

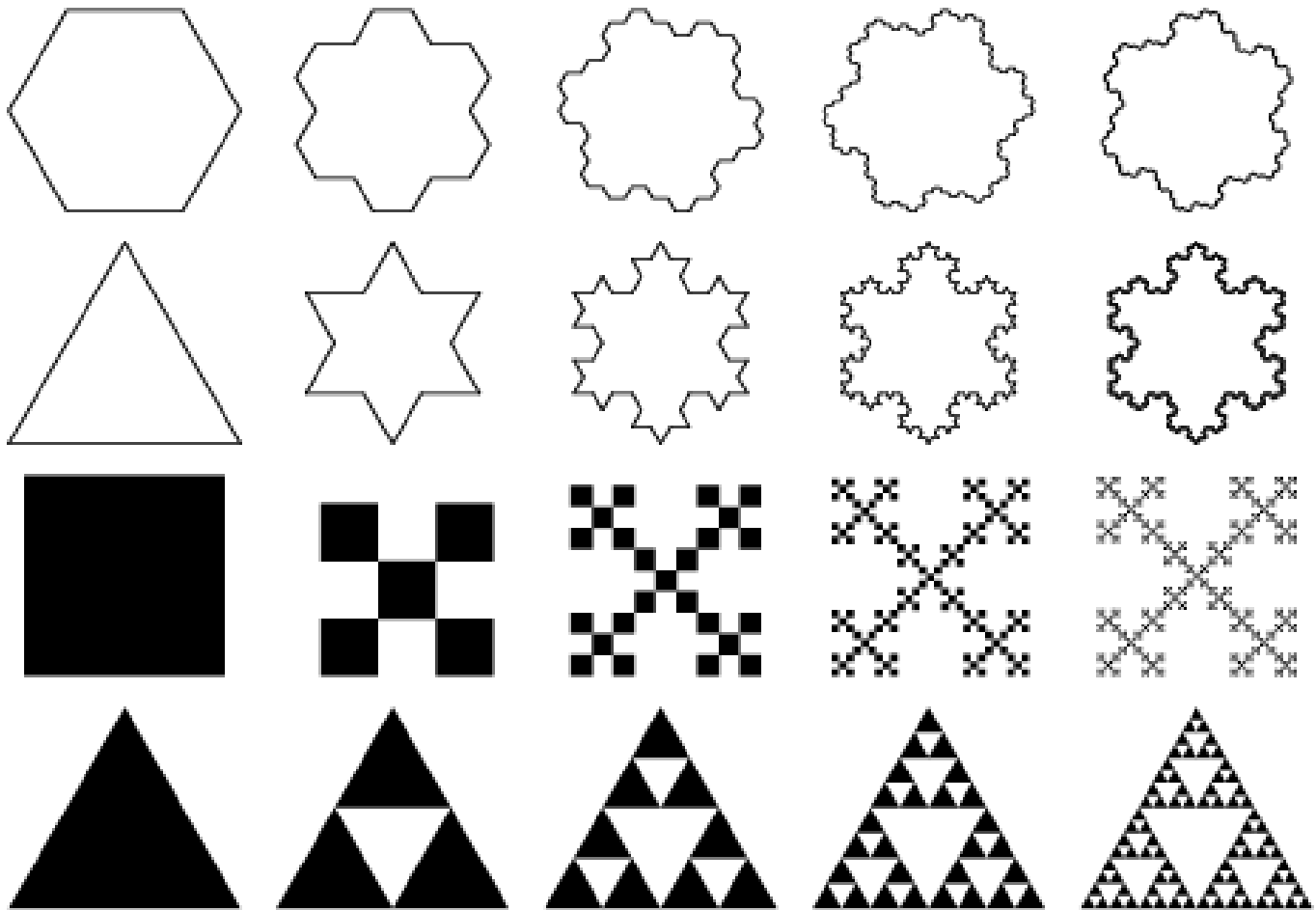
UML의 기본

- 1.1 서론
- 1.2 모델링
- 1.3 FUM
- 1.4 연습문제

2.1 서론

- SW개발시, 필수요소들
 - Modeling기법, 개발공정, 도구, ...
- FUM(Fractal UML Modeling)
 - "Fractal" – 재귀반복하면서 사물을 만듦.
 - Agile소프트웨어개발방법에 UML을 주입시킨 모델링 기법
 - "모델 중심의 Agile소프트웨어개발방법론"

Fractal : 재귀반복하면서 사물을 만듦



www.fractal.art.pl



Agile.....

- ...
- 빈번하게
- 자주
- 직접
- 민첩하게
- ...

1. ADJ

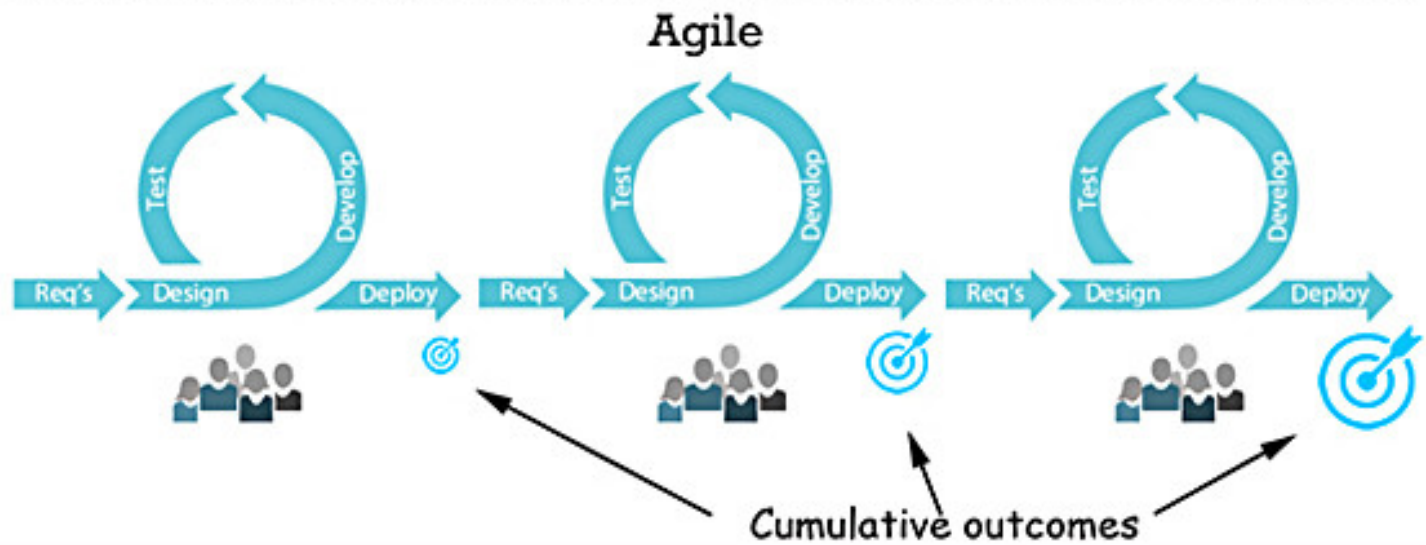
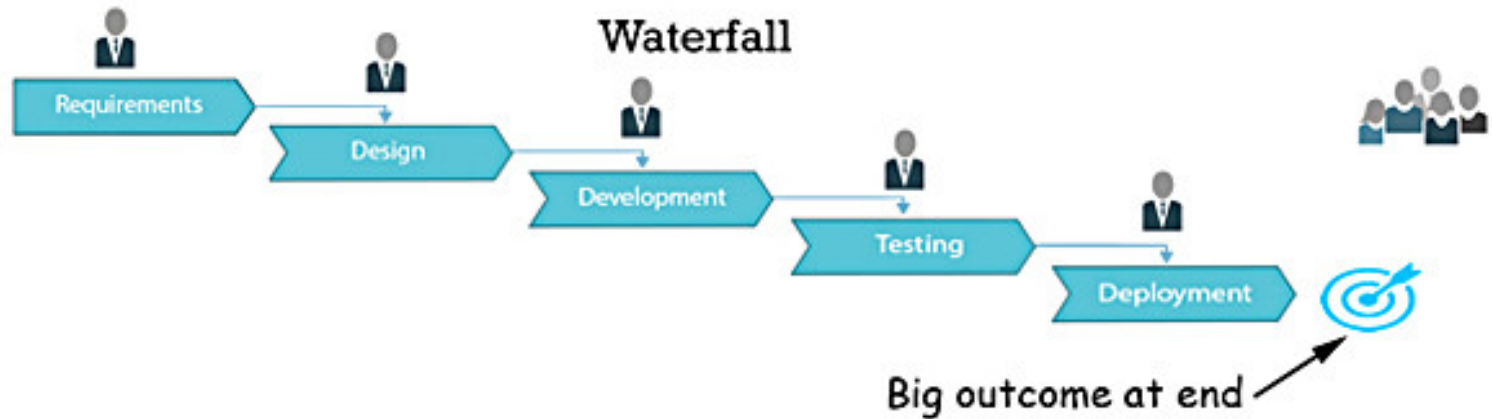
able to move your body quickly and easily.

2. ADJ

able to think quickly and clearly.

機敏하다 / 敏捷

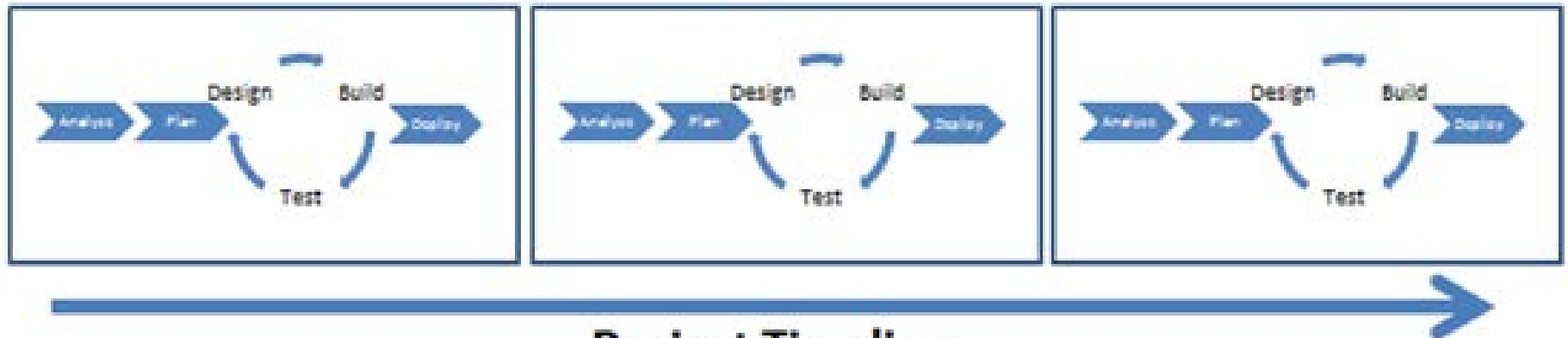
눈치가 빠르고 동작이 날쌔다.



Waterfall



Agile

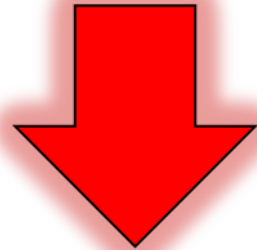


Project Timeline

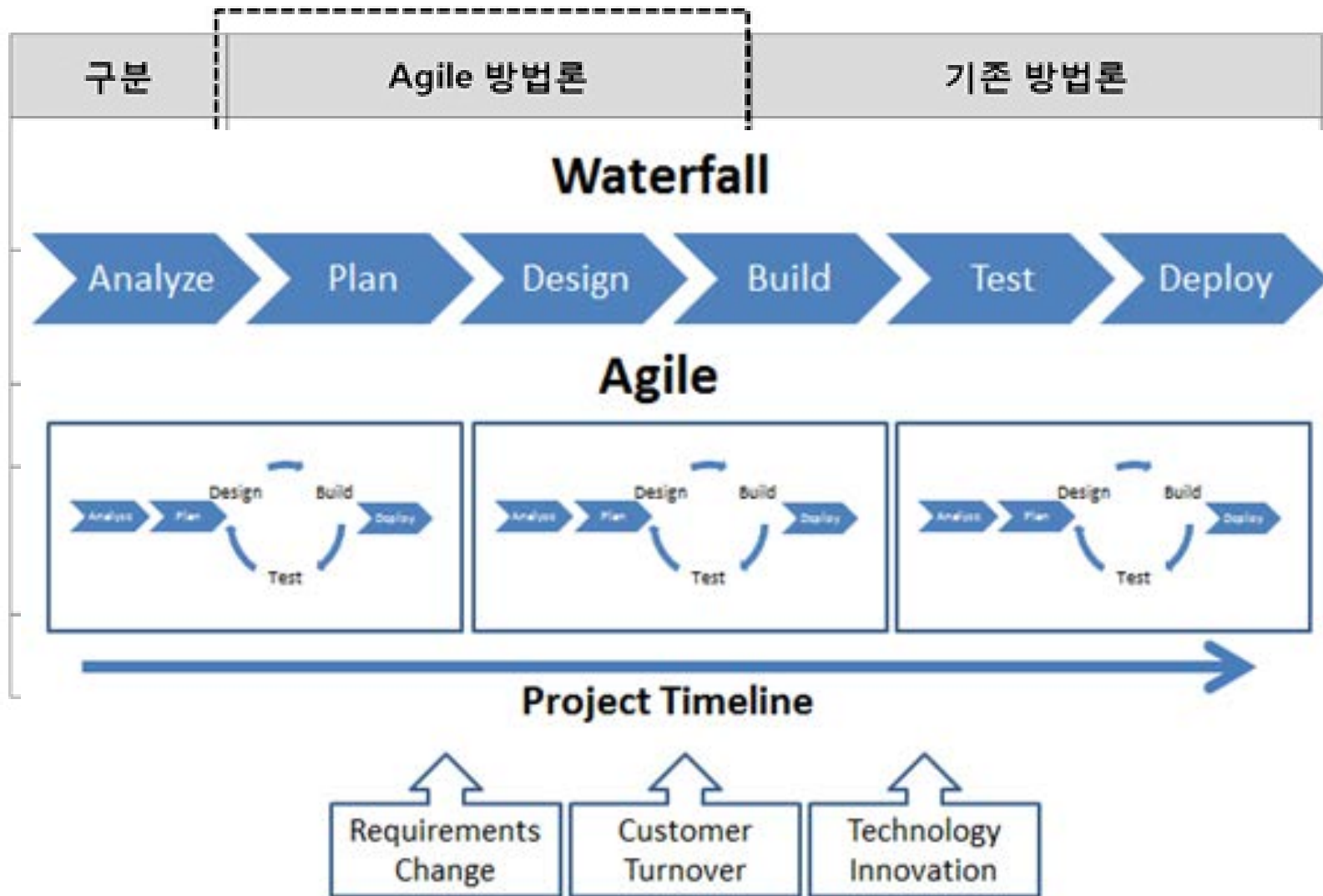


Agile SED의 등장배경

- Waterfall모델의 문제점
 - 사용자의 요구를 정확하게 반영하기 힘들
 - 지속적으로 변화하는 요구사항을 적절히 처리할 수 없음
 - 개발된 소프트웨어 모듈들이 잘 조합되지 않을 수 있음
 - 품질 저하



비즈니스의 요구에 유연하고 민첩하게 대응해야...



Agile Software Development

- 문서에 대한 부담을 줄이면서 변화에 쉽게 대응하며 고객의 입장에 초점을 맞춘 방법론
- <http://agilemanifesto.org>
- 공통 핵심 원리
 - 소프트웨어 개발은 고객 중심이어야 하고 문서의 부담은 적어야 한다. 즉, 개발자가 집중해야 하는 것은 동작하는 소프트웨어를 만들어내는 것이다.

Agile Software Development

- **Agile의 정의**

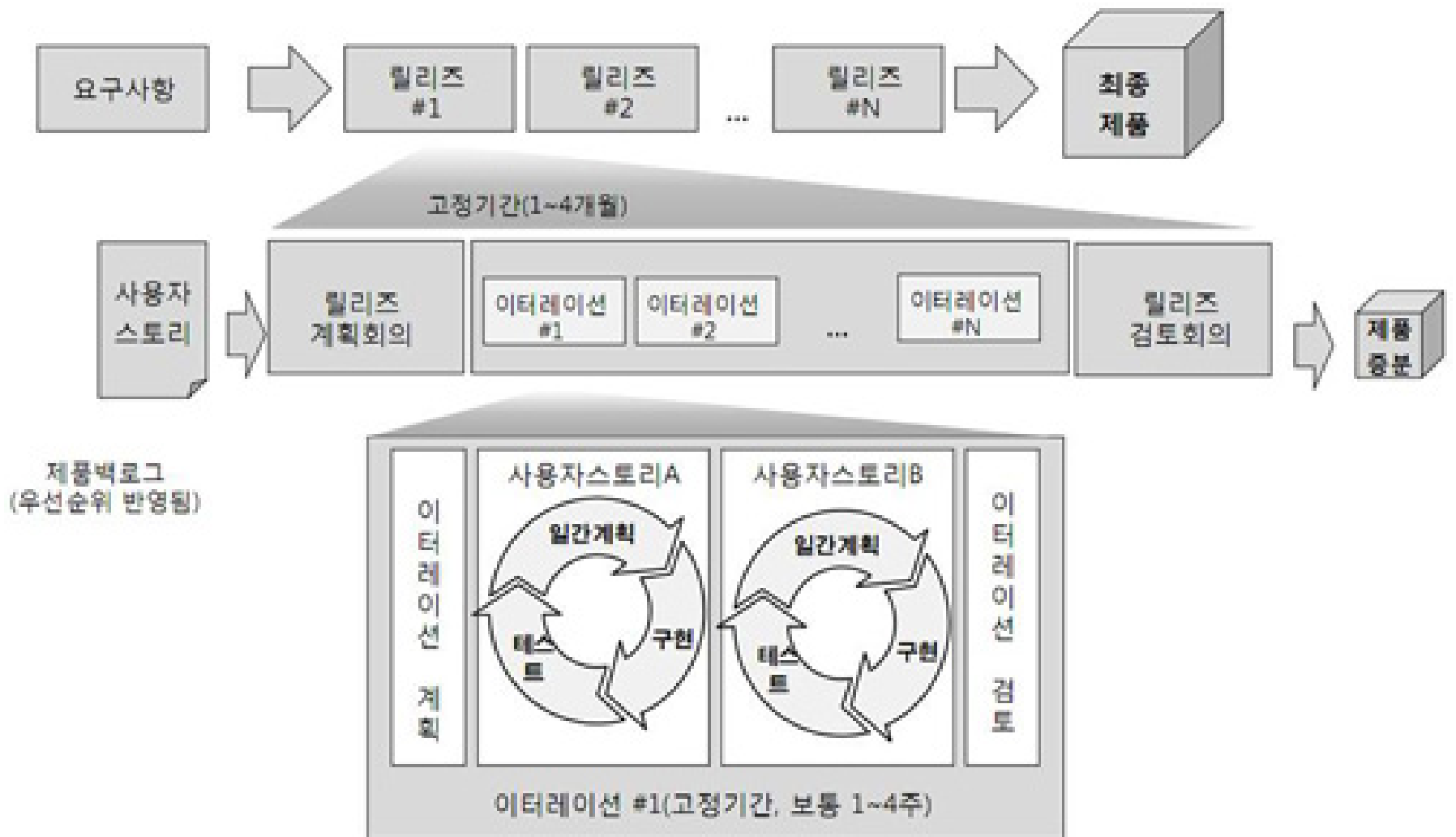
- 개발과정에서의 시스템의 변경사항을 유연하게 또는 기민하게 대응할 수 있는 방법론

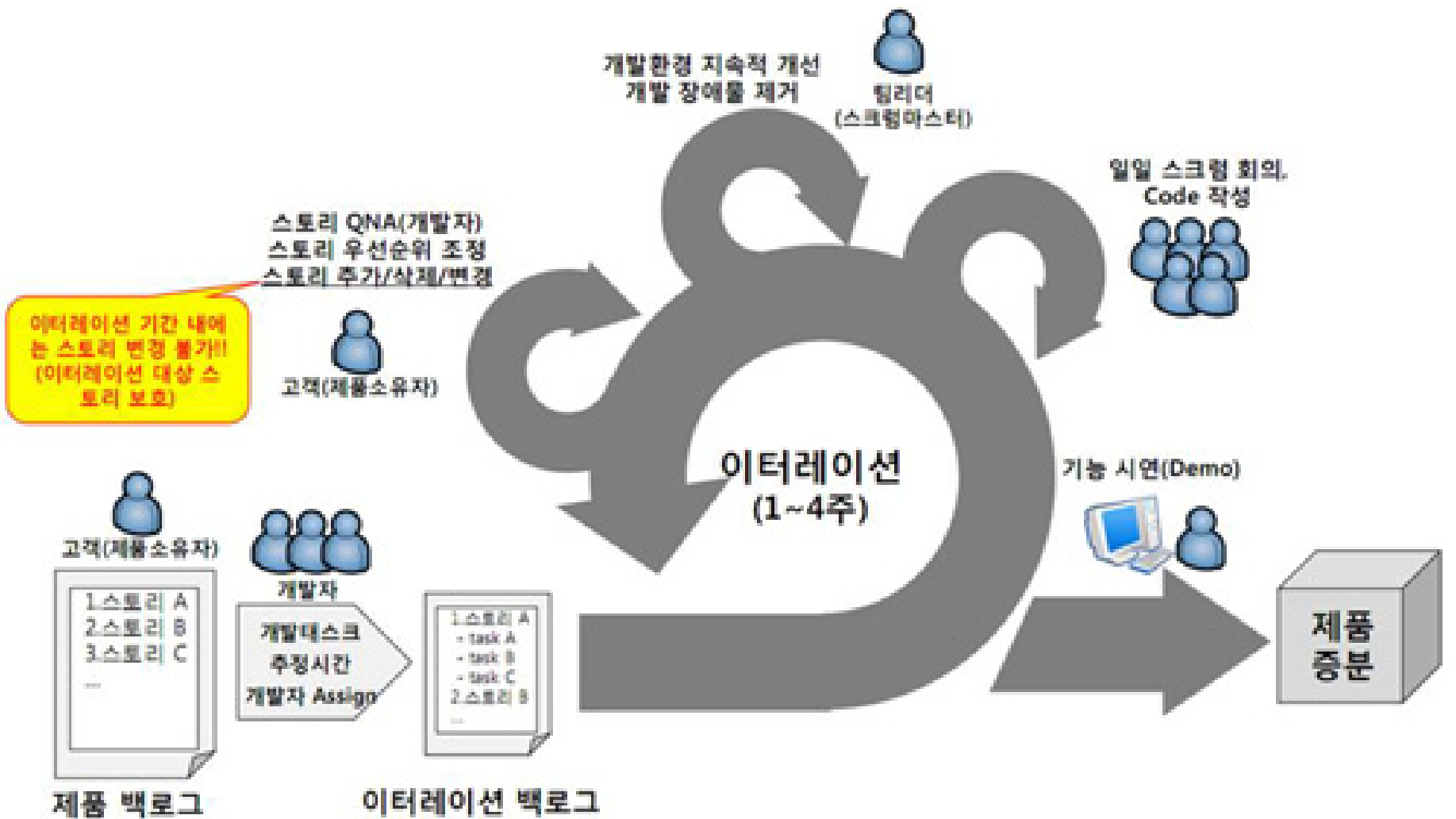
- **Agile의 등장 배경**

- 변화에 대한 대응력, 품질 유지, 비용관리의 장점을 유지하면서 시스템을 어떠한 방법으로 빠른 시간 내에 개발하느냐 하는 문제에 대해서 90년대 들어 새로운 방법론들이 나타나기 시작함.
- 2001년 XP(Extreme Programming), Crystal, ADS(Adaptive Software Development) 등 방법론 대표자들이 모여서 Agile Alliance 라는 연합체를 구성하면서 "Agile Development" 라는 공식 이름 발표

Agile 지지자의 권고 사항

- 우리는 가치 있는 소프트웨어를 일찍 자주 산출함으로써 고객을 만족시키는 것을 최우선시 한다.
- 2주에서 두 달 간격으로 작동하는 소프트웨어를 빈번하게 산출한다. 간격은 짧을 수록 좋다.
- 제대로 실행되는 소프트웨어는 1차적인 도구이다.
- 개발이 다소 늦어지더라도 요구 사항의 변화를 환영하라. Agile 프로세스는 고객의 경쟁 우위를 위해 변화를 이용한다.
- 비즈니스 인력과 개발자는 프로젝트 기간 동안 날마다 함께 일해야 한다.
- 동기 부여가 된 개인을 중심으로 프로젝트를 구축하라. 그들이 필요로 하는 지원과 환경을 제공하고, 그들이 일하는 것에 대해 신뢰를 부여하라.
- 개발 팀 내에서 그리고 개발 팀에게 정보를 전달하는 가장 효율적이고 효과적인 방법은 직접 대면하고 대화하는 것이다.
- 자발적으로 조직하는 팀에서 최상의 구조, 요구사항, 설계가 만들어진다.
- 기술적인 우수함과 훌륭한 설계 확장의 민첩성에 지속적으로 관심을 갖는다.
- Agile 프로세스는 지속적인 개발을 촉진한다. 스폰서, 개발자, 사용자는 끝까지 일정한 속도를 유지할 수 있어야 한다.
- 단순함- 행해지지 않는 작업량을 최대화하는 기술-은 필수적이다.
- 정기적으로 팀이 어떻게 효과적으로 운영될 수 있는지를 고려해 보고 이에 따라 행동을 조율하고 수정해야 한다.





2.2 Modeling?

SW개발의 주요작업들

1. 요구사항 정의
2. 요구사항 내용분석
3. 분석결과를 토대로 설계
4. 설계결과를 토대로 프로그래밍
5. 테스트

기존에는 "4."를 최대중요시

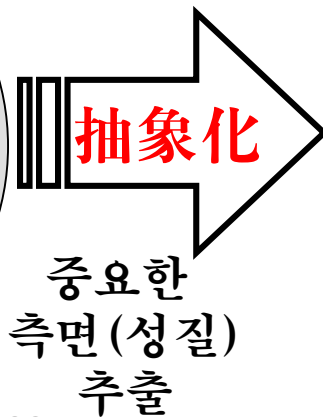
[1~3의 사고방식/표현방법]과 [4의 사고방식/표현방법]
사이에 커다란 "갭"이 존재

→ 1~5의 작업들간의 연계가 부적절
현실세계와 SW세계 사이의 원만한 연계가 부족

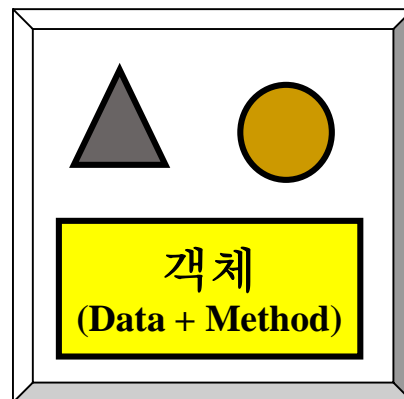
OO Modeling

- 객체를 사용하여 모델링 수행
 - SW개발의 각 작업에서 “**공통적인 사고방식**”을 사용 → 가능한 한 “**갭**”이 발생되지 않도록
 - “공통적인 사고방식” = “**객체**”

현실세계



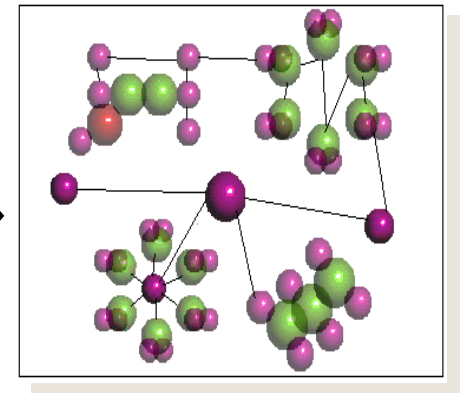
소프트웨어개발



객체지향
분석/설계/프로그래밍



컴퓨터세계

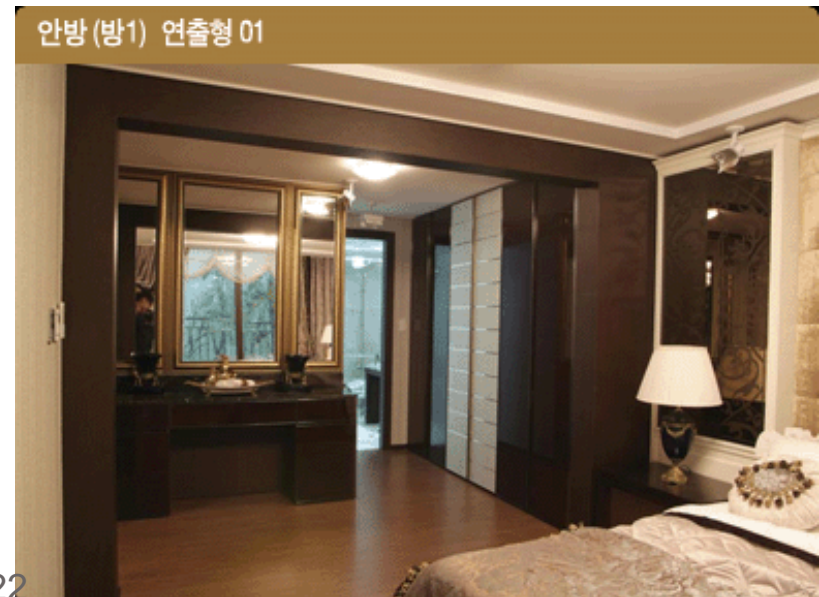


객체들간의
메시지授受作用에 의해
요구된 기능 실현

Modeling

- Model
 - 문제대상을 특정한 방향/관점에 맞추어, 이용자들이 이해하기 쉽도록 기술한 것.
 - 예: 플라스틱모델, 지도, CG표현물,...
 - 추상화(Abstraction)
 - **Abstraction** is a fundamental human capability, it let us **filter out nonessential details** about a complex problem or structure.

Some Examples of Model



Some Examples of Model



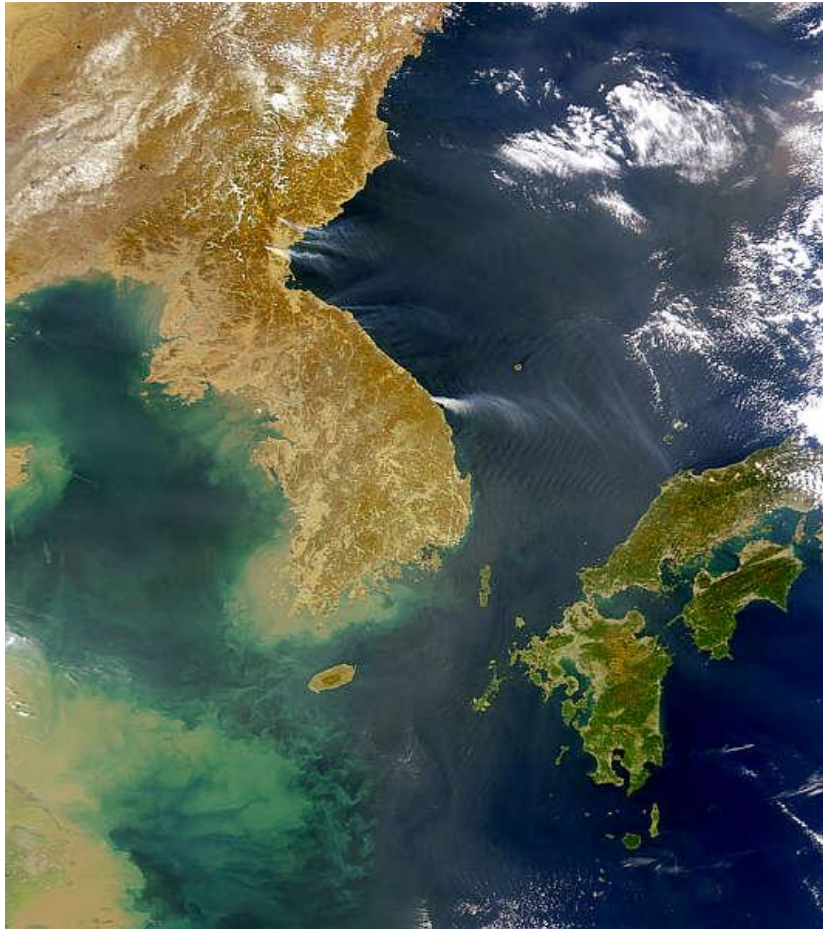
Some Examples of Model



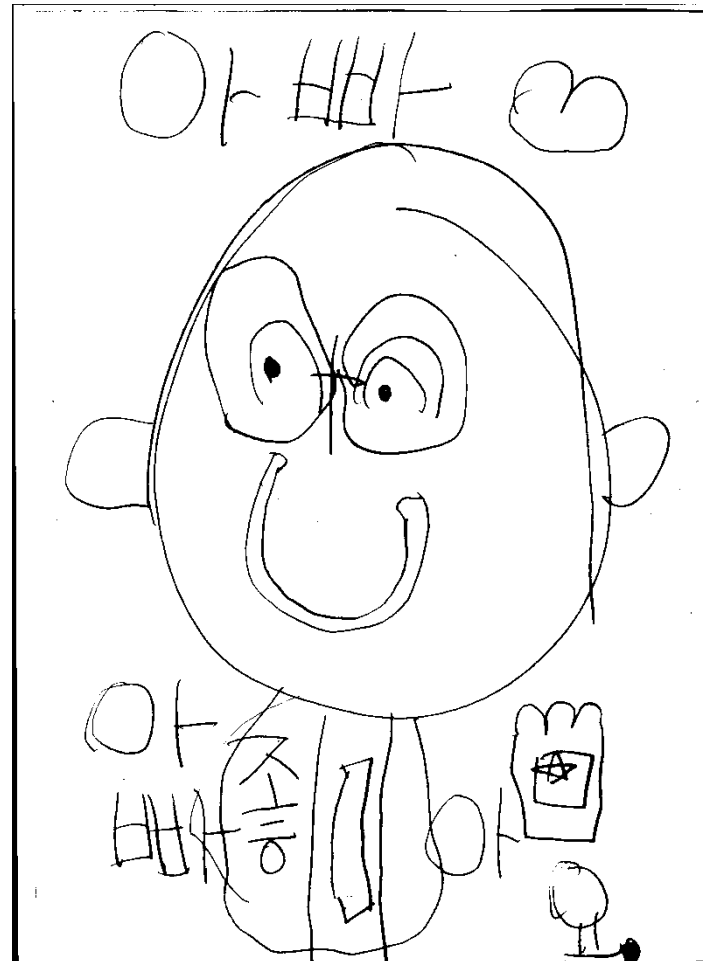
Modeling

- Model
 - 문제대상을 특정한 방향/관점에 맞추어, 이용자들이 이해하기 쉽도록 기술한 것.
 - 예: 플라스틱모델, 지도, CG표현물,...
- 추상화(Abstraction)
 - **Abstraction** is a fundamental human capability, it let us **filter out nonessential details** about a complex problem or structure.

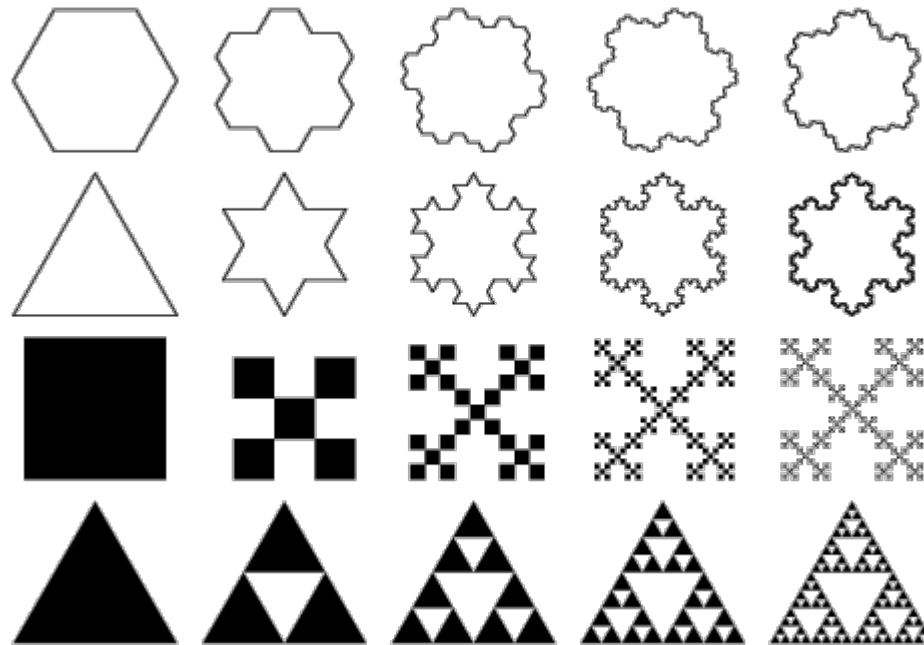
An Example of Abstraction(1)



An Example of Abstraction(2)

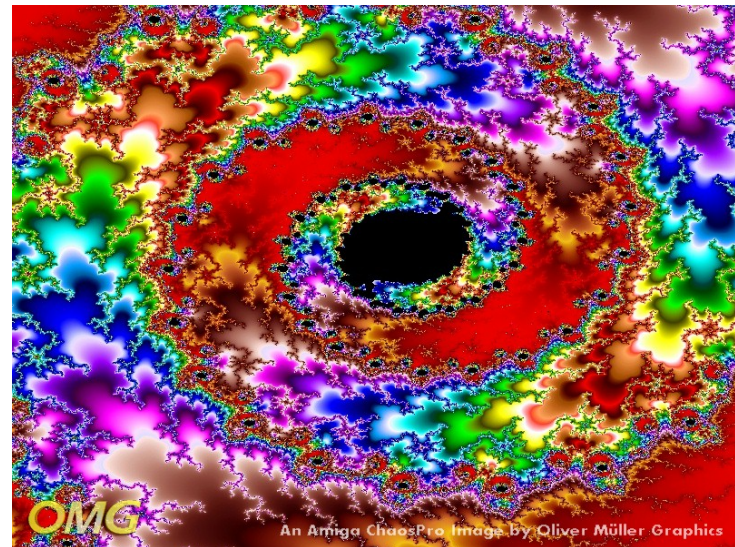
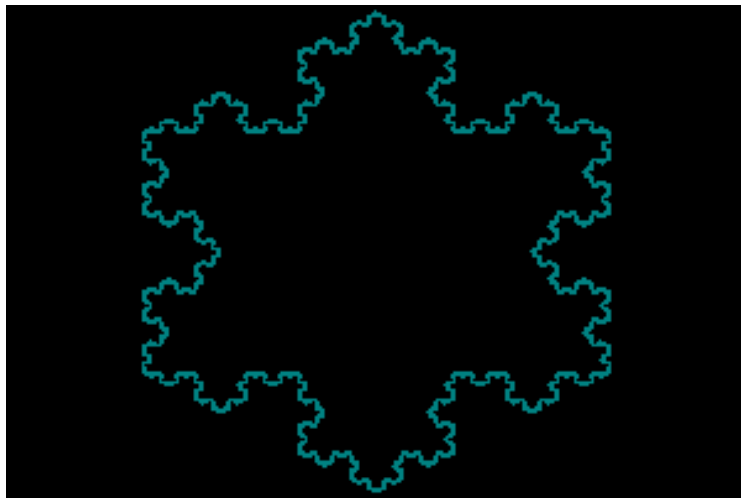


2.3 FUM(Fractal UML Modeling)



Fractal

- 도형의 일부분을 어디까지 확대하더라도 그 내부에 동종의 복잡한 구조가 내장되어 있는 도형



Fractal 도형 만들기 vs. FUM

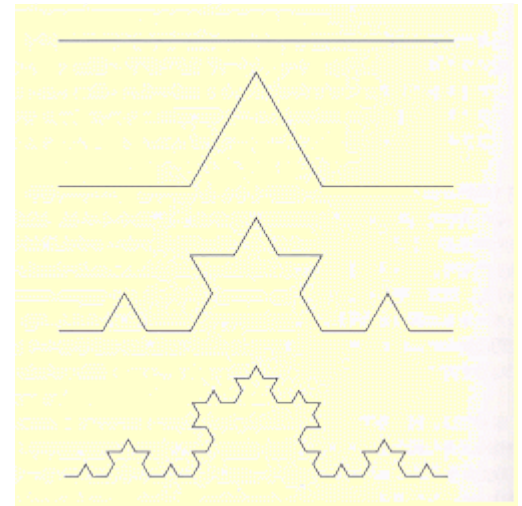
“선분을 3등분하여,
그중에서 가운데 것을 집어올린다.”

위 규칙을 반복.

현실세계는 복잡&변화무쌍.

그러나, 어떤 규칙이 존재.

이런 규칙 찾아내면 매우 아름다운 형태 발견할 수 있어.



단순 직선을 조금씩 수없이 반복 변형시켜나가듯,

현실세계의 문제로부터 반복변형시켜서 SW로서 가동시킬 수 있는 정도의 정밀도를
갖출때까지 반복....반복...

→ FUM(Fractal UML Modeling)

Modeling(cont.)

- SW Modeling
 - 개발대상에 대하여, 다양한 관점에서 모델을 만들어서 조합/변형, 최종적으로 프로그램 작성.
 - 이 책에서는 4단계 Modeling을 사용
 - 요구모델링
 - 개념모델링
 - 사양모델링
 - 구현모델링

UML의 역사

Years	version	특징
'89~'94	방법론 전쟁시대	50여개 이상의 OOAD방법론 亂立
1994	UM(Unified Method) 0.8	Booch기법 + OMT기법(G. Booch , J. Rumbaugh)
1996. 6	UML 0.9	OOSE기법(Ivar Jacobson)과 통합
1996.10	UML 0.91	
1997. 1	UML 1.0	주요 OO기술 전문업체 가세, OMG제안
1997. 11	UML 1.1	OMG 표준화
2000. 2	UML 1.3	
2001. 9	UML 1.4	
2003. 3	UML 1.5	
2005. 7	UML 2.0	컴포넌트기술 접목, 임베디드SW개발에 적용가능
2007. 10	UML2.1.2	
2009. 1	UML2.2	UML profile제공 → 특정용도에 맞게 자유롭게 확장가능
2010. 5	UML2.3	
2011. 7	UML 2.4.1	...
2017. 12	UML 2.5.1	https://www.omg.org/spec/UML/About-UML/

UML



- OMG표준
- 대다수 SW개발현장에서 UML을 사용
- UML은 “모델 표기법”
 - 모델링기법, SW개발공정이 아님!
- UML에 의해 작성된 도면
 - 커뮤니케이션 도구



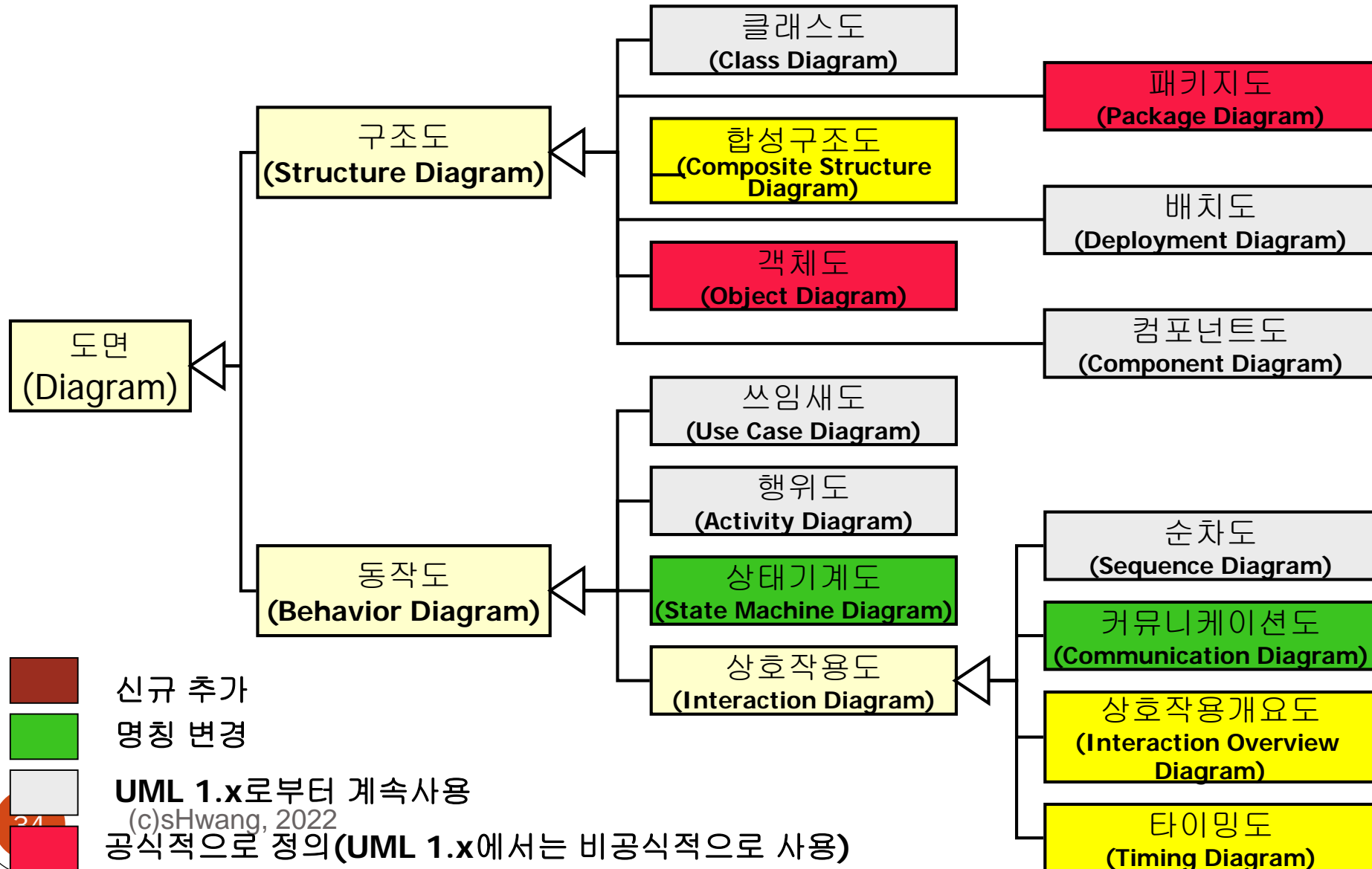
UML의 적용

- SW개발시의 설계도면 작성
- 실세계에 존재하는 모든 사물/개념을 표현가능
 - MOF(Meta-Object Facility)

UML 1.x의 도면들

명칭	역할
Use Case Diagram	시스템과 사용자사이의 수수작용(쓰임새)을 표현
Sequence Diagram	객체들사이의 상호수수작용을 시간적 관점에서 표현
Collaboration Diagram	객체들사이의 상호수수작용을 2차원평면에 표현
Class Diagram	클래스와 클래스들 사이의 관계를 표현
Object Diagram	객체와 객체들사이의 관계를 표현
StateChart Diagram	객체의 상태변화를 표현
Activity Diagram	객체들의 행위변화를 표현
Component Diagram	소프트웨어의 내부구성(컴포넌트와 컴포넌트들 사이의 관계 등)을 표현
Deployment Diagram	시스템동작시의 구성(객체/컴포넌트의 물리적 배치상황)을 표현

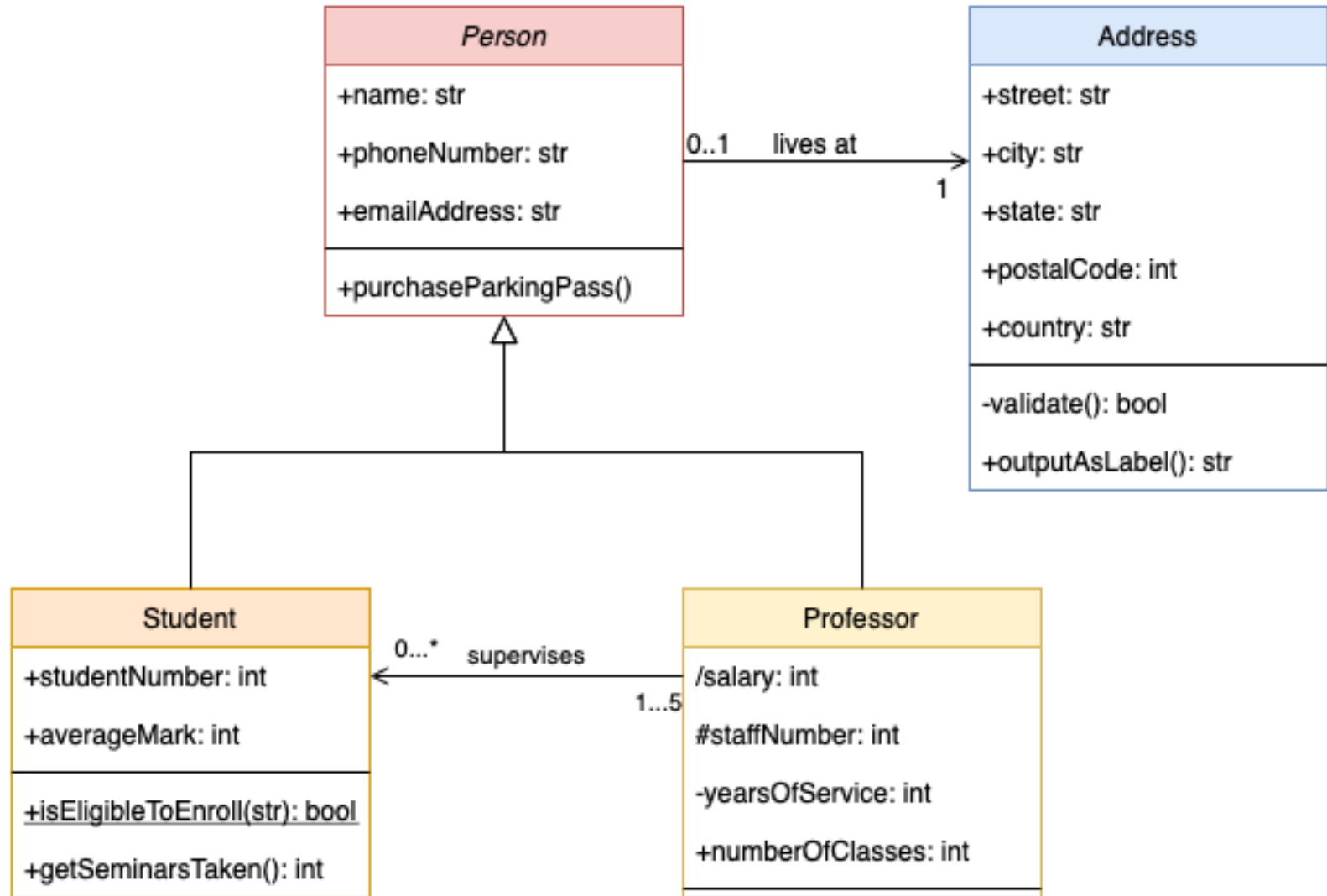
UML diagrams(ver. 2.X)



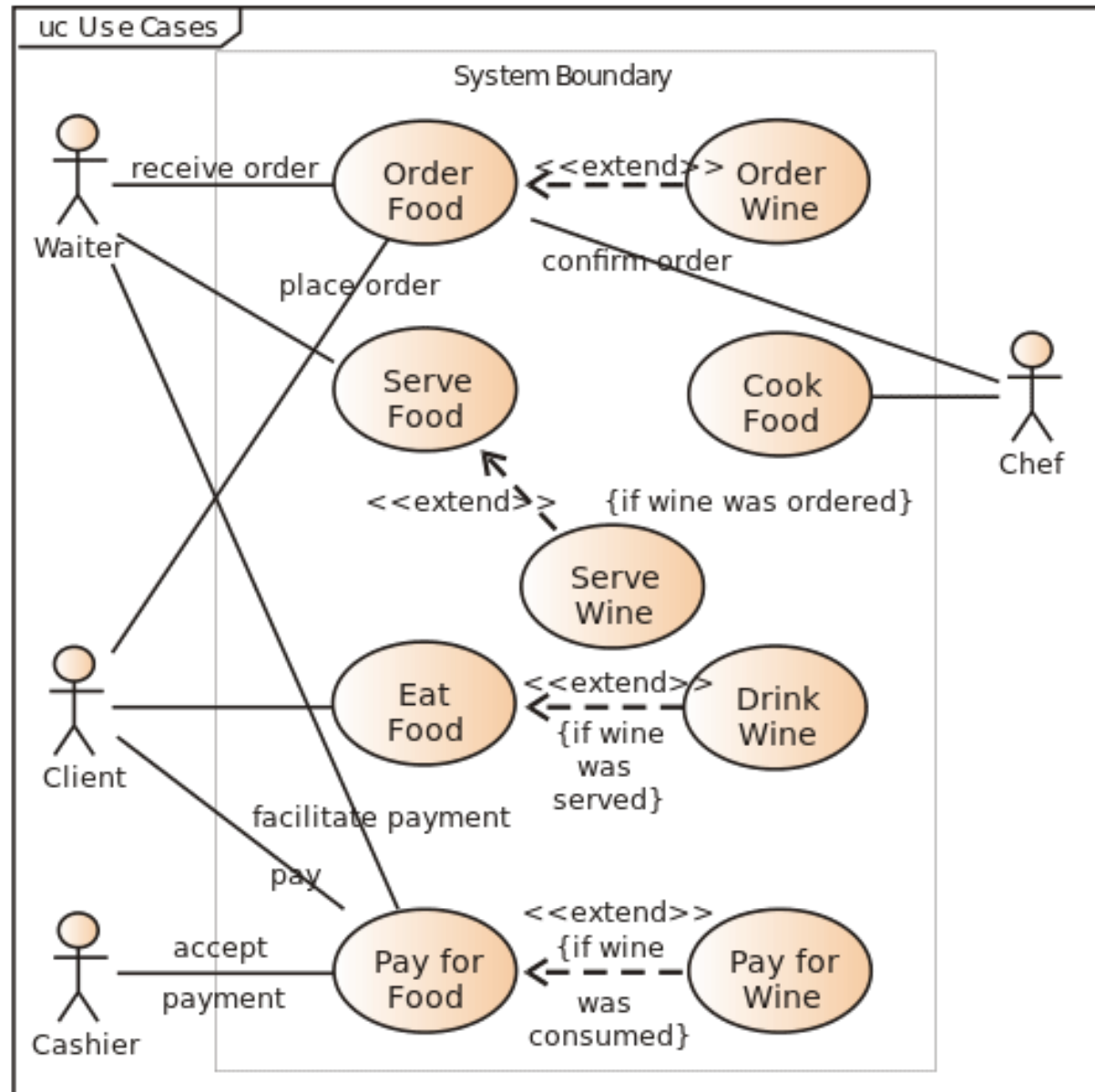
UML 2.X의 Diagrams

Diagram	주요 특징
Object Dgm	어떤 시점에서 시스템 내의 객체들간의 정적인 관계를 표현
Class Dgm	클래스들간의 관계를 토대로, 시스템의 정적인 구조를 표현
Composite Structure Dgm	클래스와 컴포넌트 등의 내부구조를 표현
Component Dgm	컴포넌트(소프트웨어부품)들간의 정적인 구조를 표현
Package Dgm	패키지들간의 의존관계 및 내포관계 등을 표현
Depolyment Dgm	시스템의 물리적인 구성 및 배치를 표현
Activity Dgm	업무의 흐름 및 처리실행순서 등을 표현
UseCase Dgm	유저의 관점에서 바라본 시스템의 기능을 표현
Sequence Dgm	객체들사이의 상호수수작용(메시지송수신)을 시간흐름에 따라서 표현
Communication Dgm	객체들사이의 관계를 토대로 상호수수작용을 표현
StateMachine Dgm	시스템을 구성하는 객체들의 상태전이 및 이에 대응하는 동작을 표현
Timing Dgm	객체가 시간축에 따라서 어떻게 상태전이를 수행하는지를 표현
Interaction Overview Dgm	Interaction Dgm들(Seq. D & Com. D)이 서로 어떤 관계를 가지고 있는지를 표현

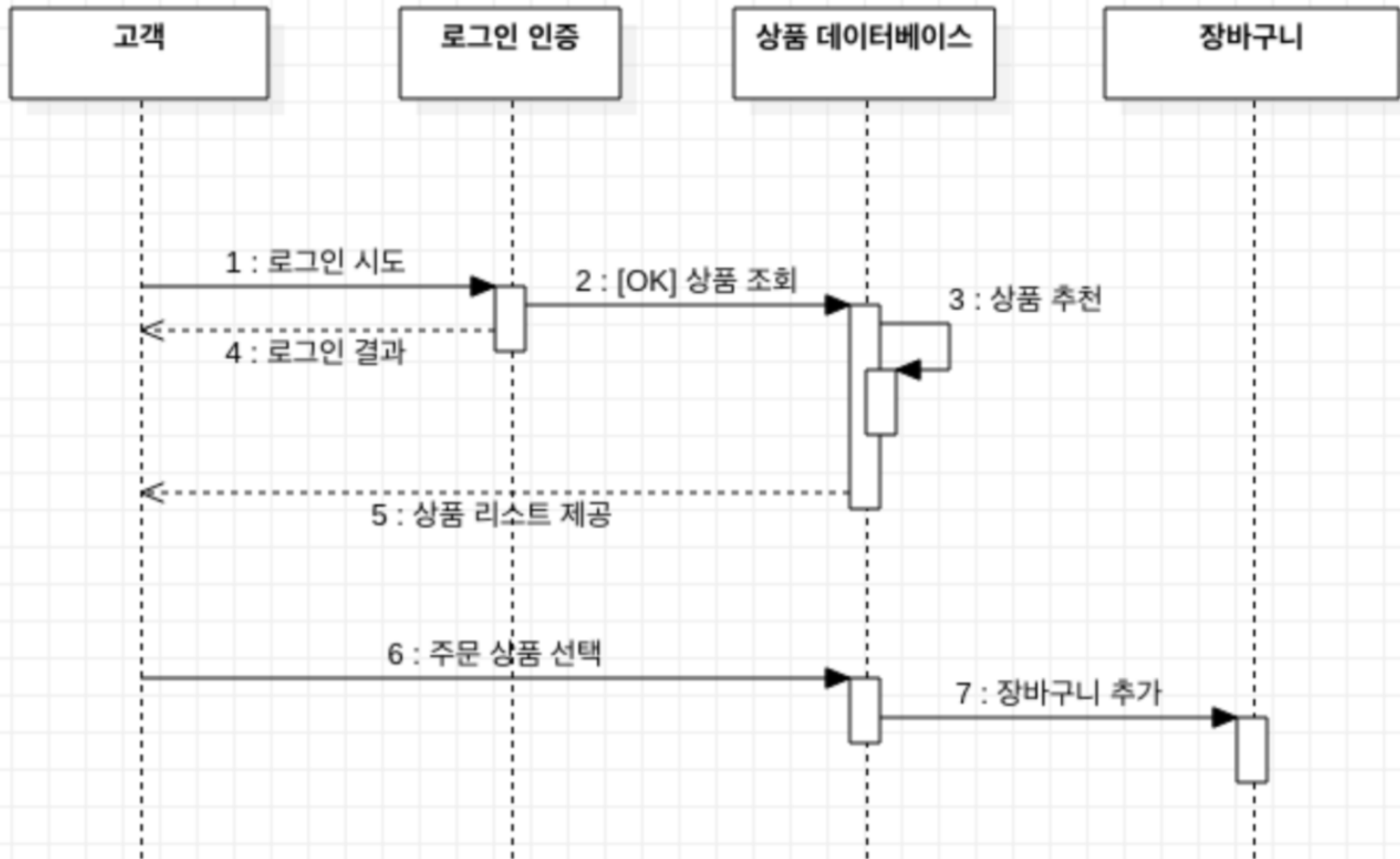
Class Diagram



Use Case Diagram



Sequence Diagram



선문대 중앙도서관



선문대 중앙도서관

1. 도서관 2층 도서검색PC앞에 간다.
2. 대출희망 도서를 검색한다.
3. 서가에 가서 책을 찾아낸 후에 대출창구로 간다.
4. 대출자가 신분증(학생증, 신분증)을 도서관직원에게 제출한다.
5. 도서관직원은 대출자의 정보(학생증, 신분증)와 대출도서정보를 컴퓨터에 기록한다.
6. 컴퓨터가 대출자와 도서에 대하여 검사한다.
7. 도서관직원이 대출자에게 책을 전달한다.
8. 대출창구를 떠난다.

선문대 중앙도서관



1.4 연습문제

- A시립도서관의 정보관리시스템



A시립도서관의 정보관리시스템



- 현재 : 비전산화 상태
 - A시립도서관에서는, 아직 도서관직원이 카운터에서 서적 대출목록을 대조하면서 대출업무 수행
- 우선, “수장자료 대출업무의 전산화”가 목표
 - 카운터에서 도서관 직원이 단말기 이용하여 대출업무 수행
 - 추후에는, 이용자가 단말기 조작하여 원격예약도 가능하도록
 - 대출에 관한 규칙 : 도중에 변경될 수도...
 - 한번에 대출가능한 수량 : 서적(10권이하), 음악 및 영상물(5개이하)
 - 대출기간 : 2주이내
 - 수장자료의 등록 및 이용자 등록도 필요