

Subject-Oriented Programming 을 이용한 SNMPv3 Architecture

설계에 관한 연구

김대수, 김태일

한국전자통신연구원 라우터기술연구부 라우터구조팀

A Study on SNMPv3 Architecture Design with Subject-Oriented Programming

Dae Su Kim, Tae Il Kim

ETRI

요 약

RFC 2271 에는 망 관리 플랫폼인 SNMP architecture 를 더 이상의 수정을 가하지 않는 모듈구조로 정의되어 있다. 즉, SNMP application 을 개발함에 있어 새로운 application 을 추가할 때, 기존의 software 에 수정을 가하지 않는 것이 중요한 요소이다. 이 논문에서 제안하는 subject-oriented programming 은 위에 언급한 관점에 주안점을 둔 방법으로 기존에 개발된 SNMP application 의 기본 구조를 유지한 채 software 를 개발할 수 있는 방법이다. Subject-oriented programming 은 object-oriented programming 에 기초를 두고있다.

1. 서론

RFC 2271 에는 SNMPv3 architecture 를 모듈화 된 형식으로 기술하고 있다. 프로그램을 개발함에 있어 모듈화 된 형식을 이용하는 이점은 어떤 특정한 태스크를 수행하는 모듈들의 집합으로 하나의 프로그램을 구성할 수가 있다. Object-oriented programming 에 있어서 모듈은 클래스가 될 수 있다. 하지만, 객체지향 프로그래밍은 사전에 잘 정의된 구조를 가지고 있기 때문에 기존의 프로그램을 upgrade 하는데 있어서 어려움이 존재한다 [1]. Subject-oriented programming 은 전체적인 하나의 시스템이 콤포지션(composition)를에 의한 subject 들의 구성으로 구성된다. 그리고 subject-oriented programming 은

object-oriented programming 에 기초를 두고 있으므로 object-oriented programming 의 중요한 방법인 클래스와 데이터 은닉, 상속성등을 사용하여 프로그램을 개발한다[2]. 본 논문의 구성은 다음과 같다. 2 절에서는 SNMPv3 architecture 에 대해서 살펴보고, 3 절에서는 subject-oriented programming 에 대해서 살펴본다. 그리고, 4 절에서는 subject-oriented programming 을 이용한 SNMPv3 architecture 구현 방법에 대해서 살펴본다. 끝으로 5 절에서는 본논문의 결론을 맺는다.

2. SNMPv3 Architecture

2.1 SNMP Entity

SNMP entity 는 application 과 SNMP engine 으로 구성되어

있다. 그림 1.에 SNMP entity 가 도시 되어 있다. SNMP application 과 engine 은 각각 모듈로 구성되어 있다[3]. 각각의 역할은 다음의 부 절에 설명된다. 그림 1.에서 보듯이 SNMP entity 는 모듈화 된 형식으로 구성되어 있다.

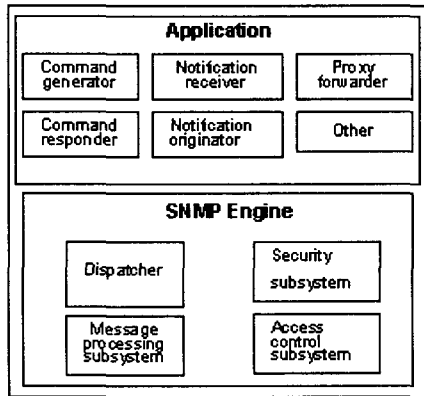


그림 1. SNMP Entity

2.2 SNMP application 과 engine 의 역할

SNMPv3 architecture 는 application 과 SNMP engine 으로 이루어지는데, application 은 관리되어 지는 정보를 제어 하고, SNMP engine 은 SNMP entity 외부와의 보안을 담당하는 인터페이스 역할을 한다. 개별적인 역할은 다음과 같다. Application 모듈을 살펴보면 다음과 같다. Command generator 는 SNMP Get, GetNext, GetBulk, Set request PDU 를 생성하여 정보 요청 기능을 수행하고, command receiver 는 command generator 에 의해 생성된 메시지를 받아 이에 대한 처리를 한다. Notification originator 는 trap 과 inform PDU 를 생성하고, notification receiver 는 이를 받아 그 정보에 해당하는 일을 수행한다. Proxy forwarder 는 SNMP message 를 forwarding 하는 모듈로, 구성형태에 따른 option 모듈이다. 다음을 SNMP engine 모듈로 Dispatcher 는 다양한 version 의 SNMP message 를 지원하는데, 위에서 언급한 application 과 다른 SNMP engine 모듈과 상호 작용하여 SNMP message 를 외부와 주고 받는다. Message processing subsystem 은 message 를 외부로 보내기 위한 준비 단계

로서, 수신된 메시지의 디코딩 역할을 담당하는데, message processing 은 메시지의 SNMP version 과 보안에 관련이 있다. Access control subsystem 은 교환되는 정보의 접근권한을 제어한다. 그림 2.에 application 과 SNMP engine 으로 구성된 SNMPv3 manager 가 도시 되어 있다.

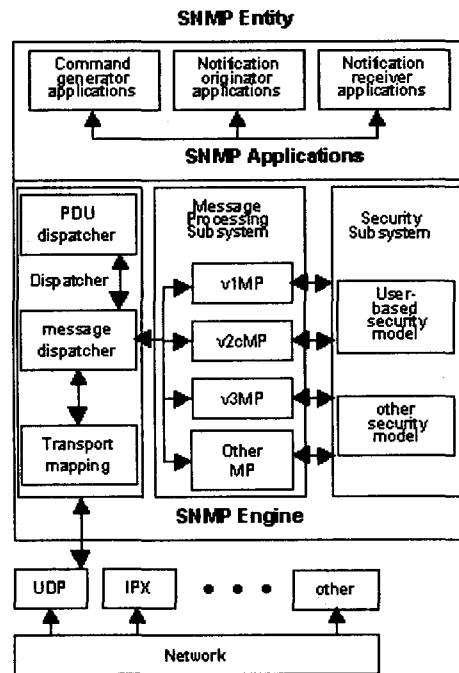


그림 2. SNMPv3 망관리 Manager

3. Subject-Oriented Programming

3.1 Overview of Subject-Oriented Programming

Subject-oriented programming 은 객체지향 프로그래밍에 기반을 둔 프로그래밍 방법으로서 객체지향 프로그래밍에 중요한 요소인 데이터 은닉, 다형성, 상속성등을 사용하여 프로그래밍을 한다[1,]. Subject-oriented programming 에서 subject 는 상대와 방법 규정의 집합체이다. 그림 3 에서 tree 라는 한 객체에 대한 여러 subject 를 보여준다. 즉, 한 tree 라는 application 을 여러 subject 로 나누어 구현한 다음 그 subject 를 콤포지션 규칙을 이용하여 하나의 application 으로 구현하는 방법이다. 그림 3.은 순서에 따라 sales, bird, woodsman, tree 라는 subject 로 명명되어질 수 있다. 콤포지션 규칙은 프로그

램이 구성 인자인 클래스와 함수들간의 상호 연관성을 규정하는 것으로 상호 연관성을 바탕으로 새로운 하나의 조합된 프로그램을 이끌어 내는 방법이다.[4, 5]

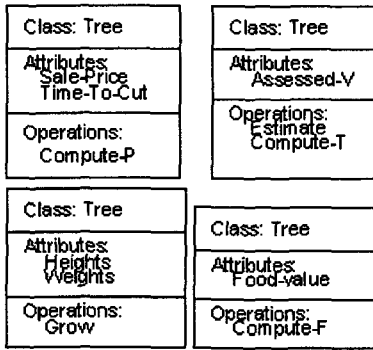


그림 3. tree 와 관련된 subject

대표적인 콤포지션 규칙에는 merge 가 있다. Merger 규칙은 수행되는 함수가 각각의 subject의 특성에 맞게 기능을 수행하게 하는 규칙으로 수행결과에 의해 subject들 간에는 상호 데이터의 변화를 가져와 각 subject를 하나의 집합체로 묶는 역할을 한다. Subject-oriented programming은 subject model로 나타낼 수 있다[1]. Subject model에는 구성요소로 subject, composition,

subject-oriented universe, subject

activation 있고 각각의 표현은 기술된 순서대로 다음과 같다.

$S = (N, I, D, P),$

N -> 클래스 이름, I -> 인터페이스, D -> 함수, P -> 기초 클래스 함수

$C = (R, Q)$

R -> 조합 규칙, Q -> 구성요소(subject)

$U = (M, A)$

M -> subject의 조합,

A = (OID, SA) -> universe activation으로 객체 식별자(oid)와 subject activation인 (SA)로 구성

$A = (S, T, C)$

S -> subject, T -> 상태 함수, C -> 함수의 인스턴스

4. Subject-oriented programming을 이용한 제안모델

그림 2.에 보인 SNMPv3 메니저를 subject-oriented programming을 이용하여 구현하기 위하여 각각의 모듈을 subject로 가정하고 subject model에 따라 제안한 모델은 그림 4.와 같다. 각각의 subject는 composition 필드에 merge 조합 규칙에 따라 구성하고, universe activation에는 그에 대응하는 subject activation을 두어 실제 기능을 수행하게 된다.

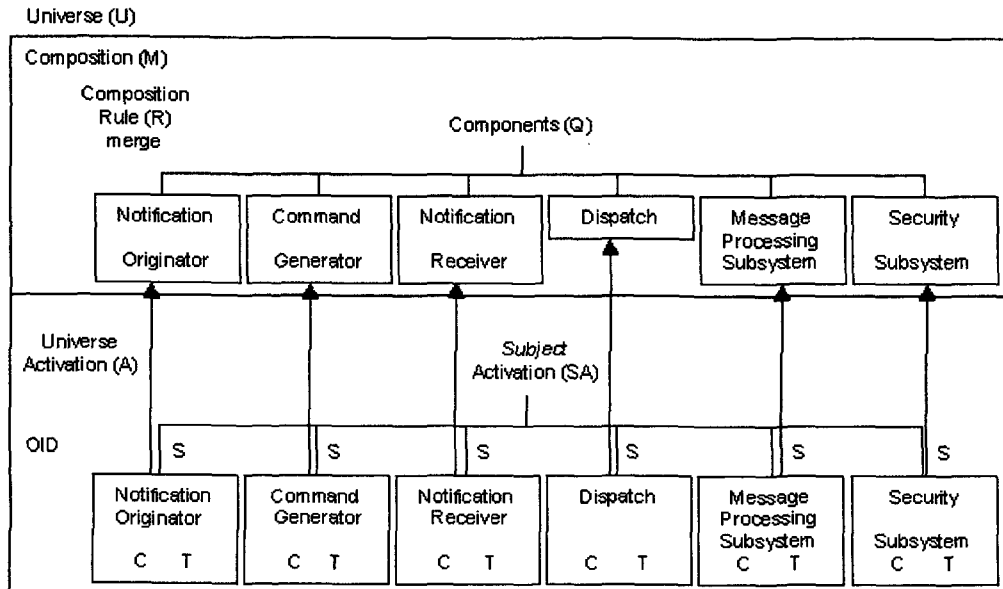


그림 4. 제안된 모델

5. 결론

본 논문에서는 subject-oriented programming 을 이용한 SNMPv3 architecture 설계 방법에 관하여 제안을 하였다. 망 관리는 네트워크의 효율적, 안정적 이용측면에서 중요한 부분을 차지한다. 즉, 네트워크 관리 차원에서 볼 때 새로운 기능 추가라는 면 또한 중요하다고 생각하여 subject-oriented programming 방법을 제안하였고 향후 과제로 제안한 방법으로 실제적으로 구현하는 것이다.

참고문헌

1. Harold Ossher, William Harrison, Frank Budinsky and Ian simmonds, "Subject-Oriented Programming: Supporting Decentralized Development of Objects", IBM Thomas J. Watson Reaserch Center.
2. William Harrison and Harold Ossher, "Subject-Oriented Programming (A Critique of Pure Objects)", OOPSLA'93, pp. 411-428.
3. William Stallings, "SNMP, SNMPv2, SNMPv3 and RMON 1 and 2." ADDISON-WESLEY.
4. ROB POOLEY and PERDITA STEVENS, "Using UML SOFTWARE ENGINEERING WITH OBJECTS AND COMPONENTS." ADDISON-WESLEY.
5. Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, "DESIGN PATTERNS: Elements of Reusable Object-Oriented Software." ADDISON-WESLEY.