

석사학위논문

엔터프라이즈 IP 네트워크 연결 정보
관리 시스템 설계 및 구현

김 은 희 (金 銀 姬)

정보통신학과 (네트워크 전공)

포항공과대학교 정보통신대학원

2007

Design and Implementation of Enterprise
IP Network Connectivity Information
Management System

엔터프라이즈 IP 네트워크 연결 정보
관리 시스템 설계 및 구현

Design and Implementation of Enterprise IP Network Connectivity Information Management System

by

Eun-Hee Kim

Department of Computer and Communications Engineering
POSTECH Graduate School for Information Technology

A thesis submitted to the faculty of POSTECH Graduate School for Information Technology in partial fulfillment of the requirements for the degree of Master of Engineering in the Department of Computer and Communications Engineering.

Pohang, Korea

December 20, 2006

Approved by

Major Advisor

엔터프라이즈 IP 네트워크 연결 정보 관리 시스템 설계 및 구현

김 은 희

위 논문은 포항공과대학교 정보통신대학원 석사 학위
논문으로 학위논문 심사위원회를 통과하였음을 인정합
니다.

2006년 12월 20일

학위논문심사 위원회 위원장 홍원기 (인)

위 원 서영주 (인)

위 원 송황준 (인)

MCC
20052607

김 은 희, Eun Hee Kim, Design and Implementation of Enterprise IP Network Connectivity Information Management System, 엔터프라이즈 IP 네트워크 연결 정보 관리 시스템 설계 및 구현, Department of Computer and Communications Engineering, 2007, 50P, Advisor: J. Won-Ki Hong, Text in Korean.

ABSTRACT

Currently, most of the enterprise networks are relying on man operation, thus it is difficult to apply the modified network topology in real time. This problem causes inconsistency in real network topology and man operated network topology map, which is the main cause for difficulties in network management. Therefore, this thesis searches network devices for effective network management and presents a system that is produced to discover physical connectivity information of the searched network devices.

We use the Bridge MIB to find information of port connectivity between devices. The network devices, which operate in Layer2 refer to the MAC Address of the packet header's destination and send it to the next node. The MAC Address of the destination is learned and saved in the Bridge MIB. Moreover, the device which operates in Layer 3 observes the MAC Address of atTable and saves the connectivity information of devices. This thesis compares and analyzes this information to each MAC Address of the device port to find connectivity information of devices.

This thesis proposes IP matching algorithm of each port to produce a connectivity information management system. Moreover, the algorithm's requirements are analyzed and the system is designed and implemented to propose a method for producing a network topology map that is necessary for network management. This thesis proposes an algorithm which follows the standard for finding network devices as well as connectivity information of devices. Furthermore, the proposed algorithm is implemented and applied to the POSTECH network which validates its role in supporting network management system.

목차

I. 서론.....	1
II. 관련연구	4
2.1 SNMP.....	4
2.2 Bridge MIB	7
2.3 MIB-II의 at Group.....	10
2.4 관련 시스템	11
2.4.1 OpUtils Switch Port Mapper	11
2.4.2 CISCO Network Assistant.....	12
2.4.3 IBM Netcool/Precision (Tivoli software)	13
III. 시스템 요구사항	14
IV. 시스템 설계.....	17
4.1 포트별 IP 매칭 알고리즘.....	17
4.1.1 네트워크 장비 스캐닝 알고리즘	18
4.1.2 네트워크 장비 그룹핑 알고리즘	20
4.1.3 네트워크 장비별 정보 수집 알고리즘.....	22
4.1.4 네트워크 장비의 포트별 IP 매핑 알고리즘	27
4.2 Database 설계	30
4.2.1 Database Schema Diagram	30
4.2.2 User Table	31
4.2.3 IP_LIST Table	32
4.2.4 Device_LIST Table	32
4.2.5 IfTable Table.....	33
4.2.6 VLAN Table	34
4.2.7 BasePort Table	35
4.2.8 Bridge Table	35
4.2.9 Connectivity Table.....	36
4.3 시스템 구조	37
4.3.1 장비 스캐너(Device Scanner).....	38
4.3.2 장비 그룹퍼(Device Grouper).....	38
4.3.3 장비 정보 수집기(Device Information Collector).....	39

4.3.4	연결 정보 작성자(Connectivity Mapper)	39
4.3.5	구성 관리자(Configuration Manager)	39
4.3.6	Web Server	39
V.	구현	41
5.1	개발환경	41
5.2	Configuration file.....	42
5.3	네트워크 장비 스캐너	43
5.4	장비 그룹핑	44
5.5	네트워크 구성도	45
5.6	장비의 상세 정보.....	47
VI.	결론 및 향후 과제	49
	참고 문헌	51

그림 목차

그림 1. SNMP 네트워크 관리 구조.....	5
그림 2. Bridge MIB의 구조.....	7
그림 3. MIB-II의 at Group의 구조	10
그림 4. 포트별 IP 매칭 알고리즘에 대한 순서도.....	18
그림 5. 관리 대상 장비의 IP 리스트를 생성하기 위한 순서도	19
그림 6. 장비 리스트 중 동일한 장비를 그룹핑하기 위한 순서도.....	21
그림 7. 포트 매칭을 위한 장비별 정보수집 순서도	23
그림 8. 장비의 포트별 연결 정보를 생성하기 위한 순서도.....	28
그림 9. 데이터베이스 스키마 다이어그램.....	30
그림 10. 전체 시스템 구조	37
그림 11. 장비 스캐닝 수행 후 결과화면	43
그림 12. 장비 그룹핑 수행 후 결과화면	44
그림 13. 장비 리스트 화면에서 대표 IP 변경 후 결과화면	45
그림 14. 네트워크의 포트별 연결 정보 화면	47
그림 15. 특정 네트워크 장비의 상세 정보 화면	48

표 목차

표 1. SNMP 버전별 특징	6
표 2. sysServices에 따른 장비의 분류.....	20
표 3. ifTable의 예시	25
표 4. dot1dTpFdbTable을 ‘public@1’로 읽은 결과.....	26
표 5. dot1dTpFdbTable 을 ‘public@10’으로 읽은 결과.....	27
표 6. 사용자 정보 테이블 항목.....	31
표 7. IP_LIST 테이블 항목	32
표 8. Device_LIST 테이블 항목	33
표 9. IfTable 항목.....	34
표 10. VLAN 테이블 항목	34
표 11. BasePort 테이블 항목	35
표 12. Bridge Table 항목.....	36
표 13. 연결 정보 테이블 항목	36

I. 서론

현재는 네트워크 중심의 컴퓨팅 시대로 정보의 공유와 교환이 네트워크를 중심으로 이루어지고 있다. 대부분의 기업, 학교, 공공기관 등은 갈수록 빠른 네트워크를 요구하는 시대에 발맞춰 수십 개의 스위치나 라우터를 연결하여 고속 LAN을 구축해서 네트워크 서비스를 제공하고 있다.

종이문서를 들고 결제를 받는 것에서 전자결제 시스템으로의 변화, 전화보다는 메신저로 기업내 또는 기업간의 업무 관련 의사 소통 및 E-mail을 통한 업무 지시와 보고, 인터넷을 통한 다양한 쇼핑 등은 현재 우리 생활 깊숙이 들어와 있는 네트워크의 실상을 보여준다. 즉, 갑자기 네트워크에 문제가 생겨 네트워크가 다운된다거나 속도가 떨어지는 문제는 기업내 또는 기업간 업무를 마비시키며 더 나아가서는 기업의 이윤과도 직결되어 있으므로 큰 문제가 아닐 수 없다. 네트워크의 원활한 사용이 정보 경쟁력뿐만 아니라 기업의 이윤과도 직결된다는 것을 알 수 있다.

네트워크에 문제가 발생할 때, 네트워크 상태와 구성을 실시간으로 파악하고 있다면 문제가 생긴 지점을 정확하고 신속하게 파악하여 빠른 대처를 할 수 있다. 이렇게 네트워크의 문제를 신속히 파악하고 대처하기 위해서는 전체 네트워크 구성에 대한 정보가 필요하다. 즉, 관리하는 네트워크에 대하여 물리적인 네트워크 연결 상태가 반영된 네트워크 구성도는 네트워크 관리에 필수적인 요소이다[1].

대부분 기업의 네트워크 구성도는 수작업에 의존하고 있다. 이것은 네트워크 구성의 변경이 네트워크 구성도에 즉각 반영되지 않는다는 것을 의미한다. 관리 초기에는 실제 네트워크 구성과 네트워크 구성도가 일치하지만 사람이 수작업으로 네트워크를 관리하다 보면, 실제 네트워크 구성의 변경 내용이 네트워크에 구성도에는 반영이 되지 않

은 경우가 생기게 된다. 또한, 관리자의 이직시 효율적인 관리를 위해 네트워크 구성도를 새로 작성하게 되고 다시 반복되는 수작업은 소요되는 시간과 비용의 손실일 뿐 아니라, 실제 네트워크 관리를 어렵게 한다[2].

실제 네트워크 구성과 수작업으로 작성한 네트워크 구성도의 불일치는 네트워크 관리를 어렵게 하는 주요 원인이다. 뿐만 아니라, 네트워크 장비에 대한 설정 정보를 검출하고 이 값을 변경하는 구성 관리와 수작업으로 만들어진 네트워크 구성도가 정확하게 연결되지 않아서 효율적인 네트워크 구성 관리가 이루어지지 않고 있다. 그러므로, 네트워크의 상태를 파악하고 문제가 발생했을 때 빠른 대처를 위해서는 네트워크의 구성도를 정확히 파악하고 있어야 한다[3].

본 연구에서는 네트워크 관리자가 네트워크의 구성 관리를 효율적으로 수행할 수 있는 네트워크 연결 정보 관리 시스템의 개발을 목표로 한다. 개발하는 네트워크 연결 정보 관리 시스템을 통하여 가장 기본적으로 제공되어야 할 정보는 네트워크를 구성하는 스위치나 라우터의 연결 정보이다. 이러한 연결 정보는 쉽게 접근할 수 있는 웹을 통하여 제공되어야 한다. 제공되는 연결 정보를 사용하는 한 예로, 문제를 일으키는 네트워크 장비를 언제 어디서나 빠르게 원격에서 차단함으로써 관리의 효율을 증대시킬 수 있다.

또한, 개발하는 네트워크 연결 정보 관리 시스템은 관리하는 모든 네트워크 장비의 구성정보를 읽어와 데이터베이스에 저장하고, 웹을 통한 인터페이스를 제공하여 쉽게 관리함으로써 관리의 효율성을 높인다. 개발하는 네트워크 연결 정보 관리 시스템에서 하고자 하는 일은 다음과 같다. 첫째, 관리하고자 하는 네트워크 장비를 자동으로 검색한다. 둘째, 검색된 장비들간의 포트별 연결 정보를 알아내어 네트워크 구성도를 생성한다. 셋째, 관리 장비들의 구성정보를 관리한다.

이 논문은 다음과 같이 구성되어 있다. 2장에서는 관련연구로 네

트위크 관리의 기능 및 구조와 SNMP 등 기본 개념을 알아보고 우리가 개발하려는 시스템과 비슷한 기능을 수행하는 기존의 시스템에 대해서 알아본다. 3장에서는 효율적인 네트워크 연결 정보 관리 시스템 개발에 대한 요구사항들을 정의하고, 4장에서는 3장에서 정의한 요구사항들을 적용하기 위한 알고리즘과 시스템 전체 구조에 대해서 설명한다. 5장에서는 4장에서의 설계를 바탕으로 실제 시스템을 구현하고, 구현한 시스템을 POSTECH 네트워크에 실제 적용한 결과를 보여준다. 마지막으로 6장에서는 이번 연구가 갖는 의의를 정리하고 향후 연구방향을 제시한다.

II. 관련연구

이 장에서는 먼저 네트워크 관리 프로토콜로 대표적인 SNMP와 그 관리 정보인 MIB 중에서도 장비의 포트 정보를 가지고 있는 Bridge MIB 에 대해서 알아보고, 일반적인 네트워크 관리 시스템이 제공해야 하는 기능들에 대해서 살펴본다. 또한 우리가 개발하고자 하는 시스템과 비슷한 기능을 제공하는 기존 제품에 대해서 알아본다.

2.1 SNMP

네트워크 관리는 안정적이면서 안전하고 효율적인 네트워크 환경을 제공하기 위해서 네트워크 상에 존재하는 다양한 자원들을 모니터링하고 제어하는 것을 말한다. 오늘날의 네트워크는 다양한 회사들의 장비와 시스템으로 구성되고 인터넷의 보편화로 그 규모가 점점 거대해지고 복잡해짐에 따라 네트워크 시스템의 환경과 성능을 유지하기 위해서 네트워크 관리가 필요하게 되었다.

네트워크 관리 프로토콜로 대표적인 것은 TCP/IP 네트워크에서 사용되는 SNMP(Simple Network Management Protocol)[4]이다. 인터넷의 관리체제인 SNMP는 IETF(Internet Engineering Task Force)[5]에 의해 표준화가 이루어지고 있으며, 네트워크 관리를 위한 관리 구조, 관리 정보의 구조, 관리 프로토콜 등에 대한 표준화 작업을 진행하고 있다. SNMP는 네트워크를 효율적으로 관리하기 위한 방안으로 만들어졌으며 초창기에는 과도기적 표준으로 단기간의 사용을 목적으로 시작되었으나 구현이 쉽고 호환이 뛰어난 장점으로 대부분의 네트워크 관리에 널리 사용되고 있다.

SNMP는 1990년 IETF에서 일차로 표준화되었고, 1995년에는 단점을 보완하고 기능을 확장한 SNMPv2[6]가 표준안으로 제시되었고, 1999년에는 이전 버전을 더욱 확장하고 보안 문제를 해결하기 위한 새로운

버전으로 SNMPv3[7]가 제안되었다. 현재 가장 많이 쓰이고 있는 것은 처음 표준화된 SNMP인 SNMPv1이고 SNMPv2가 일부 사용되고 있다.

SNMP 네트워크 관리 구조는 그림 1과 같이 매니저/에이전트 구조로 되어 있다. 관리 대상 장비에는 SNMP 에이전트(SNMP Agent)가 탑재되어서 관리대상 장비의 정보를 수집한다. 그리고 매니저(SNMP Manager)는 네트워크 관리자가 장비를 관리할 수 있도록 에이전트로부터 정보를 수집하고 처리하는 기능을 한다.

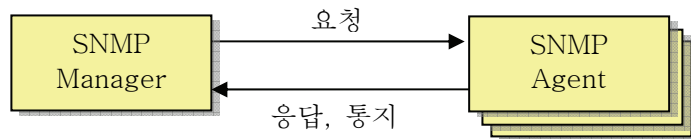


그림 1. SNMP 네트워크 관리 구조

에이전트의 정보를 매니저가 수집하는 방법은 폴링(Polling)과 트랩(Trap)의 두 가지 방법이 있다. 폴링은 매니저의 요청에 대해 에이전트가 응답하는 방법이고, 트랩은 에이전트가 매니저의 요청 없이 발생된 이벤트에 대해서 매니저에게 통보하는 방법이다.

네트워크 관리를 위해서는 관리 정보 모델, 관리 정보 정의, 정보 전달 방법의 세가지 정의를 규정하여야 한다. SNMP의 세가지 항목에 대한 정의는 다음과 같다.

- 관리 정보 모델: 관리할 정보를 나타내는 방식으로 관리 정의를 위한 문법이라고 할 수 있다. SNMP에서는 SMI(Structure of Management Information)[8][9]에 이를 정의하고 있다.
- 관리 정보 정의: 실제로 관리할 정보를 정의하는 것으로 SNMP에서는 관리 정보를 MIB(Management Information Base)[10]으로 나타낸다. 관리 정보는 표준화된 MIB이 있고, 장비 업체별로 각각의 장비에 맞게 정의한 Private MIB이 있다.

- 정보 전달 방법: 프로토콜을 의미하며 이는 매니저와 에이전트 간에 정보를 어떤 방법으로 전달할 것인가 하는 동작을 설명한 것이다. SNMP 프로토콜에서는 요청(Request)/응답(Response)의 방법과 트랩(Trap)의 방법을 사용한다.

SNMP 각 버전별 특징을 간단히 표 1에 나타내었다. SNMPv1은 SMIV1을, SNMPv2와 SNMPv3는 SMIV2를 관리 정보 모델로 이용하며, SNMPv1은 ‘Get’, ‘GetNext’, ‘Set’, ‘Trap’을 사용하여 정보를 전달한다. SNMPv2는 SNMPv1의 네 가지 오퍼레이션에 더하여 한번에 여러 노드의 값을 전달하는 ‘GetBulk’와 매니저간 통신을 위한 동작을 제공 ‘inform’이 추가 되었으며, SNMPv3은 강화된 보안 기능을 제공한다.

항목	SNMPv1	SNMPv2	SNMPv3
관리 정보 모델	SMIV1	SMIV2	SMIV2
프로토콜 동작	Get GetNext Set Trap	Get GetNext GetBulk Set SNMPv2-trap Inform	Get GetNext GetBulk Set SNMPv2-trap Inform
특징	최초 표준	SMI 정의 확장 매니저간 통신 Bulk 통신	보안 강화
표준화	1991년 완료	1999년 완료	1999년 제안
RFC 번호	1155, 1212, 1213, 1215	1901~1908, 2578~2580	2570~2576

표 1. SNMP 버전별 특징

2.2 Bridge MIB

본 논문에서는 포트별 매칭을 위해 Bridge MIB[11]을 사용한다. 전달되는 패킷의 헤더에 있는 목적지 MAC Address를 보고 다음 노드로 전달해주는 것이 Bridge 기능이며 Layer 2에서 동작한다. 이러한 Bridge 기능을 제공하는 네트워크 장비로는 Layer 2의 서비스를 지원하는 스위치, 라우터 등이 있고, 이 장비들의 네트워크 동작을 나타내 주는 것이 Bridge MIB의 역할이다. 이 Bridge 기능을 제공하는 모든 장비들이 Bridge 기능을 통해 Bridge MIB의 정보를 제공하고 있다[12]. 표준 Bridge MIB의 구조는 그림 2와 같다.

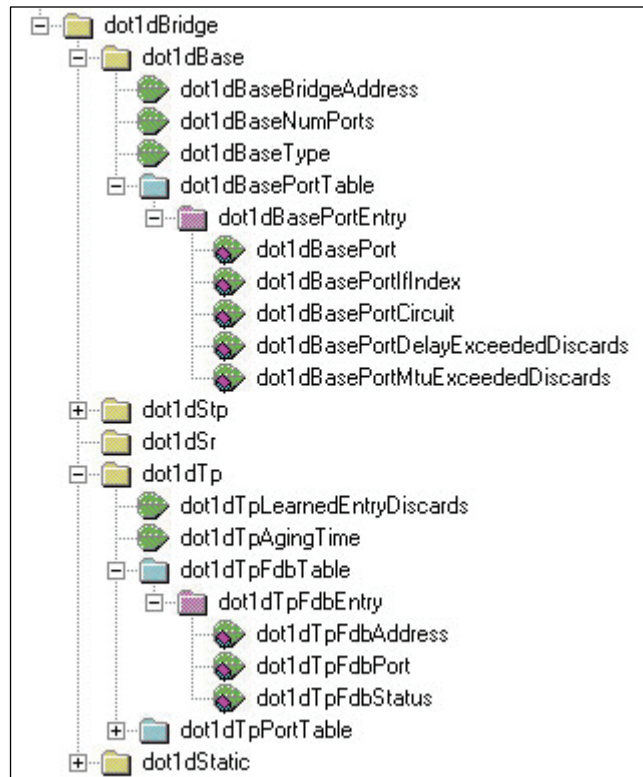


그림 2. Bridge MIB의 구조

Bridge MIB은 그림 2에서 보는 것처럼 다섯 개의 그룹으로 구성되어 있으며 각각 하는 역할은 아래와 같다. 본 논문에서 사용되는 MIB 정보는 굵은체로 표시했다.

■ **dot1dBase Group**

이 그룹은 포트 수, 포트 타입, 포트 주소와 같은 Bridge의 기본 정보를 제공한다. 본 논문에서는 이 그룹 중에서 Bridge와 연결된 모든 포트에 관한 일반적인 정보를 가진 dot1dBasePortTable을 이용한다. dot1dBasePortTable의 내부 관리 객체 중에서도 dot1dBasePort와 dot1dBasePortIfIndex는 다음과 같은 정보를 제공한다.

- **dot1dBasePort:** Bridge 기능에서 사용하는 포트 번호로 아래의 dot1dBasePortIfIndex와 함께 실제 장비에서 사용하는 포트 번호와 Bridge MIB에서 사용하는 포트 번호를 연결시켜주는 용도로 사용된다.
- **dot1dBasePortIfIndex:** dot1dBasePort에 대응되는 인터페이스를 위한 MIB-II에 정의되어 있는 ifIndex 관리 객체의 인스턴스 값을 제공한다.

■ **dot1dStp Group**

이 그룹은 STP(Spanning Tree Protocol)[13]과 관련된 Bridge의 상태를 표시하는 관리 객체들을 가진다. 만일 노드가 STP를 수행하지 않는다면 이 그룹은 필요치 않다.

■ **dot1dSr Group**

이 그룹은 Source Route Bridge 기능에 관한 관리 객체를 포함한 그룹이다. Source Route Bridge는 MAC 프레임에 전달을 위한 루트가 포함되어 있으며 이를 기반으로 MAC 프레임이 전달된다. 이 그룹은 장비간의 연결 정보로 활용되지 않는다.

■ dot1dTp Group

이 그룹은 Transparent Bridge 기능에 관한 관리 객체를 포함한 그룹이다. Transparent Bridge 기능은 학습을 통하여 라우팅 테이블을 만들고 이 테이블을 기반으로 프레임을 전달한다. 각 포트에 연결되어 있는 다른 장비의 MAC Address는 각 포트에 들어오는 MAC 프레임을 보고 알아내어 학습을 한다. 학습의 결과는 전달 테이블로 관리를 한다. 예를 들면, 포트 1번으로 MAC Address aa:bb:cc:xx:yy:zz를 가진 패킷이 들어오면 이를 전달 테이블에 기록을 하여놓고 다음에 aa:bb:cc:xx:yy:zz로 전달할 패킷이 들어오면 전달 테이블을 이용하여 포트 1번으로 이 패킷을 전달한다. 본 논문에서 네트워크 장비들의 연결 구성 정보를 도출하기 위해 사용되는 dot1dTpFdbTable의 관리 객체는 다음과 같다. 이 관리 객체들은 들어오는 MAC 프레임을 기반으로 학습한 전달 테이블의 내용이다.

- **dot1dTpFdbAddress**: 전달을 위한 도착지 MAC Address
- **dot1dTpFdbPort**: 전달을 위한 Bridge 포트 번호
- **dot1dTpFdbStatus**: 전달 테이블의 학습 상태를 표시한다. 학습 상태는 Other(1), Invalid(2), Learned(3), Self(4), Mgmt(5) 중 하나이다. 각 학습 상태의 정보는 다음과 같다.
 - Other(1): 전달을 위한 정보가 이 테이블에 명시되어 있지 않고 다른 방법으로 제공됨을 의미한다.
 - Invalid(2): 학습을 통하여 전달 정보를 취득한 후 일정 시간이 경과하여 더 이상 유효하지 않음을 의미한다.
 - Learned(3): 학습을 통하여 취득한 정보가 유효함을 의미한다.
 - Self(4): dot1dTpFdbPort에 있는 포트가 dot1dTpFdbAddress

에 있는 MAC Address의 전달 정보를 가지고 있음을 의미한다.

- Mgmt(5): 전달 정보가 dot1dTpStaticAddress에 이미 기록되어 있음을 의미한다.

- dot1dStatic Group

전달을 위한 정보가 네트워크 관리자에 의하여 고정적으로 설정되었을 경우에 이 전달 정보를 제공하는 관리 객체들이 포함되어 있는 그룹이다.

2.3 MIB-II의 at Group

멀티레이어 스위치인 경우 설정에 따라 Layer 2나 Layer 3를 선택하여 서비스를 제공하게 된다. Layer 2인 경우는 2.2장의 Bridge MIB을 이용하여 포트별 연결정보를 알아낼 수 있지만, Layer 3으로 동작하는 경우에는 ifTable의 MAC Address가 대부분 동일한 기본 MAC Address로 설정이 되어 MAC address로 포트를 구분할 수 없으며 또한, 주소 변환 테이블을 이용하여 패킷을 전달하기 때문에 아래의 그림 3의 MIB-II의 at 그룹 중 atTable을 이용한다.

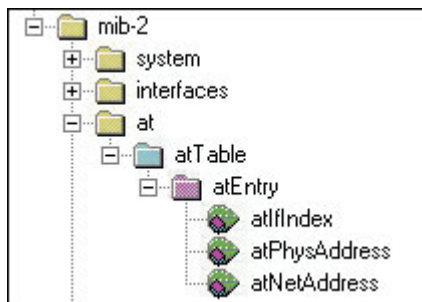


그림 3. MIB-II의 at Group의 구조

- at (Address-Translation) Group

IP Address와 MAC Address간의 매핑 테이블로 구성되며 장비의 인터페이스 별로 IP 주소와 MAC 주소를 검색할 수 있도록 되어 있다. 네트워크 장비의 인터페이스를 나타내기 위해 `atIfIndex` 오브젝트가 정의되어 있으며, 특히 `atPhysAddress` 오브젝트를 통해서 MAC 주소를 검색할 수 있다.

- **atIfIndex** : MIB-II 의 ifTable의 ifIndex와 동일한 index
- **atPhysAddress** : 매체-의존적인 물리적 주소
- `atNetAddress` : 매체-의존적인 물리적 주소에 대응하는 Network Address (예, IP 주소)

2.4 관련 시스템

이 장에서는 본 논문에서 구현한 시스템과 비슷한 기능을 수행하는 시스템들에 대해서 알아본다.

2.4.1 OpUtils Switch Port Mapper

AdventNet의 OpUtils Switch Port Mapper[13]는 ‘AdventNet ManageEngine OpUtils 4[15]’에 포함되어 있는 네트워크 모니터링 도구에 해당된다. 특정 스위치 포트들에 연결되어 있는 장비를 발견하는 것이 주된 기능이며, 이 도구를 이용해 스위치 포트에서 수동으로 추가 연결되거나 또는 제거된 장비를 확인할 수 있다. 이 도구가 수행된 후, 스위치의 각 포트에 연결된 장비 목록을 IP Address, MAC Address, 포트의 동작상태와 속도를 확인 할 수 있으며, 수행 결과에 기초하여 네트워크 포트 이용률이 보고된다. 또한 관리자는 네트워크의 모든 스위치를 단순히 IP Address 또는 MAC Address를 입력하여 추가할 수 있으며 이 도구는 추가됨과 동시에 즉시 수행되어 각각 스위치에 대한 포트 상태를 보여준다. 이 도구는 하루, 한달, 일년을 주기로 스케줄되어 운영될 수

있다.

OpUtils Switch Port Mapper는 Cisco, Nortel, HP, Foundry, Extreme Networks, Intel의 스위치에서 테스트 되었으며, Bridge MIB이 지원되는 스위치와 허브에 대해서 사용 가능하다. 허브가 스위치에 연결되어 있고, 다수의 장비가 허브에 직접 연결되어 있다면 다수의 장비들은 스위치의 하나의 포트에 연결된 것처럼 나타난다. 이 도구는 각 스위치의 포트별 상태를 아는 것에는 유용하게 이용될 수 있지만 전체 엔터프라이즈 네트워크를 관리하기 위한 스위치간의 연결 정보는 파악할 수 없다.

2.4.2 CISCO Network Assistant

CNA(CISCO Network Assistant)[15]는 시스코 스위치, 라우터 및 무선 액세스 포인트의 관리 작업을 단순화하는 PC 기반의 네트워크 관리 애플리케이션으로, 중앙집중식 네트워크 관리와 구성 성능을 제공한다. 이 어플리케이션은 시스코 장비의 초기 배치와 지속적인 유지 보수를 쉽게 하기 위해 Cisco Smartsports[17] 기능을 사용한다. Smartsports 기능은 관리되는 장비의 보안과 가용성, QoS 성능에서 일관성과 신뢰성을 보장하기 위해 장비와 포트에 대한 시스코의 최적 실행 방안을 자동으로 구현하여 네트워크 관리자의 시간을 절약하고 사람의 실수를 줄여 주며 관리 장비의 구성이 이들 애플리케이션에 최적화되도록 보장한다.

또한, CNA는 어디에서나 사용자가 쉽게 접근하여 장비를 관리할 수 있게 쉽게 접속할 수 있는 GUI의 특징을 제공한다. 구성 관리, 자산 관리를 위한 인벤토리 보고 기능, 이벤트 알림 등의 기능이 제공되며 시스코 웹사이트에서 사용자로 등록을 하면 무료로 다운 받아 사용할 수 있다. Topology view, discovering devices, Smarts propagation speed, multiple port configuration propagation 등의 기능이 제공되며, 네트워크 장비가 전부 Cisco 장비라면 전체 네트워크 구성도 생성이 가능하다. 단,

모든 장비의 ip http-server 옵션이 enable 되어 있어야 한다는 제약이 있다. ip http-server 옵션이 enable된 장비간에는 이웃한 장비의 연결 정보가 자동으로 얻어진다. 연결 정보를 얻는 방법은 Cisco 회사에서 자체적으로 구현한 것으로 표준이 아니며 공개되지 않은 방법이다. 따라서, 다른 회사 장비와의 연결 정보는 자동으로 얻어지지 않는다.

2.4.3 IBM Netcool/Precision (Tivoli software)

Netcool/Precision[18]은 네트워크 관리를 위한 IBM의 상용제품으로 실시간 네트워크 감지, 시각화된 네트워크 구성도 제공, 장애 원인 분석 솔루션을 제공한다. IP 네트워크를 자동으로 탐색해 네트워크 토폴로지(topology)를 구성한 후 매핑해 레이어 2와 3 장비들의 완벽한 구성도를 작성하며, 장비의 물리적인 포트간 연결 정보 확인 및 관리가 가능하다. 네트워크 자원을 실시간으로 인식해 지속적인 업데이트를 하며 네트워크 변경시 토폴로지 데이터베이스와 구성도를 자동으로 갱신한다. 또한, 네트워크 구성도를 통해 네트워크 장애의 근본 원인을 분석하여 관리자에게 자동으로 알려주는 기능도 제공한다. SNMP 장비로부터 SNMP, MIB 정보를 취득해 정확한 문제의 원인을 실시간, 장비 특성별로 제공하며 정상 작동하지 않는 원인 설명과 해결책을 제시한다.

또한, 탐색된 장비의 데이터를 이용해 용량, 사용량, 고객 정보, 포트 및 회선 여분 상황, 연결여부 등을 파악해 네트워크 자산 관리에도 이용한다. 이 제품은 이들이 연결 정보를 취득하는 구체적인 방법에 대해서는 공개하지 않고 있어 그 방법을 알 수 없다. 따라서, 본 논문에서는 네트워크 장비 검색 및 장비간 연결 정보를 알아내는 방법을 표준에 의거한 상세한 알고리즘을 제시하고 제시한 알고리즘을 따라 구현한 후, POSTECH 네트워크에 실제 적용하여 알고리즘을 검증하는 것을 목표로 한다.

III. 시스템 요구사항

본 논문에서 제안하는 시스템은 네트워크 장비의 연결 정보를 중심으로 구성 관리를 수행한다. 따라서 본 시스템은 관리 대상 장비를 자동으로 검색하여 한 눈에 파악할 수 있는 네트워크 구성도를 생성해야 하며, 네트워크 연결 정보 관리를 위한 각 장비의 구성 정보의 확인 및 수정을 쉽게 할 수 있도록 설계되어야 한다. 네트워크 연결 정보 관리를 위한 요구사항은 다음과 같다.

■ Discovery 기능

관리하고자 하는 네트워크의 IP Address 영역을 입력 받아서 장비들 중에서 구성 관리가 가능한 즉, 관리 대상인 장비를 자동으로 검색하여 IP Address별 장비의 목록을 작성해야 한다. 또한, 하나의 장비에 여러 개의 IP Address가 할당 되어 있을 수 있으므로 대표 IP로 그룹핑 되어서 장비 목록이 작성되어야 한다.

■ Map을 통한 연결 정보 관리 기능

각 포트별 장비간의 연결 정보를 웹을 통하여 쉽게 확인 할 수 있어야 한다. 또한, 네트워크 구성도를 장비의 종류에 따라 쉽게 구분할 수 있는 이미지로 표시하며, 장비 상태에 따라 색을 달리하여 표시하여 쉽게 정상적인 장비와 문제가 있는 장비를 구분하도록 한다. 장비의 포트별 사용 여부(enable/disable)도 색으로 구분하여 관리자가 효율적으로 장비를 관리 할 수 있도록 한다.

■ 장비 구성 관리 기능

새로운 장비를 추가하거나 더 이상 관리할 필요가 없어진 장비에 대한 삭제 및 장비의 구성 정보를 변경하는 기능이 있어야 한다. 또한, 잘못된 구성 정보의 셋팅으로 문제가 생기거나, 초

기 상태로 돌아가고자 하는 경우를 위해서 초기화 기능이 제공되어야 하며, 포트의 연결 정보를 변경 할 수 있는 기능이 제공되어야 한다. 네트워크에 문제가 생겨서 포트의 연결을 막아야 할 때, 포트 셋팅을 통해 장비에 대한 포트의 사용을 막는 등의 구성 정보 변경 기능이 제공되어야 한다.

■ 데이터베이스 관리 및 백업기능

정확한 네트워크 구조 및 시스템의 구성 정보가 실시간으로 파악되어 관리자에게 제공되어야 한다. 주기적으로 장비의 존재여부 및 구성 정보에 대한 값을 가져와 관리 정보 데이터베이스에 최신정보를 유지하며 정확한 정보를 제공 해야 한다. 관리자에 의해 변경된 구성 정보가 데이터 베이스에 저장됨과 동시에 실제 장비의 구성 정보에도 즉시 적용 되어야 한다. 현재 운영하고 있는 구성 데이터베이스와 실제 구성 현황의 일치 여부를 주기적으로 검색하고 변경된 구성에 대해서는 Event log에 기록하여 관리한다.

■ 사용자 관리 기능

사용자를 크게 관리자와 일반사용자로 분류하여 관리한다. 운영 중인 네트워크의 일관적인 관리를 위해 관리자에 의해서만 구성 관리가 되어야 하며, 일반 사용자에게는 생성된 네트워크 구성도와 일반적인 정보들을 열람할 수 있는 간단한 기능만 제공하도록 한다.

■ 쉬운 사용자 인터페이스 기능

네트워크 구성 요소들간의 연결 정보 및 구성 정보를 웹을 통해 접근하여 쉽게 관리할 수 있게 해야 한다. 즉, 언제 어디서나 쉽게 접근하여 관리하도록 관리의 효율성을 높이기 위해 웹 기반의 사용자 인터페이스를 제공해야 한다. 웹 브라우저만 있

으면 별도의 프로그램 설치 없이 언제 어디서나 본 시스템을 사용할 수 있다. 또한, 시스템이 전반적으로 사용하기 쉽고 단순하게 사용자 인터페이스가 구성되어야 한다.

IV. 시스템 설계

이 장에서는 3장에서 제시한 요구 사항을 충족시키는 네트워크 연결 구성 정보 관리 시스템을 설계한다. 먼저 연결 구성도를 생성하기 위한 포트별 IP 매칭 알고리즘을 설명하고, 이 알고리즘에 근거한 전체 시스템 구조를 제시한다.

4.1 포트별 IP 매칭 알고리즘

네트워크의 연결 구성도를 생성하기 위한 포트별 IP 매칭 알고리즘에 대한 전체 순서도는 그림 4과 같다. 처음으로 수행하는 작업은 관리 장비를 자동으로 검색하기 위해 엔터프라이즈 네트워크의 관리대역 IP Address 범위를 외부로부터 입력 받는 것이다.

다음으로 검색을 수행할 IP Address 범위를 외부로부터 입력을 받아서 해당되는 IP Address의 장비로 SNMP Get을 보내어 응답하는 장비에 대해서만 관리 장비 목록에 추가하는 것이다. 세번째는 하나의 장비에는 여러 개의 IP Address가 있을 수 있으므로 동일한 장비는 대표 IP Address를 선택해 하나의 IP로 그룹핑해서 관리 대상 목록을 작성하는 것이다. 네번째는 중복 없는 관리 대상 목록을 이용해 각 장비 포트별 연결 정보를 알아내기 위해 포트 매칭에 필요한 장비의 Interface 및 Bridge MIB, atTable의 정보를 가져와 저장하는 작업이다.

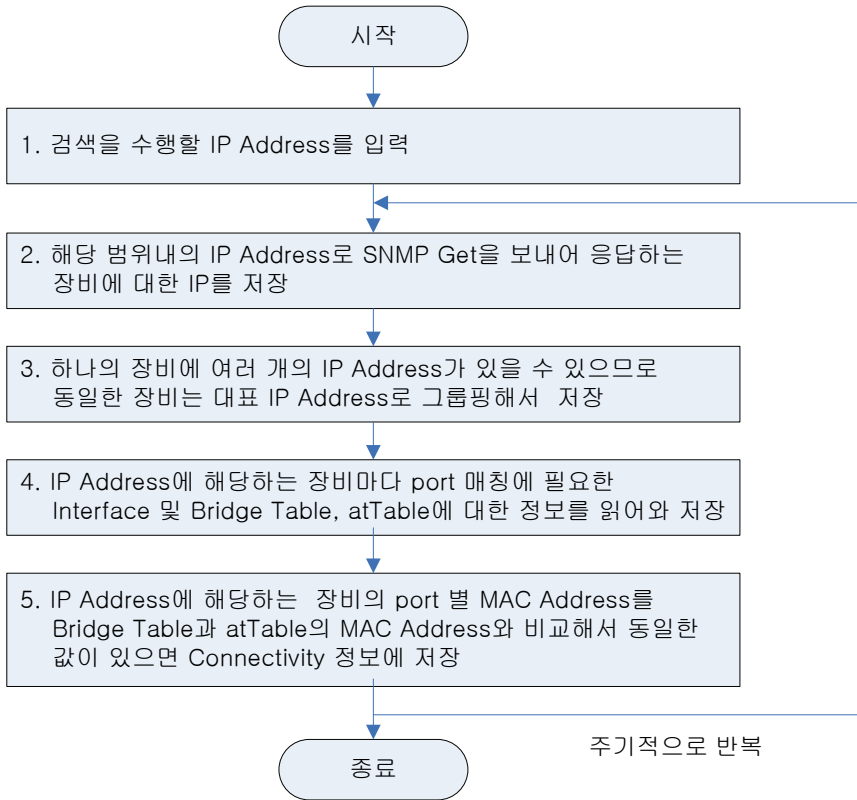


그림 4. 포트별 IP 매칭 알고리즘에 대한 순서도

마지막으로, 장비 목록에 저장된 IP Address 순서대로 장비의 포트별 MAC Address를 읽어와 다른 장비의 Bridge MIB과 atTable의 정보를 비교해서 동일한 Address가 검출되면 물리적으로 연결 되어 있는 것으로 판단하고 연결 정보를 저장한다. 이 알고리즘은 관리자의 필요에 의해 즉시 수행되거나, 관리자가 설정한 주기에 의해 반복 수행으로 네트워크 장비의 연결 정보를 최신 정보로 유지하게 된다. 다음 각 세부 장에서 그림 4의 2번부터 알고리즘부터 세부 내용을 순서대로 설명한다.

4.1.1 네트워크 장비 스캐닝 알고리즘

그림 5의 순서도는 그림 4의 2번의 세부 알고리즘으로 관리 대상

장비의 IP Address 목록을 자동으로 생성하기 위한 순서도이다. 네트워크 IP Address 범위내의 관리 대상을 자동으로 검출하기 위해서, 먼저 외부로부터 관리 IP Address 대역을 입력 받는다. 관리 대역에 해당 되는 모든 IP Address로 SNMP 에이전트의 탑재 여부 및 관리 대상 장비에 해당 되는지 알아내기 위하여 SNMP Get 메시지를 보낸다. 각각의 장비에 대하여 MIB-II의 관리 정보 중 system 그룹내의 sysServices의 값을 요청하는 SNMP Get을 보내고 Timeout을 셋팅한 시간만큼 Sleep하면서 기다린다. Timeout내에 응답이 오지 않으면 관리 대상 장비가 아니거나, 사용하지 않는 IP Address로 취급하고 종료한다. Timeout내에 sysServices에 대한 SNMP Response를 받게 되면 SNMP 에이전트가 탑재된 장비로부터의 응답이므로 그 다음단계로 연결 정보 구성도를 생성하기 위한 관리 대상 장비인 라우터나 스위치에 해당하는지 여부를 검사한다.

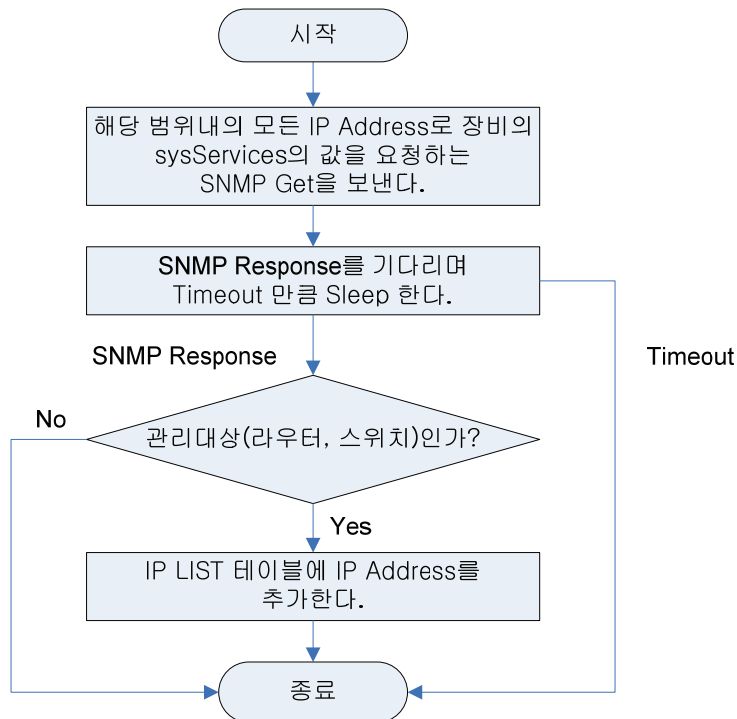


그림 5. 관리 대상 장비의 IP 리스트를 생성하기 위한 순서도

sysServices는 7비트 코드로 해석되는 값을 가지고 각 비트는 OSI의 각 레이어를 의미하며, 이 값을 기반으로 관리 대상 장비인 라우터, 스위치 등으로 분류하게 된다. 응답 받은 sysServices가 ‘2’, ‘3’의 값을 가지면 ‘L2 스위치’며, ‘6’이면 ‘L3 스위치’, ‘78’이면 ‘멀티레이어 스위치’로 판단하며 이에 해당되는 IP Address 및 sysServices정보를 저장하여 결과 목록을 작성한다. 즉, 스캔 된 장비 중 sysServices의 값이 2, 3, 6, 78인 장비들만 실제 관리 목록 대상으로 저장한다. 실제 POSTECH 네트워크를 대상으로 스캐닝 한 결과 2, 3, 6, 78의 장비가 스캔되었으며, 그들을 자세히 분석한 결과 아래의 표 2에 해당함을 확인 할 수 있었다.

관리 장비 분류	sysServices
L2 스위치	2, 3
L3 스위치	6
멀티레이어 스위치	78

표 2. sysServices에 따른 장비의 분류

4.1.2 네트워크 장비 그룹핑 알고리즘

일반적으로 라우터와 같은 네트워크 장비들은 인터페이스마다 별도의 IP Address를 갖기 때문에 여러 개의 IP Address가 하나의 라우터에 존재한다. 네트워크 장비를 IP Address로 스캔했기 때문에 중복되는 장비가 존재 할 수 있으므로 중복되는 장비를 하나의 대표 IP Address로 그룹핑해서 관리해야 한다. 그림 6의 순서도는 그림 4의 3번 알고리즘으로 중복된 장비들을 하나의 IP Address로 그룹핑 시키는 순서도이다.

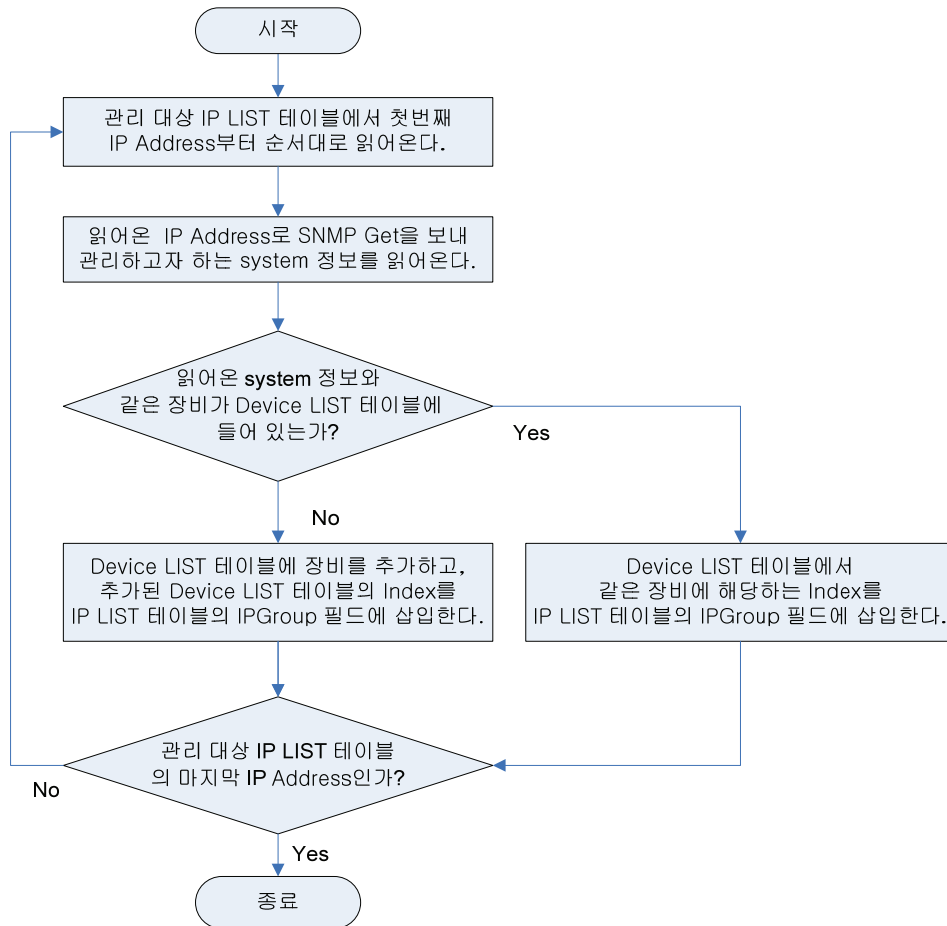


그림 6. 장비 리스트 중 동일한 장비를 그룹핑하기 위한 순서도

하나의 장비에 여러 IP Address가 할당되었을 경우에 이것을 대표 IP Address로 그룹핑하기 위해서 그림 5의 순서도에 의해 생성된 장비의 IP Address 목록에서 처음 IP Address부터 순서대로 읽어온다. 읽어온 IP Address에 해당하는 장비로 SNMP Get을 보내어 관리하고자 하는 시스템의 정보를 읽어와 Device LIST 테이블에 저장하게 된다. 저장하기 전, 읽어온 장비의 시스템 정보 중 sysDescr, sysObjectID, sysName, sysLocation을 ‘Device LIST 테이블’에 있는 각각에 해당되는 값과 비교한다. 4개의 값이 모두 같으면 동일한 장비로 간주하고 Device LIST에

추가함 없이 IP LIST에만 동일한 장비임을 특정 필드값에 표시 해 둔다. 새로운 장비로 판단되면 Device LIST에 장비를 추가하고, 추가된 장비와 동일한 장비임을 IP LIST의 특정 필드값에 표시해 둔다.

본 논문에서는 중복되는 장비의 IP Address 중 첫 번째로 읽게 되는 Address가 대표 IP Address로 선택되며, 추후에 관리자가 원하는 IP Address로 대표 주소를 지정해서 관리 할 수 있도록 하기 위해 IP LIST의 특정 필드에 동일한 장비임을 표시 해 둔다.

4.1.3 네트워크 장비별 정보 수집 알고리즘

그림 7의 순서도는 그림 4의 4번에 해당하며 포트 매칭을 위해서 장비 별로 필요한 정보를 수집하는 순서도에 해당한다. 그림 6의 결과에 해당하는 Device LIST에서 IP Address를 순서대로 추출해서 추출한 IP Address에 해당되는 장비에서 필요한 정보를 가져와 저장하게 된다. 먼저, 장비가 멀티레이어 장비인 경우, 즉, sysServices가 '78'인 경우, Layer 3에서 동작하는 장비로 설정되어 동작 할 수 있으므로 추가적으로 장비의 MIB-II 정보 중 atTable을 저장한다. 다음으로, 장비의 MIB-II 정보 중 ifTable에서 ifIndex, ifDescr, ifPhysAddress, ifAdminStatus, ifOperStatus의 값을 읽어와 데이터베이스의 ifTable에 저장한다. 저장한 ifTable 중 ifAdminStatus와 ifOperStatus가 모두 up(1)인 상태에 대해서만 ifDescr이 VLAN[19]으로 시작되는 String을 읽어온다.

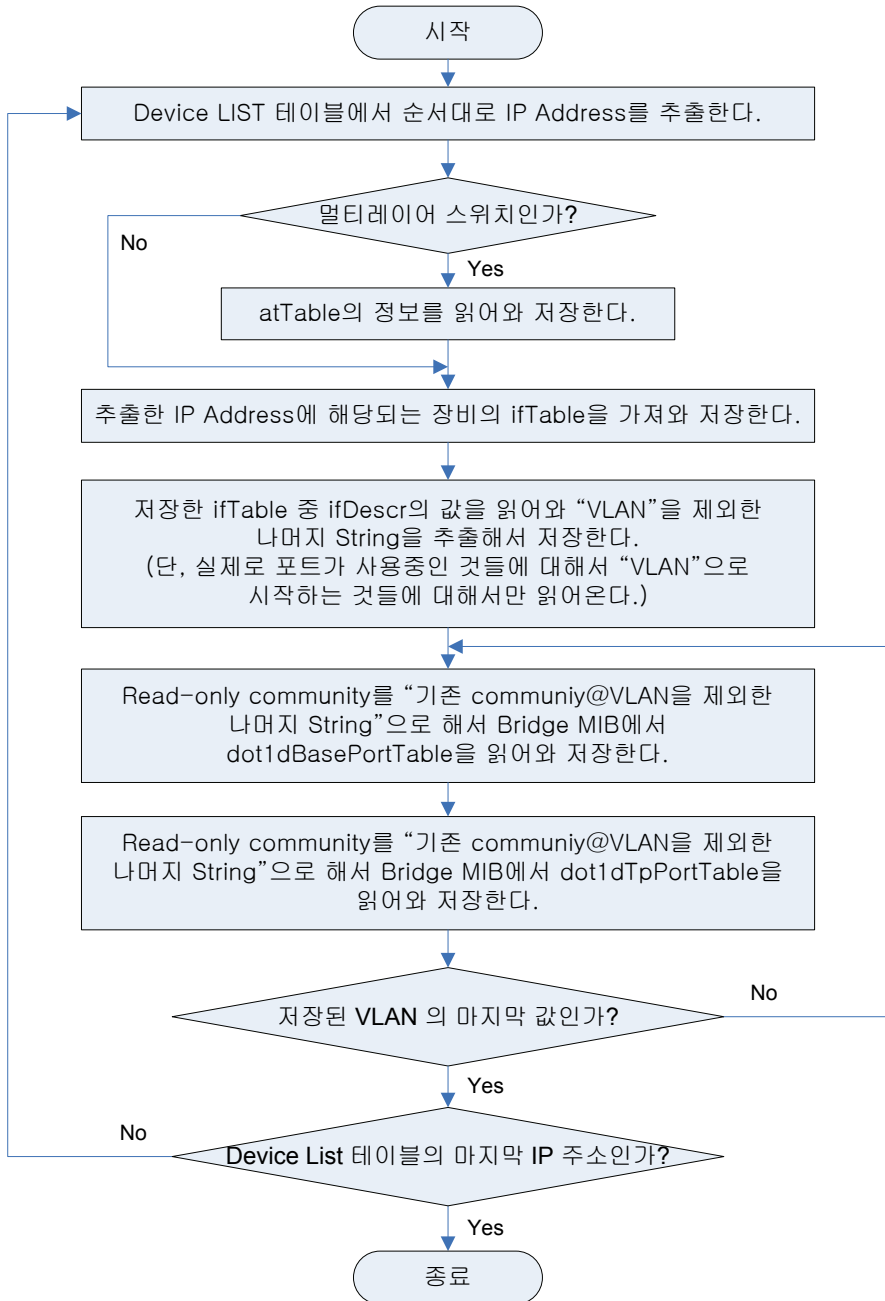


그림 7. 포트 매칭을 위한 장비별 정보수집 순서도

ifTable은 인터페이스 그룹에 속하며 인터페이스 그룹은 네트워크 시스템의 인터페이스에서 발생할 수 있는 트래픽의 통계와 조직 정보를 포함하여, 개체의 물리적 인터페이스와 관련된 정보를 가지고 있다. 그 중에서 ifAdminStatus와 ifOperStatus object는 인터페이스의 상태와 관련이 있다. ifAdminStatus는 Read-write 할 수 있는 값이므로, 관리자에 의해 인터페이스의 동작 상태가 변경될 수 있으며, ifOperStatus는 Read-only 값으로 인터페이스의 현재 실제 동작 상태를 나타낸다. ifAdminStatus가 down(2)이면 ifOperStatus 상태도 down(2)으로 설정되며, ifAdminStatus가 up(1)이 되면 ifOperStatus 상태도 up(1)이 된다. 두 object의 모든 값이 down(2)이면, 인터페이스는 관리자에 의해서 차단된 것을 의미하고, ifAdminStatus가 up(1)이고 ifOperStatus가 down(1)인 경우는 인터페이스가 실패했거나 지역적으로 꺼진 것을 의미한다. 즉, 상태값이 모두 up(1)인 경우만이 현재 실제 동작 중인 포트를 의미하게 된다.

현재 동작중인 포트만 의미가 있으므로 ifAdminStatus와 ifOperStatus가 둘 다 up(1) 상태를 가지고, VLAN으로 시작되는 String을 읽어와 앞의 VLAN을 제외한 숫자만 저장한다. 표 3은 좀 더 구체적인 설명을 위해 특정 스위치에 대한 ifTable의 일부를 표로 나타냈으며, 표 3으로 예를 들면 1, 10, 20, 30, 40을 저장하게 된다.

ifIndex	ifDescr	ifPhysAddress	ifAdminStatus	ifOperStatus
1	GigabitEthernet0/1	00.0B.5F.F8.B0.81	up(1)	up(1)
2	GigabitEthernet0/2	00.0B.5F.F8.B0.82	up(1)	up(1)
3	GigabitEthernet0/3	00.0B.5F.F8.B0.83	up(1)	up(1)
4	GigabitEthernet0/4	00.0B.5F.F8.B0.84	up(1)	up(1)
5	GigabitEthernet0/5	00.0B.5F.F8.B0.85	up(1)	up(1)
6	GigabitEthernet0/6	00.0B.5F.F8.B0.86	up(1)	up(1)
7	GigabitEthernet0/7	00.0B.5F.F8.B0.87	up(1)	up(1)
8	GigabitEthernet0/8	00.0B.5F.F8.B0.88	up(1)	up(1)
10	GigabitEthernet0/10	00.0B.5F.F8.B0.8A	up(1)	up(1)
11	GigabitEthernet0/11	00.0B.5F.F8.B0.8B	up(1)	up(1)
12	GigabitEthernet0/12	00.0B.5F.F8.B0.8C	up(1)	up(1)
13	Null0	(zero-length)	up(1)	up(1)
14	Vlan1	00.0B.5F.F8.B0.80	up(1)	up(1)
16	Vlan10	00.0B.5F.F8.B0.80	up(1)	up(1)

17	Vlan20	00.0B.5F.F8.B0.80	up(1)	up(1)
18	Vlan30	00.0B.5F.F8.B0.80	up(1)	up(1)
19	Vlan40	00.0B.5F.F8.B0.80	up(1)	up(1)
20	GigabitEthernet0/9	00.0B.5F.F8.B0.89	up(1)	up(1)

표 3. ifTable의 예시

다음으로, Bridge-MIB에 속하는 2개의 테이블을 읽어와서 저장한다. Bridge-MIB의 테이블에서 특정 index의 값을 읽기 위해서 Community String Indexing[20]이라는 문법을 사용한다. 사용 설명을 하기에 앞서, Community String Indexing에 대해서 간단히 설명하면 다음과 같다.

몇몇 표준 MIB들은 특정 SNMP 개체가 오직 하나의 MIB 인스턴스를 포함하는 것으로 간주한다. 그래서, 그 표준 MIB은 어떠한 MIB의 인스턴스를 직접 액세스하기 위한 인덱스를 가지지 않는다. 이 경우에, Community String Indexing이 표준 MIB의 각각의 인스턴스를 액세스하기 위하여 제공된다. 그 문법은 [community string]@[instance number]이다. 만일 read-only 커뮤니티 스트링이 public이라면, VLAN 25를 위한 Bridge-MIB은 ‘**public@25**’로 접근하여 읽는다. VLAN 33 이면 ‘**public@33**’로 읽는다. Community String Indexing은 단지 하나의 인스턴스를 가진 MIB을 액세스하는데만 사용되는 것이 아니다. 즉, public@25는 RFC-1213-MIB을 액세스하는 것과 동시에 VLAN 25의 Bridge-MIB을 액세스하는데 사용할 수 있다.

Bridge-MIB을 액세스하여 필요한 정보를 가지고 오기 위해서 우리는 위에서 설명한 Community String Indexing을 사용한다. 표 3의 장비에 대하여 예를 들면, read-only string이 public라 가정한다면, “public@1”, “public@10”, “public@20”, “public@30”, “public@40”으로 각각의 VLAN을 액세스 할 수 있다. 즉, 각각의 Community String Indexing을 가지고 각각의 VLAN에 해당하는 엔트리의 값을 가지고 온다[21]. Community String Indexing을 이용하여 VLAN 별로 장비의 포트에 관한 정보를 읽어오는

이유는 VLAN 별로 Bridge 기능이 제공, 관리되기 때문이다.

다음 순서로 dot1dBasePortTable의 object인 dot1dBasePort와 dot1dBasePortIfIndex를 위와 같은 방식으로 읽어와 저장한다. Community String Indexing을 이용해서 앞에서 읽어와 저장한 VLAN의 수만큼, read-only string을 [community string]@[instance number]로 [instance number]를 변경하며 모두 읽어와 저장한다. dot1dBasePort의 값은 실제 장비의 포트 번호를 가지고 dot1dBasePortIfIndex은 MIB-II의 IfTable의 ifIndex와 같은 값에 해당한다.

dot1dTpFdbTable의 object인 dot1dTpFdbAddress, dot1dTpFdbPort, dot1dTpFdbStatus를 dot1dBasePortTable를 읽어올 때와 동일한 방법으로 모두 읽어와서 저장한다. 표 4는 표 3의 장비의 dot1dTpFdbTable을 ‘public@1’로 읽어온 결과로 Bridge 테이블에 저장한다.

dot1dTpFdbAddress	dot1dTpFdbPort	dot1dTpFdbStatus
00.00.0C.07.AC.01	12	learned(3)
00.0B.60.AC.3A.8A	11	learned(3)
00.0B.60.AC.3E.4A	12	learned(3)

표 4. dot1dTpFdbTable을 ‘public@1’로 읽은 결과

패킷을 받았는데 패킷 헤더의 목적지 Address가 표 4의 ‘00.00.0C.07.AC.01’와 같다면 이 장비의 12번 포트에 전달됨을 의미한다. 표 5는 표 3의 장비의 dot1dTpFdbTable을 ‘public@10’으로 읽어온 결과이고 위와 마찬가지로 Bridge 테이블에 저장한다.

Dot1dTpFdbAddress	dot1dTpFdbPort	dot1dTpFdbStatus
00.01.59.80.0C.7F	1	learned(3)
00.01.59.80.10.08	1	learned(3)
00.01.96.5E.92.00	1	learned(3)
00.01.96.5E.92.01	1	learned(3)
00.04.9A.9C.4D.C0	5	learned(3)

00.04.9A.9C.4D.C1	5	learned(3)
00.11.25.57.D8.38	10	learned(3)
00.A0.C9.22.89.AC	1	learned(3)
00.C0.26.90.8F.E9	10	learned(3)
00.D0.B7.2C.8C.36	1	learned(3)

표 5. dot1dTpFdbTable 을 ‘public@10’으로 읽은 결과

나머지 “public@20”, “public@30”, “public@40”으로도 해당하는 테이블의 내용을 읽어와 저장한다. 표 5로 ‘VLAN10’으로 그룹 되어있는 포트가 1, 5, 10이 있음을 알 수 있으며, 5번 포트를 예로 들어 설명하면 5번 포트의 MAC 주소 ‘00.04.9A.9C.4D.C0’와 ‘00.04.9A.9C.4D.C1’를 가진 패킷이 전달되어 학습되었음을 알 수 있다.

dot1dBasePortTable은 dot1dTpFdbTable의 dot1dTpFdbPort에 해당하는 값과 IfTable 테이블의 ifIndex를 매칭시키기 위해 저장되는 테이블이다. 즉, 실제 장비의 포트 번호와 Bridge MIB에서 사용하는 포트 번호를 연결시켜 주기 위해 저장된다. dot1dTpFdbTable의 dot1dTpFdbPort의 값은 dot1dBasePortTable의 dot1dBasePort에 해당되고, dot1dBasePortIfIndex은 MIB-II의 IfTable의 ifIndex와 같은 값에 매칭된다.

Device LIST에 있는 모든 장비에 대해서 위에서 설명한 것처럼 필요한 정보를 추출해서 모두 저장한다.

4.1.4 네트워크 장비의 포트별 IP 매핑 알고리즘

마지막으로 그림 8은 그림 4의 5번에 해당한다. 그림 7의 수행 결과로 얻은 정보를 사용해서 각 장비의 포트별 연결 정보를 그림 8의 알고리즘의 순서도에 따라 검출해서 저장한다. 그림 7의 알고리즘을 따라 추출해서 저장한 정보를 가지고 장비리스트의 IP Address 순서대로 하나씩 읽어와 비교를 시작한다.

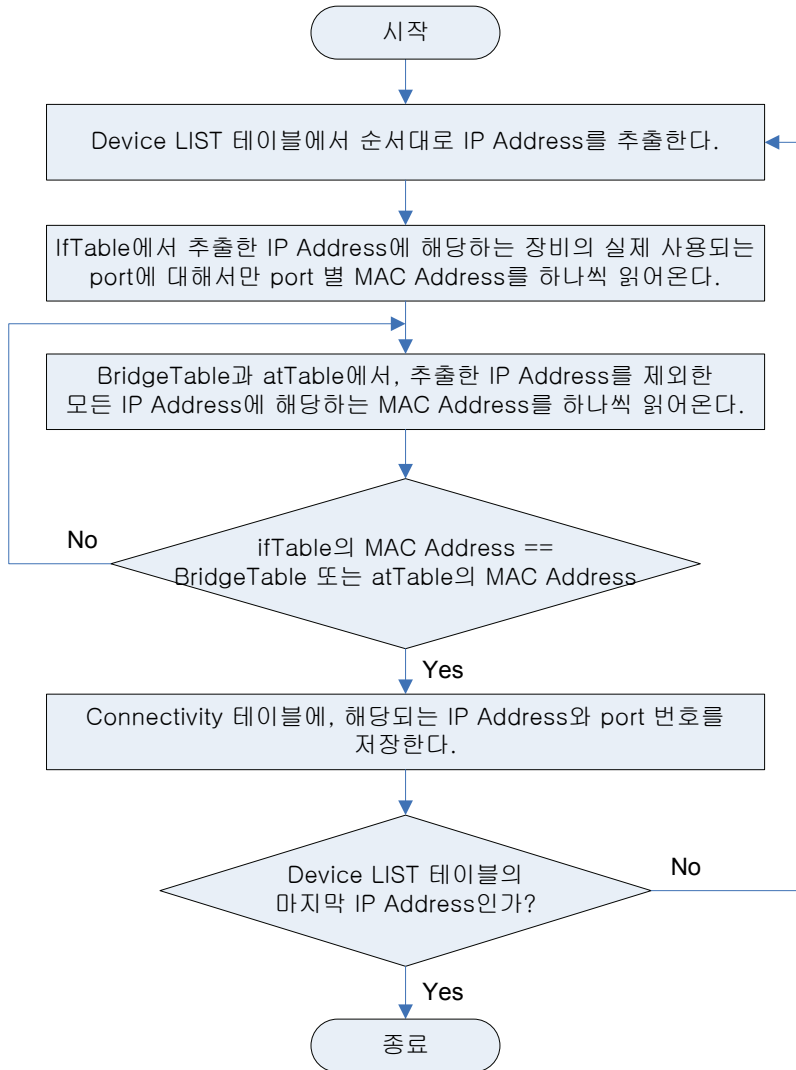


그림 8. 장비의 포트별 연결 정보를 생성하기 위한 순서도

실제 관리 장비 리스트에서 IP Address를 추출하고 추출한 IP Address에 해당하는 IfTable을 읽어온다. 그리고 IfTable에서 추출한 IP Address에 해당하는 장비의 실제 사용되는 포트에 할당되어 있는 MAC Address를 읽어온다. 다음으로 읽어온 MAC Address와 Bridge Table의 필드값 중 dot1dTpFdbAddress에 동일한 값이 있는지 비교한다.

비교해서 동일한 값이 있으면 IfTable에서 읽어온 IP Address(Source IP Address)와 포트 번호에 매칭되는 IP Address(Destination IP Address)와 포트 번호를 데이터베이스의 Connectivity 테이블에 저장한다. Bridge Table과 atTable의 끝까지 비교하고 다음 동일한 값이 나타나면 같은 방식으로 저장한다. Bridge Table과 atTable의 끝에 다다르면, 다음 IP Address를 읽어와 반복 수행한다. 모든 장비에 대해서 이 알고리즘을 수행하면, 모든 장비의 연결 정보가 검출되어 저장하게 된다.

저장된 연결 정보를 순서대로 읽어와 맵을 생성할 때, 먼저, 첫번째 Source IP Address 장비의 포트 번호에 Destination IP Address 장비의 포트 번호가 연결되어 있음을 읽어온다. 읽어온 Destination IP Address 장비와 포트 번호에 해당하는 값을 저장된 연결 정보의 Source IP Address 장비와 포트 번호에서 검색한다. 검색해서 동일한 값이 있으며, 이에 해당하는 Destination IP Address와 포트 번호가 첫번째로 읽어온 Source IP Address 장비와 포트 번호와 동일할 때에만 물리적으로 서로 연결되었다고 한다. 이와 같이 두 장비간의 포트별 연결정보는 각 장비가 상대 장비에 대해 포트별 연결정보가 서로 일치한 것에 대해서만 연결정보에 대한 물리적인 맵을 생성하게 된다.

Layer 2의 장비가 멀티레이어 장비로 Layer 3으로 동작하는 장비와 연결되어 있다면 Layer 3으로 동작하는 장비는 Bridge 기능을 하지 않기 때문에 Layer 2 장비의 MAC Address를 atTable에서 찾게 된다. 이에 따라 상대방의 포트번호와 자신의 VLAN MAC Address를 얻어 상대방 장비와 연결되어 있음을 알게 된다. VLAN은 말그대로 가상 LAN으로 여러 개의 포트를 하나로 묶어서 관리하게 되어 있으므로 본 논문에서 가져온 정보만으로는 어떤 포트와 연결되어 있는지 알 수 없다. 따라서 현재 얻은 정보로는 상대방 장비의 포트번호만 알 수 있다. 이 정보와 상대방 장비가 Layer 2 장비에서 얻은 연결 정보를 통해 장비의 포트별 연결 정보를 가져와 맵을 생성한다.

4.2 Database 설계

이 장에서는 본 시스템에서 사용되는 데이터베이스를 설계한다. 먼저 데이터베이스 스키마 다이어그램으로 전체 데이터들의 연관 관계를 설명하고, 각 테이블 별 세부 항목을 나열한다.

4.2.1 Database Schema Diagram

그림 9은 이 시스템에서 사용되는 데이터의 관계를 나타낸다. 편리한 설계를 위해 장비의 MIB의 object이름과 데이터베이스에 저장되는 테이블의 필드명을 동일하게 설계한다. 실제 적용에 있어서는 예약어에 해당하여 사용할 수 없는 필드명이 있으므로 수정하여 사용한다.

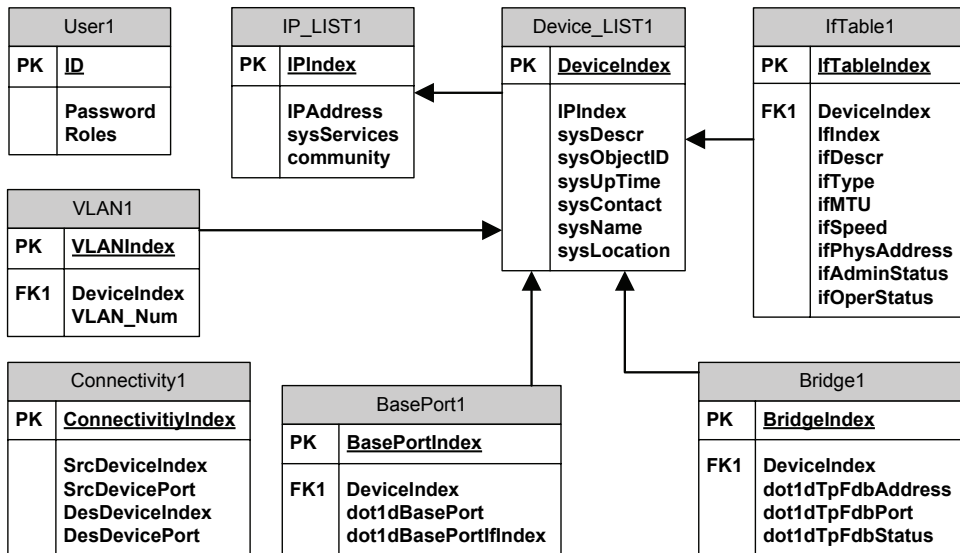


그림 9. 데이터베이스 스키마 다이어그램

‘User 테이블’은 사용권한과 인증을 위해 사용되며 관리자 시스템에 로그인을 위한 인증시에 사용된다. ‘IP_LIST 테이블’은 네트워크 관리 대역내의 모든 SNMP 에이전트가 탑재된 장비가 IP Address가 저

장되며, ‘Device_LIST 테이블’을 생성하는데 ‘IP_LIST 테이블’의 ‘IPAddress’ 필드 값이 이용된다. ‘Device_LIST 테이블’은 ‘IP_LIST 테이블’의 ‘IPAddress’ 필드 값을 가리키게 하여 중복되는 장비 없이 하나의 장비에 하나의 대표 IP Address가 매핑되게 장비들을 저장한다. ‘IP_LIST 테이블’의 각 IP Address는 대표적인 하나의 장비 IP로 그룹핑되기 위해 포함되는 ‘Device_LIST 테이블’의 ‘DeviceIndex’를 갖게 된다.

‘Device_LIST 테이블’의 PK(Primary Key)인 ‘DeviceIndex’는 ‘IP_LIST’, ‘IfTable’, ‘VLAN’, ‘BasePort’, ‘Bridge’ 테이블에서 참조되며 각각의 테이블들은 ‘DeviceIndex’를 FK(Foreign Key)로 가리킨다. ‘Device_LIST 테이블’의 ‘DeviceIndex’로 ‘IP_LIST 테이블’의 ‘IPAddress’ 값 즉, 실제 장비의 대표 IP Address로 나타낸다. 이 대표 IP Address를 기준으로 ‘IfTable’은 장비의 인터페이스에 해당하는 값을, ‘VLAN’은 장비의 VLAN 번호를, ‘BasePort’와 ‘Bridge’테이블은 장비의 Bridge와 관련 있는 포트와 MAC Address 정보를 저장한다. 마지막으로 ‘Connectivity 테이블’은 장비간의 포트별 연결 정보를 저장한다.

각각의 테이블 및 테이블의 필드값에 대한 자세한 설명은 아래에서 설명한다.

4.2.2 User Table

표 6은 사용자의 사용 권한을 관리자와 일반사용자로 나누어 제한하기 위해 사용되는 사용자 정보 테이블이다. 이 사용자 ID와 password를 사용하여 관리 시스템에 로그인하여 사용할 수 있다.

Field Name	Type	Description
ID	VARCHAR2	사용자의 ID, Primary Key에 해당
Password	VARCHAR2	사용자의 비밀번호
Roles	NUMBER	Admin(0), Normal(1)

표 6. 사용자 정보 테이블 항목

4.2.3 IP_LIST Table

표 7은 그림 5의 장비 스캐닝 알고리즘을 수행하면서 생성되는, 네트워크 장비의 IP Address 목록이 저장되는 테이블이다. ‘DeviceIndex’ 필드값은 장비의 대표 IP Address로 그룹핑을 수행하면서 ‘Device_LIST 테이블’이 생성된 후에 ‘DeviceIndex’ 필드값을 참조하여 채워지게 된다. ‘IP_LIST 테이블’에서 ‘DeviceIndex’가 동일한 값을 가진다는 것은 동일한 장비에 IP Address가 여러 개 할당되었음을 의미한다.

Field Name	Type	Description
IPIndex	NUMBER	이 테이블의 Primary Key
IPAddress	VARCHAR2	장비의 IP Address
sysServices	NUMBER	관리 대상 장비를 판단하는 기준이 되는 ‘sysServices’의 값을 저장함
community	CHAR	장비의 Read-only Community 정보
DeviceIndex	NUMBER	‘Device_LIST 테이블’ 생성시 ‘DeviceIndex’ 필드를 참조하여 채워짐

표 7. IP_LIST 테이블 항목

4.2.4 Device_LIST Table

표 8은 실제 네트워크에 존재하는 관리 대상 장비 목록을 저장하는 테이블의 필드이다. 그림 6의 장비 그룹핑 알고리즘을 수행하면서 생성되며 위의 IP_LIST 테이블의 IPAddress 필드값을 하나씩 읽으면서 해당 장비의 시스템 정보를 읽어와 중복 여부를 검사한다. 검사 후, 새로운 장비면 이 테이블에 추가하고 그 장비의 필드값인 ‘DeviceIndex’를 ‘IP_LIST 테이블’의 ‘DeviceIndex’에 추가한다. 중복된 장비면 ‘Device_LIST 테이블’에 추가없이 중복된 장비의 ‘DeviceIndex’를 ‘IP_LIST 테이블’의 ‘DeviceIndex’에 추가한다.

Field Name	Type	Description
DeviceIndex	NUMBER	이 테이블의 Primary Key
IPIndex	VARCHAR2	'IP_LIST 테이블'의 'IPIndex'를 참조하는 Foreign Key, 대표 IP Address의 Index를 나타냄
sysDescr	VARCHAR2	시스템의 운영 체제에 대한 정보
sysObjectID	VARCHAR2	엔티티에 포함되는 전산망 관리 서비스 시스템 제공자의 권한 구별 정보
sysUpTime	VARCHAR2	시스템이 마지막으로 재 초기화 된 이후로 1/100초 단위로 계산된 현재까지 시간의 총합
sysContact	VARCHAR2	장비 관리자와의 연락방법
sysName	VARCHAR2	이 장비를 위해 관리상 할당된 이름
sysLocation	VARCHAR2	장비의 물리적인 위치 정보

표 8. Device_LIST 테이블 항목

4.2.5 IfTable Table

인터페이스 그룹은 네트워크 시스템의 인터페이스에서 발생할 수 있는 트래픽의 통계와 조직정보를 포함하여, 개체의 물리적 인터페이스와 관련된 정보를 가진다. 이러한 인터페이스 그룹의 'ifTable'에서 아래 표 9의 필드명에 해당하는 값을 가져와서 저장한다. 'Device_LIST 테이블'의 'DeviceIndex'를 순서대로 읽으면서 'IPIndex'에 해당하는 'IP_LIST 테이블'의 'IPAddress'를 가져와 해당되는 장비에서 아래의 필드명에 해당하는 값을 가져와서 저장한다.

Field Name	Type	Description
IfTableIndex	NUMBER	이 테이블의 Primary Key
DeviceIndex	NUMBER	'Device_LIST 테이블'의 'DeviceIndex'를 참조하는 Foreign Key

ifDescr	VARCHAR2	인터페이스의 정보(인터페이스의 hw/sw의 버전, 상품명, 제조회사명)
ifType	VARCHAR2	인터페이스의 타입
ifMtu	NUMBER	이 인터페이스에서 주고 받을 수 있는 최대 패킷의 크기
ifSpeed	NUMBER	초당 현재 인터페이스의 속도
ifPhysAddress	VARCHAR2	각 인터페이스에 해당하는 물리적 주소
ifAdminStatus	VARCHAR2	Read-write, Up(1), Down(2)을 관리자가 설정 가능. testing(3) 상태도 있음
ifOperStatus	VARCHAR2	Read-only, 현재 인터페이스의 실제 동작상태. Up(1), Down(2), testing(3), unknown(4), dormant(5), notPresent(6), lowerLayerDown(7)의 상태 있음

표 9. IfTable 항목

4.2.6 VLAN Table

4.1.3장에서 설명한 포트 매핑을 위해 필요한 Bridge MIB을 액세스 하기 위하여 Community String Indexing을 이용한다. Community String Indexing의 문법은 [community string]@[instance number]이고, 이 중 [instance number]에 해당되는 부분은 MIB-II의 'ifTable'의 'ifDescr'값 중 VLAN을 제외한 나머지부분에 해당된다. 각 장비에 해당하는 모든 [instance number]가 표 10의 'VLAN'필드에 저장된다.

Field Name	Type	Description
VLANIndex	NUMBER	이 테이블의 Primary Key
DeviceIndex	NUMBER	'Device_LIST 테이블'의 'DeviceIndex'를 참조하는 Foreign Key
VLAN	VARCHAR2	'IfTable'의 object인 'ifDescr'의 String 중 VLAN을 제외한 값

표 10. VLAN 테이블 항목

4.2.7 BasePort Table

표 11의 ‘BasePort 테이블’은 ‘Bridge 테이블’의 ‘dot1dTpFdbPort’에 해당하는 값과 ‘IfTable 테이블’의 ‘ifIndex’를 매칭시키기 위해 저장한다. 즉, 실제 장비에서 사용하는 포트 번호와 Bridge MIB에서 사용하는 포트 번호 매칭에 이용된다. ‘Bridge 테이블’의 ‘dot1dTpFdbPort’의 값은 이 테이블의 ‘dot1dBasePort’에 해당되고, ‘dot1dBasePortIfIndex’는 MIB-II의 ‘IfTable’의 ‘ifIndex’와 같은 값에 매칭된다.

Field Name	Type	Description
BasePortIndex	NUMBER	이 테이블의 Primary Key
DeviceIndex	NUMBER	‘Device_LIST 테이블’의 ‘DeviceIndex’를 참조하는 Foreign Key
dot1dBasePort	NUMBER	Bridge 관리 정보를 가진 엔트리의 포트 번호
dot1dBasePortIfIndex	NUMBER	MIB-II의 ‘IfTable’의 ‘ifIndex’와 같은 값에 해당

표 11. BasePort 테이블 항목

4.2.8 Bridge Table

Bridge-MIB의 dot1dTp 그룹에서 dot1dTpFdbTable의 값은 실제 장비의 각 포트가 패킷을 포워딩 또는 필터링하면서 Bridge 정보를 학습하고 학습한 Mac Address를 가진다. 표 12는 이 정보를 읽어와 Bridge 테이블의 각각의 필드에 저장한다. 또한, 추가로 Layer 3로 동작하는 장비를 위해 MIB-II의 at그룹에서 atTable의 값 중 atIfIndex와 atPhysAddress를 Bridge Table에 저장한다. 본 논문에서 장비간 포트별 연결 정보를 알아내는데 사용되는 필수 테이블에 해당한다.

Field Name	Type	Description
BridgeIndex	NUMBER	이 테이블의 Primary Key

DeviceIndex	NUMBER	‘Device_LIST 테이블’의 ‘DeviceIndex’를 참조하는 Foreign Key
dot1dTpFdbAddress (atPhysAddress)	VARCHAR2	포워딩 또는 필터링 정보를 가진 Bridge를 위한 유니캐스트 Mac Address
dot1dTpFdbPort (atIfIndex)	NUMBER	‘dot1dTpFdbAddress’에 대응되는 Source Address를 가진 장비의 포트 번호
dot1dTpFdbStatus	VARCHAR2	엔트리의 상태로 other(1), Invalid(2), Learned(3), Self(4), Mgmt(5) 상태 존재

표 12. Bridge Table 항목

4.2.9 Connectivity Table

‘Device_LIST 테이블’의 ‘DeviceIndex’ 순서대로 읽으면서 ‘IPIndex’에 해당하는 ‘IP_LIST 테이블’의 IP Address를 가져온다. 가져온 IP Address로 ‘IfTable 테이블’ 장비의 각 포트에 해당하는 MAC Address를 읽어서 ‘Bridge 테이블’에 저장되어 있는 ‘dot1dTpFdbAddress’와 동일한지 비교한다. 동일한 MAC Address가 발견되면, ‘SrcDeviceIndex’와 ‘SrcDevicePort’는 현재 비교하고 있는 MAC Address에 해당하는 정보를 ‘DesDeviceIndex’와 ‘DesDevicePort’는 ‘Bridge 테이블’에서 매칭된 값으로, 비교 대상이 되는 ‘dot1dTpFdbAddress’와 ‘dot1dTpFdbPort’ 부분을 아래의 표 13에 각 필드 값으로 저장한다.

Field Name	Type	Description
ConnectivityIndex	NUMBER	이 테이블의 Primary Key
SrcDeviceIndex	NUMBER	‘Source Address’에 해당하는 Index
SrcDevicePort	NUMBER	‘SrcDeviceIndex’에 해당하는 포트 번호
DesDeviceIndex	NUMBER	‘SrcDeviceIndex’와 물리적으로 연결된 ‘Destination Address’에 해당하는 Index
DesDevicePort	NUMBER	‘DesDeviceIndex’에 해당하는 포트 번호

표 13. 연결 정보 테이블 항목

4.3 시스템 구조

대부분의 네트워크 관리 시스템들처럼 본 논문의 웹 기반의 연결 정보 관리 시스템도 관리자와 대리자간의 요청-응답 구조를 가진다.

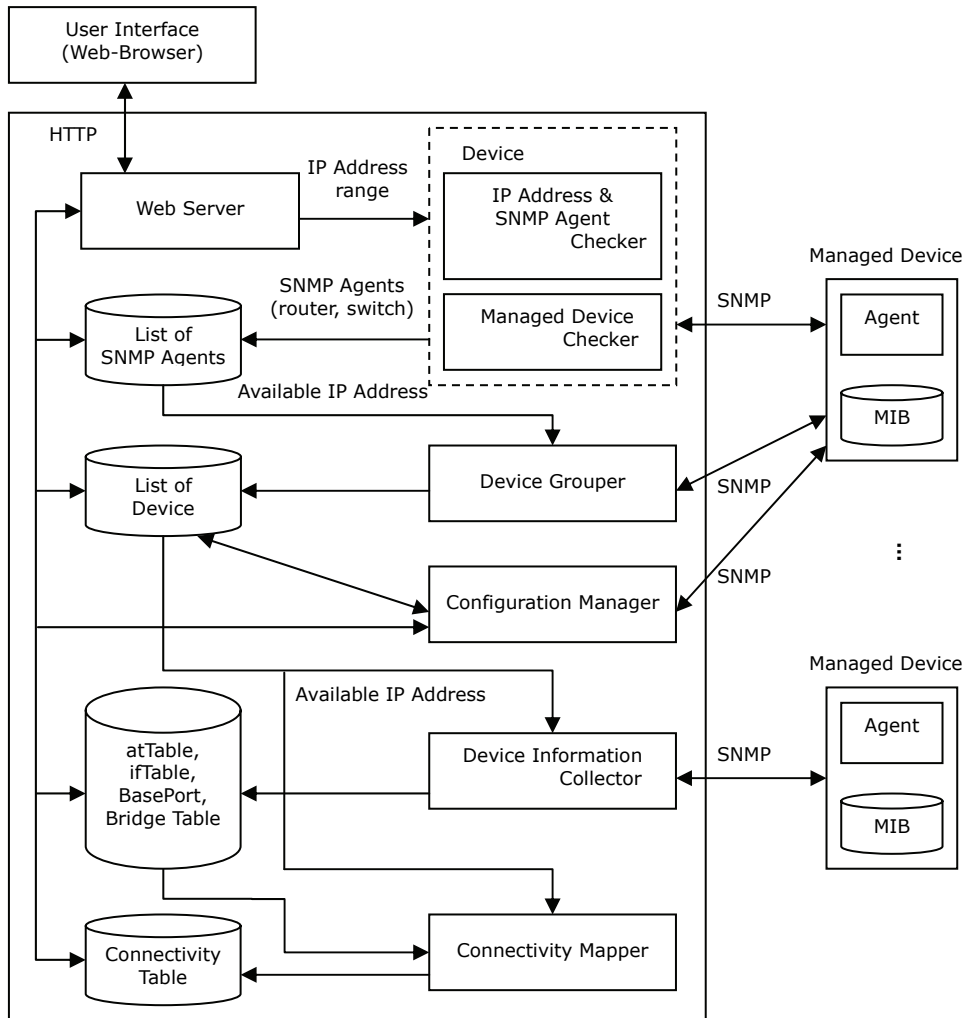


그림 10. 전체 시스템 구조

그림 10은 웹 기반의 포트별 연결 정보 관리 시스템에 관한 설계 구조로 장비 스캐너(Device Scanner), 장비 그룹퍼(Device Grouper), 장비

정보 수집기(Device Information Collector), 연결 정보 작성자(Connectivity Mapper), 구성 관리자(Configuration Manager)와 사용자 인터페이스(User Interface)등이 주요한 모듈로 이루어진다. 연결 정보 저장 및 관리를 위해 데이터베이스가 사용되며 데이터베이스도 주요한 모듈 중 하나이다.

4.3.1 장비 스캐너(Device Scanner)

장비 스캐너는 IP Address 범위 내에 있는 SNMP 에이전트를 탑재한 장비를 검색하는 모듈이다. 외부로부터 IP Address 범위를 입력 받아 해당되는 모든 IP Address로 sysServices를 요청하는 SNMP Get 을 보낸다. 이에 대한 응답의 유무로 IP Address의 사용여부와 동시에 SNMP 에이전트의 탑재 여부를 체크하게 된다. 또한, sysServices의 값으로 연결 정보를 구성하기 위한 관리 대상 장비인지를 판단하게 된다. 결과적으로 이 모듈을 수행한 후에 실제 관리 대상이 되는 장비의 IP 목록을 얻게 된다. 이 모듈은 그림 5의 관리 대상인 장비 리스트를 생성하기 위한 순서도에 해당하는 역할을 수행한다.

4.3.2 장비 그룹퍼(Device Grouper)

라우팅 기능을 제공하는 장비인 경우 하나의 장비에 여러 개의 IP 주소가 할당 되어 있을 수 있으므로 장비 스캐너를 통해 얻은 장비의 IP 목록에는 중복된 장비가 존재할 수 있다. 장비 그룹퍼 모듈은 중복되는 장비를 하나의 대표 IP로 그룹핑하는 기능을 수행한다. SNMP 장비 스캐너에 의해 작성된 장비 목록을 입력 받아 장비 그룹퍼에 의해 그룹핑을 실행하게 되면 최종적으로, 하나의 장비에 하나의 대표 IP가 매핑된 장비 목록을 얻게 된다. 이 모듈은 그림 6의 장비 리스트 중 동일한 장비를 그룹핑하기 위한 순서도에 해당하는 역할을 수행한다. 장비 그룹퍼에 의해 중복 없이 하나의 장비에 IP Address가 할당되어 실제 관리 대상 장비 목록을 얻게 된다.

4.3.3 장비 정보 수집기(Device Information Collector)

장비 그룹퍼에 의해 작성된 실제 관리 될 장비의 IP Address 목록을 순서대로 입력 받아 포트별 IP 매칭을 위해 필요한 MIB 정보인 MIB-II의 ifTable과 Bridge-MIB의 dot1dBasePortTable, dot1dTpFdbTable을 각각의 장비에 요청하여 저장한다. 또한, Layer 3로 동작하는 스위치에 대해서 MIB-II의 atTable을 가져와 저장한다. 이 모듈은 그림 7의 각 장비의 포트 매칭을 위한 정보를 수집하는 순서도에 해당하는 역할을 수행한다.

4.3.4 연결 정보 작성자(Connectivity Mapper)

연결 정보 작성자 모듈은 그림 8의 장비 별 연결 정보를 생성하기 위한 순서도에 해당하는 모듈이다. 실제 관리 대상 장비 목록의 IP Address를 순서대로 가져와 장비의 각 포트에 해당하는 MAC Address를 장비 정보 수집기에 의해 수집된 정보 중 dot1dTpFdbTable의 dot1dTpFdbAddress와 atTable의 atPhysAddress와 비교하여 장비간 실제 물리적인 연결 정보를 파악하는 기능을 수행한다. 이 모듈의 실행 후, 최종적으로 장비간 포트별 연결 정보 목록을 얻게 된다.

4.3.5 구성 관리자(Configuration Manager)

실제 관리 장비 목록에서 장비를 선택하게 되면 선택된 장비에 대해 IP Address를 입력 받아, 선택된 장비의 구체적인 정보와 포트의 사용여부 상태 등을 확인할 수 있으며 커뮤니티 변경 및 포트의 사용여부의 변경, 대표 IP Address의 변경을 하는 기능을 제공한다.

4.3.6 Web Server

사용자 인터페이스를 지원하기 위해 시스템에 웹 서버가 필요하

며 사용자의 관리 대상 목록, Connectivity Mapper에 의해 작성된 장비간의 포트별 물리적인 연결 정보를 확인할 수 있는 기능을 제공한다.

V. 구현

4장에서 설명한 설계 구조를 바탕으로 관리자가 웹 인터페이스를 통해 네트워크를 효율적으로 관리할 수 있도록 네트워크 구성도를 자동으로 생성하고 관리 정보를 쉽게 변경할 수 있는 시스템을 구현하였다. 이 장에서는 4장의 설계를 바탕으로 구현한 시스템에 대해서 자세히 살펴본다.

5.1 개발환경

웹 기반의 네트워크 구성 관리 시스템은 다음과 같은 환경에서 개발되었다.

- 하드웨어 개발 환경

CPU	Intel Pentium 4 CPU 2.40GHz
Memory	1GB

- 소프트웨어 개발 환경

Operating System	Windows 2000 Service pack 4
Language	Java (JDK 1.5.0_06)
IDE	Eclipse 3.1
Library	AdventNet SNMP API[22]
Web Server	Apache Tomcat 5.5
Web Client	JSP(Java Server Pages)
Database	Oracle database 10g[23]

5.2 Configuration file

외부로부터 입력 받은 네트워크 관리 대역을 저장하고 장비의 존재 여부 및 장비간 연결 정보에 대한 데이터베이스를 최신정보로 유지하기 위해 반복적으로 수행될 주기가 Configuration 파일에 저장된다.

- StartIP: 네트워크 관리 대역에서 시작되는 IP Address
- EndIP: 네트워크 관리 대역에서 끝나는 IP Address
- Cycle: 연결 정보를 갱신하기 위해 반복적으로 수행할 주기

5.3 네트워크 장비 스캐너

네트워크 장비 스캐너는 외부로부터 IP Address 범위를 입력 받아 해당되는 모든 IP Address로 sysServices를 요청하는 SNMP Get 을 보내어 IP Address 범위 내에 있는 SNMP 에이전트를 탑재한 장비에 대한 응답이 있는 장비에 대해서만 sysServices의 값을 판단하여 IP LIST 테이블에 저장되도록 구현되었다. 그림 11은 POSTECH 네트워크에 대해서 네트워크 장비 스캐닝을 수행한 후에 실제 관리 대상이 되는 장비의 IP 목록을 보여주는 화면이다.

IPIndex	IPAddress	sysServices	community
1	141.223.140.251	2	culture
2	141.223.140.252	2	culture
3	141.223.140.99	6	culture
4	141.223.141.252	2	culture
5	141.223.141.251	2	culture
6	141.223.142.99	6	culture
7	141.223.142.251	2	culture
8	141.223.142.252	2	culture
9	141.223.142.253	2	culture
10	141.223.142.255	6	culture
11	141.223.143.99	6	culture
12	141.223.143.252	2	culture
13	141.223.143.251	2	culture
14	141.223.144.255	6	culture
15	141.223.144.99	6	culture
16	141.223.144.252	2	culture
17	141.223.144.251	2	culture
18	141.223.145.99	6	culture
19	141.223.145.255	6	culture

그림 11. 장비 스캐닝 수행 후 결과화면

POSTECH의 네트워크 대역인 ‘141.223.1.1’부터 ‘141.223.255.255’에 대해서 스캐닝을 했다. 그림 11의 ‘IPIndex’가 6인 ‘141.223.208.99’의 장비는 실제 POSTECH 전산소에서 관리하는 기숙사의 메인 스위치에 해당하며 ‘141.223.254.84’의 대표 IP Address로 관리되는 실제 장비에 해당한다.

5.4 장비 그룹핑

장비 그룹핑은 동일한 장비를 하나의 IP Address로 묶기 위한 기능을 수행한다. 그림 12의 화면은 하나의 대표 IP로 그룹핑 됐음을 확인할 수 있다. 그림 12의 화면에서 보듯이 하나의 장비에 여러개의 IP Address가 할당된 장비에 대해서만 IPGROUP의 펼침 목록 메뉴가 나타나며 이 목록을 누르면 하나의 장비에 할당된 IP Address 목록을 확인할 수 있으며 관리자가 원하는 IP Address를 선택한 후, 수정 버튼을 누르면 대표 IP가 선택한 IP Address로 수정된다.

Index	Device Name	IPAddress	IPGroup	
11	APT8FX-1.postech.ac.kr	141.223.144.251		delete
12	APT8TX-1.postech.ac.kr	141.223.144.252		delete
13	APT9FX-1.postech.ac.kr	141.223.145.251		delete
14	SaengMyeong.postech.ac.kr	141.223.148.99	none	delete
15	Jigokplaza1-1.postech.ac.kr	141.223.149.91	none	delete
16	Jigokplaza2-2.postech.ac.kr	141.223.149.94	141.223.254.188	delete
17	APT9TX-1.postech.ac.kr	141.223.145.252	141.223.128.99	delete
18	Jigokplaza3-1.postech.ac.kr	141.223.149.96	141.223.148.99	delete
19	Jigokplaza3-2.postech.ac.kr	141.223.149.95		delete
20	Jigokplaza2-1.postech.ac.kr	141.223.149.93		delete
21	Jigokplaza3-2.postech.ac.kr	141.223.149.97		delete
22	Jigokplaza1-2.postech.ac.kr	141.223.149.92		delete
23	Jigokplaza3-3.postech.ac.kr	141.223.149.98		delete
24	Jigokplaza-Master.postech.ac.kr	141.223.149.99	none	delete
25	SYSYOON-AP.postech.ac.kr	141.223.150.160		delete
26	Jigokplaza_3F.postech.ac.kr	141.223.149.100		delete

그림 12. 장비 그룹핑 수행 후 결과화면

그림 12에서 Index 14에 해당하는 POSTECH 생명과학동 백본 스위치는 IP 목록의 첫번째 IP Address인 '141.223.148.99'가 기본값으로 대표 IP로 할당되었다. 이 장비에 할당된 IP Address로 '141.223.254.188', '141.223.129.99', '141.223.128.99', '141.223.148.99' 등을 확인할 수 있다.

그림 13의 화면은 장비 리스트를 확인 할 수 있는 그림 12에서 관리자가 원하는 대표 IP Address로 변경된 것을 보여준다. IP Address가

‘141.223.148.99’인 생명과학동 백본 스위치는 실제 POSTECH 전산소에 서는 ‘141.223.254.188’로 관리하므로 이 IP Address로 대표 IP를 변경했 다. 그림 13의 화면은 대표 IP가 ‘141.223.254.188’로 변경됐음을 확인 할 수 있다. 또한, 이 장비 리스트에서 사용되지 않는 장비에 대한 삭 제가 가능하다.

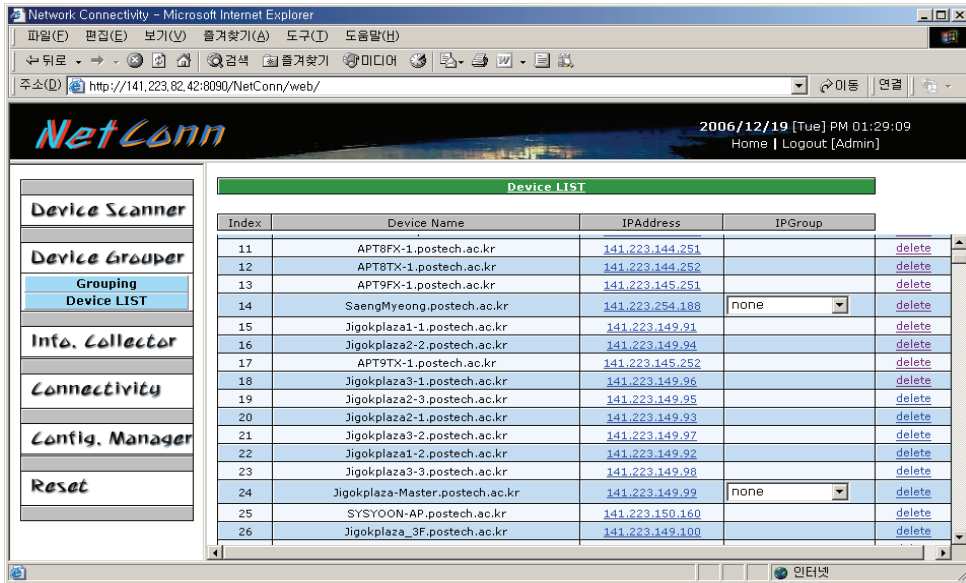


그림 13. 장비 리스트 화면에서 대표 IP 변경 후 결과화면

5.5 네트워크 구성도

그림 14은 장비간의 포트별 연결 정보를 확인 할 수 있는 화면이 다. 연결 정보 화면은 윈도우 탐색기 형식으로 구성되어 각 장비의 대 표 IP Address를 기준으로 포트별 연결 정보를 트리 구조로 보여주며 장 비 별로 포트의 연결상태를 확인 할 수 있도록 열고 닫을 수 있는 기 능이 추가되었다. 그림 14에서는 ‘141.223.254.1’에 해당되는 루트장비를 선택해야 하며 장비들로부터 트리 형태로 연결 정보를 이용해서 트리 를 확장해 나간다. 그림 14에서 실제 POSTECH 네트워크는

‘141.223.254.1’이 CORE1 스위치며 그 아래에 연결된 장비 중 다섯번째 장비인 ‘141.223.254.148’은 정보통신연구소의 스위치로 POSTECH 전산소에서 제공하는 연결 정보와 같음을 확인했다. 연결 정보의 트리구조를 맵으로 변환하는 것은 향후 연구과제 중 하나이다.

141.223.254.1의 각 포트에 연결된 장비들로는 포트 50번에 연결된 141.223.254.228(1), 41번에 연결된 141.223.254.156(11), 5번에 연결된 141.223.254.20(11) 등을 확인 할 수 있다. 뒤에 쓰인 소괄호 안의 숫자는 각 장비에서 141.223.254.1에 연결된 포트 번호를 의미한다.

그림 14은 141.223.254.148에 연결된 모든 장비 리스트를 확인할 수 있도록 리스트를 열어서 화면을 캡처한 것이며, 화면에서 첫번째 대괄호안의 숫자는 연결된 상대 장비의 포트를 의미하고 두번째 대괄호의 숫자는 상대 장비와 연결된 자신의 장비의 포트를 의미한다. 즉, ‘[40][11]141.223.254.148’의 의미는 ‘141.223.254.1’의 40번 포트와 ‘141.223.254.148’ 장비의 11번 포트가 연결되었음을 나타낸다. 이 포트 매칭 정보가 실제 POSTECH 전산소에서 제공하는 정보와 일치함을 확인했다.

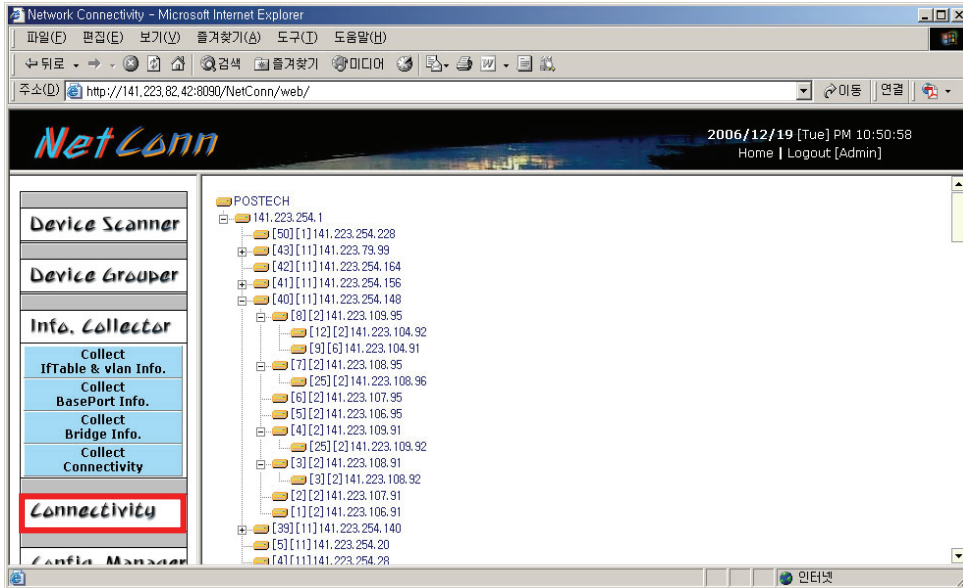


그림 14. 네트워크의 포트별 연결 정보 화면

5.6 장비의 상세 정보

그림 15는 그림 14에서 특정 장비의 상세 정보를 보고자 하나의 장비를 선택했을 때 볼 수 있는 화면에 해당한다. 장비를 선택한 후, 클릭하면 왼쪽의 연결 정보 화면과 오른쪽의 장비의 상세 정보를 동시에 볼 수 있는 페이지로 화면이 배치된다. 연결 정보의 왼쪽화면에서 선택된 141.223.254.148의 장비에 관한 상세 정보가 오른쪽 화면에 나타나며 시스템정보 및 각 포트별 장비의 상태 및 MAC Address정보를 확인 할 수 있다. 이 웹 인터페이스에서 IPAdmiStatus의 상태를 UP(1)에서 DOWN(2)상태로 변경하여 포트를 차단시킬 수 있는 기능을 제공한다.

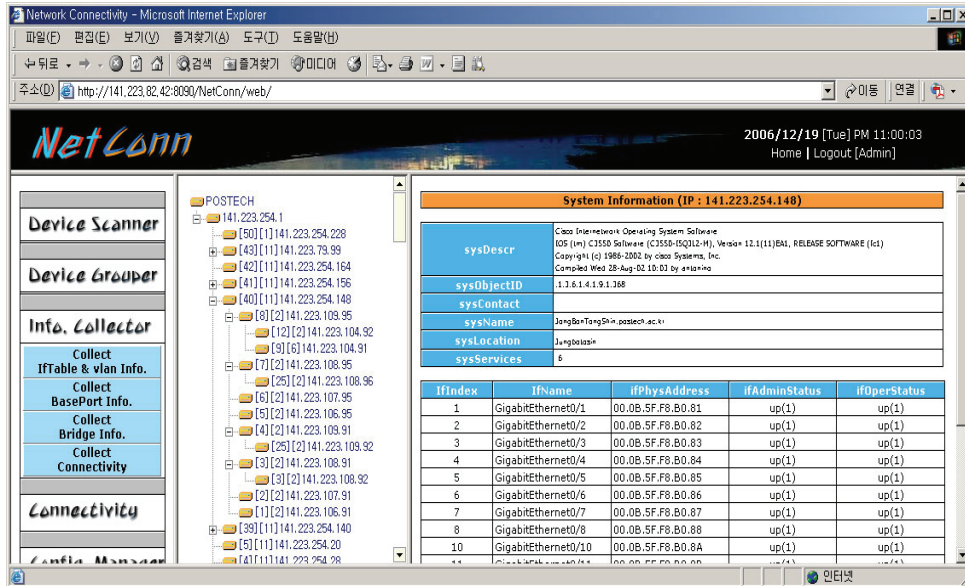


그림 15. 특정 네트워크 장비의 상세 정보 화면

VI. 결론 및 향후 과제

본 논문에서는 네트워크의 연결 정보 구성 관리를 효율적으로 수행할 수 있도록 네트워크 장비간의 물리적인 연결 정보를 알아내는 시스템을 개발했다. 먼저 SNMP 에이전트가 탑재되어 있는 모든 장비를 검색한 후 장비간의 포트별 연결 정보를 추출한다. 포트별 연결 정보를 알아내기 위하여 Bridge MIB을 사용하는데 Layer 2로 동작하는 네트워크 장비는 전달되는 패킷 헤더의 목적지 MAC Address를 보고 그 패킷을 다음 노드로 전달한다. 이 때 목적지 MAC Address를 학습하여 학습한 내용을 Bridge MIB에 저장한다. 또한, Layer 3으로 동작하는 장비는 atTable의 MAC Address를 보고 장비간의 연결 정보를 저장한다. 본 논문은 이들 정보를 장비 포트의 각각의 MAC Address와 비교 분석하여 장비간의 연결 정보를 알아낸다.

본 논문에서는 연결 정보 구성 관리 시스템을 개발하기 위한 방법으로 포트별 IP 매칭 알고리즘을 제안하였다. 그리고 그에 따른 요구사항을 분석하고 시스템을 설계, 구현하여 네트워크 관리에 필수적인 네트워크 구성도를 생성하는 방법을 제시하였다. 네트워크의 물리적인 연결정보를 찾아 맵을 구성하는 상용제품이 있지만, 연결 정보를 취득하는 구체적인 방법에 대해서는 공개하지 않고 있어 그 방법을 알 수 없다. 따라서, 본 논문은 네트워크 장비 검색 및 장비간 연결 정보를 알아내는 방법을 표준에 의거한 상세한 알고리즘을 제시했다는 점에서 의의가 있으며 또한, 제시한 알고리즘을 따라 구현한 후, POSTECH 네트워크에 실제 적용하여 알고리즘을 검증함으로써 실제 네트워크 관리에 도움을 주는 시스템을 설계 및 구현했다는 점에 의의를 둘 수 있다.

이 시스템이 제공하는 장비간 연결 정보를 이용하여 라우터나 스위치 같은 장비들의 연결 정보를 웹을 통하여 쉽게 확인이 가능하며, 장비의 정보를 웹을 통해 쉽게 접근하여 검색 또는 수정 하는 일들을 통하여 필요할 때 사용하여 쉽게 문제를 일으키는 장비를 차단하는 작

업을 원격으로 수행할 수 있다.

향후 과제로는 연결 정보의 트리구조를 맵으로 변환해 좀 더 쉬운 사용자 인터페이스를 제공하는 것이다. 네트워크의 규모가 커져 장비의 수가 늘어나도 적정 시간내에 장비를 검색할 수 있는 scalability 보장에 관한 연구가 이루어져야 한다. 또한, 장비를 검색함에 있어서 전체 네트워크에 대한 탐색이므로 빠른 시간 내에 장비 검색이 이루어지도록 performance를 향상시키는 연구가 이루어져야 한다. 부가적으로 네트워크 장애 발생시, 실제 문제가 발생한 링크 또는 네트워크 장비를 최대한 빨리 발견하기 위해 네트워크 구성도를 서로 다른 색으로 표시하여 쉽게 문제지점을 발견할 수 있도록 더 나은 GUI를 제공해야 한다.

참고 문헌

- [1] Bruce Lowekamp, David R. O'Hallaron, Thomas R. Gross, "Topology Discovery for Large Ethernet Networks", SIGCOMM'01, pp. 237~248, August 27-31, 2001, San Diego, California, USA.
- [2] 박윤규, "웹 기반의 네트워크 트래픽 모니터링 시스템의 자동 구성 및 재구성 기법", Masters Thesis, POSTECH GSIT, 2000.
- [3] Hwa-Chun Lin, Hsin-Liang Lai, Shou-Chuan Lai, "Automatic link layer topology discovery of IP networks", In Proceeding of IEEE International Conference on Communications (ICC'99), pp. 1034~1038, June 1999.
- [4] William Stallings, "SNMP, SNMPv2, SNMPv3 and RMON 1 and 2", Third Edition, Addison-Wesley, 1999.
- [5] The Internet Engineering Task Force (IETF), "IETF Home Page", <http://www.ietf.org>, Refer Dec. 2006.
- [6] J.Case, et al, "Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)", IETF, RFC 1907, January 1996.
- [7] J.Case, "Introduction to Version 3 of the Internet-standard Network Management Framework", IETF, RFC 2570, April 1999.
- [8] M. Rose, K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets," IETF, RFC 1155, May 1990.
- [9] K.McCloghrie, et al, "Structure of Management Information Version 2 (SMIv2)", IETF, RFC 2578, April 1999.
- [10] K. McCloghrie, M. Rose, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", IETF, RFC1213, March 1991.
- [11] E. Decker, "Definitions of Managed Objects for Bridges", IETF, RFC 1493, July 1993.
- [12] Y. Breitbart, M. Garofalakis, C. Martin, R. Rastogi, S. Seshadri, and A. Silberschatz, "Topology discovery in heterogeneous IP networks", In Proceedings of IEEE INFOCOM 2000, pp. 265~274, March 2000.
- [13] Cisco Systems, Inc, "Understanding Spanning Tree Protocol", http://www.cisco.com/univercd/cc/td/doc/product/rtrmgmt/sw_ntman/cwsimain/cwsi2/cwsiug2/vlan2/stpapp.htm, Refer Dec. 2006.
- [14] AdventNet Inc., "Switch Port Mapper Tool", <http://manageengine.adventnet.com/products/oputils/switch-port-mapper.html>, Refer Dec. 2006.
- [15] AdventNet Inc., "ManageEngine™ OpUtils 4", <http://manageengine.adventnet.com/products/oputils/index.html>, Refer Dec. 2006.
- [16] Cisco Systems, Inc., "Network Assistant Feature", http://www.cisco.com/univercd/cc/td/doc/product/rtrmgmt/cna/v4_0/gsg4_1/

- gsg_en/feature.htm, Refer Dec. 2006.
- [17] Cisco Systems, Inc, “Cisco Smartports”,
http://www.cisco.com/en/US/netsol/ns439/networking_solutions_package.html, Refer Dec. 2006.
 - [18] IBM, “Netcool/Precision for IP Networks”, <http://www-306.ibm.com/software/tivoli/products/netcool-precision-ip/index.html>, Refer Dec. 2006.
 - [19] VPN Consortium, “Virtual Private Network Consortium”,
<http://www.vpnc.org>, Refer Dec. 2006.
 - [20] Cisco Systems, Inc., “SNMP Community String Indexing”,
<http://www.cisco.com/warp/public/477/SNMP/camsnmp40367.html>, Refer Dec. 2006.
 - [21] Cisco Systems, Inc, “How To Get Dynamic CAM Entries (CAM Table) for Catalyst Switches Using SNMP”,
http://www.cisco.com/warp/public/477/SNMP/cam_snmp.html, Refer Dec. 2006.
 - [22] AdventNet Inc, “AdventNet API”,
<http://snmp.adventnet.com/help/snmpapi/snmpv1/>, Refer Dec. 2006.
 - [23] OSTC Inc. “Oracle Database HOWTO”,
<http://www.linux.com/howtos/Oracle-7-HOWTO.shtml>, Refer Dec. 2006.

감사의 글

부족한 저를 이곳으로 인도하셔서 2년간의 공부를 무사히 마칠 수 있게 하신 하나님께 감사를 드립니다. 또한, 언제나 든든한 버팀목이셨던 어머니께 감사를 드립니다.

DPNM 연구실에서 생활할 수 있게 하시고 부족한 저를 받으셔서 지난 2년 동안 누구보다도 큰 격려를 하였고, 열심히 지도해주신 홍원기 교수님께 큰 감사를 드립니다. 제가 제자리에 머물러 있기보다는 늘 앞으로 전진하는 것을 바라셨고 전진했을 때 누구보다도 기뻐하시며 칭찬하셨던 것을 기억합니다. 칭찬과 격려가 지금의 저를 이자리에 서게 만들었습니다. 다시 한번 감사드리며, 앞으로 어디에 있는 교수님께 배운 가르침을 바탕으로 주어진 일에 최선을 다하며 성실한 사람이 되도록 하겠습니다. 더불어 저의 논문 심사를 위해 애써주신 서영주 교수님과 송황준 교수님께도 깊은 감사를 드립니다.

1년차 시절부터 먼 대구에서 이곳 포항까지 매주 오셔서 저의 연구를 지도해주시고 아낌없는 격려해주신 계명대 교수님이시자 연구실 선배인 주홍택 박사님께 큰 감사를 드립니다. 또한, 석사생활 시작과 끝에 함께 하시며 큰 도움을 주셨던 지금 본교 연구교수이신 최미정 박사님께도 감사의 뜻을 전합니다. 그리고, 논문 발표전 오셔서 큰 격려하고 가신 지금은 고려대 교수님이신 김명섭 박사님께도 감사의 마음 전합니다. 현재, 랩장으로 든든하게 연구실을 지키며 싫은 기색 한번 없이 내공부에 많이 도움을 줬던 동기 준명이, 때론 친구처럼 온갖 투정 다 들어주면서 해결책 제시하던 착한 영준이, 여기저기 아픈데가 많아서 늘 마음 아팠지만 지금은 잘 웃고 많이 좋아진 귀여운 성철이, 주어진 일에 대한 대단한 열정과 한번도 화를 내지 않았다는 성격 좋은 창근이, 클라리넷 연주로 마음 설레게 했던 클라리넷 스승이자 다방면에 재주가 많은 멋진 병철이, 한동안 눈을 못 마주치고 바닥만 바라보고 다녀 수줍음 많은 줄 알았던 일 잘하는 광본이, 연구실 살림을 책

임지고 있으며 주어진 일에 열심을 다하는 예쁜 민정이, 모두에게 깊은 감사의 마음을 전합니다.

지금은 연구실에 없지만 저의 소중한 인연들, 마음이 잘 맞고 때론 언니 같았던 동생 영미, 같이 공부할 수 없어 가슴이 많이 아팠습니다. 많은 날들을 같이한 마음 따뜻한 소정이, 더 잘해주지 못해 미안하고 고마운 마음 전합니다. 지금은 영국으로 이민갔지만 누구보다도 맘 터놓고 지냈던 보고픈 디팔리, 힘들어 전화할 때마다 따뜻한 격려와 때마다 적절한 문자메시지로 멀리서 나의 졸업을 도운 선미에게도 감사의 마음 전합니다. 지금은 결혼한 예쁜 새신부 효진이, 착한 현미 등 그동안 같이 생활했던 모든분들께 감사의 인사 전하며 각자의 위치에서 최선을 다하여 앞으로 더 좋은 모습으로 만나게 되기를 기도합니다.

나의 단짝 친구 김하나! 나를 넘어설 수 있게 기도로 물질로 시간으로 도운 나의 귀한 친구 하나에게 큰 감사의 인사 전합니다. 또한, 나의 든든하고 성실한 교회친구 나영이, 경남에서 열심히 응원한 마음 따뜻한 진희, 나를 믿어주던 착한 선경이, 친구들과 사이에서는 바쁜 친구로 통해 그동안 얼굴보기도 힘들었던 고등학교 친구 주현, 수진이, 대학친구인 수진이, 보미, 경미, 정희, 효진, 혜은, 형경언니 모두와 이 기쁨을 나누고 싶습니다. 곧 만나러 가겠습니다.

논문을 잘 마무리 지을수 있도록 기도해주신 우리 교회 유치부 부장선생님이신 김남이 집사님과 Technical Editor인 수연이에게 감사드립니다. 또한, 바쁜 저 대신 우리반 맡아주시며 수고해주신 우리 유치부 모든 선생님께 감사드립니다. 앞으로 제가 더 열심히 하겠습니다.

마지막으로 작은 결실이지만 누구보다도 기뻐하시는 지금까지 키워주신 부모님과 가족이라서 그 무엇보다도 든든한 후원자인 오빠와 동생에게도 감사의 마음 전합니다.

이 력 서

성 명: 김 은 희

생년월일: 1976년 12월 12일

출 생 지: 경북 포항시

주 소: 경북 포항시 남구 지곡동 그린빌라 344동 402호

학 력

1995.3 – 1999.2 대구대학교 정보통신공학부 전자공학(B.S.)

2005.3 – 2007.2 포항공과대학교 정보통신대학원 정보통신학과(M.S.)

연 구 활 동

◆ Projects

- 통합 전력선 통신망 관리시스템을 위한 Manager의 구현, 한국전력 연구원 프로젝트, 2006