



Z H O N E™

For software version 1.15.1.208
Document Part Number: 830-01916-04
September 29, 2009

EtherXtend 3400 Series 1.15.1.208 Release Notes

This document describes the EtherXtend 3400 1.15.1.208 release and includes the following sections:

- [New Features, page 2](#)
- [Problems fixed, page 3](#)
- [Known issues, page 4](#)
- [Web GUI support for the EtherXtend 3400, page 5](#)
- [Upgrade the system software and the Web GUI http.tar file, page 6](#)
- [Create an EFM or N2Nbond group, page 8](#)
- [Configure a management interface on a WAN port, page 10](#)
- [Create EtherXtend bridges, page 12](#)
- [Back up and restore the system configuration, page 15](#)
- [Contact Global Service and Support, page 17](#)

This release note provides a summary of the new EtherXtend 3400 features using the feature default settings and related procedures.

For more detailed information and procedures, see the *EtherXtend 3400 User's Guide* or the *ZHONE CLI Reference Guide* at:

<http://zhone.com/support/manuals/>

For firmware downloads, see the Support download page at:

<http://zhone.com/support/downloads/>

Zhone Technologies
@Zhone Way
7001 Oakport Street
Oakland, CA 94621
USA

Copyright © 2008
Zhone Technologies, Inc.

The Zhone EtherXtend 4-port and 8-port devices are products in a family of next-generation Ethernet-enabled CPE devices. Each EtherXtend model performs the function of a network extender, an emerging class of products that lengthens the reach of packets passing from the source or destination to or from a service provider, a distance known as Ethernet in the First Mile (EFM). Zhone's EtherXtend focuses on providing high-availability links for business use that needs Ethernet over copper technology. By bonding two or more copper pairs, the EtherXtend provides several megabits of throughput for a fraction of the cost of a T1 or E1. The default bonding method for the EtherXtend 3400 is EFM bonding although you can also configure the EtherXtend with N2N bonding.



Note: The EtherXtend 3400 supports up to 2500 MAC addresses. Care should be taken not to exceed 2500 MAC addresses.

The firmware version for this release is 1.15.1.208.

This release uses ROM version 1.14.2.105 or later.

Zhone recommends that you leave the EtherXtend and the MALC in adaptive rate mode. If it is desired to run in a fixed rate mode, it is better to keep the EtherXtend in adaptive mode and configure the MALC in fixed rate mode.



Note: For complete information about EtherXtend operations, see the *EtherXtend Users Guide*.

NEW FEATURES

New features supported in 1.15.1.208

SNR monitoring for SHDSL

The EtherXtend 3400 provides the ability to set maximum and minimum thresholds for Signal to Noise Ratio (SNR) and retrain links if the thresholds are crossed.

Unlike ADSL2 which includes a feature that monitors SNR and seamlessly adjusts the rate (SRA — Seamless Rate Adaption) of the line to keep the current SNR at the target SNR, SHDSL does not have a comparable feature. Because traffic will not flow on SHDSL during retrain — retraining may be done on bonded links (only if there are multiple links in a bonded group, as it is possible to have a single link in a bond group) — to provide a seamless or near seamless retrain.

The EtherXtend 3400 provides the capacity to schedule retrains during a maintenance period. The SNR monitoring feature is supported on the EFM SHDSL cards.

Port access security

The EtherXtend 3400 provides the capability to provide security on the UDP/TCP ports which the EtherXtend 3400 uses for management. Configure the **port-access** profile to define the UDP/TCP port and the IP address or IP address subnet that allows access to that port.

The port access security feature is a white list mechanism. If a host's IP address is not specified in a **port-access** profile, users from that host cannot access on that port.

The ports used for management are:

- Telnet, port 23
- SSH, port 22
- HTTP, port 80
- HTTPS, port 443
- SNMP, port 61

If you restrict access to the SNMP port there must be a rule to allow the EtherXtend 3400 its own SNMP access.

By default, **port-access** profiles do not exist and all ports are open. After a **port-access** profile is configured for a port all other IP addresses or subnets are blocked. This restriction only takes effect after the first **port-access** profile is created.

Problems fixed

- TR 77735** The product photo shown in the Web UI now shows a EtherXtend 3400 instead of a EtherXtend 2100.
- TR 77736** The Web UI does not allow configuration of uplink bridges on Ethernet ports and does not allow downlink bridges on the SHDSL ports in CO mode.
- TR 76967** After a **set2default**, the default EFM-SHDSL bond group does not autoconfigure.
- TR 77968** Multiple N2N bond groups not supported. When the ETHX is operating in CO mode only one N2N bond group is supported. Multiple N2N bond groups are not supported.
- ◇ Logical port context has length 0 causing upstream latency on bond groups (intermittent).
 - ◇ In rare occurrences, all SHDSL bond group traffic will stop.

Known issues

This section describes known issues in this release. Known Issues with an asterisk (*) are new to this release.

During configuration, system messages may be reported. These messages may be useful for troubleshooting but should not affect operation of the system.

General

TR 79801

*When using the Web UI it is not possible to create two bridge interfaces on a EtherXtend WAN port. When using the CLI, it is possible to create two WAN bridge interfaces to configure the unit.

- ◇ ***snrmon modify mode** commands do not match the help description. The **help** command lists the choices as *automatic-once*, *automatic-daily*, *automatic-continuous*. The actual command choices are *auto-once*, *auto-daily*, *auto-continuous*.
- ◇ *When secure mode is enabled, HTTP management is still allowed.
- ◇ *The Web UI always shows zero bridge packet rate statistics. The CLI shows the correct information.
- ◇ *The Web UI does not show the correct EFM-SHDSL bond group statistics. The CLI shows the correct information.
- ◇ *The Web UI does not show the correct Ethernet port statistics. The CLI shows the correct information.
- ◇ Enabling the port status mirror feature causes the Ethernet port to bounce. When first configuring and enabling the port status mirror feature, the Ethernet port will bounce. The Ethernet port comes back immediately and operates as expected after the initial bounce.*
- ◇ When the **bond show** command is entered after performing a eth-oam loopback, the following error appears:
error: 1/1/1027: clitask0:
CliIfIndexToPortAddressGet: IfCfgMsgTypeGet failed.
The output from the **bond show** command is still displayed.
- ◇ After performing an onreboot restore, a dhcpclient error message appears.
- ◇ When adding a default route, a dhcpclient error message appears.

Web GUI support for the EtherXtend 3400

The EtherXtend 3400 provides a Web User Interface that you can use to configure, test, and administer the device.

The default login is *admin* and the default password is *zhone*.

To manage the EtherXtend 3400 using the Zhone Web User Interface:

- Check to see what IP address the DHCP server gives the EtherXtend as a DHCP client.
- Configured IP interface

To access the device through the Web Interface Tool through an IP address that is different from the default IP address/subnet, connect directly to the device using the console port or an accessible Ethernet switch/hub.

Delete the default autoconfig IP address and add the desired IP address. If the new IP address is not compatible with the address of the management PC, the connection to the device will be lost. Change the address of the management PC to be compatible with the device address.

Using the Web Interface Tool, configure a default route to the management PC. A default route must be created before the device is accessible across subnets outside the default subnet.

```
zSH> delete ip-interface-record AutoConfig/ip
zSH> interface add 1-1-1-0/eth vlan 94 172.24.94.103/24
Created ip-interface-record ethernet2-94/ip
zSH> route add default 172.24.94.254 1
```

From a PC connected to the 10/100 on the EtherXtend 3400, ensure the IP address on the PC is in the same subnet as the EtherXtend 3400 IP address.

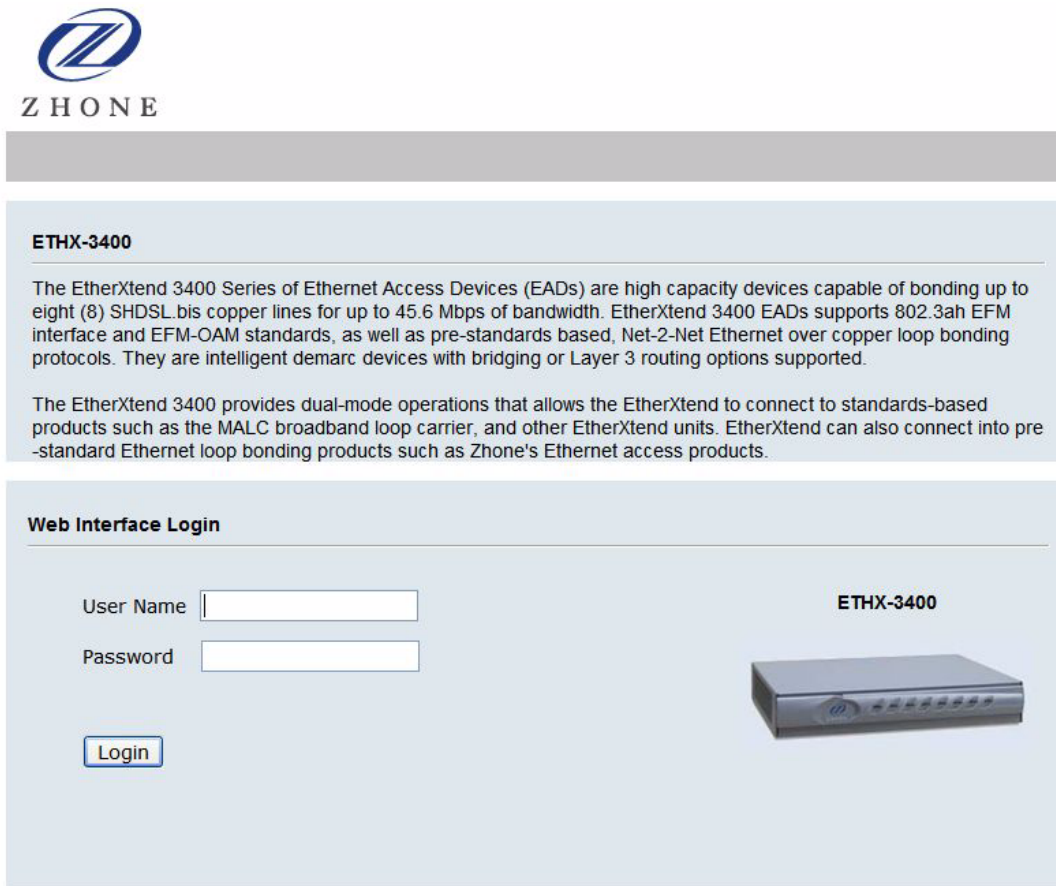
To launch the Zhone Web User Interface, in a browser URL address space on the PC enter the IP address configured on the EtherXtend 3400.

The Zhone Web User Interface launches and displays the Login window for the EtherXtend 3400.




Note: The Web UI supports only Internet Explorer and Mozilla browsers.

Figure 1: Zhone Web User Interface Login Screen



On the Login page, enter the user name and password. The default user name is *admin* and the default password is *zhone*.

Click the desired menu to display the management options. For online help, click the Help icon  or product title in any window.



Note: The **del** command can be used to delete all of the Zhone Web User Interface files if needed.

Upgrade the system software and the Web GUI http.tar file

This section explains how to upgrade the EtherXtend 3400 software and the Web GUI *http.tar* file.

When upgrading to a new software version, the *http.tar* files for the Web GUI must be upgraded along with the system software by placing the most recent *http.tar* file on the flash of the uplink card in the */card1* directory.



Note: The *http.tar* files and the binary files for the upgraded software release must match.

Upgrading the Web GUI *http.tar* file

- 1 Verify that you are at the root of the flash of the uplink card:

```
zSH> cd /card1
zSH> pwd
/card1
```

- 2 Download the latest Web GUI *http.tar* file from a TFTP server using the **file download** command.

```
zSH> file download TFTPserverAddress sourcefilename
destinationfilename
```

For example:

```
zSH> file download TFTPserverAddress ethx3400_http.tar
ethx3400_http.tar
```



Note: If a download error occurs check for and delete unnecessary files to regain flash space.

Upgrading the system software



Note: Zhone strongly recommends that you back-up your configuration to a file with the **dump** command before upgrading software.

- 1 Verify you are at the root of the flash card:

```
zSH> cd /card1
zSH> pwd
/card1/
```

- 2 Save the configuration file either locally or remotely to a host on the network using the **dump** command.

Use the **dump file** command with the following syntax to save the configuration file locally:

```
dump file filename
```

```
zSH> dump file ethx3400.cfg.date
```

Use the **dump network** command with the following syntax to save the configuration file to a remote host:

```
dump network host filename
```

```
zSH> dump network 192.168.8.21 ethx3400.cfg
```

- 3 Save the existing software images on the flash card by copying them to a different filename. For example:

```
zSH> copy ethx3400.bin ethx3400.bin.<ver>
```

Copy the rest of the software images in the same way.

- 4 Download the latest system image software for the device to the system flash using the **image** command. The command uses the following syntax:

```
zSH> image download 192.168.2.89 sourcefilename  
[destinationfilename]
```



Note: When using the **image download** command, if you do not specify a *destinationfilename*, it will be the same as the *sourcefilename*.

For example:

```
zSH> image download 192.168.2.89 ethx3400.bin
```



Note: If a download error occurs check for and delete unnecessary files to regain flash space.

- 5 Use the **systemreboot** command to reboot the system with the new configuration.

```
zSH> systemreboot
```

The system reboots again with the new configuration. This may take up to 5 minutes to complete.

Create an EFM or N2Nbond group

Creating bond groups

Add a single N2N or EFM bond group and verify bond group.



Note: Bond groups created with CLI commands must be greater than 24 and less than 100. 100-series bond group IDs are used by ZMS and 200-series bond groups are auto-provisioned/discovered.

```
zSH> bond add group 1-1-50-0/n2nbond
```

```
zSH> bond add group 1-1-50-0/efmbond
```


To add a new member to an existing bond group:

```
zSH> bond add member 1-1-101-0/n2nbond 1-1-1-0/shdsl
```

To create a bond group with multiple members:

```
zSH> bond add member 1-1-50-0/n2nbond 1-1-3-0/shdsl
1-1-4-0/shdsl
```

Displaying bond groups

Bond groups can be displayed for all existing groups, a specific group, a specific slot, or link.

To display all configured bond groups:

```
zSH> bond show all
```

Bond Groups				
Slot	GrpId	Name	Type	State
1	99	1-1-99-0	efmbond	ACT

To display a specific bond group:

```
zSH> bond show group 1-1-99-0/efmbond
```

Bond Groups				
Slot	GrpId	Name	Type	State
1	99	1-1-99-0	efmbond	ACT
Group Members				
Slot	Port	Name	Type	State
1	3	1-1-3-0	shdsl	ACT
1	2	1-1-2-0	shdsl	ACT
1	1	1-1-1-0	shdsl	ACT
1	8	1-1-8-0	shdsl	ACT
1	7	1-1-7-0	shdsl	ACT
1	6	1-1-6-0	shdsl	ACT
1	5	1-1-5-0	shdsl	ACT
1	4	1-1-4-0	shdsl	ACT

To display bond group on slot 1:

```
zSH> bond show slot 1
```

Bond Groups				
Slot	GrpId	Name	Type	State
1	99	1-1-99-0	efmbond	ACT

To display the bond link:

```
zSH> bond show link 1-1-3-0/shdsl
```

Bond Groups				
Slot	GrpId	Name	Type	State
1	99	1-1-99-0	efmbond	ACT
Group Members				
Slot	Port	Name	Type	State
1	3	1-1-3-0	shdsl	ACT
1	2	1-1-2-0	shdsl	ACT

1	1	1-1-1-0	shdsl	ACT
1	8	1-1-8-0	shdsl	ACT
1	7	1-1-7-0	shdsl	ACT
1	6	1-1-6-0	shdsl	ACT
1	5	1-1-5-0	shdsl	ACT
1	4	1-1-4-0	shdsl	ACT

Changing bond group type

Bond group type can be changed for individual bond groups or all bond groups used in a specified slot using **bond modify** command. Bond groups and bond members can be deleted with the **bond delete** command.

```
zSH> bond modify efm group 1-1-201-0/n2nbond
zSH> bond modify n2n group 1-1-101-0/efmbond
zSH> bond modify efm slot 1
zSH> bond delete group 1-1-201/n2nbond
zSH> bond delete member 1-1-201-0/n2nbond 1-1-3-0/shdsl
```

Configure a management interface on a WAN port

This section describes how to use the default management interface to remotely manage the EtherXtend 3400 and how to modify the default interface for management from a different subnetwork. By default, the AutoConfig interface is 1-1-99-0/efmbond and is built on the EFM bond group as a DHCP client. The bond group number may vary depending on which port is first to receive link.

Managing EtherXtend using the default management interface

The AutoConfig interface, 1-1-99-0/efmbond, is built on the default EFM bond group as a DHCP client using VLAN 7.

The MALC that connects to the EtherXtend can be configured either as a local DHCP server or as bridge to an external DHCP server. Refer to the MALC documentation for detailed procedures on DHCP configuration.

To configure Telnet access from the MALC to the EtherXtend, use the **host add** command on the MALC to configure a host connection between the MALC and the EtherXtend and the bond group associated with the management interface.

```
zSH> host add 1-1-99-0/efmbond vlan 7 dynamic 1 1
```

You can now Telnet to the MALC and then Telnet to the EtherXtend device.

Use the **interface show** and **route show** commands on the EtherXtend to verify the default management interface:

```
zSH> route show
Destination Routing Table
Dest                Nexthop                Cost    Owner    Fallback
-----
0.0.0.0/0           10.250.1.254          1       STATICLOW
10.250.1.0/24       1/1/99/0/ip           1       LOCAL
```

```
zSH> interface show
1 interface
Interface          Status  Rd/Address                Media/Dest Address    IfName
-----
1/1/99/0/ip       UP      1 10.250.1.54/24          00:01:47:36:2e:03     AutoConfig
```

Managing EtherXtend using a non-default management interface

To manage the EtherXtend 3400 through an interface different than the default AutoConfig address, delete the default AutoConfigIP address and then add the desired interface. If the new IP address is not compatible with the address of the management PC, the connection to the device will be lost. To restore the connection, change the address of the management PC to be compatible with the device address.

The following example configures the IP address for the system:

- 1 Delete the AutoConfig address.

```
zSH> delete ip-interface-record AutoConfig/ip
```

- 2 Add the IP interface to the bond group.

```
zSH> interface add 1-1-50-0/n2nbond 172.24.200.133/24
Created ip-interface-record 1-1-(25-99)-0-n2nbond/ip
```



Note: The default interface is reset if a **set2default** is performed without the restore option.

Verifying the interface

Use the **interface show** command to verify that the IP interface was configured correctly:

```
zSH> interface show
1 interface
Interface          Status  Rd/Address                Media/Dest Address    IfName
-----
1/1/4/0/ip         UP      1 172.24.200.133/24          00:01:47:07:ef:ee     AutoConfig
```

Creating a default route

The following example creates a default route using the gateway 172.24.200.254 with a cost of 1 (one):

```
zSH> route add default 172.24.200.254 1
```

Verifying the route

Use the **route show** command to verify that the routes were added:

```
zSH> route show
Dest                               Nexthop           Cost   Owner
-----
0.0.0.0/0                           192.168.8.1      1      STATICLOW
192.168.8.1/24                       1/1/1/0/ip       1      LOCAL
```

Create EtherXtend bridges

This section describes how to create bridges for one or more Ethernet interfaces on the EtherXtend.

EtherXtend auto-provisioning

Release 1.15.1.208 has an auto-provisioning feature that automatically re-configures the EtherXtend bridges and management IP address when the 1.15.1.208 software runs for the first time.

The auto-provisioning deletes any existing bridges and creates tagged (VLAN 0) and untagged TLS bridges on each of the LAN Ethernet ports and on the SHDSL bond group.

The auto-provisioning also deletes any existing IP addresses on the Ethernet ports or SHDSL bond group and creates a DHCP client on VLAN 7 for the SHDSL bond group. The MALC will need to be configured as a DHCP server and a host will need to be added for each bond group, for example:

```
host add 1-1-99-0/efmbond vlan 7 dynamic 1 1
```

The auto-provisioning runs on the EtherXtend only after upgrading from a version of the EtherXtend software that did not support auto-provisioning to a version that does support auto-provisioning, or following a **set2default**.

```
zSH> imCreated default bridges
Created interface AutoConfig/ip
Created default OAM interface
Auto-provisioning is complete.
```

```
zSH> bridge show
```

Type	VLAN	Bridge	St	Table	Data
tls	7	1-1-99-0-efmbond/bridge	UP		
tls	Tagged	1-1-99-0-efmbond-0/bridge	UP		
tls	7	1-1-1-0-eth/bridge	DWN		

```

tls Tagged      1-1-1-0-eth-0/bridge      DWN
tls             7 1-1-4-0-eth/bridge      DWN
tls Tagged      1-1-4-0-eth-0/bridge      DWN
tls             7 1-1-3-0-eth/bridge      DWN
tls Tagged      1-1-3-0-eth-0/bridge      DWN
tls             7 1-1-2-0-eth/bridge      DWN
tls Tagged      1-1-2-0-eth-0/bridge      DWN
zSH>

```

```
zSH> interface show
```

```

1 interface
Interface      Status  Rd/Address          Media/Dest Address  IfName
-----
1/1/99/0/ip    UP      1 0.0.0.0/0         00:01:47:08:a4:32   AutoConfig
-----
zSH>

```

To view the VLAN ID and DHCP client configuration, enter:

```

zSH> get ip-interface-record 1/1/99/0/ip
ip-interface-record 1/1/99/0/ip
vpi: -----> {0}
vci: -----> {0}
rdindex: -----> {1}
dhcp: -----> {client}
addr: -----> {10.250.1.89}
netmask: -----> {255.255.255.0}
bcastaddr: -----> {10.250.1.255}
destaddr: -----> {0.0.0.0}
farendaddr: -----> {0.0.0.0}
mru: -----> {1500}
reasmmaxsize: -----> {0}
ingressfiltername: -----> {}
egressfiltername: -----> {}
pointtopoint: -----> {no}
mcastenabled: -----> {yes}
ipfwdenabled: -----> {yes}
mcastfwdenabled: -----> {yes}
natenabled: -----> {no}
bcastenabled: -----> {yes}
ingressPacketRuleGroupIndex: --> {0}
egressPacketRuleGroupIndex: --> {0}
ipaddrdynamic: -----> {dhcpclient}
dhcpserverenable: -----> {false}
subnetgroup: -----> {0}
unnumberedindex: -----> {0}
mcastcontrollist: -----> {}
vlanid: -----> {7}
maxVideoStreams: -----> {0}
tosOption: -----> {disable}
tosCOS: -----> {0}
vlanCOS: -----> {0}
s-tagTPID: -----> {0x8100}
s-tagId: -----> {0}

```

```
s-tagIdCOS: -----> {0}
```

Creating a bridge for a bond group

Create a bridge for the bond group.

- 1 Use the **bridge add** command to add a bridge, then create the bridge path with the **bridge-path add** command as follows

```
zSH> bridge add 1-1-201-0/n2nbond downlink vlan 1
Adding bridge on 1-1-201-0/n2nbond
Created bridge-interface-record 1-1-201-0-n2nbond/
bridge
```

```
zSH> bridge-path add 1-1-202-0-n2nbond/bridge vlan 1
default
Bridge-path added successfully
```

- 2 Use the **bridge add** command to create a bridge for one or more Ethernet interfaces on the EtherXtend, then use the **bridge-path add** command to create the bridge path for that interface:

```
zSH> bridge add 1-1-1-0/eth downlink vlan 1
Adding bridge on 1-1-1-0/eth
Created bridge-interface-record 1-1-1-0-eth/bridge
```

```
zSH> bridge-path add 1-1-1-0-eth/bridge vlan 1 default
Bridge-path added successfully
```

- 3 Use the **bridge show** command to view the bridges.

```

zSH> bridge show
Type VLAN          Bridge                               St  Table Data
-----
dwn                1 1-1-201-0-n2nbond/bridge          UP  D 00:13:72:dd:51:89
                                   S VLAN 1 default [U: 3600
sec, M: 150 sec, I: 0 sec]
dwn                1 1-1-1-0-eth/bridge                UP  D 00:13:72:de:92:2e
                                   D 00:1a:6d:13:19:83

```

Creating an s-tagged uplink bridge

Follow these steps to create an s-tagged (double tagged) uplink bridge.

- 1 To configure an s-tagged uplink bridge enter **bridge add interface/type uplink tagged**.

```

zSH> bridge add 1-1-2-0/eth uplink tagged
Adding bridge on 1-1-2-0/eth

```

- 2 To create a bridge path for the uplink bridge, enter **bridge-path add interface/type global**.

```

zSH> bridge-path add ethernet2-0-0/bridge global
Bridge-path added successfully

```

- 3 To create a downlink bridge for both the VLAN and the SLAN, enter **bridge add interface/type VLAN 200 SLAN 400**.

```

zSH> bridge add 1-13-201-0/n2nbond downlink vlan 200
slan 400
Adding bridge on 1-13-201-0/n2nbond

```

```

Created bridge-interface-record 1-13-201-0-n2nbond/
bridge

```

- 4 To verify the bridges just created, enter **bridge show**.

```

zSH> bridge show
Type VLAN          Bridge                               St  Table Data
-----
dwn  200/400       1-13-201-0-n2nbond/bridge          PND
upl  STagged      ethernet2-0-0/bridge              UP  S Global default [U:3600 sec M:
150 sec, I: 0 sec]
zSH>

```

Back up and restore the system configuration

This section explains how to backup and restore a EtherXtend 3400 system configuration.

Backing up and restoring the configuration

You can back up your configuration using the **dump** command either locally or to the network.

To save the configuration locally, use the following syntax::

```
dump file filename
```

To save the configuration to the network use the following syntax:

```
dump network host filename
```

To back up and restore the configuration to and from the network:

- 1 Create the file in the destination directory of the TFTP server and make it writable (for UNIX and Sun servers).

- 2 Save the existing configuration. The following example saves the configuration to a file named *ethx3400.cfg* on the host 192.168.8.21:

```
zSH> dump network 192.168.8.21 ethx3400.cfg
```

- 3 On the EtherXtend 3400 flash card, **cd** to the *onreboot* directory:

```
zSH> cd /card1
zSH> cd onreboot
```

- 4 Copy the previously saved dump file into the **onreboot** directory and rename it **restore**:

```
zSH> file download 192.168.2.89 ethx3400.cfg restore
```

where **192.168.2.89** is the hostname where the configuration file was saved, *ethx3400.cfg* is the existing configuration file, and *restore* is the renamed filename.

The *restore* file can be validated using the **restore validate <filename>** command.

- 5 Use **set2default** to erase the system configuration and perform a reboot running the 1.15.1.208 software:

```
zSH> set2default
```

```
Ok to reset to default (system will reboot) ? [yes] or [no]: yes
```



Note: Traffic and in-band management will be down until the EtherXtend bridging and IP interfaces are manually reconfigured.

During the reboot, the system will look for the file named **restore** in the **onreboot** directory, then use that file to recreate the configuration files in the **datastor** directory. After the restore is finished, the system renames the **restore** file to **restore.done**. A system warning appears if a *restore* file is not found.

Contact Global Service and Support

Contact Global Service and Support (GSS) if you have any questions about this or other Zhone products. Before contacting GSS, make sure you have the following information:

- Zhone product you are using
- System configuration
- Software version running on the system
- Description of the issue

Technical support

If you require assistance with the installation or operation of your product, or if you want to return a product for repair under warranty, contact GSS. The contact information is as follows:

E-mail	support@zhone.com
Telephone (North America)	877-ZHONE20
Telephone (International)	510-777-7133
Internet	www.zhone.com/support

If you purchased the product from an authorized dealer, distributor, Value Added Reseller (VAR), or third party, contact that supplier for technical assistance and warranty support.

Service requirements

If the product malfunctions, all repairs must be performed by the manufacturer or a Zhone-authorized agent. It is the responsibility of users requiring service to report the need for service to GSS.

