

## 드라이퍼스 모델 기반 디자인 패턴 학습 모델 제안

문현준\*, 김정선<sup>o</sup>

\*한양대학교 컴퓨터공학과,

<sup>o</sup>한양대학교 컴퓨터공학과

e-mail: mhj5730@hanyang.ac.kr\*, kimjs@hanyang.ac.kr<sup>o</sup>

## Suggestions for learning design patterns based on the Dreyfus model

HyunJun Moon\*, Jungsun Kim<sup>o</sup>

\*Dept. of Computer Science and Engineering, Hanyang University,

<sup>o</sup>Dept. of Computer Science and Engineering, Hanyang University

### ● 요약 ●

디자인 패턴은 클래스와 객체를 활용한 23가지의 개발자 의도를 만족시켜주는 최선의 실천법들을 정리한 것이다. 디자인 패턴은 설계 경험과 객체지향 패러다임의 기반 지식들을 필요하므로 실질적인 패턴 학습에 어려움이 있다. 디자인 패턴 학습에 대한 도움을 제시해 줄 수 있는 가이드라인으로 기술 습득 모델에 활용하는 드라이퍼스 모델을 적용하는 것을 제안하고자 한다. 드라이퍼스 단계별 모델을 기반으로 단계 별 디자인 패턴 학습 단계를 제시한다.

**키워드:** 드라이퍼스 모델(Dreyfus Model), 디자인 패턴(Design Pattern)

## I. 서론

소프트웨어공학 지식체계 프로젝트(Software Engineering Body of Knowledge project / SWEBOOK)는 전문 소프트웨어 엔지니어가 반드시 갖추어야 할 능력을 구성하는 지식 영역들에 대하여 총 10가지로 구분하였다. 10가지의 지식 영역 중 하나인 소프트웨어 설계(Software Design) 영역은 아키텍처 레벨 또는 상세 레벨에서의 시스템 기본 구조의 정의와 모듈별 분할, 모듈의 인터페이스 정의와 모듈 내의 알고리즘 설계에 대한 역량을 의미한다. 디자인 패턴은 소프트웨어를 구성하는 모듈에 대한 설계를 학습할 수 있는 최선의 실천법에 대한 학습 모델이 되어준다. 생성(Creational), 행위(Behavior), 구조(Structural)로 총 3가지로 분류된 23가지의 디자인 패턴에 대한 학습 모델에 대하여 기술 습득의 5단계 모델을 제시하는 드라이퍼스 모델을 적용하여 단계별 학습법을 제시하고자 한다.

## II. 이론적 배경

### 1. 디자인 패턴

객체지향 언어에서는 객체가 가진 메서드와 클래스를 통해서 시스템 구축을 위한 개발 의도를 해결한다. 디자인 패턴은 클래스와 객체를 활용한 방식으로 23가지의 개발자의 의도를 만족시켜주는 최선의

실천법(Best Practice)들을 정리한 것이다. 하나의 패턴에는 다음의 4가지 요소가 반드시 들어간다. 해당 패턴의 의도를 전달하는 데 도움을 주는 패턴 이름, 패턴이 필요한 배경이 되는 문제 상황, 문제를 해결하는 데 도움을 줄 수 있는 구조적 해결책인 해법 그리고 패턴을 이용하게 될 경우 얻게 되는 결과이다. 디자인 패턴은 설계의 관점과 구축의 관점을 동시에 접근하므로 학습의 어려움이 존재한다. 이 어려움을 해결하는 데 도움이 되고자 기술 습득에 활용할 수 있는 드라이퍼스 5단계 모델을 제시하고자 한다.

### 2. 드라이퍼스 5단계 모델

드라이퍼스 모델은 철학자 드라이퍼스가 정리한 기술 습득에 대한 5단계 모델을 의미한다. 5단계의 모델은 초보자, 고급 입문자, 중급자, 숙련자, 전문가로 분류 하는데 각 단계의 주요 특징은 다음과 같다.

#### 1. 초보자(Novice)

사전 경험이 없는 단계이다. 일을 해내기 위한 능력에 관심이 많다. 이들을 관리하기 위해선 규칙이 필요하다.

#### 2. 고급 입문자(Advanced Beginners)

고정된 규칙에서 조금씩 벗어나는 단계이다. 자신만의 작업을 통해서 경험을 쌓을 수 있다. 하지만 여전히 문제 해결에는 어려움을 느낀다.

**3. 중급자(Competent)**

문제 영역에 대한 개념적인 모델을 정립할 수 있다. 새로운 문제 해결방법을 찾아내기 시작한다. 과거의 경험을 활용하여 학습할 수 있다.

**4. 숙련자(Proficient)**

더 큰 개념적인 틀을 통해 이해하고자 한다. 사례 연구(Case Study)를 통해 학습한다. 상황에 따라 규칙을 변형할 수 있다.

**5. 전문가(Expert)**

적절한 맥락에서 활용 가능한 방대한 경험을 갖추고 있다. 직관적으로 더 나은 해결책을 선택할 수 있다.

**3. 모델 제한**

드라이퍼스 모델에 기반한 디자인 패턴 학습 모델의 표는 다음과 같다. 전문가의 단계는 제외하였다.

Table 1. 드라이퍼스 모델 기반 디자인 패턴 학습 단계

드라이퍼스 모델	디자인 패턴 학습
초보자(Novice)	객체지향 패러다임 이해, 합성, 상속, 기초적인 UML 이해, 디자인 패턴의 개요 이해.
고급 입문자(Advanced Beginners)	디자인 패턴 3분류에 대한 의도, 문제, 해결책, 결과, 패턴 구조 학습. 이해를 위한 특정 도메인 문제 접근.
중급자(Competent)	실제 어플리케이션에 적용된 디자인 패턴 학습. 라이브러리, 프레임워크 모델 기반 학습. 디자인 패턴 활용 논문 학습.
숙련자(Proficient)	특정 도메인과 관련된 어플리케이션 제작에 디자인 패턴 활용 후 전문가 피드백.

모델별 디자인 패턴 학습 모델을 통해서 자신의 기술 단계에 맞게 디자인 패턴 학습을 시도하는 것에 도움을 제공하고자 한다.

**IV. Conclusions**

디자인 패턴은 객체지향 패러다임을 활용하여 개발자의 설계 의도를 이해할 수 있는 최선의 실천법을 모은 것이다. 하지만 설계와 구축 경험으로 인한 어려움과 객체지향에 대한 기반 지식의 필요성으로 디자인 패턴 학습에 어려움을 가지고 있다.

이에 대하여 드라이퍼스 모델을 기반으로 하는 학습 단계를 제시하여 디자인 패턴 학습에 가이드라인을 제공하고자 한다. 이후에는 고급 입문자를 위한 각 패턴별 현실 도메인 예시와 중급자를 위한 실제 어플리케이션에서 활용된 라이브러리와 프레임워크 기반의 예시를 추가해서 디자인 패턴 학습 이해에 도움을 제공할 수 있는 최적의 예제를 추가하고자 한다.

**REFERENCES**

[1] Gamma, E. et al. Design Patterns – Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994.

[2] Dreyfus S, Dreyfus H. A five stage model of the mental activities involved in directed skill acquisition. California University Berkeley Operations Research Center [monograph on the Internet]; 1980. Available from: <http://www.dtic.mil/dtic/index.html> [downloaded 12 January 2009].

[3] P. Bourque and R.E. Fairley, eds., Guide to the Software Engineering Body of Knowledge, Version 3.0, IEEE Computer Society, 2014; [www.swebok.org](http://www.swebok.org).