

P R E S E N T A T I O N

30주차 주제 DDD

DDD ~Strategic Design~

by mun

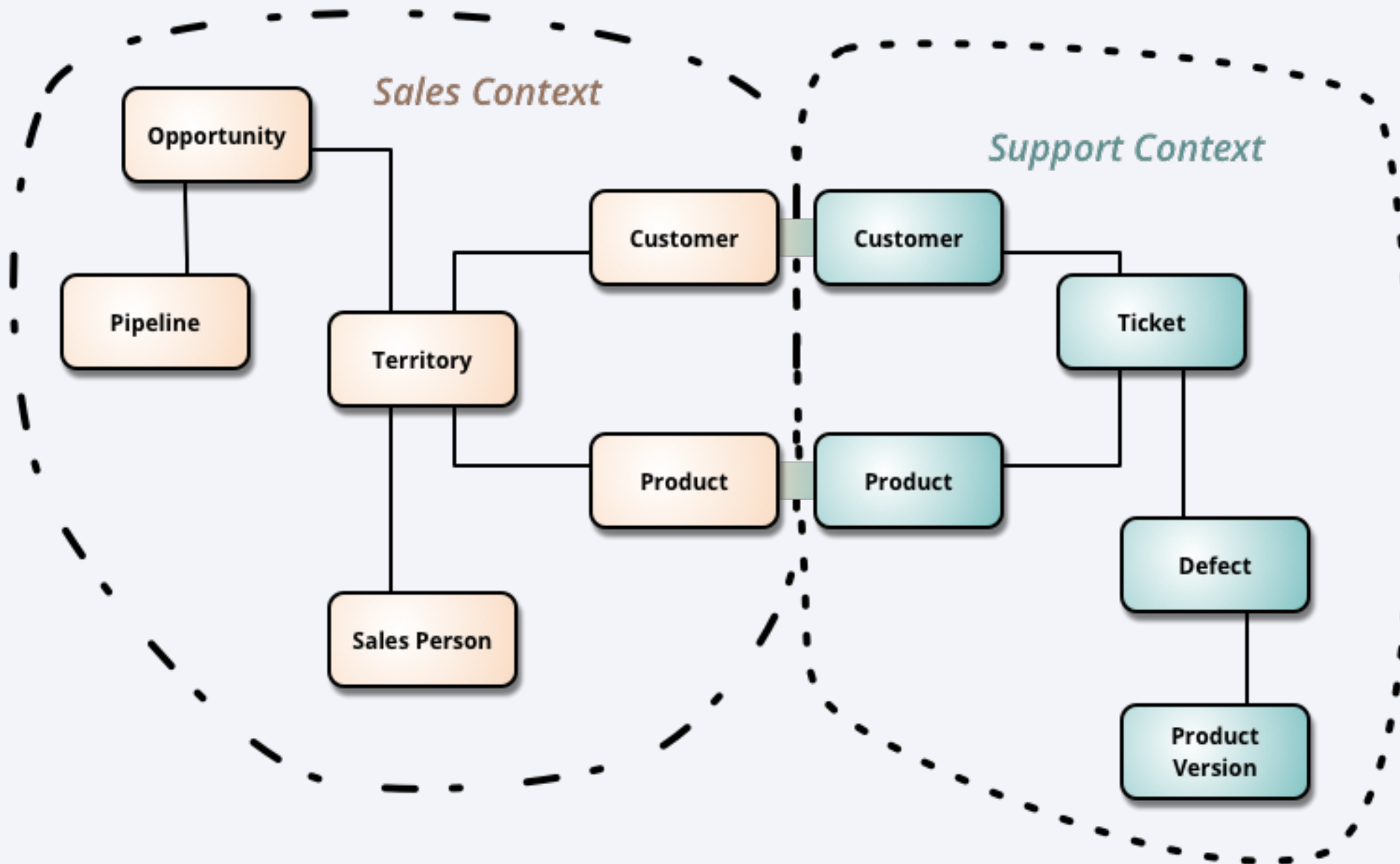


DDD

Domain Driven Design

- 도메인 주도 설계
- 도메인과 일치하도록 소프트웨어를 모델링하는 데 중점을 둔 소프트웨어 설계 접근 방식
- 소프트웨어 코드의 구조와 언어가 비즈니스 도메인의 용어를 일치시켜 나간다는 특징
- 데이터 중심의 접근법을 탈피해서 순수한 도메인의 모델과 로직에 집중하는 것

DDD에서의 Domain



- 유사한 업무의 집합
- 어플리케이션은 비즈니스 도메인별로 나누어 설계 및 개발 될 수 있다

비즈니스 도메인 별로
나누어 설계하는 방식이다

Domain Driven Design



- 비즈니스 Domain별로 나누어 설계하는 방식
- 기존의 일방향 소통구조를 탈피하여
현업과 IT의 쌍방향 커뮤니케이션을 매우 중요하게 생각
- 핵심 목표는 모듈간의 의존성 최소화, 응집성 최대화
- 개념적 설계 (Strategic Design) 와
구체적 설계 (Tactical Design) 로 나눌 수 있다

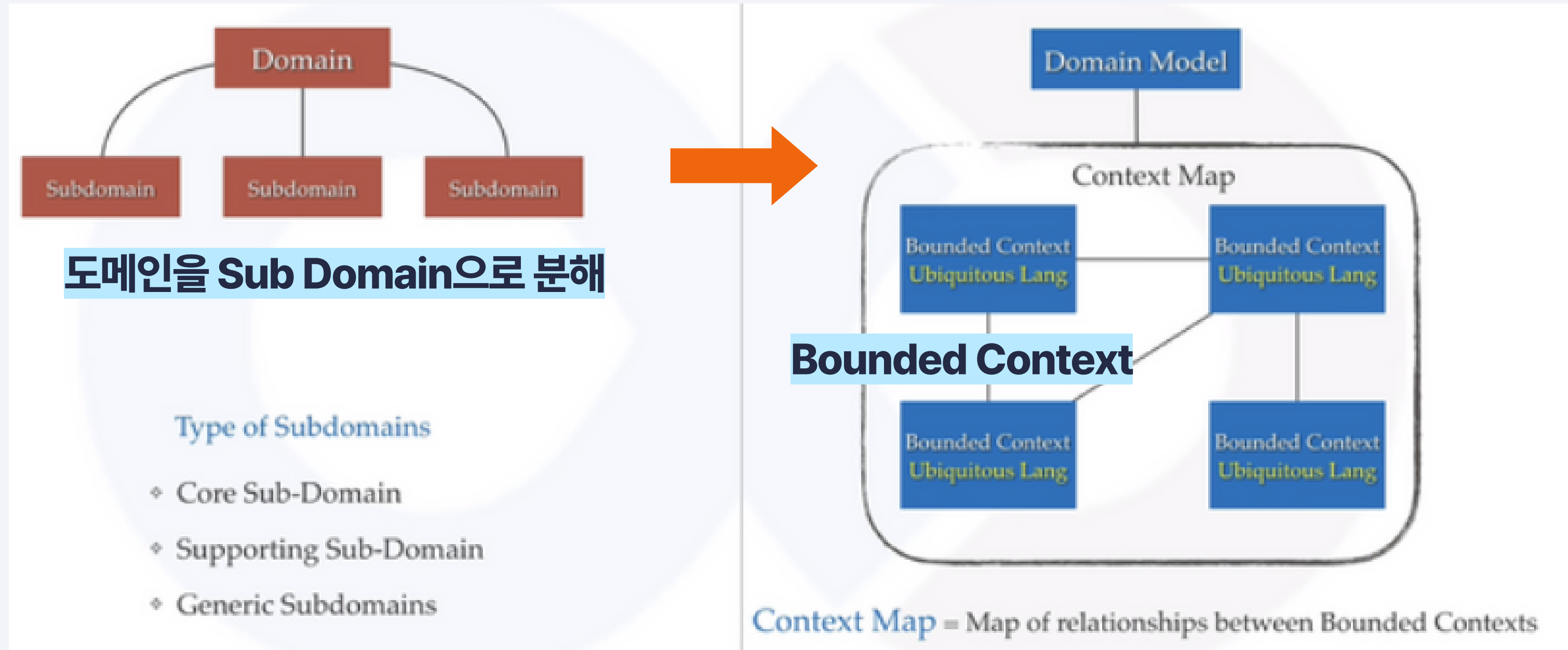
Strategic Design: 개념 설계

Business Domain의 상황(Context: 대상사용자, 상황)에 맞게 설계

- 전략적 설계를 위해 Business Domain의 상황(Context)을 Event storming으로 공유하고, 비즈니스 목적별로 서비스들을 그룹핑하는 과정
- **Bounded Context**
도메인의 사용자, 프로세스, 정책/규정 등을 고유한 비즈니스 목적별로 그룹핑한것
사용자, 프로세스, 정책/규정들을 그 도메인의 Context라고 말할 수 있다
즉, Bounded Context는 도메인 안의 서비스를 경계 지은 Context의 집합이라고 할 수 있다
- **Domain Model**
비즈니스 도메인의 서비스를 추상화한 설계도
도메인을 Sub Domain으로 분해한것 + Bounded Context = Domain Model

Event storming

비즈니스 도메인 내에서 일어나는것들을 찾아 Bounded Context를 식별하는 방법론



- 비즈니스 도메인의 서비스를 추상화한 설계도
도메인을 Sub Domain으로 분해한것 + Bounded Context = Domain Model

Event storming

비즈니스 도메인 내에서 일어나는것들을 찾아 Bounded Context를 식별하는 방법론



- **Sub-Domain**
유사한 업무를 그룹핑한 것

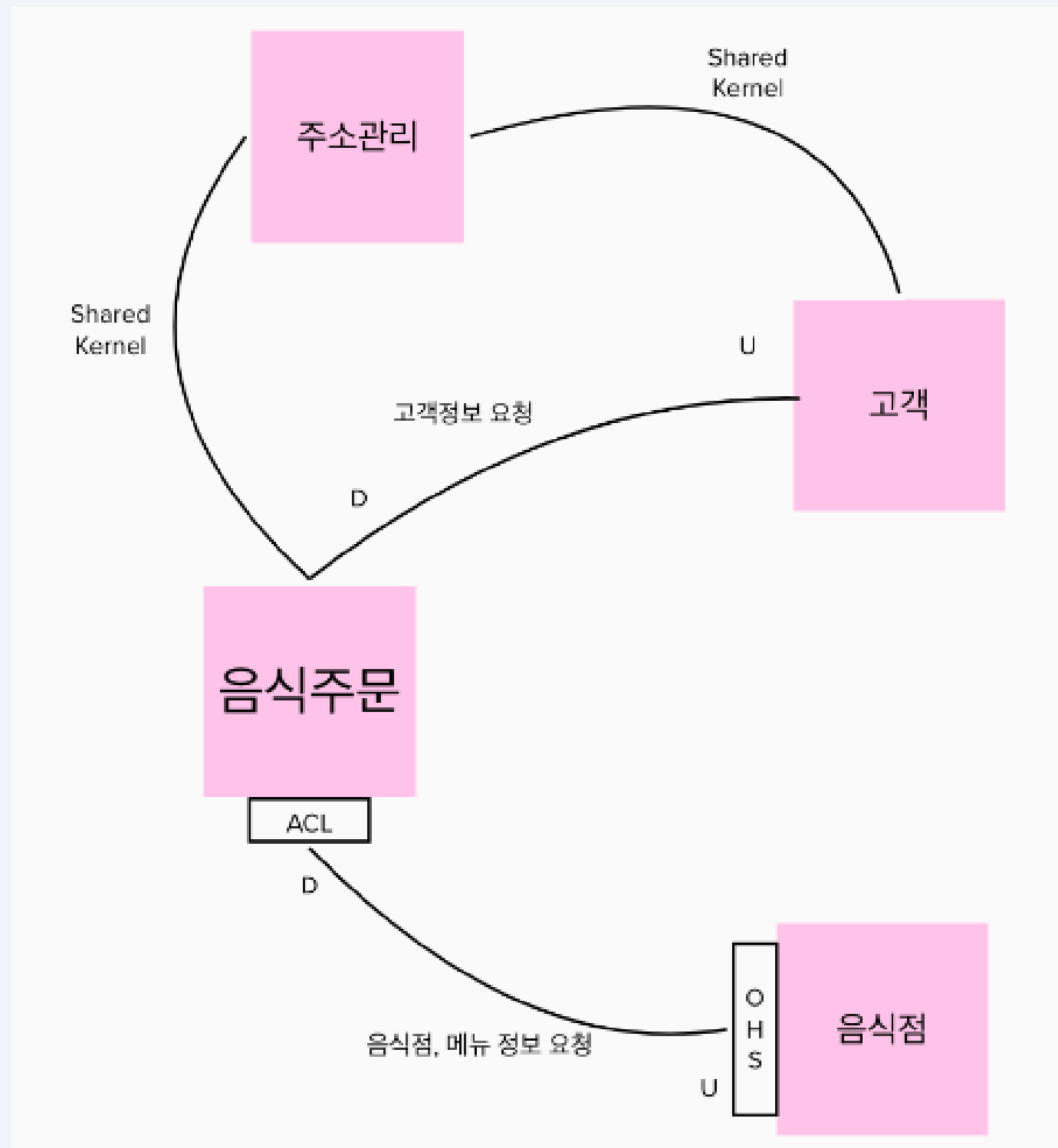
Strategic Design: 개념 설계

Business Domain의 상황(Context: 대상사용자, 상황)에 맞게 설계

- 전략적 설계를 위해 Business Domain의 상황(Context)을 Event storming으로 공유하고, 비즈니스 목적별로 서비스들을 그룹핑하는 과정
- **Bounded Context**
도메인의 사용자, 프로세스, 정책/규정 등을 고유한 비즈니스 목적별로 그룹핑한것
사용자, 프로세스, 정책/규정들을 그 도메인의 Context라고 말할 수 있다
즉, Bounded Context는 도메인 안의 서비스를 경계 지은 Context의 집합이라고 할 수 있다
- **Domain Model**
비즈니스 도메인의 서비스를 추상화한 설계도
도메인을 Sub Domain으로 분해한것 + Bounded Context = Domain Model
- **Context Map**
Bounded Context간의 관계를 나타낸 도식화한 다이어그램

Event storming

비즈니스 도메인 내에서 일어나는것들을 찾아 Bounded Context를 식별하는 방법론



- **Context Map**
Bounded Context간의 관계를 나타냄

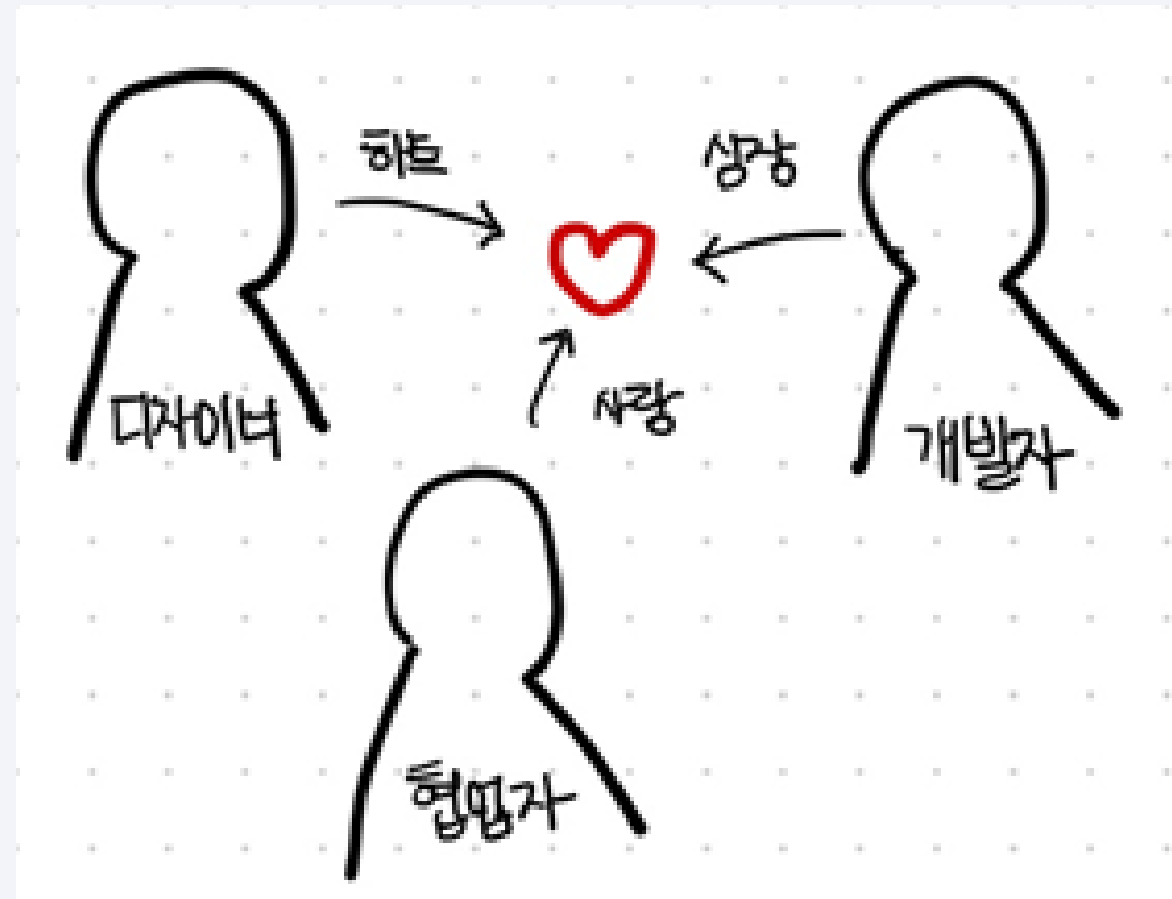
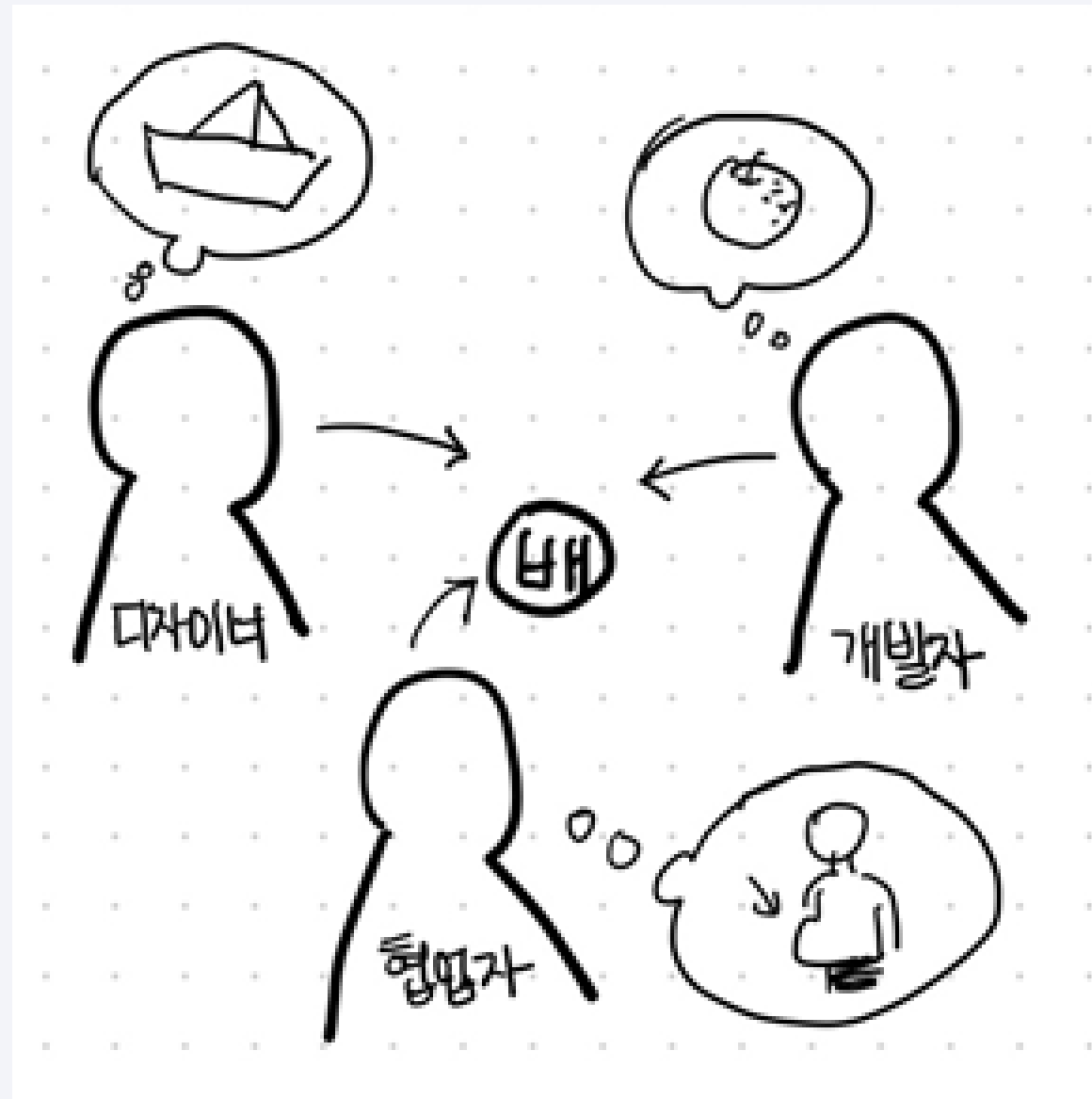
Strategic Design: 개념 설계

Business Domain의 상황(Context: 대상사용자, 상황)에 맞게 설계

- 전략적 설계를 위해 Business Domain의 상황(Context)을 Event storming으로 공유하고, 비즈니스 목적별로 서비스들을 그룹핑하는 과정
- **Bounded Context**
도메인의 사용자, 프로세스, 정책/규정 등을 고유한 비즈니스 목적별로 그룹핑한것
사용자, 프로세스, 정책/규정들을 그 도메인의 Context라고 말할 수 있다
즉, Bounded Context는 도메인 안의 서비스를 경계 지은 Context의 집합이라고 할 수 있다
- **Domain Model**
비즈니스 도메인의 서비스를 추상화한 설계도
도메인을 Sub Domain으로 분해한것 + Bounded Context = Domain Model
- **Context Map**
Bounded Context간의 관계를 나타낸 도식화한 다이어그램
- **Ubiquitous Language**
참여자들이 동일한 의미로 이해하는 언어

Strategic Design: 개념 설계

비즈니스 도메인 내에서 일어나는 것들을 찾아 Bounded Context를 식별하는 방법론



- **Ubiquitous Language**
참여자들이
동일한 의미로 이해하는 언어

Strategic Design: 개념 설계

비즈니스 도메인 내에서 일어나는것들을 찾아 Bounded Context를 식별하는 방법론

- **Event storming**
비즈니스 도메인 내에서 일어나는것들을 찾아
Bounded Context를 식별하는 방법론

Step1. Domain Event 정의

비즈니스 도메인내에 발생하는 모든 이벤트를 과거형으로 기술

음식종류가
선택되었다

음식점이
선택되었다.

음식목록조회
되었다

음식점 메뉴를
선택되었다

배송지가 입력되었다

주문내역
조회 되었다.

주문취소 되었다

음식점
평가가
나왔다.

- 오렌지색 사용
- 이벤트가 발생하는 시간 순서대로 붙임
- 동시 수행되는 이벤트는 수직으로 붙임
- 중복된 것을 없애거나 합침

주문내역
조회 되었다.

결제방식이 선택되었다

결제가
되었다

주문접수 되었다

주문처리
상태가 조회
되었다

Step2. Tell the story

도출된 이벤트로 도메인의 업무 흐름을 이해하고 토론하여 보완

음식종류가
선택되었다

음식점이
선택되었다.

음식목록조회
되었다

음식점 메뉴를
선택되었다

배송지가 입력되었다

주문내역
조회 되었다.

주문취소 되었다

음식점
평가가
나왔다.

- 도메인 전문가가 도출된 이벤트로 업무 흐름 설명
- 상호 질문을 통해 도메인 이벤트를 추가하거나 조정

결제방식이 선택되었다

결제가
되었다

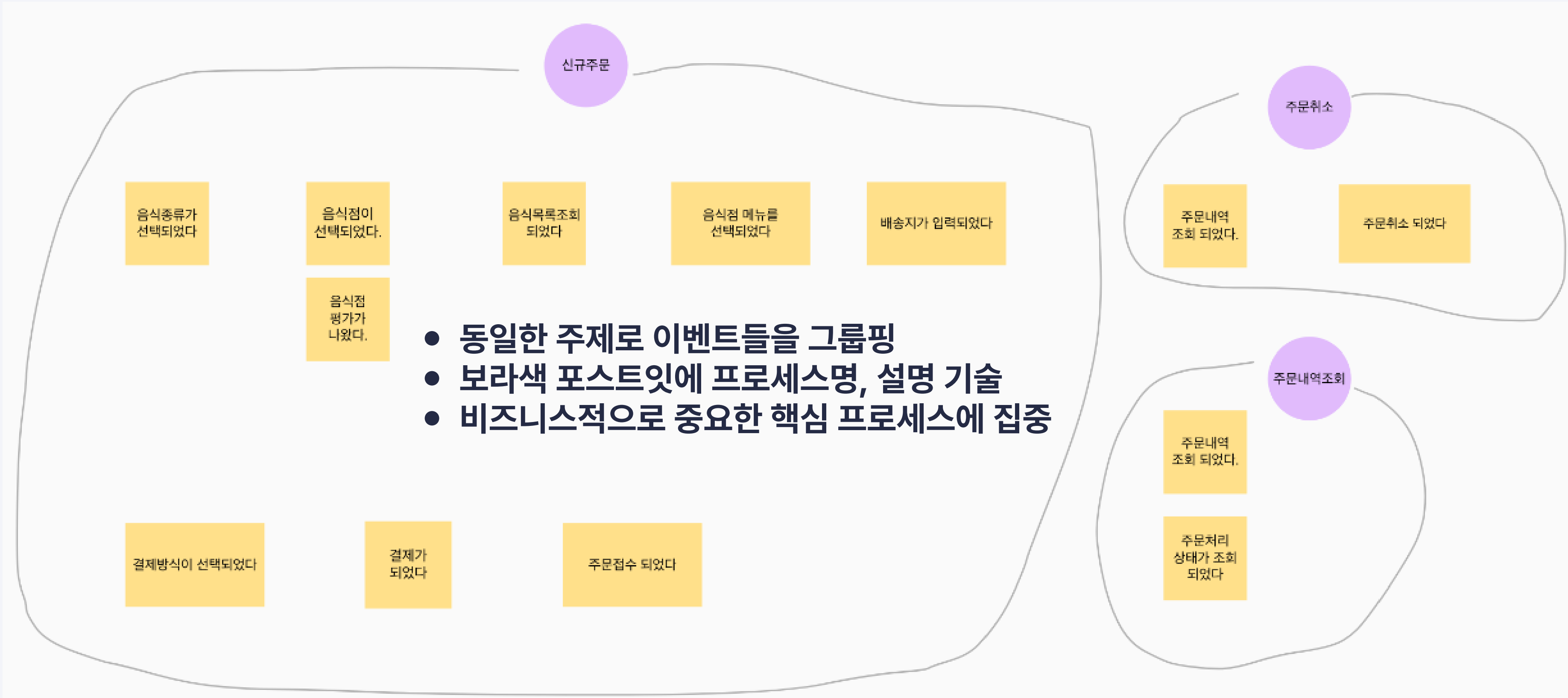
주문접수 되었다

주문내역
조회 되었다.

주문처리
상태가 조회
되었다

Step3. 프로세스로 그룹핑

이벤트들을 프로세스로 그룹핑



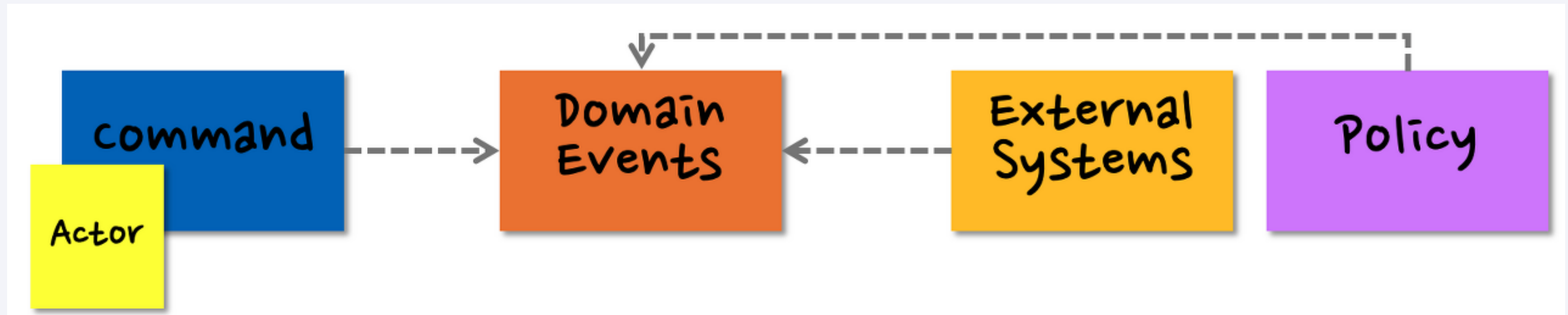
Step4. Command 정의

이벤트를 발생시키는 명령을 현재형으로 정의하며 명령형으로 기술



Step5. Trigger 정의

행위를 일으키는 행위자와 이벤트를 일으키는 외부 시스템과 정책/규정을 정의



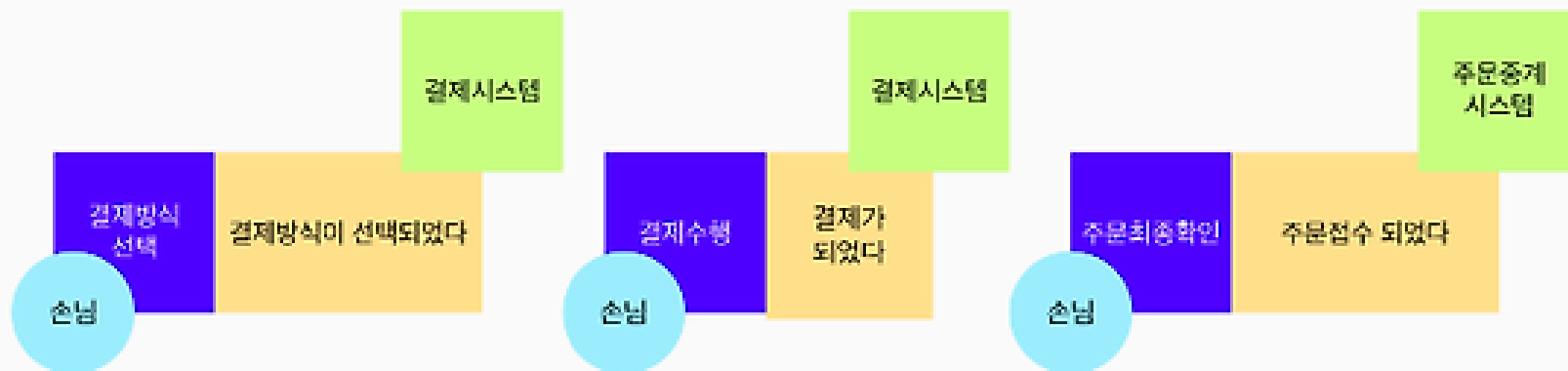
Step5. Trigger 정의

행위를 일으키는 행위자와 이벤트를 일으키는 외부 시스템과 정책/규정을 정의



• 행위자: 행위 밑에 겹쳐 붙임 (하늘색)

• 외부 시스템: 이벤트의 우측 상단에 겹쳐 붙임 (초록색)



• 관련 정책이나 규정:
이벤트 우측 하단 (보라색)



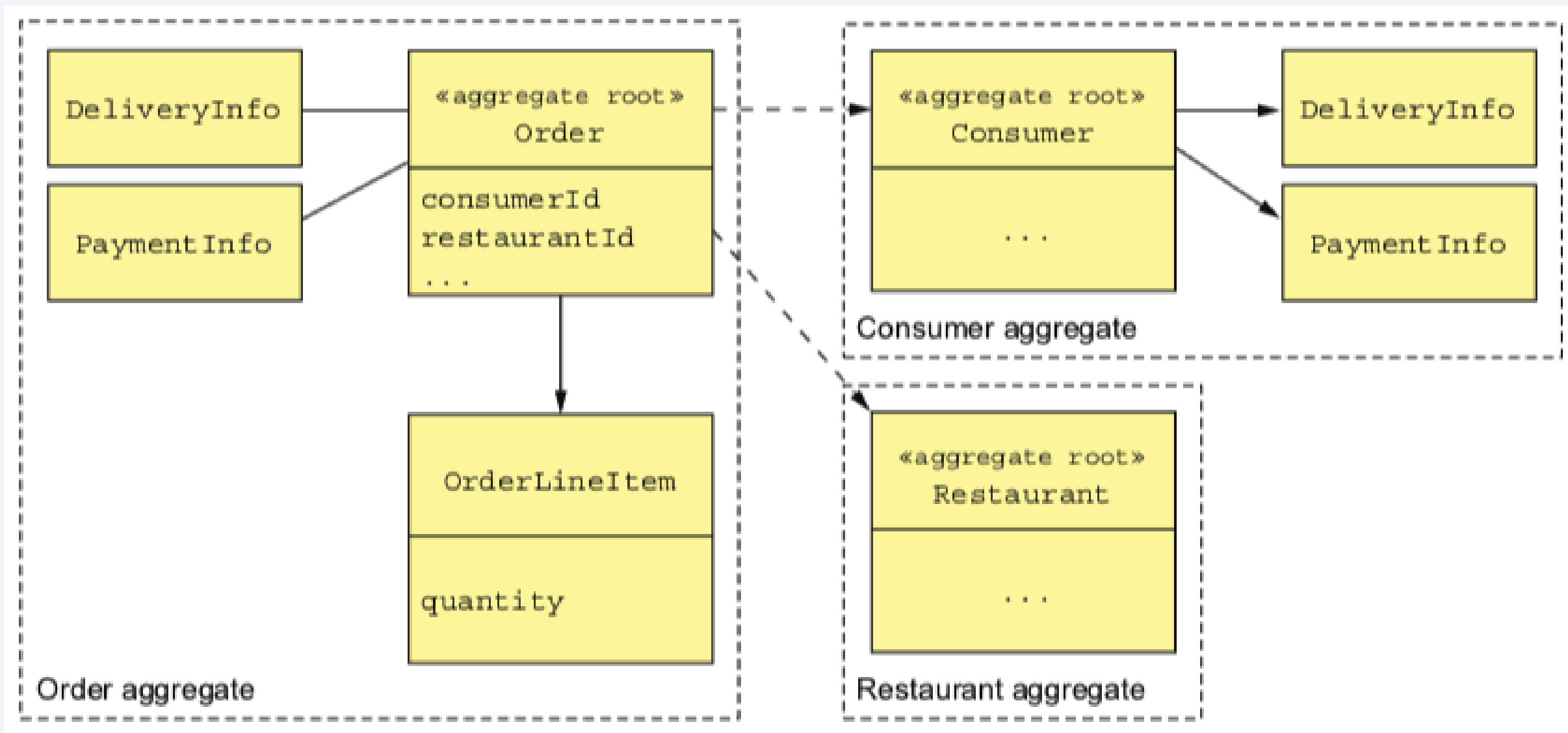
Step6. Aggregate 정의

Command 수행을 위해 CRUD해야 하는 데이터 객체 정의

- Entity와 VO의 집합
- Event storming때 여기까지 도출하는것은 어려운 경우가 많다
- 해당 단계에서는 엔티티까지 정의하고 다음 단계 진행을 추천

Step6. Aggregate 정의

Command 수행을 위해 CRUD해야 하는 데이터 객체 정의



- **Aggregate**
고유의 비즈니스 목적 수행을 위한
데이터 객체들의 집합

Step6. Aggregate 정의

Command 수행을 위해 CRUD해야 하는 데이터 객체 정의



Step6. Aggregate 정의

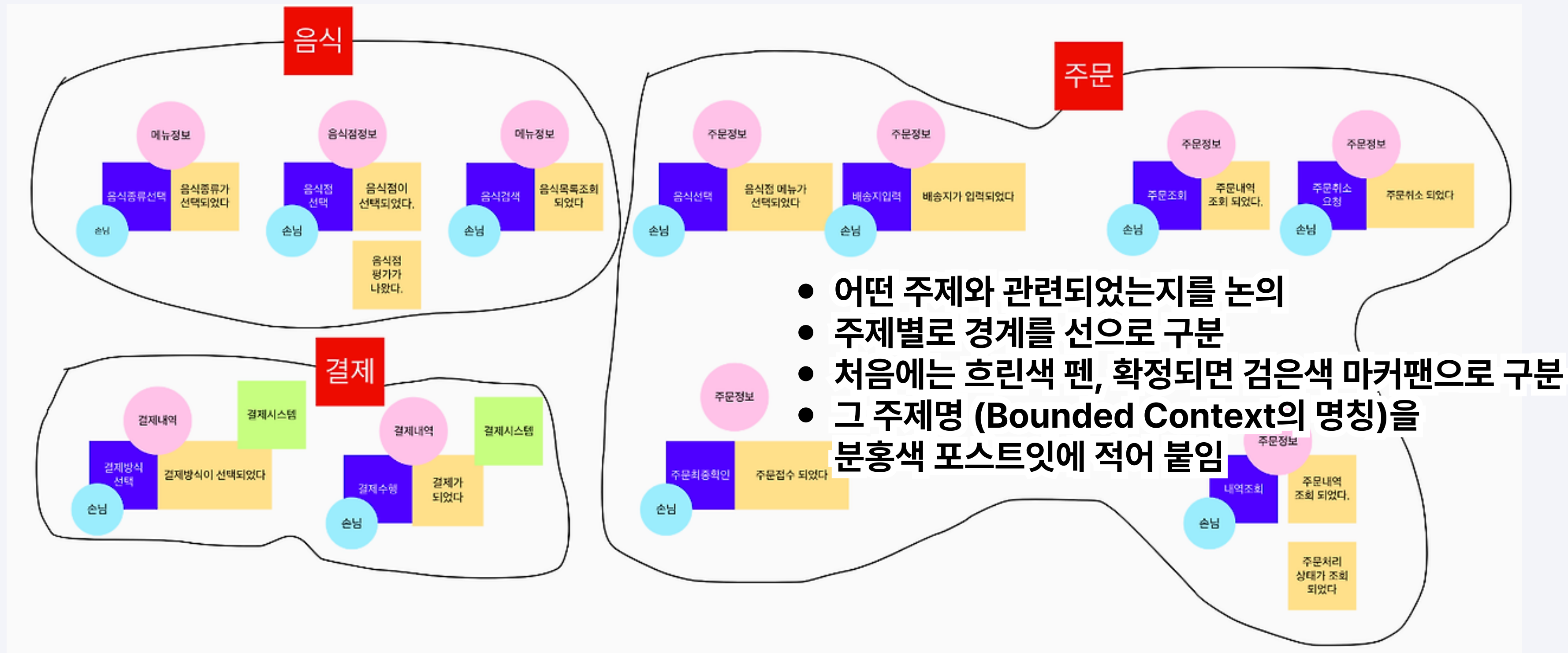
Command 수행을 위해 CRUD해야 하는 데이터 객체 정의

- 유사한 목적의 데이터들도 하나로 묶는다



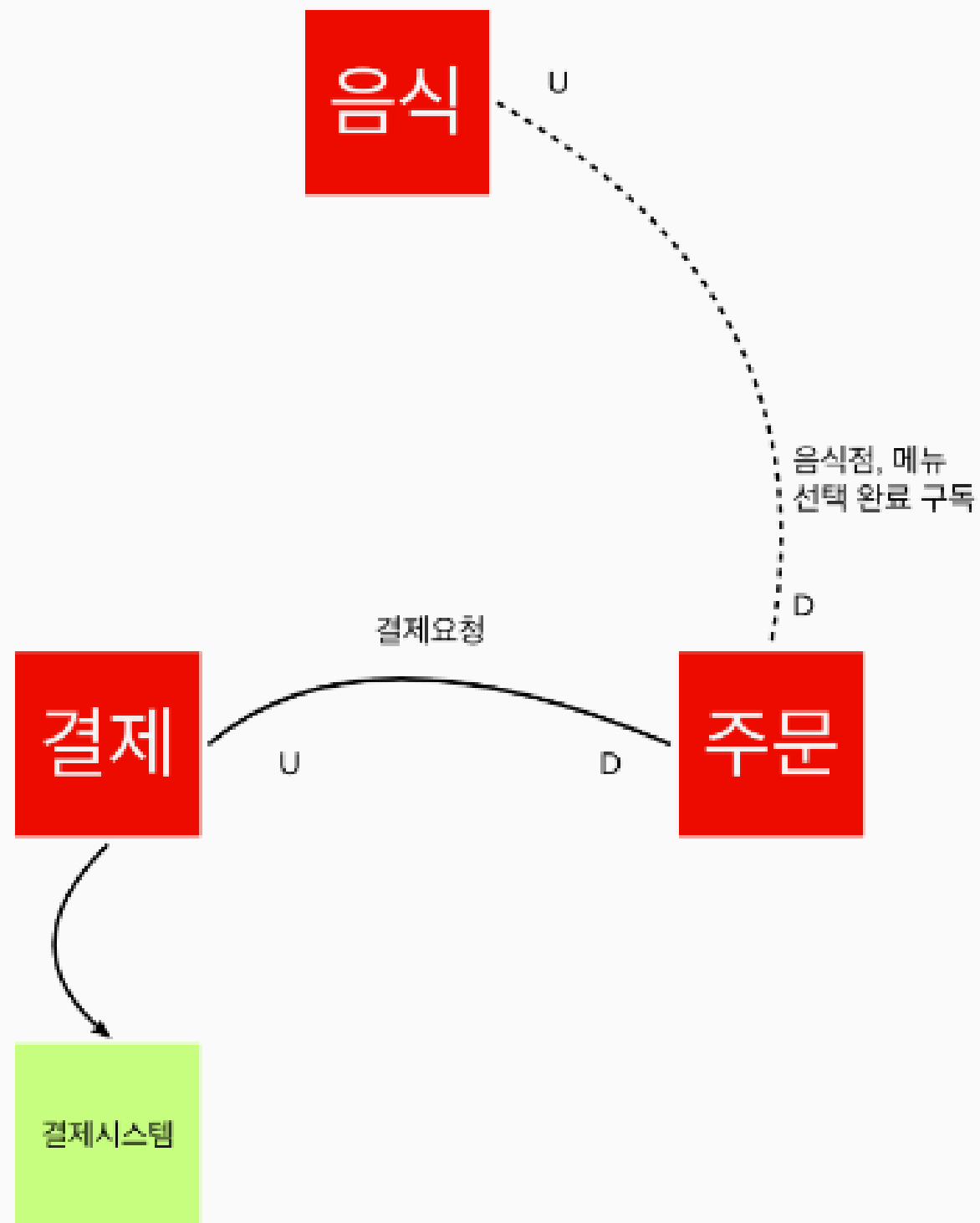
Step 7. Bounded Context정의

Bounded Context: 사용자, 프로세스, 정책/규정 등을 고유한 비즈니스 목적별로 그룹핑한것



Step 8. Context Map 작성

Bounded Context간의 관계를 도식화



- 동기 처리를 하는 경우 실선, 비동기 처리를 하는 경우 점선 사용

음식B/C Context Map

음식점, 메뉴를 선택 완료하였다는 메시지가 오면
주문B/C가 주문을 위한 어떤 처리를 한 후
결제를 요청하고 주문을 완료하는 관계를 표현

**이벤트 스토밍을 해보며
DDD 개념 설계를 연습해보아요!**

감사합니다

