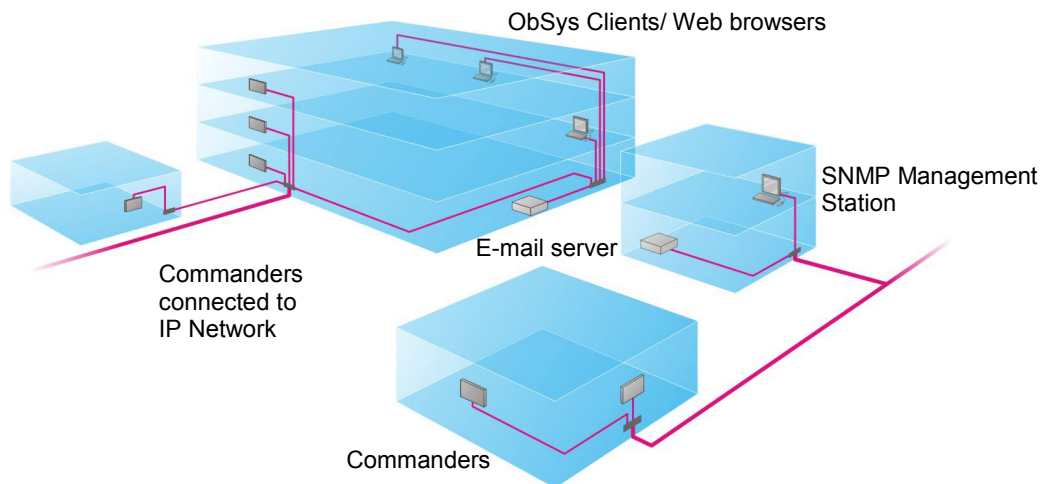


Introduction to Commander Alarms

Introduction

Commander is a powerful IP-based controller providing a configurable link between an Ethernet network and the connected system. This document describes the operation and configuration of the alarm event handling modules within Commander.

The alarm handling capabilities of Commander include alarm distribution and routing, event history, SNMP trap notification, e-mail notification, and the monitoring of values.



Contents

[Alarm Overview](#) – description of an alarm event and the alarm modules available within Commander

[Generating Alarm Events](#) – sending alarms to Commander and monitoring values for an alarm condition

[Alarm Router](#) – configure the distribution and routing of alarm events

[Alarm History](#) – stores the last 100 alarms for viewing

[Alarm E-mail](#) – reformats and sends alarm events to an e-mail recipient

[SNMP Trap Events](#) – reports alarm states as SNMP traps to a management station

Further Information

For general engineering information about Commander and details of the interface, please refer to the Engineering Guide for the Commander interface installed.

For information relating to the Network connectivity of Commander including network configuration, IPBus, the user database, Web server, BACnet device, and authentication server, refer to the [Introduction to Commander Networking](#).

For information relating to the control capabilities of Commander including programmable control, on-off time control, and data logging refer, to the [Introduction to Commander Control](#).

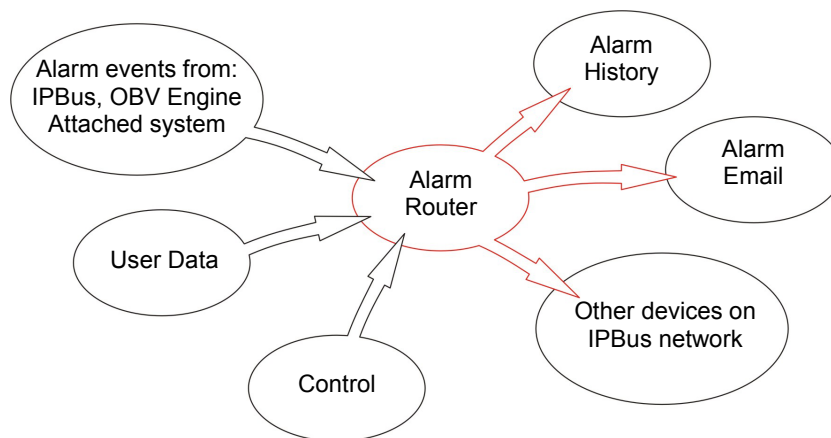
Alarm Overview

Alarm events may be sent to notify a user with information about the condition of equipment on site. This information may indicate a smoke sensor detecting a fire, a door opening, a temperature sensor detecting a high temperature, or a user logging on or off.

The range of alarm events generated depends on what equipment is installed and how it has been engineered. An alarm event always contains the following information:

- Point name – a description of the point generating the alarm, e.g. 'Building 4 Room 2E5'
- Condition – the state of the point generating the alarm, e.g. 'Fire'
- Priority – an indication of how important the alarm is. A number in the range 1 to 9, where 1 is most important
- Date detected – the date and time the alarm was detected by the system.

Commander contains several modules to assist in the generation, distribution, routing, and notification of alarm events.



Alarm events generated from the connected system, Commander, and devices on the IPBus network, may be sent to the Commander's Alarm Route module for distribution.

If a system is unable to generate its own alarm events, the User Data module can be used to monitor a value and then generate an alarm event. These alarms are automatically sent to the Alarm Route module.

The Alarm Route module sends alarms to other alarm objects within Commander, the connected system, and on the IPBus network. The destinations within the Alarm Route module can be configured to distribute and route alarms depending on selectable criteria, such as text comparison of the alarm text, and alarm priority. Destinations may also be enabled and disabled using the Time Controller or OBVEngine modules.

Commander contains modules to assist with the notification of alarm events. The Alarm History module contains a list of the last 100 events received, and may be viewed from the Web Server and XOM compatible products such as ObView. The Alarm Email module sends alarm events to an e-mail recipient.

In addition to these alarm events, Commander can also be configured to report SNMP trap events to a management station trap handler using the Simple Network Management Protocol (SNMP). The SNMP Trap module contains a list of conditions which when triggered send a corresponding trap message on the Ethernet network.

Generating Alarm Events

Commander can have alarm events passed to it or, if a system is unable to generate its own alarms, generate alarm events by monitoring values in the User Data module.

Alarm events generated from the connected system, Commander, and devices on the IPBus network may be sent to the Commander's Alarm Route object 'ALARM' for distribution.

User Database

Commander can collect values from objects in the connected system and other IPBus devices with the User Data module. It acts as a database of summary information that is aimed at users rather than engineers. Users can view and set the information in the database in several ways.

Collected values can be checked against acceptable levels, and alarms generated when values are out-of-range or communications fail.

The User Data Setup module contains 20 pages, with each page containing 16 configurable objects. Use object engineering software, such as ObView, to configure the User Data Setup module.

Each of the 16 objects within the page should be configured to store a value using the following parameters:

- Reference – a short reference, unique within the page
- Label – a brief label describing the object
- Type – specifies the type of value stored by the object: text string; integer; floating point number; on-off/no-yes value; on-off times; or a date & time value
- Adjustable – enables adjustment of the value by a user
- Access Level – controls user access when adjusting the value
- Value High/Low limit – the typical operating range of the value. These limits are used when a user adjusts the value, and to generate an alarm message when the value exceeds this range
- Remote Action – an option to read a remote object from the connected system, or write the value provided by a user to the connected system
- Remote Object – an object reference to read or write. The object may be from within the Commander, or accessible via the connected system or IPBus network
- Remote Rate – the frequency in which the Remote Object should be read
- Remote Fails – the number of consecutive failed attempts at reading/writing the remote object (used for troubleshooting)
- Alarm Priority – enables the sending of an alarm message if the value is outside its high/low limits or Remote Fails exceeds its threshold
- Alarm State – the current alarm state of the value.

To enable the generation of alarm event messages from User Data, configure the Value High/Low limits and an Alarm Priority. All alarms generated are then automatically sent to the Alarm Route module.

For further information on the User Data module, refer to the to the [*Introduction to Commander Networking*](#).

Alarm Router

The Alarm Route module is used to route alarms to one or more destinations in an intelligent way. All alarms generated within the Commander are automatically sent to this module for onward routing.

The Alarm Router can send alarms to 8 destinations. When an alarm is received it is checked against a set of criteria, so that only certain alarms are sent to a particular destination.

The following criteria are configurable for each destination to decide if the alarm is to be routed. All the criteria checks must pass for the alarm to be routed.

- Enable – the destination must be enabled and a destination object specified for alarms to be routed. The enable could be controlled from an external source, e.g. from the Time Controller module
- High/Low Priority – if the high and low priority limits are set, the alarm priority must be within this range
- Comparison String – if any of the three comparison strings are set, the alarm event must begin with any one of the strings. Comparison strings have the format: 'Point name|Condition|Priority'.

A queue of alarms is maintained for routing to the destinations. An alarm remains on the queue until it is delivered to all the destinations whose criteria it matches. If several destinations have the 'Member of Any Set' option enabled, then the alarm will only be delivered to the first available destination in this group.

Each destination has a Route Object to specify the object reference of where the alarm should be routed. The Route Object can be any alarm destination within Commander, the connected system, or on the IPBus network.

Alarm destinations within Commander include the Alarm History object 'HD.ALARM' and Alarm Email device object 'ED.ALARM'.

Alarm destinations accessible via the IPBus network or connected system may include the ObSys Alarm Manager, interactive mobile phone SMS server (GSMSMS), Alarm Translator (AlmXlate), Alarm Printer (AlmPrint), AlarmStore, Alarm Re-prioritiser (AlmPrior), and Alarm Router (AlmRoute), etc. These interfaces may reside in other Commanders, Compass Points or ObSys modules. Search for help on these products for more information.

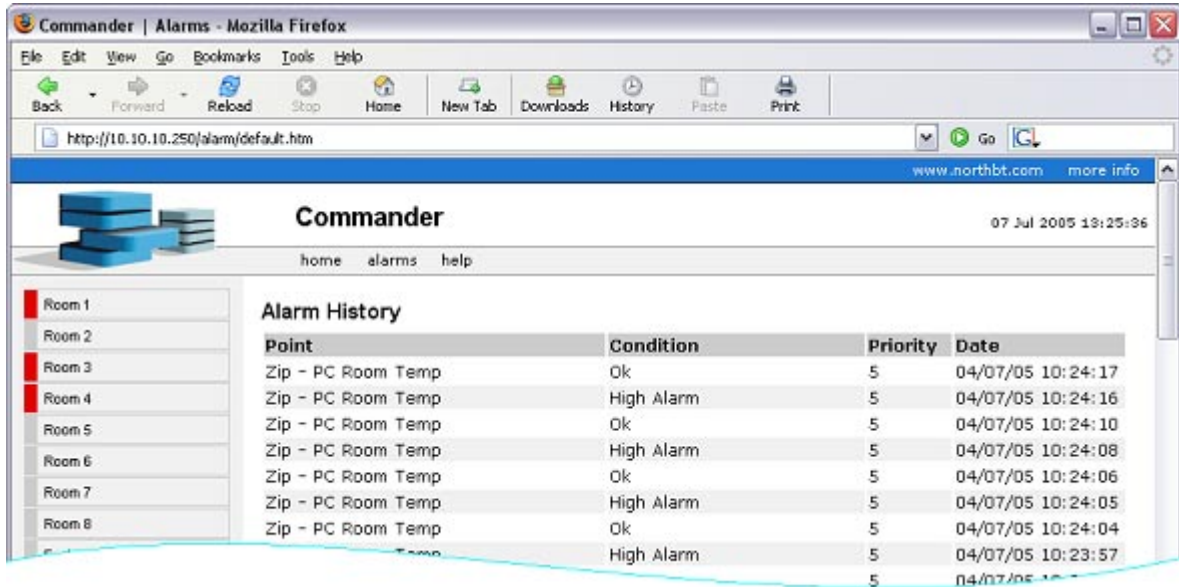
Alarm History

The Alarm History module within Commander can be used as an alarm destination from the Alarm Router.

Alarm History contains a list of the 100 most recent alarm events sent to it. As a new alarm is received, the oldest alarm is automatically removed. No user action is required.

The list of alarms may be viewed from the Web Server module and XOM compatible products, such as the North ObSys AlmView application.

The following image shows an example of the alarm history page served by Commander.



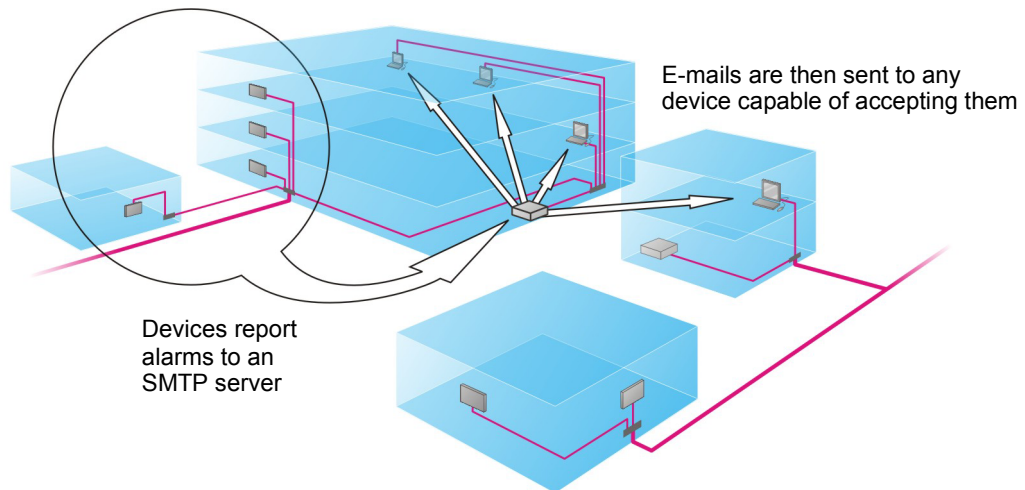
For further information on the Web Server module, refer to the to the [Introduction to Commander Networking](#).

Alarm E-mail

The Alarm Email module within Commander can be used as an alarm destination from the Alarm Router.

The Alarm Email module is used to send alarm events as an e-mail message to a recipient. The module is configured with a primary recipient e-mail address, and up to three carbon-copy e-mail addresses.

Commander sends the alarm e-mail message to an SMTP (Simple Mail Transport Protocol) server, which either stores the e-mail for later retrieval, or sends it onward to another e-mail server.



The Alarm Email module can be configured to send alarm e-mail messages formatted using HTML or text. The following image is an example on an HTML e-mail message:



To assist in the tracing of configuration problems, a debug enable option is available. This enables the monitoring of messages between Commander's Alarm Email module and the SMTP server.

SNMP Trap Events

The Simple Network Management Protocol (SNMP) is used in the management of Ethernet-type data networks. SNMP traps enable a device to notify a management station of significant events by way of an unsolicited SNMP message. Many Ethernet devices such as printers, routers, etc, can send SNMP traps to indicate their condition.

Commander can be configured to send SMTP trap notifications to a management station.

The SNMP Trap module contains a list of 100 conditions. When a condition changes state from No (0) to Yes (1), an SNMP trap notification is sent to the management station.

Configuring Commander's SNMP Settings

To enable Commander to send SMTP traps, configure the SNMP Trap Setup module with an IP address of the management station and an SNMP community name.

Commander also supports the sending of trap notifications to four additional management stations. Add the IP address for each additional management station. The same SNMP trap notification will be sent to all IP addresses.

The SNMP community name is a simple form of security. For the management station to accept traps from a device their community names must match. Although the community name is available to allow engineers to secure their network management, it is standard practice to leave the community set at its default value of 'public'.

The North MIB reference should also be specified. This is explained in the following section.

Configuring the SNMP Management Station

For a management station to understand the traps sent from the Commander, a Management Information Base (MIB) file is required. The MIB file is a description of what each trap represents. For example, trap 1 is a heartbeat signal to notify the user that the Commander is still online.

Without a MIB file the trap messages from Commander are unusable.

Contact North for more information on MIB files, and a quotation for our MIB file creation service.

Controlling Commander's Traps

The simplest way to control a trap would be for a device within Commander, the connected system, or on the IPBus network, to write directly to the trap state.

Using the OBVEngine module, additional control of the SNMP traps may be achieved. For example, ObVerse code could toggle a trap state on alarm so the trap notification is resent every 30 seconds to the management station. ObVerse code could also collect several values from the connected system and calculate a collective state for the trap.