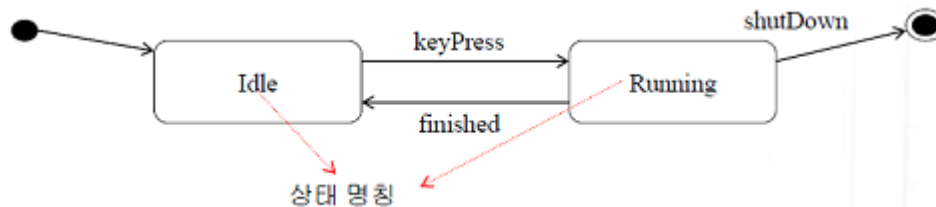


# State Machine Diagram

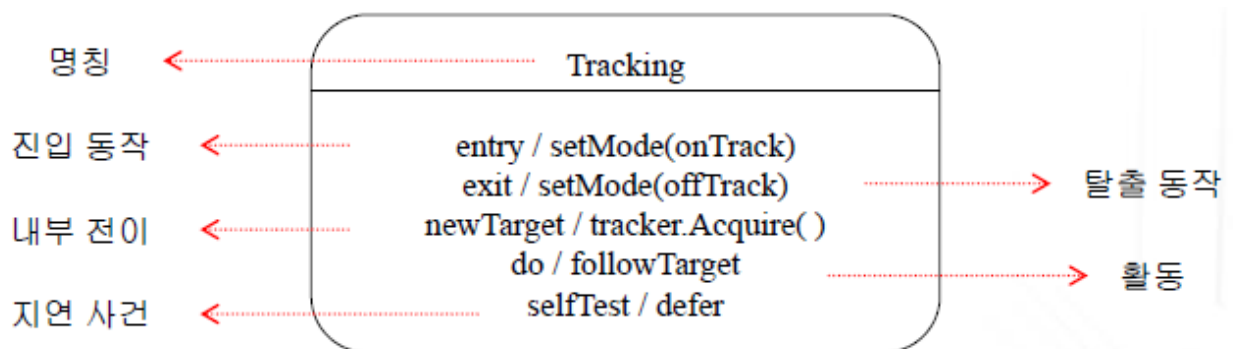
- 객체의 상태를 모델링
- Discrete Behaviours들을 FSM 형태로 표현
- Real Time / Mission Critical System / 특정목적을 가진 장치

## State

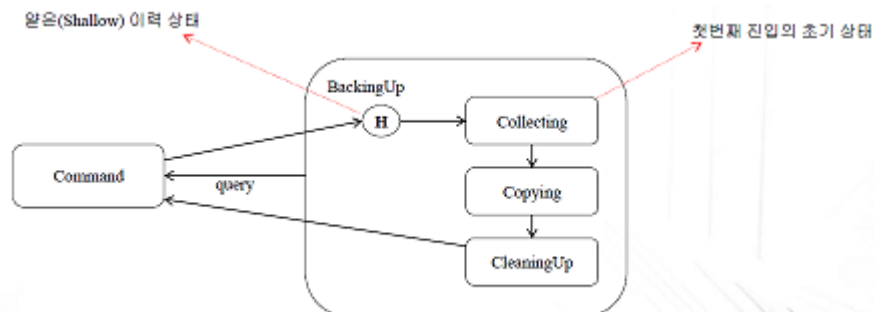
- 객체 생명 주기 동안에 가질 수 있는 특정 조건, 해당 기간동안 객체는 조건이 만족된 상태에서 활동을 수행
- 구성
  - 명칭 : 특정 상태와 다른 상태를 구분 짓는 문자열로 구성된 이름
  - 진입/탈출 동작 : 상태에 진입/탈출 때 수행되는 동작
  - 내부전이 : 상태 변경 없는 상태 내의 전이
- 초기 상태 / 종료 상태



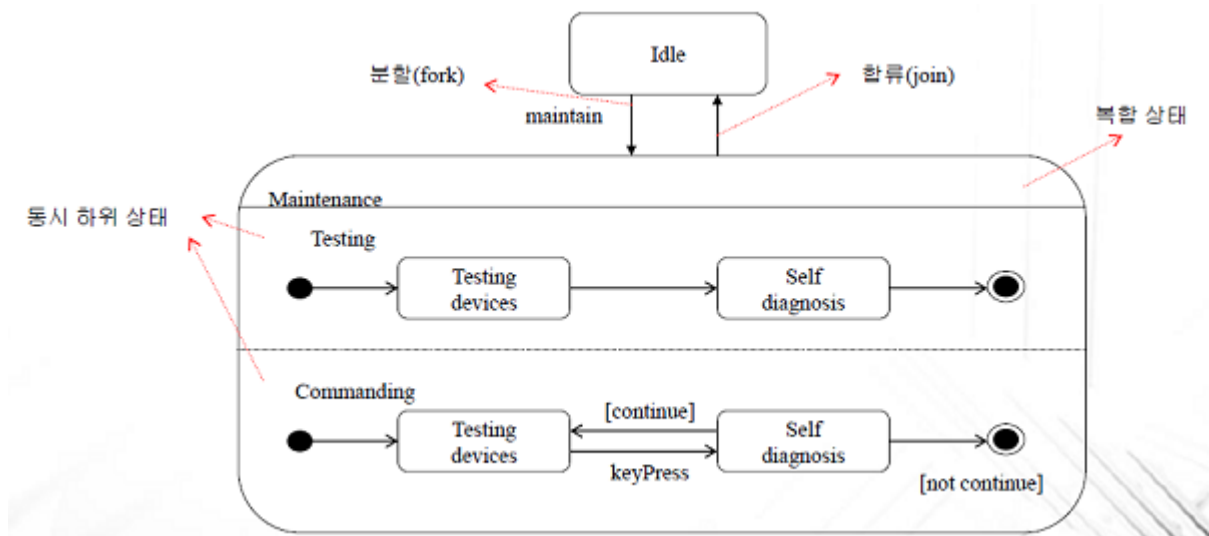
- 내부 표현



- 하위상태 표현 가능
- 이력 상태 : 객체가 복합 상태를 떠날 때 활동한 마지막 하위 상태를 기억하고 있는 객체 모델을 만들 때 사용



- 동시 상태 : 객체가 동일 상태에서 동시에 실행되는 하위 상태가 두 개 이상 나타날 때 사용



## Transition(전이)

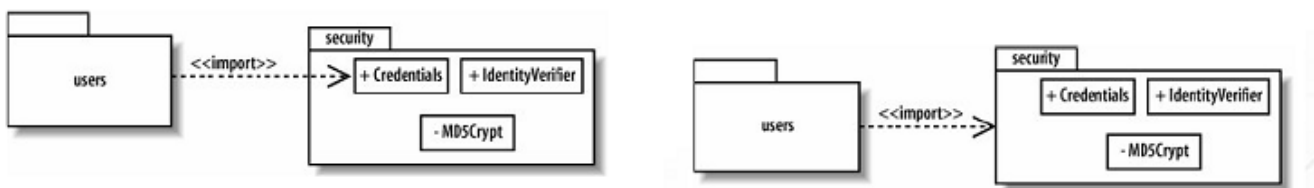
- 특정 객체가 현 상태에서 어떤 동작을 수행한 후 지정된 조건이 만족되어 다음 상태로 들어가는 두 상태간의 관계

## Package Diagram

- 그룹핑 구성체
  - 상위 레벨 단위로 그룹핑
  - 관련된 클래스들을 구조화
  - 패키지가 패키지를 포함 가능
- 용도 : 대형 시스템의 주요 요소들 간의 의존성 관계 표현, 어플리케이션의 의존성 제어

## 이름 영역

- 모든 클래스는 포함 된 패키지 내에서 **유일한 이름** 보유
- 완전한 경로명

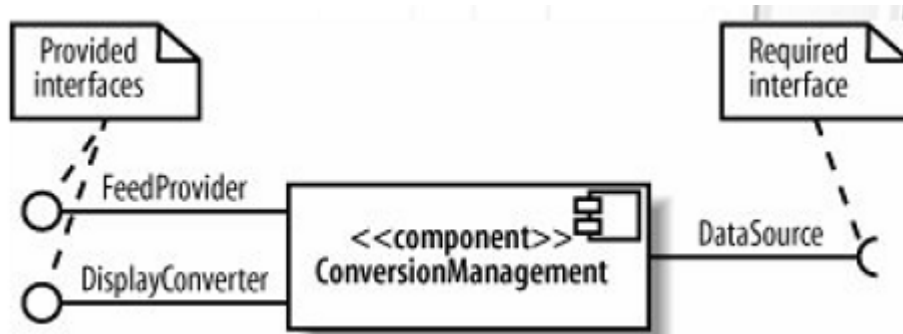


## Component Diagram

- 시스템의 부품을 관리하고 재사용
  - Component** : 캡슐화되고, 재사용 가능하고, 교체 가능한 S/W의 모듈화
  - 직사각형에 **<<Component>>** 표시
- Provided Interface** : 컴포넌트가 Realize하는 인터페이스
- Required Interface** : 컴포넌트가 동작하기 위해 필요로 하는 인터페이스

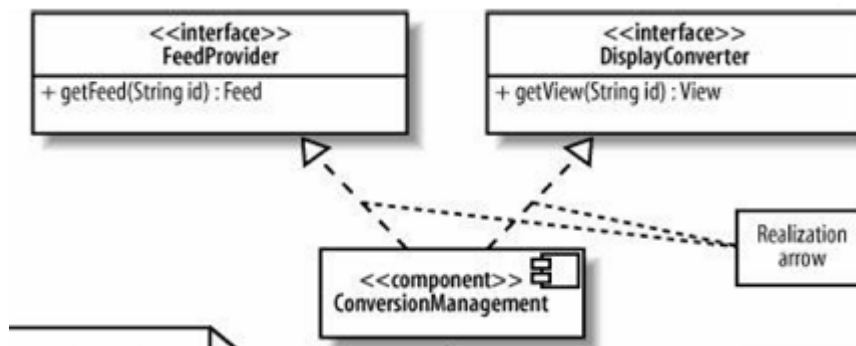


## Ball and Socket Notation

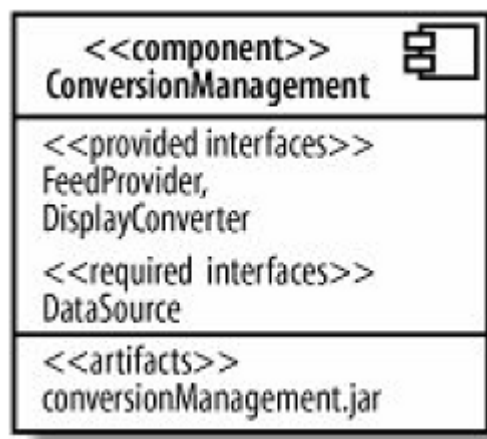


## Stereotype Notation

- 인터페이스의 오퍼레이션을 나타낼 때 사용

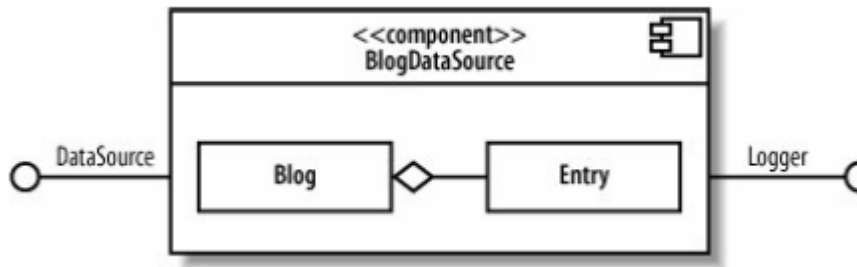


## Listing Notation



## Component 구현 클래스

- 컴포넌트를 Realize하는 클래스들을 의존성과 함께 표현



## Views

- Black-Box View : 컴포넌트의 내부 구현에 대해 아무것도 작성하지 않음
- White-Box View : 어떤 클래스, 인터페이스, 컴포넌트들이 기능 동작을 도와주는지 표현

## Deployment Diagram

- 시스템의 물리적인 배치 레이아웃 표현
- 어떤 소프트웨어 부분이 어떤 하드웨어 상에서 실행되는지를 표현
- 통신 경로 : 어떻게 노드 간에 통신이 이루어지는지 표현, 사용되는 프로토콜 정보를 라벨로 표시

## Node and Condition

- 노드의 구성 : 노드를 패키지로 그룹화하여 구성, 노드간의 의존/일반화/연관 관계를 표현
- 연결 : 노드간의 관계를 표현하며 연관관계가 가장 일반적(물리적 연결을 표현)
- 노드 : 소프트웨어를 호스트 할 수 있는 것
- 디바이스 노드(하드웨어) / 실행 환경 노드(소프트웨어, 운영체제, 프로세스)로 구분
- 산출물 포함 : 소프트웨어의 물리적 표현, 파일, 나열하여 표시
- 꼬리표 값 : 노드/산출물에 추가 정보 표시
  - ex) 벤더, OS, 장소
- 시스템과 관련된 HW, NW, SW 제품 등 전반적인 요소들에 대한 내용을 표현

