1. User side signalling (USI + ISP + ISPIF)

Authors: Timo Pärnänen, Juhana Räsänen

The purpose of user side signalling is to provide user applications a Q.2931 signalling interface to ATM network. This was developed as a part of TOVE project, because no such implementations were available for Linux environment at the end of 1996. User signalling was done quite quickly and only for demonstration purposes, so only a brief description is given in this document.

1.1 1 Introduction

User signalling is implemented as a daemon that runs in a client machine. The daemon opens and listens the signalling VC (VPI 0, VCI 5) for communication with the switch controller signalling protocol stack. The user signalling daemon consists of four parts: SAAL (Signalling ATM Adaptation Layer), user side Q.2931, USI (User Signalling Interface) and ISP (Internal Signalling Protocol).

SAAL is the same as in switch controller side (CPCS+SSCOP+UNI-SSCF) and user side Q.2931 is only a slightly modified version of network side Q.2931. USI is the user side equivalent of the Call Control of the switch and it was designed and developed in TOVE project. ISP is defined in the Linux ATM documentation and it is the part of the signalling daemon that communicates with Linux kernel. When a user wants to open a SVC (Signalled Virtual Circuit), kernel translates the system call to an ISP message that is sent via a special socket to the signalling daemon. ISP transforms this message to an OVOPS++ messenger and send it to USI for processing.

1.2 2 Architecture

User signalling protocol stack is presented in Figure 1. The relations between protocols and the interfaces can be seen, as well as how the signalling daemon connects to the kernel with two sockets.

USI is a full-featured protocol with six-state FSM, but it does not communicate peer-to-peer with the other end, only drives the Q.2931 protocol. ISP is a stateless protocol whose function is to convert ISP messages from the kernel to OVOPS++ messengers and vice versa.

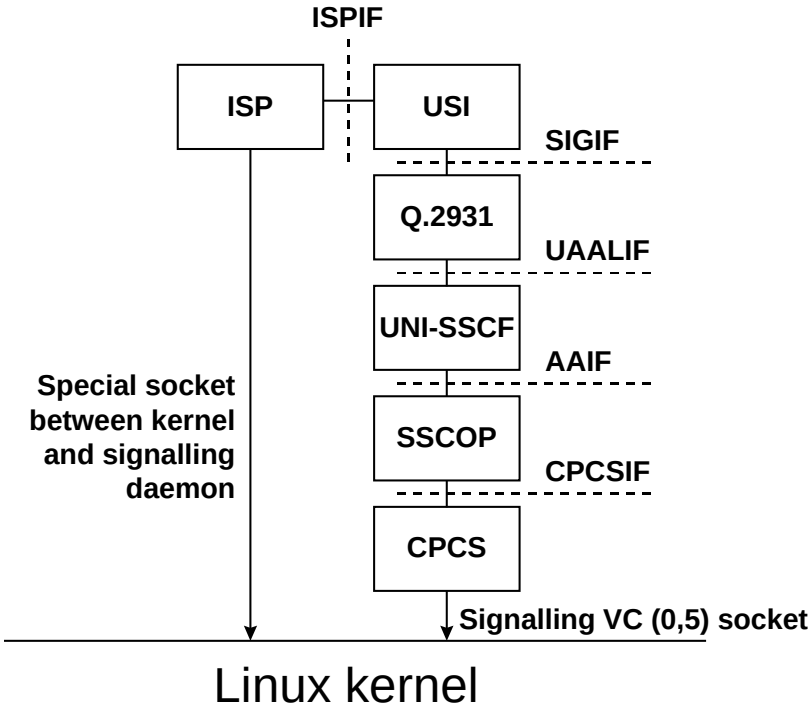

Figure 1 User signalling protocol stack

1.3 3 Implementation details

No detailed description of the implementation is given.

1.4 4 Features implemented

Only a very minimal set of user side signalling functions is implemented to support initial tests of the switch controller software by setting up and tearing down connections. For example, the address management (eg. with ILMI, Integrated Local Management Interface) is not supported, so applications must be given the local ATM address of the workstation “by hand”.

1.5 5 Known bugs and flaws

Probably many due to hasty development of the user side signalling.

1.6 6 Future development

There are no plans to improve user side signalling further, as it serves its purpose in its current form. Because user signalling is usually bundled with the ATM software package of the end system, no resources of TOVE will be sacrificed for such development. (Linux ATM doesn’t include Q.2931 signalling but only UNI 3.x, so a minimal Q.2931 signalling daemon became necessary for testing.)

1.7 7 Statistics

The development was done within about two weeks in December.

Activity	Research	Design	Coding	Reviews	Total
Duration	10	10	50	0	70

Table 1 Duration of activities

Lines Of Code (LOC)	Number of files	Number of classes
2006	34	25

Table 2 Metrics

1.8 8 References