

DOMAIN DRIVEN DESIGN

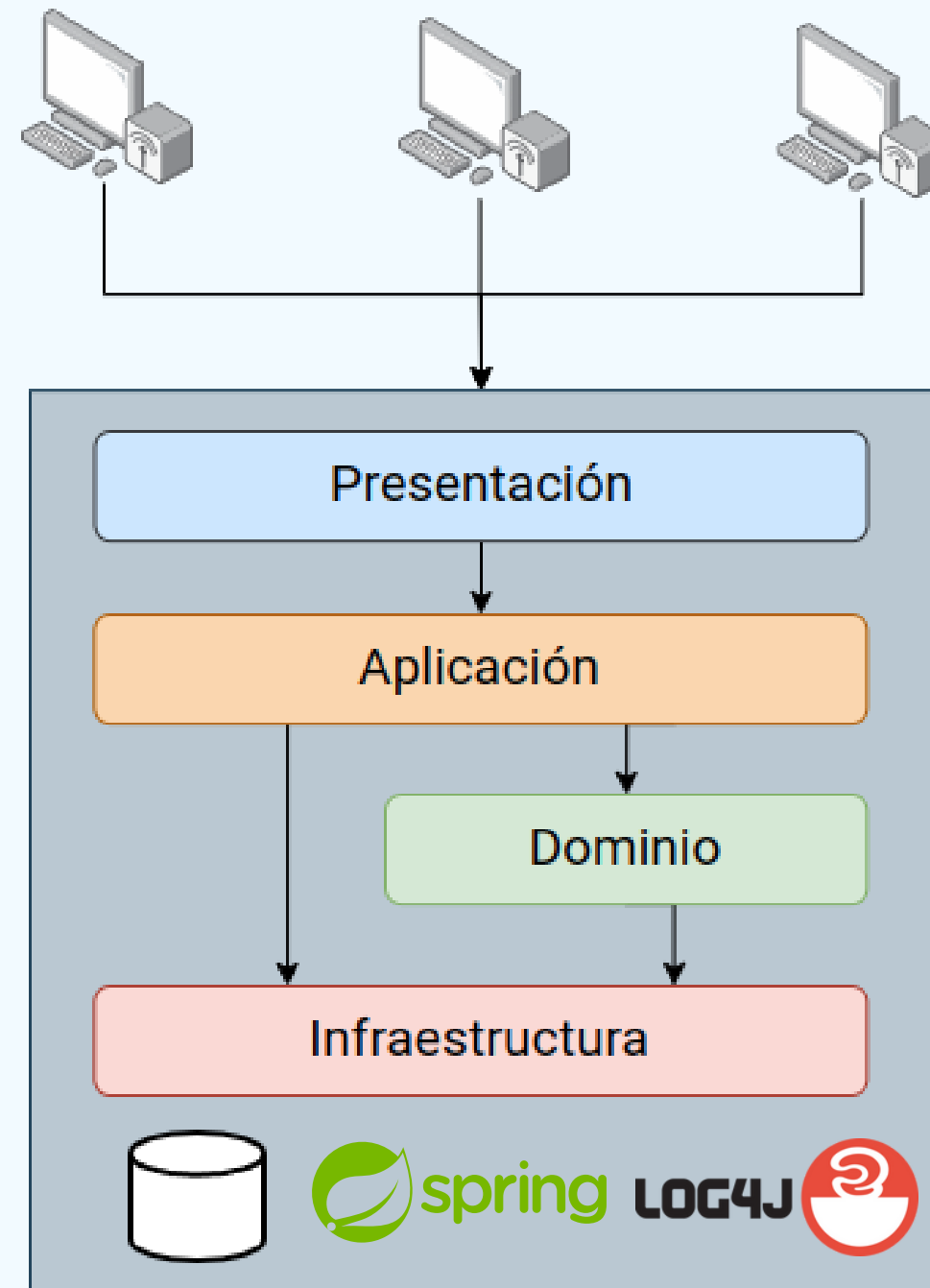
RESUMEN

Daniel Blanco Calviño

LA IMPORTANCIA DEL DOMINIO

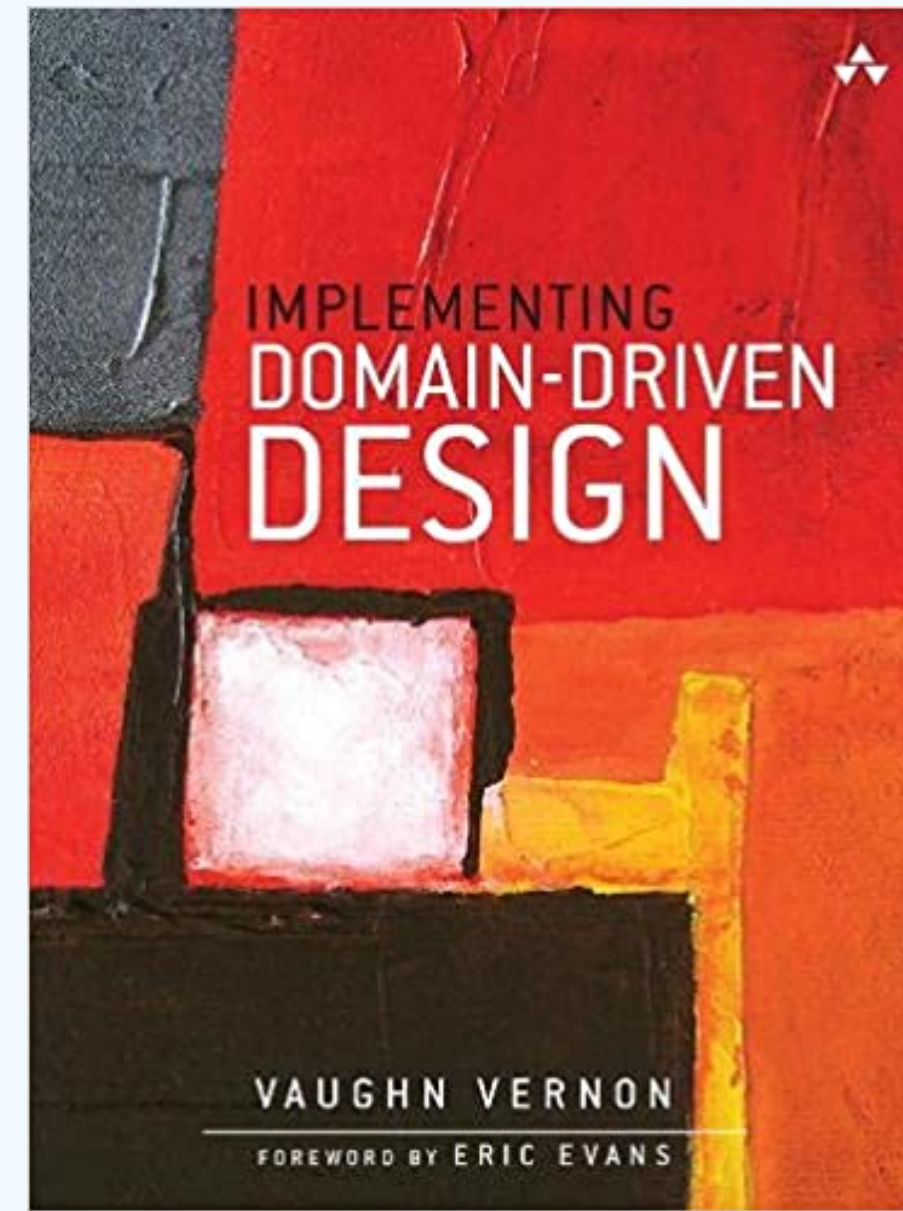
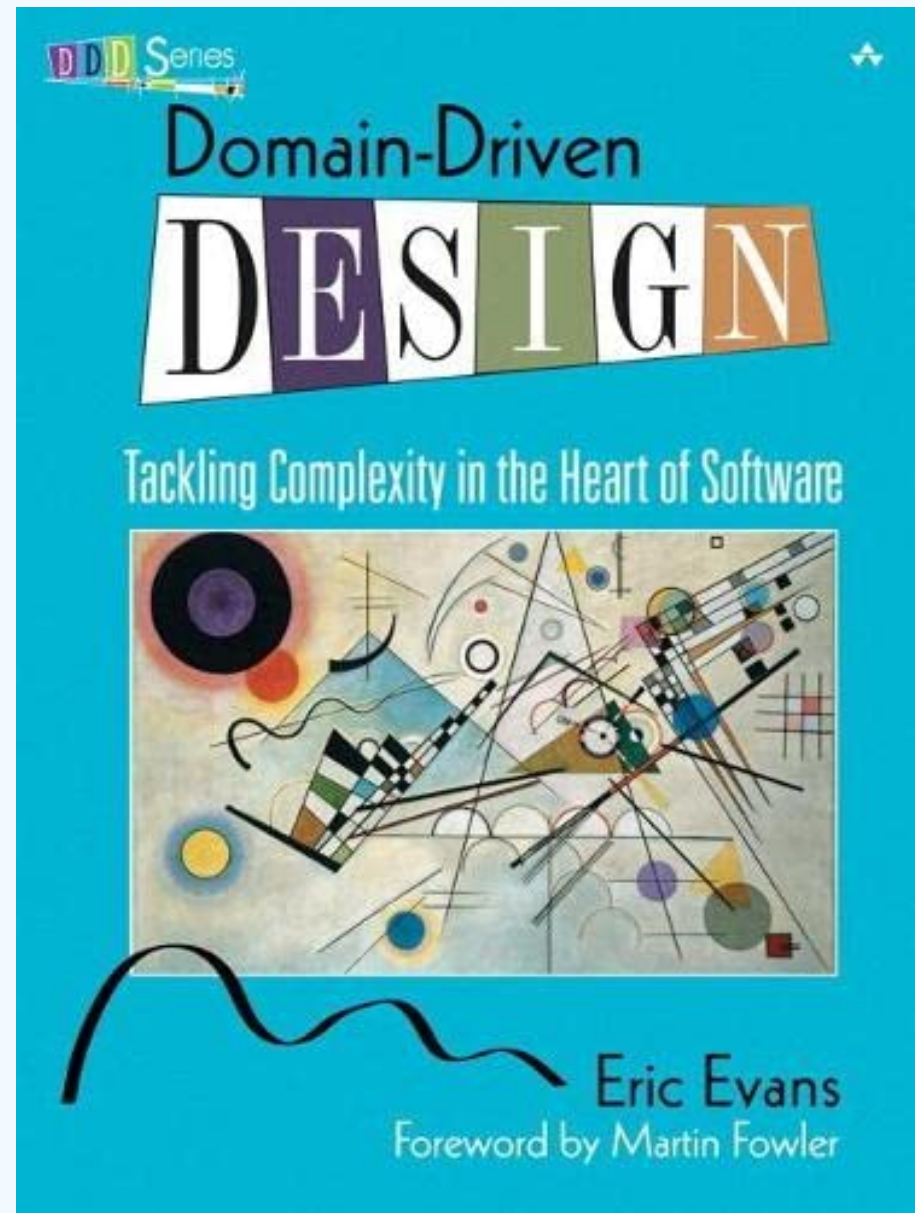
- El DDD se centra en la importancia de **entender bien el dominio** de nuestro problema para crear buen software.
 - En contraste al enfoque tradicional, centrado en los datos que tratamos.
- Para identificar los elementos del dominio se utiliza
 - **Lenguaje ubicuo**, brainstorming.
 - Identificación de los **contextos acotados**.
 - **Mapeo de contextos**.
- Todo esto se hace con el fin de **separar la lógica de aplicación** (casos de uso) **de la lógica de dominio**.
 - Los casos de uso varían mucho más frecuentemente.

AISLAR EL DOMINIO



MODELAR EL DOMINIO

- **Entidades.**
 - Elementos del dominio con entidad propia.
 - Comportamiento + datos.
- **Value Objects.**
 - Elementos que almacenan sólo datos.
 - Creados para representar de forma más clara los atributos de las entidades.
- **Aggregates.**
 - Conjunto de entidades y value objects con un sentido común.
- **Domain Services.**
 - Elementos del dominio con lógica que no tiene cabida en ninguna de las entidades.



DOMAIN DRIVEN DESIGN

RESUMEN

Daniel Blanco Calviño