

PB007

12. cvičení

Lucie Fabriková

FI MU, podzim 2012

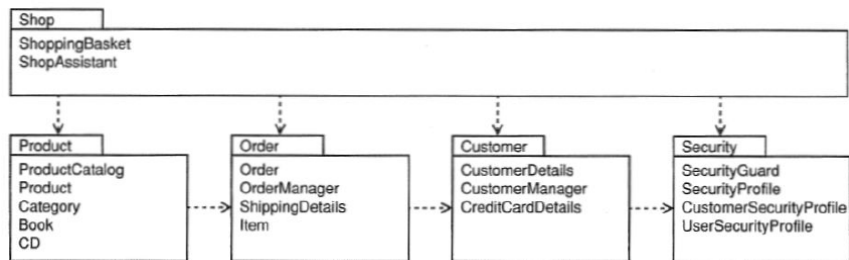
14. 12. 2012

- [https://is.muni.cz/auth/el/1433/podzim2012/PB007/um/35424437/35424448/12\\_DiagramNasazeni.pdf](https://is.muni.cz/auth/el/1433/podzim2012/PB007/um/35424437/35424448/12_DiagramNasazeni.pdf)
- <https://is.muni.cz/auth/el/1433/podzim2012/PB007/um/35424437/ro/index.qwarp?prejit=2822999>

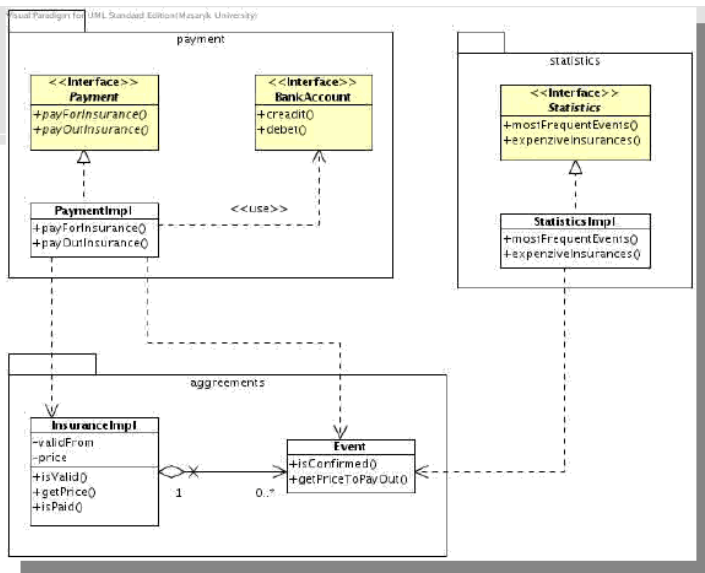
## Balík

- seskupení významově souvisejících prvků
  - případy užití,
  - **třídy**,
  - realizace případů užití
- mezi balíky mohou být závislosti: *use*, *import*, *access*,...
- cílem je minimalizovat závislosti mezi balíky
- tvorba balíku – hledání tříd se silnými vazbami:
  1. dědičnost, 2. kompozice, 3. agregace, 4. závislosti

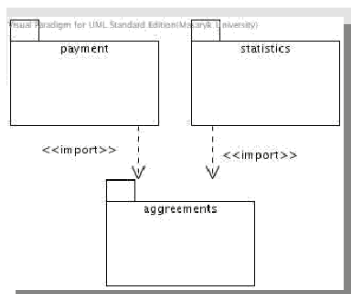
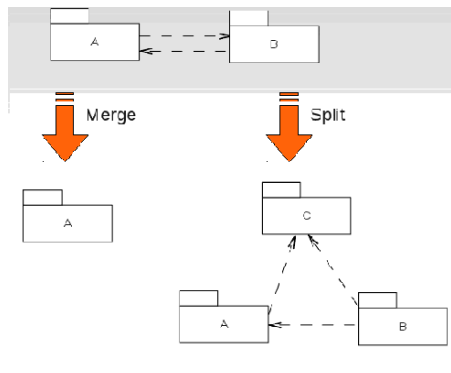
# Diagram balíku



# Diagram balíku



- Odstranění cyklických závislostí mezi balíky
- Diagram balíků z předchozího slidu



Přehlednější diagram balíků

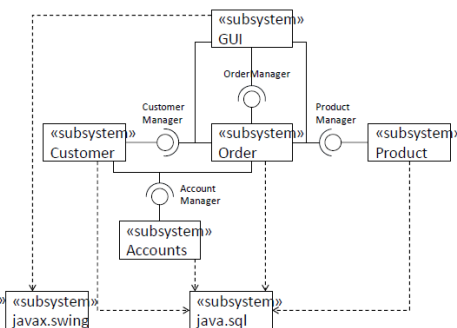
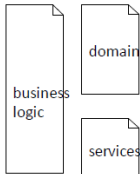
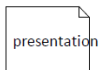
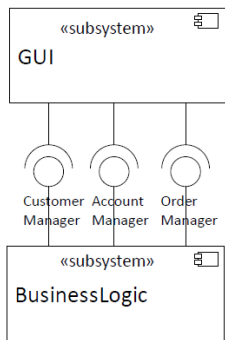
**Diagram komponent (Component diagram)** zobrazuje způsob (hierarchického) rozdělení systému na samostatné části a komunikační vazby mezi nimi, čímž definuje architekturu systému.

## Diagram obsahuje:

- Komponenty – softwarové komponenty (fyzicky samostatné části systému), vnitřně soudržné, navenek komunikující jen přes definovaná rozhraní
- Rozhraní – rozhraní pro komunikaci mezi komponentami (oddělení specifikace od implementace)
  - *provided* rozhraní definují nabízenou funkcionalitu (služby/metody)
  - *required* rozhraní definují požadovanou funkcionalitu (služby/metody)
- Vazby mezi rozhraními – napojení *required* rozhraní na *provided* rozhraní



# Rozhraní mezi komponentami



© Clear View Training 2010 v2.6

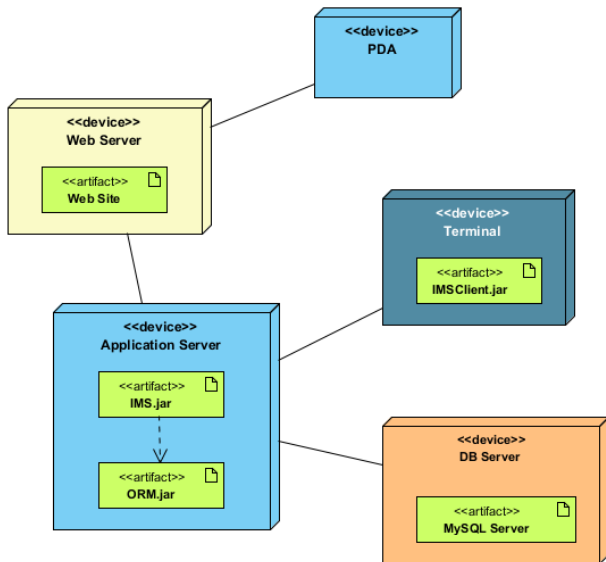
83



## Uzly

- hardwarové uzly, na kterých bude systém spouštěn
- do každého uzlu lze vnořit další uzel
- stereotypy uzlů:
  - *device*,
  - *execution environment*
- součásti uzlu:
  - **komponenta**,
  - **rozhraní** pro komunikaci s komponentami
  - **artefakt**
- vazby mezi uzly: **asociace** – komunikační kanály, **závislosti**; mohou obsahovat název komunikačního protokolu

# Diagram nasazení



## **Komponenta** (zopakování z předchozích slidů)

- fyzicky samostatné součásti systému – softwarové komponenty
- black-box jednotka funkcionality
- vyznačuje se vyžadovaným a poskytovaným rozhraním
- logická (subsystem) nebo fyzická (EJB) entita

## **Artefakt**

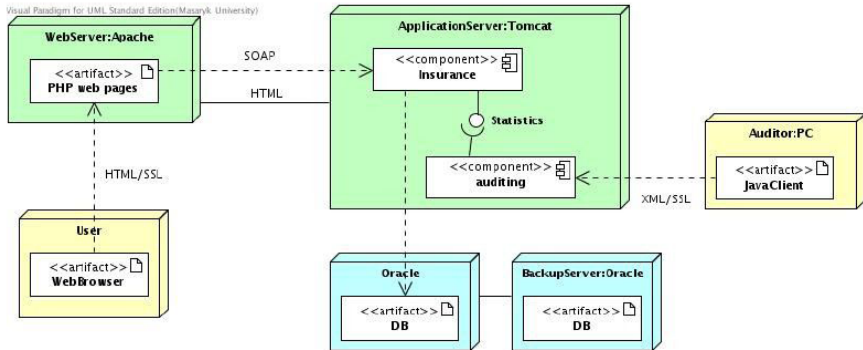
- konkrétní projev komponenty
- např. zdrojový kód, .jar, skript,...

## **Rozhraní mezi komponentami** (zopakování z předchozích slidů)

- poskytované / vyžadované
- vazby mezi rozhraními – napojení vyžadovaného rozhraní na poskytované

# Diagram nasazení

Visual Paradigm for UML Standard Edition (Masaryk University)



- Upravte návrhový model tříd podle připomínek.
- Vytvořte 2 diagramy: **diagram balíků** a **diagram nasazení**.
- *Ad diagram balíků*: v diagramu tříd doplňte balíky a přesuňte do nich jednotlivé třídy. Následně vytvořte samostatný diagram balíků, který bude obsahovat jen samotné balíky (bez tříd) a závislosti mezi nimi.
- *Ad diagram nasazení*: rozmyslete, z jakých komponent/subsystémů je vytvořen váš systém a přes jaké rozhraní komunikují. Diagram nasazení mapuje tyto komponenty/subsystémy na jednotlivé hw uzly.
- **do středy 19. 12. 2012, 20:00, odevzdávárna PB007 → Týden 12**
- ve formátu *cisloSkupinky\_deployDiagram.pdf*