KobrA Overview

1 Challenges

2 Existing Methods

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics

An Overview of th KobrA Meth



Netobjectdays, Erfurt
October 12th, 2000

Dirk Muthig

IESE

Fraunhofer Institut
Experimentelles
Software Engineering

Component-Based S/W Development (CBSD)

KobrA Overview
1 Challenges
2 Existing Methods
3 Basic Principles
4 Framework Engineering
5 Product Lines
6 Characteristics

The assembly of self-contained software capsules to create new software capsules

Expected benefits -

- Reuse (in the small)
- enhanced quality
- rapid application development
- COTS Marketplace

BUT

- today, being a component means being a JavaBean, CORBA application or COM object
  - components primarily an implementation concept
- ⇒ how can the component paradigm play a bigger role in “front end” development activities?

KOBRA

Copyright © KOBRA Group 2000

Product-Line Engineering

KobrA Overview
1 Challenges
2 Existing Methods
3 Basic Principles
4 Framework Engineering
5 Product Lines
6 Characteristics

The generation and application of generic software assets that are reusable across a family of target products

Expect benefits -

- rapid application development
- Reuse (in the large)
- improved maintenance

BUT

- existing approaches are very high-level and unconnected with implementation issues
- ⇒ how can product-line approaches be coupled more tightly with implementation approaches (especially CBSD)?

KOBRA

Copyright © KOBRA Group 2000

Visual Modeling

KobrA Overview
1 Challenges
2 Existing Methods
3 Basic Principles
4 Framework Engineering
5 Product Lines
6 Characteristics

The creation and evolution of graphical (UML) models to visualize specific aspects of software elements and their environment

Expected benefits -

- early validation
- early architecture description/verification
- explicit documentation
- standard representation

BUT

- Support for component-development and software product lines is minimal
  - based on “binary module” view of components
- ⇒ How can the UML best be used in conjunction with CBSE and product lines?

KOBRA

Copyright © KOBRA Group 2000

Problems with Existing Methods

KobrA Overview

1 Challenges

2 Existing Mtds

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics

Unified Process

- Complicated
- Limited view of components
- Fundamental conflict between being “use-case drive” and “architecture centric”

Catalysis

- Complicated
- Unfocused

OPEN

- Very high level
- unprescriptive

Fusion

- “Flat” and unscaleable

KOBRA

Copyright © KOBRA Group 2000

A System = A Component

KobrA Overview

1 Challenges

2 Existing Methods

3 Basic Principles

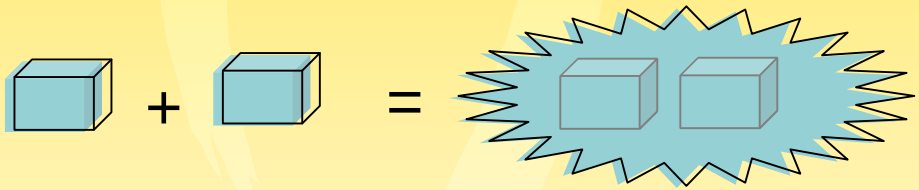
4 Framework Engineering

5 Product Lines

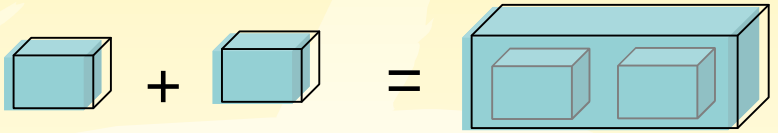
6 Characteristics

Many approaches view component development and component assembly as different activities

- development with components ≠ development of components



- In KobrA component development and assembly are the same thing



⇒ a system = a component

KOBRA

Copyright © KOBRA Group 2000

The Principle of Locality

KobrA Overview

1 Challenges

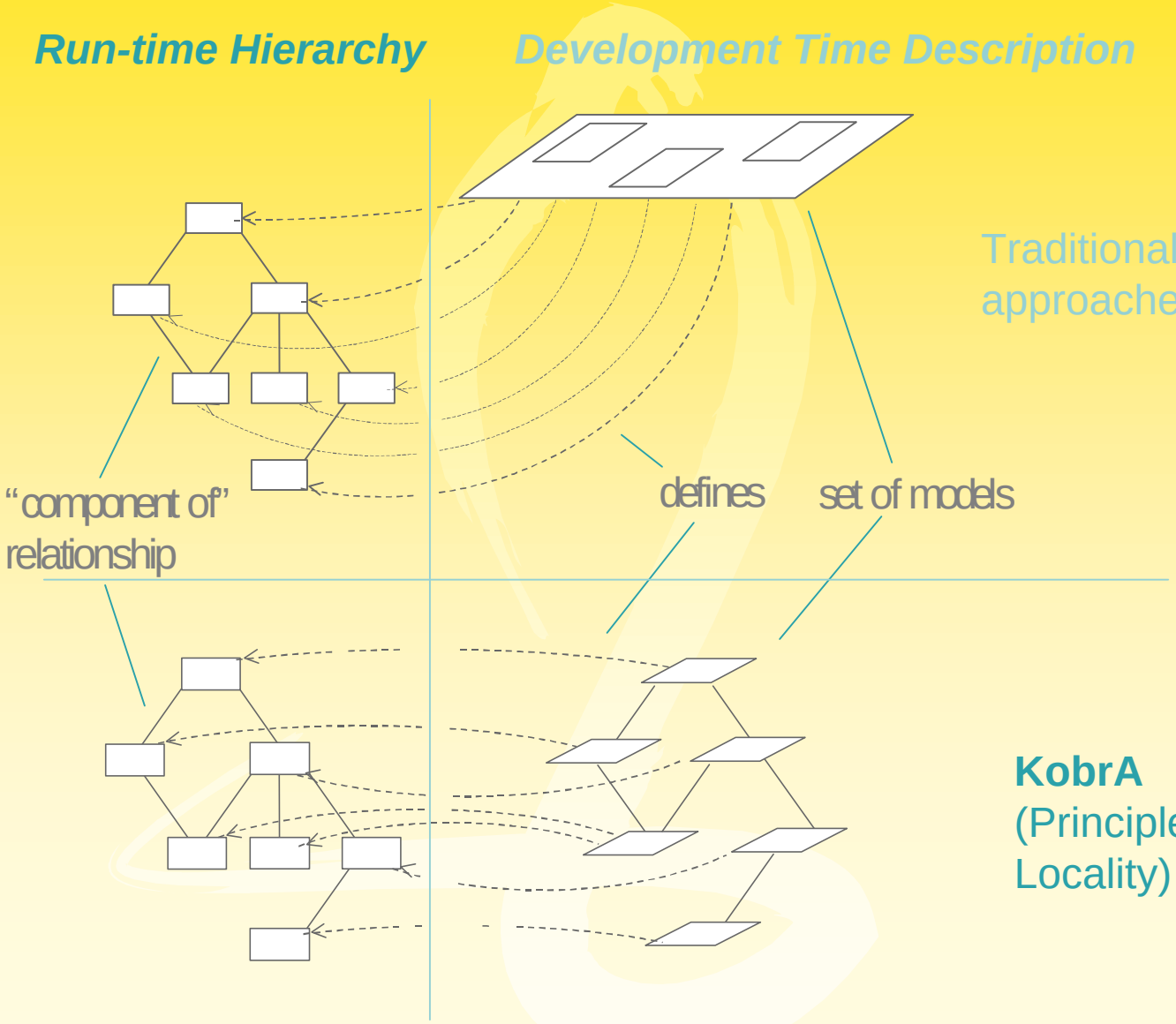
2 Existing Methods

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics



Traditional approaches

KobrA (Principle of Locality)

KOBRA

Copyright © KOBRA Group 2000

Component Model Suite

KobrA Overview

1 Challenges

2 Existing Methods

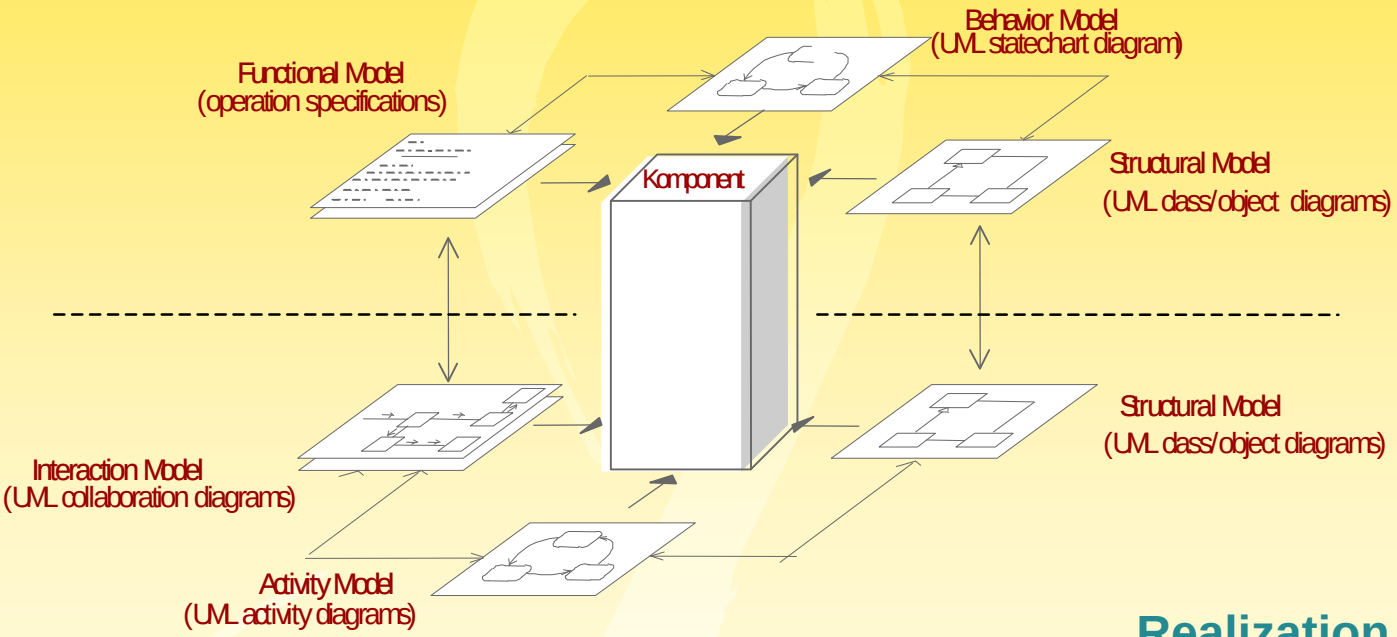
3 Basic Principles

4 **Framework Engineering**

5 Product Lines

6 Characteristics

Specification



Realization

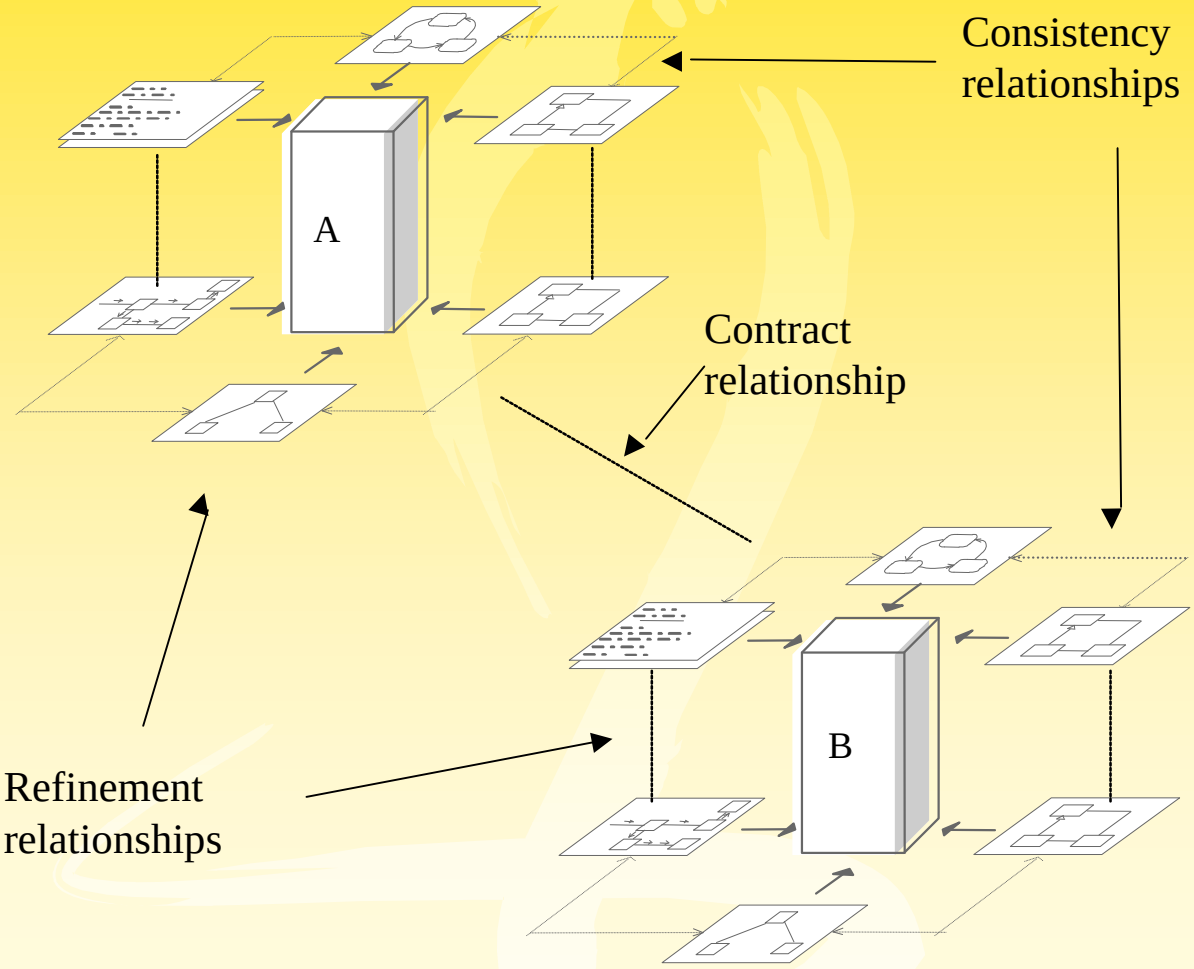
KOBRA

Copyright © KOBRA Group 2000

Framework Correctness Rules

KobrA Overview

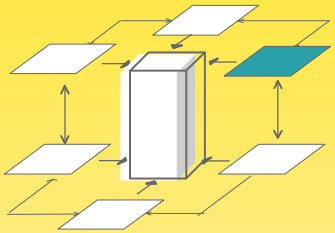
- 1 Challenges
- 2 Existing Methods
- 3 Basic Principles
- **4 Framework Engineering**
- 5 Product Lines
- 6 Characteristics



KOBRA

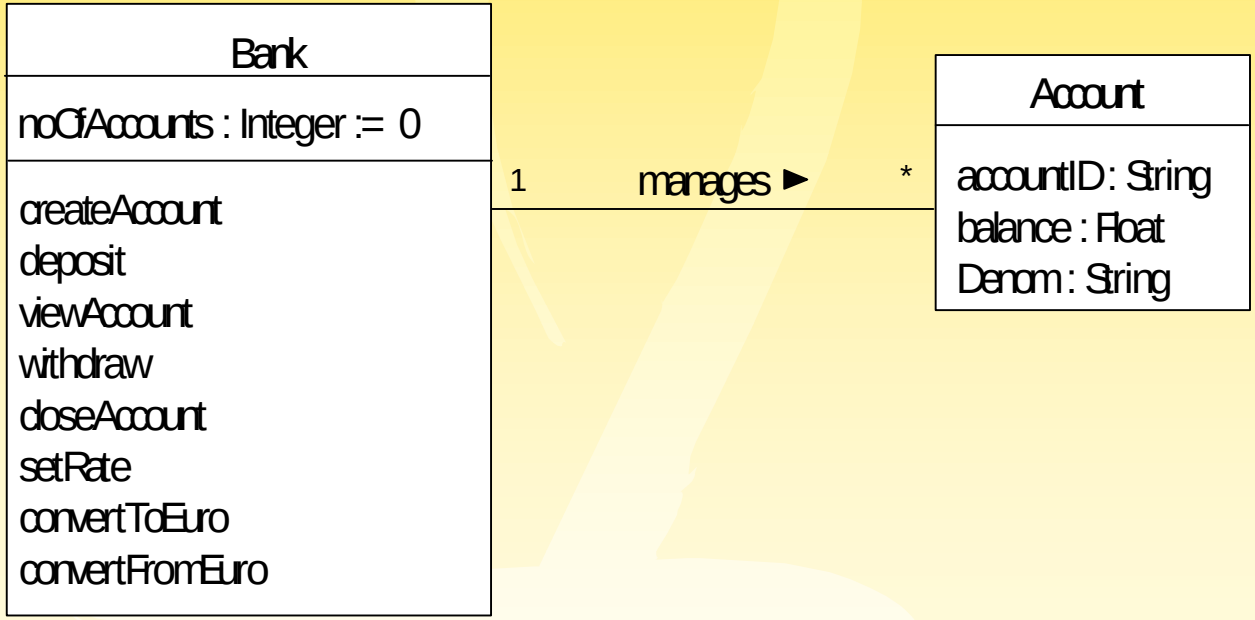
Copyright © KOBRA Group 2000

Bank Specification Structural Model



Bank

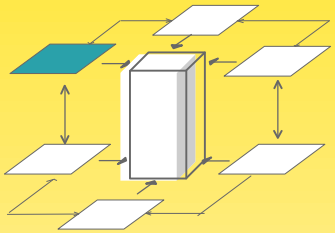
Bank Specification Class Diagram



KOBRA

Copyright © KOBRA Group 2000

Bank Specification Functional Model (Contd)



Bank

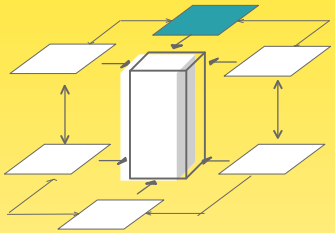
withdraw operation specification

Name	withdraw
Informal Description	An amount of money in a particular currency is withdrawn from an account.
Constraints	--
Receives	ID : String currency:String amount:Float
Returns	A boolean indicating whether the withdrawal was possible
Changes	account with ID = ID
Assumes	--
Result	If ID or currency are not valid, the operation has been aborted and an exception has been raised. Else if account.balance >= amount, account.balance := account.balance – amount and „true“ has been returned Otherwise „false“ has been returned.

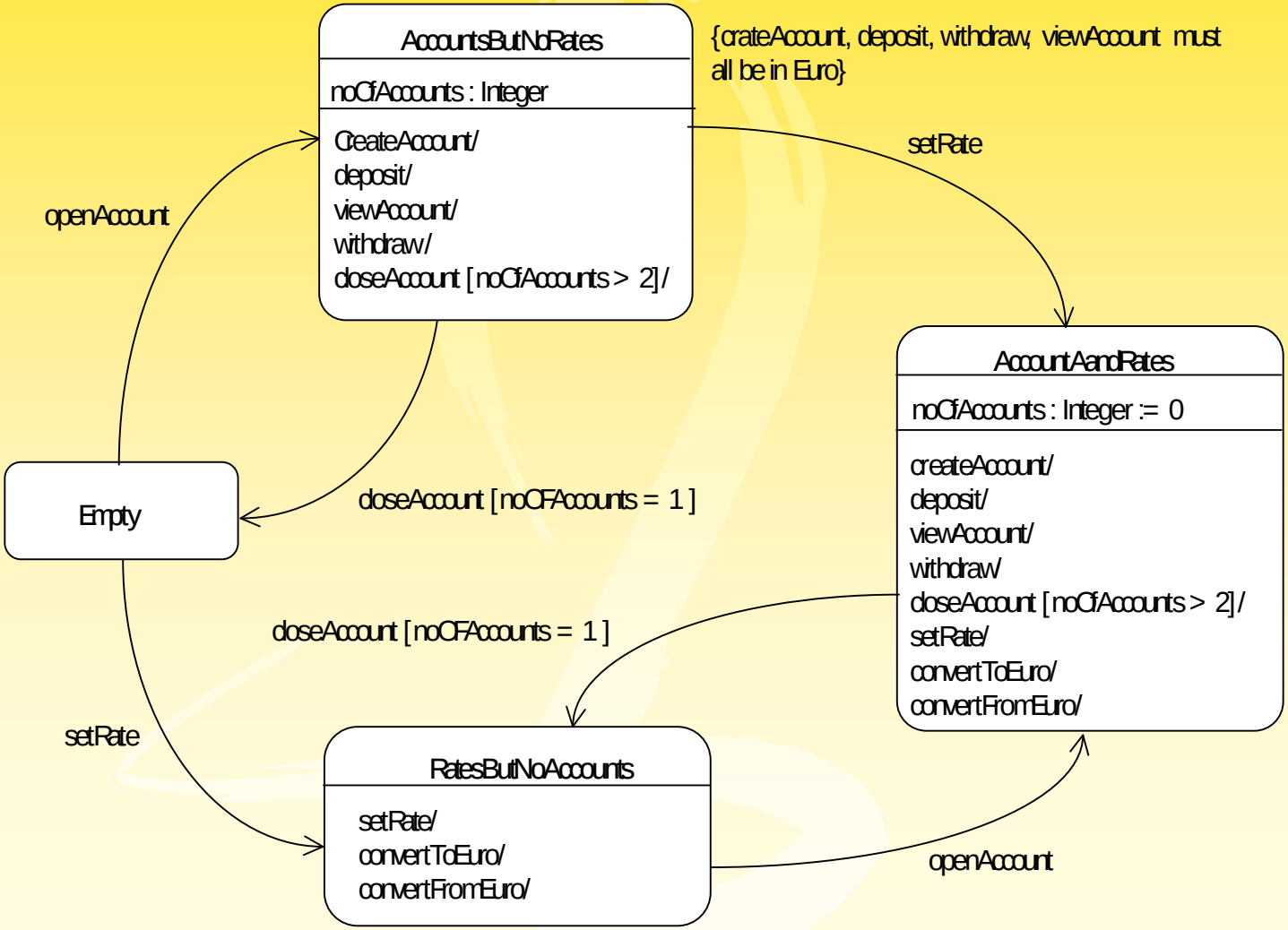
KOBRA

Copyright © KOBRA Group 2000

Bank Specification Behavioral Model



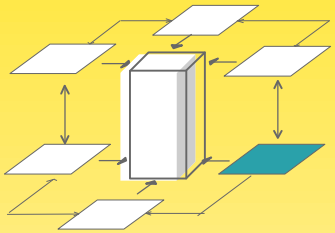
Bank



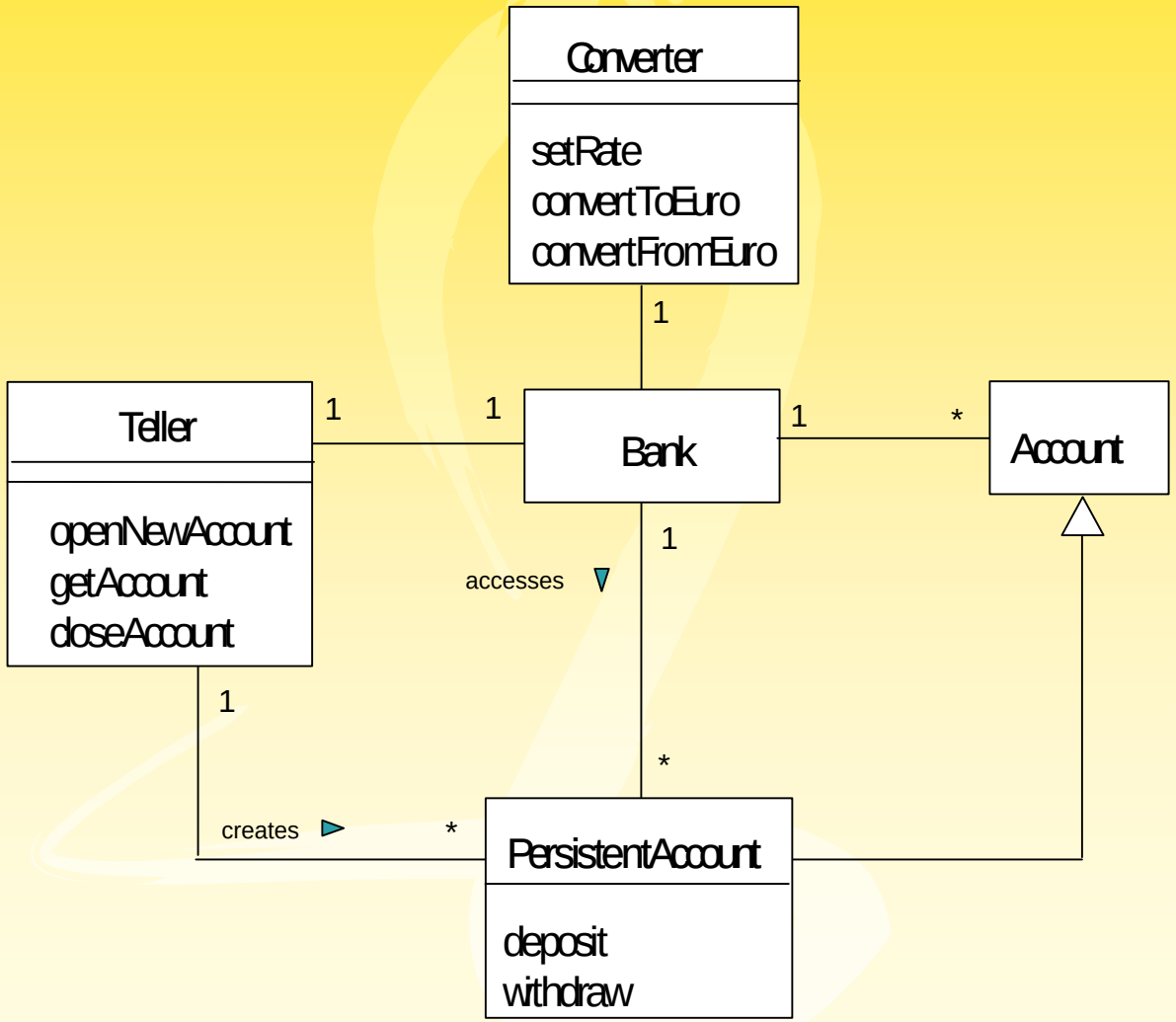
KOBRA

Copyright © KOBRA Group 2000

Bank Realization Structural Model



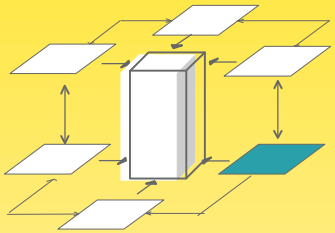
Bank



KOBRA

Copyright © KOBRA Group 2000

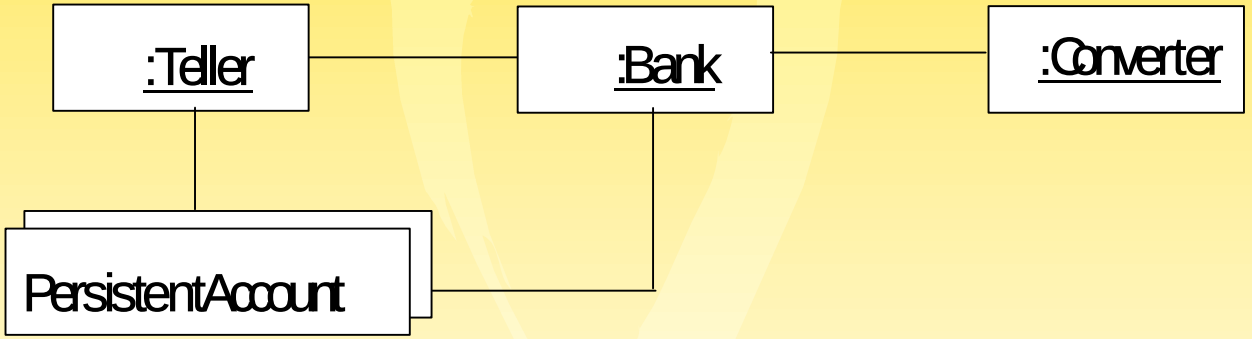
Bank Realization Structural Model (Contd.)



Bank

Bank Specification Object Diagram

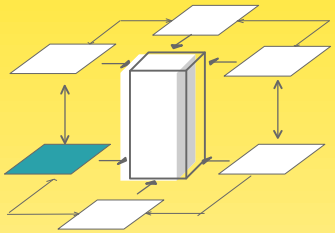
– effectively the Bank component's architecture



KOBRA

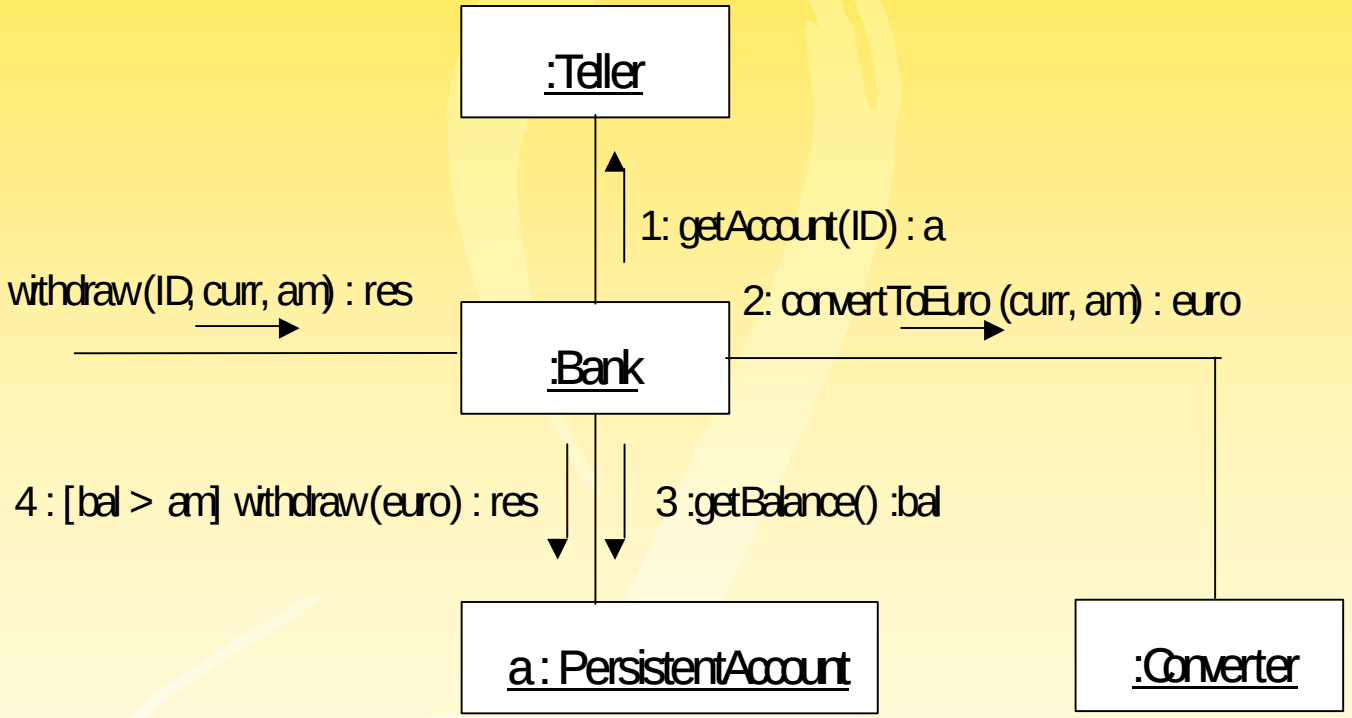
Copyright © KOBRA Group 2000

Bank Realization Interaction Model (Contd)



Bank

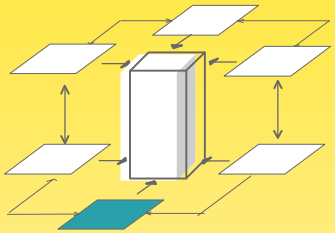
withdraw Collaboration Diagram



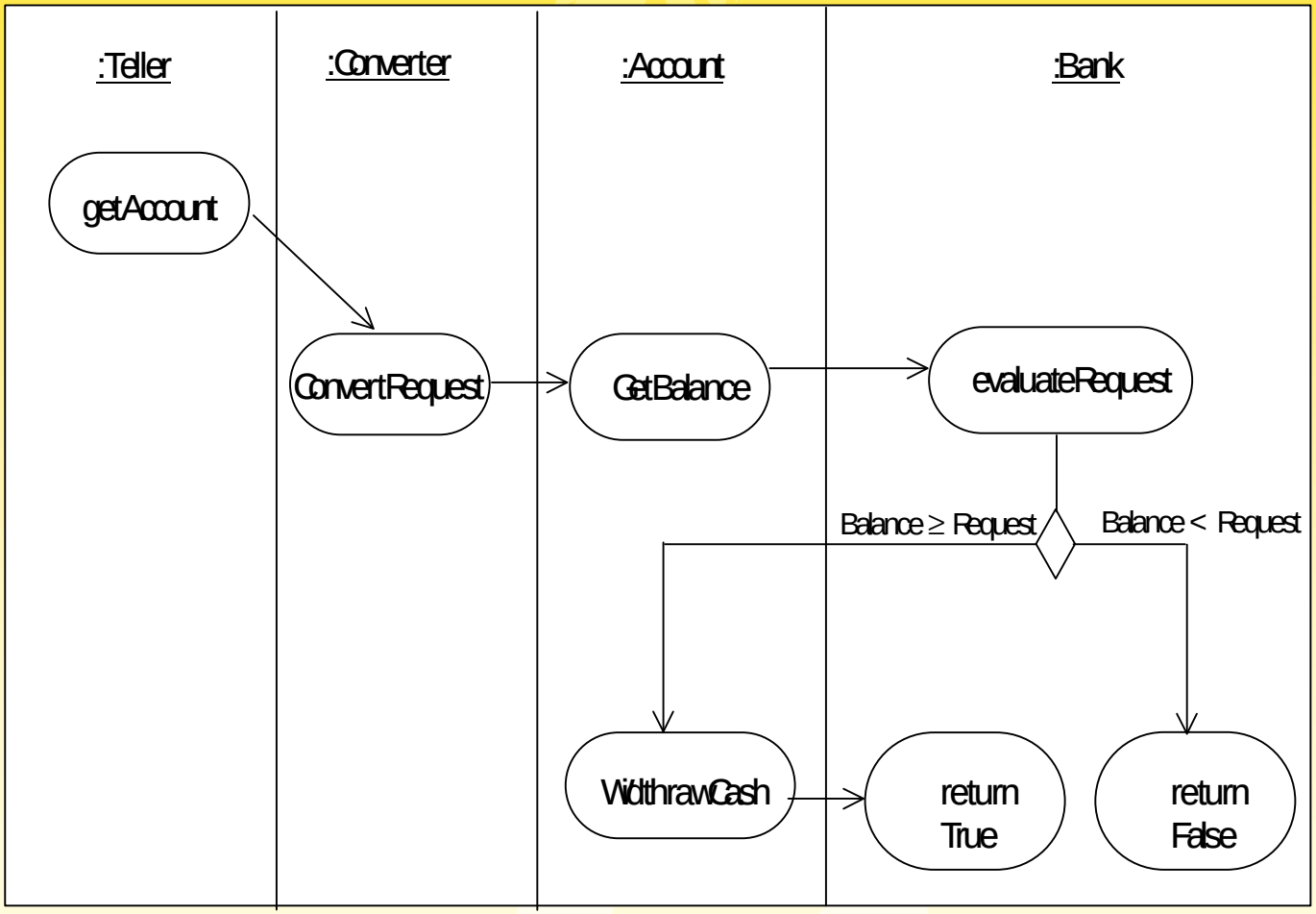
KOBRA

Copyright © KOBRA Group 2000

Bank Realization Activity Model



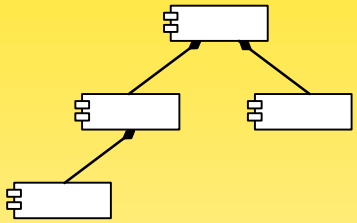
Bank

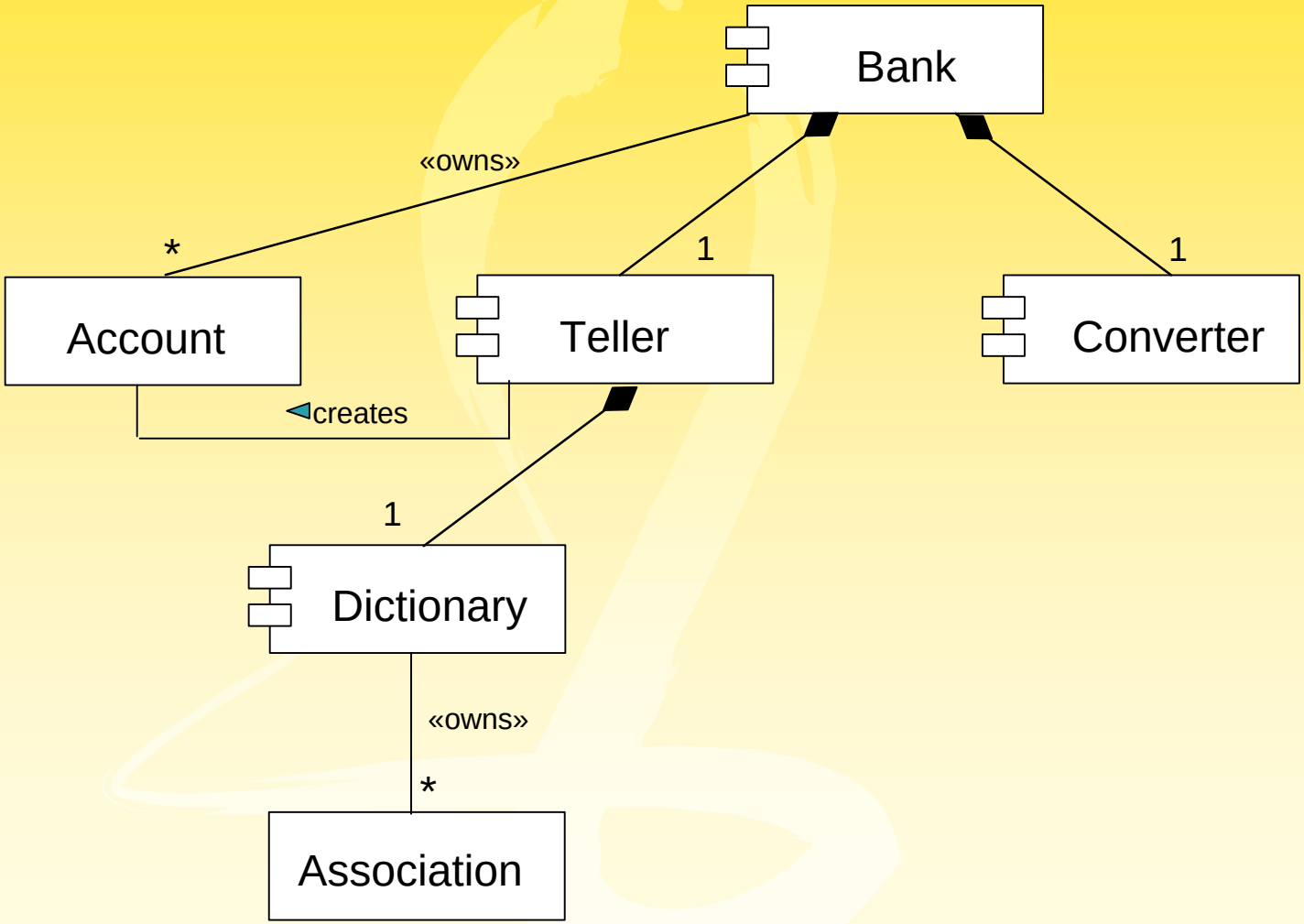


KOBRA

Copyright © KOBRA Group 2000

SIB Component Hierarchy

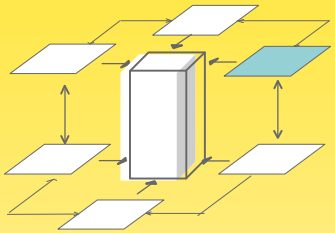




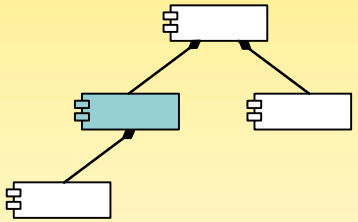
KOBRA

Copyright © KOBRA Group 2000

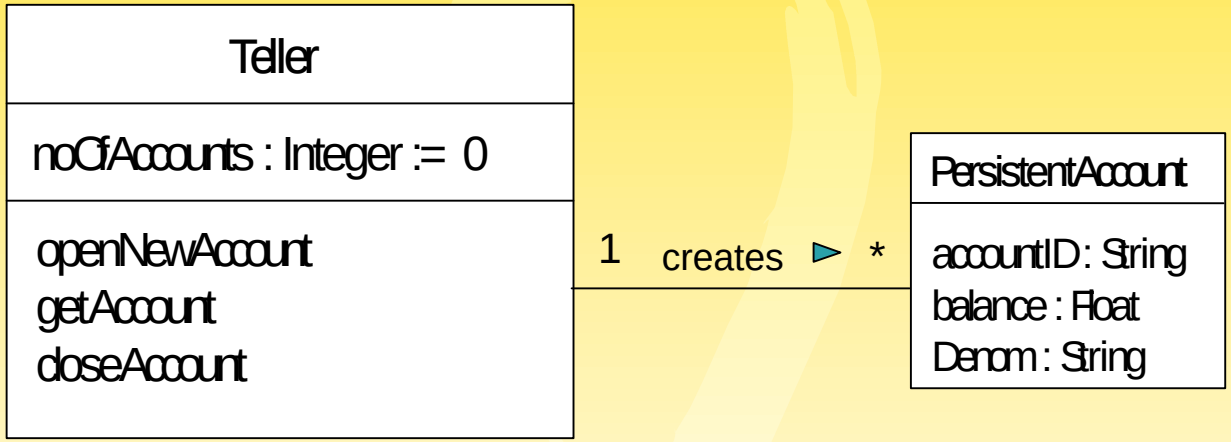
Teller Specification Structural Model



Teller



Teller Specification Class Diagram



KOBRA

Copyright © KOBRA Group 2000

Framework Engineering

KobrA Overview

1 Challenges

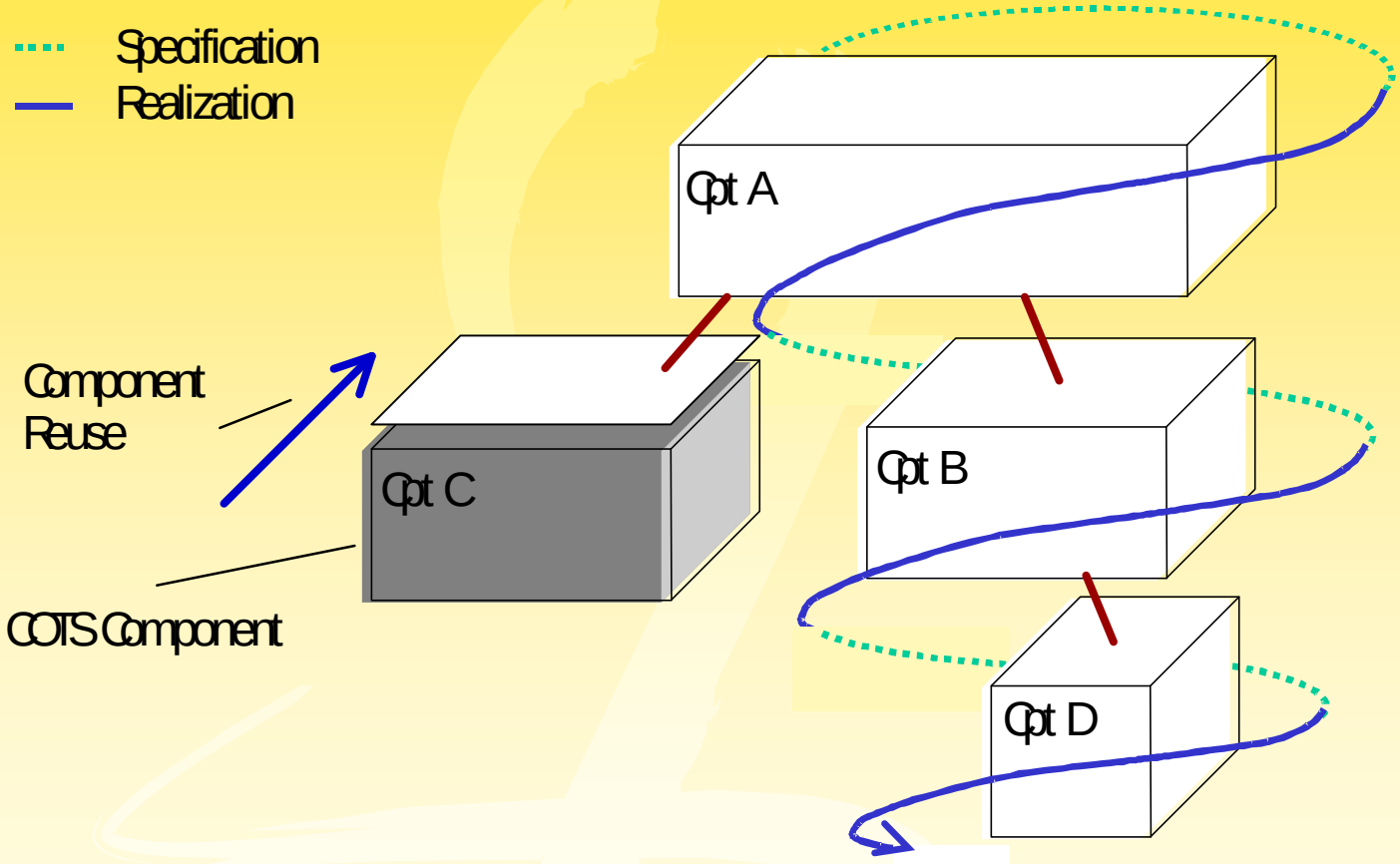
2 Existing Methods

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics



KOBRA

Copyright © KOBRA Group 2000

Implementation and Building

KobrA Overview

1 Challenges

2 Existing Methods

3 Basic Principles

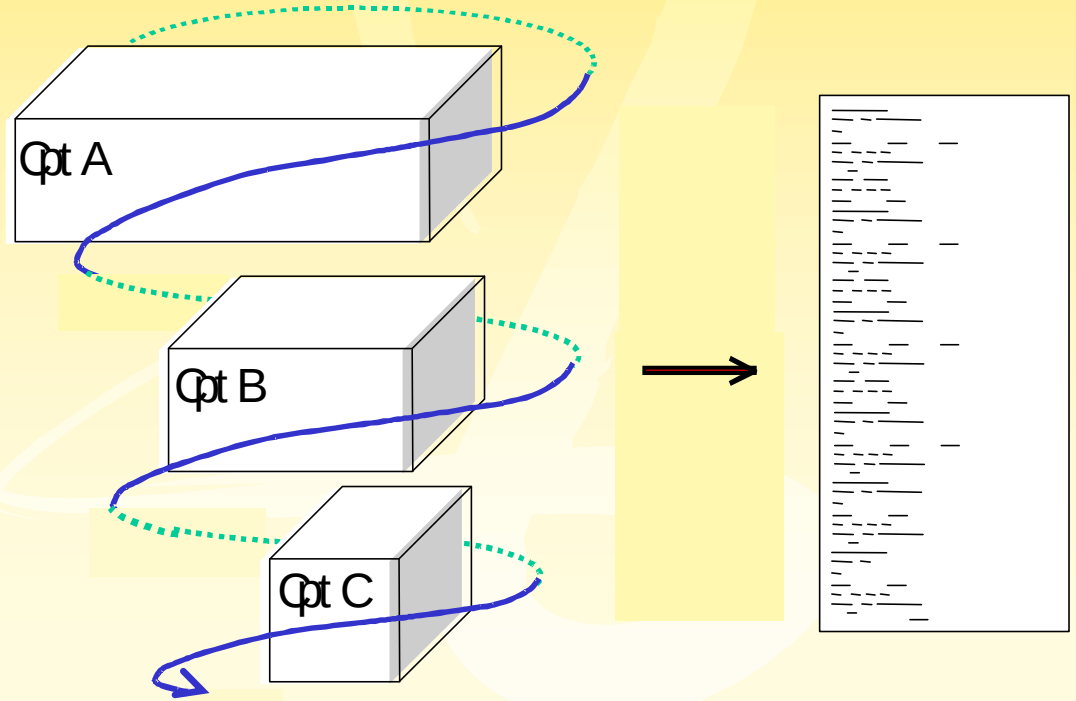
4 Framework Engineering

5 Product Lines

6 Characteristics

Mapping component realizations into an executable form represents a separate dimension

- ❒ executable images can be created after each realization
- ⇒ incremental development



KOBRA

Copyright © KOBRA Group 2000

Product Line Engineering

KobrA Overview

1 Challenges

2 Existing Methods

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics

Most software development organizations develop a line of similar but slightly different products

KobrA describes a family of similar systems (i.e. components) by modeling

- variabilities
- decisions

A component (class) with (non-empty) decision models (i.e. optional features) is known as a **generic class**

A tree of generic components represents a generic framework

- decision models are related hierarchically
- decision model resolution implies the resolution of all lower decision models

=> Incremental introduction of product lines

KOBRA

Copyright © KOBRA Group 2000

Generic Component Models

KobrA Overview

1 Challenges

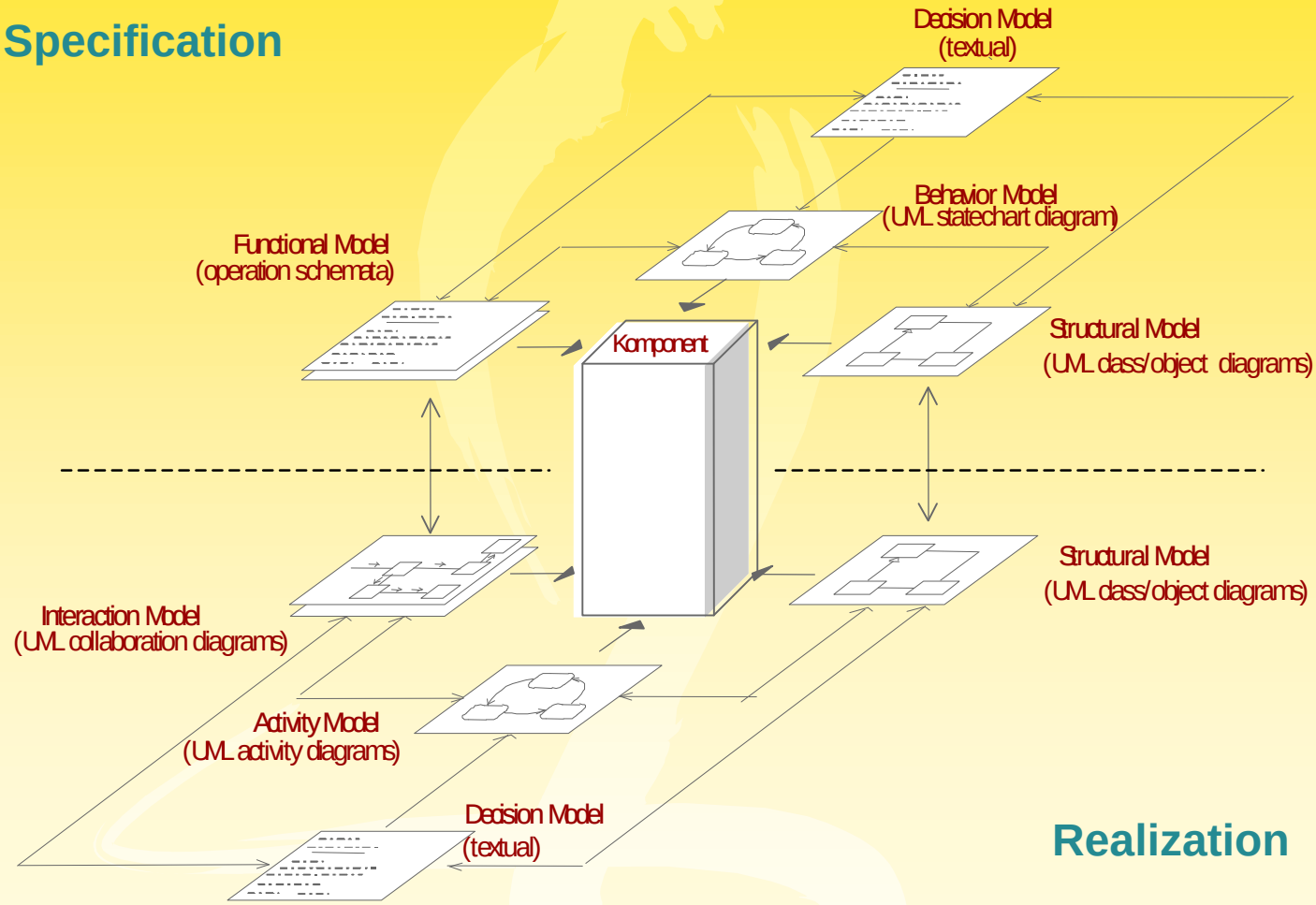
2 Existing Methods

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics



KOBRA

Copyright © KOBRA Group 2000

Separation of Concerns

KobrA Overview

1 Challenges

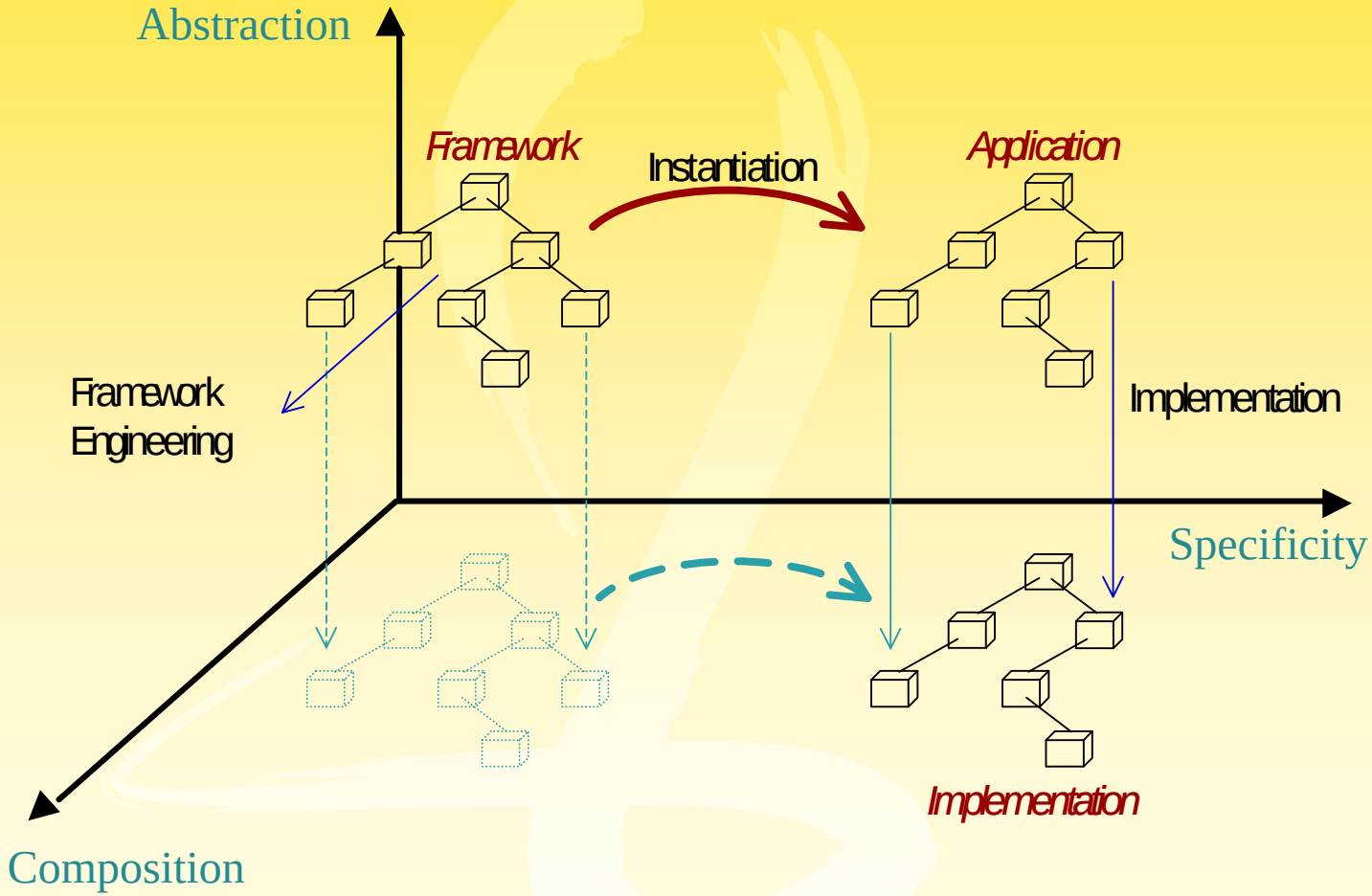
2 Existing Methods

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics



KOBRA

Copyright © KOBRA Group 2000

Method Prescriptiveness

KobrA Overview

1 Challenges

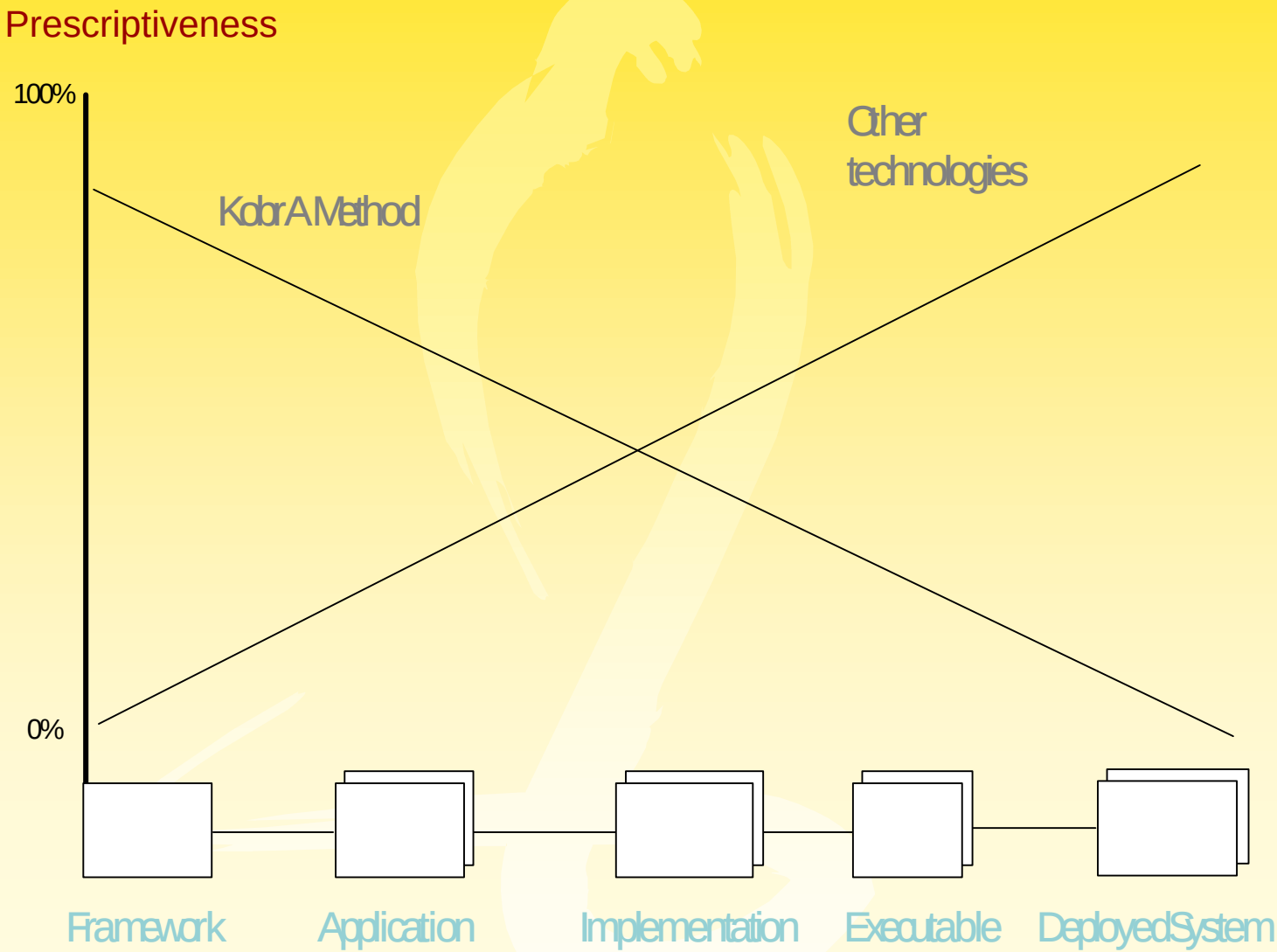
2 Existing Methods

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics



KOBRA

Copyright © KOBRA Group 2000

Main Characteristics

KobrA Overview
1 Challenges
2 Existing Methods
3 Basic Principles
4 Framework Engineering
5 Product Lines
6 Characteristics

- Simple and Systematic
  - Strict separation of concerns
  - UML compatible
- Uniform treatment of systems and components
  - component assembly = component creation
  - Fractal-like product, Recursive process
- Incremental support for product lines
  - Incremental addition/resolution of decision models
- Integrated Quality assurance
  - Inspections, testing, quality modeling
- Flexible Implementation Approach
  - Incremental introduction of component concepts
  - Compatible with most implementation technologies (including JavaBeans, CORBA, COM)

KOBRA

Copyright © KOBRA Group 2000

Further Information

KobrA Overview

1 Challenges

2 Existing Methods

3 Basic Principles

4 Framework Engineering

5 Product Lines

6 Characteristics

Method Web Page

- www.iese.fhg.de/KobrA

Forthcoming Book

- Atkinson et. al., *Component-Based Product Line Engineering with the UML*, Addison-Wesley, 2001

Papers

- C. Atkinson, J. Bayer, O. Laitenberger and J. Zettel, Component-Based Software Engineering: The KobrA Approach, CBSE'2000 , Limerick, Ireland. 200
- C. Atkinson, J. Bayer and D. Muthig, "Component-Based Product Line Development: The KobrA Approach," SPLC1, Pittsburgh, August. 2000

Contact Person

- Dr. Oliver Laitenberger
- +49 6301 707 223, Oliver.Laitenberger@iese.fhg.de

KOBRA

Copyright © KOBRA Group 2000