

전자교환기 DATA BASE 구성에 관한 고찰

김철규, 김창수

대우통신 교환기 개발본부 제 1 개발실

A STUDY ON THE DATA BASE STRUCTURE OF ELECTRONIC SWITCHING SYSTEM

CHUL KYU KIM, CHANG SOO KIM

SWITCHING SYSTEM R & D LAB. 1 DAEWOO TELECOM.

ABSTRACT

Nowdays switching software designs are based on the concept of maximizing efficiency. This idea is to put through the most call, in the fastest time, using the fewest possible resources. So the memory usages are embossed one of the most important part to be considered in the switching software. This paper discusses the general concepts of switching software at first, then describes the switching data base from the view point of the access time, memory efficiency, and call processing environment.

1. 서론

현대 사회가 산업사회로 부터 정보화 사회로 변모해 가면서 이에따른 종합 통신망 구축의 필요성 대두로 정보화 사회의 핵심인 전자교환기 기술의 중요성은 더욱더 점점 하고 있다. 교환기를 대상으로 한 System 공학이라 일컬어지는 교환기 기술이 전기 기계식에서 핵심 제어부에 Computer를 내장한 축적 Program 제어 (SPC) 방식으로 전자화 하면서 교환기 기술은 Computer와 반도체 그리고 Software 기술등의 총아로 등장하고 있다. 본 고에서는 이러한 SPC 방식을 원용하고 있는 전자교환기의 일반적인 Switching Software 구조를 살펴보고, 전자교환기 Data base 구성을 토대로 Access time, memory 운용의 효율성, System의 증가요인, 그리고 Call Processing 수행의 관점에서 분석한다.

2. Switching Software 설계시 고려 사항

오늘날 호처리능 최소화 Resource를 통해 가장 신속하게 모든 호처리능 가능하게 하는 효율적 구대화라는 방향으로 나아가고 있다. 따라서 Switching software implementation 시 다음과 같은 요인을 고려해야 한다.

1) Processor Speed

교환기 System 제어기에 Instruction Cycle 은 전체 교환 System 이 단위시간에 얼마나 많은 일을 할 수 있는가의 지표가 된다. 초기 교환 System 에 비해 오늘날 교환 System 의 Cycle time 은 10배 이상 빨라졌고 이는 오늘날 교환기가 같은 시간에 초기 교환기에 비해 기약 10배 이상의 일을 할 수 있음을 의미한다. 그러나 실제적으로 이 Processor speed 가 Switching capacity를 결정하는 모든 요인은 되지 못한다. 왜냐하면 한 교환기에 있어 다양한 종류의 Requirement 가 있게되고 이에따라 Switching capacity 를 결정하는 요소를 또한 다양하게 되므로 전체적인 System capacity 는 Processor speed 를 비롯한 여러요인에 의해 결정되기 때문이다.

2) Memory Usage

교환기의 Memory 는 Fixed memory 와 Variable memory 로 구분될 수 있으며 양자에 대한 배분은 Processing trade off 와 cost 를 고려하여 결정하여야 한다. Fixed memory 란 하나의 호를 진행시키기 위해 필요한 최소한의 Program 과 data 를 갖고 있는 Memory 이며 Generic program 과 Office data 의 일부가 이에 속한다. 이에 반해 Variable

memory는 가입자적 종류나 System의 Capacity에 관련된 Data를 갖고 있는 memory로서 가입자 정보, Equipment정보가 이에 속하며 한 전화국의 가입자 수나 중계선 수가 늘어날수록 Variable memory size는 증가하게 된다. 이러한 memory는 system cost에 중요한 요소로 작용하여, 교환기 software의 설계자는 적은 양의 instruction code를 갖는 program과 구조화된 data base를 통하여 memory효율의 극대화를 추구함을 장시 고려하고 있다.

3) Program Design

교환기 program작성시 지나치게 적은 양의 instruction code의 사용을 고집하다 보면 자칫 program이 복잡해지거나 실행시간이 약해질 수 있다. 따라서 program작성은 실행시간이 보장되고 program이 지나치게 복잡해지지 않는 범위 내에서 적당한 양의 memory를 사용하는 것이 가장 합리적인 것이다.

또한 processor의 hardware design 개선을 통한 방법도 고려할 수 있는데 call processing에 자주 사용되는 instruction의 집합을 하나의 instruction화 함으로써 해당 instruction cycle을 줄이는 방법도 있다.

3. Switching Software의 구성

교환기 software는 크게 generic program과 office data로 나눌 수 있다. Generic program이란 switching system의 기능을 수행하기 위한 모든 logical instruction의 집합이고 office data는 generic program에 사용되는 전화국과 가입자에 대한 내용의 data를 말한다. 이러한 software 구조를 대략 도시하면 그림 1과 같다.

1) Generic Program

교환기에 필요한 다양한 function을 제공하는 generic program은 교환국마다 동일한 내용을 지닌 program으로 Operating system, Call processing Maintenance & Administration의 기능을 수행한다. 이는 단일 program이 아닌 많은 program module들의 집합으로서 이들 각 module들은 해당되는 특수한 기능을 수행한다. 이러한 기능들은 executive control program에 의해 activate되며 이들 program들은 buffer와 temporary storage area를 통하여 호처리 및 운용보전에 필요한 message를 교환한다.

2) Office data

Office data는 교환기에 필요한 다양한 data를 제공하는, 교환국마다 그 크기나 내용이 상이한 data base로서 design시 generic program과 독립되어 처리되도록 구성한다.

Office data는 generic program수행시 reference가 되며 customer data와 equipment data로 나눌 수 있다.

- 가입자 line data, 가입자 전화번호 data
- Trunk data, Route data
- Special service feature
- 과금 data 등

Equipment data는 교환국의 hardware configuration에 관계되는 data로서 frame 수, frame 당 circuit 수 addressing 범위, 그리고 교환국 network configuration data가 이에 속한다. 이러한 equipment data를 통해 generic program은 전화국의 physical environment를 결정한다.

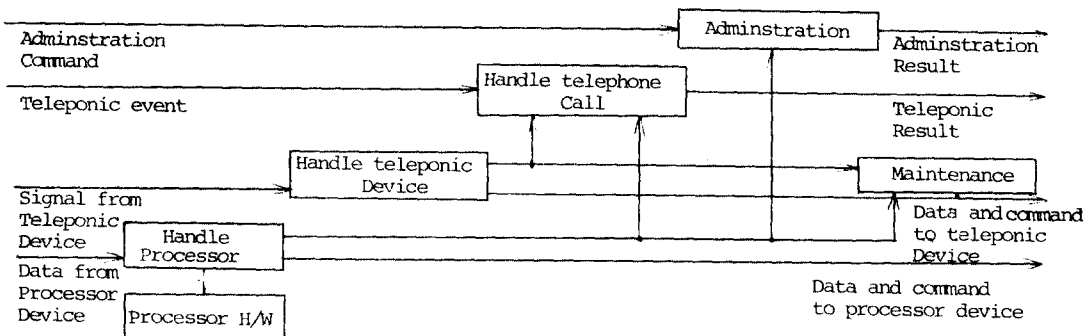


그림 1. Switching Software의 구조

이밖에 office data는 data의 지속성에 따라 permanent data, semipermanent data, 그리고 transient data로 구분할 수 있고 data의 내용에 따라서는 configuration data와 administration data로 구분되며 이것에 대한 설명들은 본 고에서는 생략한다.

Office data architecture는 generic program과의 연관 관계를 최소화 시켜 교환기의 data의 내용이나 크기가 generic program에는 가능한 반영되지 않도록 구성한다. 따라서 office data의 구조를 상호 hierarchical linked list로 구조화하고 (그림 2) 각 linked list의 node는 하위 level이 존재할 때는 하위 level의 pointer를 갖고 하위 level이 존재하지 않을 때는 data를 갖고 있는 table 형식이 되도록 하여 generic program과의 관계를 배제시킨다.

Trunk data, Route data, Subscriber line data 등에 걸쳐 사용된다.

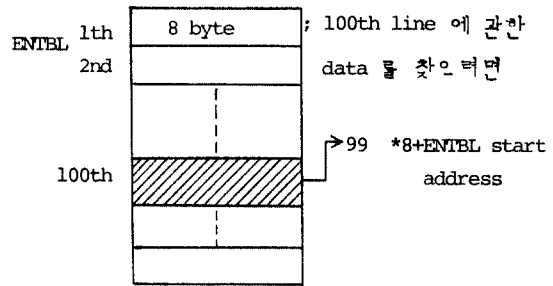


그림 3. Indexed sequential table 구조

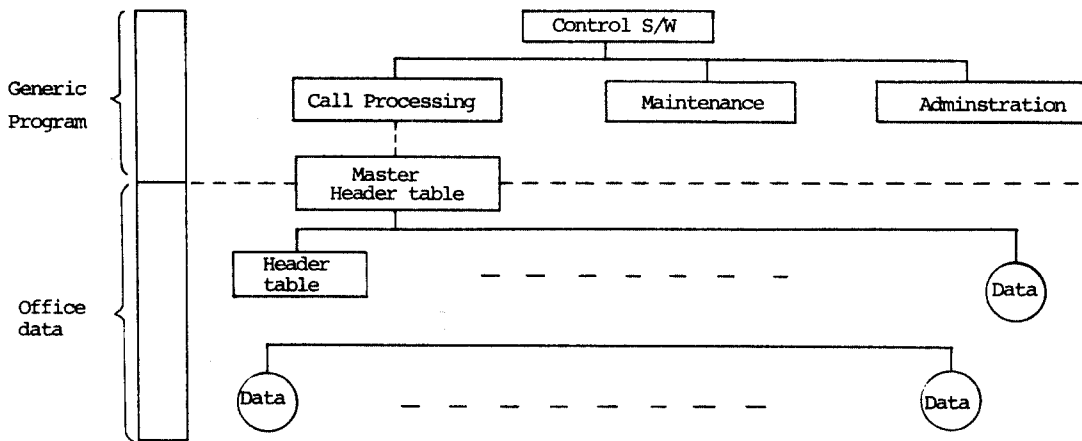


그림 2. 전자교환기 Data 구조

4. 전자교환기 Data base구조

1) 구조

교환기 data 구성은 대개 table 형식으로 link 되어 있고 그 structure에 따라 다음과 같이 배열할 수 있으나 실제적으로는 다음의 구조들을 변형 또는 상호 혼합한 형태로 구성되고 있다.

i) Indexed Sequential Table

이는 data 구성중에 가장 기본적인 형식으로서 그 구성은 그림 3과 같다. 이 형식은 data들이 한 address에서 시작해 순차적으로 배열되어 있고 한 data item의 증감의 폭이 일정할 때 사용되는 구성으로 교환기 data에 Equipment number table, Directory number table,

ii) Level index table

이것은 해당 data를 바로 찾는 것이 아니라 data를 찾기 위한 table을 level 별로 두어 level을 따라가면서 data를 access하는 방식이다. 이는 그림 4에서와 같이 data 추가시만 memory를 할당하게 되므로 memory를 효율적으로 이용할 수 있으며 data의 개수나 크기가 가변적일 때 유용한 구성으로 가입자의 국번, 착신번호 분석 table 등이 이에 속한다.

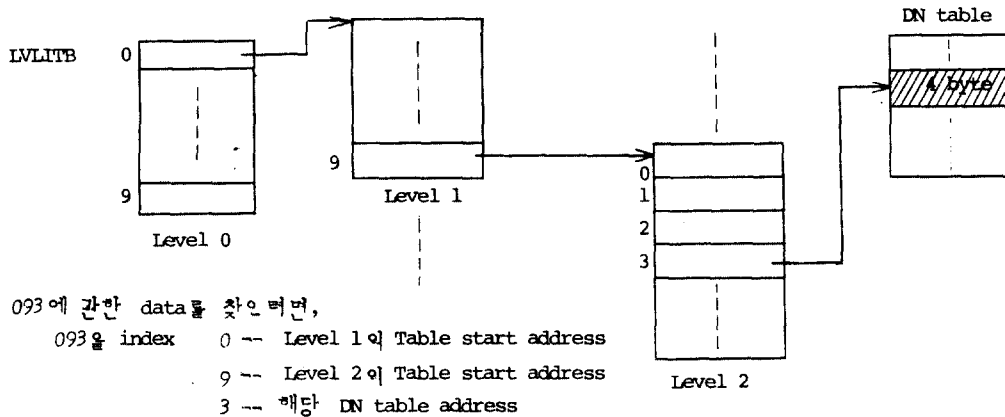
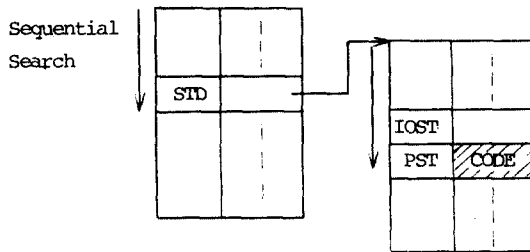


그림 4. Level index table 구조

iii) Search table

Level index시 level 의 수가 증가할수록 level 을 위한 memory 가 증가하게 되고 algorithm 또한 복잡해진다. 따라서 해당 data 종류를 key 로 하여 첫번째 table을 sequential search 한 후 그 table 내의 address로 가서 data 항목을 index 하여 필요한 data나 pointer를 얻는 방식이다.



MML. "STD PST" 의 Search시

Group name STD -- 그 Group name 에 속하는 identifier 들의 table에 start address
 Identifier PST -- 그 Command 의 code

그림 5. Search Table 의 구조

이 방식은 Man machine language code search table, PBX line hunting table 등에 사용되고 있다.
 익에 언급된 세가지 data 구성의 장단점을 기술하면 다음과 같다.

종류	장점	단점
Index Table	- 간단하고 Program 이 쉽다. - Data access 가 신속	- Memory 운용의 비효율성
Level Index Table	- Memory 운용의 효율성 - Data Insert, Delete, Update 용이	- Level의 증가로 access time 증가 - 단일 Key 로만 access - 주기적으로 재구성 필요
Search table	- Memory 운용의 효율성	- 첫번째 table 의 item 이 많을 때 search 가 늦다.

표 1. Data 구성의 장단점

이상의 장단점을 비교해 볼 때 access time 측면에서는 index table 의 이용이 유리하나 memory 의 효율 측면에서는 level index table 이나 search table 의 사용이 이상적임을 알 수 있다. 따라서 software 설계자는 data 의 종류나 그 성격을 고려하여 적당한 data structure 를 갖도록 design 하여야 한다.

2) System 증가요인 발생시 처리

교환기는 set-up 후 가입자, 중계선 수의 증가, 가입자 특수 service 인 등록 및 삭제, 그밖에 data update 등에 의하여 교환기 data base 의 증가요인 (Growth up) 이 계속 발생한다.
 Data base 는 이러한 경우에도 전체 data 구조에 영향을 미치지 않고 능동적으로 처리될 수 있도록 구성되어야 한다.

따라서 교환기의 확장을 위한 재구성은 몇몇 기본 table 이 변경만으로 가능하게 하여 system 을 일반화 시킨다. 그러므로 generic program 이 address 지정시 바로 해당 address 를 지정하여 data 를 access 하는 방식이 아닌 link table 을 이용한 indirect addressing 방식에 의한 access 가 generic program 과 office data 의 독립성이라는 측면에서 바람직하다. 예를 들면 아래와 같다.

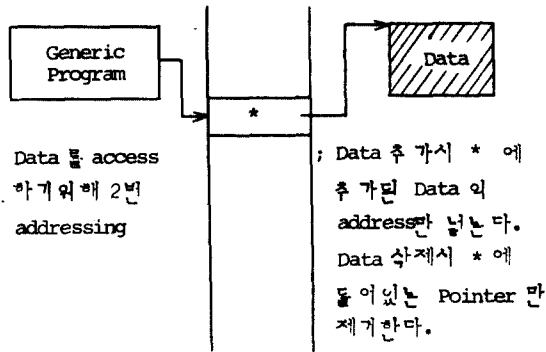


그림 6. Growth Up 시 처리 방식

두 가입자 사이에 speech path 를 제공하는 입면의 과정을 말한다. 이러한 call processing 에는 다음과 같은 6가지 기본적인 기능이 존재한다.

1. Detecting a request for service
2. Interpreting dialed digits
3. Alerting the called party
4. Establishing a path connection
5. Disconnecting the call
6. Charging

이밖에 구간 호처리와 가입자 특수 service 처리등도 있으나 이는 언급된 6가지 기능의 확장이라 볼 수 있다. Call processing 의 구조를 개괄적으로 도시 하면 그림 7과 같다.

이러한 입면의 호처리 과정에서는 generic program 이 호처리에 필요한 data, 즉 service 동굴, Line number, Directory number, Prefix data, Channel data 등을 찾아 분석하여야 하는 translation 과정이 필요하다. 이 과정은 generic program 에서 필요로 하는 data 에 대한 정보들 가장 신속하고 정확하게 generic program 이 찾아내는 과정인 것이다.

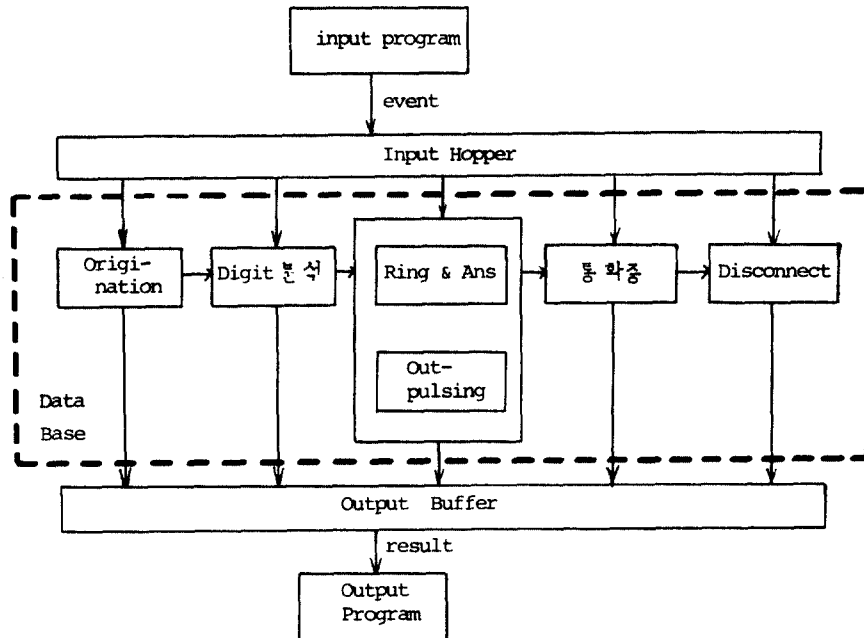


그림 7. Call Processing Software 의 개괄적 구조

3) Call processing environment
Call processing 이란 교환기 내에 있어 인가된

따라서 이러한 translation 에 사용되는 data base 는 translation 에 소요되는 시간을 최소화 시키기

위해 access time 이 빠른 data structure 를 가져야 하며 부수적으로 memory 운용의 효율성 또한 고려되어야 한다.

5. 결 론

본 고에서는 전자교환기 software 의 일반적 구조에 관하여 살펴보고 교환기 data base 구성을 access time, memory 효율성, system의 확장요인, 그리고 call processing 의 관점에서 살펴보았다. 앞으로 교환기가 대형화 하면서 처리하여야 할 data 의 양 또한 상당히 늘어날 것이며 이에 따라 memory 운용을 전문적으로 관리하는 data base management (DBM) 기법이 교환기 software 에서 더욱더 중요한 부분으로 등장하게 되었다. 따라서 이러한 교환기 DBM 은 office data 를 갖고 있는 data base 를 모든 processor 가 access 할 수 있도록 하여주는 하나의 processor 를 자체로 볼 수 있으므로, 모든 processor 는 data base 에 속한 data 구조를 자세히 알 필요가 없이 단순히 간단한 request 만으로 DBM 에 의하여 data base 를 access 할 수 있도록 구성하되 수록되는 data 의 성격에 따른 최적의 data structure 및 access 방식이 충분히 감안된 형식으로 구성되어야 한다고 본다.

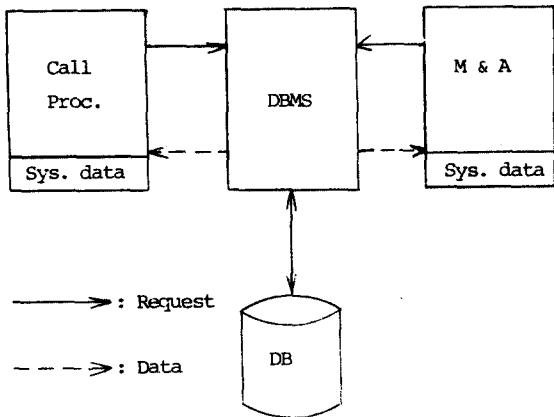


그림 8. DBMS 를 도입한 교환기 S/W 의 구조

참고로 이러한 형식이 고려된 data base management 도입시 기대되는 효과를 열거하면 다음과 같다.

1. 모든 process 에 균일하고 높은 systematic access 가 가능
2. Data design 을 배제한 software design

이 가능

3. Portability 의 향상
4. Data security 의 향상
5. Integrity 와 flexibility 의 향상

참 고 문 헌

1. 전자통신연구소
"전자교환기 개발사업 총괄 보고서"
1985
2. John C McDonald
"Fundamentals of digital switching" 1983
3. David Kroenke
"Data base processing" 1977
4. 이성근, 김영식
"Analog 전자교환기의 data base 구성에 관한 고찰"
1985 교환연구회 논문집
5. A Gardiner, L Katzschner,
C Vander Straeten
"ITT 1230 Software design for digital switching"
1979 Electrical Communication Volume 54
6. M.T.Hills, S.Kano
"Programming electronic Switching System"
IEEE Telecommunication series 1979