

Architektur von Informationssystemen

Praktikum 1

Sven Dettmers

Milena Dreier

Tobias Meurer

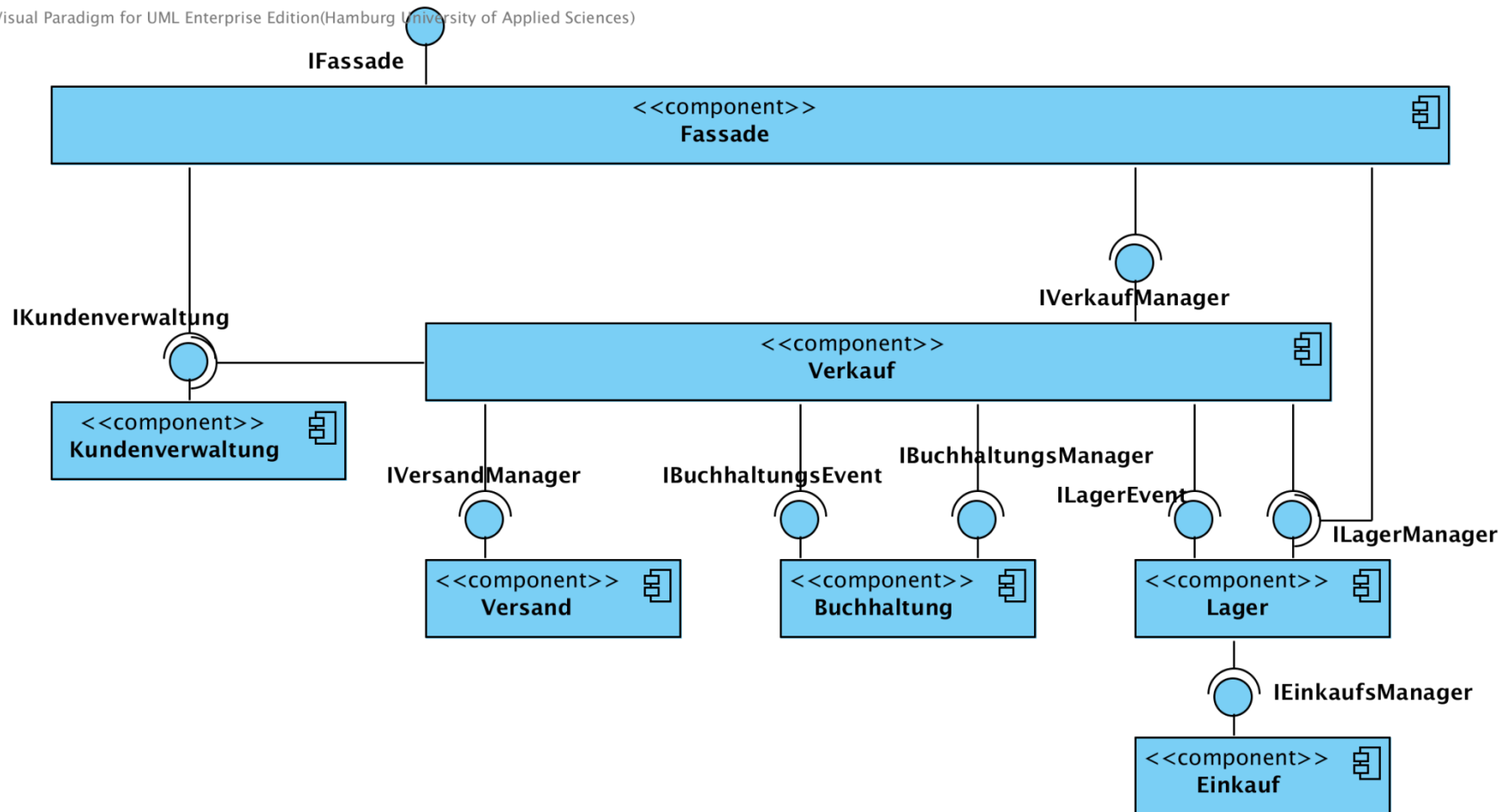
Inhalt

- HES
 - Komponentenschnitt
 - Schnittstellen
- OR-Mapping

HES

Komponentenschnitt - Übersicht

Visual Paradigm for UML Enterprise Edition (Hamburg University of Applied Sciences)



Komponentenschnitt - Klassen

- Verkauf
 - Auftrag
 - Angebot
 - Menge
- Lager
 - Produkt
 - Wareneingangsmeldung
 - Warenausgangsmeldung

Komponentenschnitt - Klassen

- Einkauf
 - Orderbuch
 - Ordersatz
 - Lieferant
 - Einkaufsinfosatz
 - Bestellung
- Kundenverwaltung
 - Kunde

Komponentenschnitt - Klassen

- Buchhaltung
 - Rechnung
 - Zahlungseingang
- Versand
 - Lieferung
 - Transportauftrag

Schnittstellen

- ```
public interface IVerkaufManager {
 public AngebotsTyp erstelleAngebot(String
 aKundenNr, Date aGueltigBis, Date
 aGueltigAb, Map<ProduktTyp, int>
 aProduktListe, KundenTyp aKunde);
 public AuftragTyp erstelleAuftrag(
 AngebotsTyp aAngebot);
}
```



# Schnittstellen

- ```
public interface ILagerManager {  
    public void bucheWareneingang(  
        LieferscheinTyp aLieferschein);  
    public void reserviereProdukteFuerAuftrag(  
        AuftragsTyp aAuftrag);  
}
```
- ```
public interface ILagerEvent {
 public void
 schreibeFuerWarenReserviertEventEin(
 AuftragsTyp aAuftrag, Listener
 aListener);
}
```

# Schnittstellen

- ```
public interface IEinkaufsManager {  
    public BestellungTyp bestelleProdukt(  
        ProduktTyp aProdukt);  
    public void bestaetigeWareneingang(  
        BestellungTyp aBestellung,  
        WareneingangsmeldungTyp  
        aWareneingangsmeldung);  
}
```

- ```
public interface IKundenverwaltung {
 public void erstelleKunde(KundenTyp
 aKunde);
 public KundenTyp getKunde(String
 aKundenID);
}
```

# Schnittstellen

- ```
public interface IBuchhaltungsManager {  
    public void erstelleRechnung( AuftragsTyp  
        aAuftrag);  
    public void zahlungseingangBuchen( double  
        aBetrag, String aKundenNr);  
}
```
- ```
public interface IBuchhaltungsevent {
 public void
 schreibeFuerRechnungBezahltEventEin(
 AngebotsTyp aAngebot);
}
```

# Schnittstellen

- ```
public interface IVersandManager {  
    public void erstelleLieferung( AuftragsTyp  
        aAuftrag) ;  
}
```

Schnittstellen

- ```
public interface IFassade {
 public AngebotsTyp erstelleAngebot(
 Map<ProduktTyp, int> aProduktListe,
 Date aGueltigAb, Date aGueltigBis);
 public AuftragsTyp
 erstelleAuftragAusAngebot(String
 aAngebotsNr);
 public void zahlungseingangErstellen(
 double aBetrag, String aKundenNr);
 public void bucheWareneingang(
 LieferscheinTyp aLieferschein);
}
```

# OR-Mapping

---

# Umgebung

- Programmiersprache: Java 1.7
- OR-Mapping-Framework: Hibernate 4.1.4
- IDE: IntelliJ IDEA 12
- Datenbank: SQLite 3.7.2

# Hibernate-Beispiel

## Live Präsentation