

# Refocusing your domain

Nikolaos Petros Triantafyllidis

[ntrianta@cern.ch](mailto:ntrianta@cern.ch)

IT-CM-IS

**Myself**

# The Issue

# **1. Unclear Boundaries**

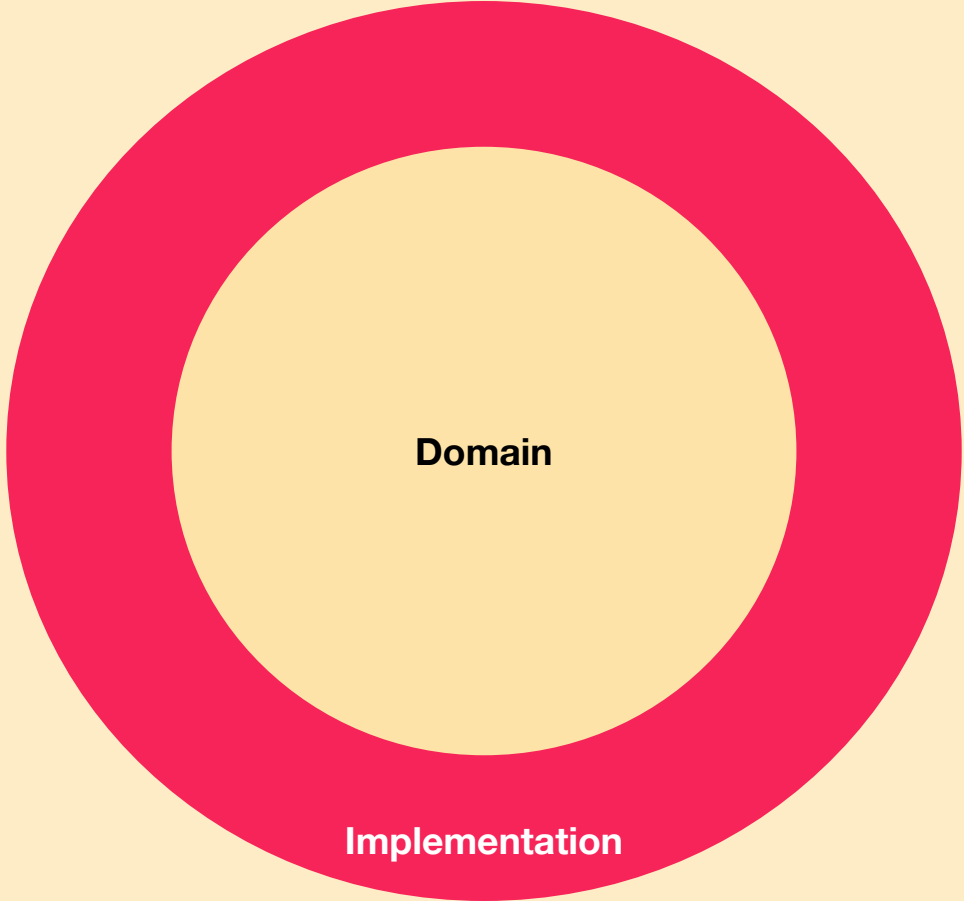
# **2. Cohesion & Coupling**

# **3. Adaptability**

# **4. Ease of testing**

# **Clean Architecture**





**Domain**

**Implementation**

**S**ingle responsibility

**O**pen/closed

**L**iskov substitution

**I**nterface segregation

**D**ependency inversion

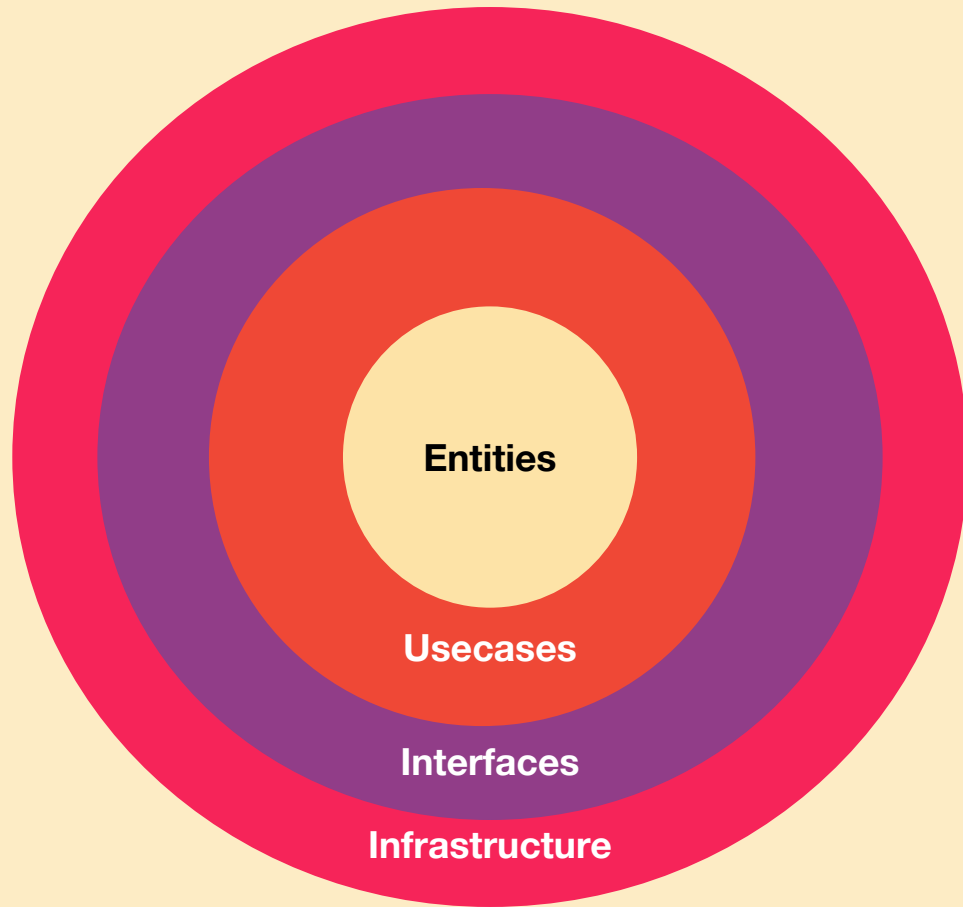
**Single responsibility**

**Open/closed**

**Liskov substitution**

**Interface segregation**

**Dependency inversion**

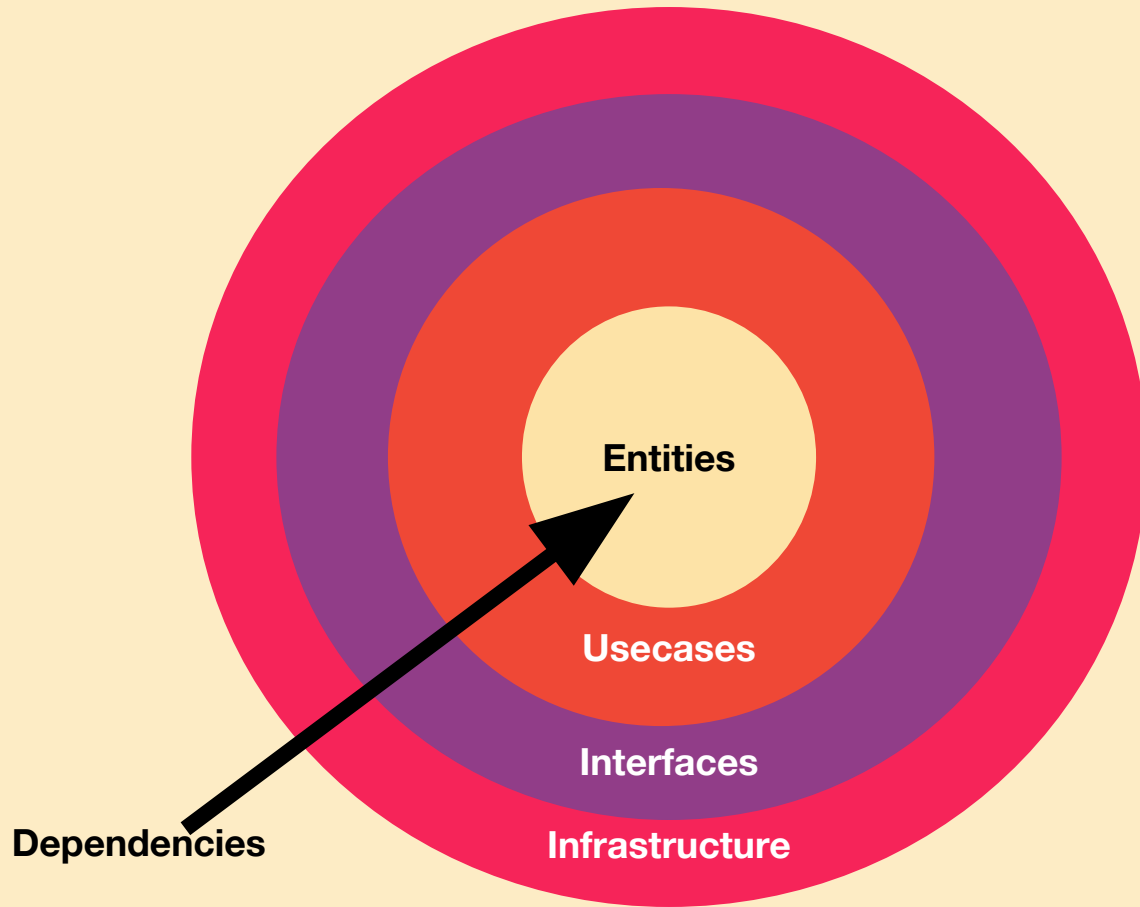


**Entities**

**Usecases**

**Interfaces**

**Infrastructure**



**Single responsibility**

**Open/closed**

**Liskov substitution**

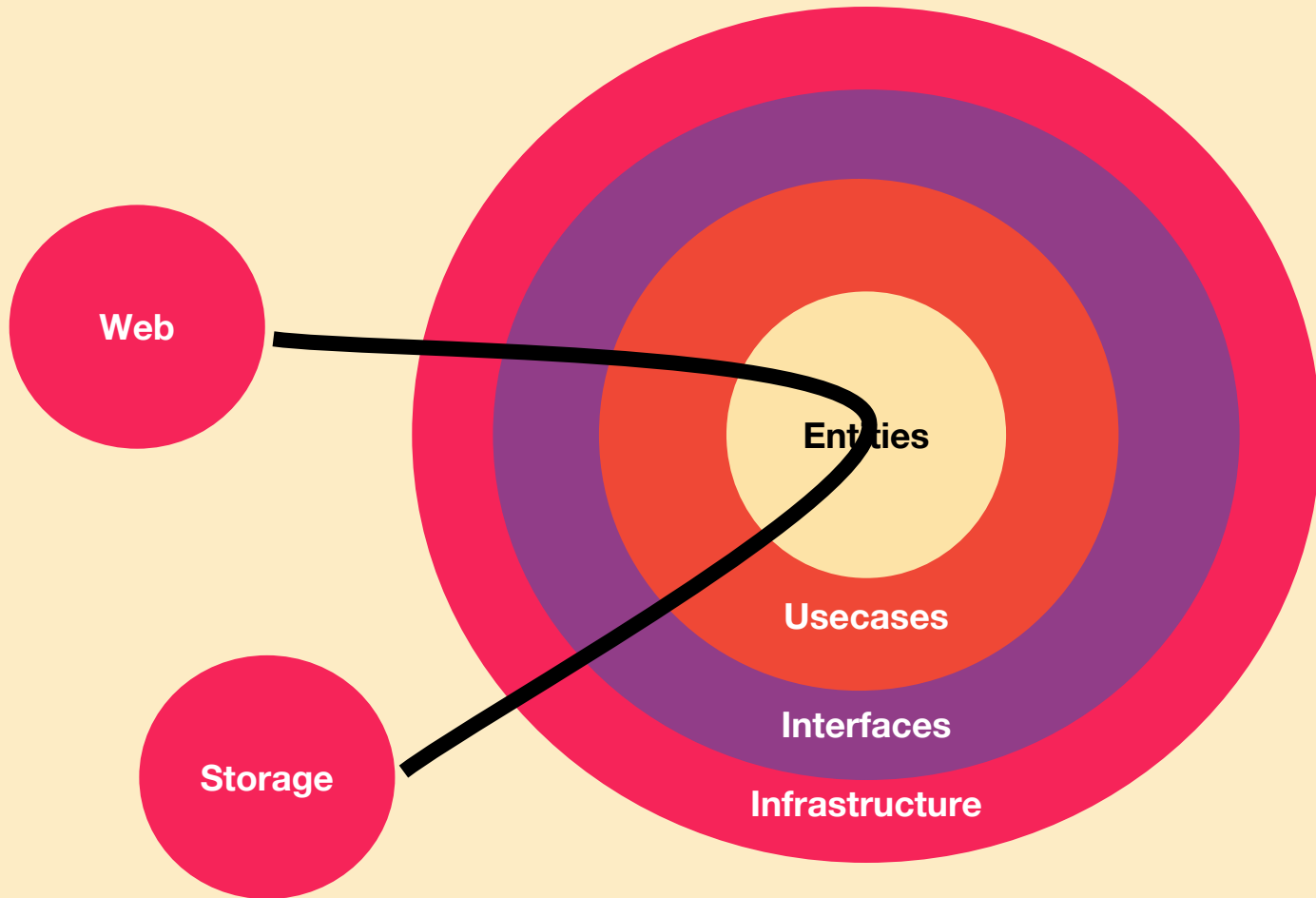
**Interface segregation**

**Dependency inversion**

	<b>Entities</b>	<b>Usecases</b>	<b>Interfaces</b>	<b>Infrastructure</b>
<b>Business</b>	Specific	Specific	Agnostic	Agnostic
<b>Application</b>	Agnostic	Specific	Specific	Agnostic

**Some Code**





	<b>Entities</b>	<b>Usecases</b>	<b>Interfaces</b>	<b>Infrastructure</b>
<b>Repositories</b>	Define	Define/Use	Implement	---
<b>Interactors</b>	---	Implement	Define/Use	---
<b>Input Handlers</b>	---	---	Implement	Use
<b>Output Handlers</b>	---	Use	Define	Implement

**The Hmm...**

# 1. Terminology

## **2. Code repetition**

# **3. Functionality**

**Merci!**

