Boot over IB (BoIB)
User's Manual

Rev 1.1

Boot over IB (BoIB) User's Manual

Document Number: 2917

Mellanox Technologies, Inc.
350 Oakmead Parkway
Sunnyvale, CA 94085
U.S.A.
www.mellanox.com

Tel: (408) 970-3400
Fax: (408) 970-3403

Mellanox Technologies Ltd
PO Box 586 Hermon Building
Yokneam 20692
Israel

Tel: +972-4-909-7200
Fax: +972-4-959-3245

Table of Contents

Revision History

Printed on February 23, 2009.

Rev 1.1 (23-February-2009)

- Support for single and dual port ConnectX® IB QDR and PCI 2.0 5.0GT/s devices
- *On dual-port ConnectX devices only*: Support for fail-over to other HCA port in case the first port fails to boot

Rev 1.0 (08-April-2008)

- First release

1 Boot over IB (BoIB)

1.1 Overview

This chapter describes "Mellanox Boot over IB" (BoIB), the software for Boot over Mellanox Technologies InfiniBand (IB) HCA devices. BoIB enables booting kernels or operating systems (OSs) from remote servers in compliance with the PXE specification.

BoIB is based on the open source project Etherboot/gPXE available at http://www.etherboot.org.

BoIB first initializes the HCA device. Then, it connects to a DHCP server to obtain its assigned IP address and network parameters, and also to obtain the source location of the kernel/OS to boot from. The DHCP server instructs BoIB to access the kernel/OS through a TFTP server, an iSCSI target, or other service.

Mellanox Boot over IB implements a network driver with IP over IB acting as the transport layer. IP over IB is part of the *Mellanox OFED for Linux* software package (see www.mellanox.com).

The binary code is exported by the device as an expansion ROM image.

1.2 Supported Mellanox HCA Devices and Firmware

The package supports Mellanox Technologies devices listed in Table 1.

Table 1 - Supported Mellanox Technologies Devices (and PCI Device IDs)

Device Name	PCI Device ID Decimal (Hexadecimal)	Firmware Name [1, 2]
MT25408 ConnectX – IB@ SDR, PCI Express 2.0 2.5GT/s	25408 (0x6340)	fw-25408
MT25408 ConnectX – IB@ DDR, PCI Express 2.0 2.5GT/s	25418 (0x634a)	fw-25408
MT25408 ConnectX – IB@ DDR, PCI Express 2.0 5.0GT/s	26418 (0x6732)	fw-25408
MT25408 ConnectX – IB@ QDR, PCI Express 2.0 5.0GT/s	26428 (0x673c)	fw-25408
MT25208 InfiniHost® III Ex	25218 (0x6282)	fw-25218
MT25204 InfiniHost® III Lx	25204 (0x6274)	fw-25204

1. Firmware can be downloaded from www.mellanox.com > Downloads > Firmware > Customized Firmware.
2. See the release notes file for the required firmware versions.

1.3 Tested Platforms

See the Boot over IB Release Notes (`boot_over_ib_release_notes.txt`).

1.4 BoIB Package

The Boot over IB package is provided as a tarball (.tgz extension). Uncompress it using the command “tar zxf <package file name>”. The tarball contains the following files:

1. PXE binary files for Mellanox HCA devices
   - HCA: Single/Dual port ConnectX IB SDR (PCI DevID: 25408)

     `CONNECTX_IB_25408_ROM-X.X.XXX.rom`
   - HCA: Single/Dual port ConnectX IB DDR (PCI DevID: 25418)

     `CONNECTX_IB_25418_ROM-X.X.XXX.rom`
   - HCA: Single/Dual port ConnectX IB DDR & PCI Express 2.0 5.0GT/s (PCI DevID: 26418)

     `CONNECTX_IB_26418_ROM-X.X.XXX.rom`
   - HCA: Single/Dual port ConnectX IB QDR & PCI Express 2.0 5.0GT/s (PCI DevID: 26428)

     `CONNECTX_IB_26428_ROM-X.X.XXX.rom`
   - HCA: InfiniHost III Ex in Mem-Free mode (PCI DevID: 25218)

     `IHOST3EX_PORT1_ROM-X.X.XXX.rom` (IB Port 1)

     `IHOST3EX_PORT2_ROM-X.X.XXX.rom` (IB Port 2)
   - HCA: InfiniHost III Lx (PCI DevID: 25204)

     `IHOST3LX_ROM-X.X.XXX.rom` (single IB Port device)
2. A `docs/` directory includes the following files:
   - `Boot-over-IB_User_Manual.pdf` – this user’s manual
   - `boot_over_ib_release_notes.txt` – release notes
   - `dhcpd.conf` – sample DHCP configuration file
   - `dhcp.patch` – patch file for DHCP v3.1.2

1.5 Reference Documents and Downloads

- The Mellanox Firmware Tools (MFT) package and documentation can be downloaded from www.mellanox.com > Downloads > Firmware Tools.
- The MLNX OFED SW stack and documentation can be downloaded from www.mellanox.com > Products > InfiniBand Software/Drivers.

2 Burning the Expansion ROM Image

2.1 Prerequisites

1. Firmware packages

   The appropriate firmware .mlx packages (fw-25408, fw-25208, and/or fw-25204) can be downloaded from Mellanox Technologies' Web site – see www.mellanox.com > Downloads > Firmware > Customized Firmware.

2. Firmware Configuration (.ini) Files

   For standard Mellanox products, .ini files are included in the firmware .mlx packages. For help in identifying the correct .ini file of your adapter hardware, please refer to *MFT User's Manual* (see Section 1.5).

3. Expansion ROM Image

   The expansion ROM images are provided as part of the SW package and are listed in Section 1.4.

4. Firmware Burning Tools

   You need to install the Mellanox Firmware Tools (MFT) package (version 2.5.0 or later) in order to burn the PXE ROM image. To download MFT, see *Firmware Tools* under www.mellanox.com > Downloads.

   Specifically, you will be using the `mlxburn` tool to create and burn a composite image (from an adapter device's firmware and the PXE ROM image) onto the same Flash device of the adapter.

2.2 Burning the Image

To burn the composite image, perform the following steps:

1. Obtain the MST device name. Run:

   ```
   mst start
   mst status
   ```

   The device name will be of the form: `mt<dev_id>_pci{_cr0|conf0}`.[1]

2. Create and burn the composite image. Run:

   ```
   mlxburn -d <mst device name> -fw <FW .mlx file> -conf <.ini file> \
           -exp_rom <expansion ROM image>
   ```

   Example on Linux:

   ```
   mlxburn -dev /dev/mst/mt25418_pci_cr0 -fw fw-25408-X_X_XXX.mlx \
           -conf MHGH28-XTC.ini -exp_rom ConnectX_IB_25418_ROM-X_X_XXX.rom
   ```

   Example on Windows:

   ```
   mlxburn -dev mt25418_pci_cr0 -fw fw-25408-X_X_XXX.mlx \
           -conf MHGH28-XTC.ini -exp_rom ConnectX_IB_25418_ROM-X_X_XXX.rom
   ```

1. Depending on the OS, the device name may be superceded by a prefix.

3 Preparing the DHCP Server in Linux Environment

The DHCP server plays a major role in the boot process by assigning IP addresses for BoIB clients and instructing the clients where to boot from. BoIB requires that the DHCP server runs on a machine which supports IP over IB.

3.1 Installing DHCP

Note: Prior to installing DHCP, make sure that *Mellanox OFED for Linux* is already installed on your DHCP server – see www.mellanox.com.

Download and install DHCP v3.1.2 from www.isc.org. A special patch for DHCP is required for supporting IPoIB. The patch file "`dhcp.patch`" is available under the `docs/` directory.

Standard DHCP fields holding MAC addresses are not large enough to contain an IPoIB hardware address. To overcome this problem, DHCP over InfiniBand messages convey a client identifier field used to identify the DHCP session. This client identifier field can be used to associate an IP address with a client identifier value, such that the DHCP server will grant the same IP address to any client that conveys this client identifier.

Note: Refer to the DHCP documentation for more details how to make this association.

3.2 Configuring the DHCP Server

3.2.1 For ConnectX Family Devices

When a BoIB client boots, it sends the DHCP server various information, including its DHCP client identifier. This identifier is used to distinguish between the various DHCP sessions. The value of the client identifier is composed of an 8-byte port GUID (separated by colons and represented in hexadecimal digits).

Extracting the Port GUID – Method I

To obtain the port GUID, run the following commands:

Note: The following MFT commands assume that the Mellanox Firmware Tools (MFT) package has been installed on the client machine.

```
host1# mst start
host1# mst status
```

The device name will be of the form: /dev/mst/mt<dev_id>_pci{_cr0|conf0}. Use this device name to obtain the Port GUID via the following query command:

```
flint -d <MST_DEVICE_NAME> q
```

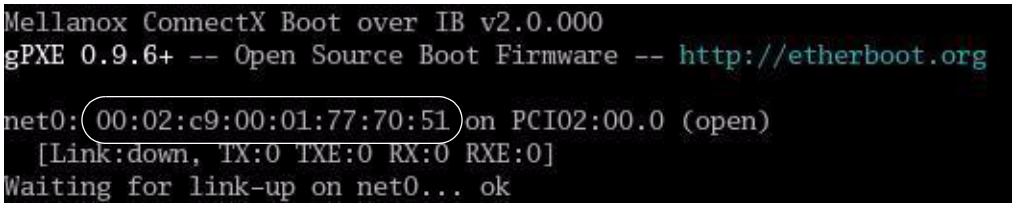
Example with ConnectX IB DDR (& PCI Express 2.0 2.5GT/s) as the HCA device:

```
host1# flint -d /dev/mst/mt25418_pci_cr0 q
Image type:      ConnectX
FW Version:      2.6.616
Device ID:       25418
Chip Revision:   A0
Description:     Node             Port1            Port2            Sys image
GUIDs:      0002c90300001038  0002c90300001039  0002c9030000103a  0002c9030000103b
MACs:                             0002c9001039     0002c900103a
Board ID:        n/a (MT_04A0110002)
VSD:             n/a
PSID:            MT_04A0110002
```

Assuming that BoIB is connected via Port 1, then the Port GUID is 00:02:c9:03:00:00:10:39.

Extracting the Port GUID – Method II

An alternative method for obtaining the port GUID involves booting the client machine via BoIB. This requires having a Subnet Manager running on one of the machines in the InfiniBand subnet. The 8 bytes can be captured from the boot session as shown in the figure below.



Placing Client Identifiers in /etc/dhcpd.conf

The following is an excerpt of a `/etc/dhcpd.conf` example file showing the format of representing a client machine for the DHCP server.

```
host host1 {
   next-server 11.4.3.7;
   filename "pxelinux.0";
   fixed-address 11.4.3.130;
   option dhcp-client-identifier = 00:02:c9:03:00:00:10:39;
}
```

3.2.2 For InfiniHost III Family Devices (PCI Device IDs: 25204, 25218)

When a BoIB client boots, it sends the DHCP server various information, including its DHCP client identifier. This identifier is used to distinguish between the various DHCP sessions.

The value of the client identifier is composed of 21 bytes (separated by colons) having the following components:

```
20:<QP Number - 4 bytes>:<GID - 16 bytes>
```

Note: Bytes are represented as two-hexadecimal digits.

Extracting the Client Identifier – Method I

The following steps describe one method for extracting the client identifier:

Step 1. QP Number equals 00:55:04:01 for InfiniHost III Ex and InfiniHost III Lx HCAs.

Step 2. GID is composed of an 8-byte subnet prefix and an 8-byte Port GUID. The subnet prefix is fixed for the supported Mellanox HCAs, and is equal to fe:80:00:00:00:00:00:00. The next steps explains how to obtain the Port GUID.

Step 3. To obtain the Port GUID, run the following commands:

Note: The following MFT commands assume that the Mellanox Firmware Tools (MFT) package has been installed on the client machine.

```
host1# mst start
host1# mst status
```

The device name will be of the form: /dev/mst/mt<dev_id>_pci{_cr0|conf0}. Use this device name to obtain the Port GUID via a query command.

```
flint -d <MST_DEVICE_NAME> q
```

Example with InfiniHost III Ex as the HCA device:

```
host1# flint -d /dev/mst/mt25218_pci_cr0 q
Image type:      Failsafe
FW Version:      5.3.0
Rom Info:        type=GPXE version=1.0.0 devid=25218 port=2
I.S. Version:    1
Device ID:       25218
Chip Revision:   A0
Description:  Node          Port1           Port2         Sys image
GUIDs:  0002c90200231390 0002c90200231391 0002c90200231392 0002c90200231393
Board ID:         (MT_0370110001)
VSD:
PSID:            MT_0370110001
```

Assuming that BoIB is connected via Port 2, then the Port GUID is 00:02:c9:02:00:23:13:92.

Step 4. The resulting client identifier is the concatenation, from left to right, of 20, the QP_Number, the subnet prefix, and the Port GUID.

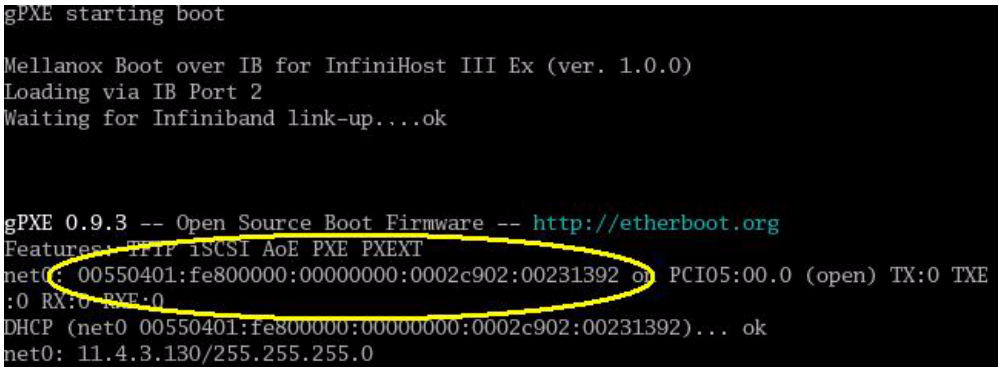
In the example above this yields the following DHCP client identifier:

```
20:00:55:04:01:fe:80:00:00:00:00:00:00:00:02:c9:02:00:23:13:92
```

Extracting the Client Identifier – Method II

An alternative method for obtaining the 20 bytes of QP Number and GID involves booting the client machine via BoIB. This requires having a Subnet Manager running on one of the machines in

the InfiniBand subnet. The 20 bytes can be captured from the boot session as shown in the figure below.



Concatenate the byte '20' to the left of the captured 20 bytes, then separate every byte (two hexadecimal digits) with a colon. You should obtain the same result shown in Step 4 above.

Placing Client Identifiers in /etc/dhcpd.conf

The following is an excerpt of a `/etc/dhcpd.conf` example file showing the format of representing a client machine for the DHCP server.

```
host host1 {
   next-server 11.4.3.7;
   filename "pxelinux.0";
   fixed-address 11.4.3.130;
   option dhcp-client-identifier = \
   20:00:55:04:01:fe:80:00:00:00:00:00:00:02:c9:02:00:23:13:92;
}
```

3.3 Running the DHCP Server

In order for the DHCP server to provide configuration records for clients, an appropriate configuration file needs to be created. By default, the DHCP server looks for a configuration file called `dhcpd.conf` under `/etc`. You can either edit this file or create a new one and provide its full path to the DHCP server using the -cf flag. See a file example at `docs/dhcpd.conf` of this package.

The DHCP server must run on a machine which has loaded the IPoIB module.

To run the DHCP server from the command line, enter:

```
dhcpd <IB network interface name> -d
```

Example:

```
host1# dhcpd ib0 -d
```

3.4 Running the DHCP Client (Optional)

Note: A DHCP client can be used if you need to prepare a diskless machine with an IB driver. See Step 8 under "Example: Adding an IB Driver to initrd (Linux)".

In order to use a DHCP client identifier, you need to first create a configuration file that defines the DHCP client identifier. Then run the DHCP client with this file using the following command:

```
dhclient –cf <client conf file> <IB network interface name>
```

Example of a configuration file for the ConnectX (PCI Device ID 25418), called `dhclient.conf`:

```
# The value indicates a hexadecimal number
interface "ib1" {
send dhcp-client-identifier 00:02:c9:03:00:00:10:39;
}
```

Example of a configuration file for InfiniHost III Ex (PCI Device ID 25218), called `dhclient.conf`:

```
# The value indicates a hexadecimal number
interface "ib1" {
send dhcp-client-identifier 20:00:55:04:01:fe:80:00:00:00:00:00:00:02:c9:02:00:23:13:92;
}
```

In order to use the configuration file, run:

```
host1# dhclient –cf dhclient.conf ib1
```

4 Subnet Manager – OpenSM

BoIB requires a Subnet Manager to be running on one of the machines in the IB network. OpenSM is part of the *Mellanox OFED for Linux* software package and can be used to accomplish this. Note that OpenSM may be run on the same host running the DHCP server but it is not mandatory.

5 TFTP Server

When you set the 'filename' parameter in your DHCP configuration file to a non-empty filename, the client will ask for this file to be passed through TFTP. For this reason you need to install a TFTP server.

6 BIOS Configuration

The expansion ROM image presents itself to the BIOS as a boot device. As a result, the BIOS will add to the list of boot devices "MLNX IB <ver>" for a ConnectX device or "gPXE" for an InfiniHost III device. The priority of this list can be modified through BIOS setup.

7 Operation

7.1 Prerequisites

- Make sure that your client is connected to the server(s)
- The BoIB image is already programmed on the adapter card – see Section 2
- Start the Subnet Manager as described in Section 4
- Configure and start the DHCP server as described in Section 3
- Configure and start at least one of the services iSCSI Target (see Section 9) and/or TFTP (see Section 5)

7.2 Starting Boot

Boot the client machine and enter BIOS setup to configure “MLNX IB” (for ConnectX family) or “gPXE” (for InfiniHost III family) to be the first on the boot device priority list – see Section 6.

Note: On dual-port network adapters, the client first attempts to boot from Port 1. If this fails, it switches to boot from Port 2. Note also that the driver waits up to 45 sec for each port to come up.

If MLNX IB/gPXE was selected through BIOS setup, the client will boot from BoIB. The client will display BoIB attributes and wait for IB port configuration by the subnet manager.

For ConnectX:

```
Mellanox ConnectX Boot over IB v2.0.000
gPXE 0.9.6+ -- Open Source Boot Firmware -- http://etherboot.org

net0: 00:02:c9:00:01:77:70:51 on PCI02:00.0 (open)
  [Link:down, TX:0 TXE:0 RX:0 RXE:0]
Waiting for link-up on net0... ok
```

For InfiniHost III Ex:

```
gPXE starting boot

Mellanox Boot over IB for InfiniHost III Ex (ver. 1.0.0)
Loading via IB Port 2
Waiting for Infiniband link-up....ok
```

After configuring the IB port, the client attempts connecting to the DHCP server to obtain an IP address and the source location of the kernel/OS to boot from.

For ConnectX:

```
Mellanox ConnectX Boot over IB v2.0.000
gPXE 0.9.6+ -- Open Source Boot Firmware -- http://etherboot.org

net0: 00:02:c9:00:01:77:70:51 on PCI02:00.0 (open)
  [Link:down, TX:0 TXE:0 RX:0 RXE:0]
Waiting for link-up on net0... ok
DHCP (net0 00:02:c9:00:01:77:70:51).... ok
net0: 11.4.3.130/255.255.255.0 gw 0.0.0.0
```

For InfiniHost III Ex:

```
gPXE 0.9.3 -- Open Source Boot Firmware -- http://etherboot.org
Features: TFTP iSCSI AoE PXE PXEXT
net0: 00550401:fe800000:00000000:0002c902:00231392 on PCI05:00.0 (open) TX:0 TXE
:0 RX:0 RXE:0
DHCP (net0 00550401:fe800000:00000000:0002c902:00231392)... ok
net0: 11.4.3.130/255.255.255.0
```

Next, BoIB attempts to boot as directed by the DHCP server.

8 Diskless Machines

Mellanox Boot over IB supports booting diskless machines. To enable using an IB driver, the (remote) kernel or `initrd` image must include and be configured to load the IB driver, including IPoIB.

This can be achieved either by compiling the HCA driver into the kernel, or by adding the device driver module into the `initrd` image and loading it.

The IB driver requires loading the following modules in the specified order (see Section 8.1 for an example):

- ib_addr.ko
- ib_core.ko
- ib_mad.ko
- ib_sa.ko
- ib_cm.ko
- ib_uverbs.ko
- ib_ucm.ko
- ib_umad.ko
- iw_cm.ko
- rdma_cm.ko
- rdma_ucm.ko
- mlx4_core.ko
- mlx4_ib.ko
- ib_mthca.ko
- ib_ipoib.ko

8.1 Example: Adding an IB Driver to initrd (Linux)

Prerequisites

1. The BoIB image is already programmed on the HCA card.
2. The DHCP server is installed and configured as described in Section 3.2, "Configuring the DHCP Server", and connected to the client machine.
3. An `initrd` file.
4. To add an IB driver into `initrd`, you need to copy the IB modules to the diskless image. Your machine needs to be pre-installed with a *Mellanox OFED for Linux* ISO image (available for download from www.mellanox.com > Products > IB SW/Drivers) that is appropriate for the kernel version the diskless image will run.

   Note: The remainder of this section assumes that Mellanox OFED has been installed on your machine.

Adding the IB Driver to the initrd File

Warning! The following procedure modifies critical files used in the boot procedure. It must be executed by users with expertise in the boot process. Improper application of this procedure may prevent the diskless machine from booting.

Step 1. Back up your current `initrd` file.

Step 2. Make a new working directory and change to it.

```
host1$ mkdir /tmp/initrd_ib
host1$ cd /tmp/initrd_ib
```

Step 3. Normally, the `initrd` image is zipped. Extract it using the following command:

```
host1$ gzip -dc <initrd image> | cpio -id
```

The initrd files should now be found under /tmp/initrd_ib

Step 4. Create a directory for the InfiniBand modules and copy them.

```
host1$ mkdir -p /tmp/initrd_ib/lib/modules/ib
host1$ cd /lib/modules/`uname -r`/updates/kernel/drivers
host1$cp infiniband/core/ib_addr.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/ib_core.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/ib_mad.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/ib_sa.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/ib_cm.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/ib_uverbs.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/ib_ucm.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/ib_umad.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/iw_cm.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/rdma_cm.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/core/rdma_ucm.ko /tmp/initrd_ib/lib/modules/ib
host1$cp net/mlx4/mlx4_core.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/hw/mlx4/mlx4_ib.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/hw/mthca/ib_mthca.ko /tmp/initrd_ib/lib/modules/ib
host1$cp infiniband/ulp/ipoib/ib_ipoib.ko /tmp/initrd_ib/lib/modules/ib
```

Step 5. IB requires loading an IPv6 module. If you do not have it in your `initrd`, please add it using the following command:

```
host1$ cp /lib/modules/`uname -r`/kernel/net/ipv6/ipv6.ko \
/tmp/initrd_ib/lib/modules
```

Step 6. To load the modules, you need the `insmod` executable. If you do not have it in your `initrd`, please add it using the following command:

```
host1$ cp /sbin/insmod /tmp/initrd_ib/sbin/
```

Step 7. If you plan to give your IB device a static IP address, then copy `ifconfig`. Otherwise, skip this step.

```
host1$ cp /sbin/ifconfig /tmp/initrd_ib/sbin
```

Step 8. If you plan to obtain an IP address for the IB device through DHCP, then you need to copy the DHCP client which was compiled specifically to support IB;
Otherwise, skip this step.

To continue with this step, DHCP client v3.1.2 needs to be already installed on the machine you are working with.

Copy the DHCP client v3.1.2 file and all the relevant files as described below.

```
host1# cp <path to DHCP client v3.1.2>/dhclient /tmp/initrd_ib/sbin
host1# cp <path to DHCP client v3.1.2>/dhclient-script /tmp/initrd_ib/sbin
host1# mkdir -p /tmp/initrd_ib/var/state/dhcp
host1# touch /tmp/initrd_ib/var/state/dhcp/dhclient.leases
host1# cp /bin/uname /tmp/initrd_ib/bin
host1# cp /usr/bin/expr /tmp/initrd_ib/bin
host1# cp /sbin/ifconfig /tmp/initrd_ib/bin
host1# cp /bin/hostname /tmp/initrd_ib/bin
```

Create a configuration file for the DHCP client (as described in Section 3.4) and place it under `/tmp/initrd_ib/sbin`. The following is an example of such a file (called `dclient.conf`):

dhclient.conf:

```
# The value indicates a hexadecimal number

# For a ConnectX device
interface "ib0" {
send dhcp-client-identifier 00:02:c9:03:00:00:10:39;
}

# For an InfiniHost III Ex device
interface "ib1" {
send dhcp-client-identifier \
20:00:55:04:01:fe:80:00:00:00:00:00:00:02:c9:02:00:23:13:92;
}
```

Step 9. Now you can add the commands for loading the copied modules into the file `init`. Edit the file `/tmp/initrd_ib/init` and add the following lines at the point you wish the IB driver to be loaded.

Warning! The order of the following commands (for loading modules) is critical.

```
echo "loading ipv6"
/sbin/insmod /lib/modules/ipv6.ko
echo "loading IB driver"
/sbin/insmod /lib/modules/ib/ib_addr.ko
/sbin/insmod /lib/modules/ib/ib_core.ko
```

```
/sbin/insmod /lib/modules/ib/ib_mad.ko
/sbin/insmod /lib/modules/ib/ib_sa.ko
/sbin/insmod /lib/modules/ib/ib_cm.ko
/sbin/insmod /lib/modules/ib/ib_uverbs.ko
/sbin/insmod /lib/modules/ib/ib_ucm.ko
/sbin/insmod /lib/modules/ib/ib_umad.ko
/sbin/insmod /lib/modules/ib/iw_cm.ko
/sbin/insmod /lib/modules/ib/rdma_cm.ko
/sbin/insmod /lib/modules/ib/rdma_ucm.ko
/sbin/insmod /lib/modules/ib/mlx4_core.ko
/sbin/insmod /lib/modules/ib/mlx4_ib.ko
/sbin/insmod /lib/modules/ib/ib_mthca.ko
/sbin/insmod /lib/modules/ib/ib_ipoib.ko
```

Note: In case of interoperability issues between iSCSI and Large Receive Offload (LRO), change the last command above as follows to disable LRO:

```
/sbin/insmod /lib/modules/ib/ib_ipoib.ko lro=0
```

Step 10. Now you can assign an IP address to your IB device by adding a call to `ifconfig` or to the DHCP client in the `init` file after loading the modules. If you wish to use the DHCP client, then you need to add a call to the DHCP client in the `init` file after loading the IB modules. For example:

```
/sbin/dhclient -cf /sbin/dhclient.conf ib1
```

Step 11. Save the `init` file.

Step 12. Close `initrd`.

```
host1$ cd /tmp/initrd_ib
host1$ find ./ | cpio -H newc -o > /tmp/new_initrd_ib.img
host1$ gzip /tmp/new_init_ib.img
```

Step 13. At this stage, the modified `initrd` (including the IB driver) is ready and located at `/tmp/new_init_ib.img.gz`. Copy it to the original `initrd` location and rename it properly.

9 iSCSI Boot

Mellanox Boot over IB enables an iSCSI-boot of an OS located on a remote iSCSI Target. It has a built-in iSCSI Initiator which can connect to the remote iSCSI Target and load from it the kernel and `initrd`. There are two instances of connection to the remote iSCSI Target: the first is for getting the kernel and `initrd` via BoIB, and the second is for loading other parts of the OS via `initrd`.

Note: Linux distributions such as SuSE Linux Enterprise Server 10 SPx and Red Hat Enterprise Linux 5.1 (or above) can be directly installed on an iSCSI target. At the end of this direct installation, initrd is capable to continue loading other parts of the OS on the iSCSI target. (Other distributions may also be suitable for direct installation on iSCSI targets.)

If you choose to continue loading the OS (after boot) through the HCA device driver, please verify that the initrd image includes the HCA driver as described in Section 8.

9.1 Configuring an iSCSI Target in Linux Environment

Prerequisites

Step 1. Make sure that an iSCSI Target is installed on your server side.

Tip You can download and install an iSCSI Target from the following location: http://sourceforge.net/project/showfiles.php?group_id=108475&package_id=117141

Step 2. Dedicate a partition on your iSCSI Target on which you will later install the operating system

Step 3. Configure your iSCSI Target to work with the partition you dedicated. If, for example, you choose partition /dev/sda5, then edit the iSCSI Target configuration file `/etc/ietd.conf` to include the following line under the iSCSI Target iqn line:

```
Lun 0 Path=/dev/sda5,Type=fileio
```

Tip The following is an example of an iSCSI Target iqn line:
`Target iqn.2007-08.7.3.4.10:iscsiboot`

Step 4. Start your iSCSI Target.

Example:

```
host1# /etc/init.d/iscsitarget start
```

Configuring the DHCP Server to Boot From an iSCSI Target

Configure DHCP as described in Section 3.2, “Configuring the DHCP Server”.

Edit your DHCP configuration file (`/etc/dhcpd.conf`) and add the following lines for the machine(s) you wish to boot from the iSCSI Target:

```
Filename "";
option root-path "iscsi:iscsi_target_ip::::iscsi_target_iqn";
```

The following is an example for configuring an IB device to boot from an iSCSI Target:

```
host  host1{
filename "";

# For a ConnectX device comment out the following line
# option dhcp-client-identifier = 00:02:c9:03:00:00:10:39;

# For an InfiniHost III Ex comment out the following line
# option  dhcp-client-identifier = \
# fe:00:55:00:41:fe:80:00:00:00:00:00:00:00:02:c9:03:00:00:0d:41;

option root-path "iscsi:11.4.3.7::::iqn.2007-08.7.3.4.10:iscsiboot";
}
```

9.2 iSCSI Boot Example of SLES 10 SP2 OS

This section provides an example of installing the SLES 10 SP2 operating system on an iSCSI target and booting from a diskless machine via BoIB. Note that the procedure described below assumes the following:

- The client's LAN card is recognized during installation
- The iSCSI target can be connected to the client via LAN and InfiniBand

Prerequisites

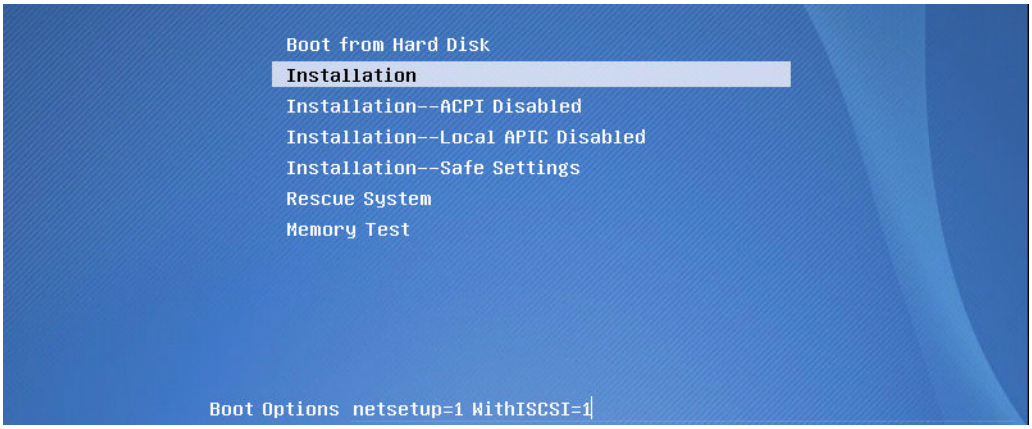
See Section 7.1 on page 18.

Warning! The following procedure modifies critical files used in the boot procedure. It must be executed by users with expertise in the boot process. Improper application of this procedure may prevent the diskless machine from booting.

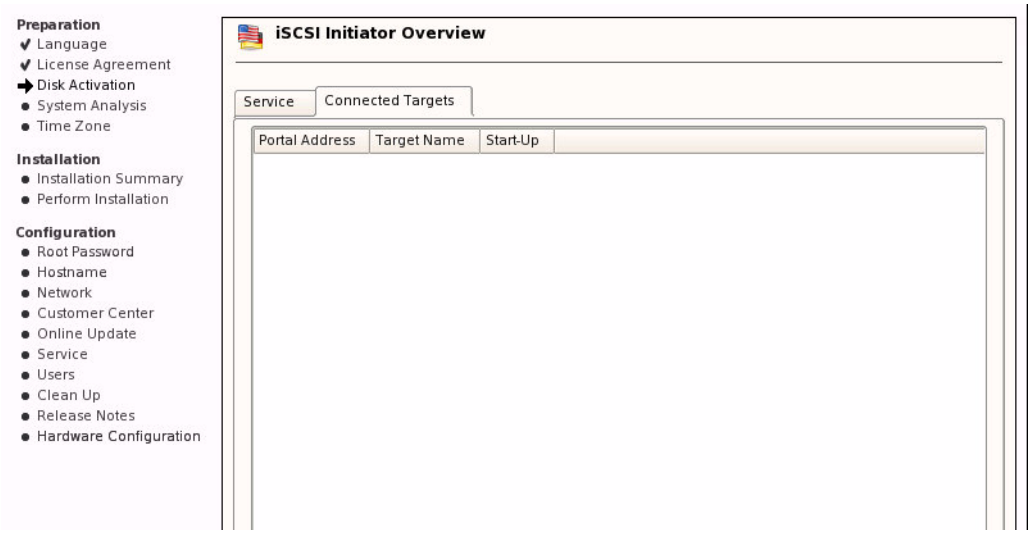
Procedure

Step 1. Load the SLES 10 SP2 installation disk and enter the following parameters as boot options:

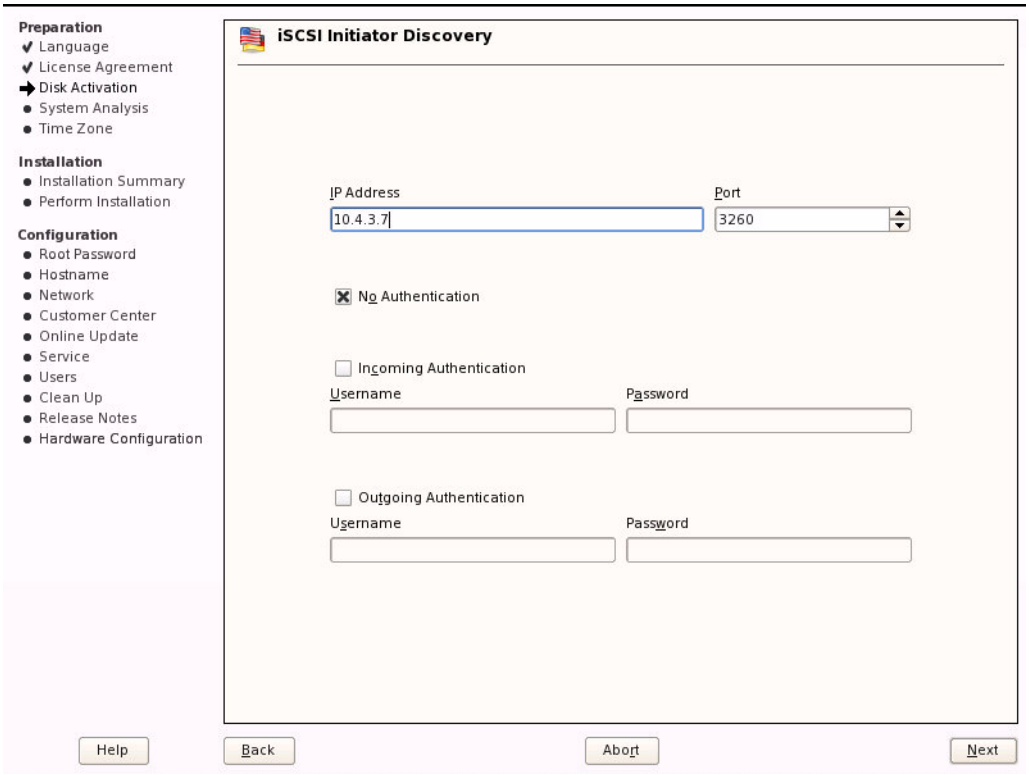
```
netsetup=1  WithISCSI=1
```



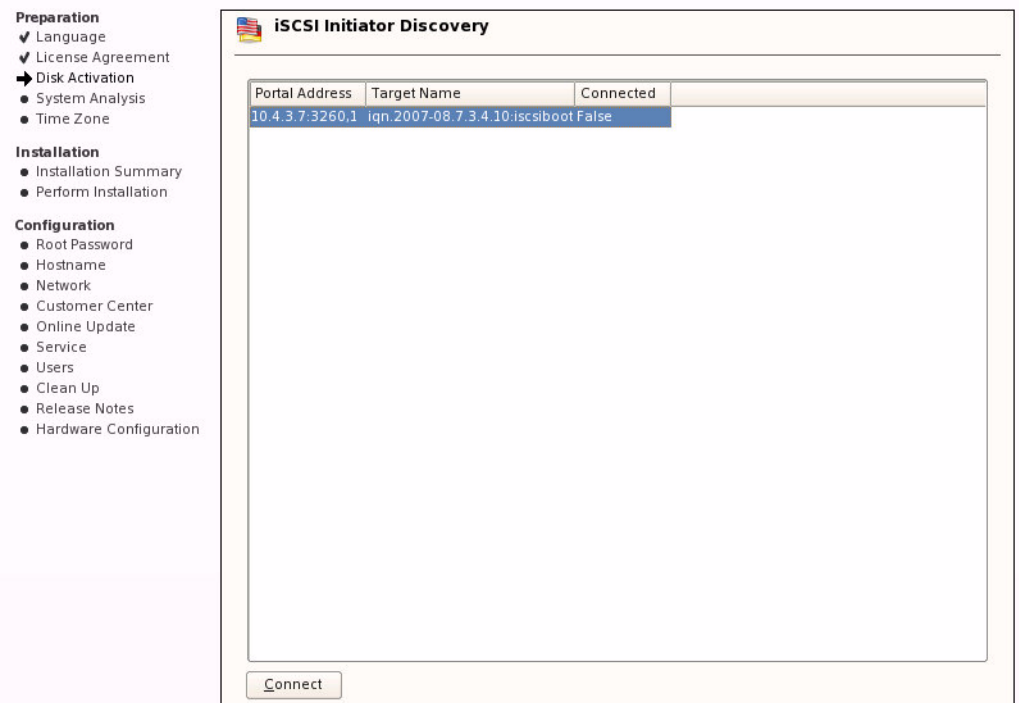
Step 2. Continue with the procedure as instructed by the installation program until the “iSCSI Initiator Overview” window appears.



Step 3. Click the Add tab in the iSCSI Initiator Overview window. An iSCSI Initiator Discovery window will pop up. Enter the IP Address of your iSCSI target and click Next.

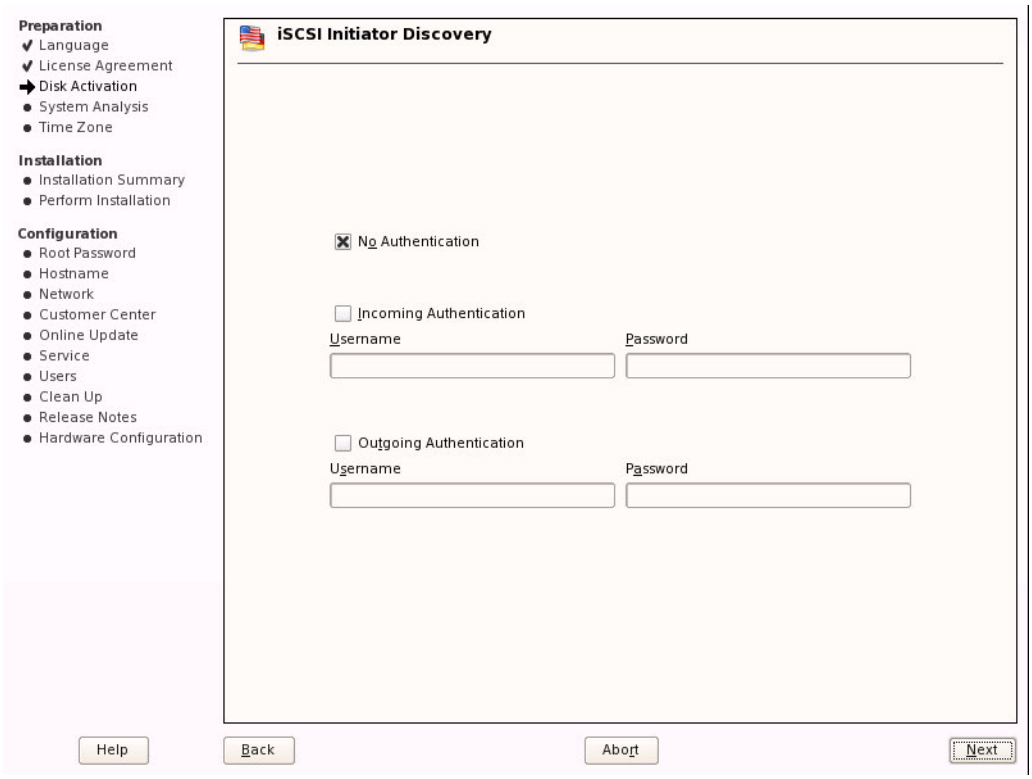


Step 4. Details of the discovered iSCSI target(s) will be displayed in the iSCSI Initiator Discovery window. Select the target that you wish to connect to and click Connect.

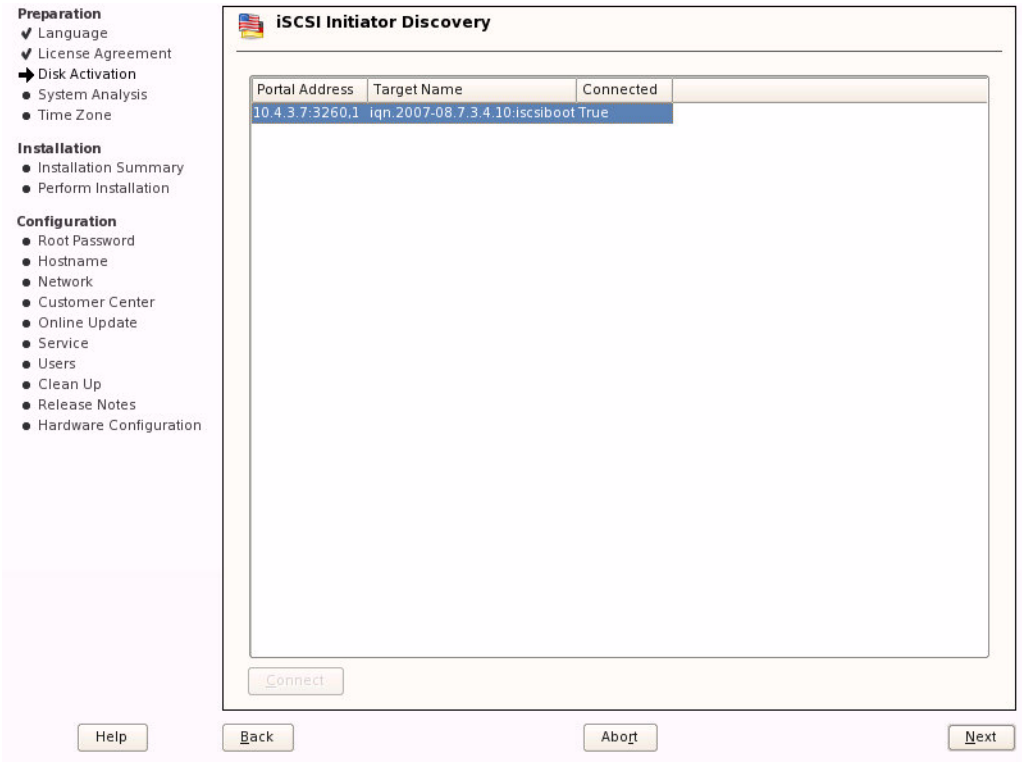


Tip If no iSCSI target was recognized, then either the target was not properly installed or no connection was found between the client and the iSCSI target. Open a shell to ping the iSCSI target (you can use CTRL-ALT-F2) and verify that the target is or is not accessible. To return to the (graphical) installation screen, press CTRL-ALT-F7.

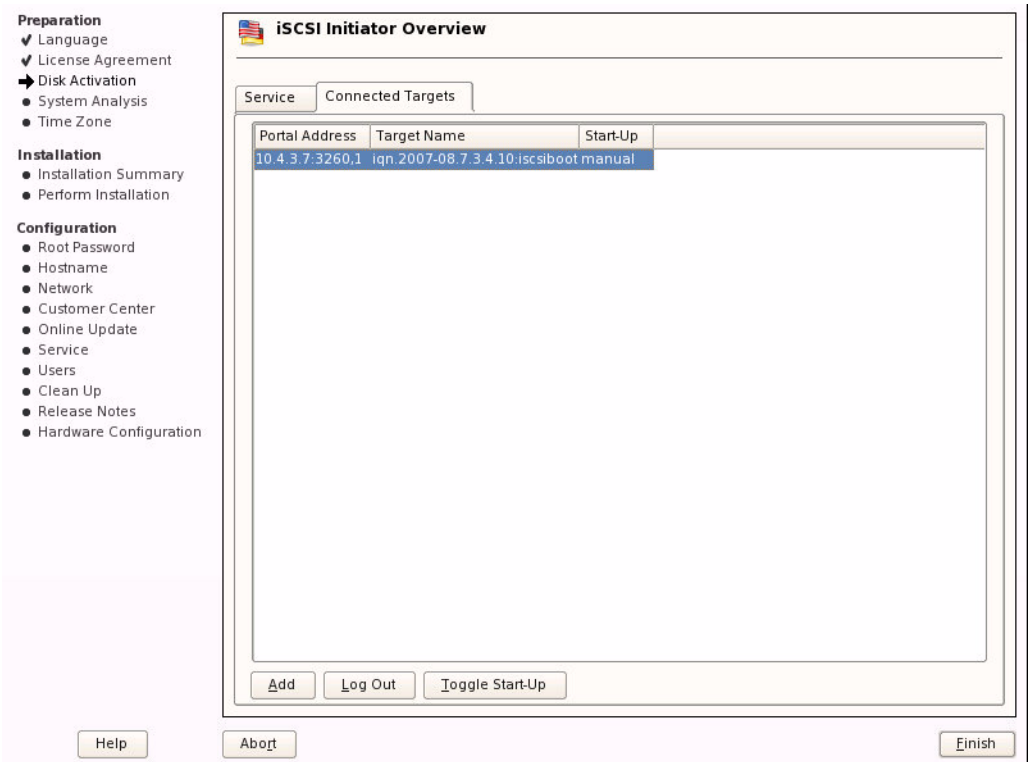
Step 5. The iSCSI Initiator Discovery window will now request authentication to access the iSCSI target. Click Next to continue without authentication unless authentication is required.



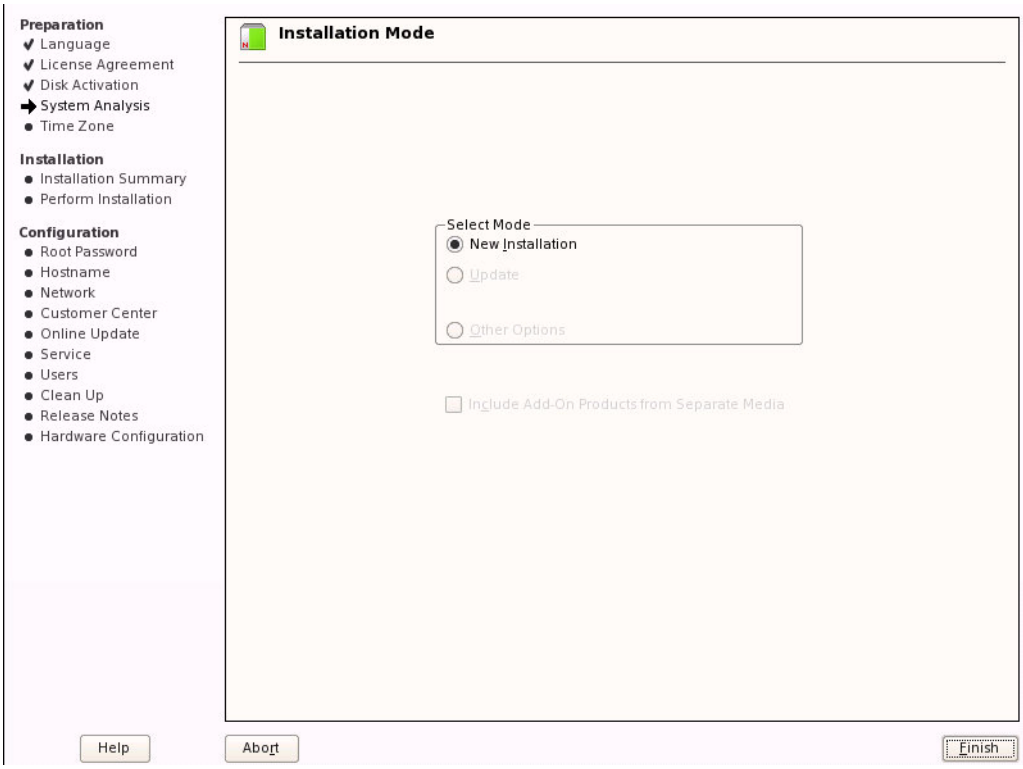
Step 6. The iSCSI Initiator Discovery window will show the iSCSI target that got connected to. Note that the Connected column must indicate True for this target. Click Next. (See figure below.)



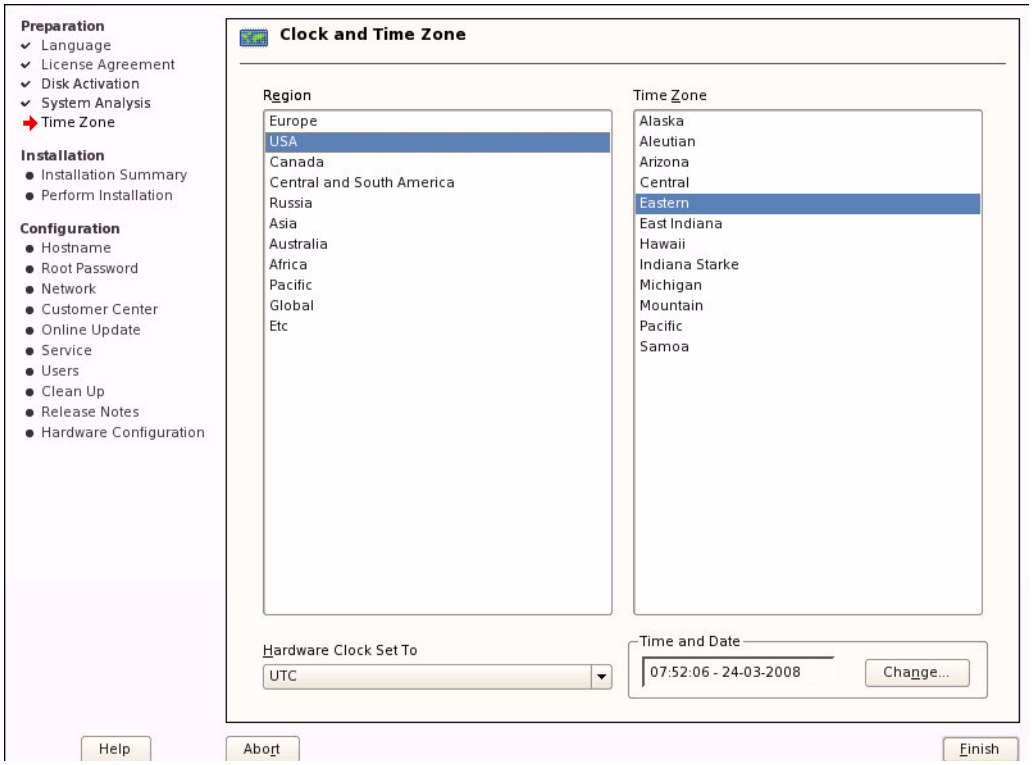
Step 7. The iSCSI Initiator Overview window will pop up. Click Toggle Start-Up to change start up from manual to automatic. Click Finish.



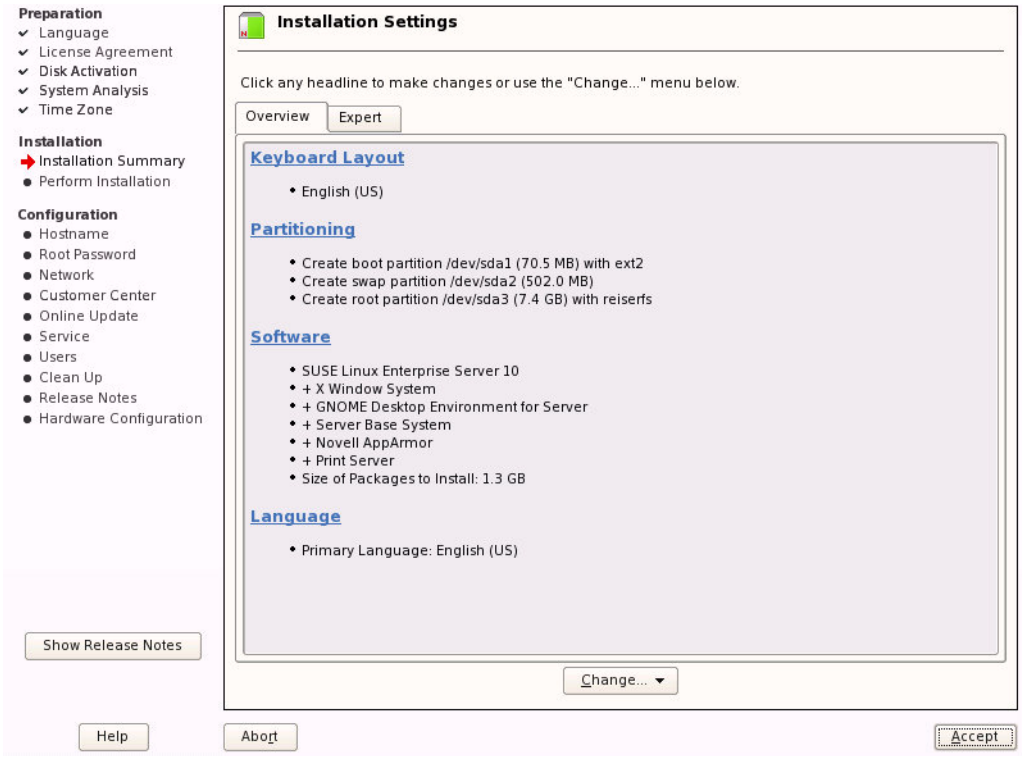
Step 8. Select New Installation then click Finish in the Installation Mode window.



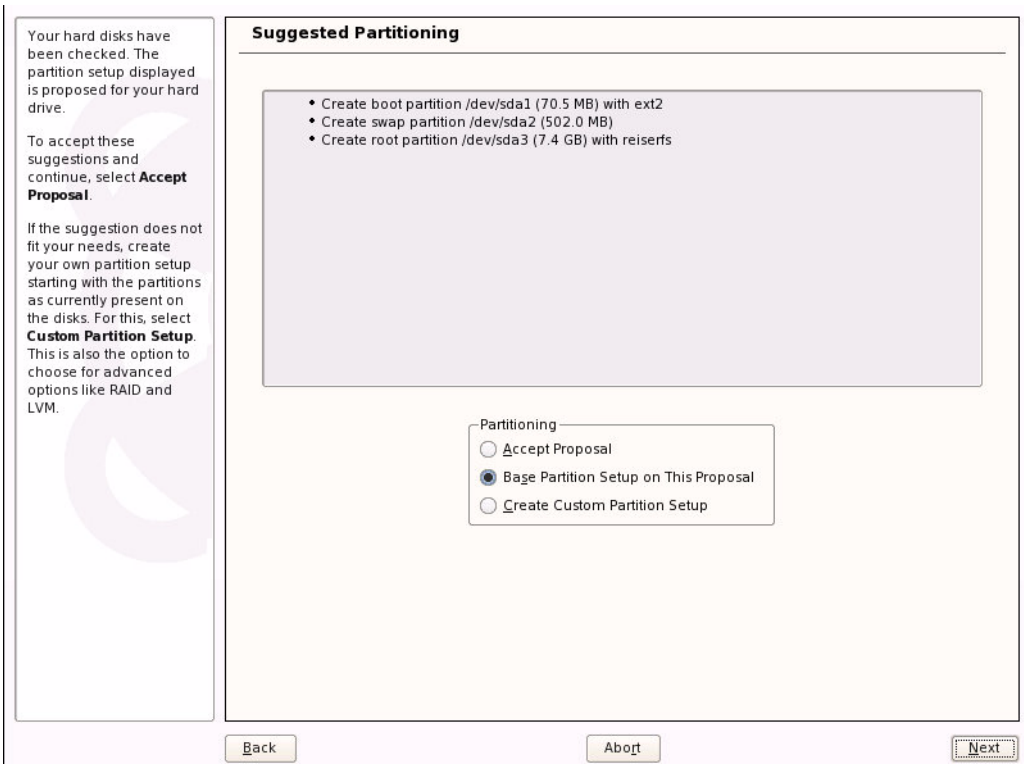
Step 9. Select the appropriate Region and Time Zone in the Clock and Time Zone window, then click Finish.



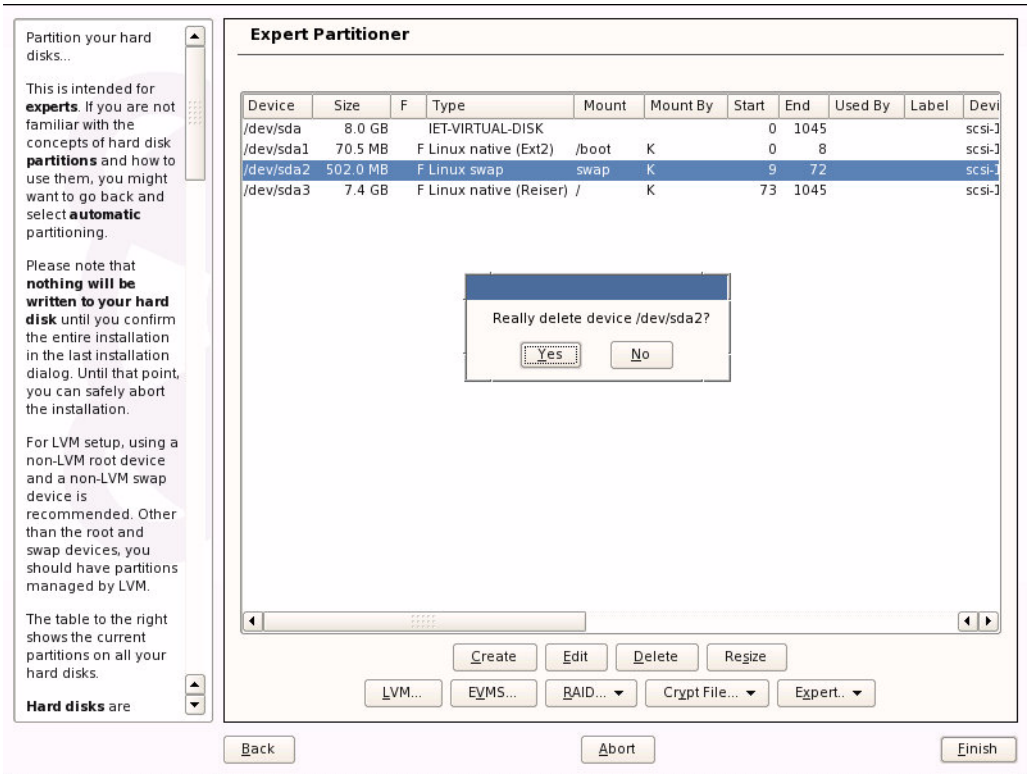
Step 10. In the Installation Settings window, click Partitioning to get the Suggested Partitioning window.



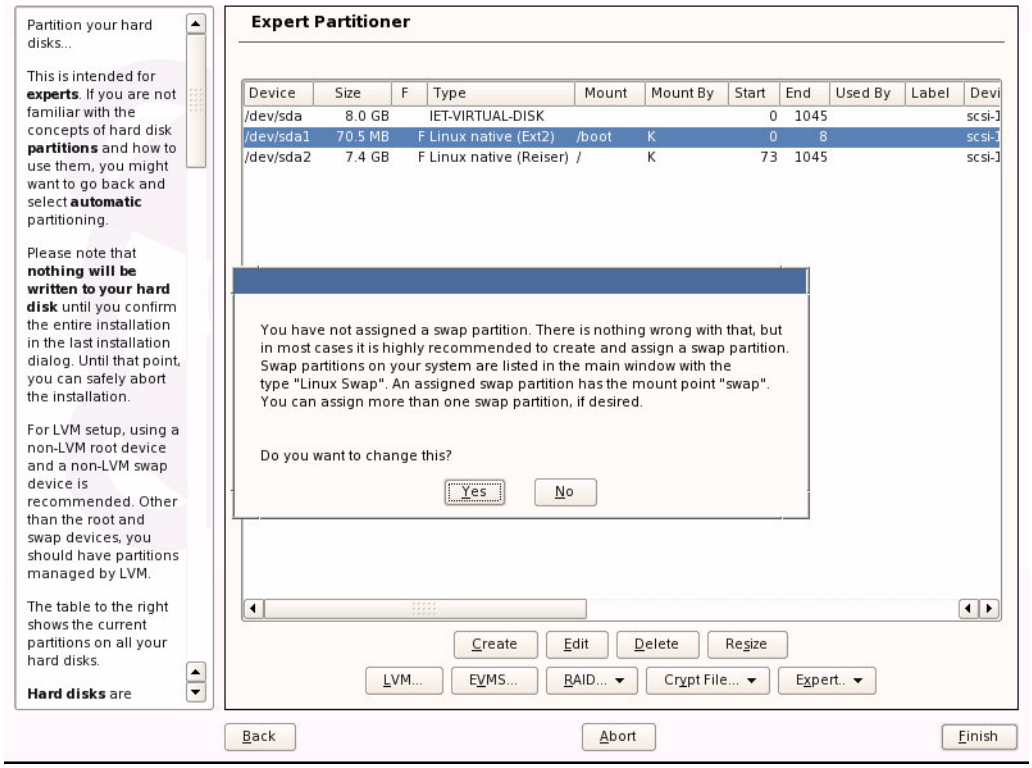
Step 11. Select Base Partition Setup on This Proposal then click Next.



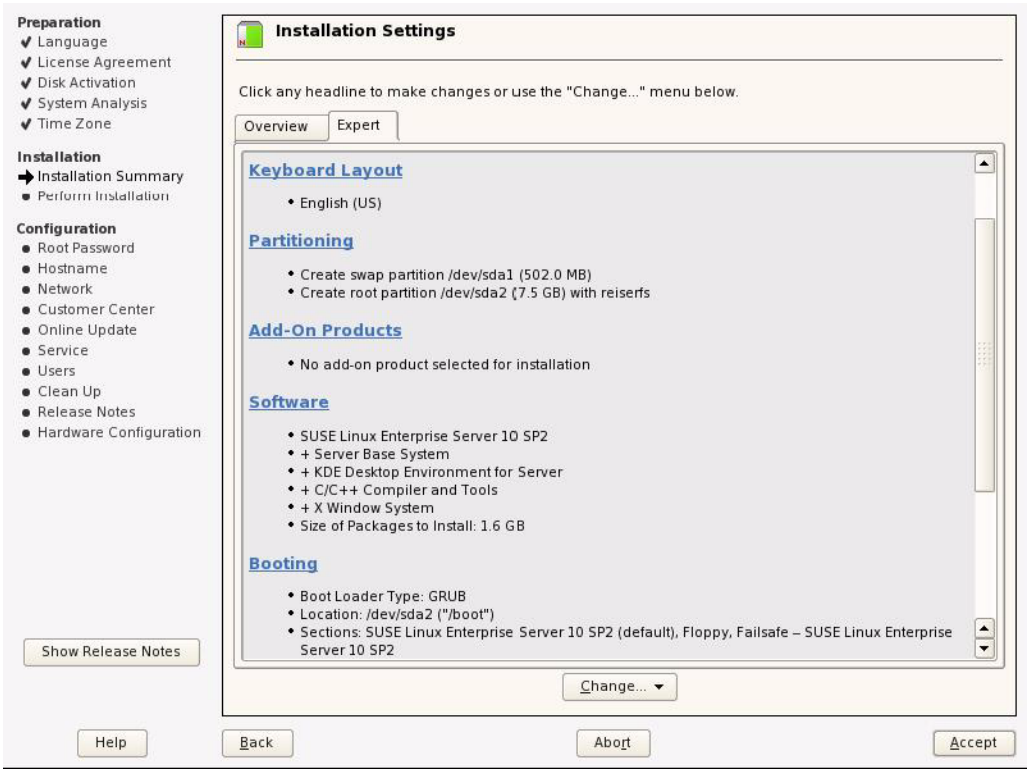
Step 12. In the Expert Partitioner window, select from the IET-VIRTUAL-DISK device the row that has its Mount column indicating ‘swap’, then click Delete. Confirm the delete operation and click Finish.



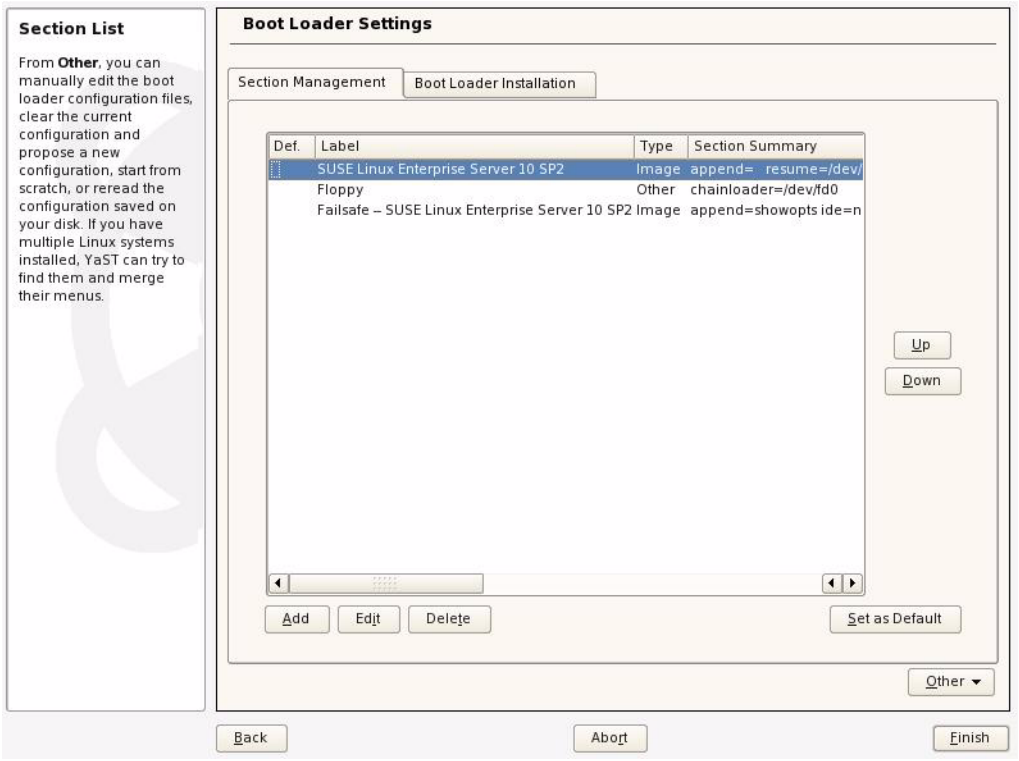
Step 13. In the pop-up window click No to approve deleting the swap partition. You will be returned to Installation Settings window. (See image below.)



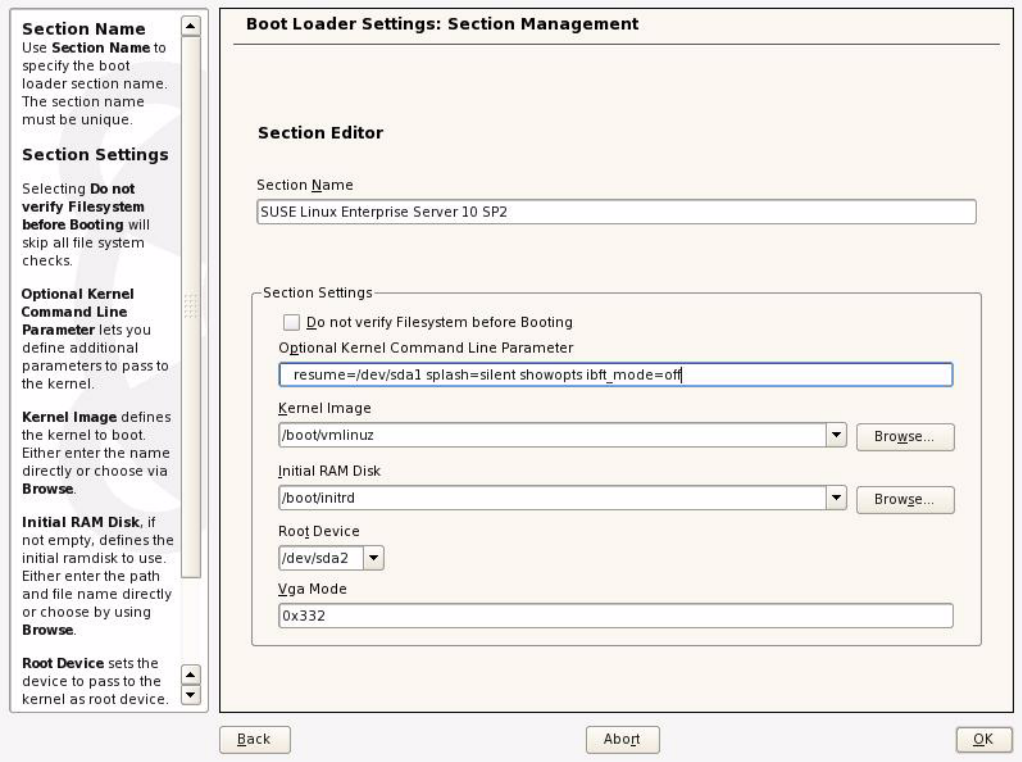
Step 14. Select the Expert tab and click Booting.



Step 15. Click Edit in the Boot Loader Settings window.

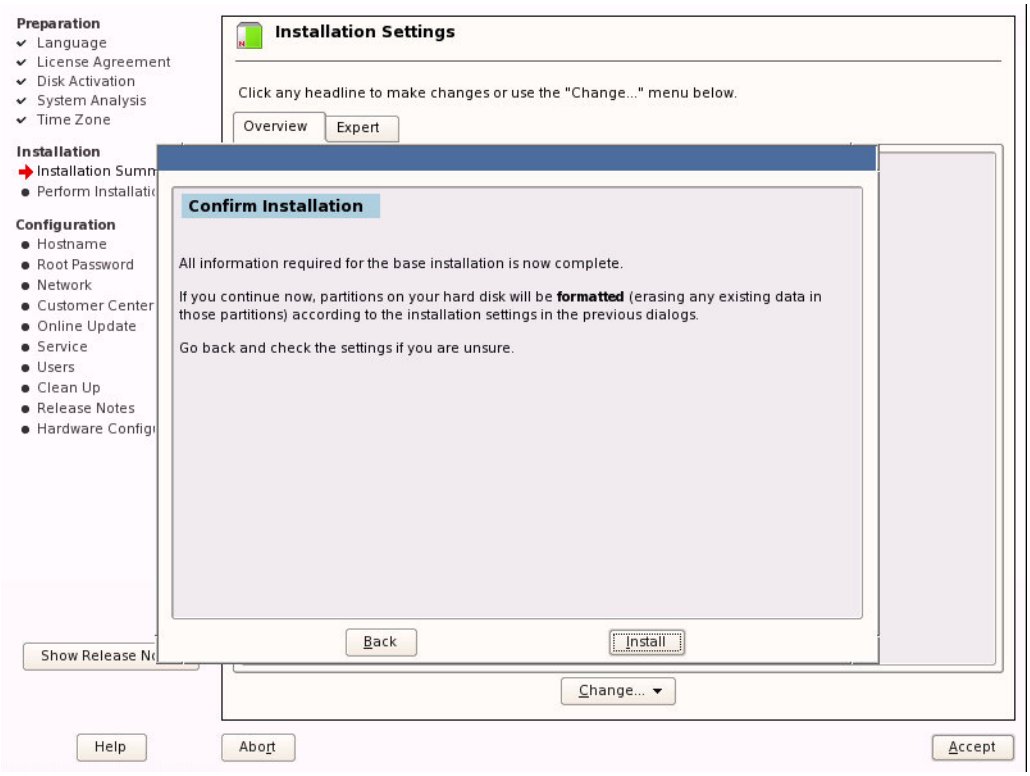


Step 16. In the Optional Kernel Command Line Parameter field, append the following string to the end of the line: "ibft_mode=off" (include a space before the string). Click OK and then Finish to apply the change.



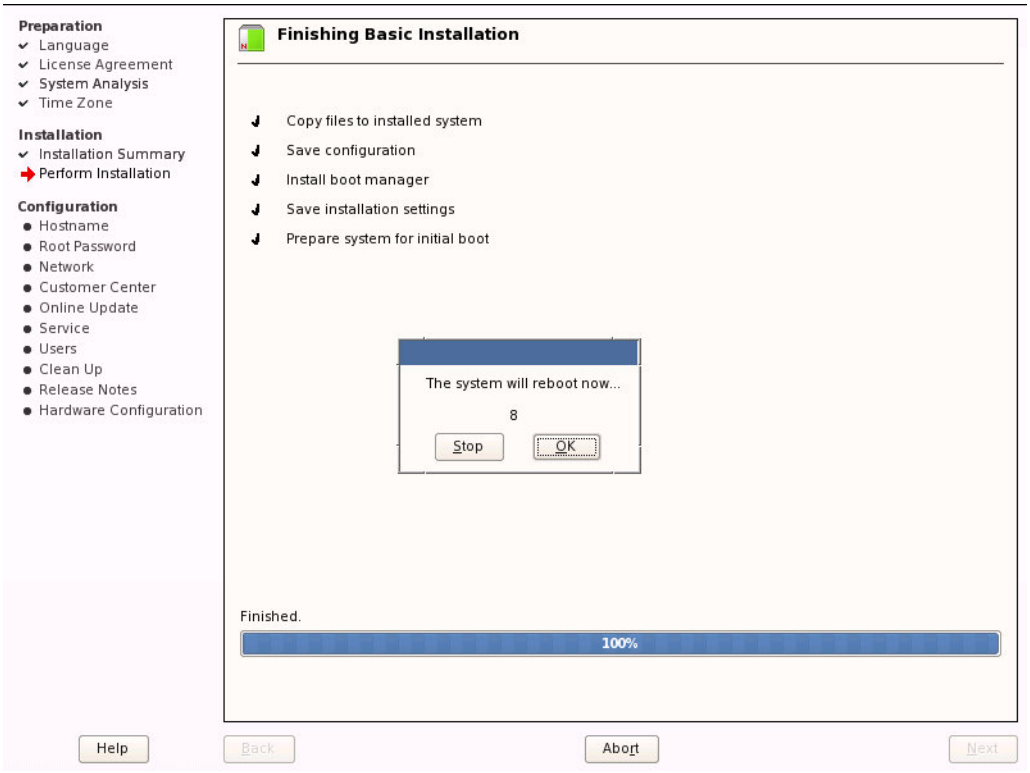
Step 17. If you wish to change additional settings, click the appropriate item and perform the changes, and click Accept when done.

Step 18. In the Confirm Installation window, click Install to start the installation. (See image below.)

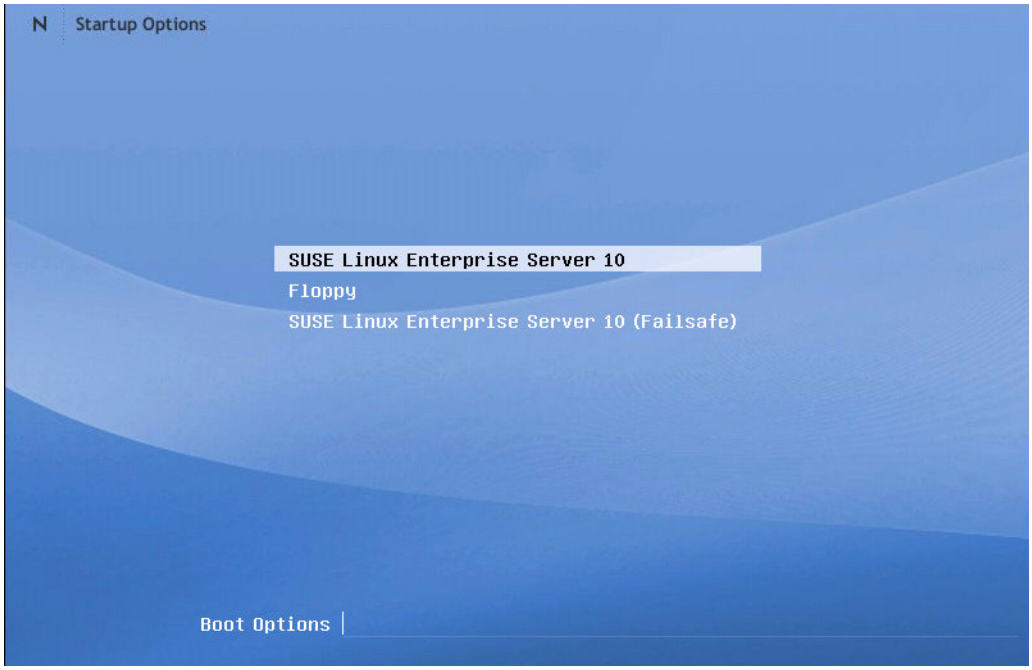


Step 19. At the end of the file copying stage, the Finishing Basic Installation window will pop up and ask for confirming a reboot. You can click OK to skip count-down. (See image below.)

Note: Assuming that the machine has been correctly configured to boot from BoIB via its connection to the iSCSI target, make sure that “MLNX IB” (for ConnectX family) or gPXE (for InfiniHost III family) has the highest priority in the BIOS boot sequence.



Step 20. Once the boot is complete, the Startup Options window will pop up. Select SUSE Linux Enterprise Server 10 SP2 then press Enter.



Step 21. The Hostname and Domain Name window will pop up. Continue configuring your machine until the operating system is up, then you can start running the machine in normal operation mode.

Step 22. (Optional) If you wish to have the second instance of connecting to the iSCSI Target go through the IB driver, copy the `initrd` file under `/boot` to a new location, add the IB driver into it after the load commands of the iSCSI Initiator modules, and continue as described in Section 8 on page 20.

Warning! Pay extra care when changing `initrd` as any mistake may prevent the client machine from booting. It is recommended to have a back-up iSCSI Initiator on a machine other than the client you are working with, to allow for debug in case `initrd` gets corrupted.

In addition, edit the `init` file (that is in the `initrd` zip) and look for the following string

```
if [ "$iSCSI_TARGET_IPADDR" ] ; then
  iscsiserver="$iSCSI_TARGET_IPADDR"
fi
```

Now add before the string the following line:

```
iSCSI_TARGET_IPADDR=<IB IP Address of iSCSI Target>
```

Example:

```
iSCSI_TARGET_IPADDR=11.4.3.7
```

10 WinPE

Mellanox BoIB enables WinPE boot via TFTP. For instructions on preparing a WinPE image, please see http://etherboot.org/wiki/winpe.