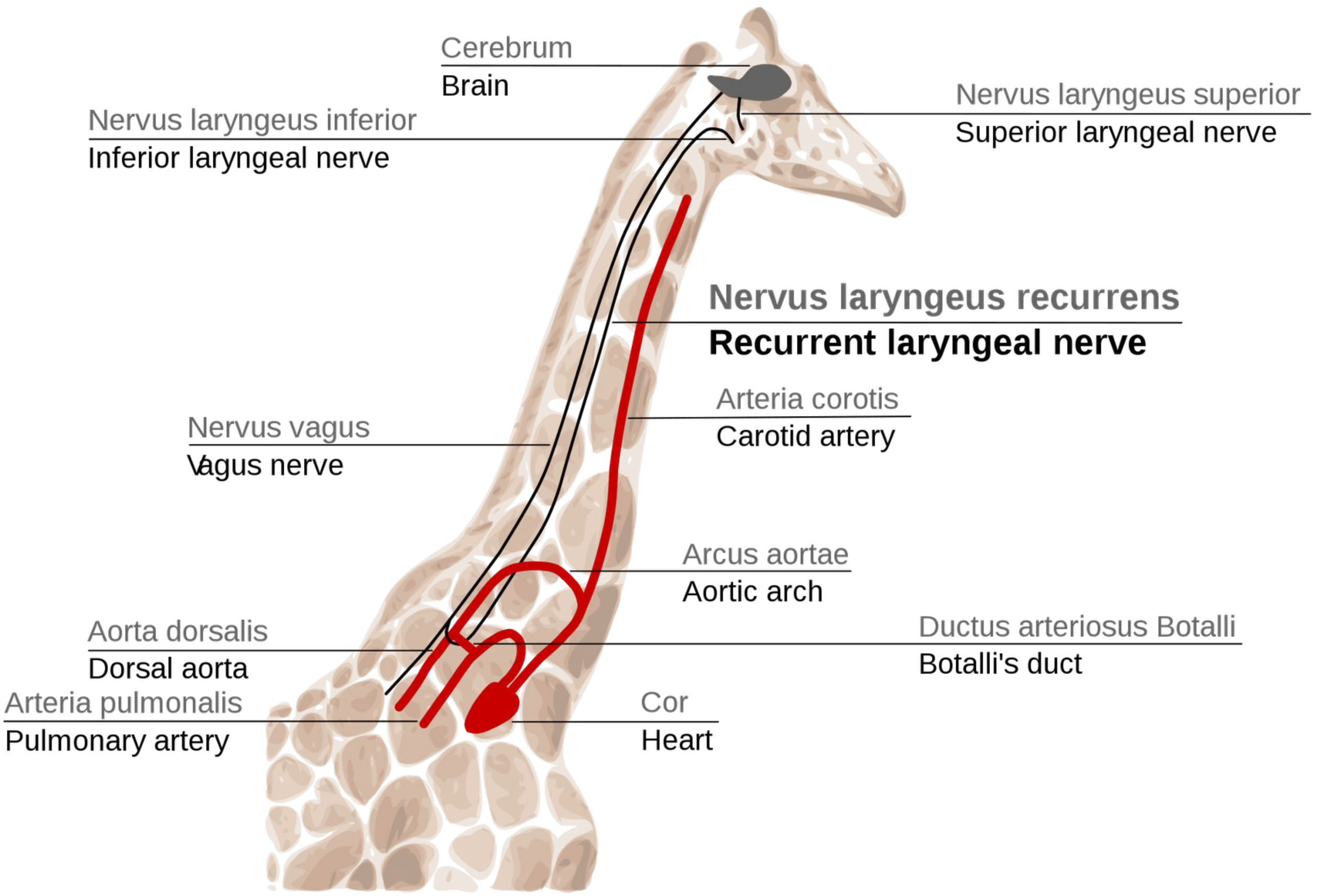
Business Problem Driven Design

Good architectures allow major architectural decisions to be deferred. The job of an architect is to defer decisions as long as possible. So that decisions can be made later with most possible information.

Show less

There is no **ideal**, we all have **blind spots**

And it’s not easy **to get read of legacy,** even in nature



DDD

Strategic thinking

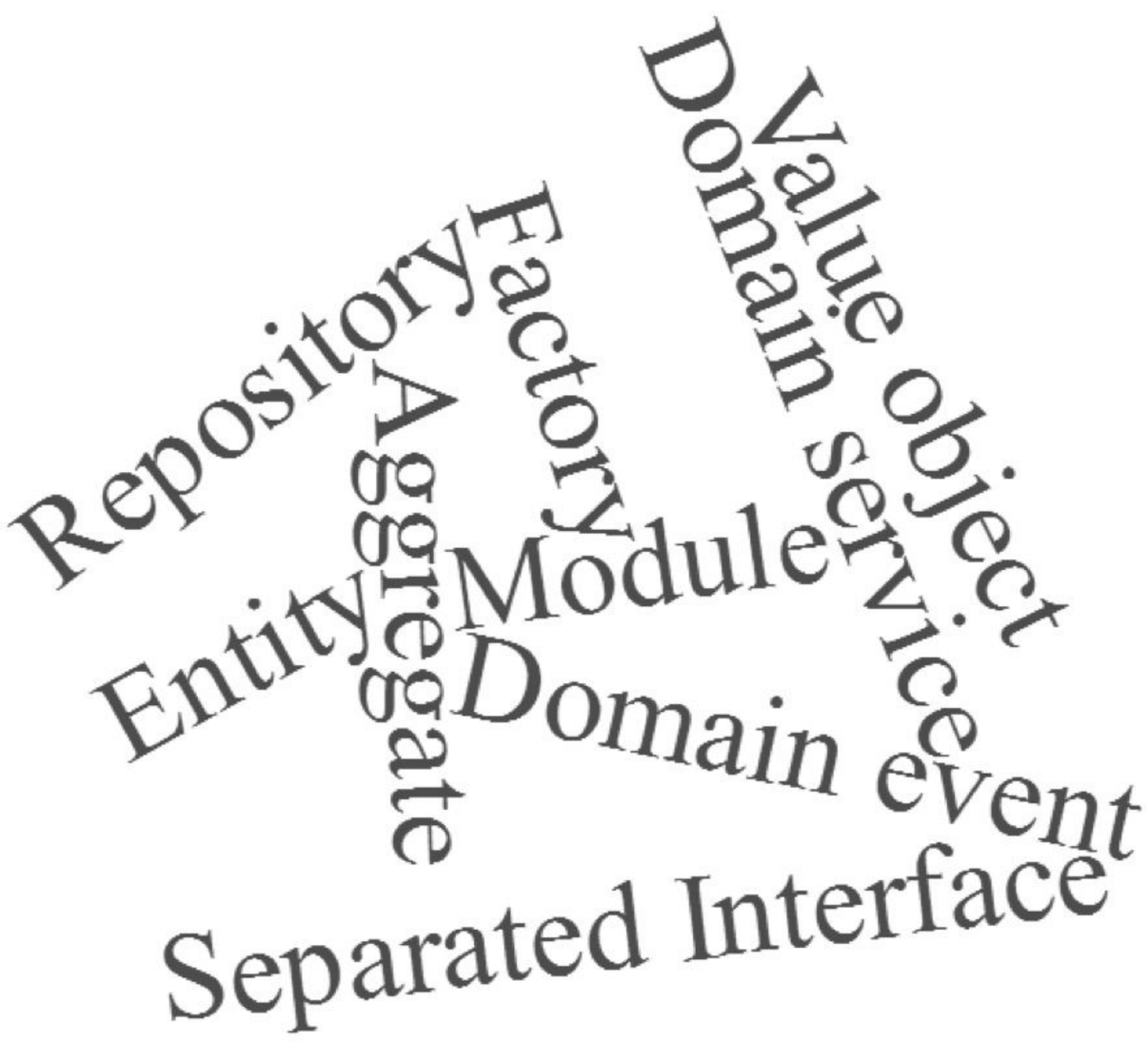
Interaction with people

Fighting complexity

Architecture approaches

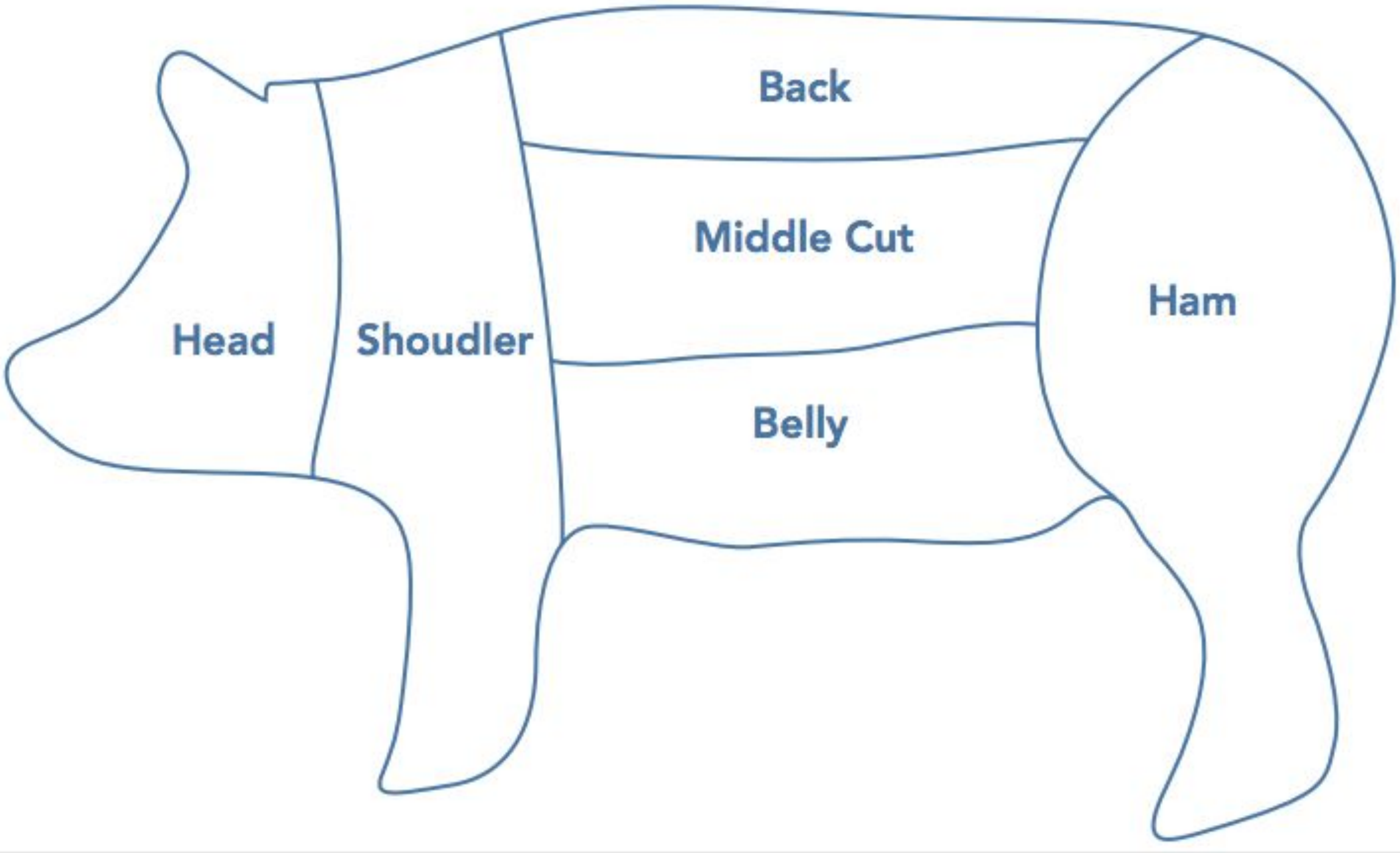
Tactical patterns

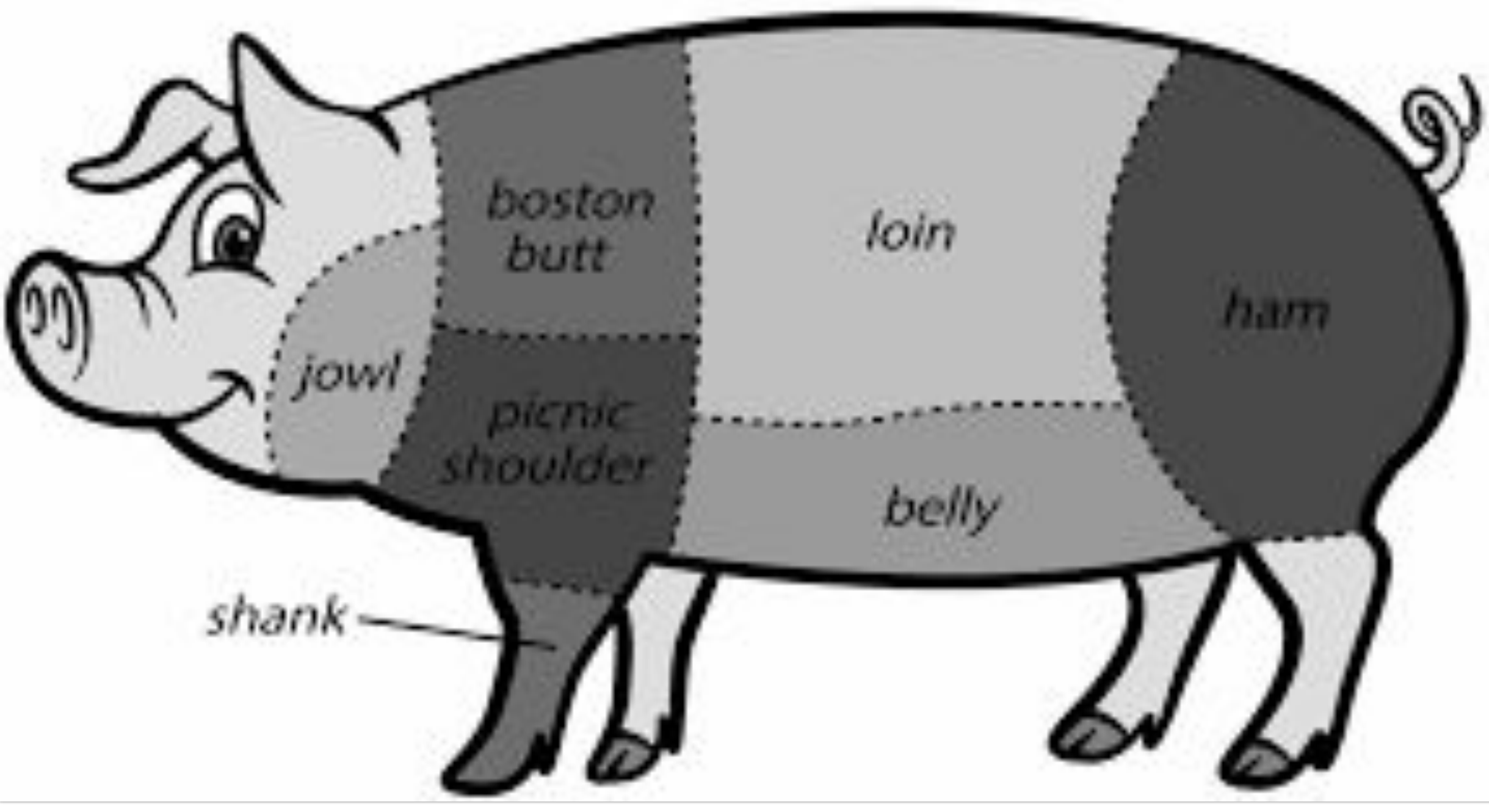
Context Map
Bounded Context
Anticorruption Layer
Ubiquitous Language
Domain model
Subdomains



Leave two pages to draw during the conference

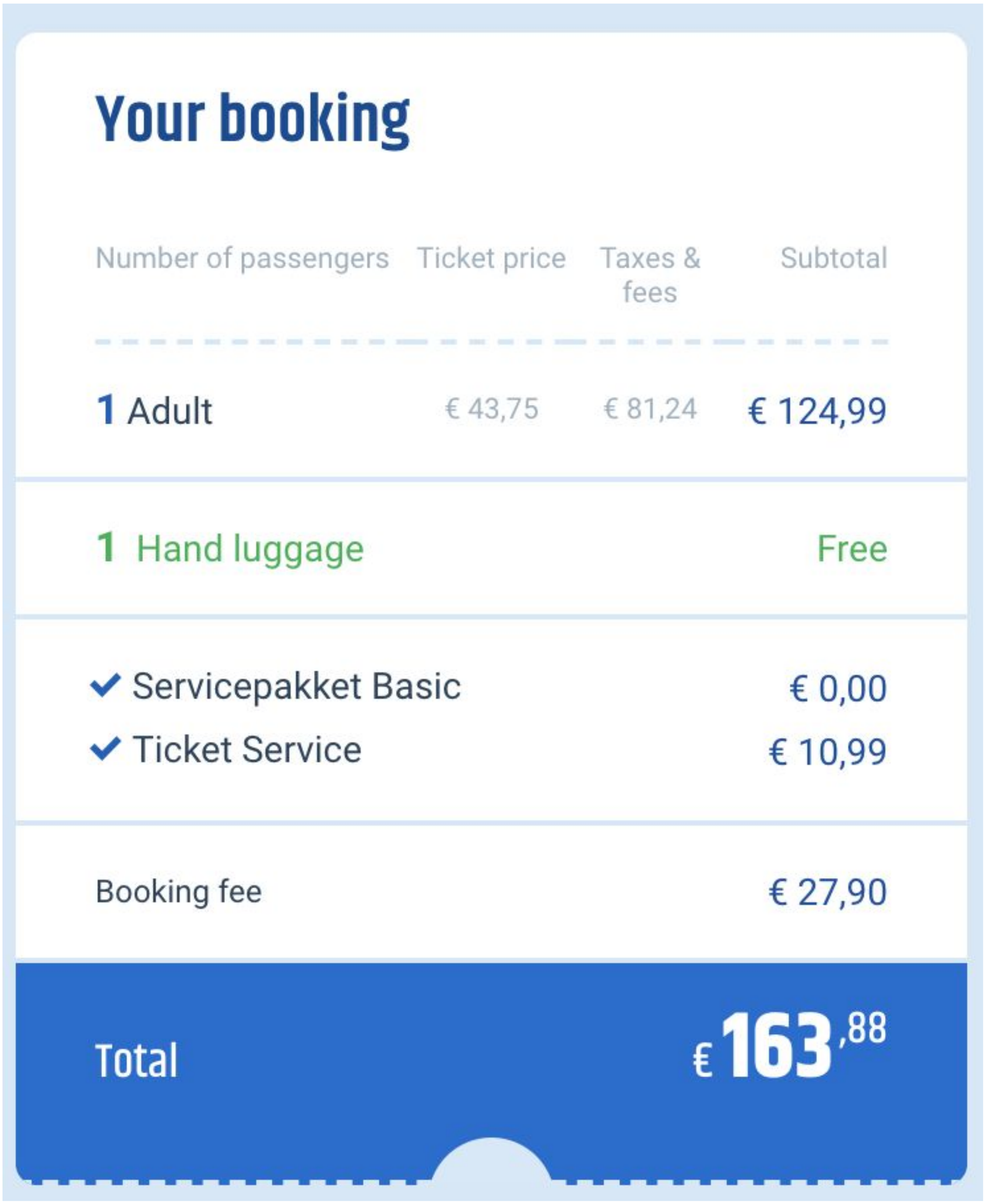
Even stakeholders in same domain can use different “language”



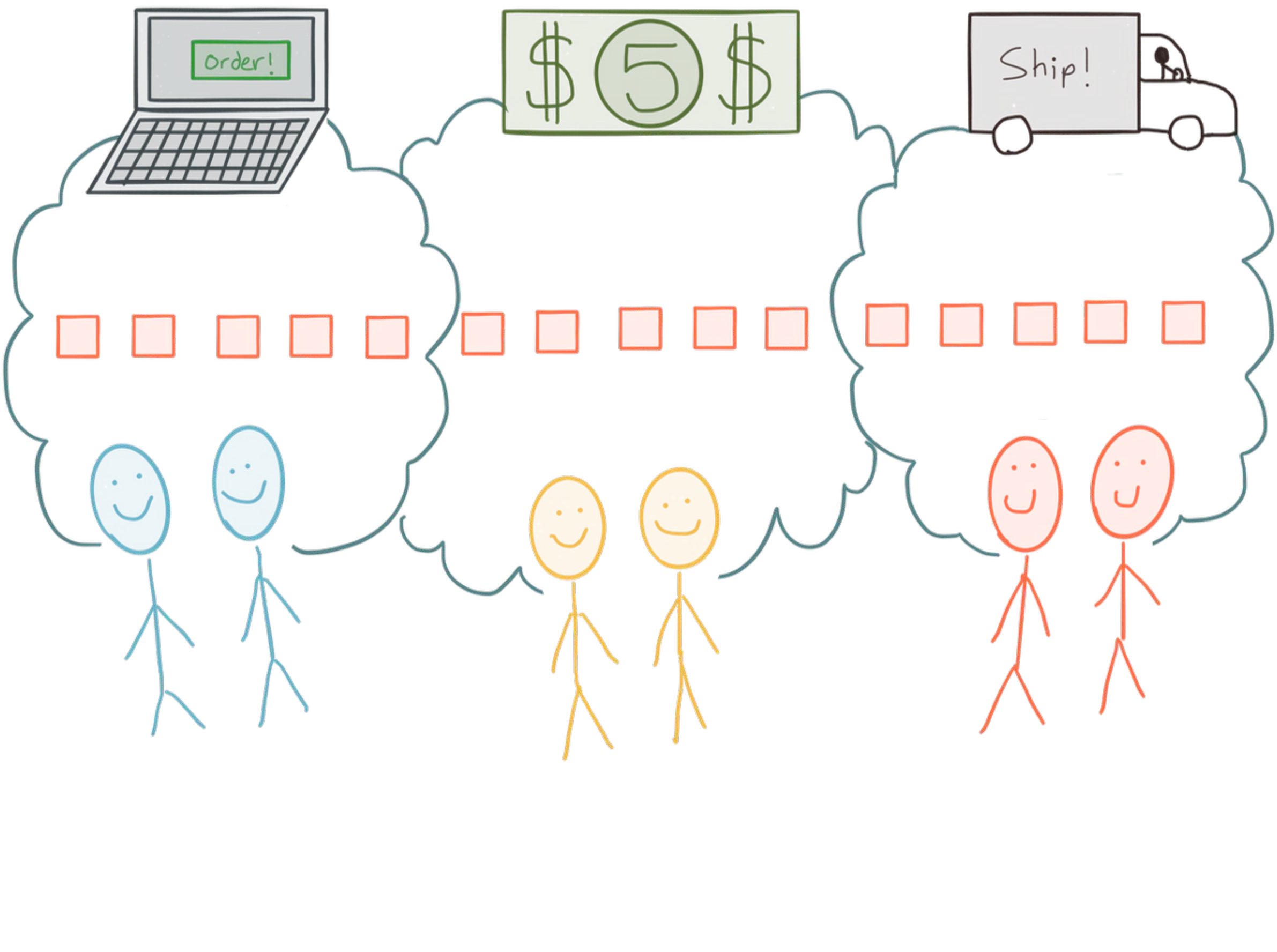


Business used **3 different names** for same component:

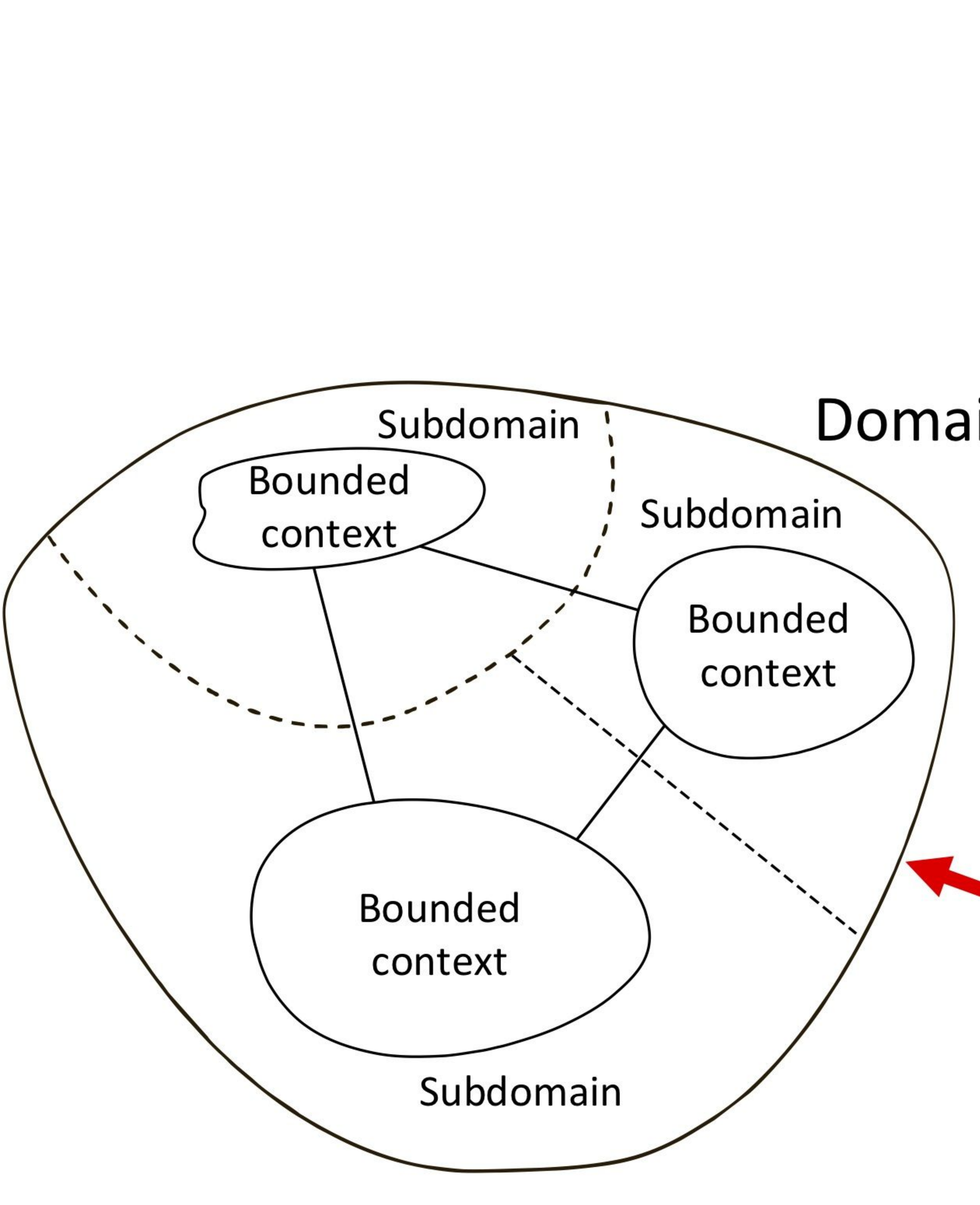
- Cost component
- Receipt component
- ShoppingBasket component

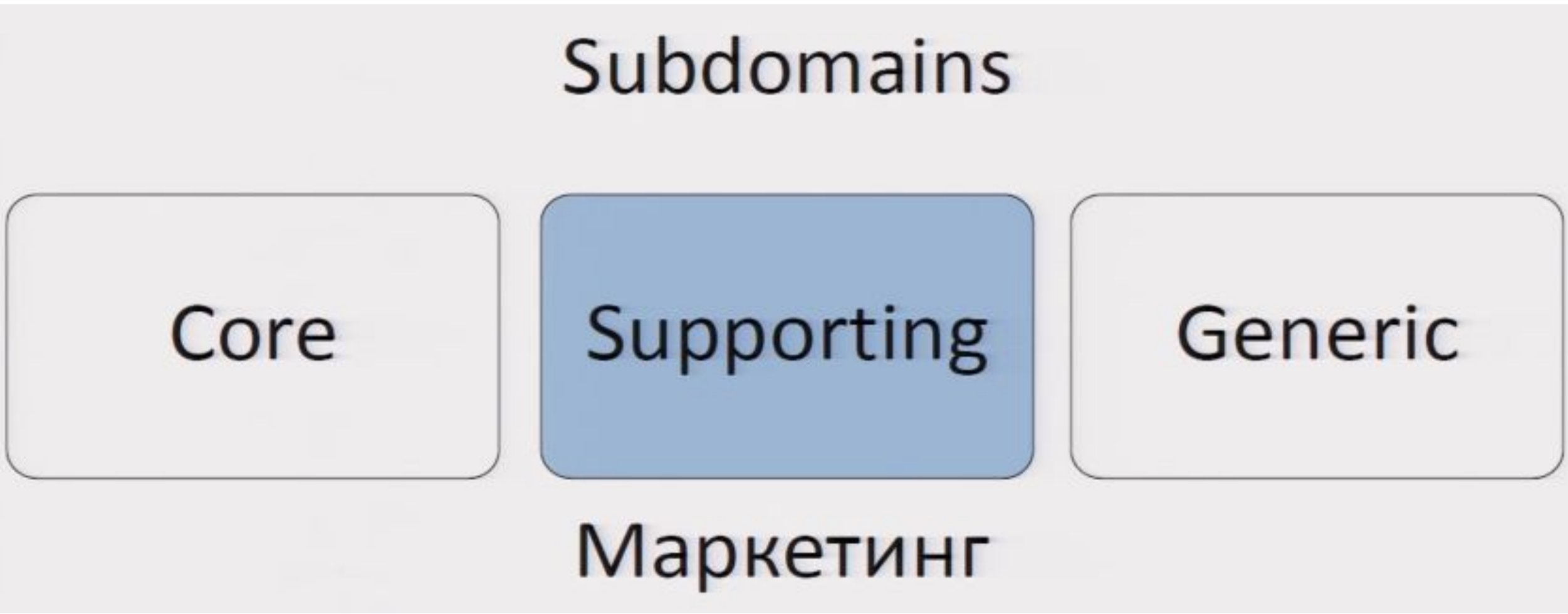


Does it feel ok? It could, if we talk about the same thing but in terms of different domains:



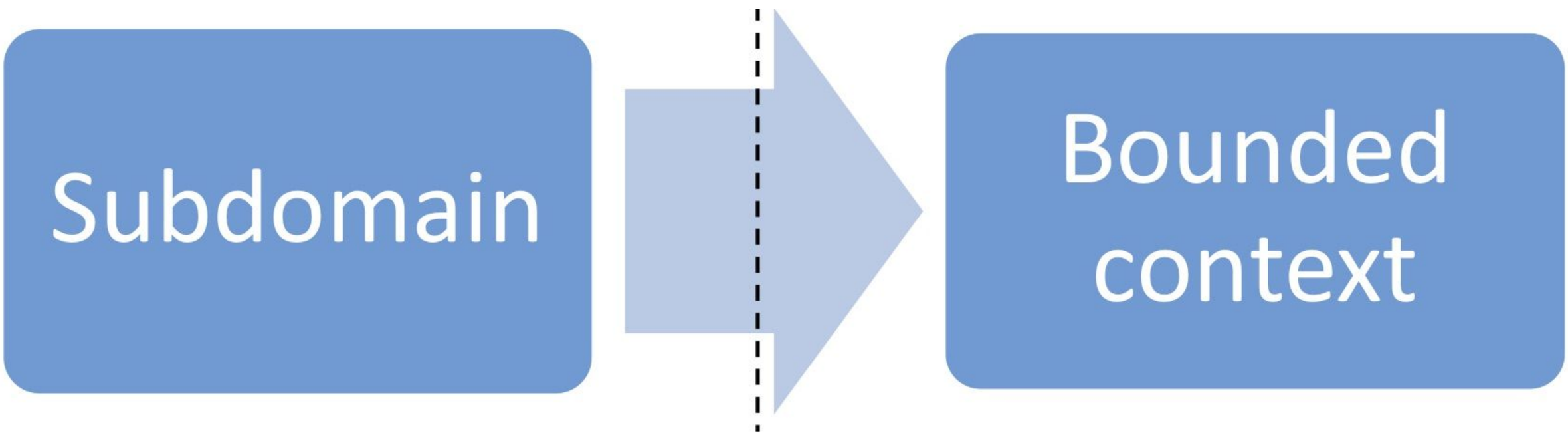
Ubiquitous Language


Domai
Subdomain
Bounded context
Subdomain
Bounded context
Bounded context
Subdomain


Subdomains
Core
Supporting
Generic
Маркетинг

Business problem

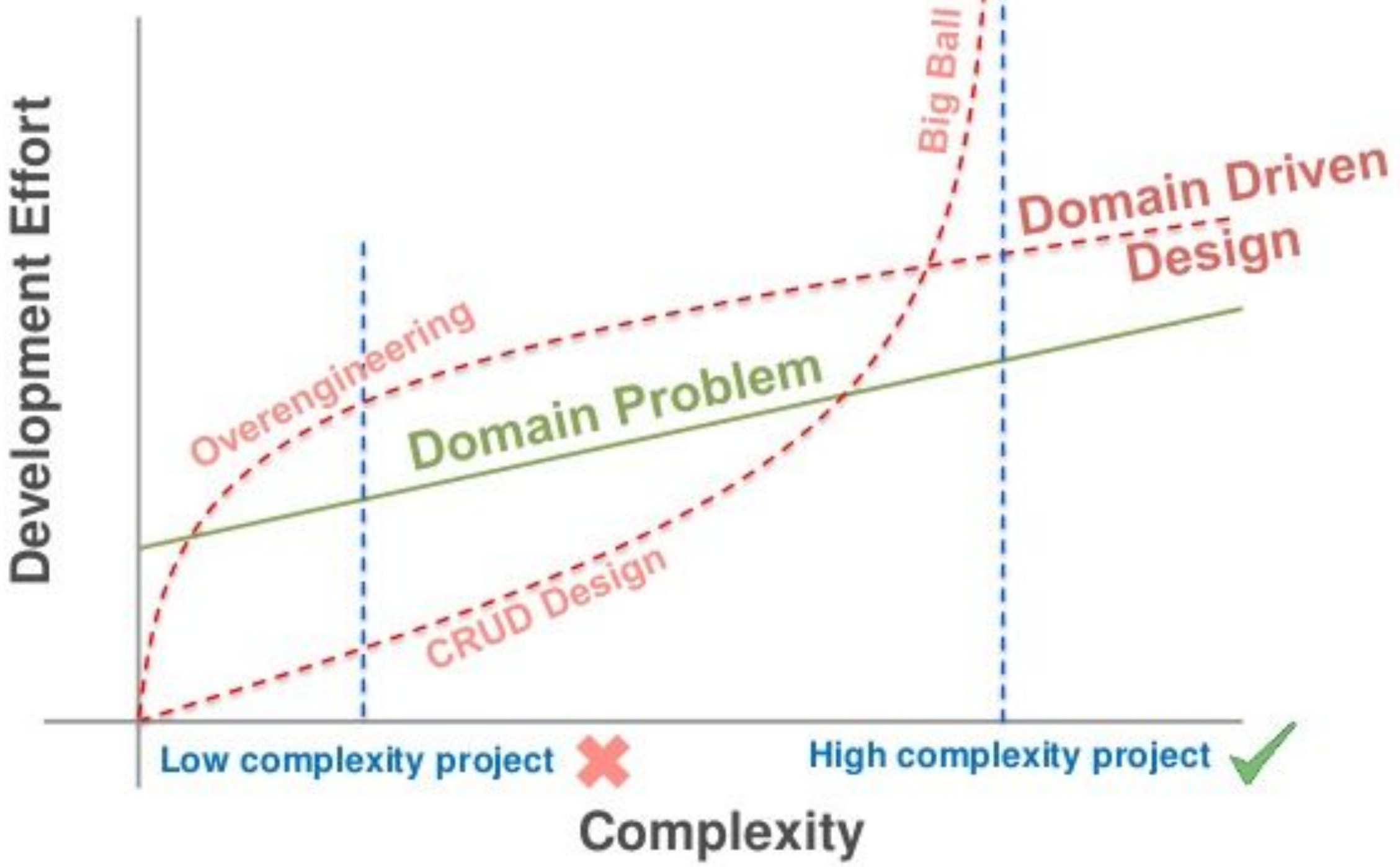
Software solution



«Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure.»

Conway Law

When and Why apply DDD?



Let’s check, do we follow DDD on our projects?

Domain

[draw during presentation]