

# Specification Proposals for Customizable Business Components

**Jörg Ackermann**  
University of Augsburg, Germany

Jörg Ackermann: Specification Proposals for Customizable Business Components. WCEM'03 2003 / 1

## Overview

Introduction

Specifying customizing data

Specifying customizing services

Specifying customizing impact

Summary

Jörg Ackermann: Specification Proposals for Customizable Business Components. WCEM'03 2003 / 2

## Customizable Business Components

- **Adaptation of components often necessary**
- **Our focus: Customizing (setting parameters) on business level**
  - Planned adaptation
  - Assigning values to predefined data fields
  - Influences structure and behavior
  - Impacts many business transactions
- **Properties of customizing**
  - No implementation knowledge required
  - Fits well to easy black box reuse
  - Widely used technique (for instance in SAP R/3)
- **Customizing options must be specified**

Jörg Ackermann: Specification: Proposals for Customizable Business Components. WCEM'03 2003 / 3

## Business level customizing

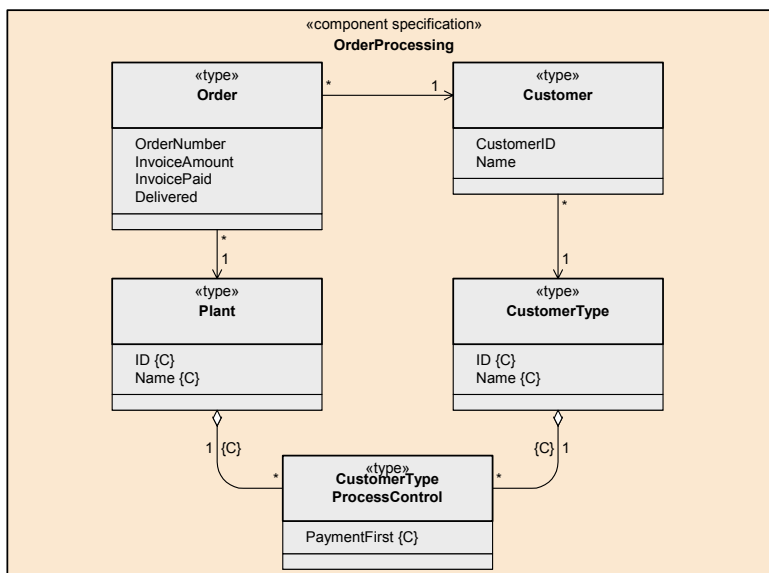
- ***Customizing on business level* = settings related to business tasks of component**
- **Examples:**
  - Definition of organizational units (e.g. companies, plants)
  - Selection between different variants of a business process
  - Definition of business control data (e.g. daily working hours of warehouse)
- **Examples for technical settings:**
  - Database in use, Operating system, Type of component middleware
- **We consider business level customizing**

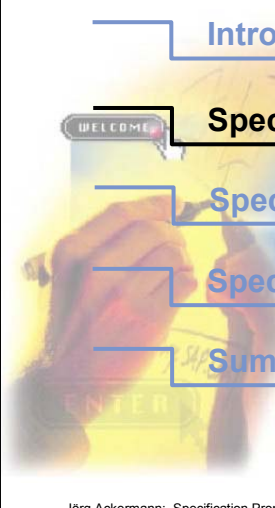
Jörg Ackermann: Specification: Proposals for Customizable Business Components. WCEM'03 2003 / 4

## General approach

- **Identify specification relevant facts of customizing**
  - For customizing of business components only limited experiences available
  - Extensive experiences from COTS-software like SAP R/3
  - Investigation of customizing of SAP R/3
- **Set general requirements for specification proposals**
  - Use techniques from memorandum as widely as possible
  - Make specification of business and customizing aspects as close as possible
- **Integrate specification proposals into framework of memorandum**

## Specification example: Component OrderProcessing





Introduction

**Specifying customizing data**

Specifying customizing services

Specifying customizing impact

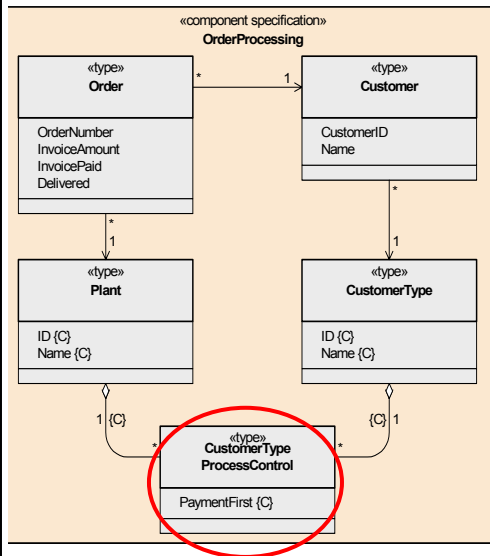
Summary

Jörg Ackermann: Specification Proposals for Customizable Business Components. WCEM'03 2003 / 7

## Customizing groups (1)

- **Customizing group = collection of customizing fields**
  - Form logical unit on a semantic level
  - Will be maintained together
  - Groups can have many instances
- **Example: Customizing group *CustomerTypeProcessControl***
  - Regulates how orders are handled
  - Customizing fields: Plant, CustomerType, PaymentFirst

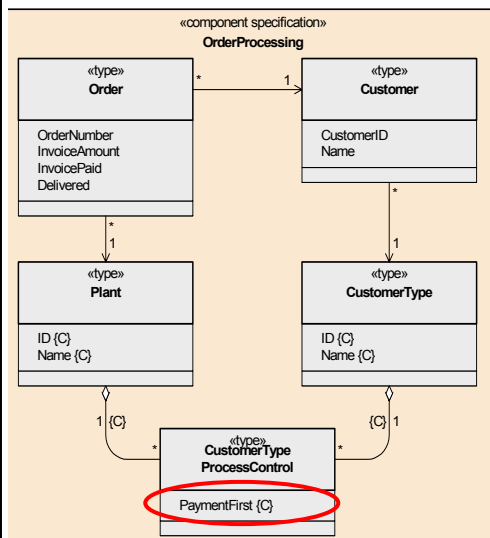
## Customizing groups (2)



Jörg Ackermann: Specification Proposals for Customizable Business Components. WCEM'03 2003 / 9

- **Issue: Customizing groups**
- **Customizing groups mapped to types**
  - Represented in UML diagram
  - Instances of types naturally represent group instances
- **Example: Customizing group *CustomerTypeProcessControl***

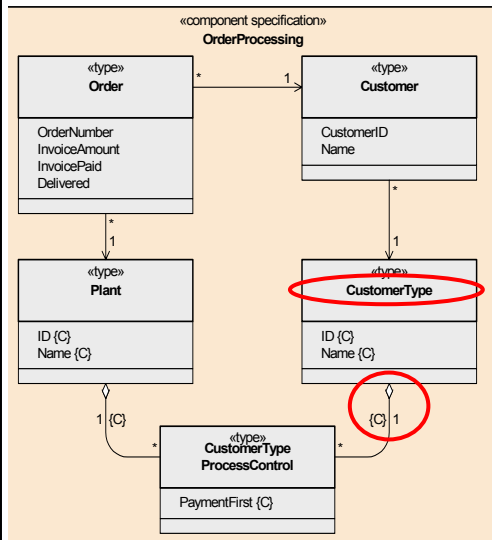
## Customizing fields (1)



Jörg Ackermann: Specification Proposals for Customizable Business Components. WCEM'03 2003 / 10

- **Issue: Customizing fields with fixed value range**
- **Fields are mapped to attributes**
  - Belonging to type
  - Labeled by {C}
  - Value range expressed by data type
- **Example: Customizing field *PaymentFirst***
  - Is of type boolean

## Customizing fields (2)



- **Issue: Customizing fields with customizing dependent value range**
- **Fields are mapped to an association**
  - Name of field = rolename or name of type
  - Labeled by {C}
  - Value range expressed by type at association end
- **Example: Customizing field CustomerType**
  - Value range of type CustomerType

Jörg Ackermann: Specification Proposals for Customizable Business Components. WCEM'03 2003 / 11

## Dependencies

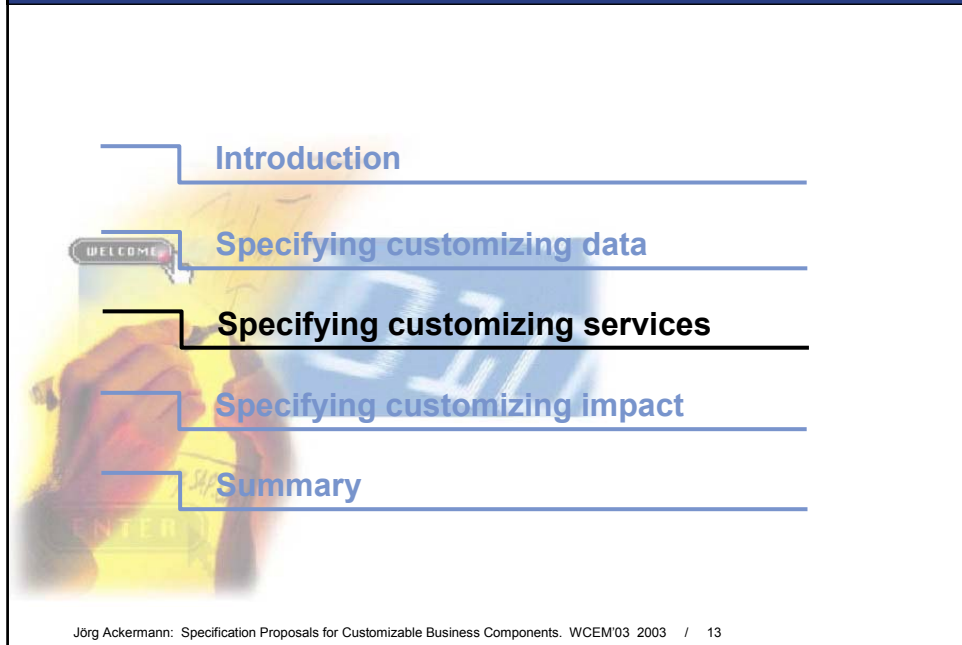
- **Issue: Dependencies between customizing fields**
- **Mapped to invariants**
  - Expressed by OCL expression
  - Logically augment Information Model (type diagram)
  - Part of behavioral level
- **Example for Plant:**
  - Plants are uniquely identified by its ID

OrderProcessing

```

inv: Plant->forall(p1, p2 | p1 <> p2
    implies p1.ID <> p2.ID)
  
```

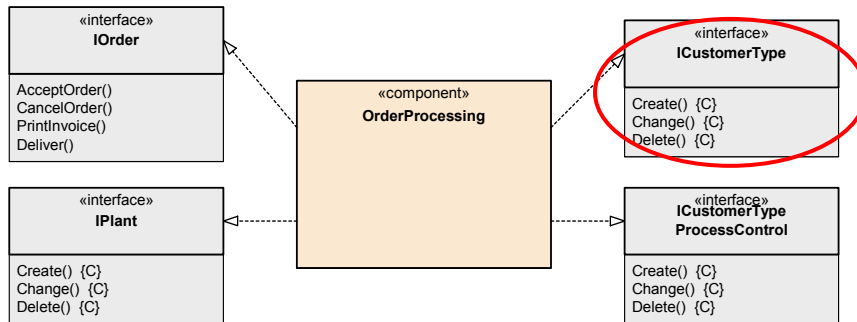
Jörg Ackermann: Specification Proposals for Customizable Business Components. WCEM'03 2003 / 12



## Customizing services

- **Issue: Customizing services**
  - Means to manipulate customizing data
  - Conceptually represent customizing activities
- **Services mapped to operations**
  - Only manipulate one customizing group
- **Operations collected to interfaces**
  - All services manipulating one customizing group
  - Typical services: create, change, delete

## Interfaces for customizing services



- **Interfaces and operations shown in UML interface model**

- Together with the business services (see interface *IOrder*)
- Distinguished by label {C}

- **Example: Interface *ICustomerType***

- Customizing services *Create*, *Change* and *Delete* manipulate customer types

## Specification of customizing services

- **Customizing services listed on task level**

- Explanation of purpose and effect

- **Signature described on interface level**

- **Restrictions regarding operation of one service**

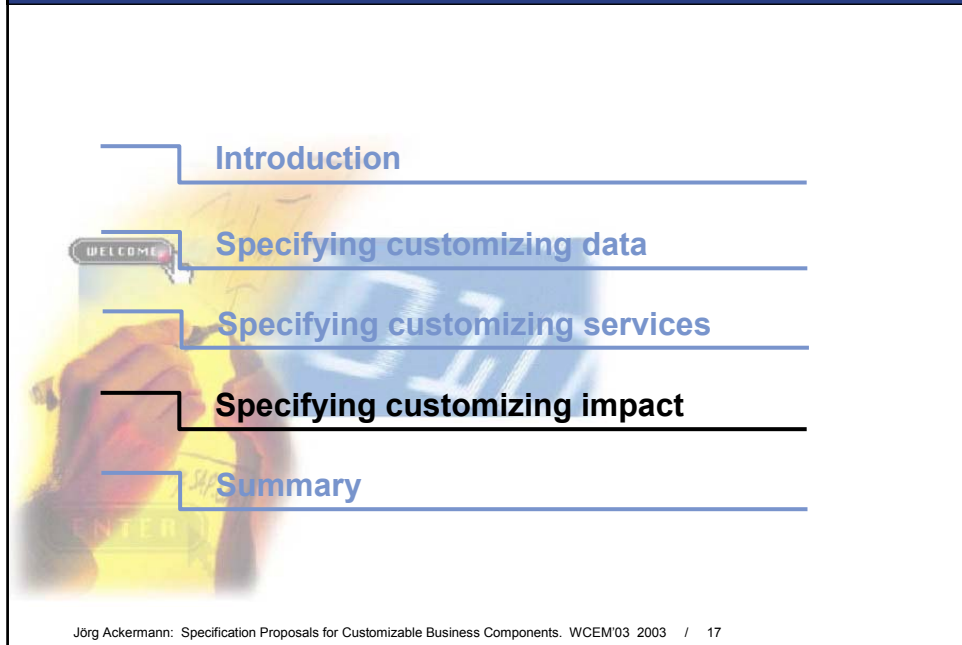
- Expressed as OCL pre and postconditions
- Part of behavioral level

- **Restrictions regarding interplay of several services**

- Expressed as OCL pre and post conditions with temporal operators
- Part of coordination level



## Overview



Introduction

Specifying customizing data

Specifying customizing services

**Specifying customizing impact**

Summary

## Impact of customizing

- **Customizing settings impact business services**
- **Types of possible impact:**
  - Structure of business entities and their relations
  - Business services: behavior, required input parameters, results
  - Accepted order of different business services
- **Impact on business services must be specified**
- **Consequences for objects on**
  - Behavioral and coordination level
  - Possibly for interface and quality level

## Specification proposals (1)

- **Business data / services and Customizing data / services considered together**
  - One united information model and one united interface model
- **Dependencies between Business services and customizing data analogously expressed as other dependencies**
- **Example:**
  - If payment is necessary first, then no delivery without payment

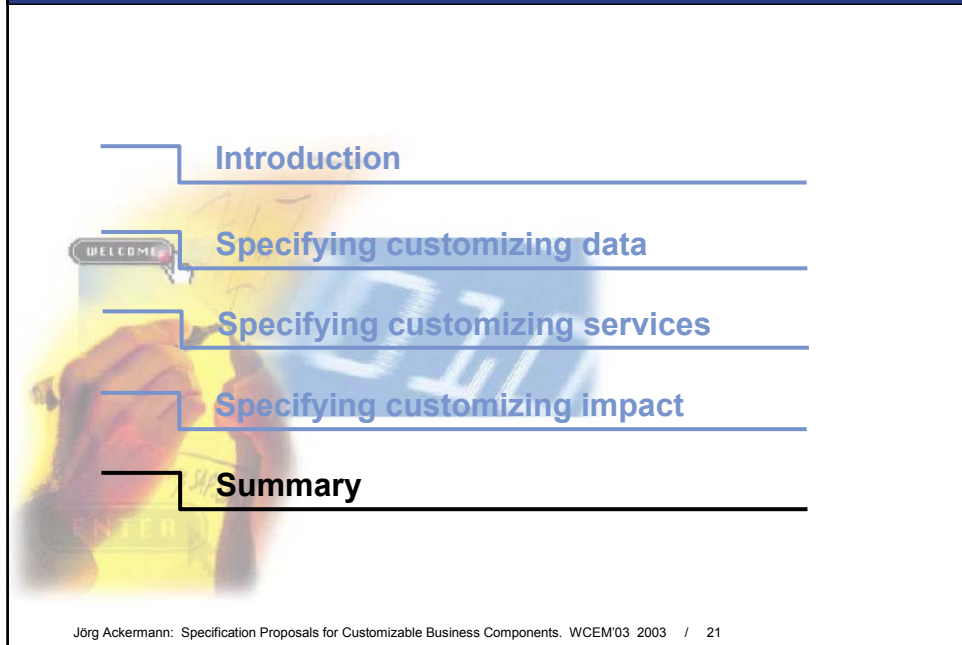
OrderProcessing::IOrder::Deliver(at: Order)

```
pre: at.InvoicePaid = true or  
      CustomerTypeProcessControl->any(pc | pc.Plant = at.Plant and  
      pc.CustomerType = at.Customer.CustomerType).PaymentFirst = false
```

## Specification proposals (2)

- **Feasibility of approach**
  - Approach technically possible for all types of customizing dependencies
  - For simple dependencies well suited
  - For complex dependencies: direction of further research

## Overview



## Summary

- **Identified customizing facts to be specified**
- **Made specification proposals**
  
- **Mapping of customizing facts onto UML model elements**
- **No new notation techniques necessary**
  - Techniques from memorandum can be used
  - Some enhancements suggested
- **Proposals how to uniformly specify customizing facts**
  
- **Outlook:**
  - Specification of complex dependencies shall be improved