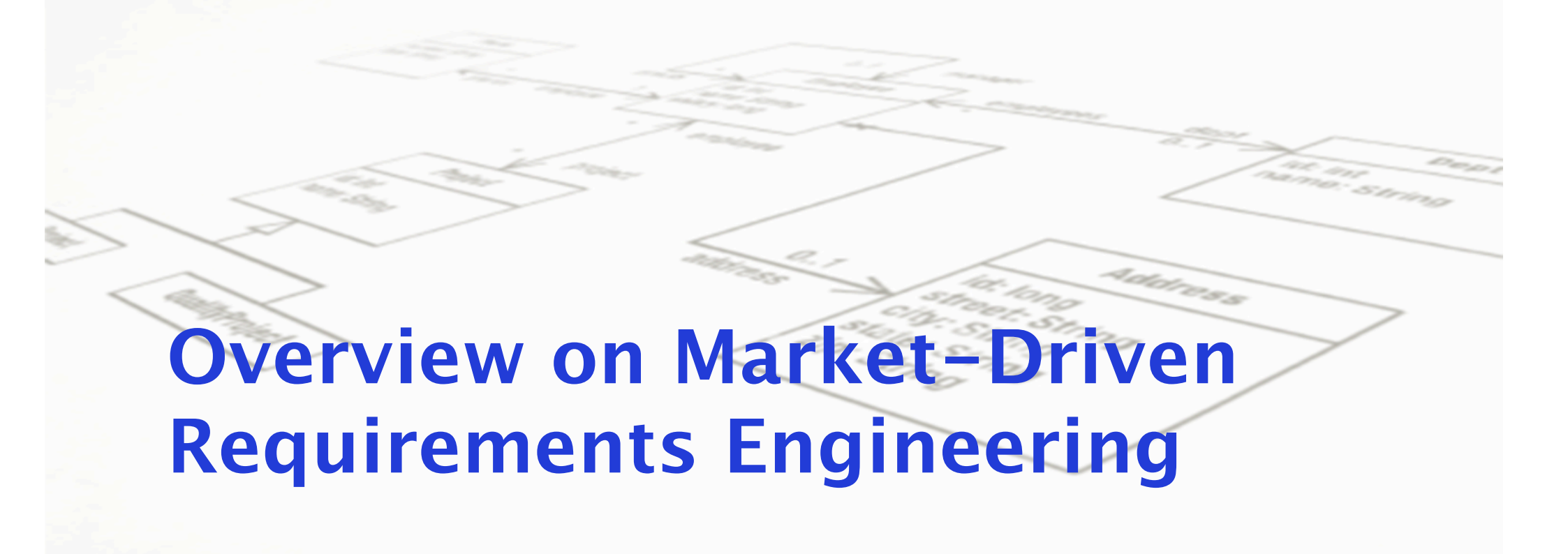


A faded UML class diagram serves as the background for the top half of the slide. It shows several classes: 'User' with attributes 'id: int', 'name: string', and 'password'; 'Address' with attributes 'id: long', 'street: string', 'city: string', 'state: string', and 'zip: string'; and 'Dept' with attributes 'id: int' and 'name: string'. Relationships include 'has' (User to Address), 'knows' (User to User), 'works at' (User to Dept), and 'lives at' (User to Address). Multiplicities like '0..1' and '1' are visible on some association lines.

Overview on Market-Driven Requirements Engineering

Requirements Engineering II

December 15th 2008

Michael Sommerhalder, Krzysztof Dabkowski, Stefan Schurgast

Agenda

- Objective
- Customer– vs. market–orientation
- Challenges of the MDRE approach
- Life cycle models
- Opportunities and challenges of lifecycle models

Objective

“The primary objective of market-driven development is to deliver the right product at the right time [...] success is determined by sales, market share, product reviews, etc.”

Customer– vs. market–orientation

- Customer-oriented
 - ◆ Stakeholders: known
 - ◆ Users: known
 - ◆ Lifecycle: one release (or several spread in time)
 - ◆ Focus: compliance to specifications and agreements
- Market-oriented
 - ◆ Stakeholders: unknown
 - ◆ Users: unknown
 - ◆ Lifecycle: regular releases, the whole product
 - ◆ Focus: time-to-market, return-of-investment

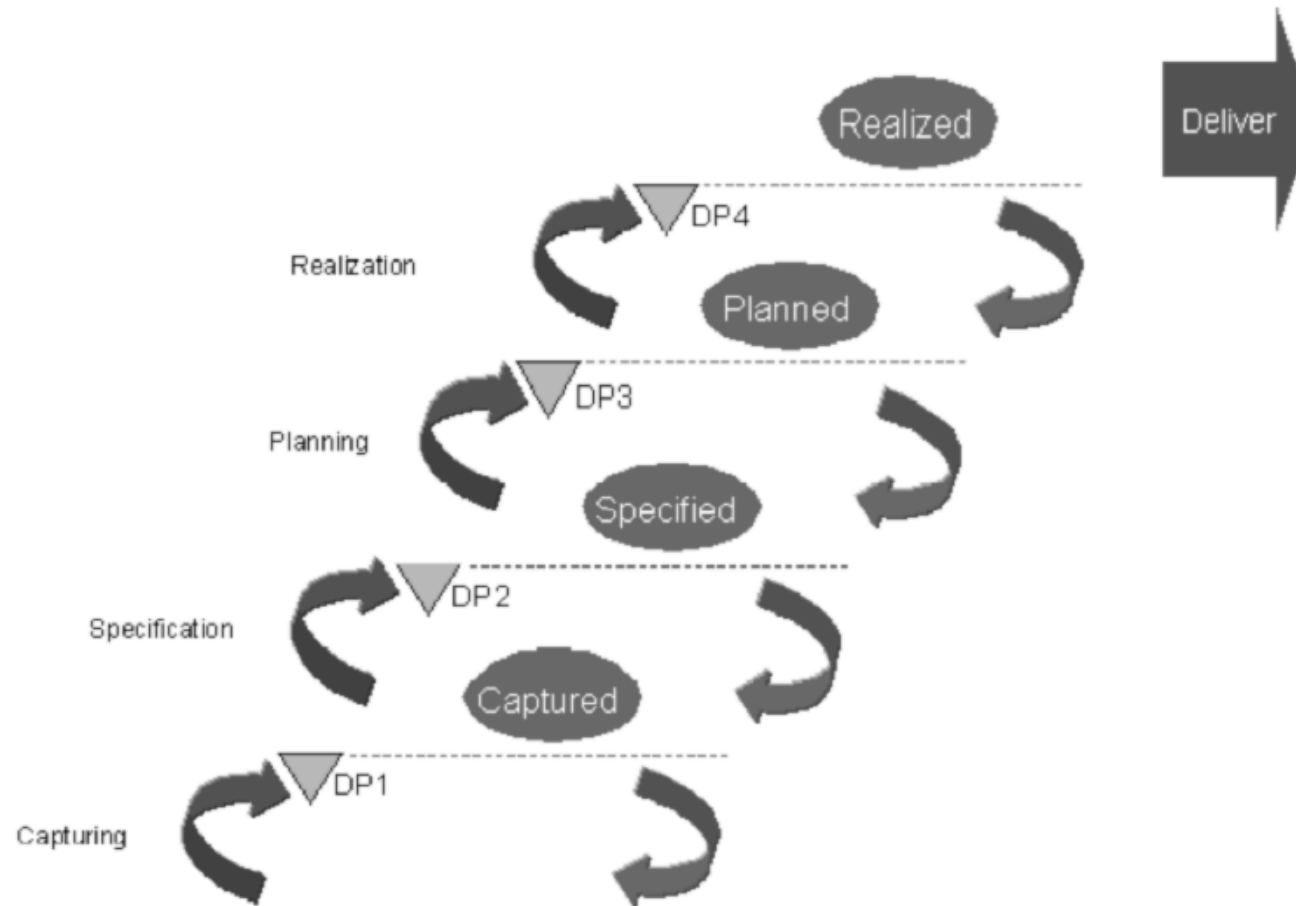
Challenges for MDRE approach

- Balancing market pull and technology push.
- Chasm between marketing and development.
- Organizational instability and market turbulence.
- Simple tools for basic needs.
- Requirements dependencies.
- Cost–value–estimation and release planning.
- Overloaded Requirements Management.

Life cycle models – 1

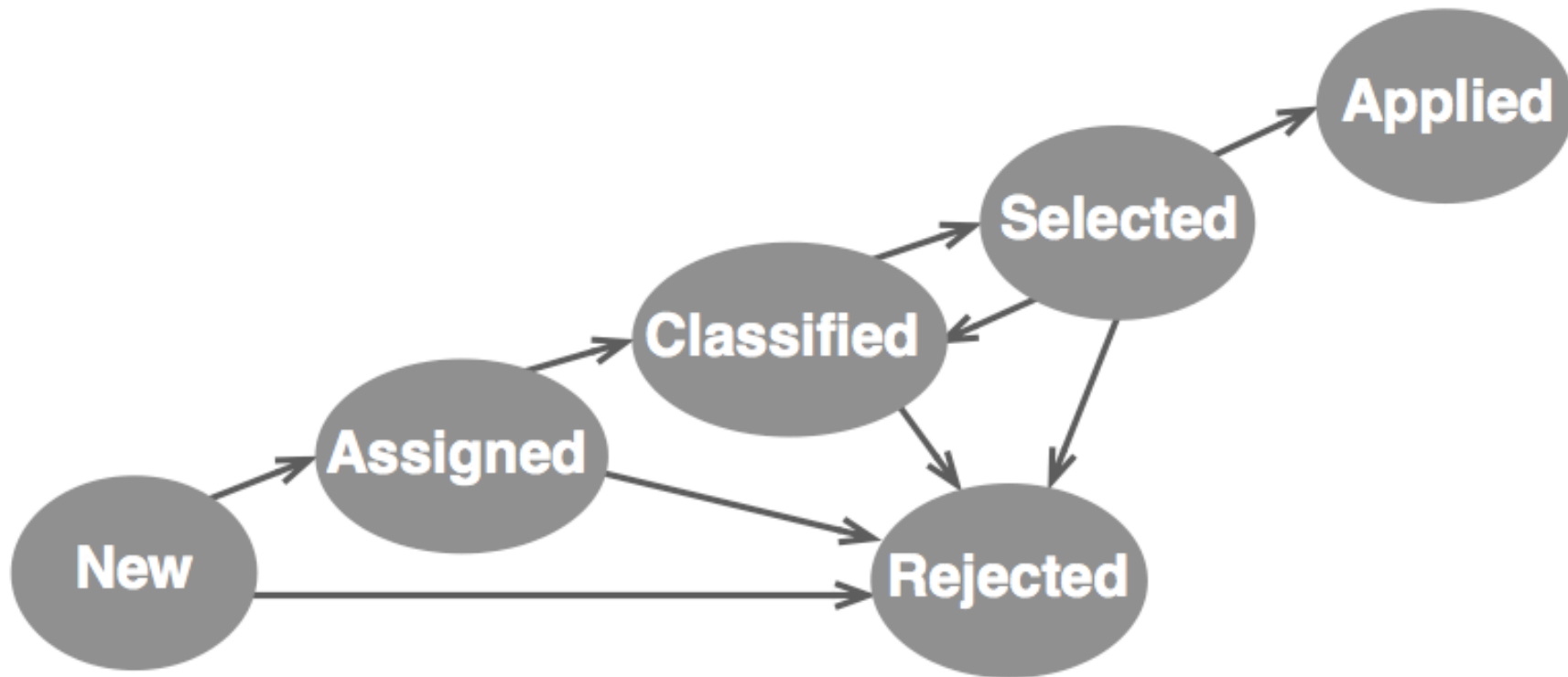
- RDEM Process

- ♦ software development as a learning process
- ♦ Assigning values to attributes
- ♦ Progression in the process defined as “what is already known?”



Life cycle models – 2

- REPEAT Process
 - ♦ Strong focus on requirements selection and release planning



Comparison RDEM / REPEAT

	RDEM	REPEAT
Roles	Application Domain Team (mandatory: Domain Manager, Architect, Development Manager, Lead Analyst, Lead Designer, Lead Tester; but 20+ exist)	Customer, Issuer, Expert, Requirements Management Group, Requirments Team, Construction Team, Test Team
Activities	Capturing and specification, planning, realization	Elicitation, documentation, validation

Opportunities of lifecycle approach

- Methodology independence
 - ♦ Approach for requirements engineering doesn't influence the process
- Balanced quality
 - ♦ Attribute-centric view takes focus away from specific methodology to quality attributes
- Progress indication
 - ♦ Ratio of what is known and what still has to be investigated is clear at all times
- Flexibility
 - ♦ Infrastructure for requirements management is very much adaptable, attributes introduced just at the point when needed, e.g.

Challenges of lifecycle approach

- Requirements dependencies
 - ◆ Management of requirements dependencies is central as it makes sure the right fragments of information are connected to each other
- Synthesizing priorities
 - ◆ When market place global: very important to synthesize information about market segments around the world
- Definition of attributes and states
 - ◆ Difficult to tell which attributes and states fit which situation. Both models are currently being revised

Literature

- [Carlshamre 2000]: Carlshamre, P., B. Regnell (2000). “Requirements Lifecycle Management and Release Planning in Market-Driven Requirements Engineering Processes”, International Workshop on Requirements Engineering Process: Innovative Techniques, Models, and Tools to Support the RE Process (REP2000), Greenwich, UK.
- [Regnell 2005]: Regnell, B., S. Brinkkemper (2005). “Market-Driven Requirements Engineering for Software Product”, in A. Aurum, C. Wohlin (eds.), *Engineering and Managing Requirements*, Springer.
- [van de Weerd 2006]: van de Weerd, I., S. Brinkkemper, R. Nieuwenhuis, J. Versendaal, L. Bijlsma (2006). “Towards a Reference Framework for Software Product Management”, 14th IEEE International Requirements Engineering Conference (RE106), Minneapolis, USA.