

Bridging DDD · BA/RE

Why semantic precision matters in the age of AI

Gottfried Szing · DDD & BA/RE Meetup · 2026



About me

Gottfried Szing

- Freelance Business Analyst & Requirements Engineer
- Board member IIABA Austria Chapter
- Meetups organizer
 - Requirements Engineers and Business Analysts Vienna
 - DDD Vienna
 - Microservices, Reactive and Distributed Systems Vienna

✉ gottfried@szing.eu

🌳 <https://linktr.ee/gottfried.szing>



Two Ideas

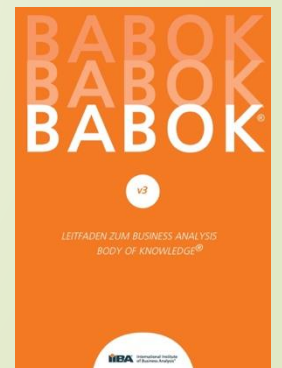
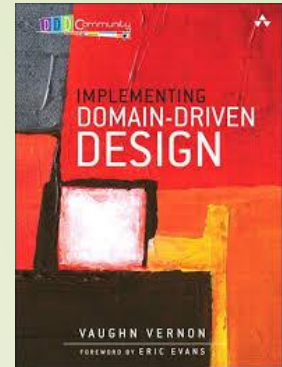
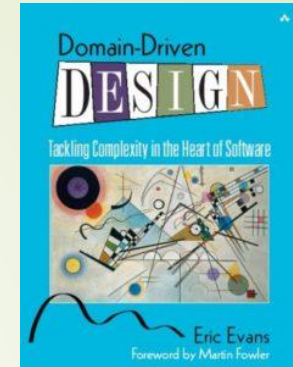
1. Stop treating DDD, BA, and RE as separate.
2. These disciplines are the ones that makes AI* actually useful.

* AI = LLMs

DDD and BA and RE

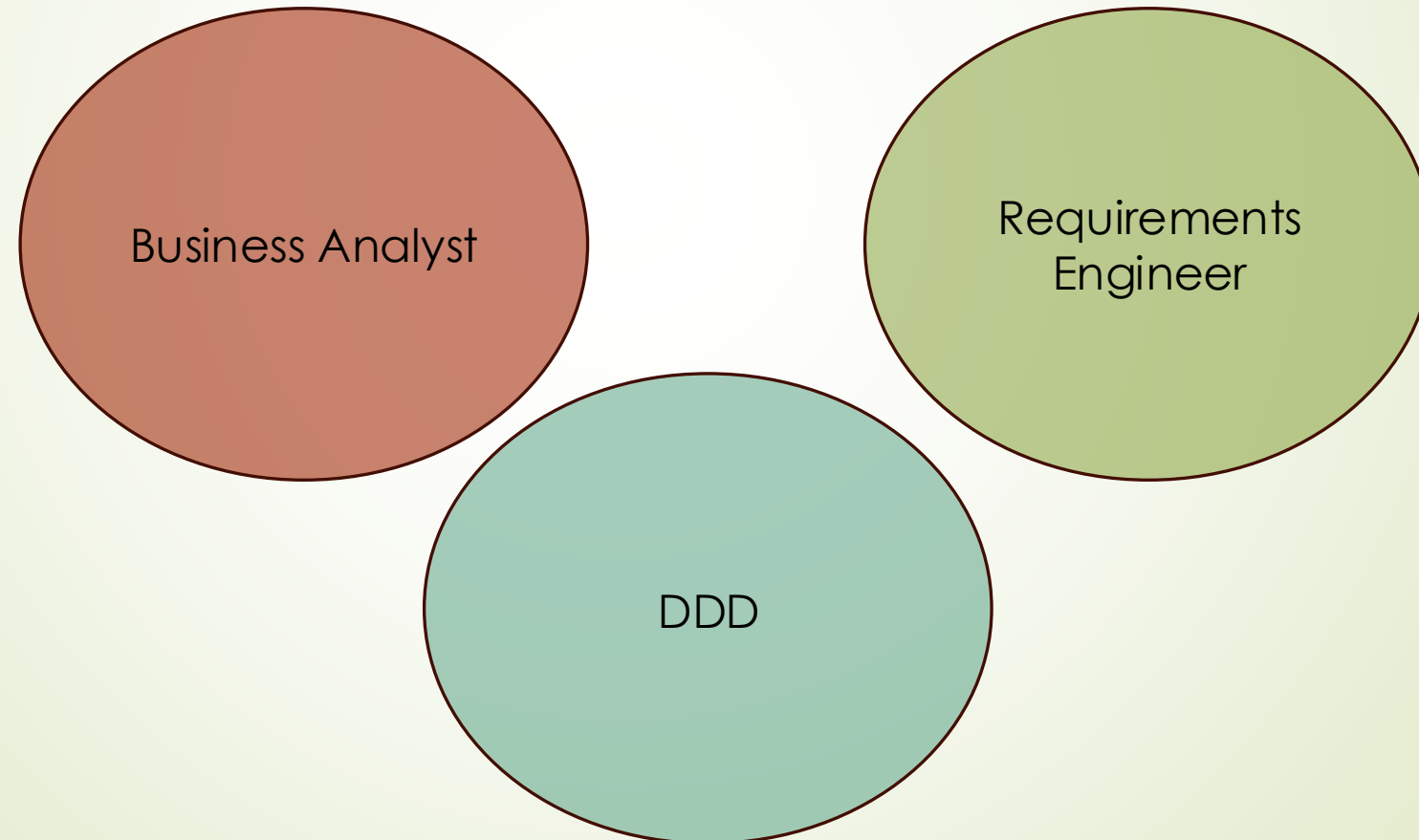
- **Domain-Driven Design (DDD)**
 - Community driven, Blue/Red Book
 - Designing software around the business domain.
- **Requirements Engineering (RE)**
 - IREB - International Requirements Engineering Board
 - Understanding stakeholders desires, delivering a system
- **Business Analysis (BA)**
 - IIBA - International Institute of Business Analysis
 - Aligning business needs to drive solutions

Different disciplines & approaches. Same underlying concerns.

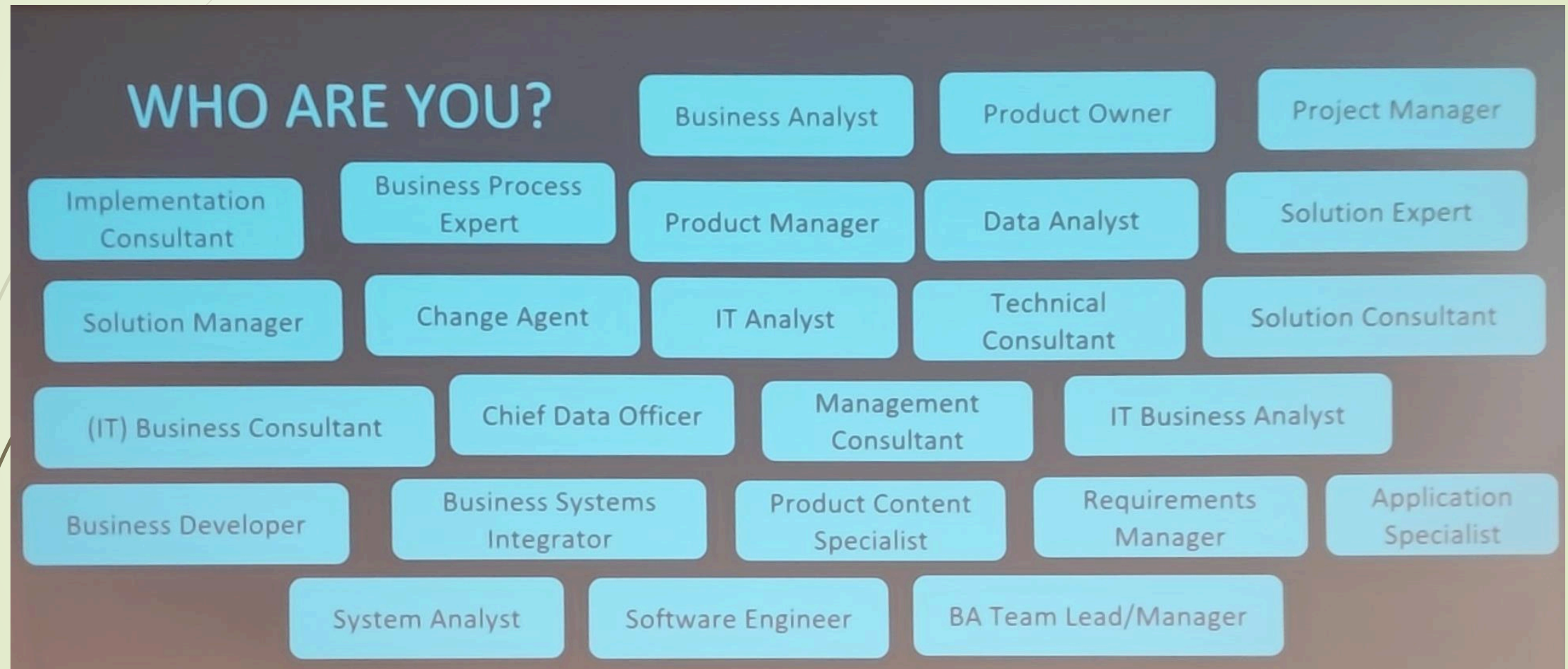




The Intersection



Not About Titles & Roles



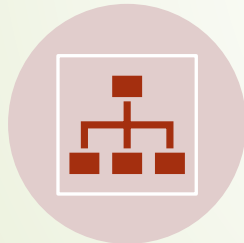
DDD in 90 Seconds



Ubiquitous Language - One shared, precise vocabulary agreed upon by domain experts and technical teams.



Bounded Context - A term means what it means inside a defined boundary. E.g. a "policy" in insurance ≠ a "policy" in HR.



Domain Model - A structured representation of business concepts, rules, and relationships.



Context Map - Showing bounded contexts and how they relate to each other.

If any of this sounds familiar, BABOK and CPRE have been telling you to do it for years - under different names.

DDD ↔ BA ↔ RE

DDD	BA	RE
Ubiquitous Language	Glossary, Concept Modeling	Glossary, shared project vocabulary
Bounded Context	Solution Scope, Stakeholder domains	System Context & Context Boundary (<i>Kontextabgrenzung</i>)
Domain Model	Conceptual Data Model, Business Rules Analysis	Conceptual models (data / function / behavior)
Context Map	Stakeholder & Interface Analysis	Context diagram, external interfaces

Different vocabulary. Same intent.



What DDD adds for BA/RE

- BA/RE often define terms. → DDD defines meaning within boundaries.
- BA/RE often model structure. → DDD connects language to behavior.
- BA/RE define scope. → DDD operationalizes boundaries.

DDD is what both frameworks already recommend.



Concrete Advantages for BA/RE

- **Sharper elicitation** - Ubiquitous Language surfaces disagreements that synonyms hide
- **Cleaner scope** - Bounded contexts make "in/out" defensible in BA/RE terms
- **Smoother BA-to-dev handover** - one model, not two
- **Living documentation** - glossary becomes a working artifact, not a sign-off deliverable



Then Als Entered the Picture



The New Consumer of Requirements

- **From Human-to-Human...**
- **To Human-to-Agent**

- Human developers can often "read between the lines" of a vague requirement.
- AI cannot - it requires the absolute precision



AI Amplifies Ambiguity

- LLMs and AI agents now consume requirements and glossaries as input
- Output quality is bounded by input precision
- AI outputs that sound plausible but miss business reality
- Ambiguity a human could resolve through context, an AI will confidently hallucinate around

AI scales semantic quality problems. Scaled human hallucination.



Better Input, Better AI Output

1. **Ubiquitous Language** aligns AI interpretation with business meaning.
2. **Bounded contexts** map naturally to agent responsibilities and tool boundaries.
3. A **bounded context** is a natural scope limit and supports context window discipline

The AI handles the syntax. Humans own the semantics.



Example

\$ Generate a scenario in the style GIVEN-WHEN-THEN for...

- ▶ Without domain context:

"GIVEN an entity, WHEN the person pays, THEN the end date is increased"

- ▶ With a domain context:

"GIVEN a policy, WHEN the policy holder pays the renewal, THEN the policy is valid for another year"

Same AI. Different results.



Takeaways

1. DDD, BA, and RE **overlap heavily**
2. **Semantic precision** increases the of value AI
3. Ubiquitous Language and boundaries are now **operational assets**
4. In an AI-augmented organization - mastering Ubiquitous Language and Bounded Contexts is what makes **us indispensable**.



Thank you

Gottfried Szing

Freelance Business Analyst & Requirements Engineer

✉ gottfried@szing.eu

🌳 <https://linktr.ee/gottfried.szing>

