



## Polling—SNMP and ICMP

The topics in this appendix describe the SNMP versions that LMS Fault Manager supports. They also describe how the ICMP and SNMP polling processes that LMS uses work.

The following topics are covered:

- [SNMP and ICMP Polling](#)
- [How LMS Calculates ICMP Polling Intervals](#)

### SNMP and ICMP Polling

These topics describe the polling LMS uses to obtain reachability and network health information:

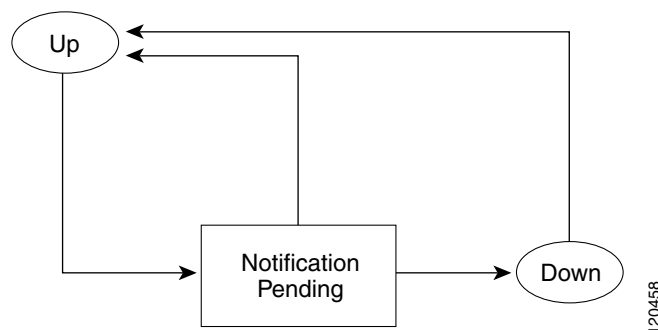
- [ICMP Polling](#)
- [SNMP Polling](#)

### ICMP Polling

LMS uses a high-performance, asynchronous ICMP poller. The poller uses two threads: one sends polls, and the other receives polls. These separate operations allow polling to continue at a stable rate.

An ICMP poll will determine an element to be in one of three possible states, as shown in [Figure C-1](#): Up, Notification Pending, and Down.

**Figure C-1** Three Possible States of an Element During a Polling Cycle



- When an element in the Up state fails to respond to an ICMP poll, LMS moves it to the Notification Pending state until LMS can determine whether it is up or down.
- The element returns to the Up state when the poll exceeds the maximum success retry count.
- The element is returned to the Up state when it responds to an ICMP poll before the maximum failure entry retry count is exceeded.

If it does not respond in time, it is placed in the Down state. LMS does not poll the element again until the next scheduled polling cycle. An element stays in the Down state until it responds to an ICMP poll.

- When the element responds, LMS changes the status of the element to the Up state.

LMS performs this polling for each managed IP endpoint.

## SNMP Polling

By default, the SNMP poller uses ten synchronous polling threads. The SNMP poller supports the following SNMP versions:

- SNMP V1
- SNMP V2C
- SNMP V3



### Note

SNMP V3 authentication and encryption pass phrases should not contain any special characters, such as !, @, #, \$, %, ^, &, \*, due to the limitations in the tool used by fault management to poll devices.

LMS uses a 32-bit counter in its correlation analysis for SNMP V1. For SNMP V2C or V3, LMS uses high-capacity, 64-bit counters. This is critical to avoid *wrapping* which is the overflow of counters between polls—for high-speed data links).

LMS supports polling devices with multiple IP addresses because the SNMP poller supports multiple IP addresses for each SNMP agent. The poller automatically switches to an alternate IP address during failures.

These topics describe SNMP polling in more detail:

- [Just-in-Time Polling](#)
- [Consolidating Requests to Optimize Polling](#)
- [Coordinating ICMP and SNMP Polling](#)

## Just-in-Time Polling

The MIB variable poll list of the SNMP Poller is driven by a Just-In-Time polling algorithm. This algorithm ensures that only those MIB variables needed for correlation are polled. When an element is re-enabled, or comes back up, the variables are automatically polled again.

## Consolidating Requests to Optimize Polling

The SNMP poller consolidates as many attributes as possible into a single SNMP GET request. This consolidation is not restricted to variables from the same SNMP table. The poller continually adapts to changes in the MIB variable poll list.

If the SNMP poller encounters a nonfatal error while consolidating, the SNMP poller responds differently to an SNMP V1 agent than to an SNMP V2C or V3 agent.

This is because an SNMP V1 agent will stop processing a request upon encountering an error, while an SNMP V2C or V3 agent continues processing a request upon encountering an error. (SNMP V2C or V3 agents handle errors on a per-OID basis.)

- If an SNMP V1 agent encounters a nonfatal error during a GET request seeking multiple variables, the SNMP poller suspends the polling of the affected variable. If it continued to poll that variable, the remainder of the request would have to be resent after receiving the error. This could impact SNMP V1 agent performance.

The SNMP poller continues to poll the unaffected variables. An affected variable is one that, for example, has become unavailable because of a configuration change.

This enables the SNMP poller to operate efficiently with an SNMP V1 agent during unexpected device configuration changes.

- In contrast, if an SNMP V2C or V3 agent encounters a nonfatal error during a GET request seeking multiple variables, the SNMP poller continues to poll both the affected and unaffected variables.

## Coordinating ICMP and SNMP Polling

LMS links its ICMP and SNMP pollers; the SNMP poller will not send requests to any SNMP agent IP address that the ICMP poller has determined is unreachable.

These IP addresses are added to a Do Not Poll list, which the SNMP poller checks before sending SNMP requests. It does not send requests to addresses on this list.

If the SNMP agent has multiple IP addresses, the SNMP poller checks for each address in the Do Not Poll list:

- If an address does not appear on the list, the SNMP poller sends a request to it.
- If all addresses for an agent are on the list, the SNMP poller considers the agent unreachable and temporarily suspends sending SNMP requests to it. As soon as an agent's IP address becomes responsive (per the ICMP poller), the address is removed from the list, and SNMP polling resumes.

## How LMS Calculates ICMP Polling Intervals

LMS calculates ICMP polling intervals for a system (for example, a switch or router) as an offset of the reachability setting's polling interval for a system.

System reachability is monitored using a combination of ICMP (Ping) requests for IP status and SNMP requests for interface, port, and card status. If a device does not respond to an ICMP poll, it is placed on a Do Not Poll list.

LMS calculates ICMP polling intervals as an offset of the reachability setting's polling interval for a system. The following is an example of a calculation based on the default value of 240 seconds.

1. LMS calculates the offset using this formula:

```
offset = 60;
If (offset > pollingInterval * 0.5) {
offset = pollingInterval * 0.5;
}
```

2. LMS calculates the ICIM polling interval using this formula:

$$icimPollingInterval = pollingInterval - offset$$

Thus, the default polling intervals are as follows:

- ICMP polling interval is 3 minutes.
- SNMP polling interval is 4 minutes.