

# 실시간 자동제어 시스템을 위한 하이퍼미디어 방식의 사용자 대화 시스템 개발

김도현, 이기준

부산대학교 전자계산학과 정보공학연구소

## A Development of HyperMonitor: Hypermedia for User-Interface in Real-Time Control System

Do-Hyun Kim, Ki-Joune Li

Dept. of Computer Science, Pusan National Univ.

Email: {dhkim, lik}@spatios.cs.pusan.ac.kr

### 요 약

본 논문에서는 최근 들어 많은 응용분야에 연구되고 있는 하이퍼미디어 방식의 사용자 대화 방식을 이용하여 실시간 제어 시스템의 사용자 대화를 개발한다. 실시간 시스템의 동적인 측면을 반영하기 위한 자료의 일관성, 자료 갱신 방법 등의 개념을 사용하여 정적인 방법위주의 하이퍼미디어의 개념, 기능 등을 수정, 보완한다. 그리고 객체 지향형 설계와 구현을 통해 여러 실시간 응용분야에 적용할 수 있도록 실시간 자료와 노드, 링크를 객체화하여 Class 틀을 개발하는 것을 목적으로 한다.

### 1. 서론

최근의 기술 산업용 시스템의 품질의 중요요소로 보기 좋고 사용하기 편리한 사용자 대화 방식이 대두되고 있다. 이는 복잡한 현장지식을 지닌 전문가의 부족 등에 대한 전문가를 대체할만한 수준의 효과적인 기능을 제공하는 사용자대화를 요구하는 최근의 시장동향에서도 알 수 있다. 또한 사용자 대화 방식의 개발비용 증가로 개발 생산성 및 유지보수의 향상이 절실하게 요구되어 개발하기 쉽고, 수정하기 쉬운 사용자 대화 방식이 요구된다. [7]

본 논문에서는 객체 지향형 하이퍼미디어 방식의 사용자 대화 시스템인 하이퍼모니터(HyperMonitor) 개발을 연구목적으로 설정한다. 2장에서는 실시간 시스템에서 사용자 대화를 구현할 때의 요구사항을 사용자 대화, 실시간 조건의 입장에서 서술한다. 3장에서는 하이퍼모니터의 정보 모델링에 대하여 서술한다. 하이퍼모니터 시스템 모델링, 노드와 링크 모델링, 그리고 실시간 시스템의 객체 모델링이 개별적으로 설명된다. 4장은 하이퍼모니터의 설계와 구현에 대하여 설명한다. 이 장에서는 하이퍼모니터 관리자, 하이퍼모니터 객체 관리 모듈, 그리고 실시간 자료 관리 모듈로 나뉘어 서술한다. 5장에서는 하이퍼모니터 시스템의 각 모듈을 Class Library화하는 작업에 대해 서술한다. 6장의 결론에서는 현재 하이퍼모니터의 구현현황, 구현시 문제점, 그리고 앞으로 구현해야 할 중요 기능을 제시한다.

## 2. 실시간 제어 시스템의 요구사항

### 2.1 사용자 대화 분석

실시간 제어 시스템은 기능이 복잡하여 기능을 이용하는 사용자와 시스템간의 대화가 시스템의 절대적인 설계조건이 된다. 따라서 실시간 시스템의 사용자 인터페이스가 가져야 할 조건에

대해서 알아본다. [6, 7, 8]

- 1) 간단, 명료해야 한다.
- 2) 사용자 친근성이 있어야 한다.
- 3) 경고 기능이 있어야 한다.
- 4) 빠른 실시간 자료 갱신 능력이 있어야 한다.
- 5) 확장성이 있어야 한다.
- 6) 사용자 대화 개발의 편리성이 있어야 한다.

이상의 조건 중에서 빠른 실시간 자료 갱신 능력은 가장 중요한 기능이라 할 수 있다. 이는 실시간 데이터베이스와 하이퍼모니터간의 자료 일관성을 지켜준다.

### 2.2 실시간 조건의 분석

실시간 시스템과 하이퍼모니터와의 실시간 조건은 실시간 자료의 시간적 일관성이다. 즉 현재 실시간 데이터베이스에 있는 실시간 자료의 값과 화면에 나타난 노드의 실시간 값이 일치해야 한다는 것이다.

하이퍼모니터는 실시간 데이터베이스의 자료를 노드로 표현하여 사용자에게 하이퍼링크(HyperLink)를 제공한다. [1] 그러나, 실시간 데이터베이스의 자료는 매우 동적으로 갱신되어 실시간 자료의 갱신이 하이퍼모니터에 잘 반영되지 않으면 실제 데이터베이스에 있는 자료와 하이퍼모니터의 노드에 있는 자료간에 불일치가 생길 수 있다. 예를 들면, 보일러의 온도가 이미 위험수위를 넘어 경보가 발생했는데도 화면에 나타난 보일러의 온도가 정상이라면 사용자는 정보 발생의 원인을 제대로 분석하지 못하게 된다. 그러나 실시간 자료의 갱신을 모두 하이퍼모니터에 반영하는 것도 시스템에 많은 부하를 가져와 시스템이 빠른 시간 내에 제대로 작동하지 않을 수 있다. 따라서 빠른 시간 내에 필요한 자료 일관성 기능을 제공하는 시스템을 구현해야 한다.

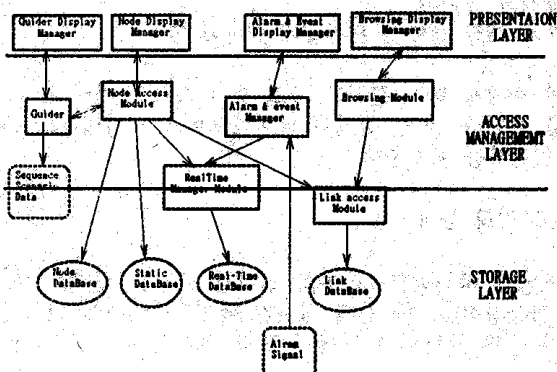
하이퍼모니터 정보 모델링

1 하이퍼모니터 시스템 모델링

하이퍼모니터는 크게 3부분으로 나뉜다. 노드와 관련 정보를 화면에 나타내는 표현 계층(Presentation Layer), 각 데이터베이스에 효율적인 자료 접근 방법을 관리하는 노드-링크 액세스 계층(Access Management Layer), 각 데이터베이스를 효율적으로 관리하는 노드-링크 저장 계층(Storage Layer)으로 나눌 수 있다. [2] 노드-링크 저장 계층은 달리 노드-링크 네트워크 계층으로 언급할 수 있다. 이는 하이퍼미디어(Hypermedia)를 노드와 링크의 네트워크로 정의할 수도 있기 때문이다. <그림 1>은 하이퍼모니터 시스템을 세부적으로 나누어 각 계층의 역할을 세부 수행단위로 모델링한 것으로 Rumbaugh의 객체 표기방법을 사용하였다. [3]

각 계층은 독립적이면서도 서로 상호동작이 가능해야 한다. 따라서, 각 계층간의 정보를 공유할 수 있는 인터페이스가 필요하다.

표현계층과 노드-링크 액세스 계층간에는 각 자료를 실제 화면에서 어떻게 나타낼 것인가에 대한 방법이 필요하다. 즉, 화면에 자료를 표현하는 데 필요한 정보를 특정한 형태의 자료 구조로 규정하는 데, 이를 표현 추상화 구조(Presentation Specification)라 한다. [4] 이 형태는 노드 모델링(Modeling)시에 언급하도록 한다. 표현 추상화 구조 외에 노드-링크 액세스 계층과 노드-링크 저장 계층사이의 인터페이스도 필요한데, 이는 각 단위 노드의 식별자를 이용하여 두 계층간의 정보를 나눈다. 노드-링크 저장 계층은 독립적인 노드-링크 내부자료 계층(Within-Component Layer)을 가지고 있다. [4] 이는 실제 노드나 링크의 자료들이 저장되어 있는 형태에 관한 정보를 다루는 계층이다. 하이퍼모니터는 문자뿐 아니라 그래픽 자료, 이미지 자료, 음성 자료 등의 멀티미디어를 지원하므로, 노드-링크 내부 자료 계층은 매우 중요한 역할을 수행하는 계층이다.



<그림 1> 하이퍼모니터 시스템 내부 구조 및 계층

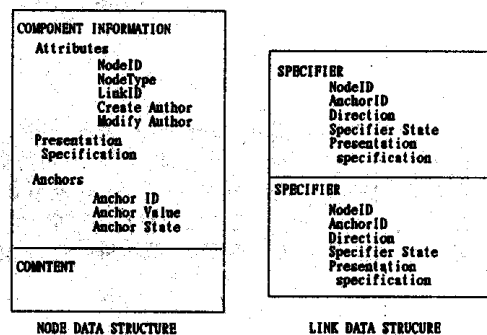
3.2 노드와 링크 모델링

하이퍼모니터의 가장 중요한 두개의 자료는 노드와 링크이다. 아래 <그림 2>에 노드와 링크의 구조를 나타내었다. 노드는 크게 노드 정보(Component Information)와 노드 내용(Content)으로 나뉜다. 노드 정보는 속성(Attribute), 표현 추상화 구조, Anchors로 구성된다. 속성은 각 노드에 대한 정보-노드의 이름, 노드를 생성한 사람 등-를 표현한다. 표현 추상화 구조는 노드가 화면에 표현될 때 사용되는 정보-X, Y 좌표, 텍스트, 그래픽 정보-등이다. Anchor는 다른 노드로의 네비게이션(Navigation)이나 스펠(Span)을 나타내기 위한 객체이다.

노드는 이런 Anchor의 집합으로 이루어져 있다. 링크는 Specifier의 집합으로 표현되는 데, Specifier는 <그림 2>에 주어진 정보를 가지고 있다. Direction은 From, To가 있다. From은 Specifier에 지정된 노드 식별자가 근원지 노드이고 현재 활성화 된 노드가 목적지 노드이다. To는 그 반대가 된다. Direction은 브라우저(Browser)를 구현할 때 효과적으로 사용된다 [5].

3.3 실시간 시스템의 객체 모델링

실시간 시스템은 빠른 자료 갱신을 목적으로 하기 때문에 실시간 자료를 어떻게 효율적으로 구조화하여, 이 자료를 하이퍼모니터에 실 시간적으로 얼마나 빨리 반영시키느냐고 중요한 문제

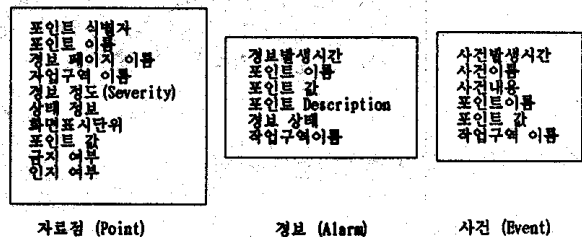


<그림 2> 노드와 링크의 자료구조

가 된다. 이는 실시간 데이터베이스와 하이퍼모니터와의 자료 일관성 유지를 위해선 필수적으로 갖추어져야 할 요건이다.

실시간 시스템에서 사용되는 중요한 자료로는 경보(Alarm), 사건(Event), 자료점(Point)을 들 수 있다. 자료점은 실시간 시스템에서 가장 기본이 되는 객체로 실시간 현장에서 센서(Sensor)를 통하여 하이퍼모니터에 전달되는 자료이다. 자료점의 모든 정보는 디스크에 저장되나 수시로 변경되는 정보는 메모리에서 똑같이 관리된다. 이는 앞에서 언급된 실시간 시스템과 하이퍼모니터의 자료 일관성 문제를 해결하기 위한 방법이다.

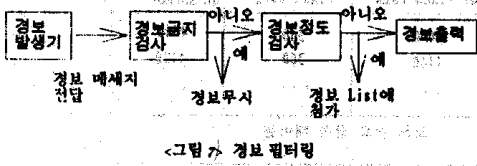
사건은 각 자료점의 자료 값의 변화뿐 아니라 시스템을 운용하면서 일어나는 모든 현상을 말한다. 경보는 사건의 일종이나 사건과는 독립적으로 하이퍼모니터에 메시지가 전달된다. 이는 실시간 시스템에서는 위험 상황의 탐지 능력이 매우 중요하므로, 신속한 경보인지, 경보 발생을 하이퍼모니터에 전달하기 위함이다. <그림 3>는 앞서 언급된 세 가지 모델을 나타낸 것이다.



<그림 3> 실시간 자료 모델링



정보 List에 삽입한다.

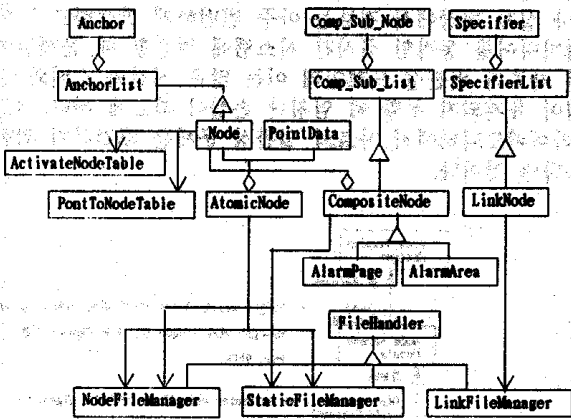


<그림 8> 경보 필터링

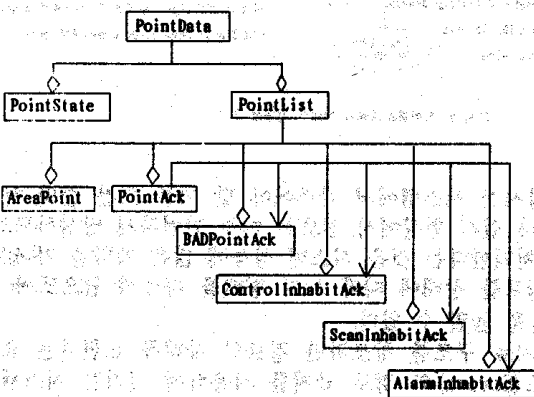
### 5. 하이퍼모니터 CLASS LIBRARY 설계와 구현

사용자 대화 비용의 증가와 유지, 보수에의 편리성을 위하여 객체 지향형 접근 방식을 사용한다. 이를 위해 Class Library를 만드는 것이 중요하다. 이는 하이퍼모니터가 실시간 시스템의 각 응용 분야에 적용되기 위한 필수조건이라고 할 수 있다. 또한 이 Library간의 내부 자료 인터페이스도 효과적으로 설계되어야 한다.

<그림 8>는 Rambought의 객체 지향 설계 표기법을 사용하여 노드와 링크 관련 Class Design을 구현한 것이다. <그림 9>는 포인트 관련 Class Design을 구현한 것이다. 각 Class의 데이터를 보호하기 위하여 다른 Class에서는 직접 접근을 하지 못하고 오직 함수를 사용하여 접근을 가능하게 하였다. 즉 각 Class간의 인터페이스는 Class내의 함수를 통하여 구현하였다.



<그림 8> 하이퍼모니터 Class Design



<그림 9> 포인트 Class Design

현재 실시간 시스템의 사용자 대화 방식은 테이블, 메뉴 방식의 사용자 대화 방식에서 점차 현장 기술자의 인지 능력을 향상시켜 주는 사용자 대화 방식으로 바뀌어 지고 있다. 본 논문에서는 현재 활발히 연구되고 있는 하이퍼미디어를 실시간 제어 시스템에 적용한 하이퍼모니터를 설계, 구현하였다. 그리고, 하이퍼모니터의 각 자료를 객체로 모델링 하여 객체 지향형 자료 구조로 정의하였다. 실시간 제어 시스템의 시스템 응답 시간을 줄이기 위해 자주 사용되는 자료점의 속성을 메모리에 적재해 두어 빠른 응답 시간을 유지하게 설계하였다. 또한 실시간 시스템의 자료의 일관성을 위하여 효율적인 필터링 과정을 COS 필터(Filter)에 적재하여 빠르고 정확한 자료의 재변경을 수행할 수 있으며 경보와 사건의 필터링 작업도 시스템의 성능을 향상시킨다.

향후 과제로서는 실시간 환경의 여러 응용 분야에 적용하기 위해서 하이퍼모니터의 각 모듈을 단위 모듈화 하여 인터페이스 저작 도구(Authoring Tool)와 함께 하이퍼모니터 패키지의 형태로 개발할 계획이다. 그리고 하이퍼모니터의 기능을 확대하여 실제 실시간 제어 시스템에 적용될 수 있는 MMI(Man Machine Interface)를 구현할 것이다

### 7. 참고 문헌

- [1] Conklin, J., "Hypertext: An Introduction and Survey", IEEE Computer, Vol.2, No.9, 17-41, September 1987.
- [2] Halasz, F., and Schwartz, M., "The Dexter Hypertext reference Model", Proceeding of Hypertext Standardization Workshop, January 1990, NIST.
- [3] J.Rumbaugh, M. Blaha, W. Premeriani, F. Eddy, W. Lorensen, "Object-Oriented Modeling and Design", Prentice-Hall Inc, 1991.
- [4] Kaj Gronbaek, and Randall H. Trigg, "Design Issues for a Dexter-Based Hypermedia System", Communication of the ACM, Vol.37, No.2, 40-49, Febra 1994.
- [5] Jakob, N., "The art of navigating through Hypertext", Communication of the ACM, Vol.33, NO.3, March 1990.
- [6] 이기준, "실시간 제어시스템을 위한 하이퍼미디어시스템의 기능적 요구사항들", HCI'94 학술대회, p178-189.
- [7] M. Tendiaouri, C. Kolski, and P. Millot, "An Approach towards the Design of Intelligent Man-Machine Interfaces used in process Control", Journal of Industrial Ergonomics, Vol.14 No.8, 1991, pp.345-361
- [8] P.Allen, and A.Burns, "ADDS-A User Interface Management Scheme for Real-Time System", Proc. 2nd Conf. S/W Engineering for Real-Time Systems, Cirencester Uk, September 1989, pp.111-115

### 6. 결론 및 향후 과제