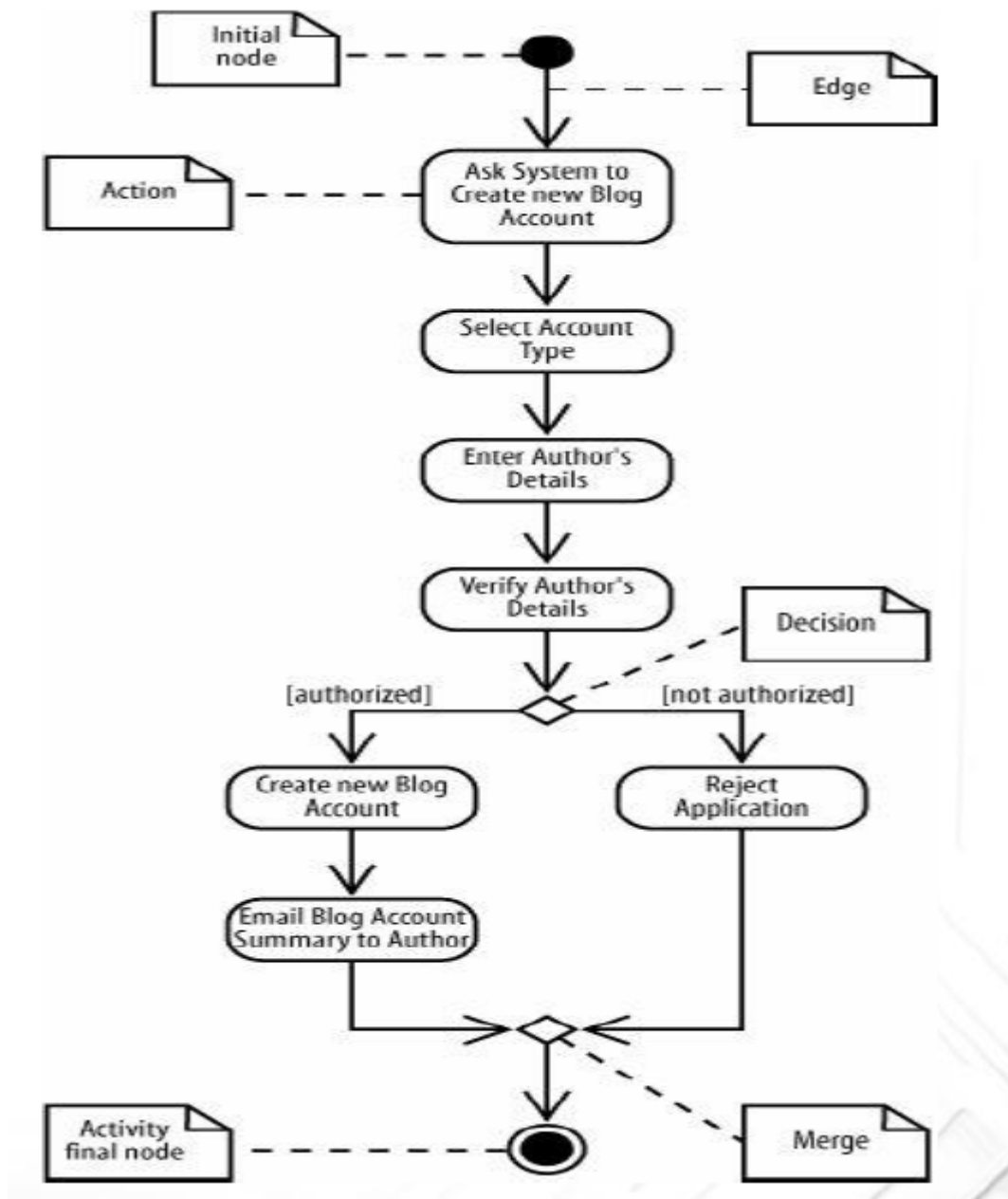


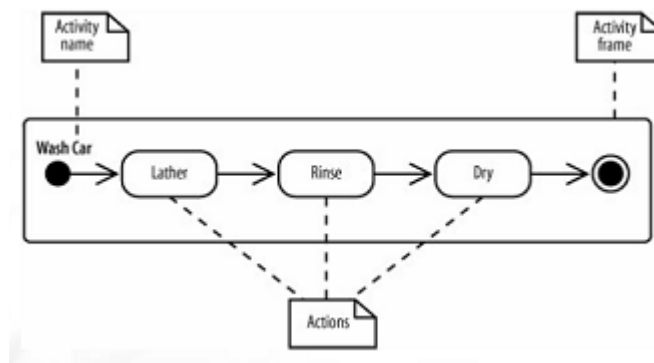
# Activity Diagram

- 시스템의 Workflow를 모델링하는 도구
  - 시스템이 목적을 달성하는 방법을 나타냄
  - 시스템에서 발생하는 프로세스의 흐름을 보여줌
  - 비즈니스 프로세스를 표현하는데 좋다



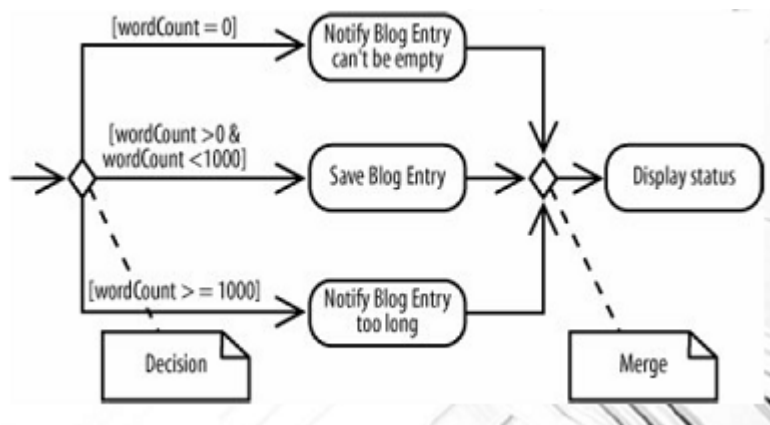
## Action & Activity

- Action : 프로세스 완료에 있어서의 Active Steps
- Activity : 모델링 하고 있는 프로세스



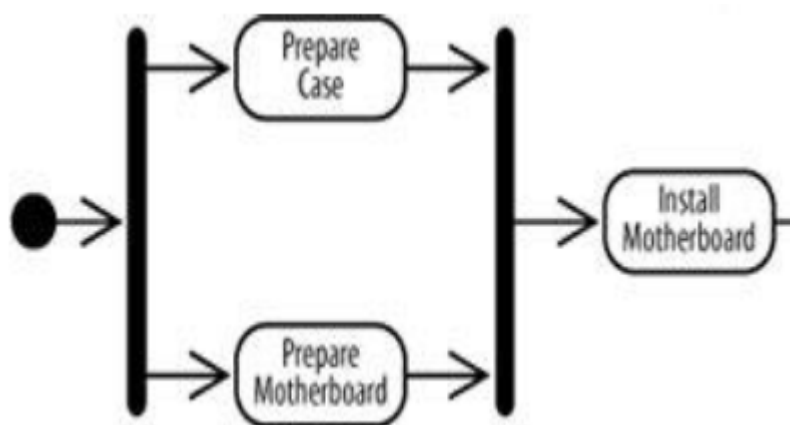
## Decision & Merge

- Decision : 조건에 따라서 다른 행동으로 이어질 때
- Merge : 조건문의 끝을 알림(다시 흐름 통합)



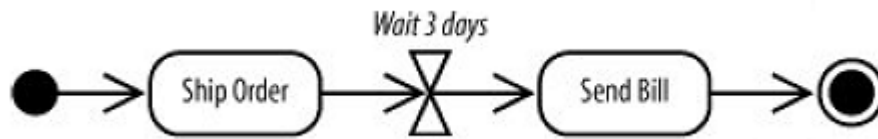
## Fork & Join

- Fork : 여러개의 Outgoing Flow, 동시에 시작되는 일련의 Actions
- Join : 여러개의 Incoming Flow, 계속 Activity를 진행하기 위해 끝나야 하는 액션들의 시점



## Time Events

- 모래시계 모양으로 그려짐, 기다려야 하는 시간 기재
- Incoming flow가 없는 Time event = 계속해서 반복되는 Time Event

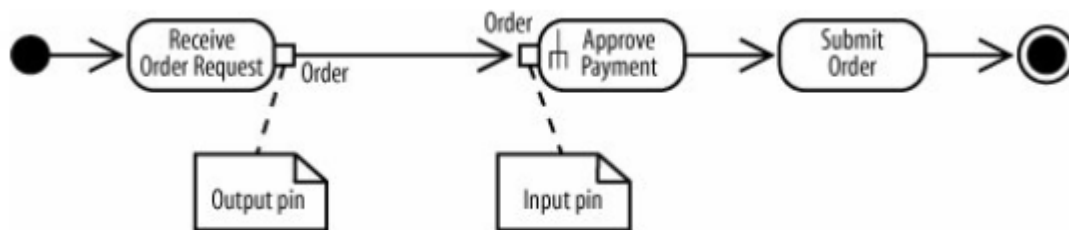


## 다른 Activity 호출



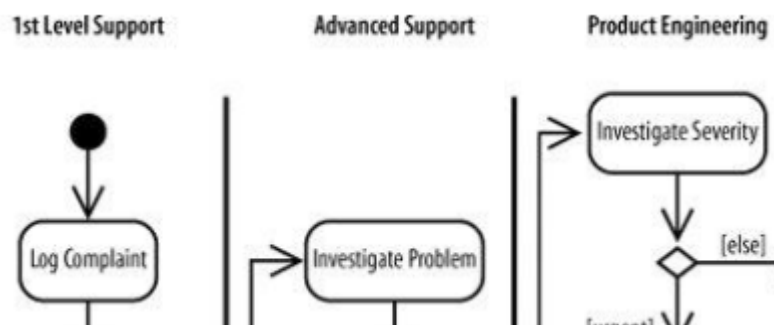
## Objects

- Activity의 특정 시점에서 가용한 객체(해당 객체가 생성, 사용, 삭제 되는걸 표현)
- Input pin : Action에 input됨
- Output pin : Action으로부터 Output됨



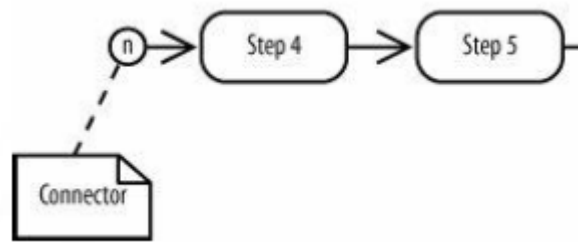
## Partitions

- 어떤 참가자가 액션의 책임자인지 표시

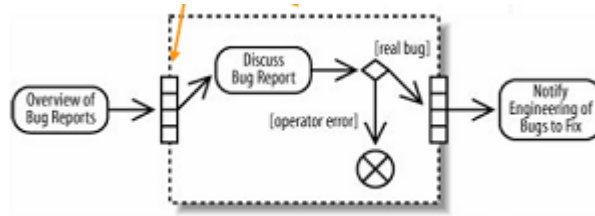


## 분할

- Connectors : 다이어그램을 풀때 사용
  - 라인 대신 Connect Edge
  - 동그라미로 그려짐



- Expansion Regions : 해당 지역의 액션들이 Input이 들어올때마다 이뤄진다는걸 표시



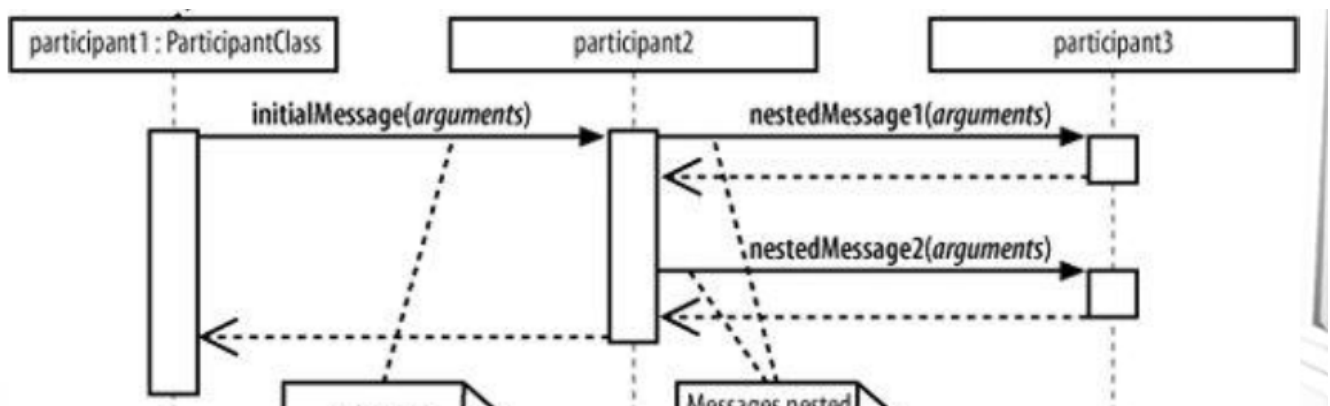
## Sequence Diagram

- 동적 표현하는 다이어그램
- 시스템 실행 시 생성되고 소멸되는 객체 표기
- 객체들 간에 주고받는 메시지를 표시
- 한 사건이 발생하였을 때 여러 객체간 관계를 시간 순서에 따라 정의(시간 중심)
- Use Case 내의 객체와 내포되어 있는 메시지를 표현
- Use Case Description의 흐름대로 시간에 따른 시나리오 표현
  - Parameter 들이 VO로 추상화

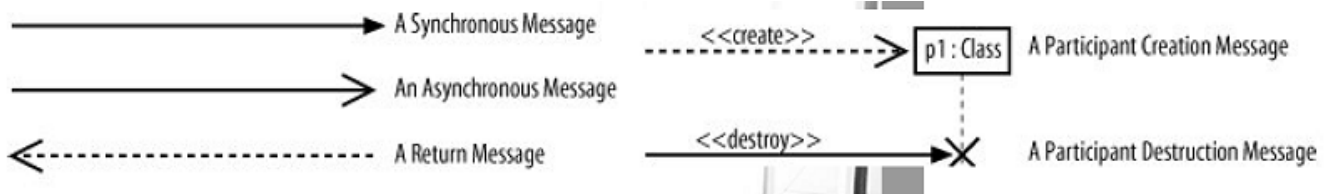
## Events, Signals, Messages

- Events : 시그널/메세지를 위한 Building Block
  - 시그널은 시스템 설계자가 주로 사용하는 용어
  - 메세지는 소프트웨어 설계자가 주로 사용하는 용어
- Signatures : attribute = signal\_or\_message\_name (arguments) : return\_type

## Nested Messages

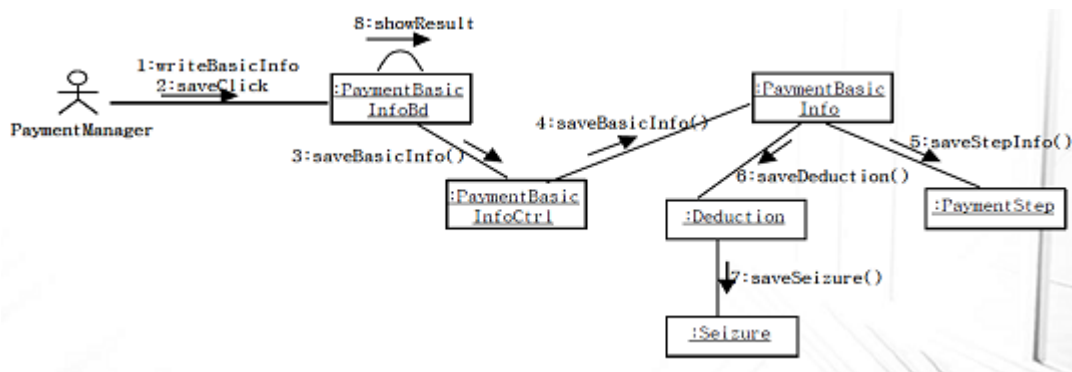


## Message Arrows



## Communication Diagram

- Collaboration Diagram의 다른 이름
- 객체와 객체들의 관계로 구성된 Communication을 보여주며, 상호 간에 전달되는 메세지도 보여줌
- 메세지를 주고받는 객체들을 Link로 연결하고, 메세지에 일련번호를 부여하여 표현
- 메세지를 주고받는 객체들의 구조적인 관계를 강조



## Sequence VS Communication

- 참가자들을 효율적으로 보여주는 것 : Communication
- 참가자들 사이의 링크를 보여주는 것 : Communication
- 메세지 시그니처 보여주기 : Draw
- 평행적 메세지 보여주기 : Draw
- 비동기적 메세지 표현 : Sequence
- 읽기 쉬운 메세지 오더링 : Sequence
- 다이어그램 만들기/유지 : Communication