SC35-A-14

IMPLEMENTING THE TELECONTROL APPLICATION FUNCTIONS DEFINED BY IEC 870-5-104 USING SNMPv3 SERVICES

by

G. Sánchez[1]; A. Barbancho[2]; I. Gómez[3]; J. Luque[4]; V. Medina[5]

Department of Electronic Technology

University of Seville

(Spain)

gemma@us.es[1]; ayboc@us.es[2]; igomez@us.es[3]; jluque@us.es[4]; vmedina@us.es[5]

ABSTRACT.

This paper justifies and describes the use of the telecommunication network management protocol SNMP versión 3 to implement the application functions defined by IEC 870-5-104 [1], and the replacement of the functional profile defined by this standard with a profile based on the TCP/IP protocol suite, including SNMP in its application layer.

This work is part of a project financed by the Spanish Ministry of Science and Technology (TIC2000-1114).

1. INTRODUCTION.

In the field of electrical network telecontrol the protocols being traditionally used were either specific of a manufacturer, or result of standardizing efforts in this specific area (for instance, the EPA model from IEC). This kind of solutions makes sense in an application with strong real time restrictions, slow transmission speeds, and a high homogeneity of equipment to be telecontrolled. However, during the last years, although real time restrictions are still present, it does not happen the same with the other two determinants. In effect, from one side, the available means of communication today have removed speed restrictions prevailing till some years ago; from the other, the intelligence available in many of the equipment present in substations or control centers makes the supplied information not simply being an alarm contact or a set of measurements semantically very restricted. [2]

In this new context, the use of classic protocols must be reevaluated, thus being possible the consideration of solutions more flexible and with a higher rate of commercial penetration and standardization, although of lesser efficiency. This approach opens a new field of solutions that permits the exploitation of all the rich and dynamic experience of a field in constant evolution such as network management for electrical network telecontrol.

Specifically it is proposed the use of the "de facto" network management standard SNMPv3, member of the TCP/IP architecture, to carry out the telecontrol functions defined by the IEC 870-5-104 standard. This companion standard defines the network access for the functional profiles defined by IEC 870-5-101, based on the EPA model, using standard transport profiles; and the correspondence between the application functions defined by IEC 870-5-5 and the services provided by TCP/IP. It is not based on the EPA model as it includes a transport and network layer. The profile proposed by IEC 870-5-104 combines the application layer of IEC 870-5-101 with transport functions provided by a TCP/IP-based WAN. This permits stations to be connected via data networks made up of nodes that store and retransmit messages, and provide virtual circuits among those. TCP/IP supports different kinds of networks, such as X.25, frame relay, ATM and ISDN.[3]

This new approach based on SNMPv3 gives us independence from vendors since free implementations of SNMPv3 exist. Additionally, this could be applied to other fields of telecontrol besides electrical networks, like industry or intelligent home.

2. PROTOCOL STACKS.

The proposed approach replaces the stack based in the standard protocols of the IEC 870-5 series (specific of telecontrol) that drive the communication between RTUs and control centers by a stack of protocols non specific of telecontrol. To be exact, the services of the telecommunication network management protocol SNMPv3 are used to implement the telecontrol specific application functions defined by IEC 870-5-104. [4]

Both stacks of protocols are compared by equating their equivalent layers. Figure 1 shows three protocol stacks: the first shows the standards selected from the IEC 870-5 series for each layer of the EPA model by the companion standard IEC 870-5-101; the second one is the protocol stack defined by IEC 870-5-104, based on the previous adding network access via the TCP/IP protocol suite; the last protocol stack reflects our new approach, replacing the application layer from IEC 870-5-104 by SNMPv3 services implementing an API providing its functions.

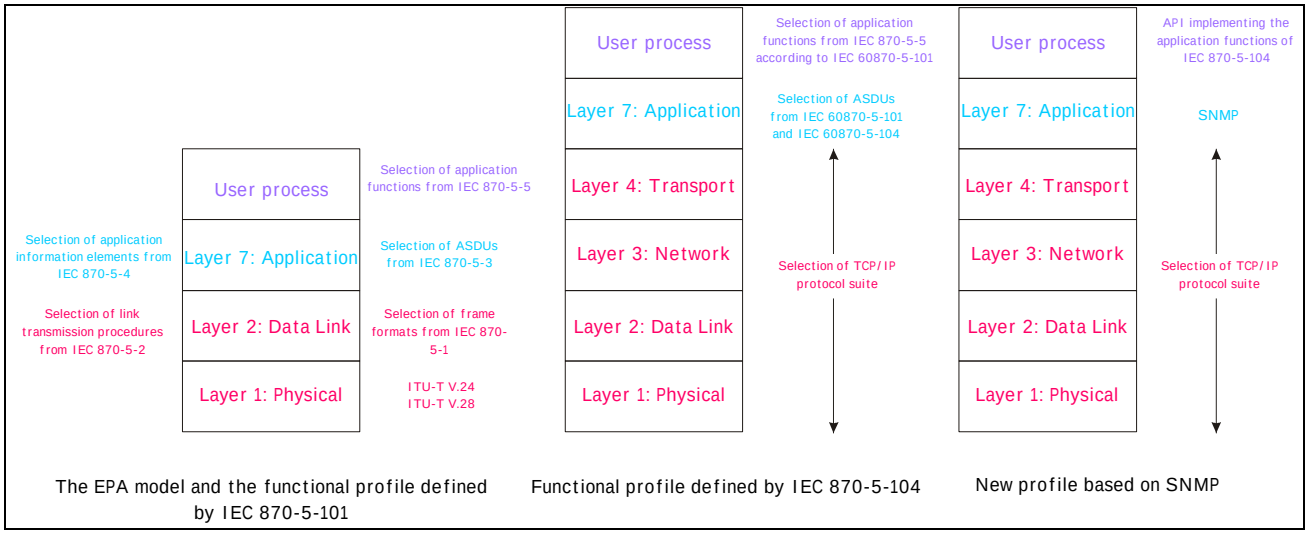

Figure 1 [5] [6] [7] [8] [9] [10]

In this approach, control centers play the role of SNMP managers and RTUs are the SNMP agents.

3. RTUs MIB.

A MIB must be designed to carry out this tasks. This MIB contains the information handled by the RTU in an organized way and is stored in the agent present in it. The manager can access to this MIB via SNMP services, such as *SETREQUEST* (to set the value of a variable), *GETREQUEST* (to get the value of a variable) and *GETBULKREQUEST* (to get the values of many variables).

4. IMPLEMENTATION OF THE APPLICATION FUNCTIONS.

The implementation of the most interesting functions of the application layer from IEC 870-5-104 via the SNMPv3 services will be exposed now.

Command transmission. [1]

To implement this application function via SNMPv3 services a MIB variable is defined for each command. This variable would contain one of the next values representing the state of the command: *Inactive command*, *Selection request*, *Accepted selection*, *Refused selection*, *Interruption of selection request*, *Interrupted selection*, *Execution request*, *Executing command*, *Refused command* and *Executed command*. Both stations modify the value of this variable to reflect the evolution of the command, when the agent does it a *TRAP* is sent to notify the manager of this change of state.

The initial state is *Inactive command*, the control center must check that the desired

command is in this state before request it, otherwise this could not be requested. This is done by a *GETREQUEST* PDU over the variable for this command.

The evolution of the value of this variable depends on the kind of command:

Select and execute commands.

The manager begins the selection phase setting the value of the variable to *Selection request* by *SETREQUEST*. Then the agent could:

- Accept the selection setting the value of the variable to *Accepted selection* by *SETREQUEST* and sending a *TRAP PDU* notifing this. The agent stays in this state until the manager:
  - Requests the execution of the command setting the value of the variable to *Execution request* by *SETREQUEST*. Then the agent sets it to *Executing command*, sends a *TRAP PDU* to notify this to the manager and executes the command. Once the execution has finished the agent sets the value of the variable to *Executed command* and sends the *TRAP PDU* to inform the manager. As response, this sets the value of the variable to *Inactive command* by *SETREQUEST*. Figure 2.
  - Requests the interruption of the selection procedure setting the value of the variable to *Interruption of selection request* by *SETREQUEST*. The agent sets the value of the variable to *Interrupted selection* and sends a *TRAP PDU* informing the manager that the interruption of the selection proccess has been done. As response, the manager deactivates the command setting the value of the variable to *Inactive command* by *SETREQUEST*. Figure 3.
- Refuse the selection setting the value of the variable to *Refused selection* and sending a notifing *TRAP PDU*. When the manager receives this, it deactivates the command setting the value of the variable to *Inactive command* by *SETREQUEST*. Figure 4.

Direct commands.

To request the execution of the command the manager sets the value of the variable to *Execution request* by *SETREQUEST*. Then the agent could:

- Accept the command setting the value of the variable to *Executing command*, sending a *TRAP PDU* to inform the manager about this and executing the command. Once the execution of the command has finished the agent notifies it, setting the value of the variable

to *Executed command* and sending the *TRAP PDU* to the manager. As response, this sets the value of the variable to *Inactive command* by *SETREQUEST*. Figure 5.

- Refuse the command, setting the value of the variable to *Refused command* and sending a *TRAP PDU* to inform the manager about that. As response, this sets the value of the variable to *Inactive command* by *SETREQUEST*. Figure 6.

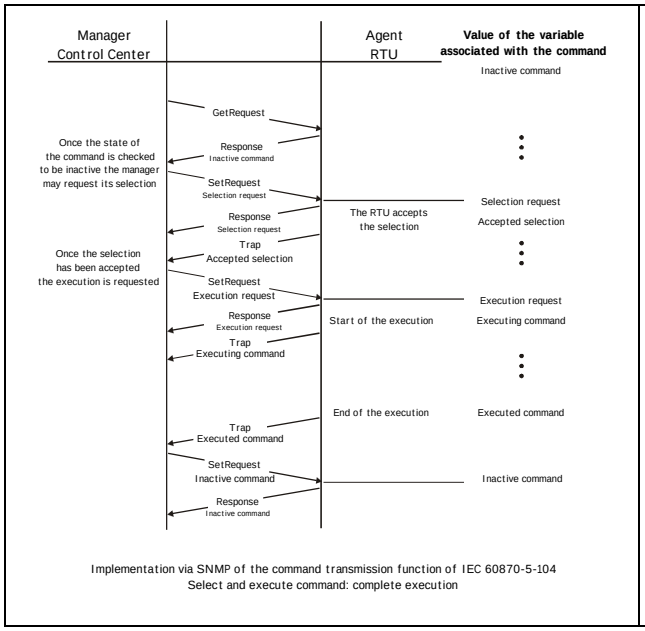

Figure 2

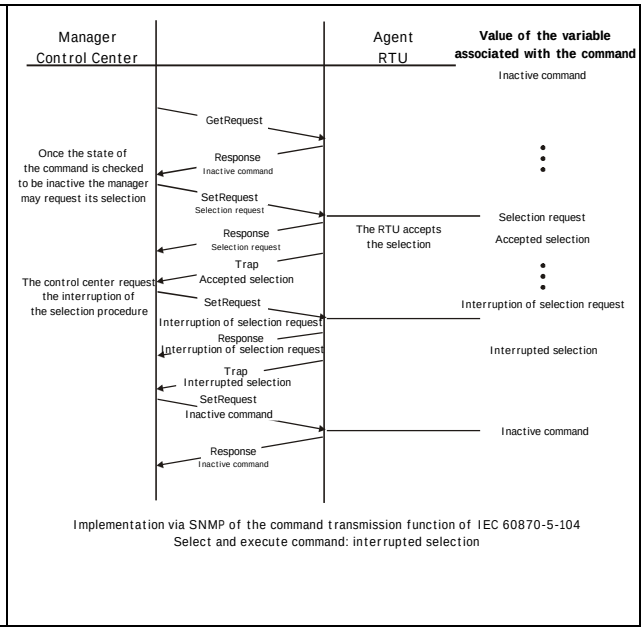

Figure 3

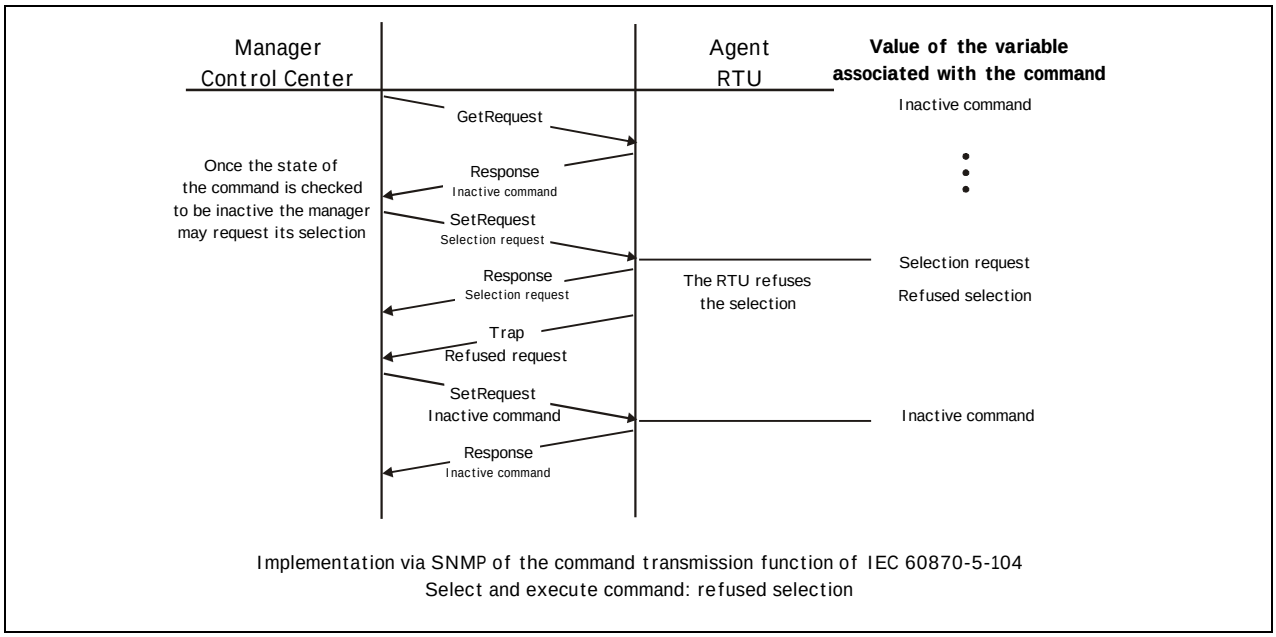

Figure 4

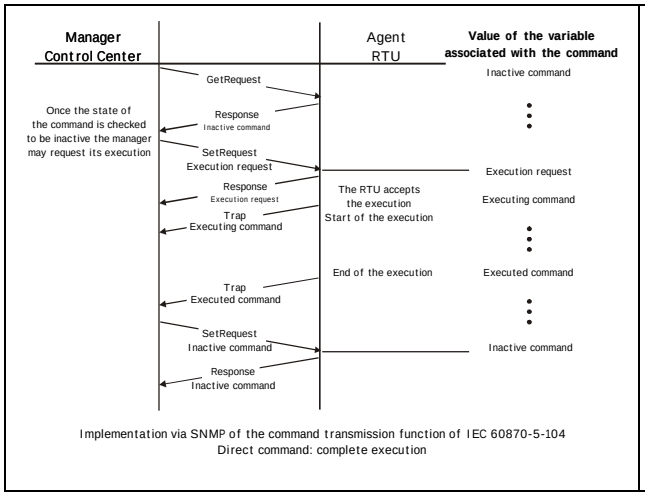

Figure 5

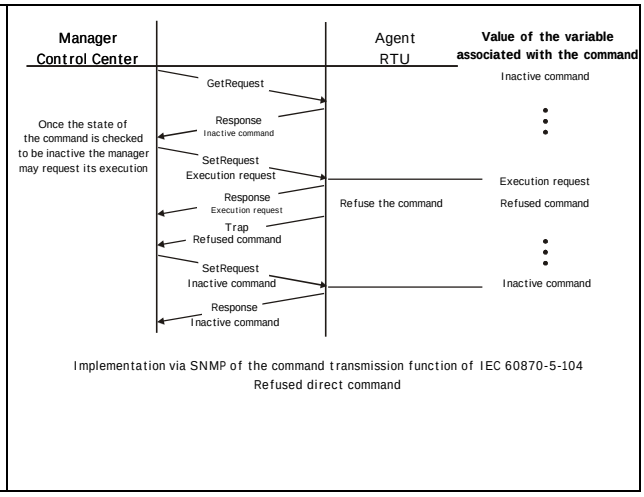

Figure 6

Parameter loading. [1]

The implementation of this application function via SNMP simplifies it very much, reducing it to a *SETREQUEST PDU* updating simultaneously the value of the MIB variables associated with the desired parameters. This makes the two phases described by IEC 870-5-104 unnecessary.

Local changes of parameters may be notified to the control center by a *TRAP PDU*.

Acquisition of events. [1]

This application function is implemented via the *TRAP PDU*. This PDU is sended to the manager when an event occurs in the agent. This has the advantage of making unnecessary the poll of the agent by the manager for events since they are immediately reported.

Transmission of integrated totals. [1]

The acquisition of values from the counters can be ordered by the manager o by local timers in the agent.

The acquisition ordered by the manager is implemented by means of two commands that this sends to the agent: the *incremental information acquisition command* and the *integrated total acquisition command*, these cause the execution of a process that stores the values from the counters in MIBs variables, resetting these counters to zero for incremental information.

The transmission of the values obtained is done by means of a *GETREQUEST* or

GETBULKREQUEST PDU for the MIBs variables containing these values.

The acquisition may be activated in the agent by local timers that cause the execution of the mentioned process. The transmission is carried out by *TRAP PDUs*.

Cyclic data transmission. [1]

The agent overwrites and transmits the values of some variables to the manager at cyclic intervals. This is done with low priority which means that it can be interrupted by event-triggered communications requests.

To implement this application function with SNMP, a MIB variable is defined. This variable reflects the existence of cyclic data in the agent waiting to be sent. Its activation causes the transmission of a *TRAP PDU* to the manager. Its reception makes the manager to request cyclic data by *GETREQUEST* or *GETBULKREQUEST* when it finishes its current taks. This is called *trap directed polling*.

General interrogation. [1]

This application function is implemented by *GETBULKREQUEST* which permits the manager to request the value of many MIB variables from the agent. This PDU is used by the manager as many times as necessary to get the value of all the desired variables of all the agents.

Events can interrupt the general interrogation, since if an event occurs in the agent a *TRAP PDU* is sent to the manager so that it is punctually notified and can interrupt that process to serve the event.

Clock synchronization. [1]

The manager updates the value of a MIB variable containing temporal information by *SETREQUEST*. This can be periodically done.

Data acquisition by polling. [1]

Two classes of data can by trasnmitted from the agent to the manager: events are called class 1, they are prioritary; cyclic data are called class 2 data.

The acquisition of events is carried out by the procedure described in the section "Acquisition of events".

To implement the acquisition of cyclic data a MIB variable is defined. This is an structure

with optional fields representing these data, if there are cyclic data waiting to be transmitted the value of its field in the structure is not NULL. The acquisition is performed by reading the value of this variable by *GETREQUEST*. Once these data have been transmitted the values of the respective fields in the structure are set to NULL until they are changed by the system.

5. CONCLUSIONS.

This new approach presents several advantages that will be exposed below.

- The possibility of using free implementations of SNMP that dissociate us from the vendors.
- As its name suggests, SNMP is a very simple protocol. Its implementation is easy in large networks and the management information needing exchange takes few network resources.
- The wide experience in SNMP makes it robust and well-known since there are many groups analysing and improving it, given its importance in the right operation of the networks in many organizations.
- SNMP means a standard way of managing devices from a broad spectrum of types, being perfectly applicable to telecontrol devices.
- It simplifies a lot the procedures to carry out the application functions.
- SNMP allows the user to choose the variables he wishes to monitor in an easy way.
- SNMPv3 provides security facilities, such as authentication, encryption and timeliness checking.

6. REFERENCES.

[1] The International Electrotechnical Commission. "Telecontrol Equipment and Systems. Part 5: Transmission Protocols. Section 104: Network Access for IEC 60870-5-101 Using Standard Transport Profiles". IEC 60870-5-104. First Edition, 2000-12.

[2] G. Sánchez, I. Gómez, F. Pérez, V. Medina, *"Using standard management protocols in electrical network telecontrol"* SC35 CIGRE 2001 Colloquium. Zagreb (Croatia).

[3] R. Baumann, L. Björk, D. C. Cooper, J. Florencio, J. Hegge, P. Ligtvoet, M. A. López, J.M. Selga, H. Spelt, T. Watson. "Telecontrol Protocols and the Importance of Testing Protocol Implementatios Against the Standards". Electra no. 188 February 2000, pages 89 to 99.

[4] W. Stallings *"SNMP, SNMPv2, SNMPv3, and RMON1 and 2."* Third Edition. Addison Wesley. 1999.

[5] The International Electrotechnical Commission. "Telecontrol Equipment and Systems. Part 5: Transmission Protocols. Section 1: Transmission Frame Formats". IEC 870-5-1. First Edition, 1990-02.

[6] The International Electrotechnical Commission. "Telecontrol Equipment and Systems. Part 5: Transmission Protocols. Section 2: Link Transmission Procedures". IEC 870-5-2. First Edition, 1992-04.

[7] The International Electrotechnical Commission. "Telecontrol Equipment and Systems. Part 5: Transmission Protocols. Section 3: General Structure of Application Data". IEC 870-5-3. First Edition, 1992-09.

[8] The International Electrotechnical Commission. "Telecontrol Equipment and Systems. Part 5: Transmission Protocols. Section 4: Definition and Coding of Application Information Elements". IEC 870-5-4. First Edition, 1993-08.

[9] The International Electrotechnical Commission. "Telecontrol Equipment and Systems. Part 5: Transmission Protocols. Section 5: Basic Application Functions". IEC 870-5-5. First Edition, 1995-06.

[10] The International Electrotechnical Commission. "Telecontrol Equipment and Systems. Part 5: Transmission Protocols. Section 101: Companion Standard for Basic Telecontrol Tasks". IEC 870-5-101. First Edition, 1995-11.