

Ordering Application

ARCHITECTURE

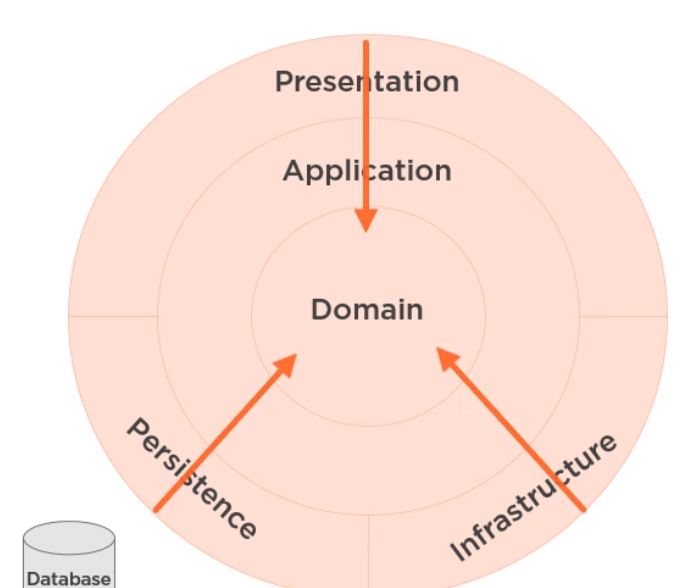
Bujji Krishna R B

# Goals

* To build a scalable, reliable, fault tolerant Order processing application.
* It should not be tied to any particular framework. It should be easy to switch frameworks without making any big changes.
* It should be easy to extend and maintain.
* It should be easily testable (unit/integration/functional). Enable test driven development.
* Easy to build a User Interface which uses this application.

## Application Architecture.

To achieve the above goals, the application architecture chosen for building this product was: Domain Centric Architecture



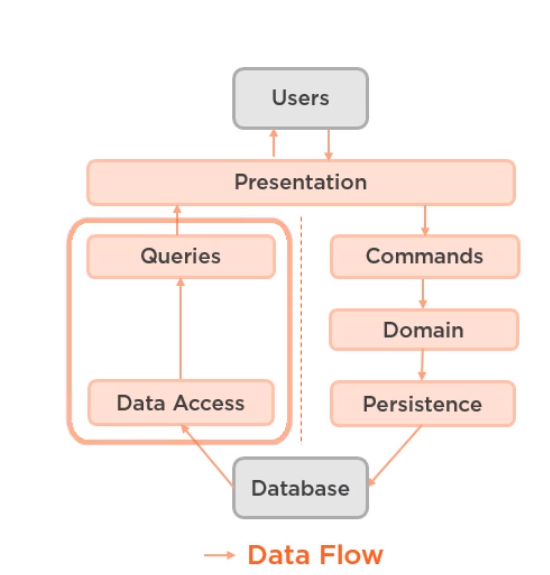
**Pros:**

* Focus on domain.
* Less Coupling.
* Allows for Domain Driven Design.
* Easy to create Commands and Queries (CQRS). (More efficient design, optimized performance, easy to incorporate Event sourcing).
* Natural enabler for Micro Services with clear defined boundaries.

**Cons:**

* Requires more thought.
* Initial Higher cost.

# CQRS: (Single Database Implementation)

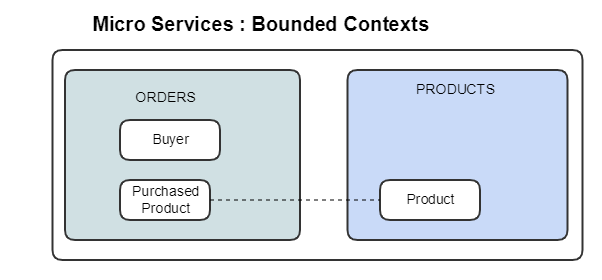


# Component Diagram

# 

# Micro services :

The two bounded contexts are: Orders and Products.



# Testable architecture:

