

# Objektové metody modelování

UML diagramy 1. část

Tutoriál II



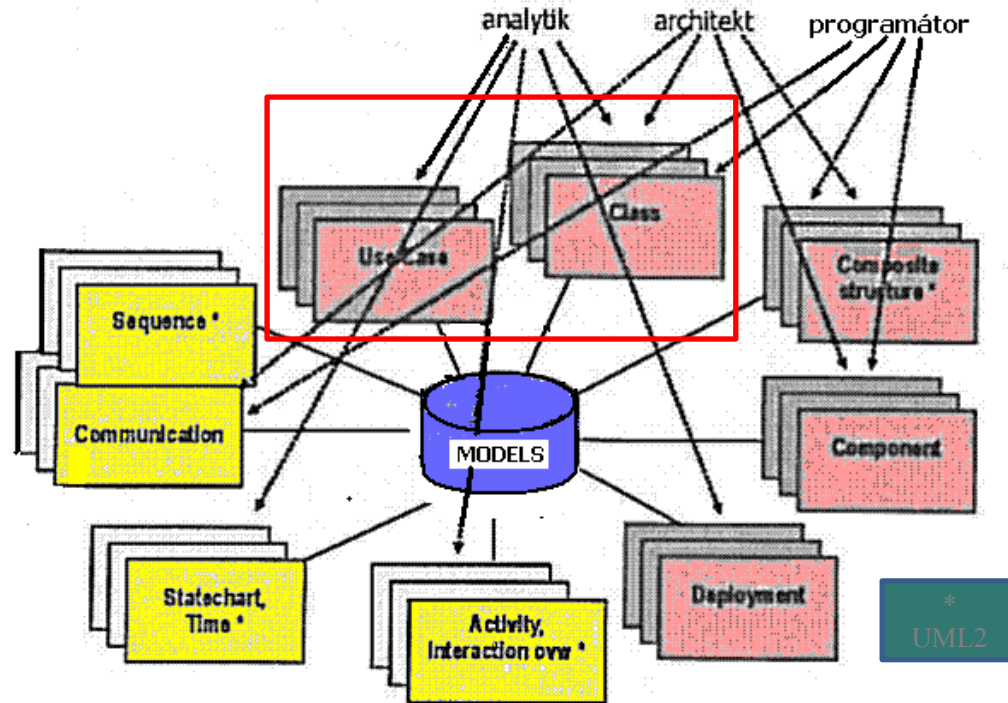
**SLEZSKÁ  
UNIVERZITA**

**OBCHODNĚ PODNIKATELSKÁ  
FAKULTA V KARVINĚ**

**RNDr. Zdeněk Franěk, Ph.D.**

<b>Statická struktura</b>	<b>Dynamické chování</b>	<b>Správa modulů</b>
diagram tříd (Class Diagram)	use case diagram	balíčky (Packages)
objektový diagram (Object Diagram)	sekvenční diagram (Sequence Diagram)	subsystémy (Subsystems)
komponentový diagram (Component Diagram)	diagram činností (aktivit) (Activity Diagram)	modely (Models)
diagram nasazení (Deployment Diagram)	diagram spolupráce (Collaboration Diagram)	
	stavový diagram (Statechart Diagram)	

Zdroj: <http://objekty.vse.cz/Objekty/MethodikyANotace-UMLDiagramy>





**Use-Case  
Model**



**Analysis  
Model**



**Design  
Model**



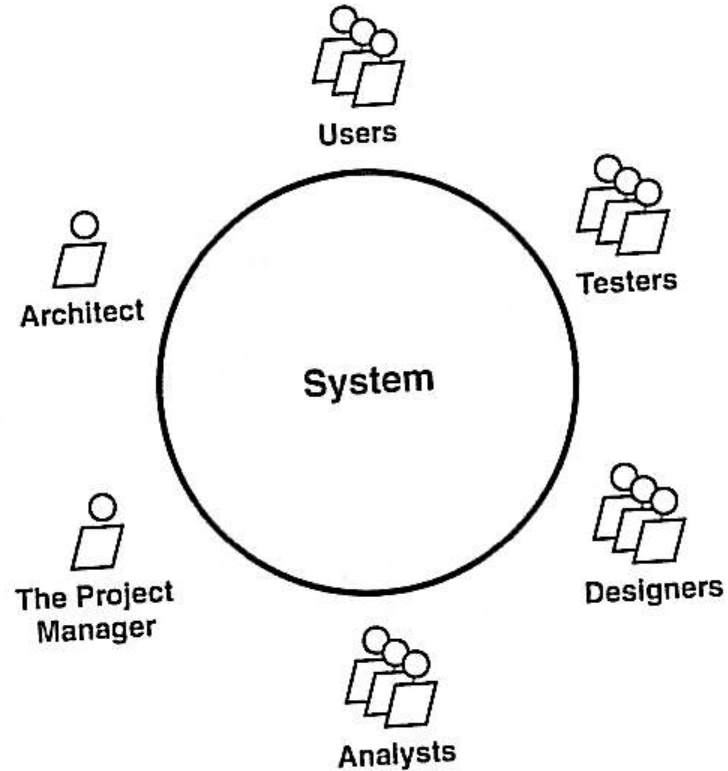
**Deployment  
Model**



**Implementation  
Model**

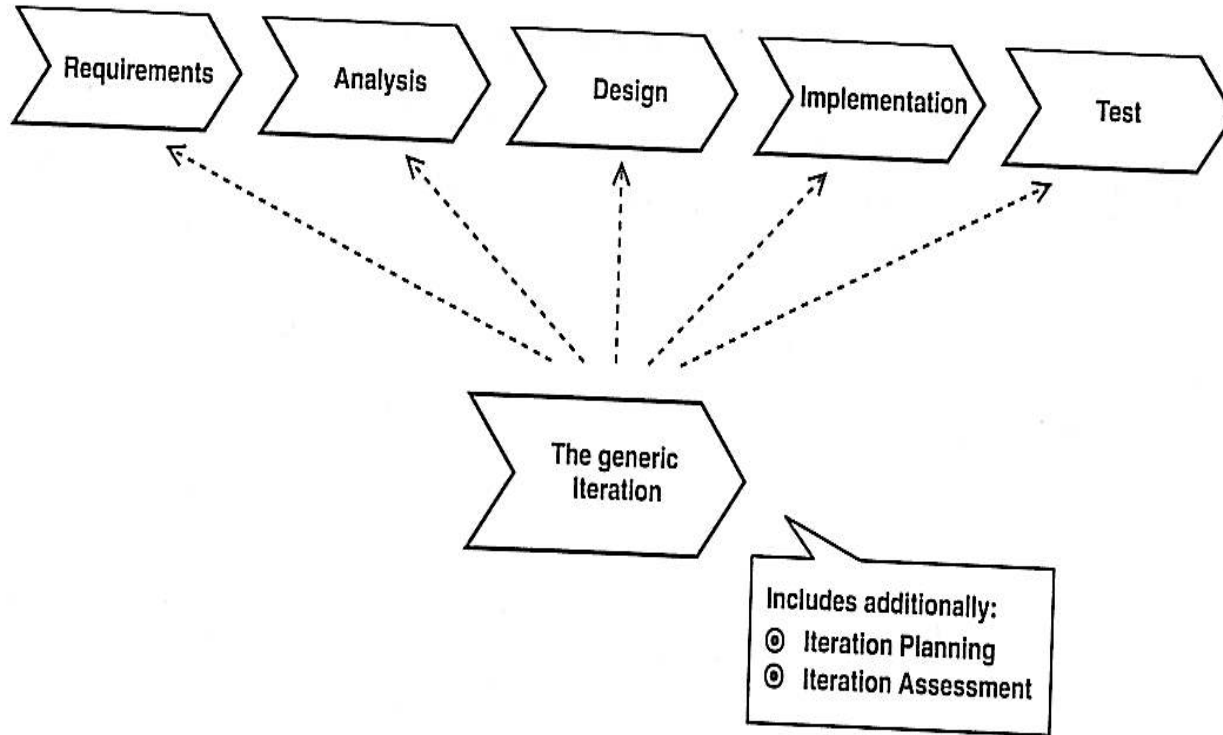


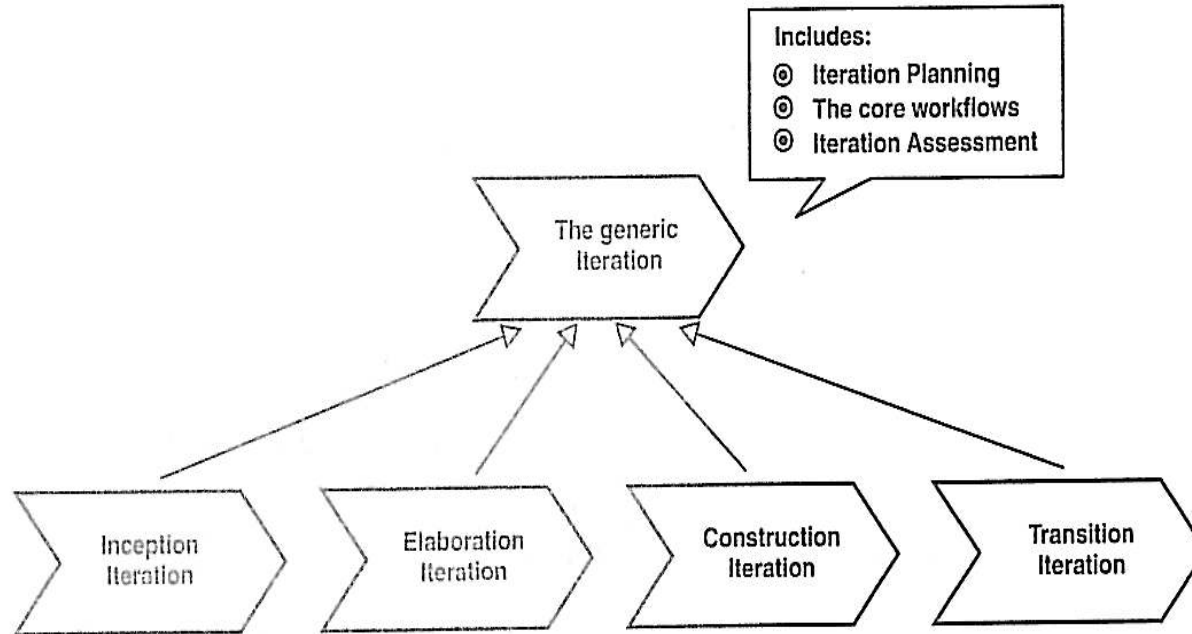
**Test  
Model**





# Iterační proces postupu vývoje SW









## **Seskupení:**

*Dekompozice systému v objektově orientovaném návrhu je řešena tvorbou seskupení tříd (packages).*

*Diagram seskupení zobrazuje strukturu a vztahy mezi seskupeními navrhovaného informačního systému.*

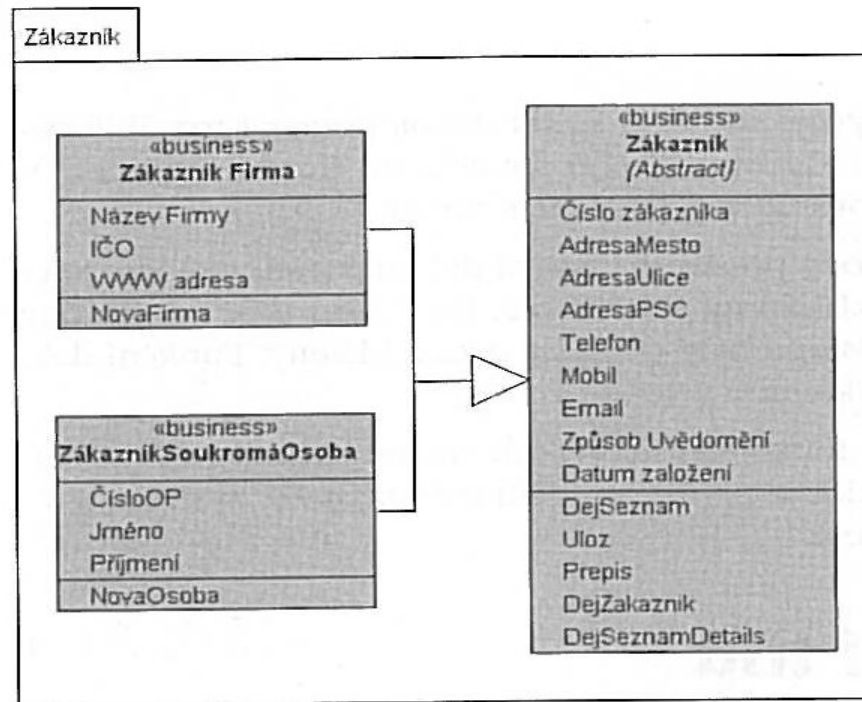
*Seskupení tříd se používá pro třídy, jejichž objekty spolu logicky souvisí a komunikují mezi sebou, tyto třídy jsou pak schopny poskytnout ucelený balík služeb (rozhraní).*

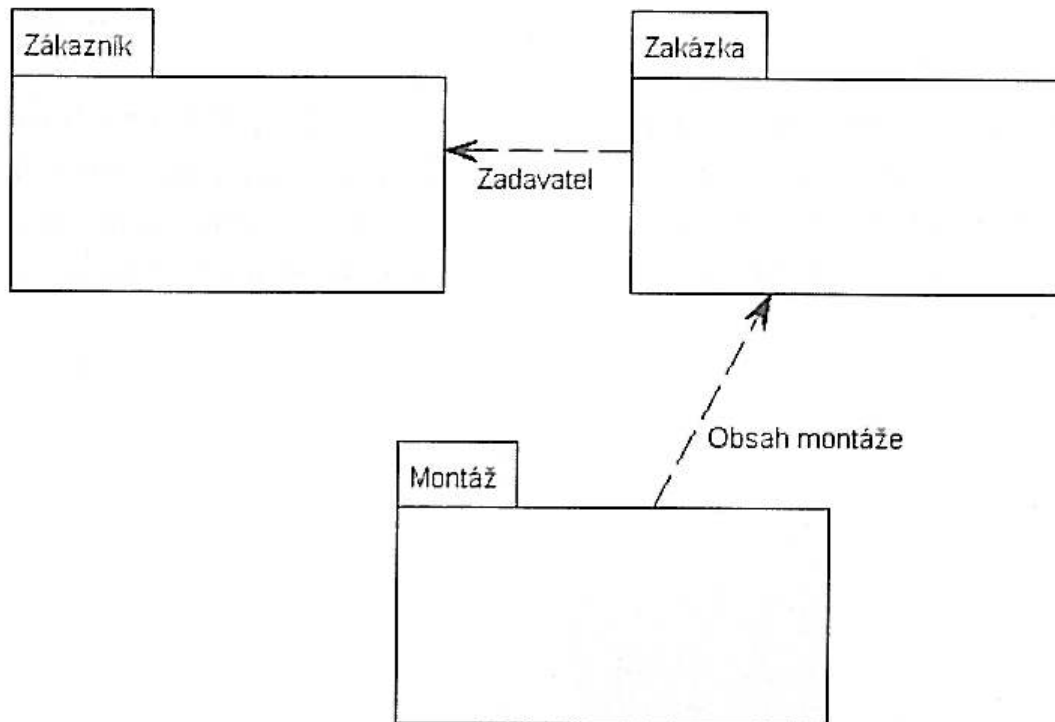
*Seskupení tříd se často používají při návrhu větších projektů.*

*Seskupení tříd a návrh jejich rozhraní je nezbytným modelováním při komponentovém vývoji informačních systémů.*

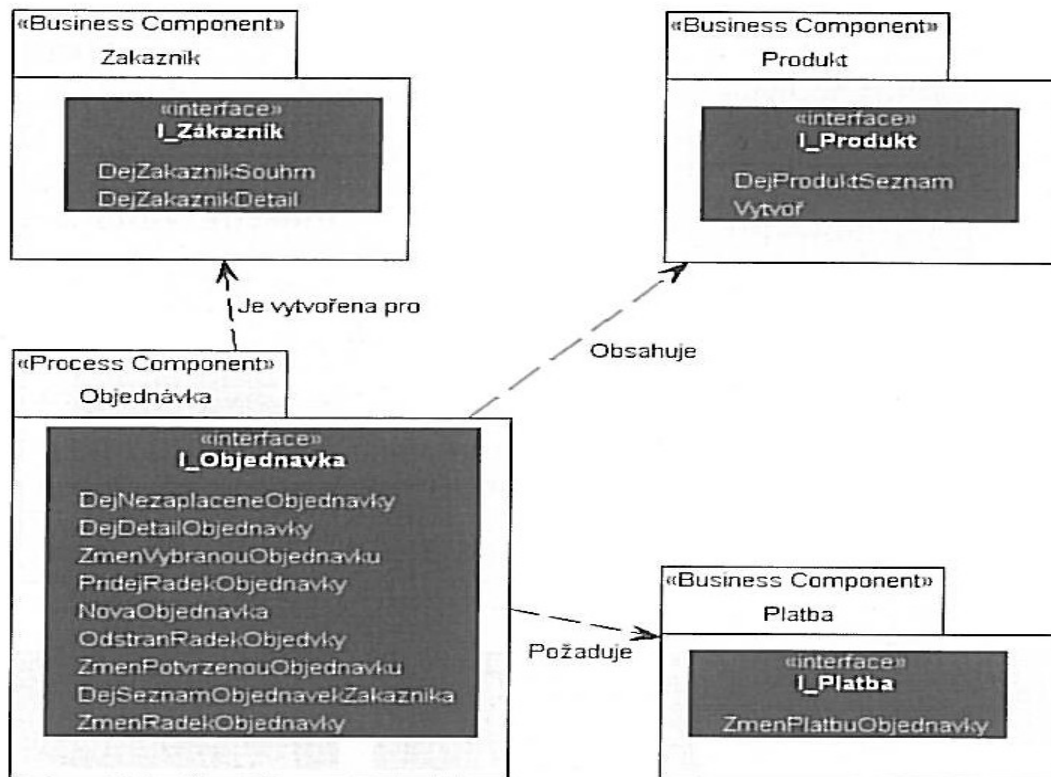
---

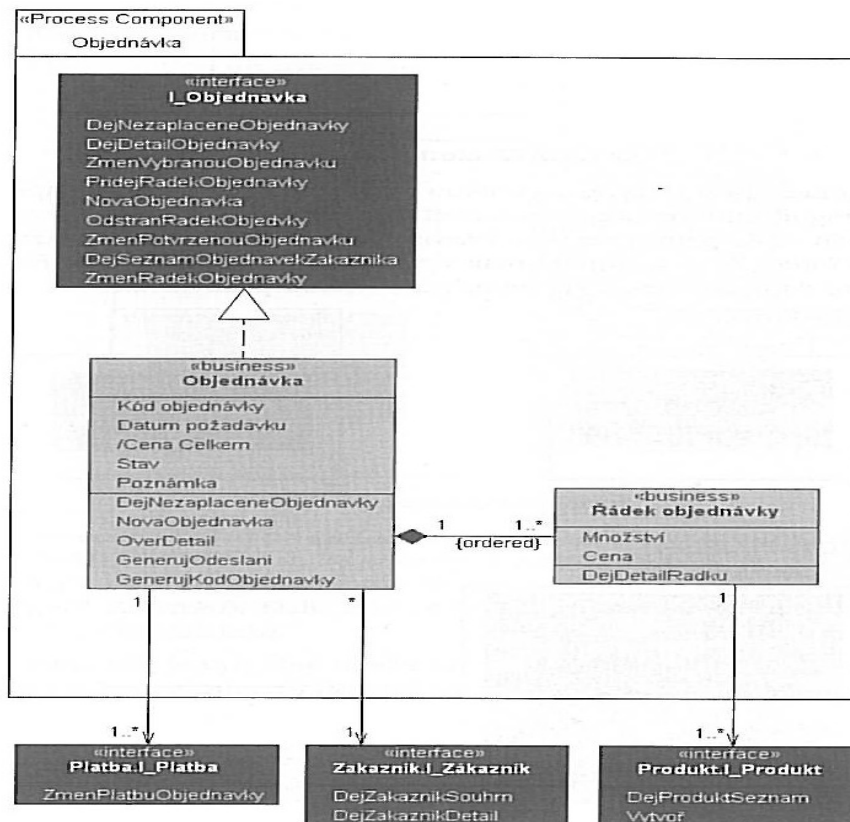
# Příklad diagramu seskupení tříd „Zákazník“



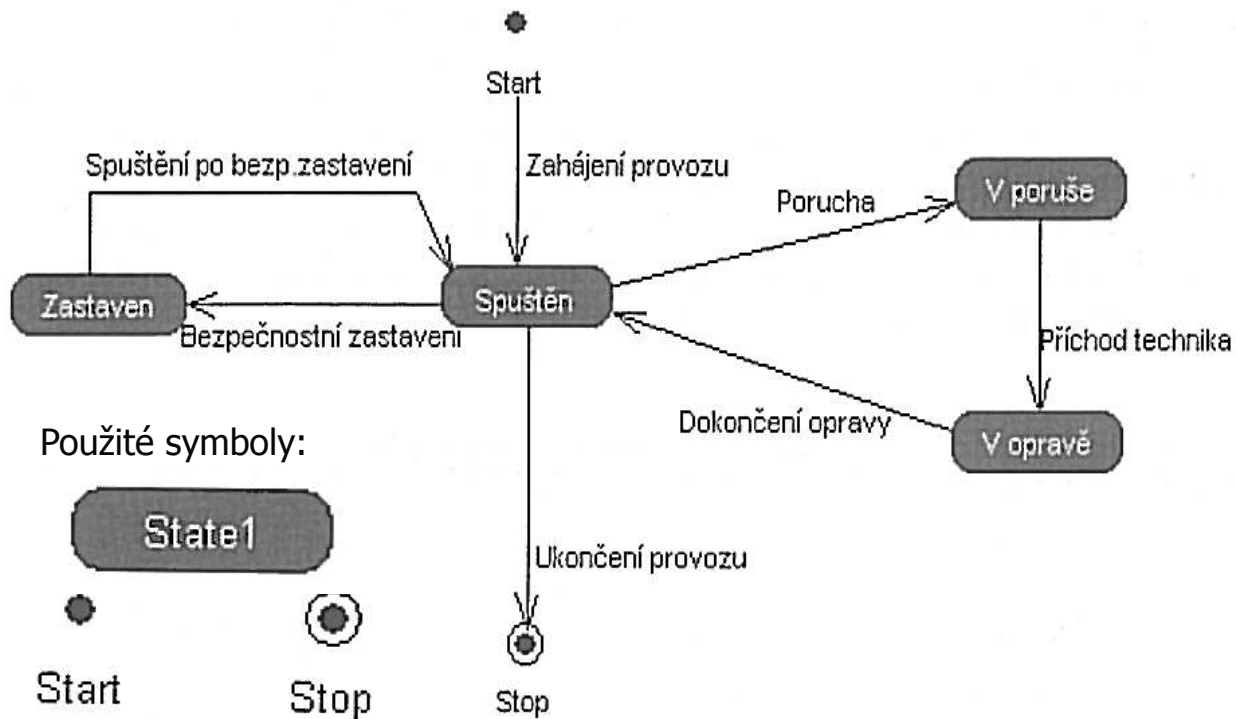


# Seskupení a jejich rozhraní

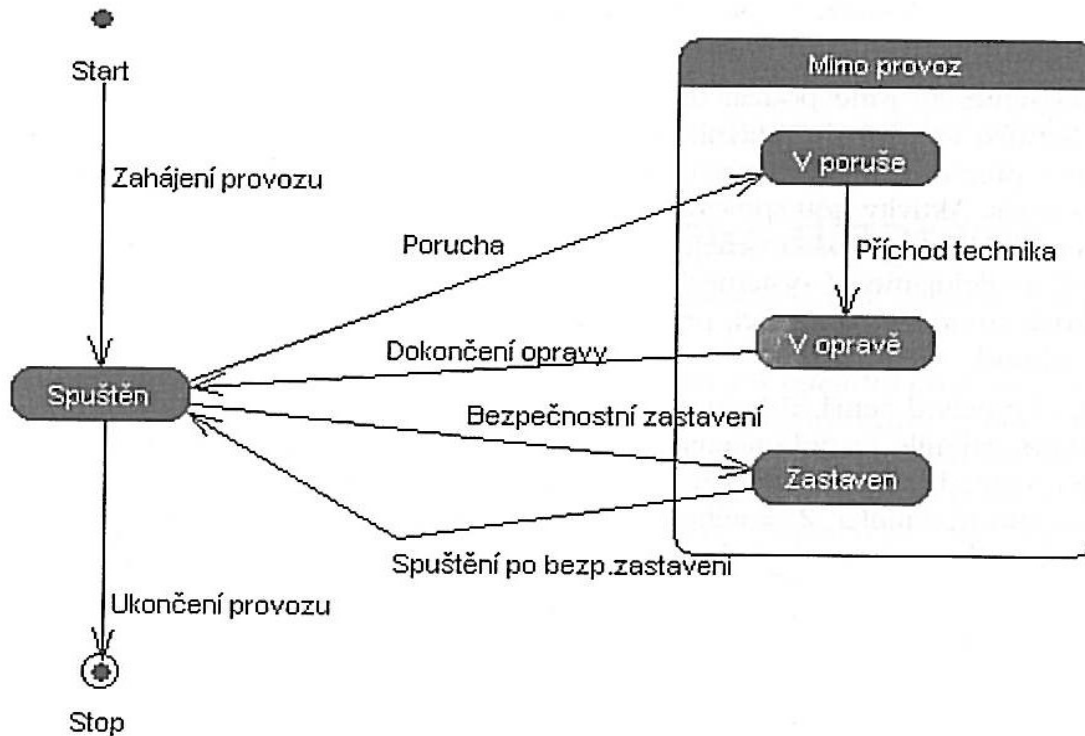




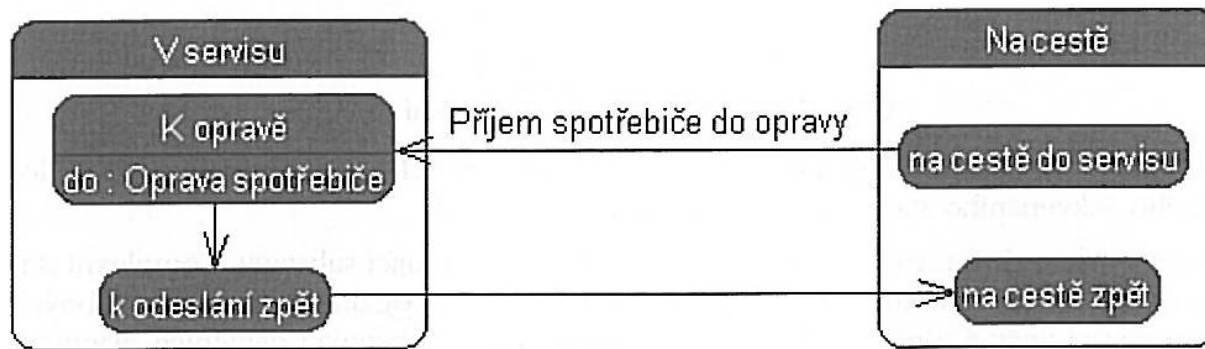
# Příklad stavového diagramu



# Složený stav s jeho podstavy



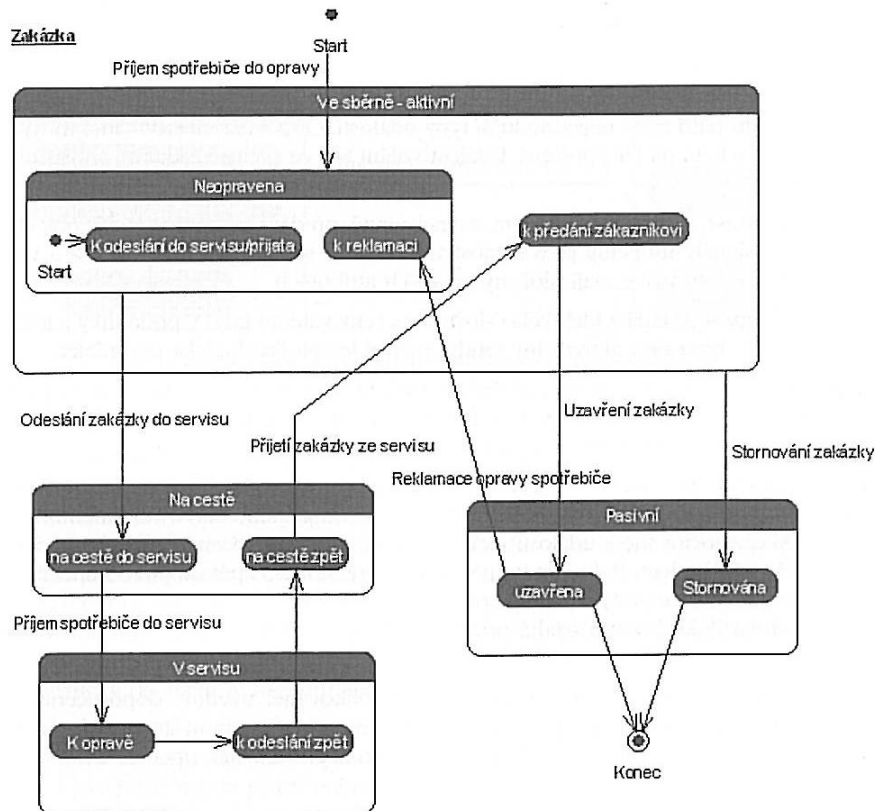
# Přechody – příklad výřez stavového diagramu

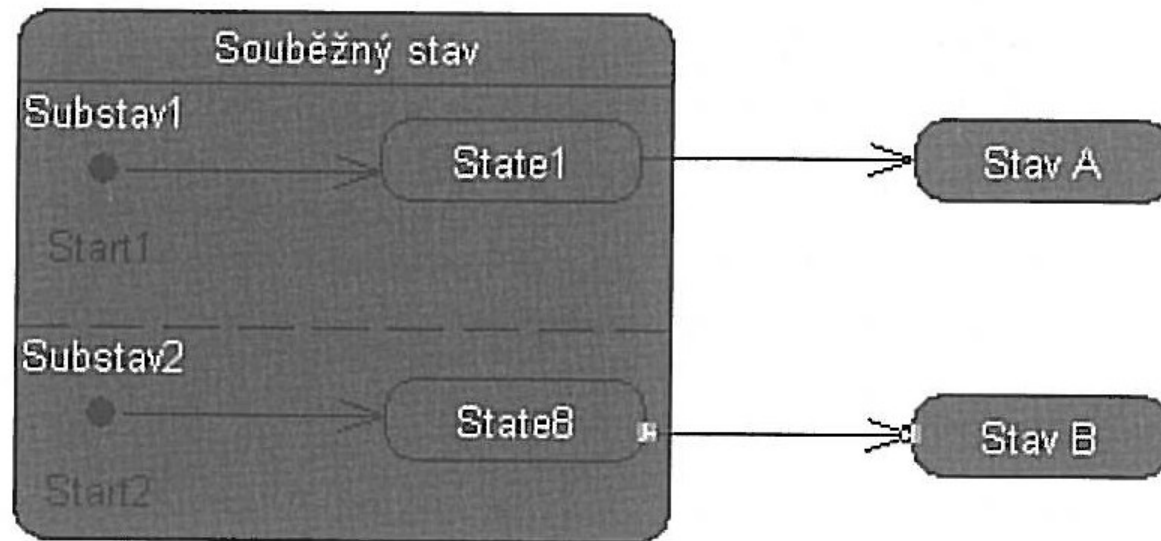




- Událost volání Nejjednodušší typ události, odpovídá metodě dané třídy, je vlastně požadavkem na její spuštění, ve svém důsledku spouští sérii akcí
- Signální událost Reprezentuje asynchronně předávané zprávy mezi objekty
- Událost změny Skládá se s klíčového slova when, podmínky a akce. Je aktivována, pokud je logická podmínka splněna.
- Časová událost Událost může být vygenerována po určité době, after (20 minut)
- Entry je spuštěna automaticky, kdykoliv se do daného stavu dostaneme přechodem
- Exit Akce asociované s událostí exit jsou analogicky provedeny, kdykoliv je daný stav opuštěn přechodem.

# Příklad - Stavový diagram pro třídu zakázka





## **Stavové diagramy:**

*Na stavových diagramech se setkáte se symboly:*

- *stavů*
- *přechodů*
- *událostí*

*stav = situace v životě objektu*

*přechod = změna stavu objektu na základě stimulu (události)*

*událost = něco významného, co se stane v jistém čase a nemá trvání*

*Pokročilé techniky stavových digramů:*

- *složené sekvenční stavy – zavádí nadstavy a jejich podstavy*
- *souběžné stavové diagramy – pro modelování nezávislých sad chování objektů*

*Využití stavových diagramů doporučujeme pro objekty se složitým životním cyklem jako kontrolu ostatních modelů.*

## Viz public a cvičení

Příklad – Přijímací řízení

Příklad class diagram – rozvrh

Příklad use case – bankomat

Příklad diagram aktivit (úvod) - zadávání diplomových a  
bakalářských prací do IS/STAG

---



Kanisová H., Muller M.: UML srozumitelně, Computer press, 2004, ISBN 80-251-0231-9

Jacobson I., Booch G., Rumbaugh J.: The Unified Software Development Process, Addison Wesley Longman, 1999, ISBN 0-201-57169-2

Software ke cvičení:

CASE: Rational software development platform, IBM

Architect Enterprise fmy Sparksystem, trial verze

Viz <http://www.sparxsystems.com.au/products/ea/trial.html>

- UML (Unified Modeling Language) je notace
  - vizualizace
  - specifikace
  - tvorby
  - dokumentaceproduktů s podílem software
- Přidané hodnoty
  - Otevřený standard OMG
  - Pokrývá celý životní cyklus
  - Podpora nástroji

- Model je

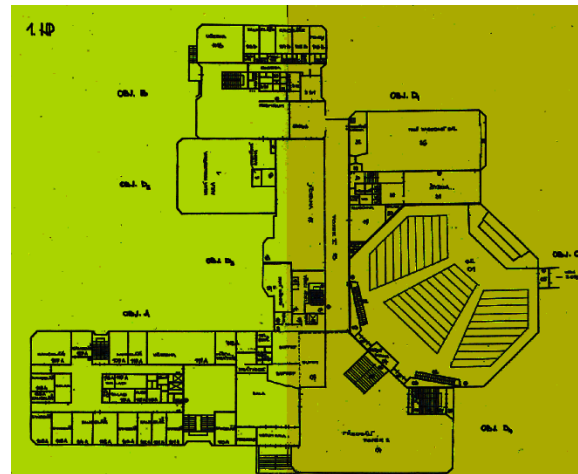
- Zjednodušená reprezentace reality
- Popis systému z jednoho úhlu pohledu
- Abstrakce s konkrétním účelem

- Proč modelujeme

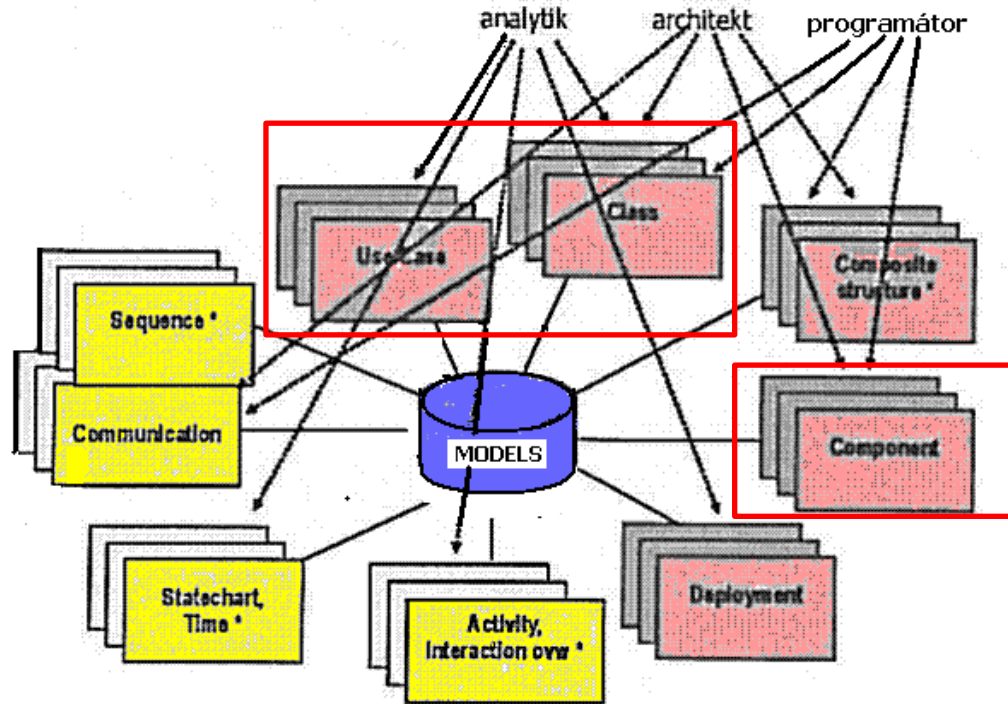
- Usnadnění úvah díky vyšší abstrakci
- Lepší porozumění vytvářenému systému
- Složitý systém není možné vnímat vcelku

- Diagram

- jeden pohled do modelu (1:N)
- grafické znázornění









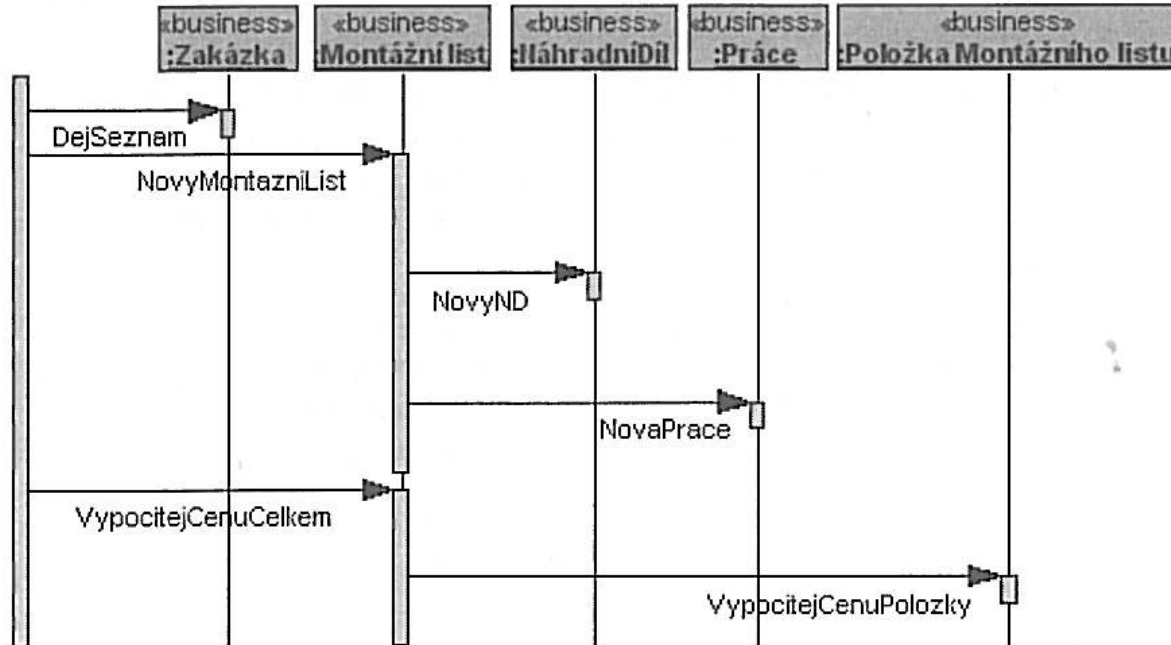
- Pro modelování spolupráce objektů se používají dva základní typy interakčních diagramů:
  - Sekvenční diagramy
    - Object Sequence diagram
  - Diagramy objektové spolupráce
    - Object Collaboration Diagrams
- Jedná se o převod slovního popisu scénáře případu užití na model interakce předem identifikovaných tříd.

# Scénář případu užití založit montážní list

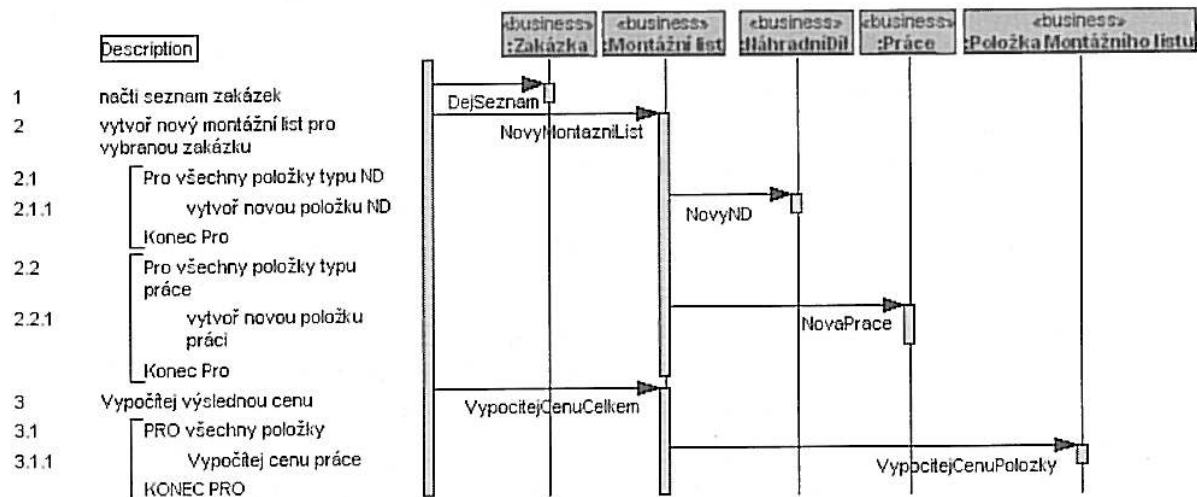


Krok	Role	Akce
1	Uživatel	dá pokyn k zobrazení seznamu existujících zakázek
2	System	zobrazí seznam zakázek
3	Uživatel	vybere zakázku, pro kterou chce vytvořit montážní list a zvolí funkci <i>Založení montážního listu</i>
4	System	založí nový montážní list pro vybranou zakázku (číslo montážního listu přidělí automaticky) a převezme do něj informace zakázky
5	Uživatel	vloží položky montážního listu, které jsou typu „náhradní díl“ a typu „pracovní úkon“
6	System	uloží zadané položky do zakázkového listu
7	Uživatel	spustí funkci <i>Výpočet ceny</i>
8	System	vypočte cenu použitých náhradních dílů, pracovních úkonů a cenu zakázky celkem
8	Uživatel	spustí funkci <i>Tisk montážního listu</i>
9	System	vytiskne montážní list

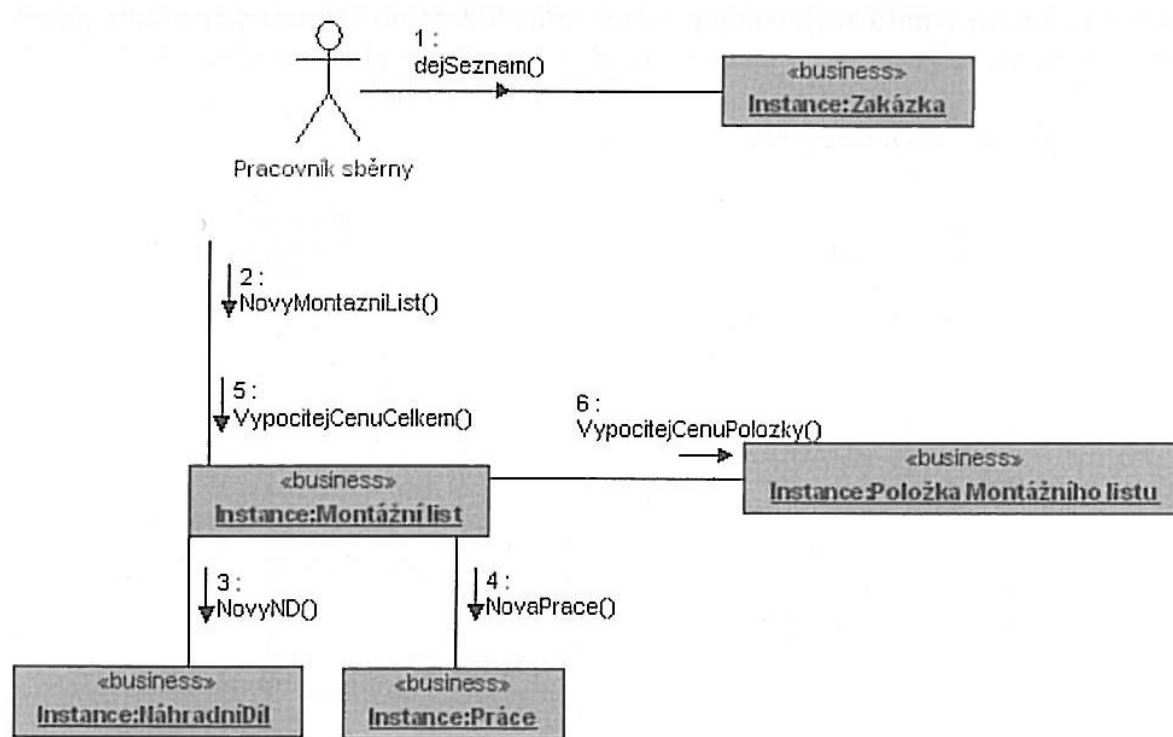
# Sekvenční diagram případu užití založit montážní list



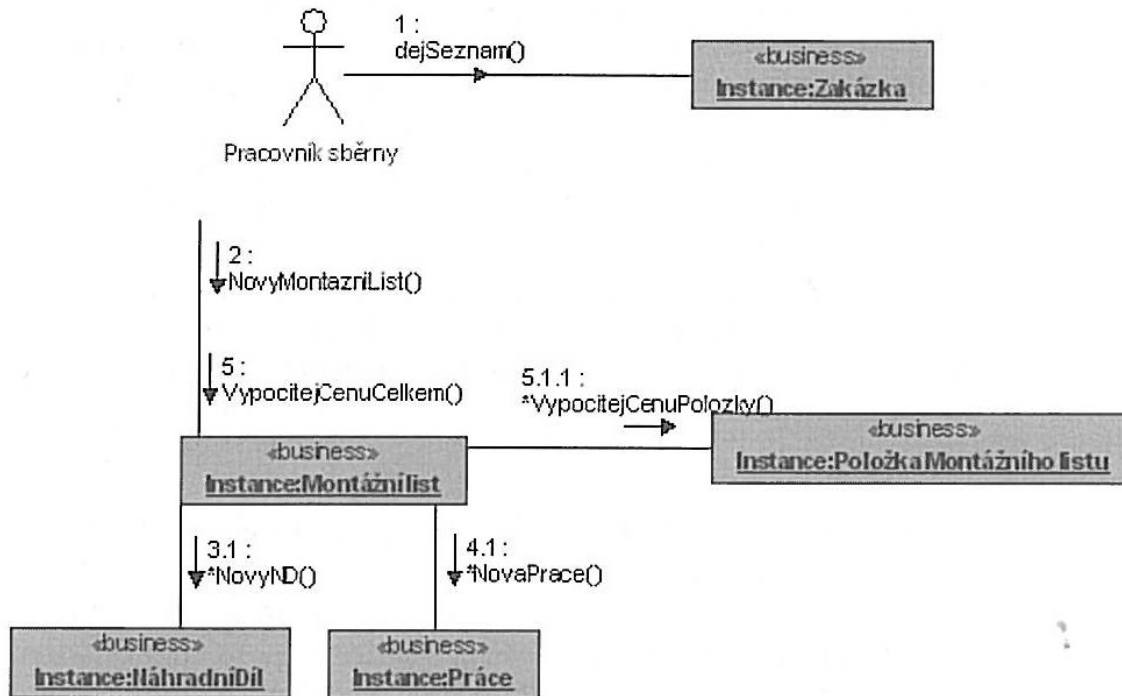
# Sekvenční diagram s popisem kroků scénáře



# Diagram objektové spolupráce případu užití montážní list



# Diagram objektové spolupráce s číslováním zpráv





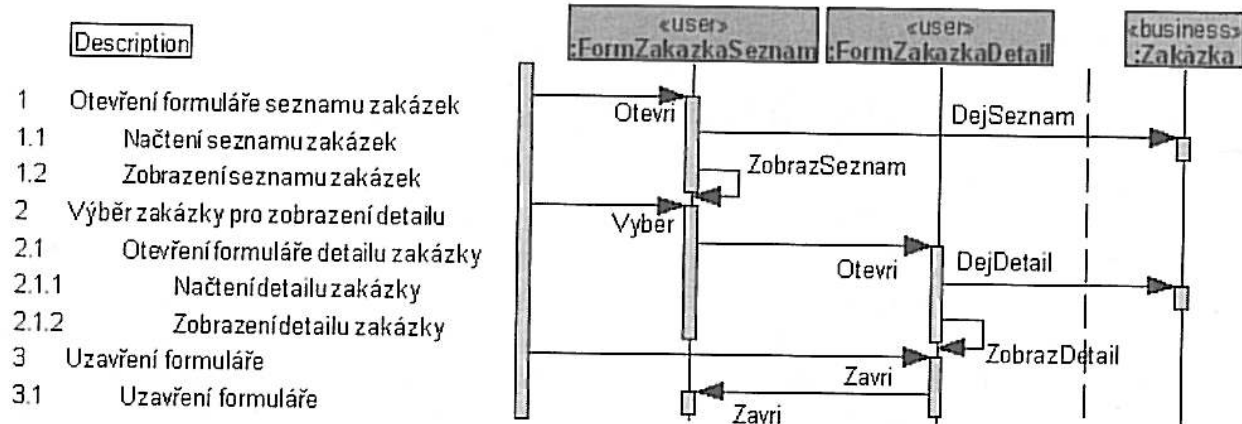
## Případ užití: *Zobrazit detail zakázky*

Krok	Role	Akce
1	Uživatel	dá pokyn k zobrazení seznamu existujících zakázek
2	System	zobrazí seznam zakázek
3	Uživatel	vybere zakázku, pro kterou chce zobrazit detail
4	System	zobrazí detail vybrané zakázky
5	Uživatel	uzavře formulář detailu i seznamu zakázky

---



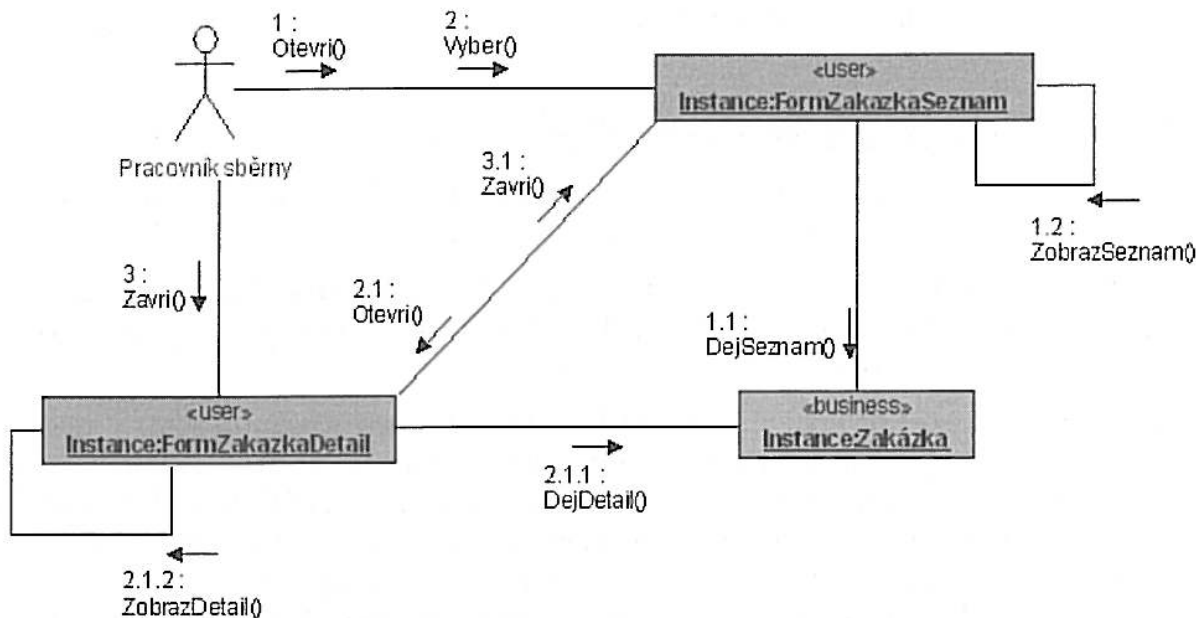
## Zobrazení detailu zakázky s objekty rozhraní



# Diagram objektové spolupráce



## Diagram objektové spolupráce případu užití Zobrazit detail zakázky s objekty rozhraní



Statická struktura	Dynamické chování	Správa modulů
diagram tříd (Class Diagram)	use case diagram	balíčky (Packages)
objektový diagram (Object Diagram)	sekvenční diagram (Sequence Diagram)	subsystémy (Subsystems)
komponentový diagram (Component Diagram)	diagram činností (aktivit) (Activity Diagram)	modely (Models)
diagram nasazení (Deployment Diagram)	diagram spolupráce (Collaboration Diagram)	
	stavový diagram (Statechart Diagram)	

Zdroj: <http://objekty.vse.cz/Objekty/MethodikyANotace-UMLDiagramy>

## Viz cvičení

Příklad class diagram – rozvrh

Příklad use case – bankomat

Příklad diagram aktivit (úvod) - zadávání diplomových a bakalářských prací do IS/STAG



Viz public a složka souborů v elearningu



## PREZENTACE TÉMAT SEMINÁRNÍCH PRACÍ

Příklady na elearning portálu



**Děkuji za pozornost**

**Otázky?**

---