Diploma Project

Manage Telecommunication equipment using web services

Task book

Final Version

Professeurs:
Philippe Joye
François Buntschu

Mandatory:
Daniel Gachet

Expert:
Nicolas Mayencourt

Students:

Thierry Kiki
David Schneider

Table des matières

1 Introduction

The company KEYMILE wishes a utility to manage its next generation telecommunication equipment with a system using web services. Actually, the management of the object model is performed either over an embedded command line (ECLI), syslog, SNMP or with the exchange of proprietary XML messages (KM-KOAP). The aim of this project is to find standardized solutions using web services (MMI) or to offer access via a web browser (HMI).

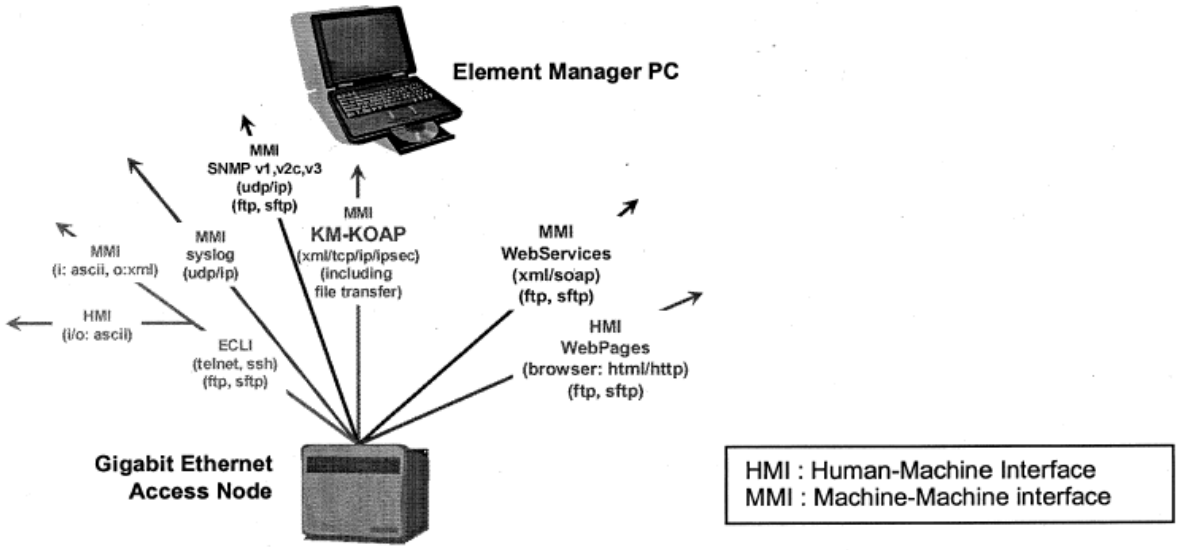

Fig. 1: MILEGATE management interfaces

2 Descriptions of project

As this project needs to be adapted to the existing system, we need to respect a few constraints.
In the following image, the relations between the KEYMILE file describing the internal object model (SFD, XML) and the AccessPoint Definition File ADF (proprietary, XML).

2.1 Actual State

At the moment, the management system uses ADF which is a collection of SFD and describes one single unit in the MileGate. The core unit and access point of the MileGate is in slot 11.

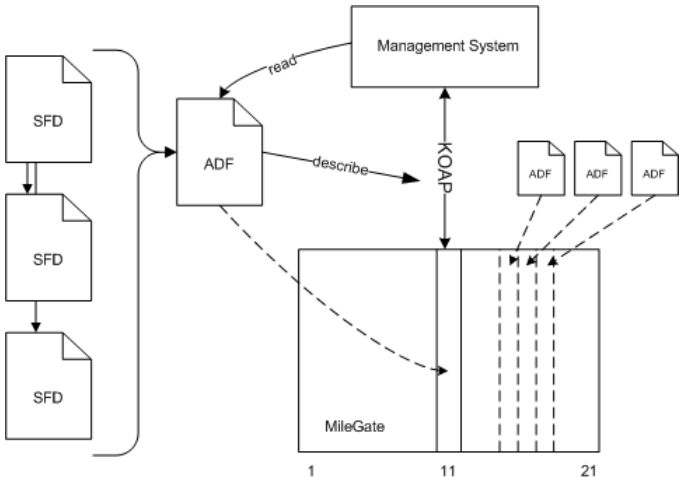

Fig. 2: Existing system

If we represent the actual communication more in detail, we see how the existing management utilities access the MileGate Object Model.

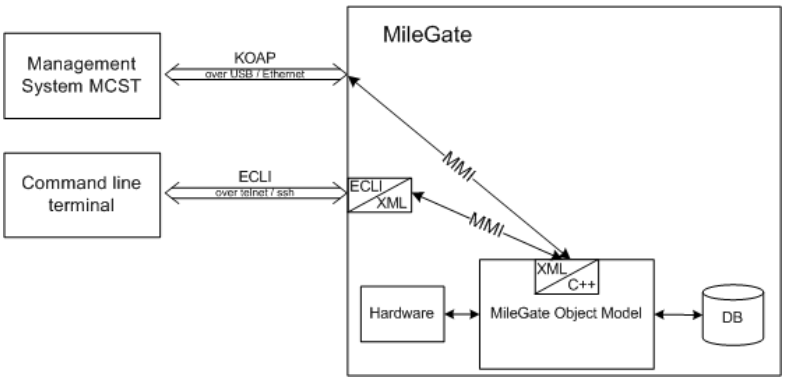

Fig. 3: Existing access methods

2.2 Work to perform

We have two new approaches for accessing the MileGate Object Model. The tasks to perform are represented in red.

2.2.1 Machine-Machine Interface (MMI)

The Web Service Description (WSDL file) which would finally be created will be the input for the client framework. The framework will generate code (for example Java, C, C++, Perl, Pyton, PHP, ..) automatically according to the constraints defined in the web service description.
The messages of the type SOAP (transported over HTTP) are treated within the embedded HTTP server and afterward transformed from SOAP-XML into the proprietary XML format which is the only accessible interface of the MileGate.

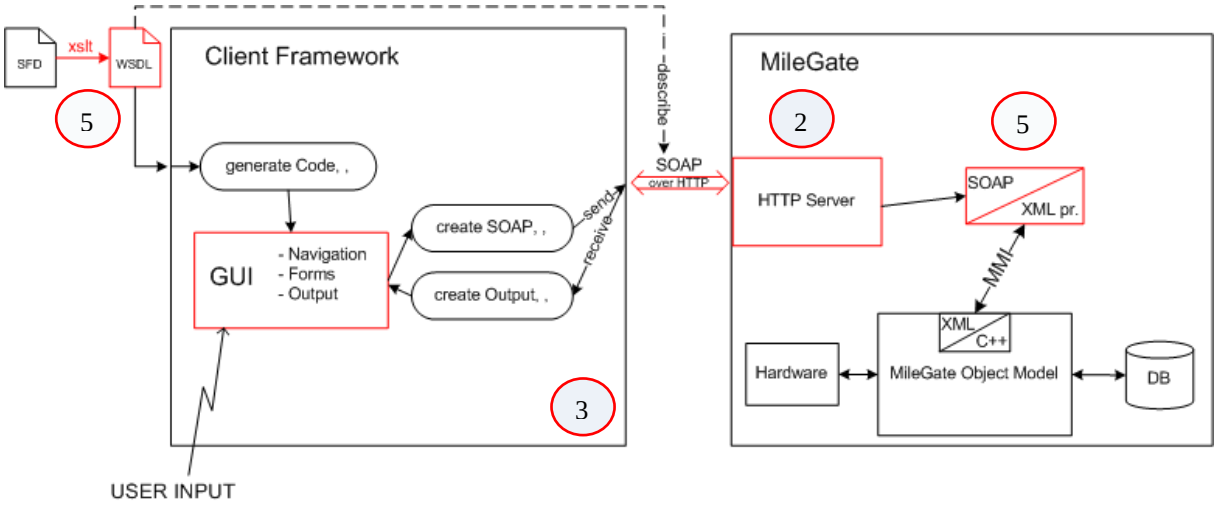

Fig. 4: Approach with web service 4 6

2.2.2 Human-Machine Interface (HMI)

To be accessible by humans, the MileGate should provide HTML files generated at runtime. Therefore, we connect the HTTP Server not with the Client framework but with a web browser.

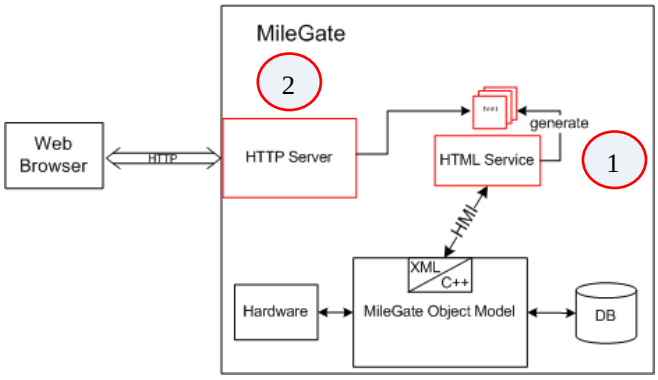

Fig. 5: Approach with generation of HTML files

For both approaches the MileGate Object Model manages the call of the C++ routines from the proprietary XML messages. In addition, it communicates with the hardware and affects storage/request of data.

3 Objectives

3.1 Side issue

- Find a good way to generate on the fly HTML pages within the MilGate which providing a web browser access. 1
- Survey and evaluate different embedded HTTP servers running on Linux and/or VxWorks for the MilGate. 2

3.2 Main issue

- Survey and evaluate different client Frameworks and describe their compatibilities with the web services. 3
- Describe the flow of information from the KEYMILE files which describes the internal structure through the embedded HTTP Server to the MileGate 4
- Define the web service and the necessary transformation. 5
- Implement a prototype using the web service (MMI). 6

4 References

Keymile' documents:

Introduction to the MileGate XML - Management Interface
Web Services Interface for Milegate
Web Services - Personal documents of Marcel Bellorini

Specifications:

WSDL 1.1, www.w3.org/TR/wsdl
WSDL 2.0, www.w3.org/TR/wsdl20
SOAP 1.1, www.w3.org/TR/soap
SOAP 1.2, www.w3.org/TR/soap12
XSLT 1.0, www.w3.org/TR/xslt
XSLT 2.0, www.w3.org/TR/xslt20

Books:

- Programming Web Services with SOAP (Jamel Snell, O'REILLY)
- Web Services : Principles and Technology (M. P. Papazoglou, PERSON)
- Services Web avec SOAP, WSDL... (J-M Chauvet, EYROLLES)

5 Initial Planning