

13주차 2차시 분석 방법론 종류

【학습목표】

1. 객체 모델링, 동적모델, 기능모델의 개념을 설명할 수 있으며, 모델의 유도방법을 설명할 수 있다.
2. 코드와 요단의 방법으로 객체지향분석을 다섯 단계로 구분할 수 있다.

학습내용1 : 람바우 · GE의 방법

1. 객체 모델링

1) 객체 다이어그램(object diagram)은 다음과 같이 나타냄.

- ① 클래스(class) 표현 기호 : 사각형.
- ② 객체의 인스턴스(instance) 표현 기호는 모서리 둥근 사각형.
- ③ 클래스는 속성(attribute)과 연산(operation)을 가짐.

* 속성

- 클래스가 가지는 값을 나타내는 것으로서 「자료형(data type)」과「기본 값(default value)」으로 구분됨.

* 연산

- 「인수 리스트(argument list)」와「결과형(result type)」으로 정의됨.

2) 객체와 클래스의 연결

- ① 각자 어떤 참고서든지 참고 가능함

3) 일반화와 상속성

- ① 일반화(generalization)에 의해서 정의된 상위클래스의 속성은 하위클래스에 상속되어 사용됨. 이 상속 관계는 일정한 순간 측면에서 볼 때 하위에 존재하는 여러 개의 객체 중에서 어느 하나에만 해당됨.
- ② 이러한 관계를 객체 다이어그램(object diagram)에 나타낼 경우에 「삼각형」을 사용함.

4) 자금의 이동, 회수와 예금을 기록하고 고객의 데이터를 관리하고 수집하거나 조직의 판매관리 또는 생상품 안내를 기록

5) 모듈

- ① 거대하고 복잡한 업무를 해결하기 위해서는 모듈(module) 단위로 구분하여 처리함이 효과적임.
- ② 모듈화 추진과정에서 유의사항은 다음과 같음.
 - 동일·유사한 기능이나 목적에 필수적인 객체는 모두 동일한 모듈로 작성함.
 - 모듈규모의 적정성을 염두에 두어 지나치게 크거나 작지 않도록 해야 함.
 - 동일 모듈 내에서 클래스 이름(class name)은 유일해야 함 : 서로 다른 모듈에 동일한 클래스 이름이 존재해도 무방함.

- 상속구조·조립구조를 가지는 클래스들은 동일 모듈로 작성해야 함 : 다른 모듈로 작성하면 어려움이 많이 발생함.

2. 동적모듈

1) 시간의 흐름에 따라 변하는 시스템 특징을 파악하기 위하여 동적모델(dynamic model)을 작성함.

2) 동적모델의 주된 적용분야는 다음과 같음.

① 응답시간 개념이 중요한

- 실시간 시스템(real time system).

② 객체 사이의 관련이 수시로 이루어지는 상호작용 시스템(interactive system).

③ 정적(static)인 자료관리가 핵심 기능인 데이터베이스 시스템은 변화가 없으므로 작성하지 않음.

3) 동적모델 작성에 포함되어야 할 핵심적 사항 다음과 같음.

① 특정 시점에서 객체의 상태(state).

② 그 상태에서 객체의 활동.

- 활동 결과에 의한 객체의 상태 변화

4) 일반적인 동적모델의 작성 절차는 다음과 같음.

① 시나리오 작성.

② 사건의 정의.

③ 객체상태다이어그램 작성.

④ 동적모델 작성.

5) 기초 개념

① 사건과 시나리오

- 시스템의 활동을 전제로 하는 개념이 동적모델(dynamic model)임.

- 어떤 객체가 다른 객체를 자극하여 원하는 활동을 유발하기 위하여 「사건(event)」이 필요함.

- 사건의 발생 원인은 다음 두 가지가 있음.

- 주로 다른 객체에 의해서 발생함.

- 경우에 따라서는 일정한 시간 경과로 인해서 발생하기도 함.

- 객체는 사건에 대해서 무조건 반응하지 않고, 자기에게 관련된 경우에만 반응함.

- 시스템을 수행시키는 사건의 연속을 시나리오(scenario)라고 함.

- 시스템의 활동은 정해진 목적과 관련을 가지고 있는 사건의 연속으로 규정 가능함.

- 시스템 활동에서 발생하는 사건들을 차례대로 나열하여 정리한 내용이 「시나리오」임.

- 시나리오는 「사건의 순서」, 「사건에 관련된 객체」를 정확히 파악 가능하다는 장점을 가짐.

- 전형적인 시나리오의

예) 각자 참고 서적에서 참조 요망.

- 시나리오 표현의 부족한 사항을 보완하기 위해서 시나리오의 내용을 한 눈에 인식할 수 있도록 역할과 시간의 흐름에 따른 사건의 전개과정을 그래프 형태로 나타낸 것을 「사건추적도(event trace)」라고 함.

- 기본적인 사건추적도 작성 규칙은 다음과 같음.
 - 시나리오에 나오는 객체는 수직선 위에 기재함.
 - 사건은 시나리오에 나온 순서대로 위에서 아래로 기재함.
 - 화살표로 사건 내용의 전달 방향을 표시함.
 - 화살표 위에는 전달정보·사건 이름을 기재함.

② 상태 다이어그램

- 사건의 진행에 따라서 객체의 상태가 어떻게 변화해 가는지 명확히 이해하기 위하여 「사건의 진행에 따라서 객체의 상태가 변화되는 과정을 효과적으로 파악할 목적」으로 「상태다이어그램(state diagram)」을 작성함.
- 상태 다이어그램 작성 시 주요한 규칙은 다음과 같음.
 - 구성요소 중에서 상태명, 사건, 화살표 등은 모든 상태 다이어그램에 반드시 기재해야 하지만, 그 외의 구성요소들은 불필요 시에 생략함.
 - 상태를 나타내는 기호는 「타원」임. 그 속에「상태명」을 기재함.
 - 사건을 나타내는 기호는 「화살표」임. 화살표의 방향은 상태의 변화 방향 으로 취급함.
 - 「사건명」이나「전달정보」는「화살표」위에 기재함.
 - 특정상태에서 지속적인 수행활동은 「do : 활동」형태로 기재함.
 - 후속상태의 결정조건이 존재하는 경우 「[조건)」형태로 기재함.
 - 순간적 수행행위가 존재할 경우에는 「/행위」형태로 기재함.
 - 모든 출력사건에 동일행위가 존재할 경우에는 「종료/행위」형태로 모아서 기재.
 - 모든 입력사건에 동일행위가 존재할 경우 「시작/행위」형태로 모아서 기재.
- 모든「객체는 역할이 다르기」때문에 각 객체는 서로 다른「사건」과「상태」를 가짐. 그러므로 각 객체에 대하여 개별적으로 「상태 다이어그램」을 작성해야 함.
- 상태 다이어그램 작성 시에 객체의 상태에 개수(個數)가 지나치게 많아지면 복잡해지기 때문에 복잡한 여러 가지 상태 중에서 유사한 것끼리 그룹화(grouping) 하여 간단한 상태 다이어그램을 작성하는 것을 「상태의 일반화(state generation)」라고 함.
 - 여러 계층의 다양한 상태 다이어그램의 작성이 가능함.
 - 상위계층에서 하위계층으로 가면서 단계적으로 객체의 활동을 구체화 가능함.
- 어떤 특정한 상태가 자신을 구체화하기 위하여 하위상태를 가질 경우 이를 「중첩상태(nested state)」라고 함.
- 결국 동적모델은 각 객체별로 작성된 상태 다이어그램을 모아 놓은 것임.
 - 각 객체별로 작성된 상태 다이어그램을 직접적으로 연결하지 않는다는 점임.
 - 각 객체별로 작성된 상태 다이어그램들은 공유사건을 해결하기 위하여 간접적으로 연결됨.

3. 기능모델

1) 객체분석 과정에서

- ① 첫 단계에서 작성되는「객체모델(object model)」대상 시스템의「무엇(what)」을 정의하고,
- ② 두 번째 단계에서 작성되는 「동적모델(dynamic model)은「언제(when)」를 정의하며,
- ③ 마지막단계에서 작성되는 「기능모델(function model)」은 「시스템에서 제공되어야 할 기능」을 정의함.

2) 기능모델을 작성하는 과정에서는 이전의 두 가지 단계에서 부분적으로 기술 (description)된 시스템의 기능에 대하여

- ① 어떤 과정을 거쳐서 입력된 내용이 출력으로 변환되는가,
- ② 제공하려는 기능 구현에 제약요소가 무엇인지 구체적으로 기술해야함.

3) 기능모델을 작성하는 과정은 다음과 같이 네 단계로 진행됨

- ① 입출력 자료 결정.
- ② 자료흐름도 작성.
- ③ 기능명세서 결정.
- ④ 제약사항 파악.

4) 입출력 자료 결정

- ① 기능모델 작성에서 우선적으로 결정되어야 할 사항은
 - 외부에서 받아들여 처리에 사용하는 입력자료(외부객체가 시스템에 제공하는 원시자료)가 무엇인지 결정되어야 함.
 - 내부에서 처리가 완료된 후에 외부에 어떤 출력자료를 제공할 것인지 결정해야 함.

5) 자료흐름도 작성

- ① 기능모델 표현 도구로서 자료흐름도(data flow diagram : DFD)를 작성함.
- ② 이 자료흐름도는 프로세스, 자료흐름, 자료저장소, 단말 등의 요소로 작성됨. 구체적인 기호·작성법은 참고문헌 참조.

6) 기능명세서 작성

- ① 프로세스 기능명세서(process function specification)는 자료흐름도에서 표현된「프로세스(process)」는 그 역할을 함축적으로 나타내고 있는 관계로 구체적인 내용 파악이 어려움. 이를 보완하기 위해서 자료흐름도의 프로세스에서 수행될 기능(function)의 내용이 무엇인지 간략하게 나타내는 것이 프로세스 기능명세서임.

② 프로세스 기능명세서 작성 도구

- 구조적 언어, 의사코드, 자연언어, 디시전 테이블 등등

7) 제약사항 파악

① 각 프로세스

- 특정한 객체의 전부·일부분으로 구성됨
- 그러므로 객체 사이의 특정한 관련으로 인해서 발생하는 문제들은 프로세스 활동에 직접적인 영향을 미치기 때문에 기능모델을 작성하는 과정에서 이들을「제약사항」으로 파악하여 별도로 정리해 두어야 함.

② 기능모델을 작성하는 과정에서 프로세스(process)의 활동에 영향을 미치는 제약사항

- 다수 객체에 영향을 미치는 사항.
- 서로 다른 시점에서 여러 인스턴스(instance)가 충족되어야 할 사항.
- 프로세스(process)의 입력 값이나 출력결과가 충족해야 할 사항.

4. 모델의 유도방법

- 1) 제기된 업무에 대해서 모델을 유도하는 과정에서 미비사항 혹은 부족사항 등으로 인 해서 단번에 완전한 객체모델을 작성하는 것은 불가능함.
- 2) 그러므로 최초에 작성한 모델을 기초로 하여 개선과정을 되풀이하는 「반복적 모델화(iterative modeling)」를 통해서 완전한 모델을 유도하도록 해야함.

학습내용2 : 코드와 요든의 방법

1. 코드와 요든의 방법

- 1) 객체지향분석(object oriented analysis : OOA)를 다음과 같은 다섯 단계로 제시 하고 있음.

- ① 객체 식별
- ② 구조 식별
- ③ 주제 정의
- ④ 속성과 인스턴스의 연결
- ⑤ 연산과 메시지의 연결

2. 객체 및 구조 식별

- 1) 하위클래스의 공통적인 속성을 상위클래스로 정의하면 그 내용은 언제든지 상속하여 사용이가능함.
- 2) 그러므로 객체의 식별과 선정되면

- ① 객체에 분류구조(classification structure)의 원리를 적용하여 각 객체에 대한 인스턴스를 정의하고, 인스턴스 이름을 결정함.

- ② 특수화(specialization)에 의해서

- 하위클래스를 생성하거나
- 이전 단계에서 누락되었던 클래스를 새로 추가하여
- 연결하는 전체-부분구조(whole-part structure)를 이용하여 객체구조를 결정하여 표현함
- 전체부분구조를 나타내는 기호는 「삼각형」임.

3. 주제 정의

- 1) 객체모델 작성에서 거대·복잡한 시스템은 전체를 하나로 모델링 하는 것보다 해당 시스템을 주제(subject)별로 분할하는 것이 효과적임.

- 2) 주제별 분할은

- ① 이전 단계에서 찾아낸 상위클래스를 주제후보로 결정한 후에,
- ② 그와 밀접한 관계가 있는 객체들을 적절한 규모의 단위로 그룹화 (grouping) 하는 것을 의미함.

3) 거대하고 복잡한 시스템의 경우에 구조모델들을 이해하거나 사용하기 어려움.

4) 그러므로 이 경우에 객체와 구조모델을 간략하게 표현하여 전체 내용을 쉽게 파악할 수 있도록 하는 기법이 「주제정의(subject definition)」임.

4. 인스턴스와 메시지 연결

1) 독립적으로 정의된 객체 사이에 존재하는 관계에 따라서 상호 연결하여 표현하는 것은 매우 중요함.

- 연결방법 참고서 참조 요망.

【학습정리】

1. 람바우 • GE의 방법의 객체 모델링을 알아본다.

2. 람바우 • GE의 방법의 동적모델을 파악한다.

3. 람바우 • GE의 방법의 기능모델을 이해한다.