you will use for the lab. The file will be saved with a .bsn extension.
3. [Loading Configuration Files*](#) – contains the starting running configuration files that can be as detailed or as sparse as you wish.
4. [Completed Configuration Files](#) – contains the final configurations on the devices after all steps in the lab document have been completed.

Step 1: Create the Lab Document

Everyone knows how to create a document, so I will not describe the steps here other than to give you a couple of things to think about. This document, named TestLab.xps, will end up in the lab pack created here.

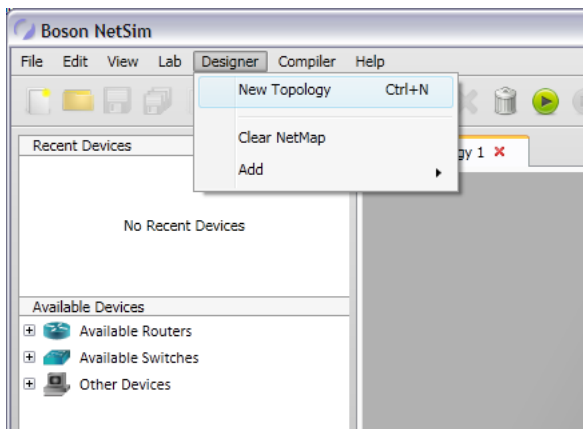
1. Create a draft of the lab document first so that you have an outline of the tasks you want performed. This will also allow you to plan the network topology you want to simulate in the lab.
2. Add a graphic of the topology you created, and label the interfaces appropriately.
3. When appropriate, describe why a step is to be performed.
4. When you show device output in the document, use a different, monospace, font such as `Courier New` so that it lines up correctly and is easier to read.
5. Boson NetSim requires XPS files for lab documents; save the original file as well as the XPS document so that it will be easier for you to make revisions later if needed.

[Return to Top](#)

Step 2: Create the Topology

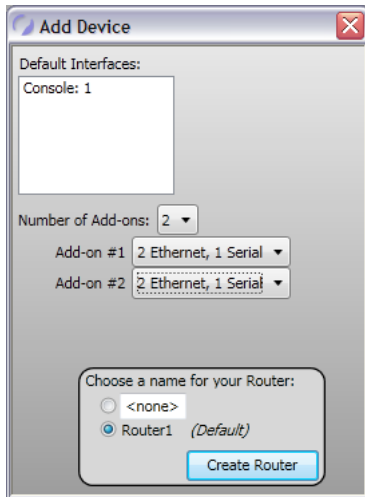
1. Start NetSim, and click **Designer > New Topology**.

In my document, I created a network design in MS Visio and pasted the graphic into the lab document. I will use this design to create the network topology that will be simulated in the lab.



2. On the navigation pane, expand the **Available Routers** list. Click and drag the 3640 router to the NetMap canvas.

3. In the **Add Device** dialog box, select **2** in the **Number of Add-ons** field.

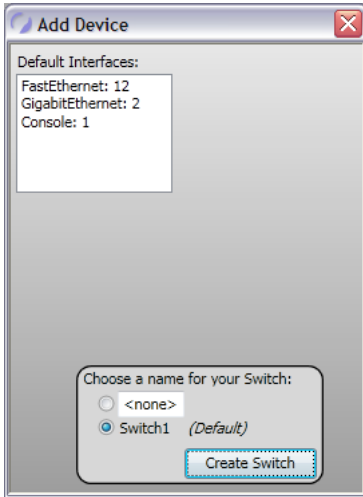


4. In the **Add Device** dialog box, select **2 Ethernet, 1 Serial** as the connection types in the **Add-on #1** and **Add-on #2** options.

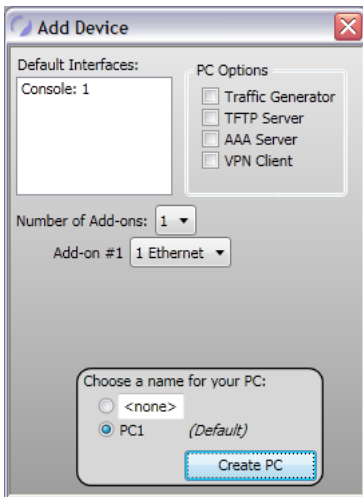
We won't use all of these connections since we are only making a simple topology.

5. Choose a name for the router; you can use the default name or type a name in the text box. You can rename a device if desired by clicking the name in the NetMap and modifying it.
6. Click **Create Router**.
7. Repeat steps 3 through 6 to create a second router, **Router2**.
8. On the navigation pane, expand the **Available Switches** list. Click and drag the 3550 switch to the NetMap canvas.

- Click **Create Switch** to create **Switch1**, which is the default name for the device.



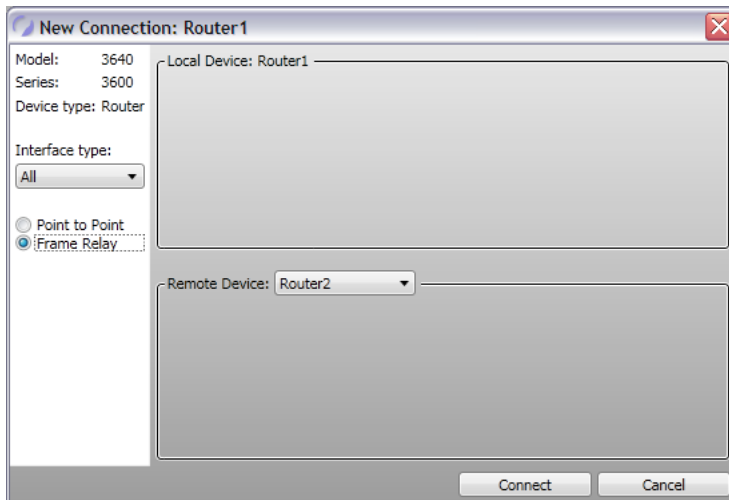
- Repeat steps 8 and 9 to create **Switch2**.
- On the navigation pane, expand the **Other Devices** list. Click and drag the WinPC device to the NetMap canvas
- Click **Create PC** to add **PC1**, which is the default name for the device.



- Repeat step 12 to create **PC2**.

Now that the devices have been added to the topology, we need to connect them together. Routers can be connected together using Serial or Ethernet connections. However, in this example we will connect them using Frame Relay. Let's begin by connecting the routers to each other, followed by adding connections to the other devices in the topology.

1. Right-click **Router1**, and select **New Connection**.
2. In the **New Connection** dialog box, select the **Frame Relay** radio button and click **Connect**.



3. In the **Name** field, enter **FR-1** to name the new device in the NetMap.
4. Select **Router1** in the **1st Router** drop-down list and Router1's **Serial0/0** interface in the **Interface** drop-down list.
5. Select **Router2** in the **2nd Router** drop-down list and Router2's **Serial0/0** interface in the **Interface** drop-down list. Then assign **201** in the **DLCI Destination** field.

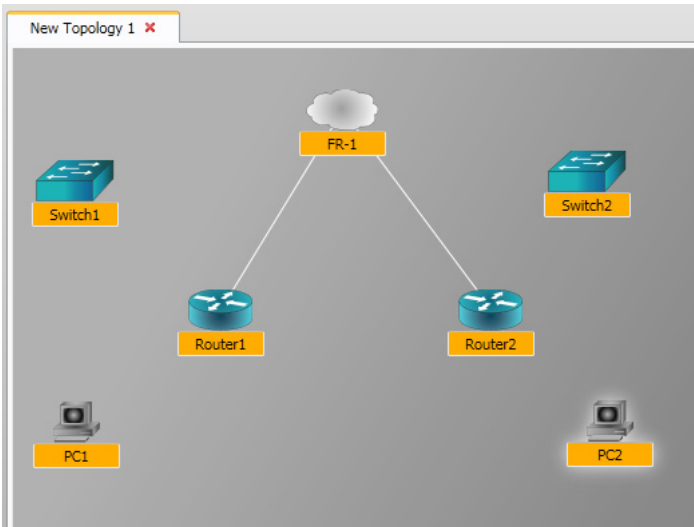
- Assign a DLCI value of 102 in the first DLCI Destination field, and configure a DLCI value of 201 in the second DLCI Destination field. Your configuration should look like this:

Name: FR-1 Apply Settings X

Number of Nodes: 2

From:	To:
1st Router: Router1 Interface: Serial0/0	DLCI Destination 102 Router2
2nd Router: Router2 Interface: Serial0/0	DLCI Destination 201 Router1

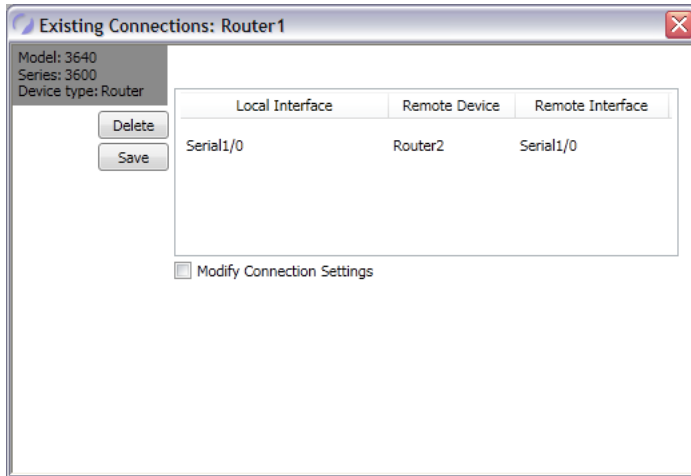
- Click **Apply Settings** to create the new Frame Relay cloud named FR-1.
- Move the devices in your topology around until it looks something like the graphic below:



Note: If you need to delete one of the links you create in the following steps, you should do the following:

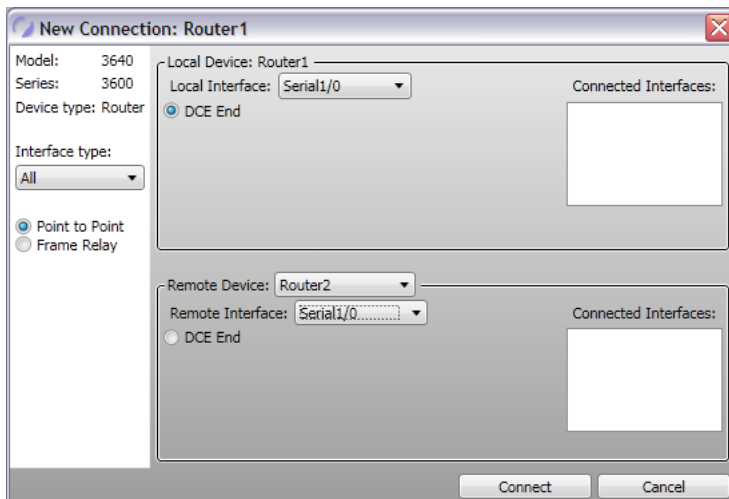
- Right-click one of the devices on either end of the link you wish to delete, and select **Existing Connections**.

- In the **Existing Connections** dialog box, select the interface on the device that is part of the link you wish to delete and click **Delete**.



Now let's add a serial link between Router1 and Router2 using the following steps:

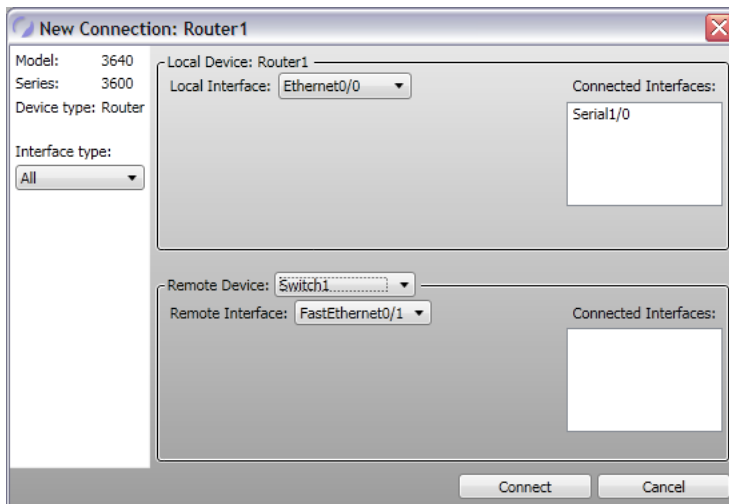
- Right-click **Router1**, and select **New Connection**.
- In the **New Connection** dialog box for Router1, select the **Serial1/0** interface from the **Local Interface** dropdown list.
- Select **Router2** in the **Remote Device** drop-down list.
- Select the **Serial1/0** interface from the **Remote Interface** drop-down list.
- Leave the selection for the DCE end of the link set to Router1. Your configurations should look like this:



- Click **Connect** to create the link.

Now let's add an Ethernet link between Router1 and Switch1.

1. Right-click **Router1**, and select **New Connection**.
2. In the **New Connection** dialog box, select **Ethernet0/0** from the **Local Interface** drop-down list.
3. Select **Switch1** in the **Remote Device** drop-down list.
4. Select **FastEthernet0/1** from the **Remote Interface** drop-down list.

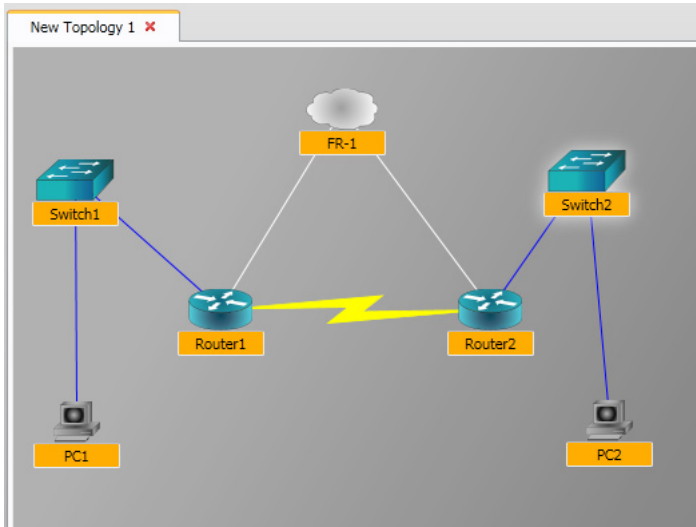


5. Click **Connect** to create the link between Router1 and Switch1.


Create the following remaining connections using the steps outlined above:

1. On Switch1, create a connection between Switch1's FastEthernet0/2 interface and PC1.
2. On Router2, create a connection between Router2's Ethernet0/0 interface and Switch2's FastEthernet0/1 interface.
3. On Switch2, create a connection between Switch2's FastEthernet0/2 interface and PC2.

Your final topology diagram should look similar to the one in the graphic below.



Now that you've created your topology diagram, you need to save it to the folder that will store all of the files for the lab you are creating.


1. Click the Save icon () on the quick-launch toolbar.
2. Choose or create a folder on your Desktop named **TestLab**.
3. In the **File name** field, name the file **TestLab** and click **Save**; the file will be saved as a *.bsn file type.
4. You might receive a message that the file was saved. Click **OK**.

The topology file you just created could also be used in multiple labs. You would only need to copy the TestLab.bsn file to the folders of other labs you wish to create. This simplifies the process of creating labs because you can create a single topology file that can be used in multiple labs.

[Return to Top](#)

Step 3: Create the Loading Configuration Files

At this point, you could create a lab using only the lab document and the topology file you just created. However, let's work through the steps needed to create loading configuration files. In our test lab, we are going to create loading configurations for Router1 and Switch1 that sets them to load with a host name set to their device name.

1. Click the Start Simulator icon () on the quick-launch toolbar.

2. In the console of Router1, enter the following commands to configure a host name of **Router1**:

```
Router>enable
Router#configure terminal
Router(config)#hostname Router1
```

3. Select **Switch1** from the **Devices** drop-down list.

4. In the console of Switch1, enter the following commands to configure a host name of **Switch1**:

```
Switch>enable
Switch#configure terminal
Switch(config)#hostname Switch1
```

5. Click the Save All icon () to save the configuration of both devices.

[Return to Top](#)

Step 4: Create the Completed Configuration Files

Let's create the final piece of our test lab: the completed configuration files. Remember that completed configuration files contain the running configuration that we want the user who follows our lab document to have when they finish working through our lab document. Our lab document has the user pinging between PC1 and Router1, so we need to configure some IP addresses on the devices and enable links.

In this example TestLab, the following are the commands the user should issue to enable a ping between PC1 and Router1. You should issue these commands on the appropriate devices so that you can create the complete configuration file.

On PC1:

```
C:>ipconfig /ip 192.168.100.2 255.255.255.0
C:>ipconfig /dg 192.168.100.1
```

On Switch1:

```
Switch1>enable
Switch1#configure terminal
Switch1(config)#ip default-gateway 192.168.100.1
```

On Router1:

```
Router1>enable
Router1#configure terminal
Router1(config)#interface ethernet 0/0
Router1(config-if)#ip address 192.168.100.1 255.255.255.0
Router1(config-if)#no shutdown
```

Finally, the user should issue a ping from PC1 to Router1 with a resulting successful ping.

On PC1:

```
C:>ping 192.168.100.1
Pinging 192.168.100.1 with 32 bytes of data:

Reply from 192.168.100.1: bytes=32 time=52ms TTL=241
Reply from 192.168.100.1: bytes=32 time=69ms TTL=241
Reply from 192.168.100.1: bytes=32 time=55ms TTL=241
Reply from 192.168.100.1: bytes=32 time=72ms TTL=241
Reply from 192.168.100.1: bytes=32 time=53ms TTL=241
```

Ping statistics for 192.168.100.1:

```
    Packets: Sent = 5, Received = 5, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 52ms, Maximum = 72ms, Average = 60ms
```

After you have issued the previous commands, you can save the running configurations on all of the devices so that you can use them as grading configurations; use the following steps.

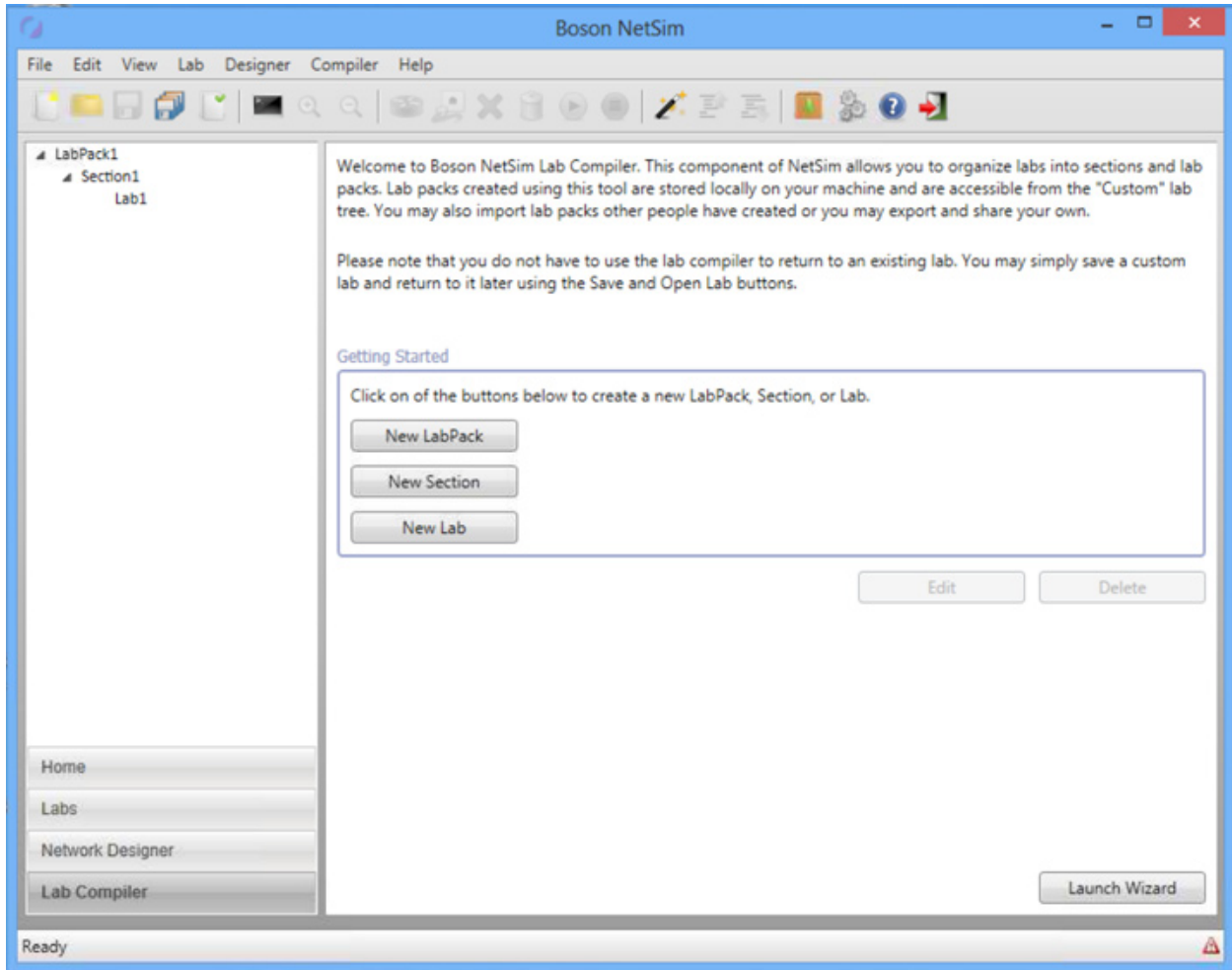
1. Click **File > Save As**.
2. In the **Save As** dialog box, navigate to the TestLab folder you created earlier on your Desktop.
3. Type **Completed** in the **File name** field, and click **Save**.

[Return to Top](#)

Step 5: Use the Lab Compiler to Create the Lab Pack

You've now created the required files (lab document, topology, and completed configuration) and optional file (loading configuration) needed for our lab, but you still have to create the lab file so you can share it with students. This section will detail how to use the Lab Compiler to create the lab and export it so that you can share it.

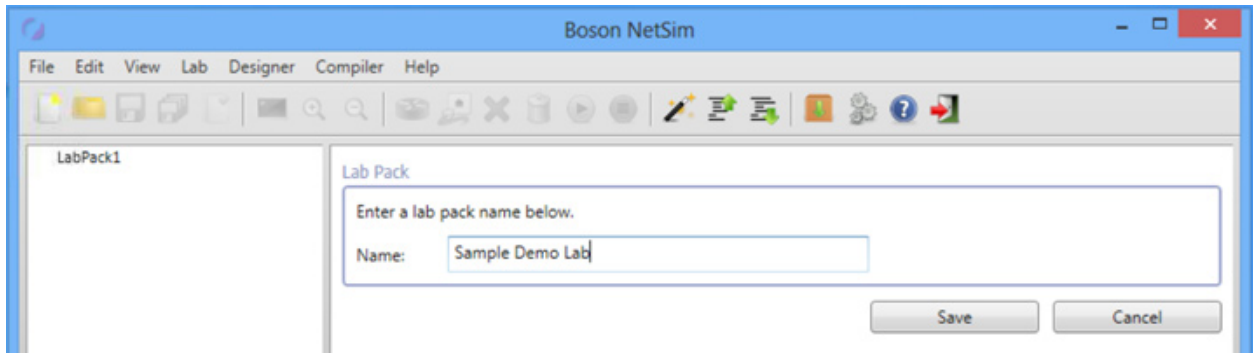
Click **Lab Compiler** in the navigation pane to open the Lab Compiler.



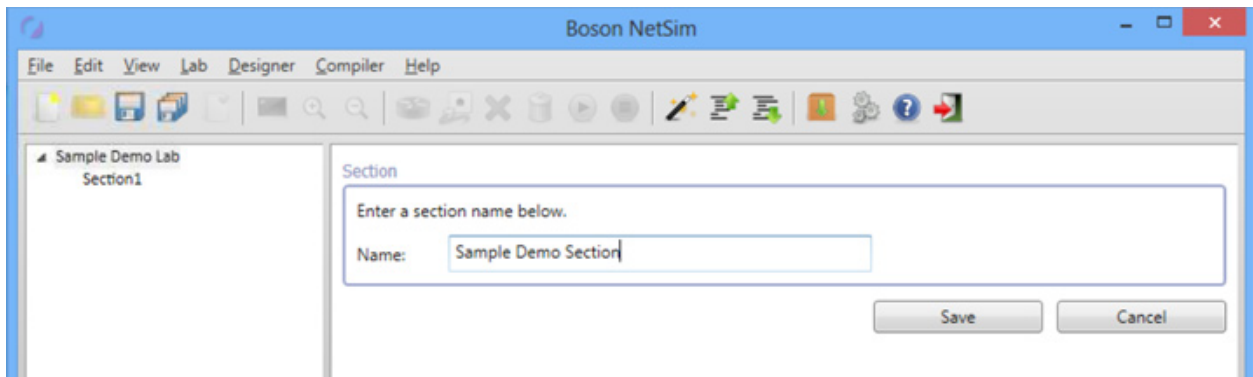
Perform the following steps to create the lab pack. The completed process is shown above just before the lab is saved.

1. Click **New Lab Pack** in the **Getting Started** section.

2. Enter a name for the lab pack. This is a lab pack title that will be displayed in the tree.



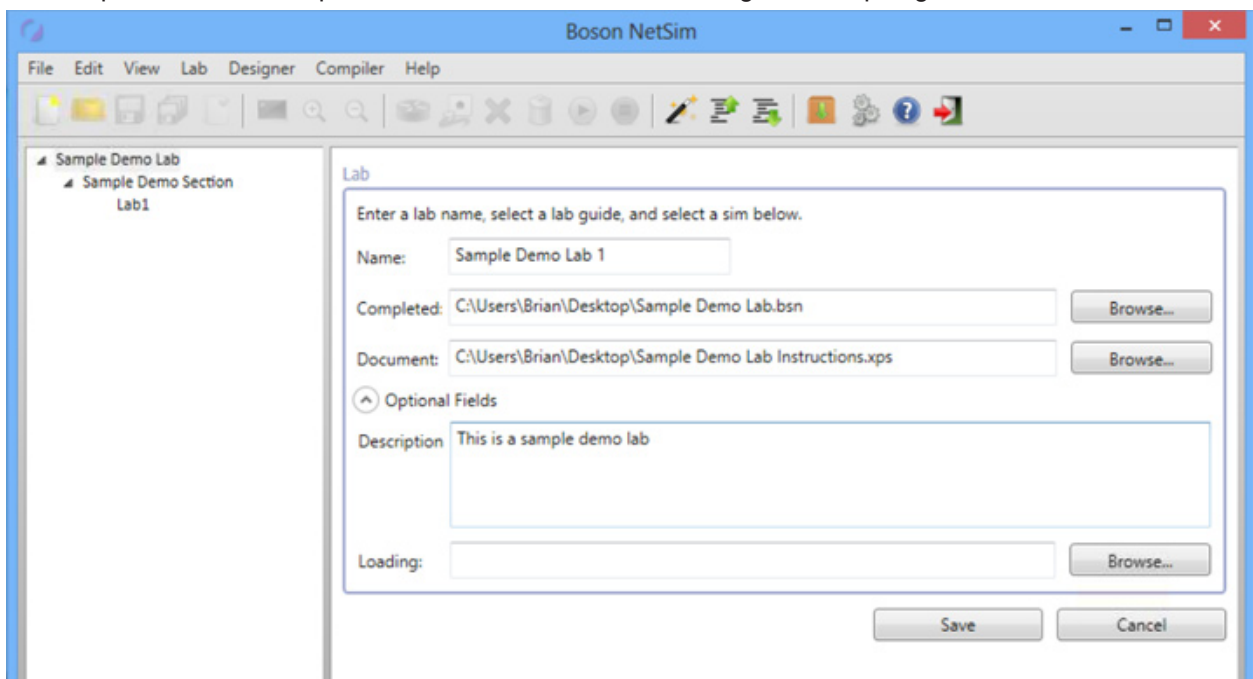
3. Click **Save**.
4. In the custom lab tree on the left, select the title of the lab pack where you want to add a new section.
5. Click **New Section** in the **Getting Started** section.
6. Enter a name for the section. This is a section title that will be displayed in the tree.



7. Click **Save**.
8. In the custom lab tree on the left, select the title of the section where you want to add a new lab.
9. Click **New Lab** in the **Getting Started** section.
10. Enter a name for the lab. This is a lab title that will be displayed in the tree.
11. Click **Browse**, and add a completed Boson Network file (.bsn).
12. Click **Browse**, and add a document. The only document files accepted are XPS documents (.xps).

You may optionally add a description and a loading BSN file to the lab. If neither file is desired, continue with step 9 below.

13. Click Optional Fields in the Lab section.
14. Enter a description for the lab. A description is displayed for every lab underneath the Lab Tree.
15. Add a loading Boson Network file (.bsn). The topology in the specified loading file must match the topology in the completed file from step 4. You will receive an error message if the topologies do not match.



16. Click **Save**.
17. To add additional labs to the lab pack section, repeat steps 1 through 12 and step 16 (and optional steps 13 through 15) until the lab pack section is complete.
18. You can edit, add, delete, import, and export lab packs, sections, and labs with the options on the Compiler menu and on the quick-launch toolbar.

Now that we have created a custom lab pack, lab section, and lab, we need to export it so that it can be used by the students.

1. Select the name of your lab pack on the navigation panel, and click **File > Export > Lab Pack**.
2. Type a name for your lab pack in the **Export Lab Pack** dialog box.
3. Click **Save**.

[Return to Top](#)

Although this document is rather lengthy, the process of creating a custom lab is fairly quick with a bit of practice. Now that you have a lab pack file saved, you can distribute it to your students and have them import it into their copy of NetSim by clicking **File > Import**.