

## Bridging the Gap Between Problem and Solution Space with Domain-Driven Design

**Authors :** Anil Kumar, Lavisha Gupta

**Abstract :** Domain-Driven Design (DDD) is a pivotal methodology in software development, emphasizing the understanding and modeling of core business domains to create effective solutions. This paper explores the significance of DDD in aligning software architecture with real-world domains, with a focus on its application within Siemens. We delve into the challenges faced by development teams in understanding domains and propose DDD as a solution to bridge the gap between problem and solution spaces. Key concepts of DDD, such as Ubiquitous Language, Bounded Contexts, Entities, Value Objects, and Aggregates, are discussed, along with their practical implications in software development. Through a real project example in the automatic generation of hardware and software plant engineering, we illustrate how DDD principles can transform complex domains into coherent and adaptable software solutions, echoing Siemens' commitment to excellence and innovation.

**Keywords :** domain-driven design, software architecture, ubiquitous language, bounded contexts, entities, value objects, aggregates

**Conference Title :** ICSEA 2024 : International Conference on Software Engineering Advances

**Conference Location :** Goa, India

**Conference Dates :** December 09-10, 2024