ERICSSON

Literature Study
within
Master thesis project

MIB modeling of AXE Regional Processors

by
Per Holmgren
2000-10-20
at
Ericsson Utvecklings AB

ERICSSON

Abstract

The motivation for this thesis is that Ericsson wants to start using open standards and platforms for controlling their AXE telephony exchanges out on the field. Up until now Ericsson has based their AXE communication on signals, and not any standard protocol. The idea for this thesis is first to investigate what informations regarding operations and maintenance are transferred back and forth between a central and the exchanges out on the field. From these informations most of the necessary data for operations and maintenance can be extracted and put into quite simple but numerous objects that model the variables and tables in an exchange. T hese objects will then be put in a MIB (Management Information Base). Then simulation is made by the use of software that simulates the SNMP (Simple Network Management Protocol) protocol with a manager and managed agents.

Introduction

The motivation for this thesis is that Ericsson wants to start using open standards and platforms, in which the IP-family is a part and widely spread. With the complexity that the future holds for the telecommunication business Ericsson believes they can not hold on to their own standards for all eternity. A big reason for Ericsson's success in the area is that the AXE exchanges are built on modularity, which makes it relatively easy to add or remove functionality in them. In the development of new hardware/software Ericsson also ensures that it is back compatible.

An Ericsson AXE exchange system is simplifiedly divided into two major parts: a CP (Central Processor) and RPs (Regional Processors). There are many types of RPs and fewer types of CPs. An exchange out on the field is built with a subset of the total number of RPs that are available, depending on what type of exchange it is (wired telephone, gsm, umts etc.), and the exchange is managed by a CP through signaling. Those signals (thousands in total) have been developed by Ericsson over the years and now the signals are collected in a database where they each are described byte for byte.

Each RP has software which is also divided into two major parts: the APT (telephony applications part) and the APZ (maintenance and operations part). For this thesis it is the APZ part that is of interest. The APZ can contain up to 16 blocks, i.e. programs, that deal with things such as file management, communication supervision, error handling etc. The operating system of an RP is also considered as a part of the APZ, i.e. the operating system is a block in the APZ. The operating system supervises all other blocks in the RP, both APZ and APT blocks.

This thesis will first be focused at the signalling between the CP and RPs. However, sometimes sidesteps are done to look at the communication between different RP blocks and also sometimes then look at raw program code to determine which block is/should be responsible for the data representation. This can sometimes be hard to determine from the outside just by looking at the signalling between the CP and RPs, because there can sometimes be interactions between different blocks which are not visible by only looking on the direct signalling between a CP and RP block.

Since the AXE system is based in CPs and RPs which interact as managers and agents it is natural to look at the SNMP protocol, which is the most spread protocol for network management and also a part of the open standard IP-family. An SNMP manager uses a model of the agents it is controlling by using a predefined database from which the SNMP manager can fetch and set necessary variables in the controlled agents. The database is called a MIB (Management Information Base), which is stored in the manager. The MIB models the data and variables that are available in the agent(s).

A MIB is built like a hierarchical tree which eventually branches out into leaves. Each leaf make out one object where each object can consist of one simple variable or reference sequences of other objects to make out tables. In the objects there are also definitions of access rights, current status and also a textual description of what the variable in the object is supposed to represent. A MIB at least consists of the standard MIB [4] which is necessary for the underlying layers of the IP stack for communication etc. It is then possible for for example enterprises to extend the MIB tree and add their own objects in a dedicated enterprise branch to model their own agents.

Background

There was also an investigations team in 1998 that looked on in what direction to aim for the future when it comes to dealing the management of stand alone platforms[4]. This report suggested that TCP/IP suite should be looked at, such as SNMP, telnet, TFTP etc.

At Ericsson no work has really been done before on the RP blocks on which this report will focus. Some work on modeling other RP blocks using MIBs has been done before though [2]. That work implemented a few MIB objects in a Regional Processor with Group switch interface (RPG), mainly to see how hardware and software would respond.

As a main source of theory a book by William Stallings [1] has been used and some RFCs [7][8][9] as complement to that book.

The reader of this report should have a general understanding of networking and the TCP/IP suite. The report will briefly explain the SNMP/MIB/ASN.1 protocols and languages, but the reader should have some prior knowledge of the underlaying TCP/IP stack to be able to assimilate from this thesis.

References

[1] **SNMP, SNMPv2, SNMPv3 and RMON 1 and 2, the third edition by William Stallings.**

The chapters concerning SNMP, SNMPv2, MIB, MIBII and ASN.1. Where ASN.1 (Abstract Syntax Notation one) is the formal language in which a MIB is defined. This book is the main source about SNMP, MIB and ASN.1 for this thesis. The book is well written and fairly easy to read and understand if some tables and computer dumps are excluded.

[2] **MIB for an SNMP based maintenance of an RPG in an AXE10, by Pernilla Jansson and Chaowana Kusonkhum (bachelor's thesis)**

A real case study about implementing a MIB in a Group switch (RPG). The project only models a few objects. The whole process from theory until compiled code on this particular hardware is described.

[3] **Getting to know AXE**

An introduction textbook about the AXE system and how different modules, software and hardware, in an exchange interact. A well written book but with a lot of abbreviations of the different modules that are hard to keep track of.

[4] **Operation and maintenance of stand alone platforms.**

An Ericsson internal report written by staff at Ericsson Utvecklings AB in 1998. An internal study performed to explore on which techniques UAB should look into for the future. The main idea is that Ericsson should start exploring open systems suites to use in their future platforms.

[5] **Network Management by Ponthus Nyrelli.**

First part of a diploma thesis. Mainly about SNMP and a small introduction of DMI (Desktop Management Interface). An introduction to networking and an overview of SNMP. The news is the DMI part which is interesting but not very significant to my work.

[6] **Slides from the Network Management Course 2g5552 at KTH by Volker Lausch.**

A run through of network management with, mainly, SNMP, MIBs etc. Good to keep oneself updated about the general picture of networking but of course not very deep. URL: http://www.it.kth.se/edu/Ph.D/NM99/

[7] **RFC 1155, Structure and Identification of Management Information for TCP/IP-based Internets, by M.Rose and K.McCloghrie.**

The standard MIB definition. The RFC explains the MIB structure, the MIB tree, MIB types available and some macros for defining new MIB objects.

[8] **RFC 1212, Concise MIB Definitions, by M. Rose and K. McCloghrie.**

Mainly an explanation on how to construct tables by using the allowed MIB formats.

[9] **RFC 1213, Structure and Identification of Management Information for TCP/IP-based Internets MIBII, by M.Rose and K.McCloghrie.**

All updates and additions from the standard MIB to MIBII. All the new objects are also described using the formal language ASN.1.