Engenharia de Software

Padrões de Design DDD
Event Storming para Mapeamento de Domínios

Prof. Dr. Erik Aceiro Antonio

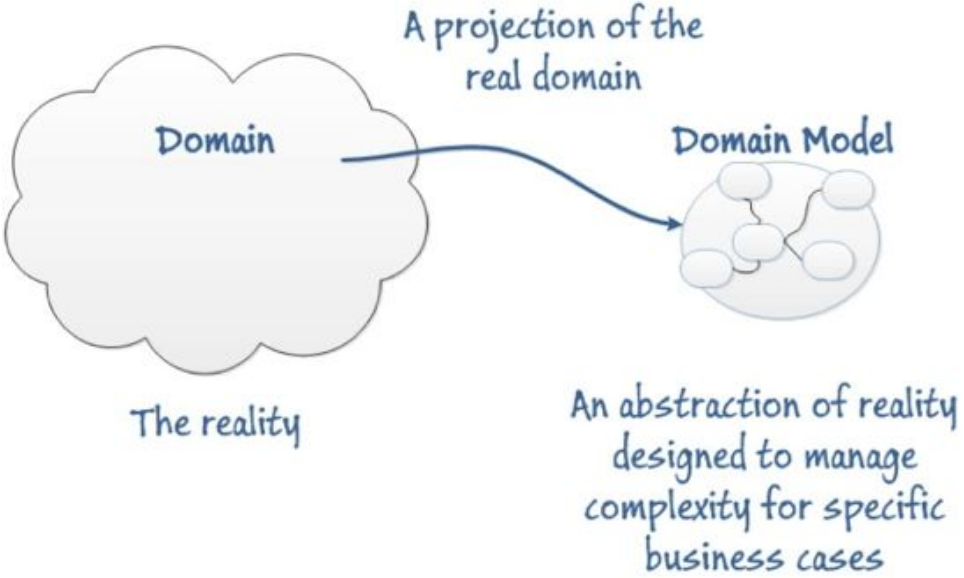
Objetivos

- Padrões DDD
  - Padrões Estratégicos
  - Padrões Táticos
- Compreender o Conceito de EventStorming
  - Domínios
  - Fronteiras e Contexto Delimitados (a.k.a *Bounded Context*)
  - Comandos
  - Agregadores
  - Entidades
  - Eventos
- **Hands-on**

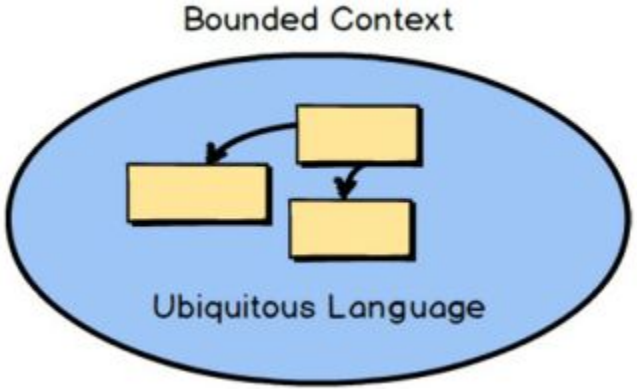

push state change with
DOMAIN EVENTS
REPOSITORIES
access with
SERVICES
access with
express model with
express change with
ENTITIES
encapsulate with
express identity with
act as root of
AGGREGATES
encapsulate with
express state & computation with
MODEL-DRIVEN DESIGN
VALUE OBJECTS
encapsulate with
model gives structure to
isolate domain expressions with
encapsulate with
LAYERED ARCHITECTURE
encapsulate with
FACTORIES
define model within
UBIQUITOUS LANGUAGE
cultivate rich model with
keep model unified by
CONTINUOUS INTEGRATION
names enter
BOUNDED CONTEXT
SHARED KERNEL
SHARED KERNEL
CORE DOMAIN
interdependent contexts form
overlap allied contexts through
CUSTOMER /SUPPLIER
relate allied contexts as
work in autonomous, clean
assess/overview relationships with
CONTEXT MAP
minimize translation
CONFORMIST
avoid overinvesting in
support multiple clients through
segregate the conceptual messes
free teams to go
OPEN HOST SERVICE
GENERIC SUBDOMAINS
translate and insulate unilaterally with
loosely couple contexts through
formalize
BIG BALL OF MUD
SEPARATE WAYS
ANTI-CORRUPTION LAYER
PUBLISHED LANGUAGE

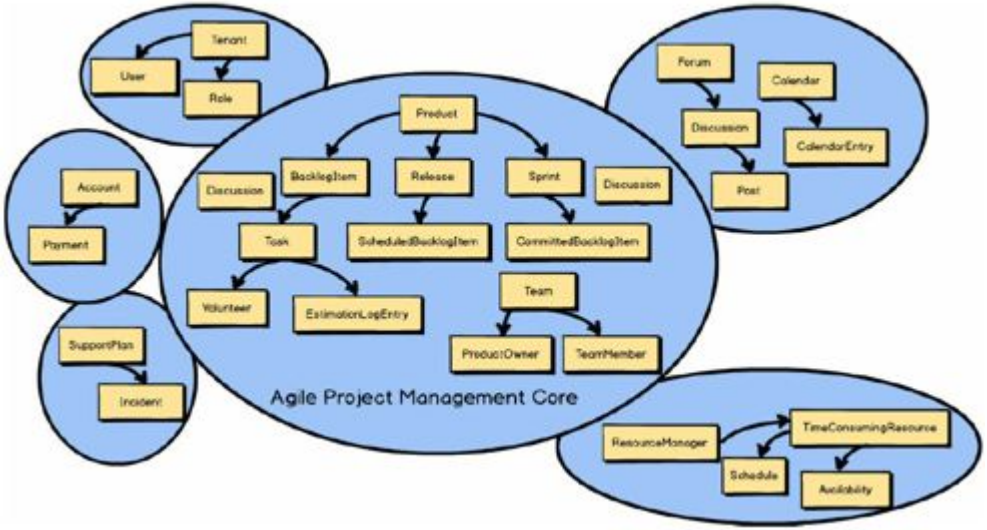
TACTICAL

STRATEGIC

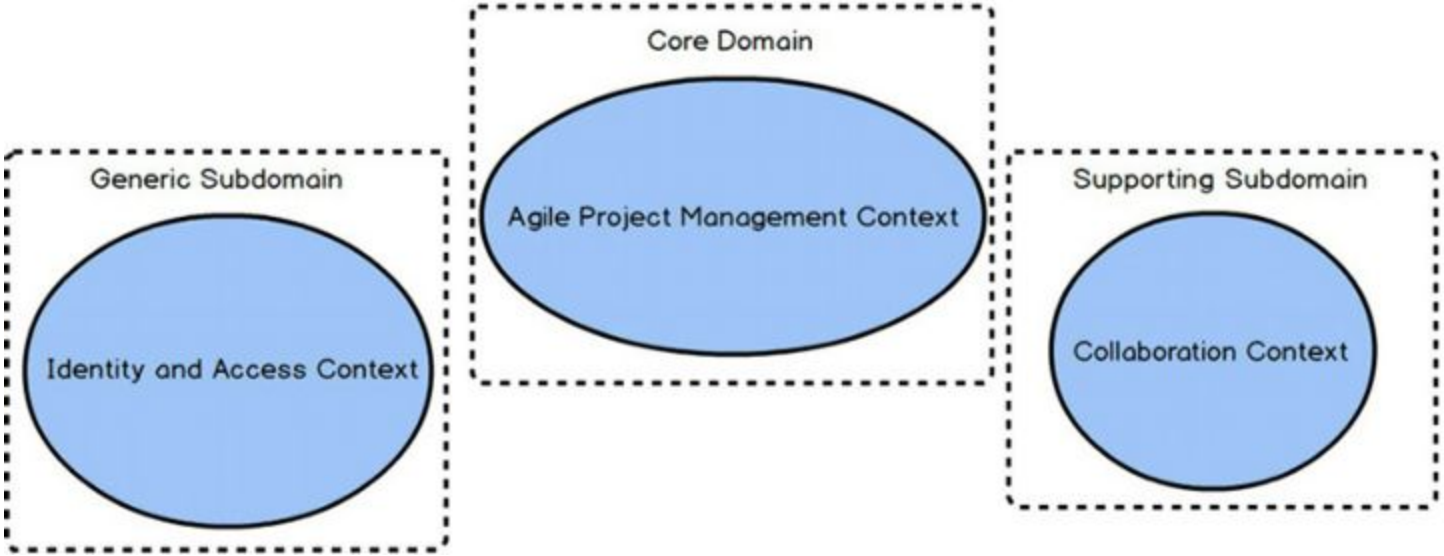

A projection of the real domain
Domain
Domain Model
The reality
An abstraction of reality designed to manage complexity for specific business cases

Bounded Context & Ubiquitous Language

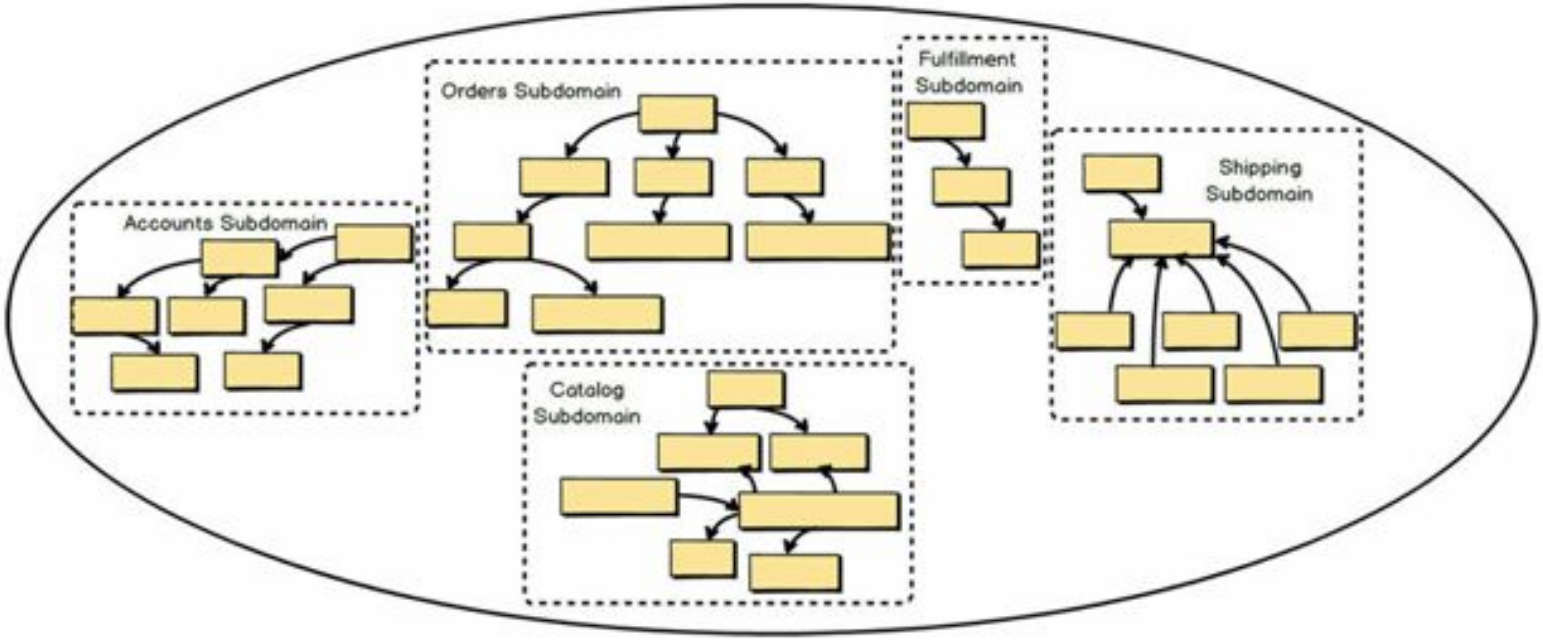




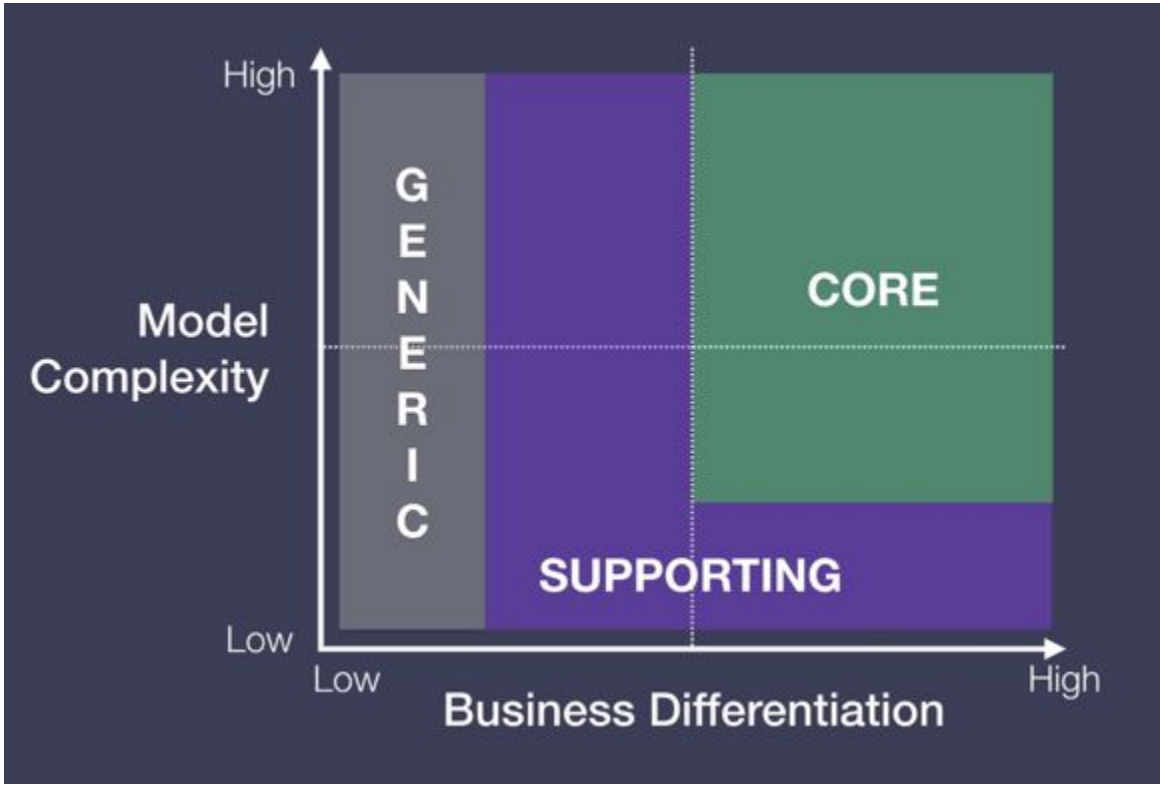
Tipos de Domínios



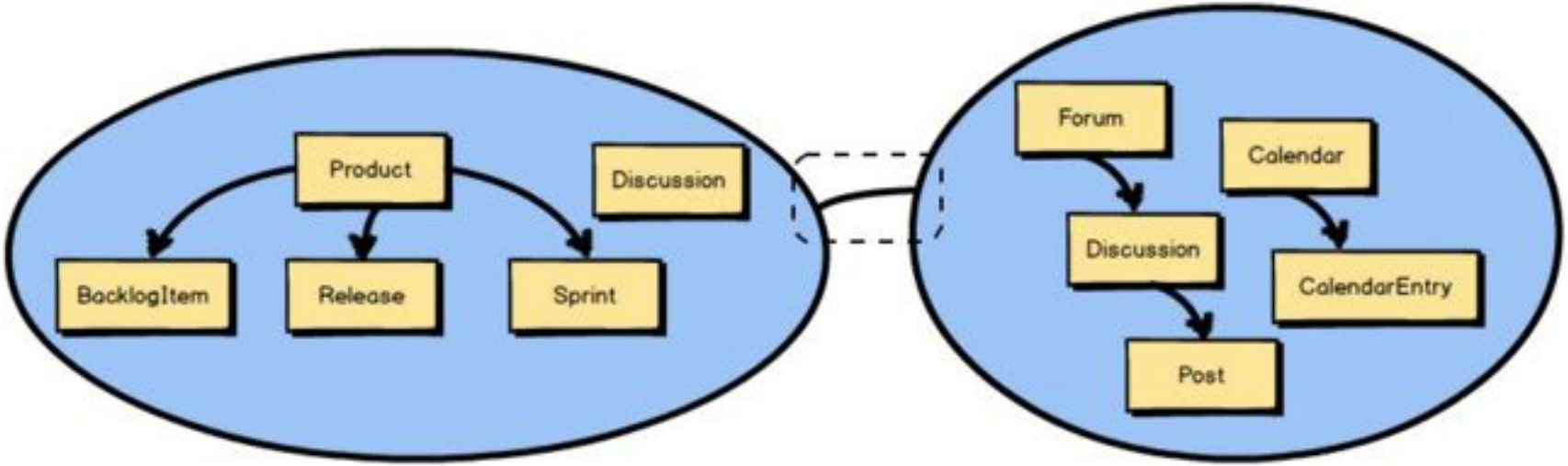
Domínios principais e secundários



Destilando os domínios

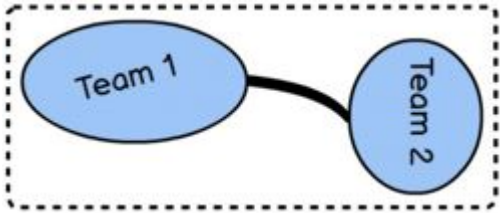


Mapeamento de Contexto (Context Mapping)

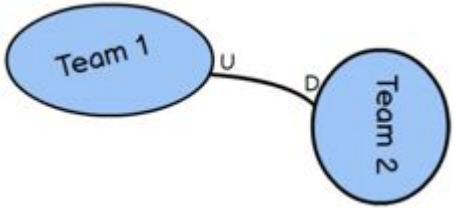


Relacionamentos entre Contextos Delimitados Partnership & Upstream/Downstream

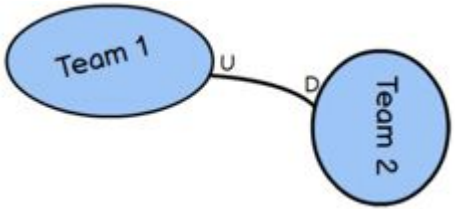
Existe uma ***relação de parceria entre duas equipes*** . Cada equipe é responsável por um Contexto Limitado



Descreve uma relação entre dois Contextos Limitados e respectivas equipes: o ***Fornecedor (Upstream) está a enviando (o U no diagrama) e o Cliente está a consumindo (o D no diagrama)***



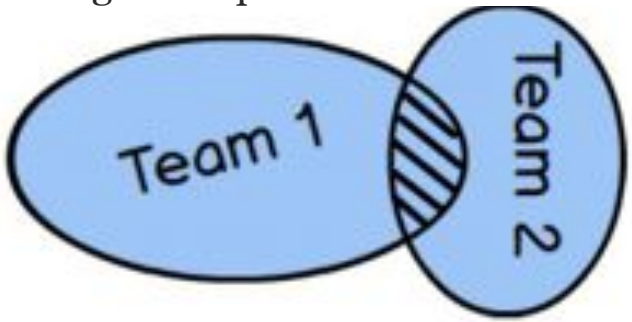
Upstream & Downstream



Shared Kernel

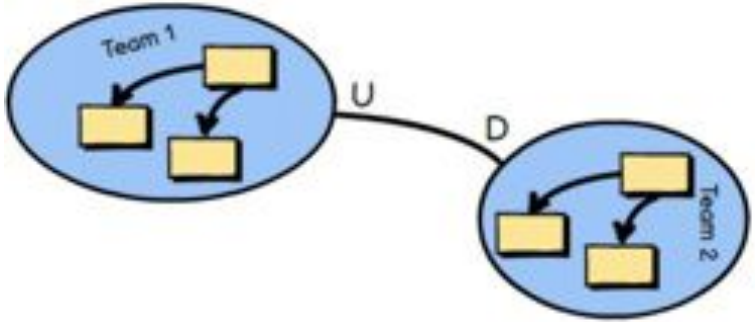
Um **Kernel Compartilhado** é um subconjunto de um modelo de domínio que duas equipes compartilham

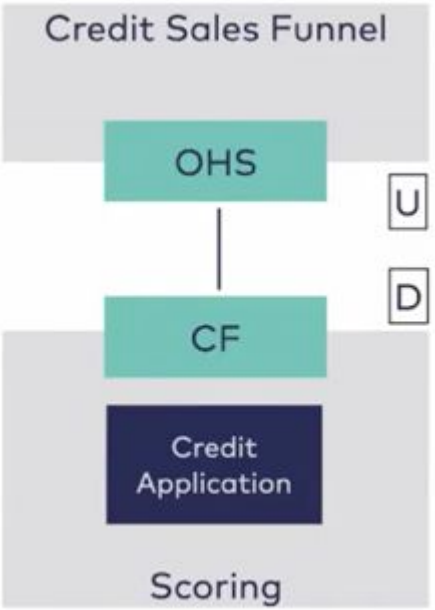
- Artefato fisicamente compartilhado entre duas equipes (Shared JARs, DB, ...)
- Alto grau de acoplamento requer uma grande quantidade de coordenação entre as equipes envolvidas



Conformist

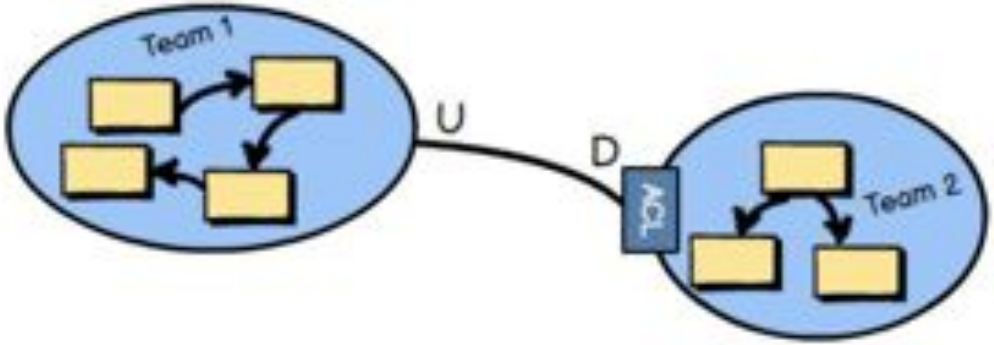
O conformista adere servilmente ***ao modelo upstream***

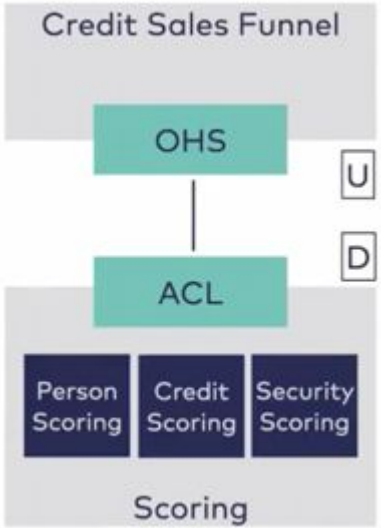




Anti-corruption Layer

A Camada Anticorrupção ***traduz um modelo para outro: (Acoplamento mais solto)*** - Transforma um modelo externo de outra equipe/contexto limitado/sistema para outro interno Reduz a quantidade de acoplamento para uma única camada A equipe que implementa a ACL é sempre downstream.

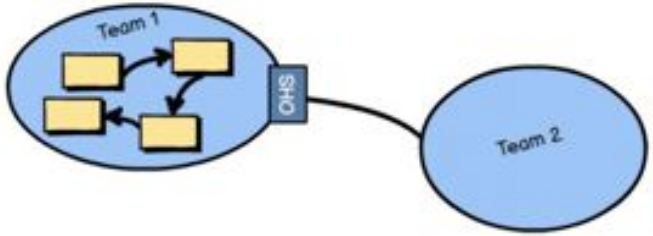


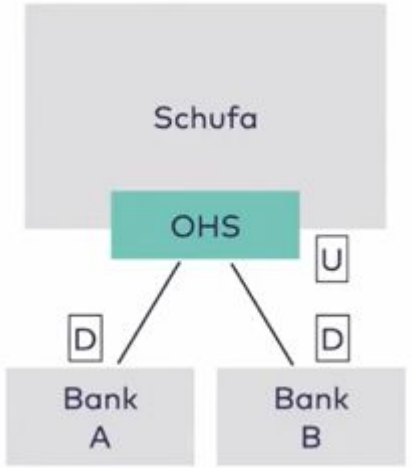


Open-Host

Define um protocolo ou interface que dá acesso ao seu Bounded Context como um conjunto de serviços

O ***serviço Open***





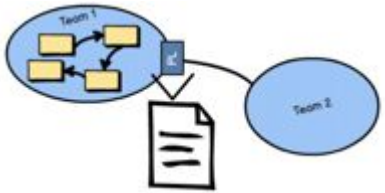
OPENAPI INITIATIVE

Publish Language

Uma linguagem bem documentada compartilhada entre contextos limitados:

- Cada contexto delimitado pode traduzir para dentro e para fora desse idior

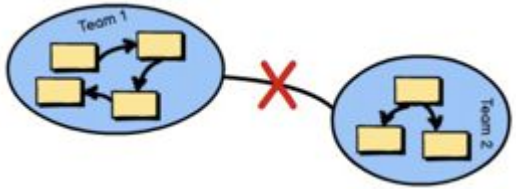
- Frequentemente combinado com o serviço de host aberto



A Linguagem Publicada pode ser definida com XML Schema, JSON

Separate Ways

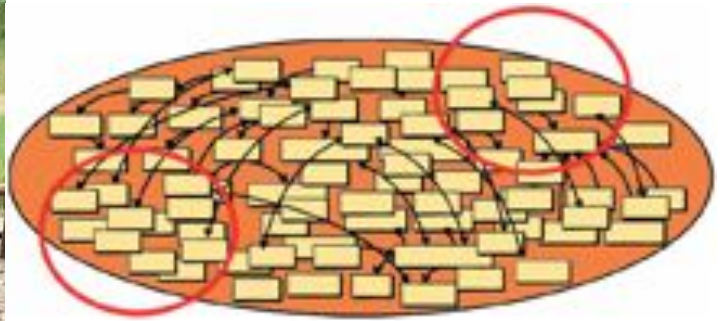
Um contexto limitado não tem conexões com outros:



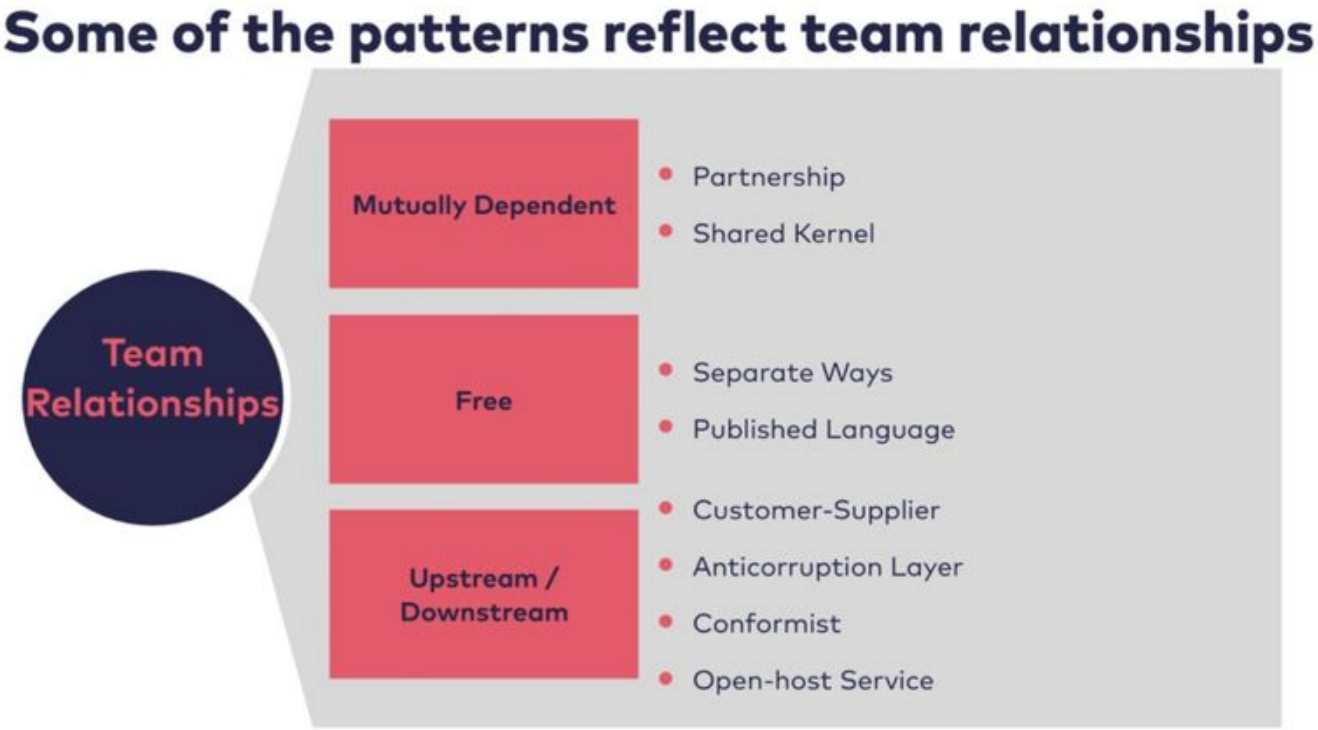
- Às vezes, a integração é muito cara ou demora muito
- As equipes escolhem caminhos separados para se concentrar em soluções especializadas

Talvez a funcionalidade que você procura não seja totalmente fornecida por nenhuma Linguagem Ubíqua.

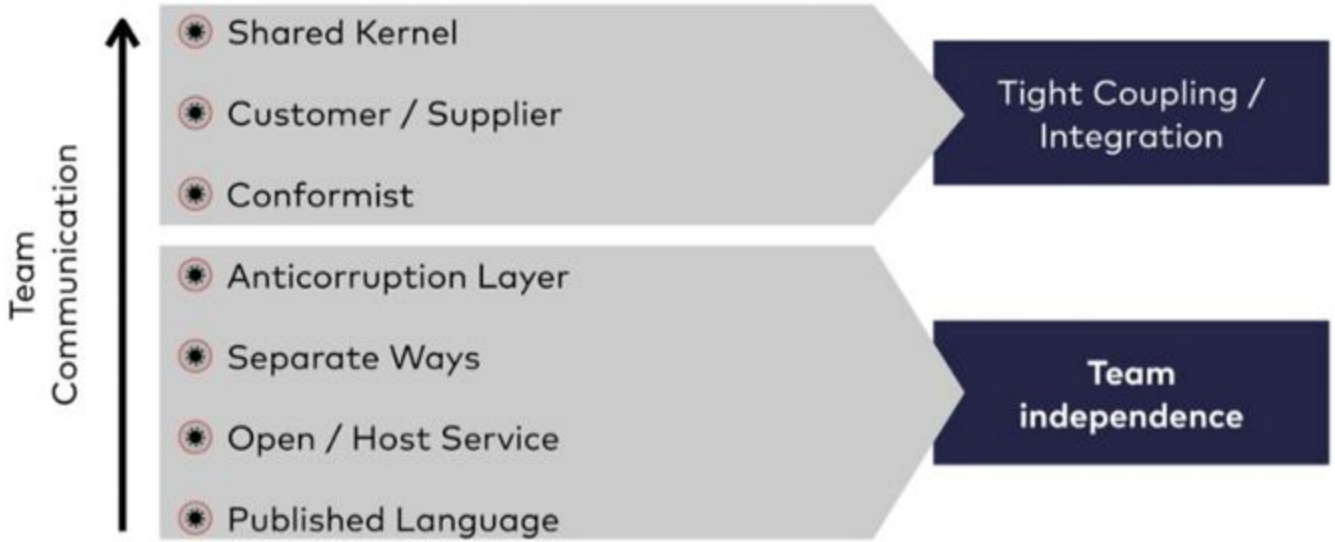
Big-ball of Mud



Relacionamento com as pessoas e times



Mind team communication



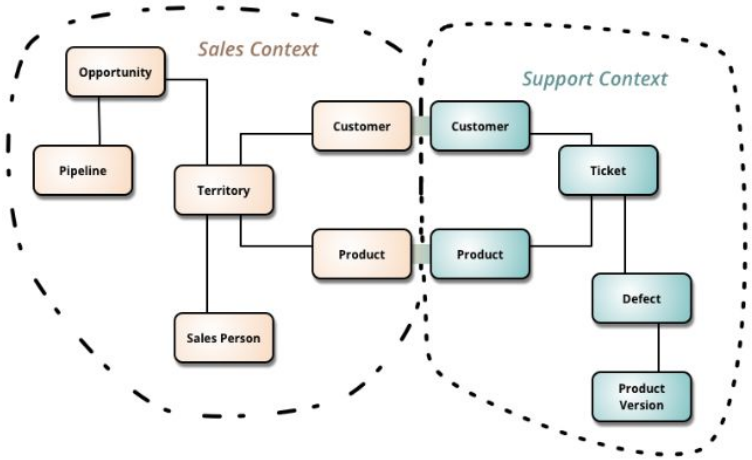
BoundedContext

15 January 2014

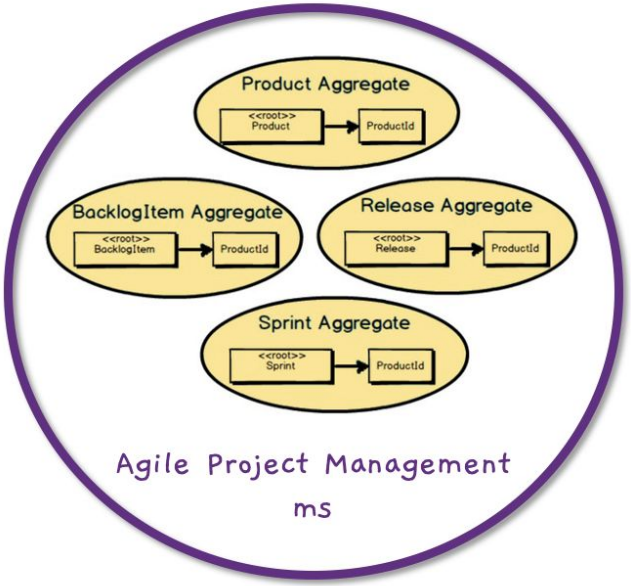
Martin Fowler

TEAM ORGANIZATION
REQUIREMENTS ANALYSIS
APPLICATION INTEGRATION
DOMAIN DRIVEN DESIGN

Bounded Context is a central pattern in Domain-Driven Design. It is the focus of DDD's strategic design section which is all about dealing with large models and teams. DDD deals with large models by dividing them into different Bounded Contexts and being explicit about their interrelationships.



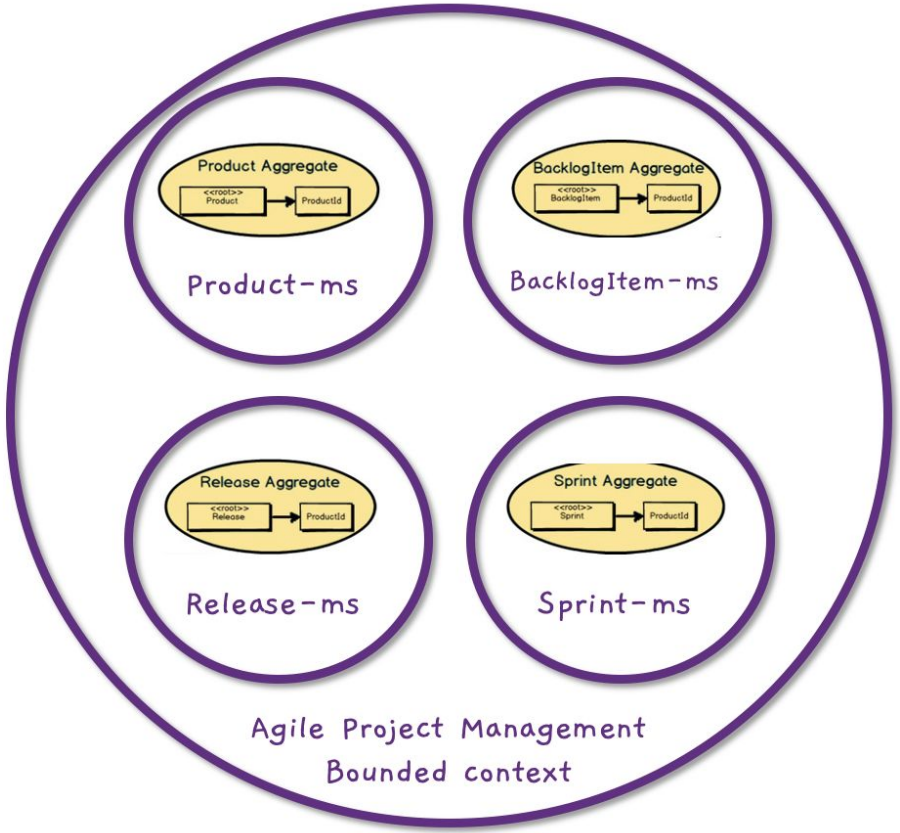
https://martinfowler.com/bliki/BoundedContext.html



1 Bounded context

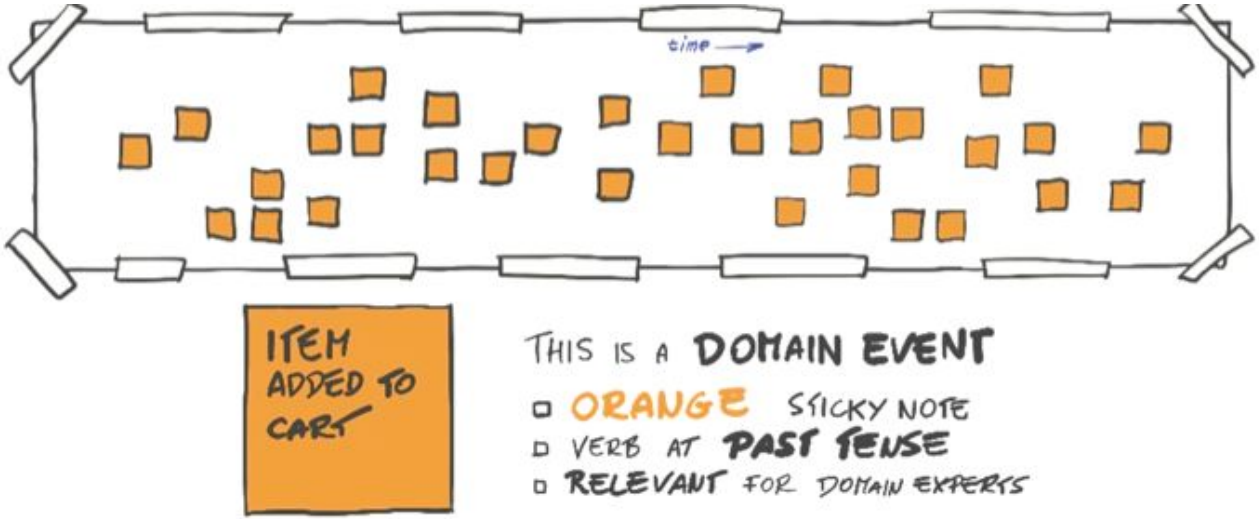
1 Micro-service

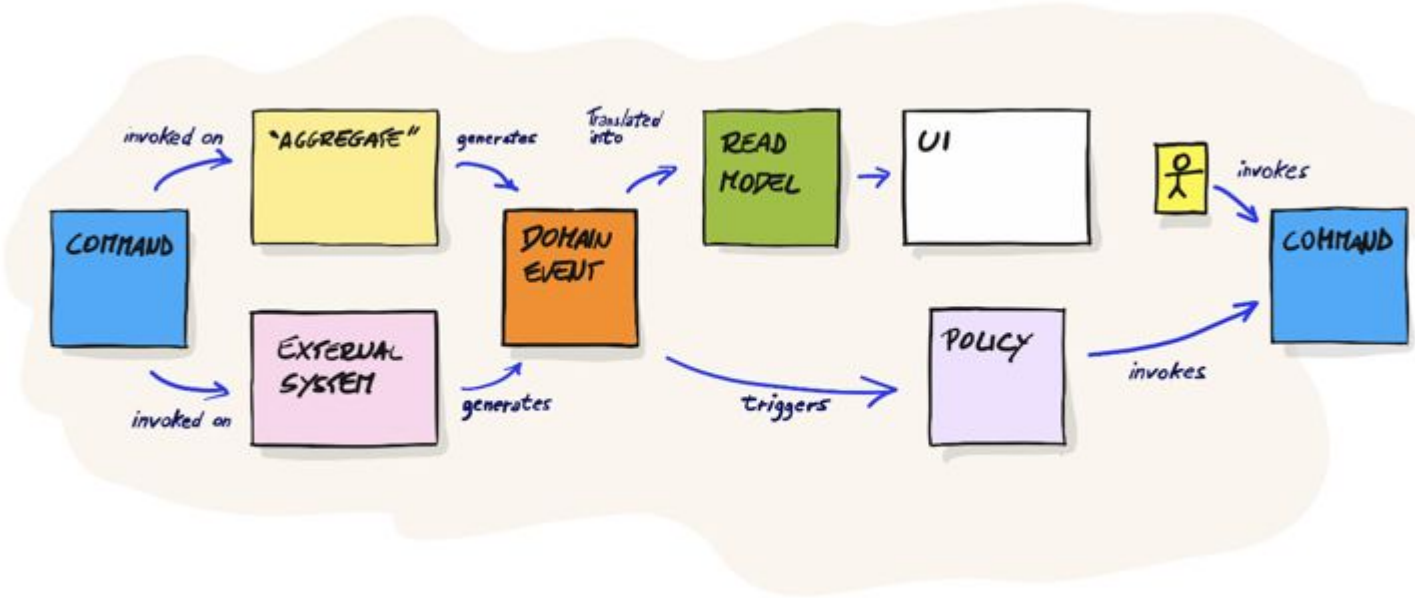
VS



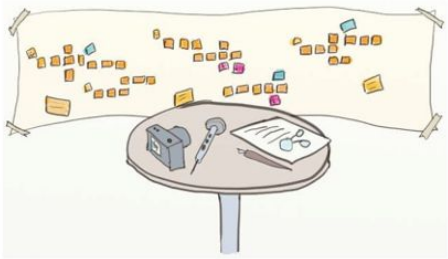
Same semantic boundary

Como descobrir Bounded Context ou Contextos Delimitados

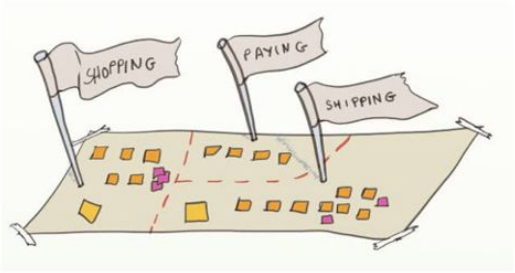



invoked on
"AGGREGATE"
generates
Translated into
READ MODEL
UI
invokes
COMMAND
DOMAIN EVENT
COMMAND
EXTERNAL SYSTEM
POLICY
invoked on
generates
triggers
invokes

Shared understanding



Context map



User stories

"When user X does Y, he wants to see Z, in order to ..."

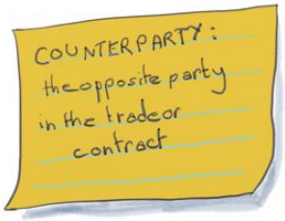
In / Out messages

Clear contracts

Next Actions

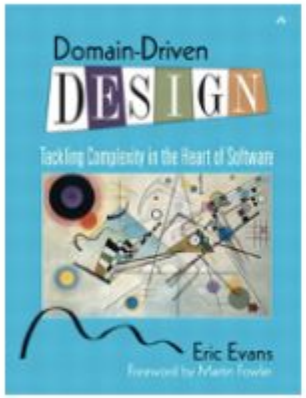
Decisions record

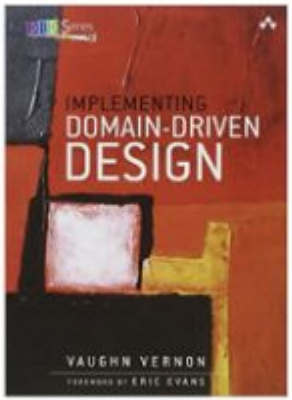
Captured domain definition



Referências

DDD. Disponível em <https://yoan-thirion.medium.com/domain-driven-design-re-distilled-685e75595a60>





Hands-on - EventStorming

1) Dividir em Equipes (Squads)
2) Usa a dinâmica de EventStorming para fazer o mapeamento de domínios de uma aplicação envolvendo os seguintes requisitos
    a) Realizar empréstimos bancários
    b) Ganhar e participar de promoções
    c) Receber notificação por celular
    d) Possibilitar a visualização de dados no celular