

데이터망을 통한 교환기 소프트웨어 원격 적용 시스템

민경원, 오원욱, 편용현, 정태진

한국통신 서울통신운용연구단

Tel. 02-526-6237, FAX 02-526-6814, kwmin@kt.co.kr

An Implementation of ESS Software Remote Application System

Kyong Won Min, Won Ook Oh, Yong Hyun Pyun, Tae Jin Chong

Korea Telecom Seoul Telecommunication O&M Research Group

Tel. 02-526-6237, FAX 02-526-6814, kwmin@kt.co.kr

Abstract

Software release over the network is the most efficacious way of software update. Software release of electronic switching system, considering its environments, is the typical case that is in need of adopting the method. The media of transferring software and data of TDX-10A electronic switching system are magnetic tapes containing them, so the MT must be delivered between remotely placed office and package generation system. SMART-10A system provides the file transfer on the data network between TDX-10A and the package generation system and ability of remote application of new software. This paper shows the implementation of file transfer between TDX-10A and computer system connected by data networks, status handling for duplicated system, and monitor process design with IPC on UNIX system.

I. 서론

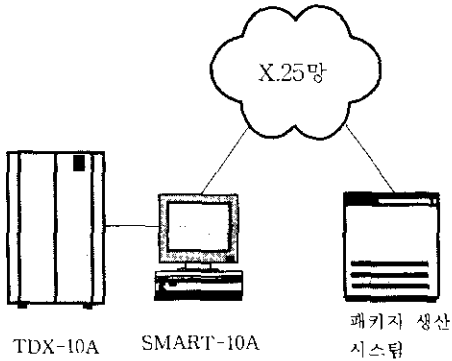
소프트웨어의 개발 자체에 관한 많은 연구와 관심에 비하여 개발된 소프트웨어의 적용의 제 문제에 대한 관심은 매우 적다고 할 수 있다. 지역적인 산재된 시스템의 소프트웨어를 변경하는 작업은 적지않은 비용과 시간을 요하나, 소프트웨어의 적용은 네트워크의 출현이후에도 상당기간 이를 이용하지 못하였다. 그러나, 최근 네트워크를 이용하여 새로운 소프트웨어 버전을 배포하는 방식이 점차 일반화되고있다. Internet 상에서 소프트웨어를 다운로드(Down Load)하여 사용하는 것이 좋은 예이며 이는 사용자 요구에의한 변경에서 자동변경으로 이룰다시 개발시 버전 제어와 연계하고 있는 추세이다. 이를 통하여 경제적인 소프트웨어 변경이 가능해지고있다.

교환 시스템은 지역적인 산재성과 다양한 서비스를 위한 빈번한 소프트웨어 변경 측면에서 네트워크를 통한 소프트웨어 적용이 강력히 요구되는 시스템이다. 본고는 전자교환기에 네트워크를 이용한 소프트웨어 적용 방식의 예이며, 이는 TDX-10A 전자교환기 상에 구현하였다.

II. 시스템 구성

TDX-10A 교환기는 다수의 프로세서를 갖는 분산제어 시스템이다. 프로세서에 로딩(Load)되는 교환기의 소프트웨어는 OS(Operating System), 응용 프로그램 및 DB(Data Base)로 구성되어 있으며 각각은 화일로서 관리된다. 이들 소프트웨어 화일들은 OMP(Operation and Maintenance Processor)가 관리하는 디스크에 저장되며 요구 시에 이로부터 각 프로세서로 해당 프로세서의 화일이 로딩되고 운용 중 각 프로세서로 부터 DB 변경 내역을 받아 이를 다시 디스크 내 DB에 반영하는 방식으로 운용된다. 디스크는 SCSI Bus로 MT 드라이버와 정합 되어 있다. 이 Bus를 통하여 디스크 화일을 MT로 백업하고, MT의 화일을 디스크로 덤프할 수 있다. TDX-10A S/W 온라인 변경 기능(Software Modification for Remote TDX-10A,이하 SMART-10A)의 기본기능은 기존의 MT를 이용한 화일 이송을 데이터망을 통한 화일 송수신으로 대체하는 기능이며 일련의 작업을 현장 및 원격지에서 수행, 감시할 수 있는 기능을 제공한다. 이러한 기능을 구현하기 위해 교환기와 데이터망 정합에 SMART-10A 시스템을 그 연동 매체로 사용하며, 기본 구성은 <그림1>과 같다.

SMART 시스템은 기존의 CAMA(Centralized Automatic Message Accounting) TDX-10A장치인 CDT(Charging Data Transmitter)를 활용하여 구현되었다¹⁾. SMART-10A 기능은 TDX-10A 교환기, CDT 시스템에 화일송수신, CDT와 원격 시스템 간의 화일송수신 기능이 기본적인 기능이며 다음 절 에서 각각의 구현 내용을 살펴본다.

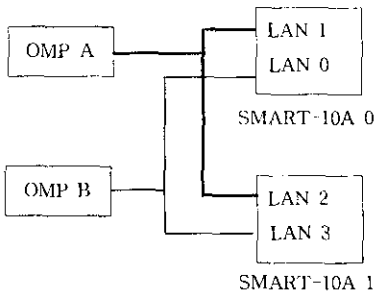


<그림 1> SMART-10A 구성

III. 교환기와 SMART-10A 정합

3.1 LAN 구성

CDT는 이중화 되어있으며, 다시 이중화된 OMP와 LAN(IEEE802.3)으로 정합되었다. 그 구성은 <그림 2>와 같다. OMP와 CDT는 기능상의 중요성으로 이중화 되어 있으나, SMART-10A의 경우 CDT 시스템의 과금기능에 영향과 부하를 최소화 하기위해 항상 CDT Standby측에서 동작하기위해 이중화되어 동작된다. 그러나, CDT 장치가 이중화 운용되지 못하는 경우 SMART, CAMA 기능이 한 시스템에서 공동 운용될 수 있으며, 각 상태는 자동으로 감지, 제어된다.



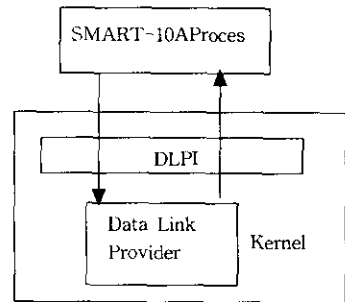
<그림 2> LAN 정합도

TDX-10A의 OMP는 각각 A, B로 이중화 되어 있으며, MCIA(Message Communication Interface Board Assembly)를 통하여 LAN에 정합된다. SMART 시스템은 일반적인 범용 LAN Card를 사용한다. 이중화를 위하여 교환기측 OMP A,B 에 각각 LAN이 MCIA에 연결되며, Active인 OMP에 연결된 LAN이 Active LAN이 된다. SMART시스템은 Active, Stand-by LAN 각각에 대하여 연결 된다. 따라서, SMART의 절

체가 이루어 지는 경우에도 항상 Active OMP측과 LAN 접속이 가능 하다. SMART측에 각 LAN은 SMART0측에 LAN0, LAN1, SMART1측에 LAN2, LAN3로서 관리 된다.

3.2 LAN 정합 프로그램 구현

SMART-10A 기능을 위한 응용 프로그램은 OSI Protocol Stack상의 Data Link Layer의 User로서 구현되며, Data Link Service(DLS) Provider 와 인터페이스는 DLPI(Data Link Provider Interface)를 통하여 이루어진다²³⁾. DLPI는 STREAMS를 근간으로 Kernel-level에서 지원되는 인터페이스 이다. 이러한 인터페이스를 사용함으로써 DLS User는 DLS Provider와 독립적으로 구현 될 수 있다. DLP 정합 개념은 TDX-10A의 OMP에도 동일하게 적용되며, Kernel은 SMART 시스템의 경우 UNIX OS Kernel, TDX-10A의 경우 자체 OS인 CROS(Real time Concurrent OS)이다. LAN 접속 응용 프로그램과 서비스를 제공하는 하부 구조의 관계는 <그림 3>과 같다. DLP 응용프로그램이 요구되는 이유는 OMP상위 Layer의 Protocol구현이 매우 어렵기때문에 Raw Mode의 Frame으로 통신이 이루어지기때문이다.



<그림 3> SMART-10A 의 DLP 정합

IV. 백업 및 덤프 기능

백업 과 덤프 기능은 교환기화일을 데이터망 정합기능이 있는 SMART-10A 시스템과 송,수신하는 기능이다. 이를 위하여 교환기와 SMART 시스템에 기능을 담당할 프로세스 및 각 프로세스간 연동 방식을 설계 구현 하였다. 백업과 덤프는 화일의 전송 방향은 상이 하나 그 제어방식에 있어서는 동일하다. .

4.1 TDX-10A내 OMP상의 백업,덤프 프로세스 구현

TDX-10A OMP내에 구현된 SMART 프로세스는 MCL_status, MCL_smart, SSL_main, SSL_smart_bkup

이다. MCL_status는 상태관리 프로세스이며, 상태의 변경을 MCL_smart 프로세스에 전달한다. LAN을 통한 백업, 덤프 요구는 MCL_smart 프로세스에 전달 되어 다시 SSL_main 프로세스로 재전달 된다. SSL_main 프로세스는 MMC, LAN으로 부터의 요구에 따라 각각 덤프요구시 SSL_smart, 백업 요구시 SSL_smart_bkup 프로세스를 활성화 한다. SSL_smart_bkup 프로세스는 백업시 교환기측 프로세스로서 수행된다.

교환기측에서 백업, 덤프를 수행은 교환기 명령에 의하여 기동될 수 있으며, 진행상황이 운용자 터미널에 출력된다. 또한, 운용자는 명령에 의하여 Active측 SMART의 프로세스 상태, 가용 디스크 공간을 출력할 수 있다.

4.2 SMART 시스템의 프로세스

SMART 시스템 내에 구현된 프로세스는 smart, smart GUI, smart command 이다. 이들간의 계층적 구조 및 IPC는 <그림4>와 같다.

GUI process	Command process
FIFO	Shared Memmory
SMART	
DLP	

<그림 4> SMART 시스템 프로세스 구조

1) smart 프로세스

백업, 덤프, 상태관리 및 LAN 정합 기능을 수행하는 주 프로세스이고 GUI 및 Command 프로세스는 사용자 정합 프로세스이다. 명령에 의하여 기동되며 동작 환경 초기화 및 인식이 완료되면 GUI 프로세스를 기동한다. 따라서, GUI 프로세스는 smart 프로세스의 자 프로세스로 기동되고 동작한다. 수행 되는 모든 작업 내역은 기록을 위하여 Log 파일로 저장된다.

2) GUI 프로세스

X Window 응용 프로그램으로 UNIX 환경에 친숙하지 않은 사용자가 SMART-10A 기능을 수행할 수 있도록 버튼 선택에 의하여 모든 작업이 가능하게 구현되었다. 작업 수행중에는 이를 감시, 중단 할 수 있는 기능을 제공한다.

3) Command 프로세스

Command Process는 데이터망을 통하여 접속, SMART-10A 기능을 수행, 감시하기 위한 프로세스이다. 명령창과 출력창으로 구분되어 동작되며, 원하는 기능의 해당 번호를 입력하여 기능을 수행한다. 원격지에서 데이터망을 통하여 SMART 시스템에 접속후

기동시킬 수 있다. smart 프로세스가 작업중에도 기동되어 작업의 진행 상태를 감시하고 현작업을 중단할 수 있다.

4.3 파일 관리

LAN을 통한 파일 전송은 1K Byte 단위로 이루어진다. OMP와 SMART 시스템내의 프로세스는 Hand Shake 방식으로 파일과 제어 패킷을 주고 받으며, 파일전송 중 발생하는 오류에 대한 정보를 교환 하고 오류 발생 시 재 전송한다. smart 프로세스는 백업의 경우 수행 시각을 이름으로 하는 디렉토리에 파일을 저장한다. 해당 디렉토리에는 백업을 수행한 주체, 즉 GUI, Command 프로세스 및 교환기, 백업 옵션, 백업된 파일 정보를 저장한 파일을 Log파일로서 관리한다. OMP측 프로세스는 별도의 Log파일을 유지하지 않는다. 덤프의 경우 운용자가 지정한 디렉토리내의 파일을 교환기로 전송한다.

V. 상태 관리

기동시 CDT상태를 검사하여 Standby측에서 기동하기 위해 수행된다. 따라서, 능동적으로 상태를 제어하지 않고 CAMA 이중화 상태에 따라 수동적으로 동작 되도록 구현되었다.

5.1 교환기측 상태관리

1) 기능 개요

SMART-10A에서는 CAMA에서 사용중인 장비를 항상 파악 하고 있어야 한다. 상태관리를 담당하는 것은 MCL_status 프로세스는 초기 시동시, 또는 LAN 상태 변화시 상태변화를 인지하여 active 및 standby LAN을 관리하는 데이터 베이스를 수정한다. 그리고 standby LAN의 변화시에는 LAN에서 데이터 수신을 담당하는 MCL_smart 프로세스를 중지시키고 변화된 SBY LAN에 MCL_smart를 재 시동한다.

2) 메시지 천이

생성된 MCL_smart 프로세스는 자기의 이더넷 어드레스를 메시지에 담아 송신하고, 지속적으로 수신된 데이터를 감지한다. 만약 수신된 메시지가 있으면 수신된 메시지별로 교환기 내부 메시지'인 IPC 메시지로 구성하여 작업 수행 시작을 제어하는 SSL_main 프로세스로 보낸다.

3) 프로세스간 통신

SMART의 절체는 OMP 프로세서의 절체에는 영향을 미치지 못하므로 SMART_daemon과는 메시지 교

환을 하지 않는다. 하지만 사용하는 LAN의 어드레스를 통보하여야 하므로 SMART_daemon에 LAN을 통하여 메시지를 보낸다. 그리고 SMART로부터 덤프나 백업 기능 수행 시작을 알리는 메시지는 MCL_smart 프로세스에서 수신하고 IPC 메시지 형태로 SSL_main에 전달된다.

5.2 SMART 시스템에서의 상태관리

1) 이중화 관리

smart 프로세스는 기동시 LAN을 통하여 교환기와 상대측 smart 프로세스로부터 OMP 및 SMART 시스템 상태 정보를 전송받는다. CDT가 이중화 상태가 아닌 단독 운용인 경우 smart 프로세스는 기동된다. 그러나, CDT 시스템이 정상작인 이중화 상태인 경우 현재 기동이 시도되는 CDT의 동작 모드가 CAMA Active 시스템인지 여부를 판단하여 CAMA Standby인 경우 SMART Active 모드로 동작한다. CAMA Active인 경우 SMART Standby 모드로 기동된다.

2) 동작상태 관리

smart 프로세스는 OMP와 10초마다 상태를 교환하고 MCIA로 부터 3초 주기로 LAN 상태 감시 정보를 수신한다. 교환기로 전송하는 정보는 동작상태와 디스크 가용량을 포함한다. 교환기 운용자는 SMART 상태 출력 명령어를 입력하여 이를 출력할 수 있다.

3) 상태 출력

변경된 상태는 GUI 메시지창에 출력되고, GUI 상태창의 출력에 반영된다. 상태창은 SMART-10A의 이중화 상태, LAN 상태 및 Active OMP를 항상 표시하고 있는데 정상인 경우 초록색으로 표시되며 비정상 상태가 발생된 감시대상은 붉은색으로 점멸되어 사용자가 즉시 인지, 조치할 수 있도록 구현되었다.

VI. SMART-10A와 운용보전망 정합

운용보전망은 교환기, 주변장치 관리 및 CAMA용의 한국통신내 X.25망이다. CDT 장치는 CAMA 화일 전송을 위하여 운용보전망 정합기능이 기 구현되었으며 SMART 시스템은 이를 이용한다.

SMART 시스템은 FTAM(File Transfer, Access and Management) 프로토콜을 통하여 운용보전망에 접속된 시스템과 화일을 주고받을 수 있다⁴⁾.

VII. 결론

SMART-10A 개발을 통하여 TDX-10A 프로그램 적용을 원격지에서 수행, 감시할 수 있게되었으며, 이를

활용하여 기존에 소프트웨어 변경이 일정 주기의 Version Release 개념에서 보완 소프트웨어 및 신규 서비스의 수시 적용 개념으로도 변경이 가능 할 것으로 사료된다. SMART-10A는 TDX-10A 교환기 소프트웨어 관리 시스템, 화일 전송을 통한 운용국 데이터 검증 시스템, TMN, 등에 활용이 기대된다.

지역적으로 산재된 교환기의 소프트웨어를 제작 적용하는 일련의 과정은 상이한 시스템들의 존재, 지역적인 분산, 요구 서비스의 동적 변경 및 다양성, 많은 인력이 동시에 소프트웨어 변경에 관여등의 측면에서 교환 시스템의 소프트웨어를 관리하는 문제는 분산 시스템의 제어, 데이터 관리, 소프트웨어 버전 제어 (Version Control)의 재 문제가 적용 됨을 알 수 있다. 그러나, 현재 까지 소프트웨어의 변경요구 관리, 구현 과정 및 적용자체는 오히려 새로운 개념과 기술의 적용을 받지 못해 왔다고 할 수 있다. 효율적인 프로그램 개발과 적용을 위해 새로운 개념의 도입에 대한 보다 많은 관심과 연구가 요구된다.

참고문헌

- [1] 한국통신, "전자교환기 집중과급처리 시스템 개발," 한국통신, 1996
- [2] OSI Work Group, "Data Link Provide Interface Specification," Unix International, 1991.
- [3] International Organization for Standardization, "Data Link Definition for Open System Interconnection," DIS 8886, 1987.
- [4] Stevens, "UNIX Network Programming," Prentice Hall, 1998.