

## 專門家 스케줄링 프로그램의 開發에 關한 研究

### -A Study on the Development of Expert Scheduling Program-

申 容 販\*  
 朴 魯 國\*\*  
 宋 文 益\*\*\*

#### ABSTRACT

Today, manpower is used as a major factor in the production of service industry that is on the increasing trend. So the research on manpower structure have been regarded as a important field.

The research on personnel scheduling, however, have got de-emphasized as compared with the field of manufacturing scheduling. As the increasing trend of service industry, an appropriate personnel scheduling for each type of service industry is needed urgently.

This paper deals with nurse scheduling problem in the field of personnel scheduling. We aim at developing prototype of the expert nurse scheduling program which is easy to use when scheduling without help of expert, and satisfies the various request of hospital scheduling policy and nurse scheduling.

#### 1. 序 論

人間만이 할 수 있는 知的 行爲(언어구사, 학습, 추론, 문제해결)를 분석하여 컴퓨터로 하여금 이들 지적행위를 할 수 있도록 하는 人工知能(Artificial Intelligence : AI)에 관한 연구는 1956년 美國 다트머스大學의 John McCarthy가 처음 사용한 이래 최근들어 컴퓨터 하드웨어의 급속한 발전에 따라 5세대 컴퓨터 言語를 채택하여 급속히 실용화 되고 있다. 人工知能에 대한 여러가지의 定義가 있으나, Rich는 "인공지능이란 인간의 사고 및 行動遂行 방식을 機械가 그대로 학습할 수 있는 기능"이라 정의하였다. 이러한 인공지능에 대한 研究가 활성화됨에 따라 기계로 하여금 인간이 문제를 분석하고 필요한 意思決定을 내리는데 도움을 줄수 있는 專門家 시스템(Expert System)이 발전되었다[1].

#### 1.1 專門家 시스템의 定義

전문가 시스템(Expert System)은 "特定한 분야의 專門家가 갖고 있는 전문지식이나 Knowhow를 컴퓨터에 事實(fact) 및 規則(rule)형태로 저장하고 어떤 상황에 대한 정보가 들어오면 축적한 지식을 바탕으로 추론과정을 거쳐 전문가의 도움없이 문제를 해결하는 機能을 가진 인공지능 프로그램"이라 정의할 수 있다[3]. 人間の 지적활동의 원천인 知識을 이용하는 전문가 시스템은 지식획득(knowledge acquisition), 지식베이스(knowledge base), 推論(reasoning), 自然言語(natural language) 등의 각 전문영역의 基礎技術을 포함하고 있다. 인간 전문가와 전문가 시스템을 다음 <표 1.1>와 같이 比較할 수 있다.

표 1.1 인간전문가와 전문가시스템의 비교

인간 전문가	전문가 시스템
창조적인 지식을 사용	구조화된 지식을 사용
상식적 지식을 사용	기술적 지식을 사용
환경 적응력이 뛰어나	환경 적응력이 떨어짐
지식의 적용범위가 넓음	지식의 적용범위가 좁음
편견이 작용할 수 있음	객관적으로 업무를 수행

\* 仁荷大學校 大學院 産業工學科

\*\* 尙志大學校 産業工學科 專任講師

\*\*\* 仁荷大學校 産業工學科 敎授

접수 : 1992. 10. 5.

확정 : 1992. 10. 23.

```

type MNR ? :
type DNR ? :
nurse name ? :      (type and ↕)
nurse class ? :    (type and ↕)
special off day ? : (type and ↕)
weekend preference ? : (type and ↕)
working shift ? :  (type and ↕)
    
```

```

nurse name ? : KIMGD (type and ↕)
nurse class ? : LPN (type and ↕)
special off day ? : 2 (type and ↕)
weekend preference ? : 1 (type and ↕)
working shift ? : DE (type and ↕)
    
```

```

nurse name ? : LEESH (type and ↕)
nurse class ? : LPN (type and ↕)
special off day ? : 3 (type and ↕)
weekend preference ? : 1 (type and ↕)
working shift ? : E (type and ↕)
    
```

```

nurse name ? : LEEYJ (type and ↕)
nurse class ? : LPN (type and ↕)
special off day ? : 4 (type and ↕)
weekend preference ? : 2 (type and ↕)
working shift ? : DEN (type and ↕)
    
```

그림 3.1 데이터 입력의 예  
위의 데이터에 따른 schedule 결과는 다음과 같다.

name	class	sun							sun						
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
SINYJ	LPN	D	D	D	D	0	D	0	0	0	D	D	D	D	
KIMGD	LPN	0	0	D	D	D	D	D	D	0	E	E	E	0	
LEESH	LPN	0	E	0	E	E	E	E	E	E	0	E	E	0	
LEEYJ	LPN	D	D	D	0	E	E	0	N	N	N	0	D	D	

그림 3.2 스케줄결과의 출력 예

위의 결과를 보면 현재의 프로그램은 MNR과 DNR은 휴일 Off를 제외하고는 전부 滿足 하나 3일의 경우 전부 D-shift에 看護員이 配定되어 있다.

따라서 이러한 경우의 문제를 해결하기 위해서는 目標指向의 후향추론이 필요하다.

#### 4. 結 論

本 研究는 전문가 시스템이 일반적으로 사용되는 分野-診斷시스템, 解析시스템, 選擇시스템-가 아닌 計劃分野에 적용하여 보았다.

本 研究의 전문가 시스템은 대화체를 사용하여 스케줄을 작성한다.

전문가 시스템의 도입으로 인한 效果는 다음과 같이 豫測할 수 있다.

- ① 재스케줄을 作成하기 容易하고 빠른 시간내에 스케줄을 作成할 수 있다.
- ② 스케줄러의 失手나 편향을 막을 수 있어 스케줄의 質을 높일 수 있다.
- ③ 制限 條件이 늘어나고, 간호원의 수가 많은 경우에도 빠르게 스케줄을 作成할 수 있다.
- ④ Dialogue box의 대안 제시 기능과 설명 기능으로 인하여 각 간호원의 스케줄에 대한 만족감을 높일 수 있다.
- ⑤ 스케줄을 作成하는 스케줄러의 費用을 줄일 수 있다.

또한, 本 專門家 시스템은 스케줄링에 관한 知識을 객체 지향적으로 모델링 하였으므로 객체지향적인 프로그래밍 技法과의 접목시 소프트웨어의 生産性이 높으리라 기대 된다.

1.2 專門家 시스템의 長·短點

전문가 시스템의 장점은 특정 업무에 정통한 전문가의 知識과 Knowhow를 構造化하여 관리·계승 (inheritance)할 수 있을 뿐만 아니라, 일반인도 專門家와 같은 수준으로 問題를 해결하며, 인간 전문가와 달리 항상 客觀的이고 精確한 판단을 내려 업무의 能率을 극대화 시킬 수 있다는 장점이 있는 반면에 創造性, 종합분석능력, 유연성에 있어서는 인간 전문가에 비하여 떨어지는 단점이 있다. 이러한 短點에도 불구하고 精確적 구조식을 推定하는 DENDRAL과 전염병의 진단과 처방을 자문하여 주는 의료진단 시스템 MYSIN과 광물탐색지연 시스템 PROSPECTOR 등은 상업적으로도 성공한 전문가 시스템이다. 또한 1980年代에 들어서 產業分野에서 응용 시스템이 출현하기 시작하였는데 현재도 각광을 받고 있는 DEC사의 컴퓨터 機器 구성결정 지원 시스템인 XCON이 전문가 시스템의 開發 열기를 주도하였고, 이후 VLSI 등 하드웨어 技術의 급속한 발전으로 전문가 시스템은 實用化 시스템을 중심으로 많은 研究가 수행되고 있다.

1.3 專門家 시스템의 構成

전문가 시스템은 인간의 지식을 이용하는 시스템이므로 시스템 구성 자체가 지식을 저장하고 관리하기에 편리하도록 설계되어 있다[6]. 인간 전문가의 머리속에 들어있는 전문지식을 사실(fact)과 규칙(rule)의 형태로 저장한 지식베이스(knowledge base)와 그 지식베이스의 내용을 토대로 문제해결에 필요한 지식을 담은 추론기관(inference engine), 사용자에게 이용의 편의를 제공하기 위한 사용자 인터페이스(user interface) 및 추론 과정을 설명하는 설명상자(dialogue box)로 구성되어 있다.

(1) 知識베이스(Knowledge-Base)

지식베이스는 問題解決에 관련된 지식이나 Know-how가 컴퓨터에서 사용 할 수 있는 형태로 構造化되어 축적된 것으로 저장되는 지식의 種類는 대상에 대한 지식, 사건에 대한 지식, 행위에 대한 지식과 問題解決 방법이나 추론제어를 담당하는 지식이 있다.

(2) 作業메모리(Working Memory)

작업메모리는 문제해결 과정에서 발생하는 狀態를 저장 가능하도록 現在의 狀態를 반영하는 사실로 구성되어 있다.

(3) 推論기관(Inference Engine)

추론기관은 주어진 문제해결을 위해 추론 과정을 거쳐 結論에 도달하도록 하는 부분으로 지식 베이스의 지식을 이용하여 지식을 재결합, 새로운 지식을 유도하고 또 다시 어떤 결론에 도달할 때까지 지식을 추론하는 機能을 담당한다.

(4) 使用者 인터페이스(User Interface)

전문가의 지식과 Know-how를 入力하기 좋도록 하고, 사용자가 전문가 시스템에서 설명을 받기 쉽도록 질문·응답 형식이나 명령어 등을 갖춘 使用者 對話 프로그램이다.

(5) 說明箱子(Dialogue Box)

전문가 시스템이 문제해결의 어떤 결론에 도달하였을 때, 이에 대한 이유를 說明하는 시스템으로 기존의 소프트웨어와 구분되는 기능의 하나이다. 즉, 어떤 規則이나 事實을 사용해서 어떻게 추론하였는지 解를 구하는 過程을 사용자에게 설명하는 기능을 담당한다.

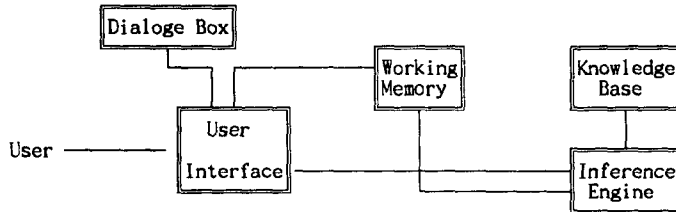


그림 1.1 전문가 시스템의 구성

표 1.2 기존의 프로그램과 전문가 시스템의 比較

	기존의 프로그램	전문가시스템
처리하는 데이터의 도출과정의 제시형태	데이터의 표현과 사용 알고리즘과정 유일해 또는 완전해	지식의 표현과 처리 추론과정 적절한 해결책

1.4 專門家 시스템의 知識表現(Knowledge Representation)과 推論(Reasoning)

전문가 시스템에서 知識表現(knowledge representation)과 推論(reasoning)은 전문가의 지식과 Knowhow를 컴퓨터에 어떻게 표현하며, 이렇게 표현한 知識으로 새로운 결론을 어떻게 구할 수 있는가에 대한 問題이다. 전문가 시스템의 관점에서 볼 때 지식표현의 設計는 ① 지식표현의 적합성, ② 추론의 적합성, ③ 추론의 효율성, ④ 지식획득의 效率性이란 4가지 목표 및 원칙에 따라서 設計한다.

전문가 시스템에서의 지식표현 방법은 다음과 같다[8].

(1) 生成規則(Production Rule)에 의한 지식표현

지식표현 방식으로 가장 널리 사용되는 방법으로 "만일 <조건부>를 만족하면 <결론부>를 실행하라"라는 의미의 IF <조건부> THEN <결론부> 문장으로 지식을 표현 한다.

IF	single and do not itemize deduction and gross income ≤ \$50,000
THEN	can use 1040A or 1040ez tax forms

그림 1.2 Rule에 의한 지식표현의 例

(2) Frame

프레임(Frame)은 1975년에 M. Minsky가 認知科學的 관점에서 제안한 지식 표현으로 의미 네트워크(Semantic Network)에 절차적 지식을 부가함으로써 知識을 표현 하는데 보다 일반적이며 자연스럽게 강력한 수단을 제공하여 준다. 프레임은 각 객체에 대한 속성과 값에 관한 情報를 가지고 있는 슬롯의 集合으로 구성된다.

프레임에 의한 지식표현은 복잡한 개념의 構造를 자유로이 기술할 수 있으며, 속성값이 추론방법에 관한 절차적 지식을 포함하고 있으므로 User Interface에 의한 Dialogue Box를 쉽게 支援할 수 있다.

(3) 의미 네트워크(Semantic network)

지식을 표현하는데 연상을 증시한 모델로서 客體, 概念, 事件등을 나타내는 노드와 객체들 간의 관계를 규정하는 링크로 구성된다. 객체들 간의 관계를 나타내는 링크의 종류는 무수히 많지만, 가장 대표적이고 基本的인 링크는 친자(class/instance) 관계를 표시하는 IS-A 링크와 노드의 性質을 표시하는 HAS-A 링크, 전체와 부분과의 관계를 나타내는 PART-OF 링크, 集合의 要素數를 나타내는 NUMBER-OF 링크가 있다.

이와 같은 意味 네트워크에 의한 지식 표현의 長點은 다음과 같다.

- ① 어떤 사물에 연관된 情報를 끌어낼 수 있는 연상적인 추론이 가능하다.
- ② 하나의 노드가 다른 노드로 부터 性質을 이어받는 계승(inheritance)이 가능하다. 예를들어 "node A is a node B" 라는 문장이 成立한다면 노드 A는 노드 B의 性質을 모두 계승한다. 성질계승은 지식의 중복을 피하고 공유가 가능하므로 기억해야 할 情報의 양을 줄일 수 있어서 效果的이다.
- ③ 새로운 노드나 링크를 첨가하기가 쉬워 지식을 추가, 제거가 쉽다.
- ④ 일반적인 技術 이외에도 예외적인 기술을 할 수 있다.

(4) 論理(Logic)

지식을 선언적인 논리식의 형태로 표현하여 三段論法에 의한 演繹的 推論을 지원하는 지식표현 방법으로 논리는 표현방식이 단순하고 문장을 추가, 수정, 삭제하는데 쉬운 반면, 지식베이스를 구성하는데 組織化할 수 없고, 사실의 수가 많아질수록 추론작업이 어려워지는 短點도 있지만 현재 인공지능 분야에서 가장 널리 사용되는 PROLOG에서 주로 사용하는 知識表現 형태이다.

(5) 客體指向的 知識表現(Object-Oriented Knowledge Representation)

객체지향적 지식표현에서는 표현하는 지식을 屬性과 값에 의해 다른 객체들과 구분하여 표현하고, 또한 階層的으로 공통된 지식의 속성과 형태를 동일한 객체로 간주하며, 프레임의 지식표현과 같이 객체간의 關係(relation)를 클래스(class)와 인스턴스(instance)로 표현한다.

객체간의 關係는 상위 클래스(Super class)에서 下位 클래스(Sub class)의 각 인스턴스에 하위 클래스의 특성(attribute) 및 속성값(value)이 계승(inheritance)되므로 하위 클래스에서는 상위 클래스와 특성을 技術하여 표현할 수 있으므로 지식의 추가·변경이 보다 자유롭고 추론의 형태도 인스턴스와 추론에 관한 절차적 지식에 의해서 이루어진다.

객체지향적 지식표현 방법은 모듈러 設計를 지향하므로 지식 객체의 변수만 변경하면 프로그램이 자

동적으로 변화되므로 소프트웨어의 재사용을 가능하게 하는 실재적이며, 有用한 프로그래밍 기법으로 최근 專門家 시스템의 지식표현에 널리 사용되고 있다.

(6) 推論(Reasoning)

전문가 시스템에서의 추론기관(Inference Engine)은 표현된 지식을 탐색하여 주어진 상황에 대한 問題의 해를 분석·추론하고, 새로운 가설을 세워 知識을 탐색하는 기능을 담당하는 부분으로 일반적으로 初期狀態에서 출발하여 목표상태로 도달하는 방향으로 지식을 추론하는 데이터 구동형(data-driven)의 전향추론(forward chaning)과 目標狀態에서 출발하여 초기상태에 도달할 때까지의 가설을 證明하는 方式으로 추론하는 목표 지향(goal-driven)의 후향추론(backward chaning)이 사용된다.

전향추론은 구하고자 하는 解가 많거나 해를 生成해서 구축해 두고자 할 때 바람직한 추론으로 設計 시스템이나 計劃시스템과 같이 일정한 제약조건을 만족하면서 부여된 요구를 만족하는 최적 또는 타당한 해를 구하는 유형의 전문가 시스템에 알맞는 推論方式이다.

후향추론은 전문가가 어떠한 原因을 규명할 때 결과를 어느 정도 예측할 수 있는 경우에 바람직한 추론으로 診斷시스템, 解析시스템, 選擇시스템과 같이 결론의 후보가 유한하고, 그 중에서 부여된 관측데이터를 보다 잘 설명하는 假說을 선택해서 그것을 결론으로 내리는 유형의 문제에 적당한 추론 방식이다.

2. 客體指向 모델링(Object Oriented Modeling : OOM)

2.1 객체지향 모델링(Object Oriented Modeling : OOM)의 概要

객체지향 개념의 근원은 처음으로 클래스 概念을 도입한 SIMULAR(1966)라는 시뮬레이션언어(simulation language)에서 그 근원을 찾을 수 있다. 이후 最初의 Symbol 중심의 對話式 프로그래밍 환경을 갖춘 Smalltalk(1983)에서 그 개념을 확고히 하였고, 이외에도 ACTORS(1981)와 FLAVORS(1981)로부터 메시지 전달이나 다중상속(Multi inheritance) 概念등 많은 影響을 받았다. 또한 人工知能 分野에서도 그 개념의 뿌리를 찾을 수 있다. 프레임(frame)에서 대상(attribute)과 속성값(value)으로 知識을 표현한 것에서 객체지향 개념을 발견할 수 있다. 그러나 객체지향 개념은 프로그래밍언어, 인공지능(AI), 전문가 시스템(ES), 데이터베이스(DB) 등 여러 분야에 걸쳐 새로운 要求에 따라 발전된 개념이기 때문에 아직 하나의 統一된 개념으로는 자리잡고 있지는 못하나 인공지능 분야와 전문가 시스템 분야에서는 널리 사용하는 되고있다[2].

객체지향 概念을 바탕으로 한 객체지향 모델링은 실세계의 개념을 그대로 담고 있는 모델을 기반으로 주어진 問題를 분석하는 새로운 思考 方法이다. 이 기법은 사용자의 요구사항에 관한 정보를 추출하고 수집하는데 수반되는 기법으로 객체지향 모델을 기반으로 한다. 모델이란 의도한 무엇인가에 대한 추상화(Abstraction)로, 실제로 만들어 보기 전에 이해하고자 하는 目的을 갖는다. 이러한 모델의 개념은 복잡한 시스템을 만들때에 여러 開發者가 갖고 있는 설계안을 모델로 만들고 사용자의 요구에 가장 부합되는 모델을 선정하여 완전한 시스템을 이루고 난후 가장 適合한 시스템을 얻을 수 있는 것이다[7].

객체지향 지향 모델링(object oriented modeling)에 의한 시스템 設計는 다음과 같은 여러가지 優點을 期待할 수 있다.

첫째, 실제의 것을 만들기 전에 모델들을 테스트할 수 있다. 하나의 다리를 만들기 전에 다리 모형을 그 다리가 세워질 環境을 模倣하여 시험해보면서 각 모델의 適合성을 확인할 수 있다.

둘째, 使用者와 대화의 수단으로 모델을 사용할 수 있다는 것이다. 대부분의 사용자가 프로그램에 관한 문의안일 수 있다. 따라서 그들은 그들의 要求事項을 개발자에게 전달하지 못하게 된다. 개발자와 사용자 사이에 모델을 통해 시스템의 一部 또는 全體의 수행을 보여주므로써 사용자는 자신의 要求事項을 정확히 전달할 수 있고 개발자는 개발초기에 완전한 요구를 얻게 된다.

2.2 客體指向 모델링의 構成要素

객체지향 모델링 기법은 객체지향 모델을 근거하여 分析하는 방법이다. 객체지향 모델은 客體모델(object model)과 動的모델(dynamic model)로 나누어 정의되고 모델링 기법은 각 모델을 만들어 나가는 방법으로 사용자의 요구를 정확히 把握하게 된다. 객체지향 모델링의 기반이 되는 두가지 모델의 特性은 다음과 같다.

## 1) 客體 모델(Object model)

시스템을 구성하는 객체의 構造-식별자, 관련성, 속성, 오퍼레이션-를 기술하는 것으로 모델링의 가장 중요한 기반이 된다. 構造的 設計 方法이 알고리즘을 기본적인 요소로 시스템을 구축하는 방법이었다면, 객체지향 방법은 객체와 클래스를 基本 要素로 시스템을 구성한다. 객체모델은 概念的인 기반 요소와 구성요소로 나누어 진다.

## (1) 객체모델의 개념적인 기반요소

- ① 추상화(Abstraction) : 추상화란 人間이 복잡한 문제를 다루는데 가장 基本的인 방법으로 다른 종류의 객체와 구분되는 객체의 필수적 屬性을 나타낸 것으로 개념적 영역을 구분지워 준다. 객체가 어떤 機能을 수행 할 것인지 ("what to do") 에 초점을 맞추어 기술하는 것과 어떻게 구현 할 것인지 ("how to do") 에 초점을 맞추어 기술하는 것과 區分하여 외부의 관점을 決定하게 하는 요소이다.
- ② 캡슐화(Encapsulation) : 추상화 구현 이전에 이루어 지는 것으로 나중에 어떠한 구현사항이 決定되어도 이는 추상화의 內部事項으로 처리될 뿐 이 객체를 사용하는 다른 사용자에게는 감추어진 事項이 된다.
- ③ 모듈화(Modularity) : 모듈화란 프로그램을 독립적으로 設計 \* 코딩하여 컴파일 가능한 모듈로 나누고, 모듈간의 최소한 연결을 갖게하는 方法이다.

## (2) 객체모델의 기반요소

- ① 객체(Object) : 객체란 속성과 이들 속성을 다루는데 필요한 행위를 함께 캡슐화 한 것으로 問題 領域의 이해증진과 실제 해결 영역의 실제적 기반요소라는 두가지 역할을 담당한다. 객체지향 분석의 관점에서 보면, 객체는 屬性과 行爲의 캡슐화이면서 문제영역의 추상화인 것이다. 각 객체는 그들이 행하는 행위에 따라 區分되는데 행위란 객체가 어떻게 동작하고 어떻게 외부의 요구에 對應하는가를 定義한다.
- ② 속성(Attribute) : 객체가 가지는 資料의 속성을 의미한다.
- ③ 행위(Procedure) : 각 객체가 수행하게 될 機能들의 집합.
- ④ 클래스(Class) : 동일한 속성, 동일한 행위, 다른 객체들과의 동일한 關聯性을 갖는 객체 집합에 대한 일반적 技術 形態라고 볼 수 있다. 클래스 내 모든 객체들은 속성의 값만 달리할 뿐 동일한 속성과 행위를 갖게 된다.
- ⑤ 一般化/詳細化(Generalization/Specialization : IS-A relationship) : 문제의 영역에 존재하는 要素들 사이의 분류 구조를 나타내는 것으로 객체들 사이의 공통성을 공유하도록 일반화시켜 원래 존재하는 객체들을 새로운 객체로 재 정의하여 서로간의 차이를 유지한다. 이 관계를 통해 問題 領域을 속성과 행위를 기반으로, 문제영역을 構成하는 객체들이 공유하는 부분을 추출하여 여러 段階로 나누면 문제영역의 계층구조를 구성하게 된다.
- ⑥ 集合關係(HAS-A relationship) : 어떤 객체의 일부분인 객체와 그 객체 의 集合으로 구성되는 객체와의 관계를 나타낸다.
- ⑦ 連結關係(Link/Association) : 객체들 사이의 모든 從屬性을 나타낸다.

## 2) 動的 모델(Dynamic Model)

객체와 관련성들이 時間에 따라 변해나가는 과정을 動的 모델(Dynamic Model)의 설정을 통해 순간순간마다 외부의 자극에 반응하는 일련의 動作을 기술하게 된다. 동적모델을 어떤 事件에 의해서 시스템이 변해가는 모양을 表現해주는 상태 전이도의 집합으로 定義할 수 있다.

동적 모델의 구성요소는 '事件(Event)', '狀態(State)'이다. 객체가 가지고 있는 속성값과 연결관계를 객체의 현 '상태'라고 정의한다. 客體는 時間이 지나면서 다른 객체에게 메시지를 보내 어떤 행위를 하게 하므로써 메시지를 받는 객체의 상태를 변하게 하거나 외부의 刺戟에 의해서 자신의 상태를 계속 변하게 하여 원하는 行爲를 수행하게 하는 外部의 자극이나 한 객체가 다른 객체로의 자극(메세지 전송)을 '事件(Event)'으로 정의한다[4].

## 2.3 客體 지향 모델링의 分析節次

객체지향 모델링(Object Oriented Modeling)에 의한 問題 分析의 절차는 다음 <표 2.3> 에서와 같이 5段階를 거쳐서 이루어 진다.

표 2.3 객체지향 분석의 단계와 목적

단 계	목 적	수 행 작 업
Step 1 문제분석의 단계	시스템에 대한 요구를 시스템이 수행하는 행위를 중심으로 파악한다.	사용자와의 인터뷰, 실제 상황에 대한 경험, 참고서적을 통해 시스템이 갖추어야 할 기능의 흐름을 'activ script'로 작성
Step 2 객체의 선정단계	시스템의 기능을 수행하는 주체로서의 객체와 행위를 요구하는 주체로서의 객체를 파악한다.	객체의 추출과 해당 객체에 대한 역할을 기술
Step 3 객체의 속성정의 단계	객체가 지닌 속성을 정의한다.	각각의 객체에 대한 속성 파악과 속성에 대한 설명을 기술
Step 4 객체의 행위추출 단계	객체가 지닌 주체로서의 행위와 피주체로서의 행위를 기술한다.	객체가 수행하는 행위에 대한 이름의 선정과 설명을 기술
Step 5 객체들간의 관련성 정의단계	객체의 행위와 속성을 기반으로 객체들간의 상속, 집합, Link / Association 관계를 결정한다.	추상화, 그룹화 등의 기법 적용으로 관련성을 추출하고 이를 모델 카드에 반영

2.4 客體指向 모델링에 의한 Nurse Scheduling 問題의 分析

간호원 스케줄 전문가 시스템(Nurse Scheduling Expert System)에서 要求하는 기능을 <표 2.3>에 따라 分析하여 보면 다음과 같다[5].

Step 1 문제 분석의 단계 :

시스템이 수행할 행위를 기반으로 분석하는 첫 段階로서 구현할 시스템의 應用分野에 대한 개발자의 이해를 도모하며, 구현할 시스템의 범위를 決定하게 된다. 問題 분석의 단계로의 접근방법은 해당 應用分野의 전문가와의 대화를 통해 구현할 시스템의 要求事項을 파악하게 되며, 아주 간결하고, 核心的이며, 定型化되어 있어서 구현사항과 같은 자세한 내용을 포함하지 않는 응용분야 기술서를 참조해서 작성하는 행위기술서는 앞으로의 客體, 屬性, 行爲, 關聯性, 抽出 단계의 가장 기본이 되는 것으로 開發될 시스템이 작동하는 흐름을 기술하며, 흐름에서 발생하는 여러가지 사항들과 각 事項에서 시스템이 대처하는 方法을 기술하므로써 시스템이 갖춰야 할, 시스템이 처리하게 될 기능을 기술한다. 다음의 <표 2.4>는 看護員 스케줄 작성에 있어서 入力作業에 관한 행위기술서 이다.

표 2.4 Nurse Scheduling / 행위기술서

Script : Nurse Scheduling

기능을 요구하는 객체	수행하는 기능	요구를 받는 객체
Scheduler	scheduling priority/horizon 입력 요구	병 원
병 원	scheduling priority/horizon 입력	Scheduler
Scheduler	scheduling priority/horizon을 출력하여 확인을 요구	병 원
Scheduler	개별 Schedule 요구사항을 입력 요구	Nurse(RN, LPN, NA)
간 호 원	개별 Schedule 요구사항을 입력한다	Scheduler
Scheduler	개별 Schedule 요구사항을 확인 요구	간 호 원
간 호 원	개별 schedule 요구사항을 확인한다.	scheduler
Scheduler	입력 자료를 보관	array
scheduler	입력정보를 전달 받는다.	입력작업
scheduler	schedule horizon을 '1'(days-on)으로 초기화 한다.	schedule array
shceduler	nurse의 weekend off와 special off를 '0'(days-off)으로 초기화 한다.	schedule array

scheduler	Number of days-on nurse $\geq$ MNR 비교	array, MNR
병원 간호원	$MNR \leq$ (Number of nurse/2)인 경우 대안을 제시한다.	array, MNR
기능을 요구하는 객체	수행하는 기능	요구를 받는 객체
scheduler	$\sum$ days-on nurse = contacted days 비교	array
scheduler	전체 schedule이 priority를 만족하는 지를 검색한다.	array, schedule priority
병원, 간호원	schedule을 요구한다.	scheduler
scheduler	schedule을 출력한다.	병원, 간호원
병원, 간호원	schedule 결과에 대한 설명을 요구	scheduler
scheduler	schedule 결과에 대한 추론과정을 출력	병원, 간호원

Step 2. 客體의 選定 段階 :

객체 선정 단계에서는 시스템을 구성하는 객체를 파악하는 段階로 Step 1의 결과물인 행위기술서를 근거로 작성한다. 行爲技術에서의 3번째 항목인 '요구를 받는 객체'를 시스템을 구성하는 객체로 抽出한다.

표 2.5 Nurse Scheduling / 객체기술서

객체의 이름	객체가 수행하는 역할에 대한 기술
scheduler	병원의 정책과 간호원의 요구를 만족하는 schedule 작성
병원	schedule에 대한 병원의 정책을 포함한다.
간호원	schedule 구성의 주 요소
array	schedule 작성시 작업영역을 제공
R N(수간호원)	schedule 구성의 주 요소
LPN(평간호원)	schedule 구성의 주 요소
NA(조부사)	schedule 구성의 주 요소
MNR(최소요구간호원수)	schedule의 적합성 판단 기준
DNR(적정운영간호원수)	schedule의 효율성 판단 기준
schedule priority	schedule의 우선 순위

Step 3. 客體의 屬性 定義 段階 :

객체가 지닌 속성을 파악하기 위해 행위기술서의 두번째 항인 '遂行하는 行爲'를 확실한 속성이 있는지 把握하고, 객체 기술서에 기록된 객체가 지닌 속성으로 기술서를 作成한다.

표 2.6 Nurse Scheduling / 속성기술서

객체의 이름	속 성
scheduler	Inference Rule & Knowledge Base
array	Scheduling Horizon, Nurse class, Shift, Contacted day
RN(Resitered Nurse)	ID_Nurse, Name, Shift, Weekend off day, Special off day
LPN(Licensed Pract- ical Nurse)	Contacted day
NA(Nurses' Aide)	ID_Nurse, Name, Shift, Weekend offday, Special off day
MNR	Contacted day
DNR	Nurse class별 최소요구 간호원 수(RN, LPN, NA)
Schedule priority	Nurse class별 적정요구 간호원 수(RN, LPN, NA)
병원	$P_1 = MNR, P_2 = \text{Special off day} + \text{Weekend off day}$
간호원	$P_3 = DNR$
	MNR, DNR
	ID_Nurse, RN, LPN, NA, Shift, Weekend off day
	Special off day, Contacted days, Name

Step 4. 客體의 行爲 抽出 段階 :

행위 기술서에 나타난 수행하는 機能은 기능의 수행을 요구하는 객체의 觀點에서 기술된 것이다. 이를 그 機能을 수행하게 될 객체의 觀點에서 바꾸어야 한다. 즉, 要求를 수행하게 될 객체가 이행해야 할 行爲로, 요구를 수행하게될 客體가 이행하는 行爲의 개념으로 變化시켜야 한다.



표 2.7 Nurse Scheduling / 기능에 대한 행위기술서

요구를 수행하는 객체	행 위
병원 scheduler	Schedule priority/horizon을 입력한다.
병원 간호원 scheduler	schedule priority/horizon을 입력받는다.
Array	schedule priority/horizon을 확인한다.
Array	개별 schedule 요구사항을 입력한다.
Array	개별 schedule 요구사항을 입력 받는다.
Array, MNR	개별 schedule 요구사항을 확인한다.
Array	입력정보를 보관한다.
Array, MNR	scheduling horizon 전체를 '1'(days-on)로 초기화한다
Array	weekend off와 special off를 '0'(days-off)로 초기화한다.
Array, MNR	days-on nurse 과 MNR을 비교한다.
Array	MNR ≤ (Number of nurse/2)인 경우 대안을 제시한다
Array, scheduler	days-on nurse in horizon = contacted days인지 검색
Array, scheduler	전체 schedule이 모든 priority를 만족하는 schedule인지를 판단한다.
nurse, 병원 scheduler	schedule을 출력한다.
nurse, hospital scheduler	schedule을 출력 받는다.
	schedule에 관한 설명을 요구 받는다.
	schedule에 관한 설명을 받는다.

Step 5. 客體들간의 關聯性 定義段階 :

이 단계에서는 객체들간의 共通性, 集合性, 從屬性 등을 파악하는 단계로 파악하게될 관련성은 객체 모델을 근거한 행위기술서, 속성기술서, 객체기술서에서 一般化/詳細化와의 관계, 집합관계, Link/Association 관계를 抽出하게 된다.

다음 <그림 2.5>에서는 看護員 문제에서 나타난 객체들간의 관련성을 의미 네트워크(Semantic Network)에 의한 IS-A relation, HAS-A relation, Link relation으로 表示하고 있다.

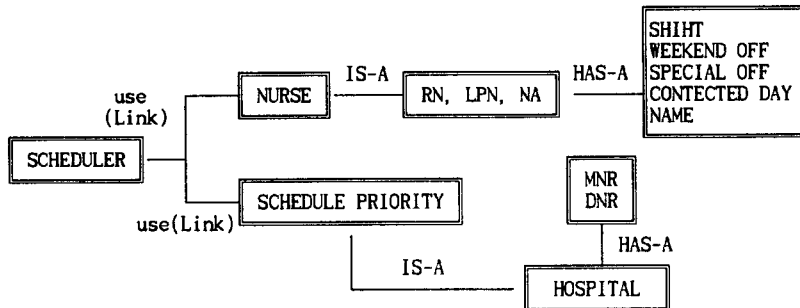


그림 2.5 의미네트워크에 의한 객체간의 관련성 표현

本 研究에서는 C言語를 사용하여 Scheduling에 관한 지식을 Rule-Base 形態로 표현하였고, 지식의 추론 형태는 데이터 구동형의 전향추론(forward chaning)과 目標指向의 후향추론(backward chaining)을 함께 사용하여 目的하는 看護員의 Schedule을 작성 하도록 하였다. 또한 User의 schedule 結果에 대한 說明을 위하여 Dialogue Box를 채택하였다.

3. Expert Nurse Scheduling Program의 構成 및 適用

3.1 프로그램상의 知識表現 例

本 研究에서는 스케줄을 작성하는데 필요한 지식을 3장에서 모델링한 客體를 다음과 같이 표현되었다.

① 간호원이 가지고 있는 객체의 코딩 예 :

```
struct nurse {
    char name[6];
    char nur_class[4];
    int spec_off;
```

```

    int weekend_off[2];
    int spec_week_off;
    int shift;
    int con_day;
    struct nurse *next;
}

```

- ② special off day와 weekend off를 할당하는 모듈의 코딩 예 :

```

nur_t=nur_init;
for(i=1;i<=nur_count;i++){
    off_day[i][nur_t->spec_off]=0;
    off_day[i][nur_t->weekend_off[0]]=0;
    nur_t=nur_t->next;
}

```

- ③ shift change에 관한 추론틀의 코딩 예 :

```

case 10 : if(schedule[i][j]==1){
            if(change==0)
                sche_comp[i][j] = 'D';
            if(change==1)
                sche_comp[i][j] = 'E';
            if(change==2)
                sche_comp[i][j] = 'N';
        } else {
            sche_comp[i][j]=='0';
            if(sche_comp[i][j-1]=='D')
                change = 1;
            if(sche_comp[i][j-1]=='E')
                change = 2;
            if(sche_comp[i][j-1]=='N')
                change = 0;
        }
}

```

- ④ 대화박스의 메시지 출력 추론틀의 코딩 예 :

```

for(i=1;i<=con_day;i++){
    message=0;
    MNR_count=0;
    for(j=1;j<=nur_count;j++){
        if(schedule[j][i]==1)
            MNR_count++;
    }
    if(MNR > MNR_count)
        message 1;
    if(message==1){

```

```

printf("%d일의 간호원 off수(%d명)가 많으므로 스케줄조정을 바랍니다", i, MNR_count)
}
}

```

### 3.2 프로그램운용 예

#### 1) 對象問題의 데이터

本 例題의 데이터는 J. Arthur의 論文에서 사용한 데이터로 本 프로그램에서는 다음과 같은 입력 과정을 거친다.

## 參 考 文 獻

1. 류성렬, 시경숙, C 인공지능 프로그래밍, 세웅, 1990.
2. 정환득, 해설중심 인공지능, 동일출판사, 1989.
3. Amar Gupta, Bandreddi E. Prasad, Principles of Expert Systems, IEEE Press, Inc., 1988.
4. Hiller, Liberman, Operation Rearch, 4th Edition, McGraw-Hill, 1989.
5. Holmes E. Miller, W .P. Pierskalla and G. J. Rath, "Nurse Scheduling Using Mathermatical Programming", OR, Vol. 24, No. 5, Sep-Oct, 1976.
6. Jae Kyu Lee, Sung Hoon Shim, "A Knowledge-Based formulation of Linear Programming Models Using UKIT-OPT", 2nd International Workshop on AI in Economics and Managemant, 1976.
7. Robert J. Mockler, Knowledge-Based Systems For Managemant Decisions, Prntice-Hall International, inc., 1989.
8. Wenran Zhang, Michael N. Huhns, "Scheduler: A Knowledge-Based system For Personnel Scheduling", Proceedings of the 17th Southestern Symposium on System Theory", 1985.