

SIDION

Zuhören. Analysieren. Beraten.

ONION ARCHITECTURE

Mögen Sie Ihren Microservice
lieber mit oder ohne Zwiebeln?

Azmir Abdi

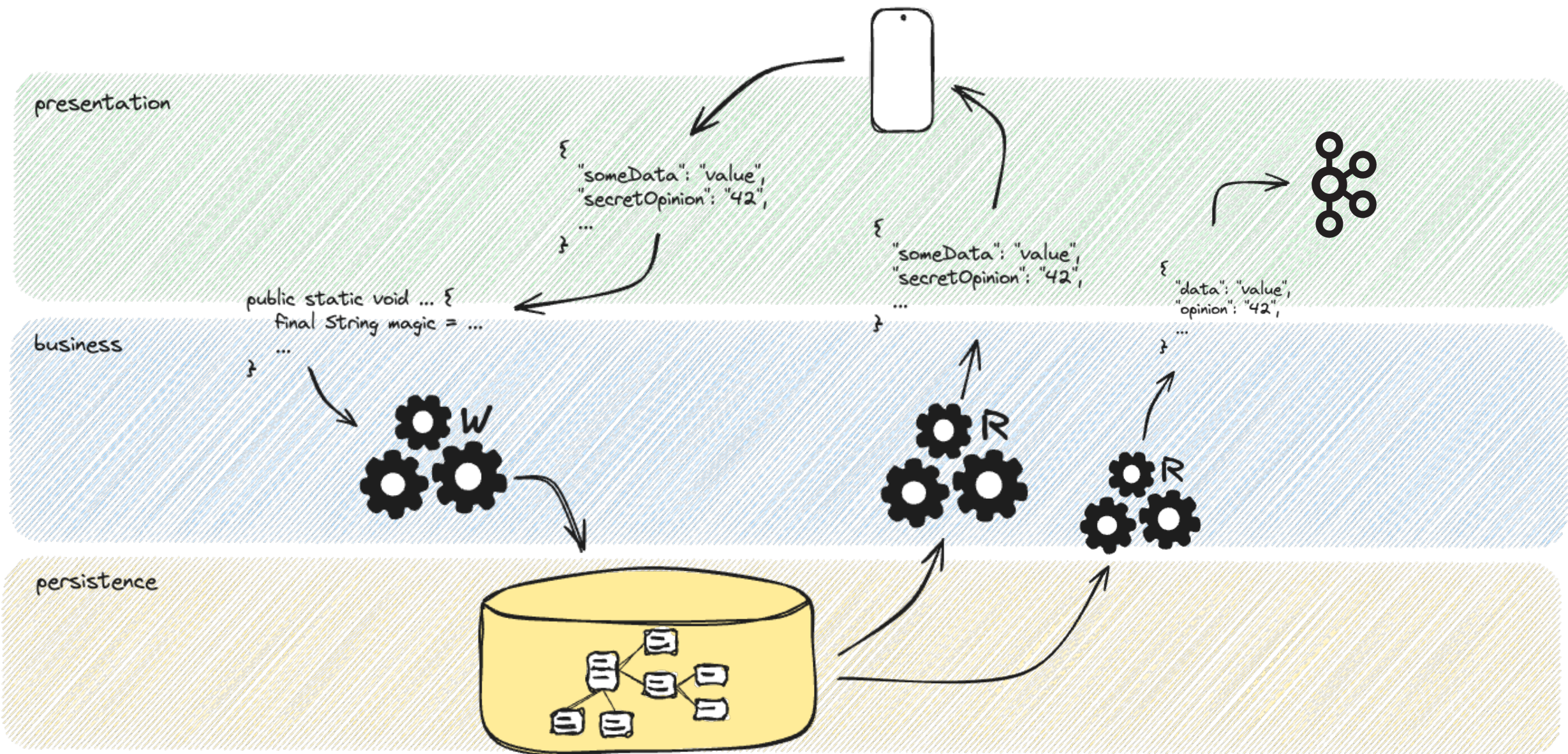
 **JAVA FORUM** 2024
stuttgart

_EAT ...

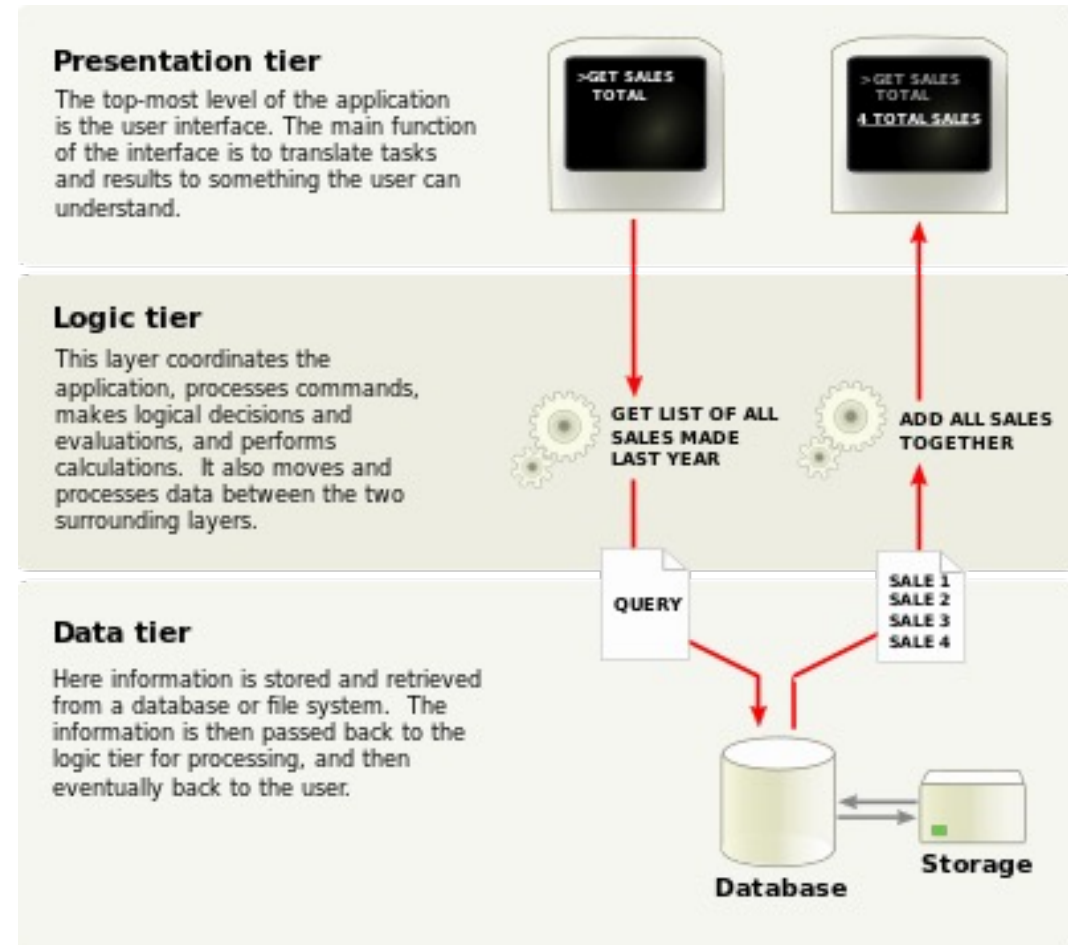
_SLEEP ...

_CODE ...

_REPEAT ...

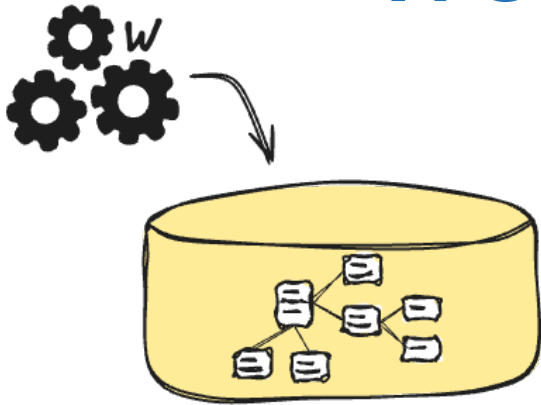


LAYERED ARCHITECTURE



Quelle: Wikipedia

NORMALIZED MODEL !!!

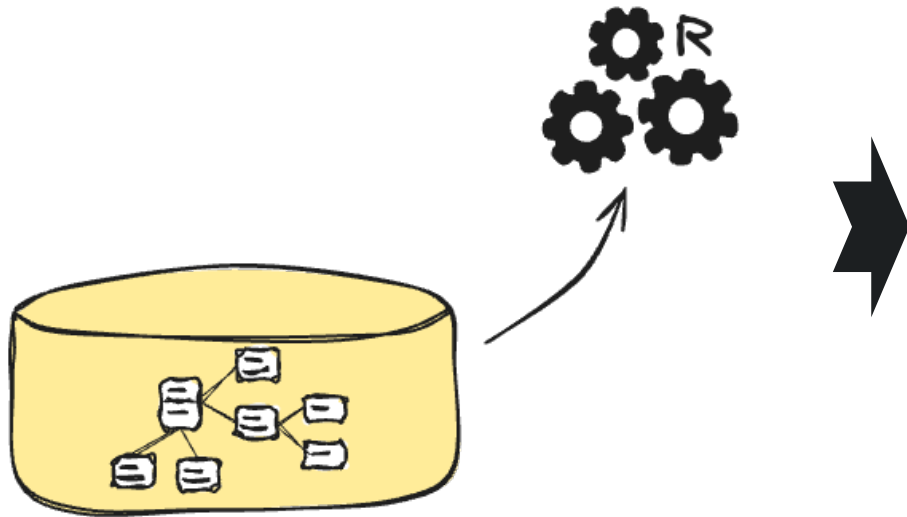


```
public class Kunde {  
  
    private CrmId crmId;  
    private Name name;  
    private Geburtstag geburtstag;  
    private Email email;  
  
}
```



```
@Entity  
public class CustomerJpa {  
  
    @Id  
    @GeneratedValue(strategy = GenerationType.AUTO)  
    private Long id;  
    private String firstName;  
    private String lastName;  
    @Transient  
    private LocalDate birthday;  
    private String crmId;  
    private String email;  
    private LocalDateTime createdAt;  
    @Version  
    private Integer version;  
  
}
```

ONE MODEL FITS ALL?



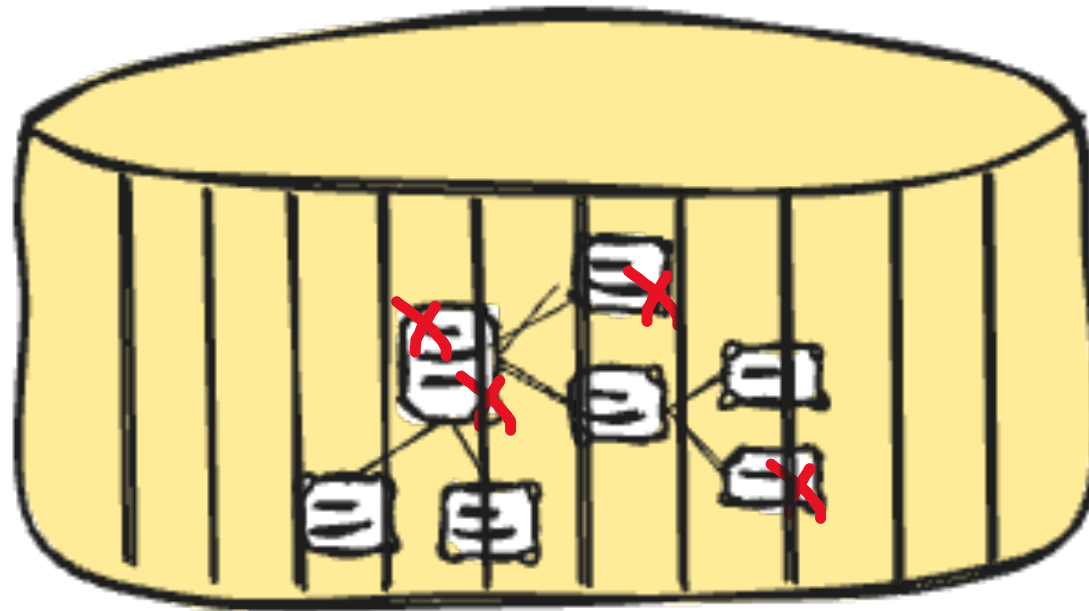
```
@Entity
public class CustomerJpa {

    @JsonIgnore @JsonbTransient @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private Long id;
    private String firstName;
    private String lastName;
    @Transient
    private LocalDate birthday;
    private String crmId;
    private String email;
    @JsonFormat(shape = JsonFormat.Shape.STRING,
pattern = "yyyy-MM-dd HH:mm a z")
    private LocalDateTime createdAt;
    @Version @JsonIgnore @JsonbTransient
    private Integer version;

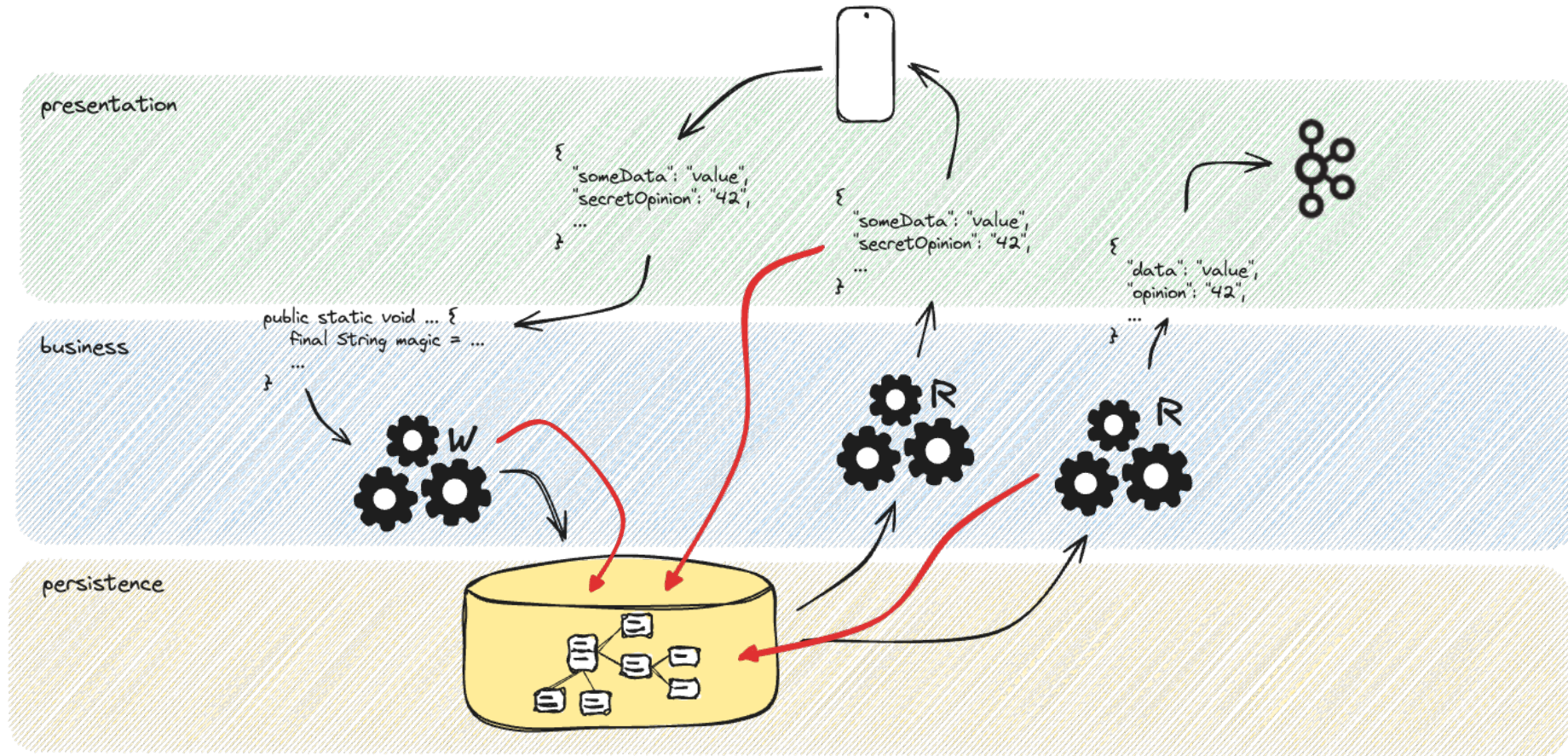
}
```

BUSINESS VS. TECHNOLOGY

A TECH-INFUSED MODEL !!!



COUPELED LAYERS !!!



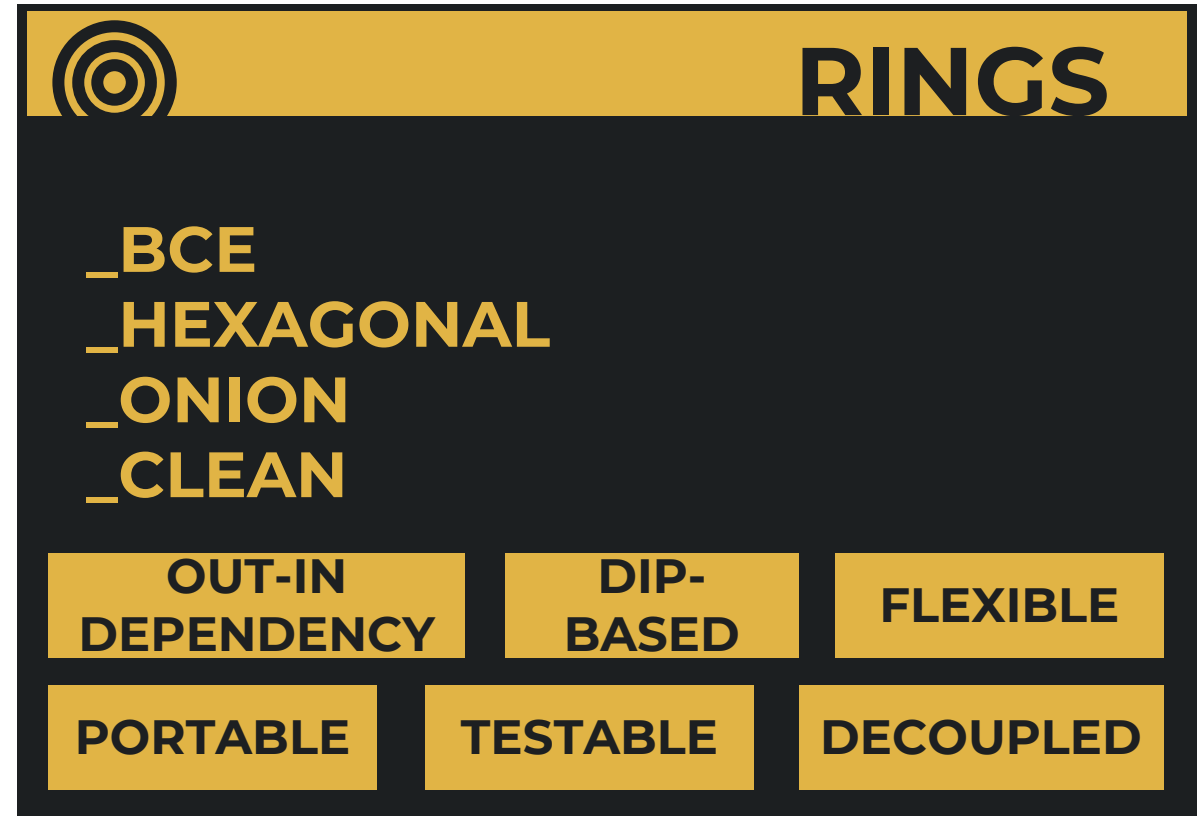
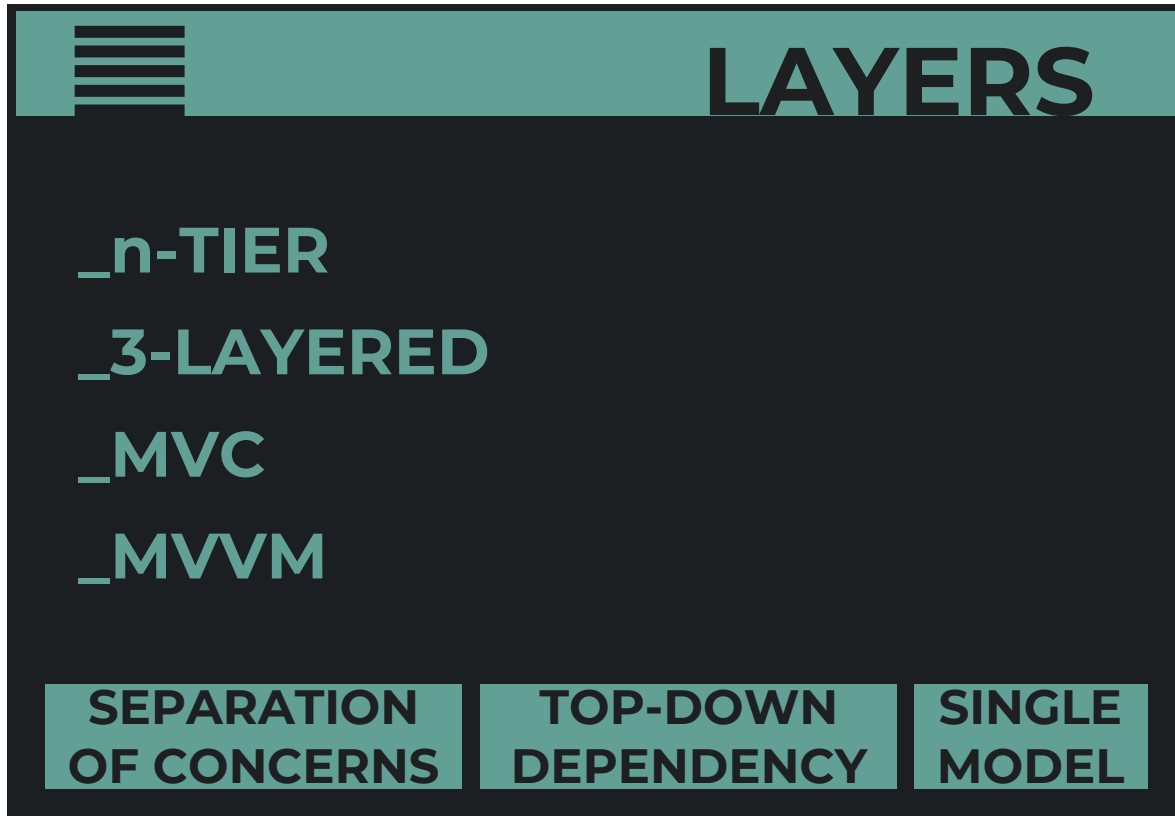


BRAUCHT ES

EINEN ANDEREN

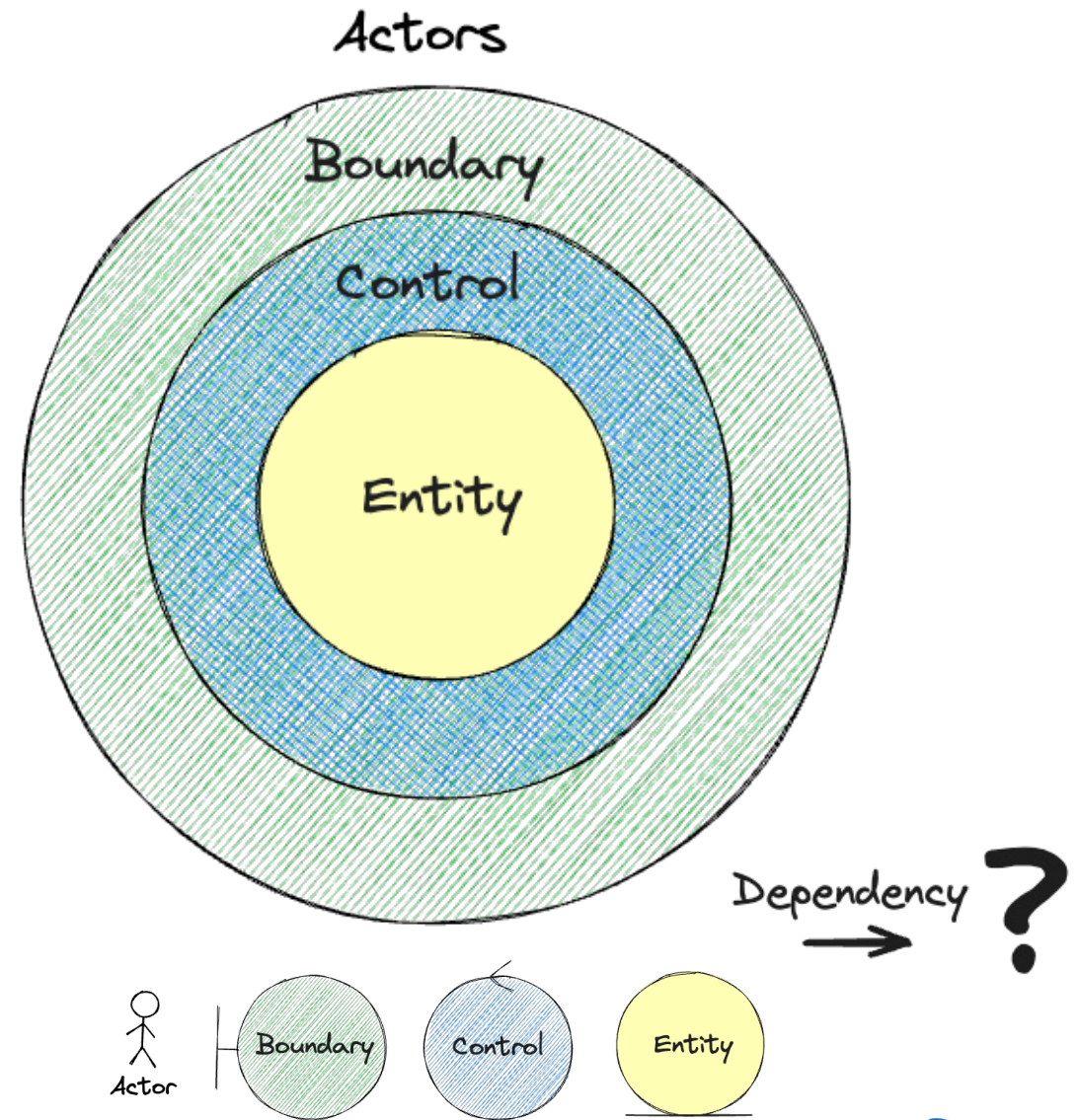
SCHNITT?

SCHNITTE



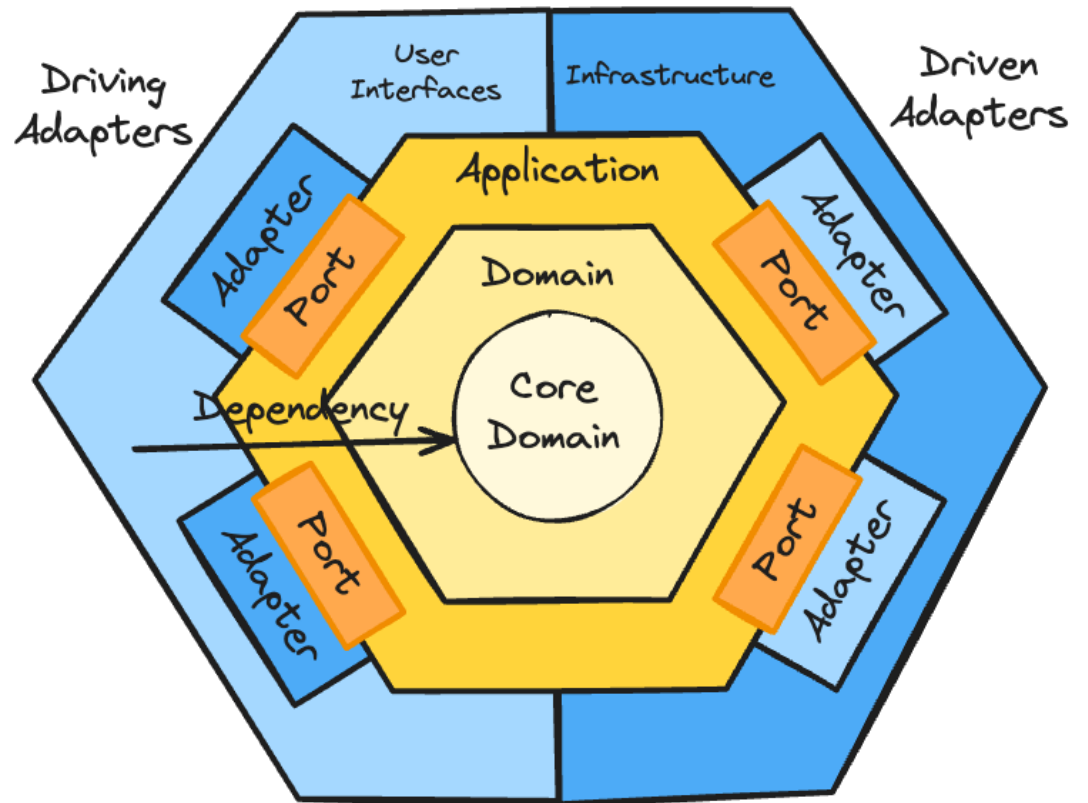
BOUNDARY CONTROL ENTITY (BCE)

* 1992 | Ivar Jacobson



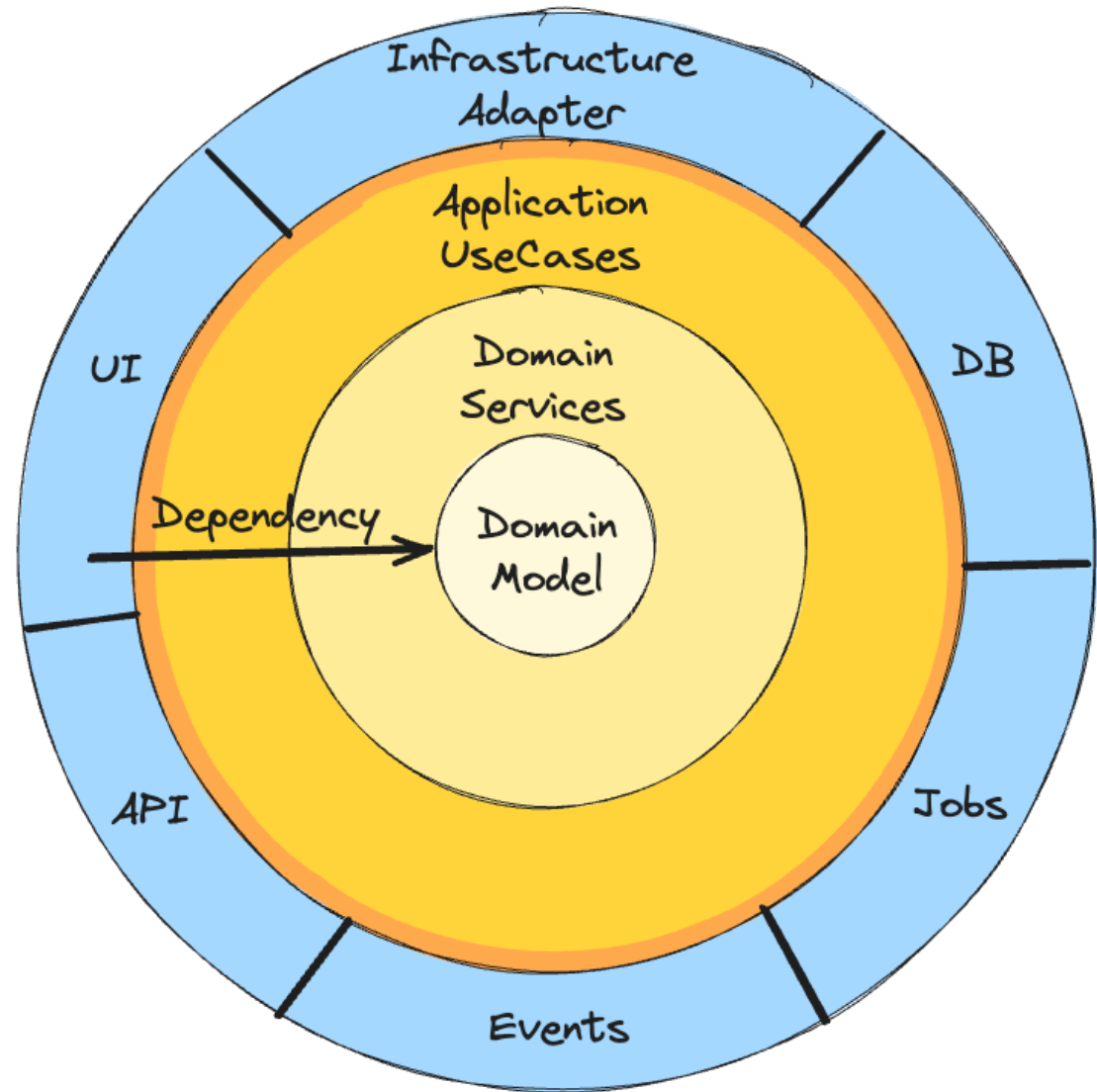
HEXAGONAL (PORTS & ADAPTERS)

* 2005 | Alistair Cockburn



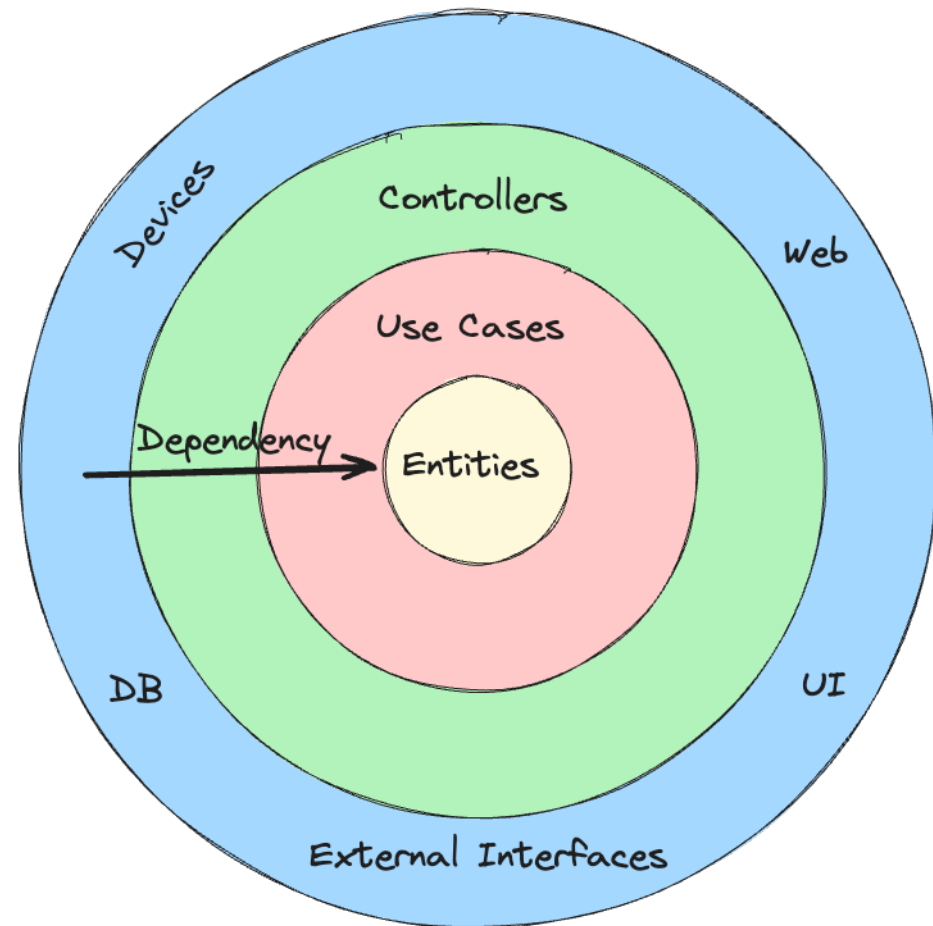
ONION ARCHITECTURE

* 2008 | Jeffrey Palermo



CLEAN ARCHITECTURE

* 2012 | Robert C. Martin



- Enterprise Business Rules
- Application Business Rules
- Interface Adapters
- Frameworks & Drivers

```

  ▾ de.sidion.onion
    ▾ customer
      ▾ adapter
        > crm
        > database
        > event
        > restapi
      > application
    ▾ domain
      > model
      > service

```



DEMO



<https://gitlab.com/sidion/vortraege/2024/javaforumstuttgart/onion>

WANN SETZE ICH ONION EIN?

PRESENTATION

**UNTERSCHIEDLICHE
MODELLE**

DOMAIN

PERSISTENCE

WANN SETZE ICH ONION EIN?

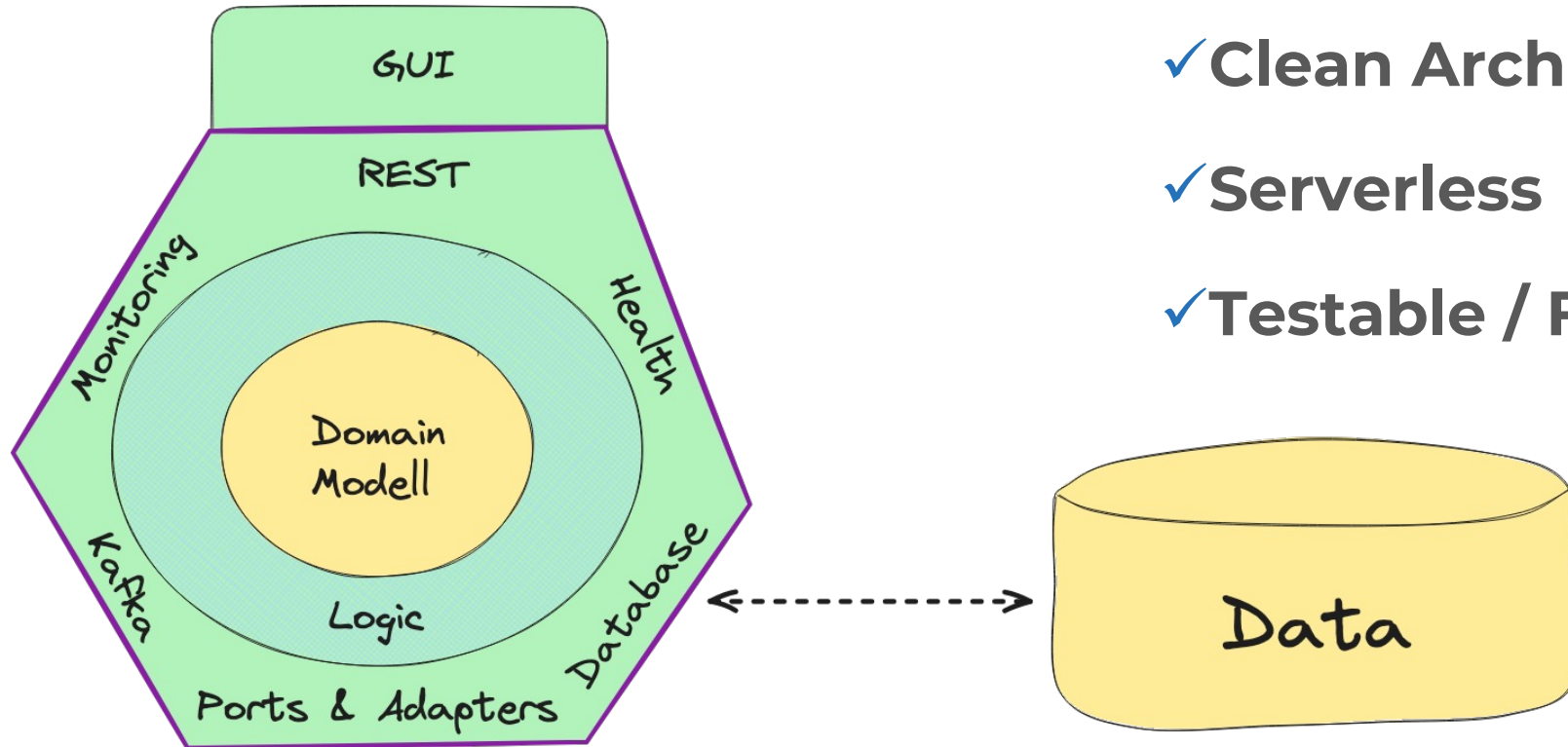
**DOMAIN MODELL
UNABHÄNGIGKEIT**

WANN SETZE ICH ONION EIN?

**TECHNOLOGIE
UNABHÄNGIGKEIT**

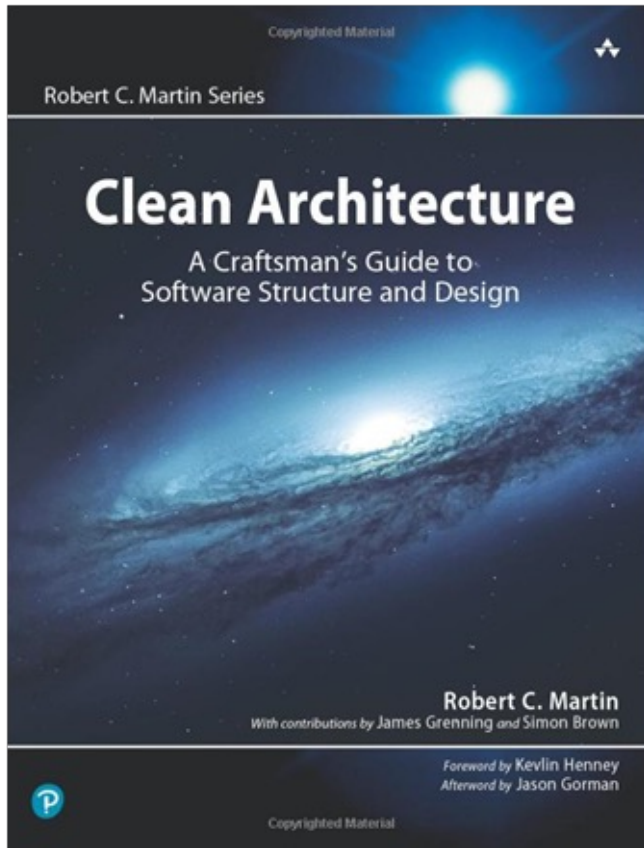
ONION ARCHITECTURE

VORTEILE



- ✓ Technology Agnostic
- ✓ Clean Architecture
- ✓ Serverless
- ✓ Testable / Flexible

QUELLEN



- <https://ccd-akademie.de/en/clean-architecture-vs-onion-architecture-vs-hexagonale-architektur/>
- <https://romanglushach.medium.com/understanding-hexagonal-clean-onion-and-traditional-layered-architectures-a-deep-dive-c0f93b8a1b96>
- <https://herbertograca.com/2017/11/16/explicit-architecture-01-ddd-hexagonal-onion-clean-cqrs-how-i-put-it-all-together/>
- <https://medium.com/ssense-tech/hexagonal-architecture-there-are-always-two-sides-to-every-story-bc0780ed7d9c>
- <https://www.happycoders.eu/de/software-craftsmanship/hexagonale-architektur/>
- <https://vaclavkosar.com/software/Boundary-Control-Entity-Architecture-The-Pattern-to-Structure-Your-Classes>
- <https://blog.cleancoder.com/uncle-bob/2012/08/13/the-clean-architecture.html>



FRAGEN

Mitdenken

Mitgestalten

Mitarbeiten

bei sidion

✉ azmir.abdi@sidion.de

[in https://www.linkedin.com/in/azmir-abdi/](https://www.linkedin.com/in/azmir-abdi/)



**VIELEN
DANK**