



통합 구현

# 메시지 채널 패턴



한국기술교육대학교  
온라인평생교육원

## ◆ 학습내용 ◆

- 메시징 채널의 기본 개념 및 사용패턴
- Rabbit-MQ를 이용한 메시지 채널 실습

## ◆ 학습목표 ◆

- 메시징 채널의 기본 개념을 이해하여 개발하고자 하는 응용 소프트웨어와 연계 대상간의 특성을 고려한 메시징 채널을 선택 할 수 있다.
- Rabbit-MQ를 이용하여 간단한 메시징 채널을 구현 할 수 있다.



## 메시징 채널

### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

##### (1) 메시지 채널의 정의

###### 메시지 채널

- 메시징 애플리케이션에서 수신자와 발신자를 연결하는 가상 파이프
- 메시징 시스템의 논리 주소로, 구현 방식에 따라 엔드포인트들은 직접 연결될 수도 있고 중앙 허브를 거쳐 연결



## 메시징 채널

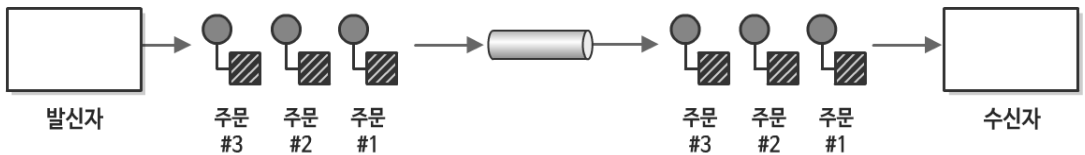
### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

##### (2) 사용패턴

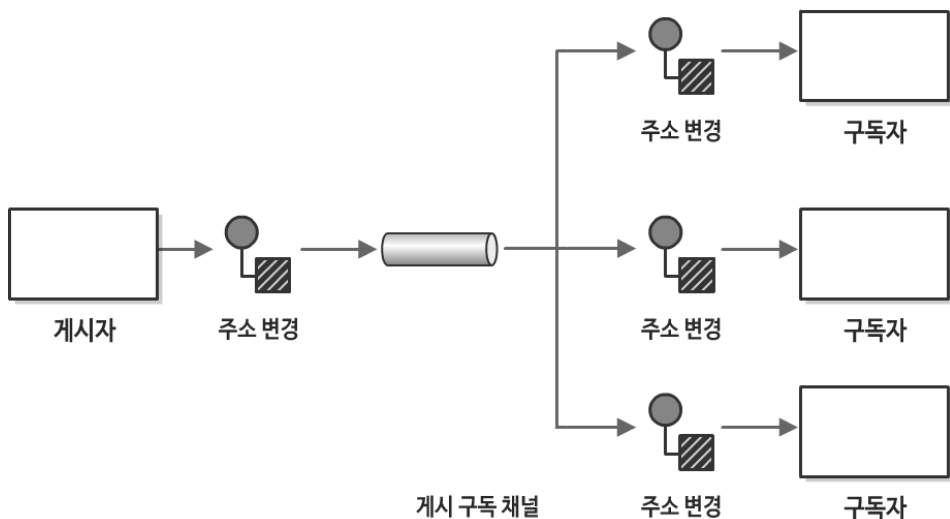
##### ① 1대1(point-to-point) 채널

- 1대 1 연결
- 하나의 수신자가 특정 메시지를 수신



##### ② 1대다(publish-subscribe) 채널

- 하나의 이벤트를 모든 관심 수신자에게 알리기 위해 사용하는 메시지 채널
- 모든 수신자는 이벤트에 대해 한번씩만 알림을 수신 받으며, 모든 수신자가 알림을 받을 때까지 채널에서 소비된 것으로 간주하지 않아야 함
- 기존 메시지 흐름을 방해하지 않고 메시지 채널을 엿들을 수 있어 디버깅 도구로 유용
- 보안 정책에 따라 제한되어야 함





## 메시징 채널

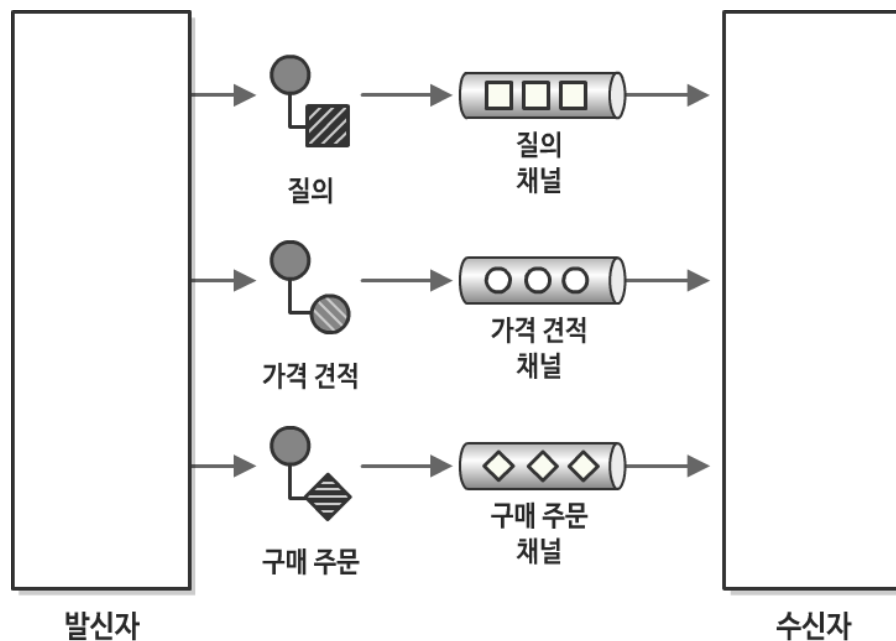
### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

##### (2) 사용패턴

##### ③ 데이터 형식 채널(Datatype channel)

- 메시지 형식
  - 데이터 구조(문자 배열, 바이트 배열, 직렬화 객체, XML 문서) + 데이터 포맷(바이트나 문자의 레코드 구조, 직렬화 객체의 클래스, XML 문서의 스키마 정의)
- 일반적인 경우
  - 발신자 측 : 메시지 헤더에 플래그(포맷 표시자)를 넣음
  - 수신자 측 : case문을 사용해 헤더 플래그 해석
- 데이터형식채널: 특정 채널을 지나는 메시지는 모두 동일한 형식의 데이터가 됨





## 메시징 채널

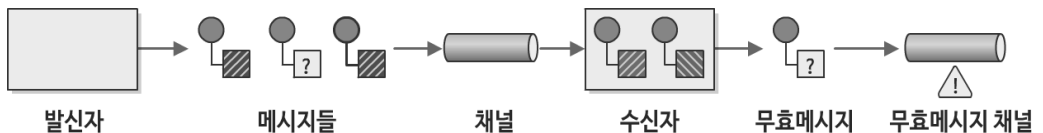
### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

##### (2) 사용패턴

##### ④ 무효 메시지 채널(Invalid Message Channel)

- 부적절한 데이터 형식이나 포맷을 가지는 등 유효하지 않은 수신된 메시지를 처리하는 방법
- 무효 메시지
  - 데이터 형식 채널이 허용하지 않은 형식의 메시지  
(ex 텍스트 메시지 형식 채널로 바이트 메시지 전송 등)
  - 수신자가 기대하는 헤더 필드 값(상관관계 식별자, 메시지 순서 식별자, 반환 주소 등)을 포함하지 않는 메시지
- 수신자가 무효 메시지를 직접 처리해야 함
  - 메시징 오류 로그 역할
  - 메시징 오류인지 애플리케이션 오류 인지 구분 (예, 잘못된 명령메시지)





## 메시징 채널

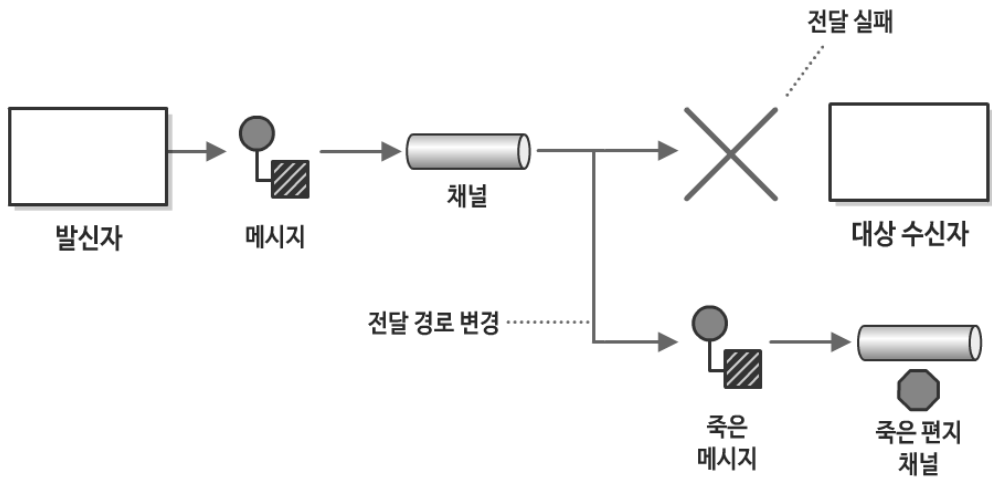
### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

##### (2) 사용패턴

##### ⑤ 죽은 편지 채널(Dead Letter Channel)

- 메시징 시스템이 제대로 전달할 수 없는 메시지 처리
- 메시지 채널 설정이 올바르지 않거나, 메시지 만료가 되는 경우 등 전달될 수 없는 경우
- 메시징 시스템에서 죽은 메시지의 결정과 처리를 자동으로 수행





## 메시징 채널

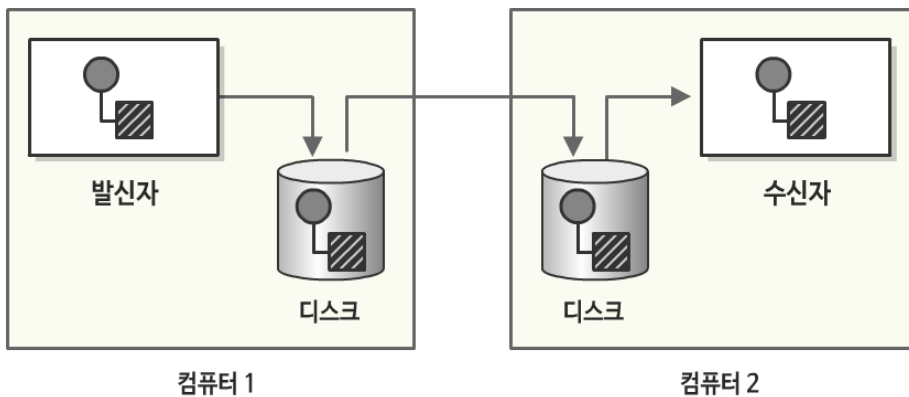
### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

##### (2) 사용패턴

###### ⑥ 보장 전송(Guaranteed delivery)

- 기본적으로 메시징 시스템은 메시지 전달이 가능할 때까지 메시지를 메모리에 저장
  - 이 경우 전원이 나가거나, 프로세스가 중단되면 메모리에 저장된 메시지는 사라지게 됨
- 메시지 전달을 보장하기 위해 내장 데이터 저장소 사용
- 발신된 메시지는 수신자가 수신할 때까지 컴퓨터 디스크에 최소 한 번은 저장됨
  - 디스크 사용을 줄이기 위하여 유지 제한 시간 설정
- 영속성, 신뢰성 증가
- 성능 감소
- 테스트와 디버깅 단계에서는 보장 전송 해제하는 것이 바람직함







## 메시징 채널

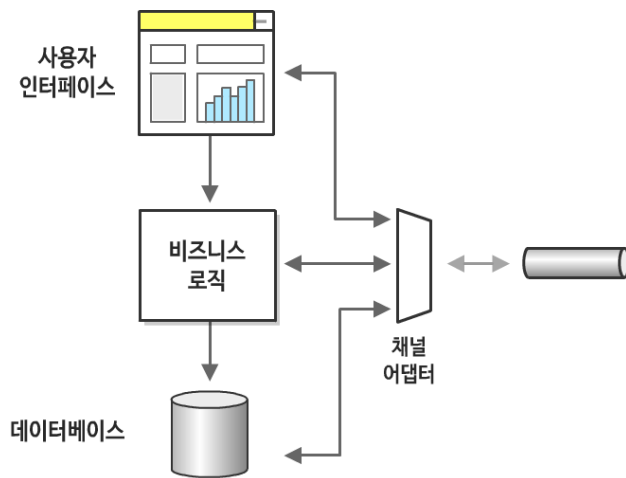
### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

##### (2) 사용패턴

###### ⑦ 채널 어댑터(Channel Adapter)

- 메시징 시스템의 메시징 클라이언트로서 애플리케이션 기능을 호출하고 애플리케이션으로부터 이벤트를 수신해 메시징 시스템을 호출
- 애플리케이션 아키텍처와 메시징 시스템이 접근하는 데이터 형식에 따라 애플리케이션의 여러 계층에 연결될 수 있음



애플리케이션의 여러 계층에 연결된 채널 어댑터



## 메시징 채널

### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

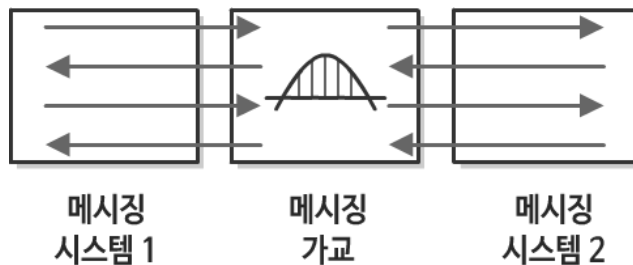
##### (2) 사용패턴

##### ⑧ 애플리케이션 계층 별 채널 어댑터

- 사용자 인터페이스 어댑터
  - HTML 기반의 HTTP 메커니즘을 사용하는 경우 어댑터가 사용자 처럼 작동하면서 사용자 인터페이스를 처리하므로 애플리케이션 내부에 접근하지 않는 장점이 있으나, 불안정적이고 속도가 낮으며 자주 변경되는 경향이 있다는 단점이 있음
- 비즈니스 로직 어댑터
  - 애플리케이션이 잘 정의된 API를 가지는 경우 해당 어댑터 사용 (사용자 인터페이스에 비해 안정적이고 효율적)
- 데이터베이스 어댑터
  - 테이블에 트리거를 추가해 테이블 변경 시마다 메시지를 전송하는 등 애플리케이션 도움없이 직접 데이터베이스 정보 추출이 가능함
  - 데이터베이스를 직접 업데이트 하는 것은 위험할 수 있음

##### ⑨ 메시징 가교(Messaging Bridge)

- 기업이 여러 메시징 시스템을 사용하는 경우나, 기업 내부 메시징 시스템과 여러 외부 메시징 시스템들 사이에서 메시지를 융합하는 방법
- 두 메시징 시스템의 해당 채널들을 개별적으로 연결하고, 한 시스템 포맷에서 다른 시스템 포맷으로 변환하는 역할을 함





## 메시징 채널

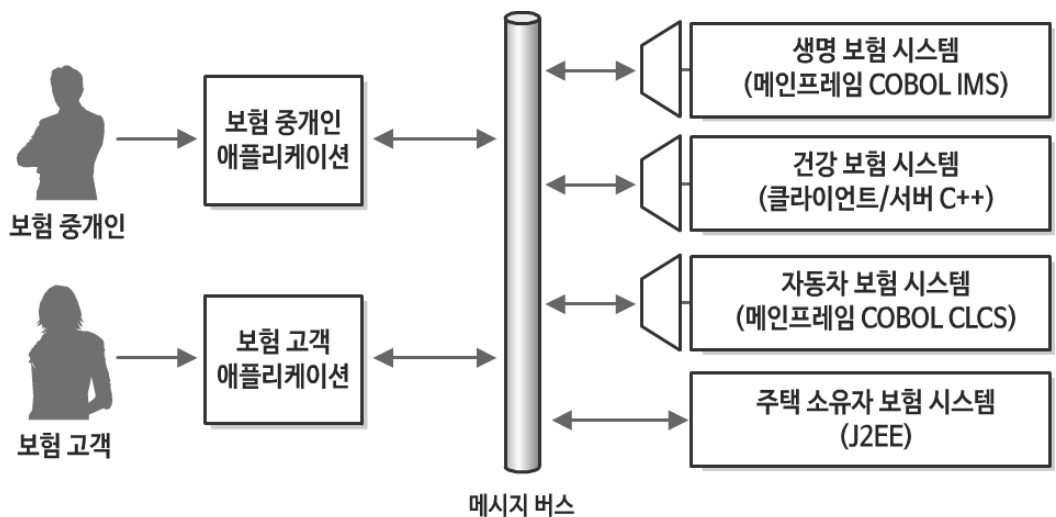
### 1. 메시징 채널

#### 1) 메시지 채널 사용 패턴

##### (2) 사용패턴

###### ⑩ 메시지 버스(Message Bus)

- 독립적으로 운영되면서도 통일된 방식으로 협력해야 하는 애플리케이션
- 각 애플리케이션이 정규 데이터 모델을 사용하여 느슨하게 결합
- 공통 통신 인프라
  - 메시지를 전달할 수 있는 메시지 전송 통신 인프라
- 어댑터
  - 일부 시스템들이 메시지 버스와 인터페이스 하는 데 사용
- 공통 명령 구조
  - 모든 참가자가 이해할 수 있는 공통 명령 이용



## ◆ 핵심정리 ◆

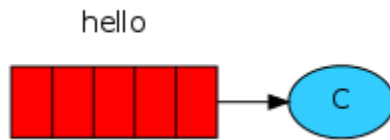
### 1. 메시지 채널 사용 패턴

- 메시지 채널
  - 메시징 애플리케이션에서 수신자와 발신자를 연결하는 가상 파이프
  - 메시징 시스템의 논리 주소로 구현 방식에 따라 엔드포인트들은 직접 연결될 수도 있고 중앙 허브를 거쳐 연결함
- 사용패턴
  - 1대1 (point-to-point) 채널
  - 1대다 (publish-subscribe) 채널
  - 데이터 형식 채널
  - 무효 메시지 채널
  - 죽은 편지 채널
  - 보장 전송
  - 채널 어댑터
  - 메시징 가교
  - 메시지 버스

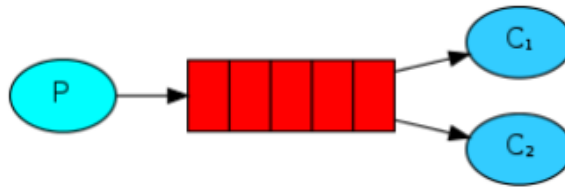
## ◆ 핵심정리 ◆

### 2. Rabbit-MQ 실습

- Point to Point
  - 단일 큐를 선언하여 메시지를 확인하는 예제



- Queue
  - 큐 하나를 공유하여 두 Client 가 메시지를 소비하는 예제



- Publish-Subscribe
  - 게시 구독 채널이 연결된 구독자에게만 메시지를 보내는 예제

