FTI Network Discovery, Health, and Status Monitoring

Nikki Cranley, Ph.D, Diarmuid Corry, M.Sc.
ACRA CONTROL INC., Maryland, USA

ABSTRACT

The ability to discover the topology and configuration of a networked Flight Test Instrumentation (FTI) system is a powerful feature enabled by the adoption of Ethernet technology. Discovery allows the FTI system to be debugged, verified against meta-data, and monitored for health and status. This paper focuses on two levels of FTI discovery, discovering the network topology and discovering the configuration of the FTI network devices. Moreover, this paper describes how the same discovery technologies may be used for health and status monitoring.

1. INTRODUCTION

The Flight Test Instrumentation (FTI) industry has instigated a significant paradigm shift from the use of proprietary closed solution towards more open standards-based systems using Ethernet technology [1, 2, 3] in recent years. There are two key factors that prompted and facilitated this change. The precursor was the development of the IEEE 1588 Precision Time Protocol (PTP). Previous network-based time synchronization protocols, such as Network Time Protocol (NTP), did not provide adequate accuracy for distributed time synchronization. PTP enabled high accuracy time synchronization ensured the ability to provide isochronous sampling in a distributed network. The second driving factor was the standardization work performed by the CTEIP Integrated Network Enhanced Telemetry (iNET) initiative. After much investigation and analysis of candidate communications technologies, the iNET working groups decided that Ethernet is the technology of choice to meet the requirements of current and future FTI [4] needs. iNET is addressing and bringing about a standardized approach to meet the needs and requirements of FTI in terms of systems management, time synchronization, Data Acquisition Unit (DAU) configuration, data transmission etc. Using open standard technologies offers the FTI community with greater flexibility and scalability in system design and more choice for multi-vendor interoperable systems.

There are a number of open standard network protocols that can be used to discover, manage, and debug networked based FTI systems. One such protocol is the Simple Network Management Protocol (SNMP) which is a standard IP network protocol [5] that can be used to interrogate, query, and configure SNMP-enabled devices. Notably, iNET has adopted SNMP as a core technology of choice for systems management. For this reason, SNMP is the focus of this paper.

The remainder of this paper is structured as follows: Section 2 provides an overview of the Simple Network Management Protocol (SNMP) which is the primary mechanism by which network and FTI discovery can be achieved. Section 3 outlines the practical use case examples of using the SNMP for discovery. Furthermore, other networking techniques that may be employed to perform discovery are discussed. Not only is discovery possible through the use of SNMP, Section 4 describes the facility of using SNMP for Quality of Service, health, and status monitoring.

2. OVERVIEW OF THE SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP)

SNMP is a simple but powerful utility that can be used to configure (SET) and discover (GET) the configuration of an SNMP-enabled device. SNMP is a component of the Internet Protocol Suite as

defined by the IETF [6]. It consists of a set of standards for network management, including an application layer protocol, a database schema, and a set of data objects. SNMP exposes management data in the form of variables on the managed systems, which describe the system configuration. These variables can then be on-demand remotely i.e. 'telecontrolled', configured (SET) and queried (GET) on the network end node. In addition, an end node may be configured to set a trap for key events (TRAP) without the need for polling.

SNMP uses an extensible and customisable Management Information Base (MIB) to describe the variables that may be accessed in the FTI device including the structure, interpretation, and read/write attributes of the supported variables. The MIB contains global variables that are common to all networked FTI devices. In networked FTI SNMP may be used to monitor and remotely configure network nodes such as:

- Data Acquisition Unit (DAU)
- Switch
- Grandmaster
- Network-Recorder.

Each of these classes of device has a set of specialised discoverable variables that are also described in the MIB.

An SNMP-managed network [7] consists of three key components as shown in Figure 1:

- **Network management system (NMS):** is a software application that monitors and controls managed devices in the networked FTI system. NMSs provide the bulk of the processing and memory resources required for network management. There may be one or more NMSs managing the network. The NMS uses UDP port 162 for SNMP.
- **Managed device**: is a managed network node (Network-Recorder, Switch, Grandmaster, DAU etc) that contains an SNMP agent. Managed devices collect and store management information and make this information available to NMSs via the SNMP protocol.
- **Agent**: is a network-management module that resides in the managed device. An agent has local knowledge of management information and translates that information into a form compatible with SNMP. The agent uses the UDP port 161 for SNMP.

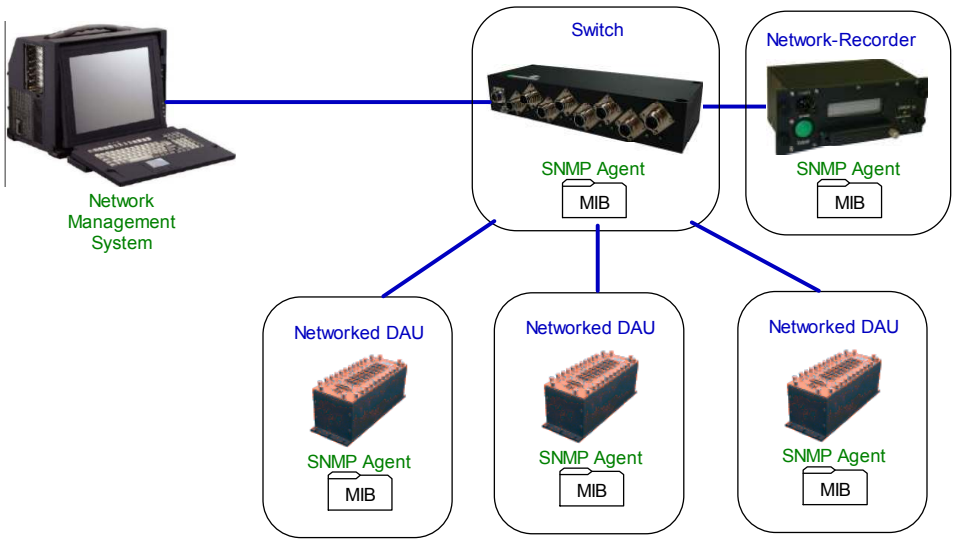

Figure 1: SNMP Architecture

The SNMP protocol is a Request-Response protocol whereby the NMS issues queries and configuration commands via SNMP to the managed device, for example the SNMP-enabled DAU as shown in Figure 2 with the SNMP messages described in Table 1.

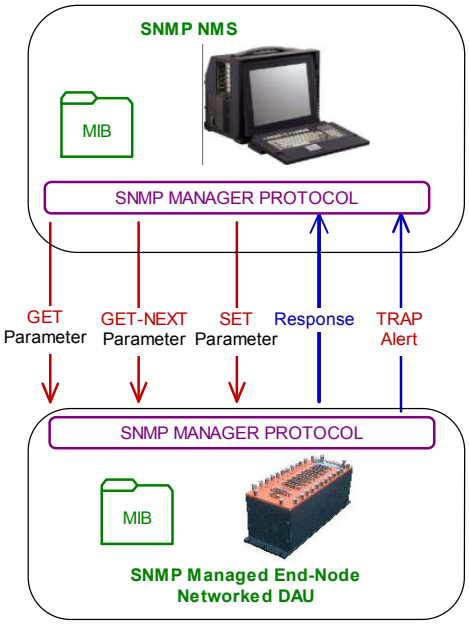

Figure 2: SNMP Messages in Action

Table 1: SNMP Messages		
SNMP Version	**SNMP Command**	**Description**
V1	GET	Used by the NMS to retrieve the value of one or more object instances from an agent. Example: GET EventNumber
V1	GETNEXT	Used by the NMS to retrieve the value of the next object instance in a table or a list within an agent. Example: GETNEXT NextEventNumber
V1	SET	Used by the NMS to set the values of object instances within an agent. Example: SET IPAddress
V1	TRAP	Used by agents to asynchronously inform the NMS of a significant event. TRAP TemperatureExceedsThreshold
V2c	GETBULK	Used by the NMS to efficiently retrieve large blocks of data.

2.1. MANAGEMENT INFORMATION BASES (MIBS)

SNMP is a protocol and does not define which information or variables are managed. The variables accessible via SNMP are organized in hierarchies with meta-data (type and variable description). The variables accessible via SNMP are described by Management Information Bases (MIBs) [8]. MIBs describe the structure of the management data of a device subsystem using a hierarchical namespace containing object identifiers (OID) as shown in Figure 3. A full listing of the MIB hierarchy can be found at [9]. Each OID is unique and identifies a variable that can be read or set via SNMP. Two types of managed objects exist:

- Scalar objects define a single object instance.
- Tabular objects define multiple related object instances that are grouped in MIB tables.

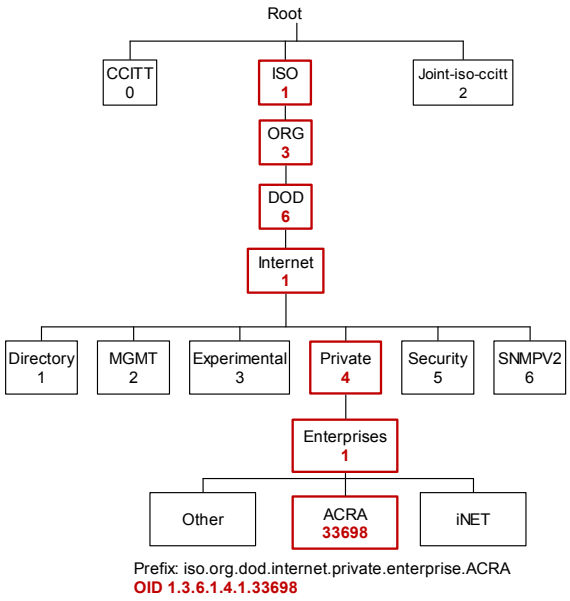

Figure 3: OID Hierarchy

A useful utility is the ability for the NMS to perform a MIB or SNMP Walk whereby the walk can be used to retrieve a sub-tree of the MIB from a specified OID branch and the values returned to the NMS. The MIB walk is achieved using SNMP-GETNEXT messages

2.2. SNMP IN ACTION

By way of a simple example of SNMP in operation as shown in Figure 4, consider a system comprised of a NMS and an SNMP-enabled DAU, where the DAU has an SNMP-Trap variable for Temperature. The NMS sends an SNMP-GET message to the DAU to retrieve the value for the Temperature Threshold variable as defined in the MIB supported by the DAU. The DAU's own SNMP-Agent interprets the received SNMP-GET message containing the OID of the Temperature Threshold variable. In this way, the DAU SNMP-Agent can retrieve the desired variable value and return the appropriate response to the requesting NMS. Should the NMS choose to reconfigure the Temperature Threshold variable on the DAU, the NMS sends an SNMP-SET message to the DAU. However the Temperature Threshold variable must have read/write attributes so that it can be reconfigured. The SNMP-Agent continuously monitors its Current Temperature over time. When the DAU detects that the Current Temperature exceeds the newly defined Temperature Threshold, the DAU can send an SNMP-Trap to the Trap listener in the NMS to warn it of the occurrence of this event.

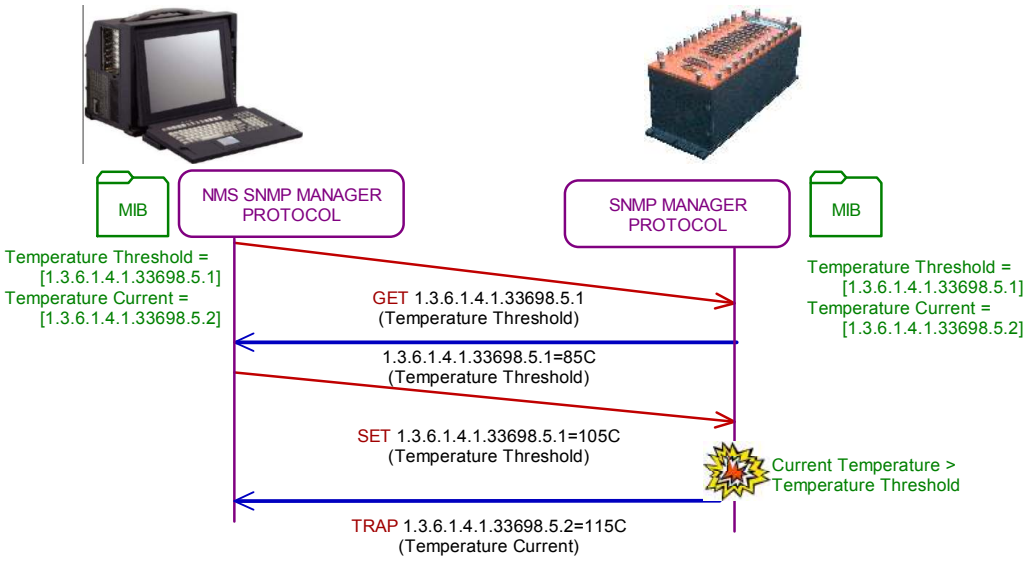

Figure 4: SNMP Example

2.3. COMMENTS ON SNMP USE

It is clear that SNMP is a powerful utility for reconfiguring and querying devices for key settings. However there are some caveats that must be considered before it is implemented.

- Although TCP transport is possible, SNMP typically runs over UDP, an unreliable protocol. Therefore, there can be no guarantee that commands/messages issued by SNMP are correctly propagated through the network. To mitigate this, each SNMP-SET operation should be safeguarded by a proceeding SNMP-GET to verify the success of the SET operation.
- SNMP allows the NMS to potentially configure critical variables that alter the devices' configuration and operation. Care should be taken when determining the variables that are "settable" via SNMP. These variables should not interrupt packet switching, data acquisition, or allow the device to enter an unstable state.
- There are several flavours of the SNMP protocol. As currently specified, SNMPv2 is incompatible with SNMPv1 in two key areas: message formats and protocol operations. SNMPv2c messages use different header and protocol data unit (PDU) formats from SNMPv1 messages. SNMPv2c also uses two protocol operations that are not specified in SNMPv1. However, RFC1908 defines two possible SNMPv1/v2c coexistence strategies: proxy agents and bilingual network-management systems.
- SNMPv1 and SNMPv2c pose a security risk in that packet sniffing can be used to monitor auto-discovery advertisements. Although this has been addressed in SNMPv3 through the introduction of encryption techniques, SNMPv3 has a significantly larger footprint and consumes onerous processing resources.

3. DISCOVERY

The benefits of discovery are three-fold, the first is the ability to perform a blind-discovery of an unknown system, secondly the verification of the current systems configuration, and thirdly provide a mechanism to discover, debug and diagnose faults in the system. Particularly in large systems, configuration errors are inevitable which are costly to debug. To name just a few, for example, from a network configuration perspective, possible faults include: using duplicate IP addresses particularly for FTI where static IP address assignment is typically used; incorrect wiring and connections; devices being added/removed to the system; the system description metadata file is lost and its current configuration is unknown.

The ability to discover the network and the configuration of the system, FTI engineers are able to react to and prevent such issues arising by tracing the fault through the network to the faulty device. Once the networks configuration is validated and its correct operation verified, the FTI engineer can focus on discovering the configuration of the individual DAUs in the system.

3.1. NETWORK TOPOLOGY DISCOVERY

Although SNMP could be used for network topology discovery, there are several other networking utilities that can be used to discover the devices in the network whereby the NMS can query the network switches for their routing tables. In this Section, two basic technologies are described, namely Broadcast ping and traceroute. The former is used to retrieve a list of the networked devices with their IP and MAC addresses. The latter is used to infer the topology of the discovered devices.

Ping is a networking tool used to test whether a particular network end node is reachable; it may also be used to self-test the network interface, or be used as a latency test. Ping sends an ICMP “echo request” packet to the target end node and listens for ICMP “echo response” reply from the end node. Ping measures the round-trip time (i.e. latency from itself to the end-node and back) and records any packet loss, and prints when finished a statistical summary of the echo response packets received, the minimum, mean, max and in some versions the standard deviation of the round trip time. To discover the devices in a network, it is possible to issue a Broadcast ping. Since the ping is broadcast, all devices in the network receive the ping and return an “echo response” to the initiator. Having received all the responses from active devices in the network, the initiator issues an “arp –a” command to retrieve the ARP cache where the ARP cache contains a list of IP addresses and MAC addresses. It should be noted, that only devices that support Broadcast ping would respond. To complete the device discovery, an SNMP-GET message can be issued to query each device for its device type, i.e. DAU, recorder, switch and so on.

An extension of pings utility is Traceroute, which is used to determine the route taken by packets across the network. Traceroute works by increasing the "time-to-live" value of each successive batch of ping packets sent. The first batch of ping packets are sent have a time-to-live (TTL) value of one (implying that they are not forwarded by the next switch/router and make only a single hop). The next batch of ping packets have a TTL value of 2, and so on. When a packet passes through a switch/router, normally the switch/router decrements the TTL value by one, and forwards the packet to the next switch/router. When a packet with a TTL of one reaches the target DAU, the DAU discards the packet and sends an ICMP time exceeded (type 11) packet to the sender. The traceroute utility uses these returning packets to produce a list of intermediate network nodes that the packets have traversed en route to the target DAU allowing for a topology map to be generated. This method can only be applied if the network devices support the TTL decrement function. Moreover, unmanaged devices (i.e. those without an IP address) will appear invisible in the network.

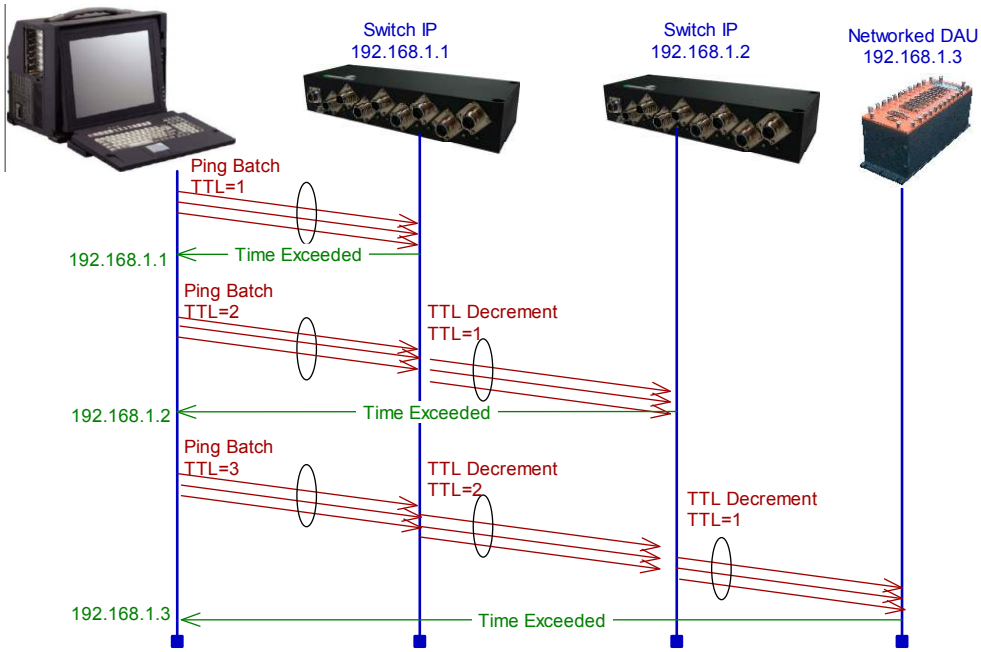

Figure 5: Trace route topology discovery

3.2. DAU DISCOVERY

Having discovered the network, the next task is the discovery of the DAUs configuration. This is largely a DAU specific and vendor specific operation. However, it should be re-iterated that SNMP should be prevented from potentially configuring critical variables that may interfere with the DAUs operation and data acquisition. Again, using SNMP the NMS can retrieve the instrument types in the DAU. This can be subsequently verified and validated against the metadata that was used to configure the DAU or equally the retrieved information can be used to generate the metadata.

4. HEALTH AND STATUS MONITORING

Monitoring the health and status of the FTI network can be performed using SNMP. In particular the Remote MONitoring (RMON) MIB [10] has been developed to address such a requirement. Where SNMP is aimed at device management, RMON is designed for monitoring network traffic. The RMON MIB contains numerous variables that can be used to diagnose faults in the network and perform statistics, history, alarm, and event monitoring. The RMON MIB has been extended in the RMON-II MIB [11] that contains more detailed network health and status variables, in particular those relating to network, transport and application layer protocols. SNMP-RMON requests are targeted at the intermediate switches and routers in order to identify Quality of Service (QoS) issues associated with the networks performance. Furthermore, the RMON MIB provides a mechanism of identifying bottlenecks in the networks design and operation allowing per-port throughput and packet-loss statistics to be gathered.

5. CONCLUSIONS

The adoption of Ethernet as a core technology for FTI, not only provides greater interoperability with all devices use common open standard technologies but also greater bandwidth, higher data rates, faster links, flexibility, and scalability. IP based protocols can be used to provide a host of new services never before possible in a non-networked FTI system. An important new service is the ability to discover and monitor the health and status of the network and networked devices. To achieve this service the Simple Network Management Protocol (SNMP) can be used. By definition, SNMP is a simple, powerful, and extensible protocol. It allows a device to support a variety of configuration settings and variables that

can be queried and configured on-demand by the end-user. This ability can be applied to provide a discovery service the entire networked FTI system, allowing not only the devices to be discovered, but also their configuration and inter-connections with topology discovery. This paper provides an introduction of the SNMP protocol and describes how it may be used to perform network topology and device discovery. Moreover, this same SNMP mechanism can be used to perform health and status monitoring of the network allowing the end-user to gather performance metrics and statistics that can be used to identify bottlenecks and congestion in the system.

REFERENCES

[1] Eccles, Lee, "Network Based data Acquisition", Proceedings of European Telemetry Conference (ETC), Munich, Germany, 2008

[2] Revaux, Nathalie and Abadie, Frédéric, "A380 Flight Test Architecture: Switched IENA from Pulse Code Modulation to Ethernet LAN", Proceedings of European Test and Telemetry Conference (ETTC), Toulouse, France, 2003

[3] Doyle, David and Canizares, Ruben, "Review of a successful distributed networked and modular FTI implementation", to be published in Proceedings of Society of Flight Test Engineers Symposium 39, 2008

[4] Grace, Thomas and Hodack, David, "Vehicle Network Technology Demonstration", Proceedings of International Telemetry Conference, 2007.

[5] IETF, RFC 3411, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", Dec 2002

[6] RFC 3416 (Standard 62) — Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)

[7] RFC 3411 (Standard 62) — An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks

[8] RFC 3418 (Standard 62) — Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)

[9] http://www.iana.org/assignments/enterprise-numbers

[10] RMON1: RFC 2819 - Remote Network Monitoring Management Information Base

[11] RMON2: RFC 2021 - Remote Network Monitoring Management Information Base Version 2 using SMIv2