

# Objektové metody modelování

## UML diagramy

Přehled diagramů  
Diagram komponent  
Diagram nasazení

Opakování a rozšíření znalostí UML diagramů  
USE CASE  
SEKVENČNÍ DIAGRAM

Přednáška 7



**SLEZSKÁ  
UNIVERZITA**

OBCHODNĚ PODNIKATELSKÁ  
FAKULTA V KARVINĚ

**RNDr. Zdeněk Franěk, Ph.D.**

**Ing. Tomáš Barčák, Ph.D.**

# UML diagramy *SHRNUTÍ a PROHLOUBENÍ ZNALOSTÍ*



## Přehled diagramů UML:

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)	



Zbývá probrat komponentový diagram a diagram nasazení

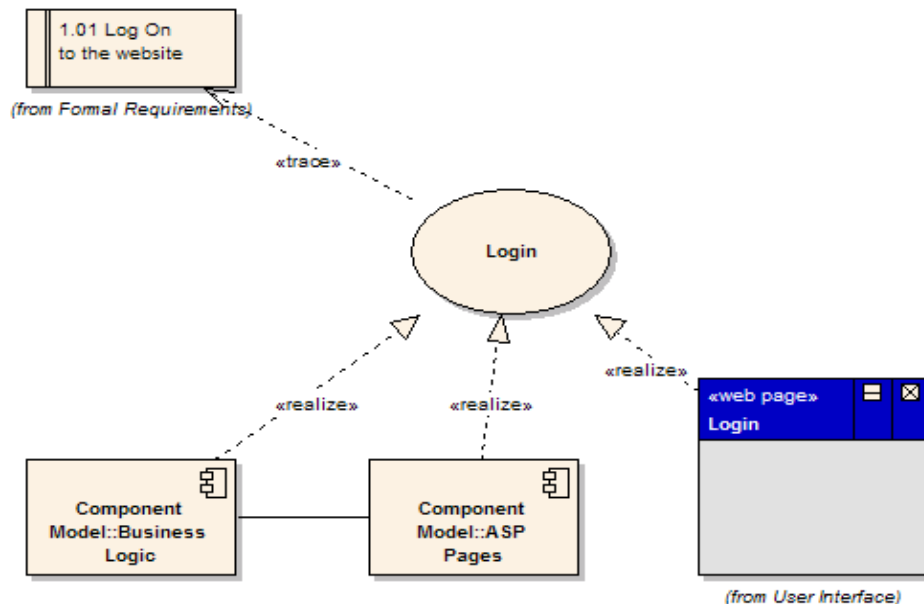


- Model komponent ilustruje softwarové komponenty, které budou využity pro vytvoření systému.
- Tyto mohou být sestrojeny z modelu tříd a napsány od základu jako nové pro systém, nebo mohou být převzaty z ostatních projektů, nebo od dodavatelů tzv. třetích stran.
- Komponenty jsou vysokoúrovňové agregace menších softwarových jednotek, a představují stavební bloky typu 'black box' pro konstrukci software

# Implementační diagramy - komponenty



Případ užití je formálním popisem funkcionality, kterou bude mít systém až bude zkonstruován. Implementační diagram je typicky spojen s případem užití, aby dokumentoval, které prvky návrhu (např. komponenty a třídy) budou implementovat funkcionalitu případu užití v systému.

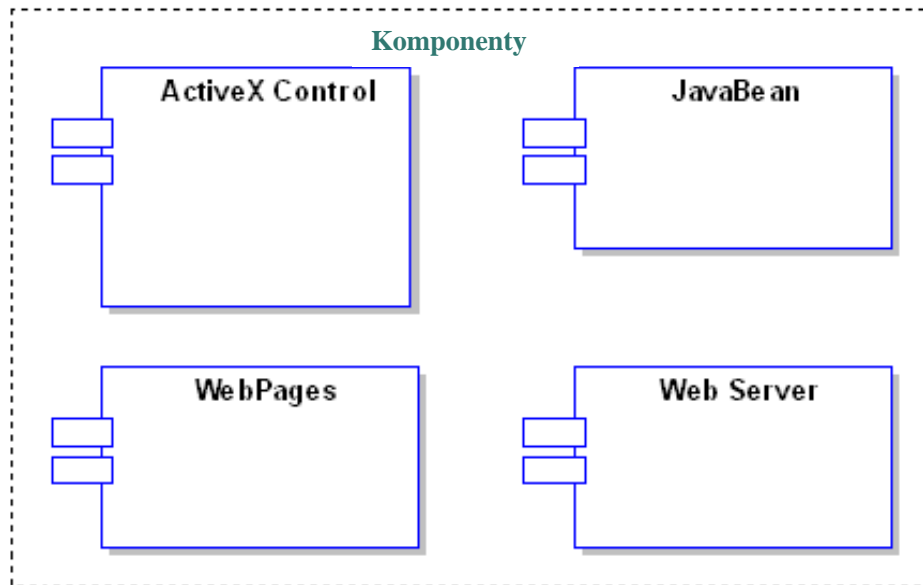


USE CASE Login – přihlášení do systému a komponenty, které přihlášení realizují. Modrá komponenta zajišťuje webovou stránku, kde uživatel, zadává své login údaje a dostává zpět info o přihlášení. Komponenty Business Logic a ASP Pages se podílí na realizaci přihlášení. Komponenta 1.01 Log on to the website sleduje (trace) proces přihlašování.

# Zakreslení komponent



Komponenta může být něco jako prvek ActiveX - buď uživatelské rozhraní nebo server s pravidly.



ActiveX je Framework technologie, kterou vyvinula společnost Microsoft pro sdílení informací mezi různými aplikacemi.

JavaBeans jsou třídy v programovacím jazyku Java, které se nějak pojmenují a k instancím těchto tříd se přistupuje pomocí těchto referencí. JavaBeans se většinou používají k programování Webových aplikací nebo Java Enterprise aplikací (Enterprise Java Beans)

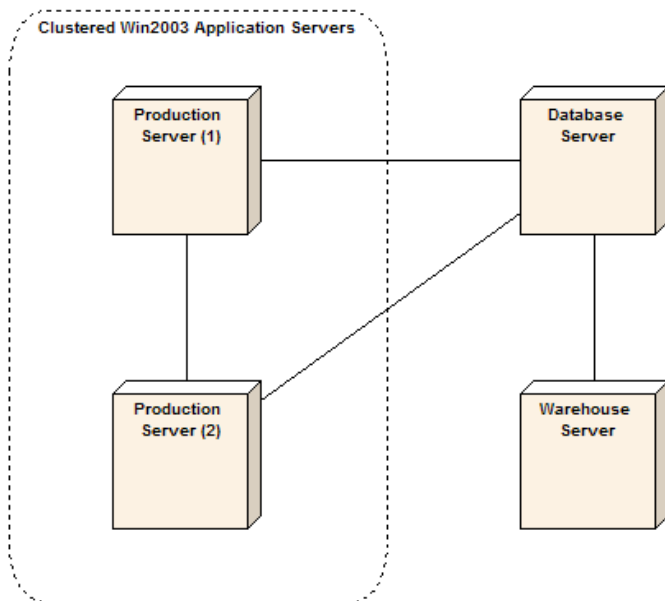
WebPages - Webová stránka je vytvořena pomocí jazyka HTML (v textu neviditelné značky), jehož vzhled je definován kaskádovými styly (CSS), a který se může dynamicky měnit pomocí JavaScriptu (tzv. dynamická webová stránka).

WebServer - Pojem webový server označuje počítač nebo počítačový program (démon), který s klientem (webovými prohlížeči) komunikuje prostřednictvím protokolu HTTP nebo zabezpečeného HTTPS.

# Fyzický Model



Fyzický model resp. model nasazení (deployment) poskytuje detailní model, jakým způsobem budou jednotlivé komponenty rozmístěny v infrastruktuře systému. Zpodrobnuje schopnosti sítě, specifikace serveru, požadavky na hardware a další informace relevantní s ohledem k rozmístění navrhovaného systému.



## Diagram nasazení (Deployment View)

Produkční server je základní server, na kterém jsou hostovány webové stránky nebo webové aplikace a uživatelé k nim přistupují. Je součástí celého prostředí pro vývoj softwaru a aplikací.

Přestože je produkční server spíše omezen na interní použití jako na pracovní server, je otevřen pro přístup koncových uživatelů. Před nasazením na produkční server musí být software nebo aplikace testovány a laděny na pracovní stanici.



DIAGRAMY CHOVÁNÍ

USE CASE

SEKVENČNÍ DIAGRAM

## Diagram užití – use case diagram

---

- Popisuje systém podle toho, co s ním budou uživatelé dělat.
- Je obzvlášť důležitý pro analytickou část vývoje systému.
- Diagram užití si je možné představit jako soubor scénářů pro použití systému.
- Popisuje sled událostí, které inicializuje účastník (aktér).





# Model případů užití

## Use Case – Případ užití

- popisuje navrhovanou funkcionalitu nového systému
- vyjadřuje oddělenou jednotku interakce mezi uživatelem (člověk nebo stroj) a systémem
- je to samostatná jednotka smysluplné činnosti:
  - logovat se do systému,
  - registrace v systému
  - vytvoření objednávky

Všechno uvedené jsou případy užití.

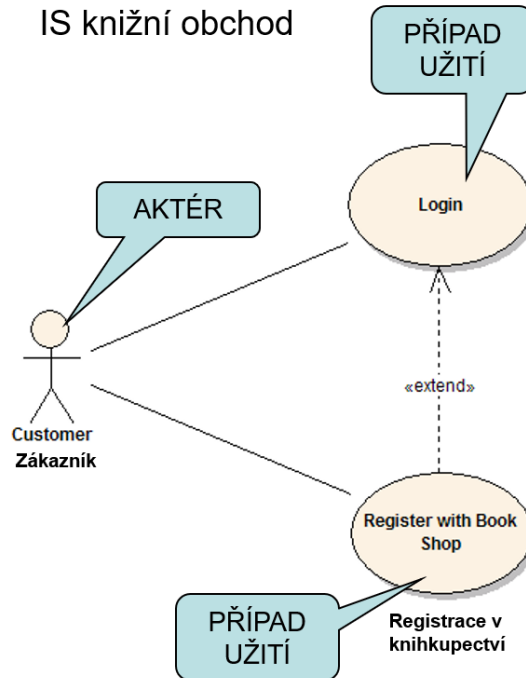


Diagram případů užití

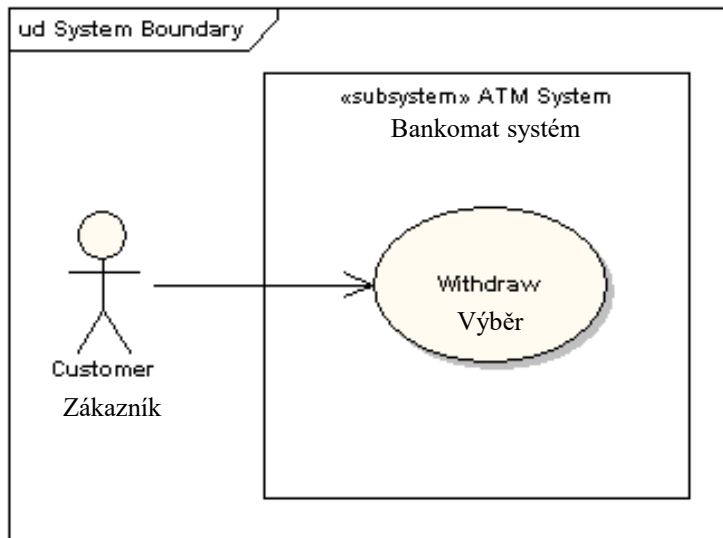
- **Jméno a popis**
  - Příklad užití je normálně pojmenován slovesnou frází a opatřen krátkým neformálním textovým popisem.
- **Požadavky**
  - Požadavky definují formální funkční požadavky, které musí případ užití poskytovat koncovému uživateli. Funkční požadavek definuje a popisuje posloupnost činností, které systém vykonává prostřednictvím aktivace uživatelem systému.

(Pozn. USE CASE musí zohledňovat i nefunkční požadavky, ty vyjadřují požadavky na vlastnost systému jako je kvalita, bezpečnost, odezva, dodržování norem a předpisů, atd.)
- **Omezení**
  - Omezení (constraint) je podmínka, za níž případ užití operuje. To zahrnuje před- post- a invariantní podmínky.
- **Scénáře**
  - Scénář je formální popis toku událostí, které se vyskytují během provádění případu užití. Definuje specifickou posloupnost událostí odehrávajících se mezi systémem a externími aktéry. Je běžně popsán textem o odpovídá jazykovému vyjádření diagramu sekvencí.

# Hranice systému (System Boundary)



Je obvyklé zobrazovat případy užití uvnitř nějakého systému a aktéry vně tohoto systému.



Příklad Bankovní IS

# Diagram případů užití



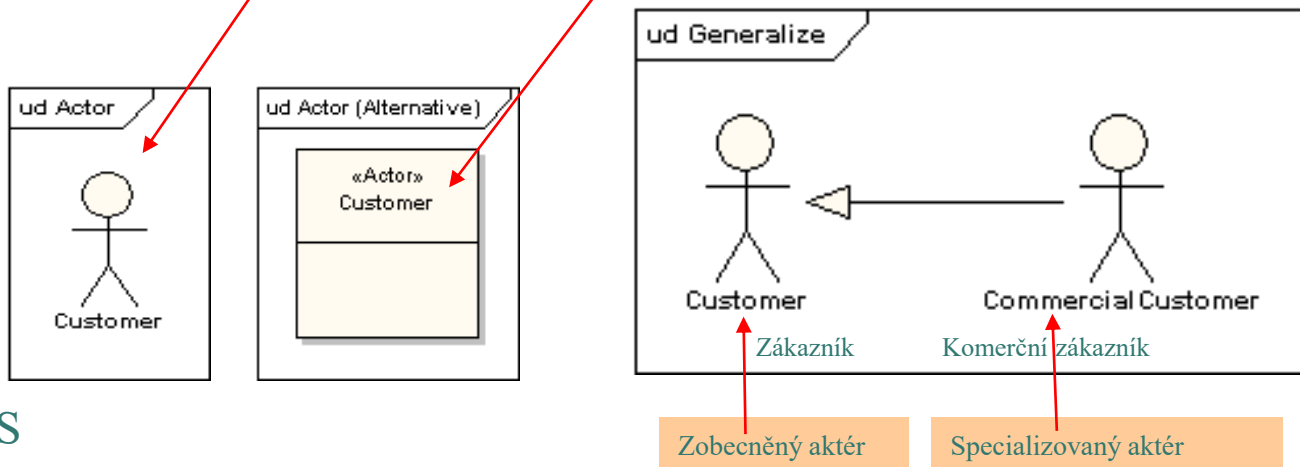
## Aktéři, participanti (účastníci)

Případ užití ukazuje interakci mezi systémem a entitami externími vzhledem k systému. Tyto externí entity se nazývají aktéry nebo participanty.

Aktéry představují role které mohou zahrnovat lidi, externí hardware nebo další systémy.

Aktér je obvykle nakreslen jako pojmenovaná schematická postava, nebo alternativně jako obdélník reprezentující třídu s klíčovým slovem «actor» (Aktér).

Aktéři mohou zobecňovat jiné aktéry



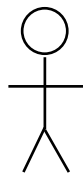
Bankovní IS

# Značky diagramu užití symboly

---



SLEZSKÁ  
UNIVERZITA  
OBCHODNĚ PODNIKATELSKÁ  
FAKULTA V KARVINĚ



*Aktér (Actor) – uživatel*



*Případ užití*

# Značky diagramu užití popis symbolů

---



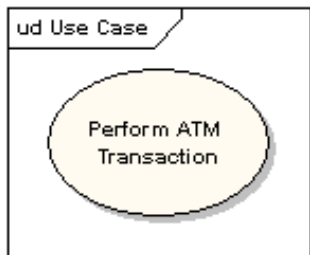
*Aktér* je část okolí systému, které komunikuje s vytvářeným systémem. Může to být člověk, ale také jiný systém předávající si informace s námi vytvářeným systémem. Pomocí aktéra určujeme hranice navrhovaného systému. Ne vždy je možné postihnout všechny uživatele systému, proto je třeba mít na paměti proč, a za jakým účelem daný systém navrhujeme, komu má sloužit.

*Případ užití* je logicky uzavřený popis komunikace mezi aktérem a vytvářeným systémem. Případ použití určuje funkce modelu, tj. pomocí jejich popisu se definují funkční požadavky na vytvářený systém. Je to posloupnost událostí iniciovaná aktérem a specifikuje vztah mezi uživatelem a systémem. Slouží vývojářům systému jako údaj o požadavcích na systém z hlediska uživatelů.

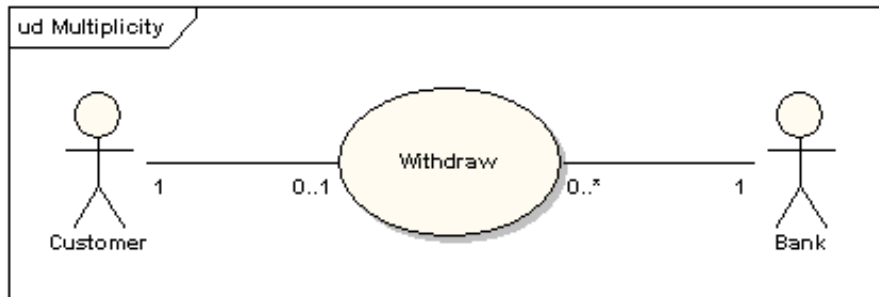
# Případy užití - násobnosti



Případ užití představuje v rámci systému jednotku smysluplné činnosti. Poskytuje z vyšší úrovně pohled na chování pozorovatelné někomu nebo něčemu z vnějšku systému. Značka pro zakreslení případu užití je elipsa.



Symbol pro použití případu užití je propojovací linka s volitelnou šipkou na konci znázorňující směr řízení. Spojnice znázorňující použití může mít dle potřeby hodnoty násobnosti na každém konci, které ukazují, že uživatel může mít v daném okamžiku pouze jednu akci výběru avšak banka může mít větší počet uživatelů, kteří provádějí své výběry souběžně.



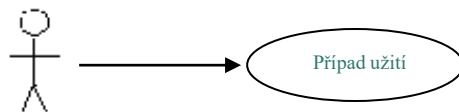
## Bankovní IS (vysvětlivky)

Customer	zákazník
Withdraw	výběr
Multiplicity	násobnost
Perform ATM Transaction	Provedení transakce v bankomatu

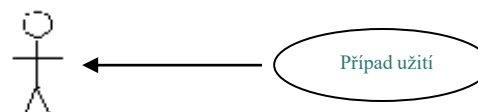
# Elementární situace v diagramu užití



- Scénář – podrobný rozpis komunikace aktéra se systémem. Je to posloupnost impulsů actorů a reakcí systému.
- Impuls – je komunikace ve směru od aktéra k systému (požadavek aktéra na systém nebo odpověď aktéra na požadavek systému).
- Reakce – je odpověď systému aktérovi nebo požadavek na aktéra.



*Impuls*



*Reakce*



# Znázornění vztahů mezi případy užití



- Vkládání – umožňuje kroky definované jedním případem užití použít ve druhém případě užití.
- Rozšiřování – dává prostředky k vytvoření nového případu užití přidáním kroků k případu užití, který již existuje. Rozšíření je možné provést jen na určitých místech v sekvenci kroků základního případu užití. Tato místa se nazývají *body rozšíření*.

<<vložit>>  
-----➔

*Vkládání (input)*

<<rozšiřuje>>  
-----➔

*Rozšiřování (extend)*

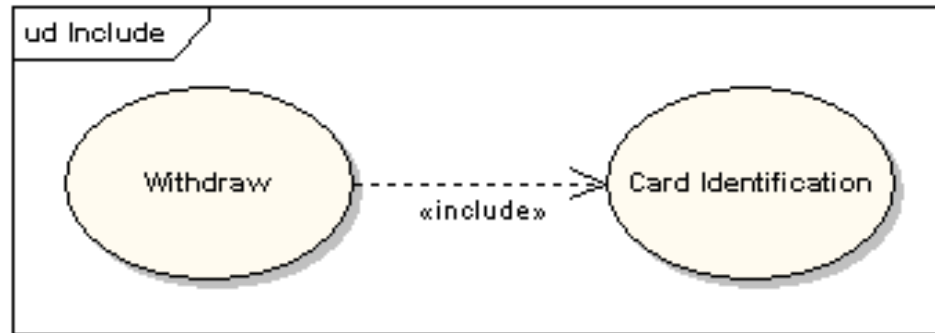
# Vložený případ užití (Including Use CASE)



Případy užití mohou obsahovat funkcionalitu jiného případu užití jako část svého normálního zpracování.

Obecně se předpokládá, že libovolný vnořený (<<include>>) případ užití bude zavolán pokaždé, když poběží základní případ užití.

Jako příklad tohoto přístupu je provedení případu užití <Card Identification> jako součásti případu užití <Withdraw> .

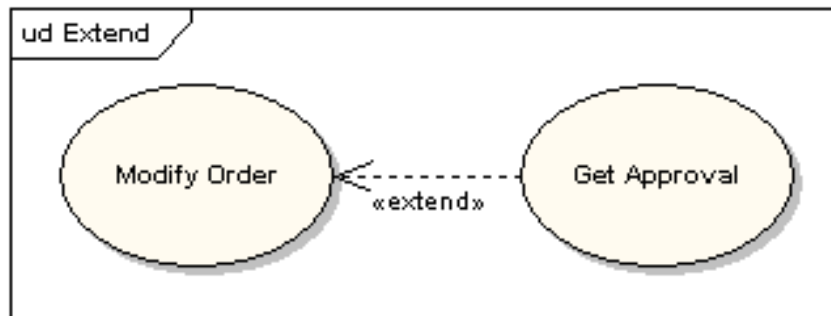


Bankovní IS – USE CASE výběr z bankomatu zahrnuje USE CASE identifikaci karty

# Rozšiřující případy užití (Extending Use CASE)



Jeden případ užití může být použit pro rozšíření chování druhého; to se typicky užívá za výjimečných okolností. Například před modifikací určitého typu objednávka zákazníka, když uživatel musí dostat souhlas od nějaké vyšší autority, může se volitelně rozšířit normální případ užití <Modify Order> případem užití <Get Approval> (Získej souhlas)



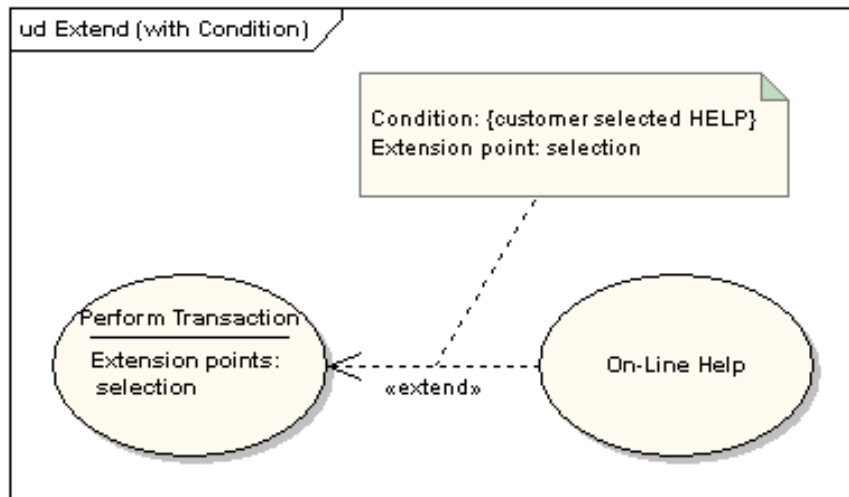
Modify Order	Upravit objednávku
Get Approval	Získej souhlas
Extend	Rozšířit, Rozšíření

IS Objednávka zákazníka IS - Při úpravě objednávky může nemusí být požadován souhlas s úpravou objednávky

# Body rozšíření - Extension Points



Bod, kam je rozšiřující případ užití přidán, je možno definovat prostřednictvím bodu rozšíření (extension point).



Bankovní IS (Internet Banking žádost o pomoc při zadávání transakce)

Extend with condition  
Perform Transaction  
Extension points  
Selection  
Customer selected  
Help

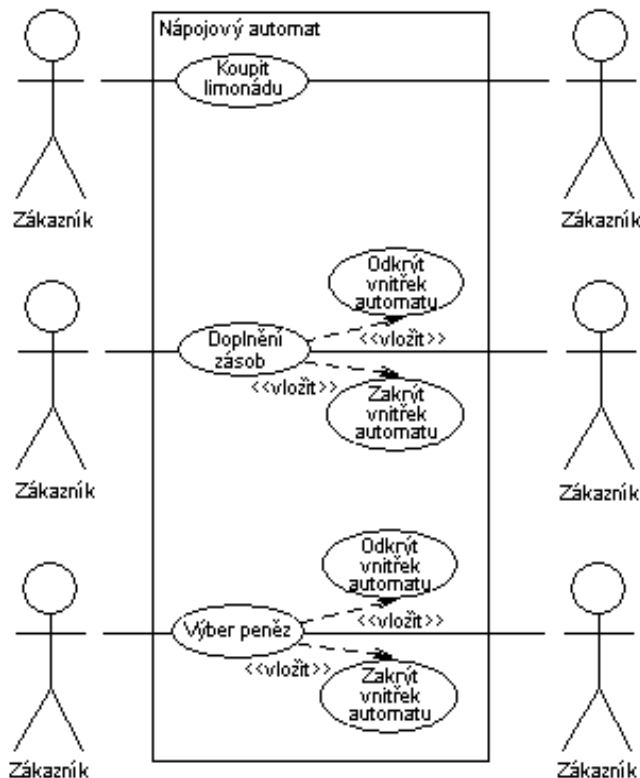
Rozšíření s podmínkou  
Vykonání transakce  
Bod rozšíření  
Výběr možnosti  
Zákazník vybírá možnost  
HELP - Pomoc

# Příklad vkládání mezi případy užití



## Nápojový automat:

- Příklad se zaměřuje na případy *Doplnění zásob* a *Výběr peněz*.
- Oba případy začínají tím, že se automat odemkne a otevře, a končí jeho uzavřením a uzamčením.
- První společnou fází lze popsat jako případ užití s názvem *Odkrýt vnitřek automatu* a závěrečnou fází jako *Zakrýt vnitřek automatu*.
- Tyto pak můžeme vložit do případů *Doplnění zásob* a *Výběr peněz*.

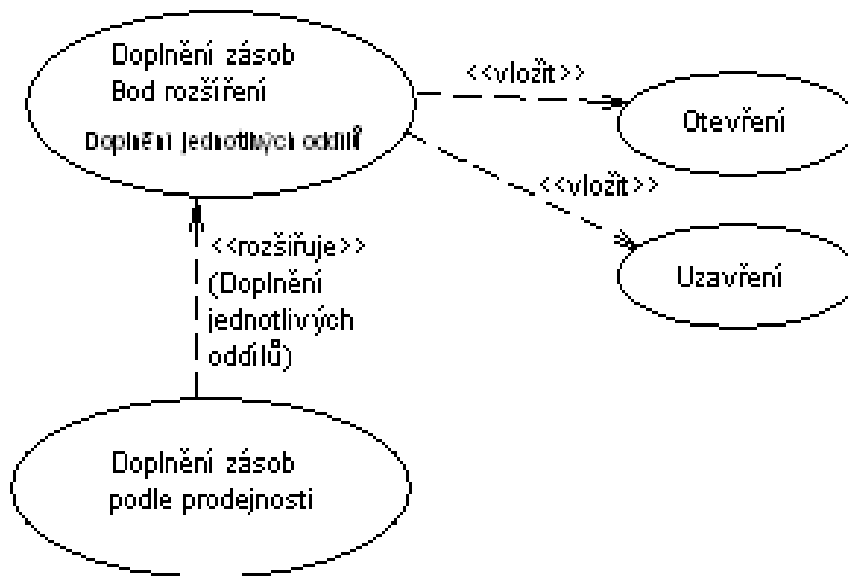


# Příklad rozšíření mezi případy užití

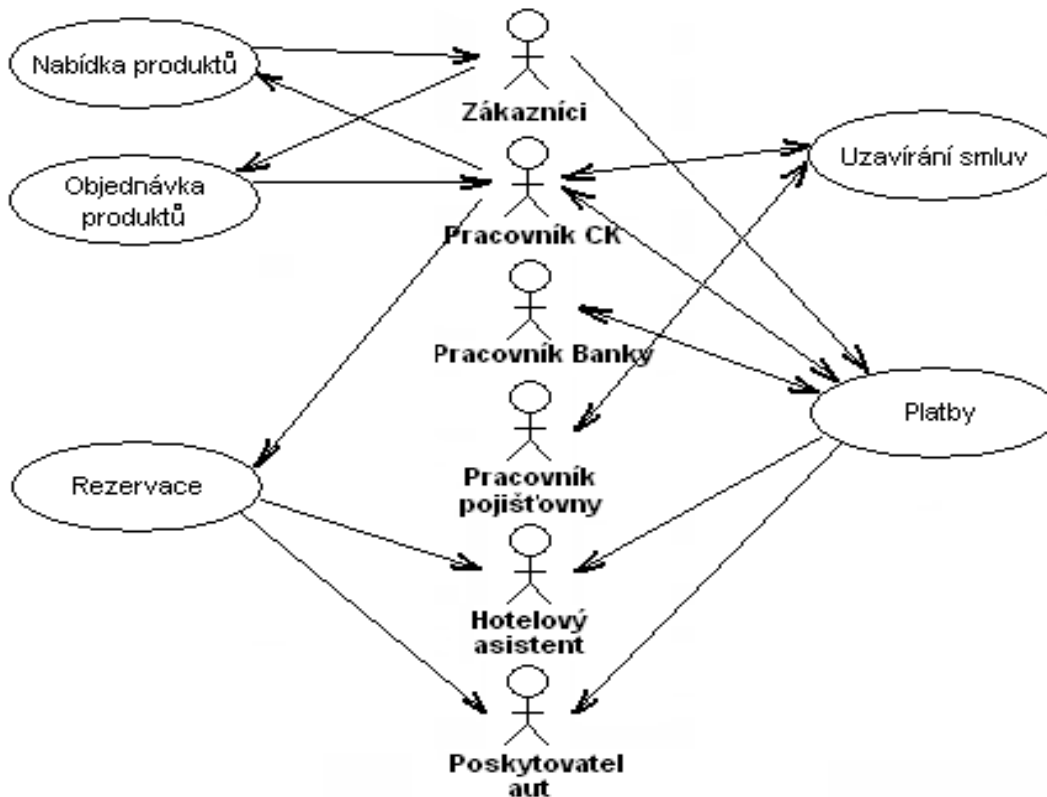


## Nápojový automat:

- Vycházejme z toho, že pro automat jsme již dříve vytvořili případ užití *Doplnění zásob*.
- Uvažme ale, že dodavatel místo aby doplnil všechny druhy limonád na max. úroveň, dodá těch které se prodávají dobře více, než ostatních.
- Tento případ se stává rozšířením původního případu *Doplnění zásob* na *Doplnění zásob dle prodejnosti*.
- Vznikl doplněním do původního případu užití.



# Příklad diagramu užití - informační systém CK



# Popis případu užití obvykle obsahuje:

---

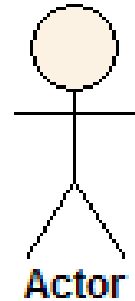
1. Obecné informace popisující případ užití.
  2. Požadavky – Věci které musí případ užití pro uživatele dělat, jako například <možnost aktualizovat objednávku>, <možnost změnit objednávku > atp.
  3. Omezení – pravidla určující, co lze, a co nelze dělat. To zahrnuje:
    - vstupní podmínky (pre-conditions) které musí platit předtím než se případ užití rozběhne, např. <create order> musí předcházet <modify order>
    - výstupní podmínky (post-conditions) které musí platit, jakmile případ užití proběhne, např. <order is modified and consistent>;
    - invarianty: tyto musí platit vždy – např. objednávka musí mít vždy číslo zákazníka
  4. Scénáře – Sekvenční popisy kroků, které se musí vykonat, aby byl případ užití proveden. Může být tvořen i více scénáři, aby se vyhovělo mimořádným okolnostem a alternativním postupům zpracování.
  5. Diagramy scénářů – Sekvenční diagramy zobrazují postup zpracování – podobně jako scénáře, avšak grafickým vyjádřením.
  6. Dodatečné atributy jako implementační fáze, číslo verze, ohodnocení složitosti, stereotyp a status.
-



# Participant, aktéři (actor)

---

- Aktér, participant, je uživatel systému.
- Zahrnuje jak živé uživatele, tak stroje.
- Aktér užívá případ užití aby vykonal určitou práci, která je užitečná pro systém.
- Množina případů užití k nimž má aktér přístup, definuje jeho celkovou roli v systému a rozsah jeho akcí.



Aktér, participant



## Požadavky

Požadavky jsou formální funkční požadavky, které případ užití musí poskytovat koncovému uživateli. Odpovídají funkční specifikaci nalezené při analýze resp. návrhu systému. Požadavek vyjadřuje, že případ užití bude vykonávat nějakou akci nebo dodávat nějakou hodnotu systému.

## Omezení.

Toto jsou formální pravidla a omezení, za nichž případ užití operuje a obsahují před (pre-) post- a invariantní podmínky. Předpodmínka specifikuje, co již muselo být splněno nebo připraveno předtím, než případ užití může začít. Post-podmínka dokumentuje, co bude platit v okamžiku dokončení případu užití. Invariant specifikuje, co musí platit v průběhu vykonávání případ užití.

## Scénáře.

Scénáře jsou formální popisy toků událostí, které mohou nastat během provedení případu užití. Jsou obvykle popsány textově a odpovídají jazykovému vyjádření sekvenčního diagramu. (viz dále)

# Vztahy obsahování a rozšiřování mezi případy užití

---



- Jeden případ užití může zahrnovat funkcionalitu jiného případu užití, jako část svého normálního zpracování. Obecně se předpokládá, že „obsažený“ případ užití bude vykonán pokaždé, když poběží jej „zahrnující“ případ užití. Příkladem může být „vypsat seznam všech objednávek zákazníků před modifikací vybrané objednávky“ – v takovém případě případ užití <list orders>, může být vykonán vždy, když bude v běhu případ užití <modify order>.
- Případ užití může být zahrnut v jednom nebo ve více případech užití, čímž pomáhá redukovat zdvojování funkcionality prostřednictvím faktorizace společného chování do případů užití, které jsou opakovaně vícekrát použity.
- Jeden případ užití může rozšířit chování druhého – typicky když se vyskytnou mimořádné okolnosti. Například, jestliže před modifikací určitého typu objednávky zákazníka musí uživatel dostat souhlas nějaké vyšší autority, pak případ užití <get approval> (dej souhlas), může dobrovolně rozšířit obvyklý případ užití <modify order> (uprav objednávku).

# Sekvenční diagramy

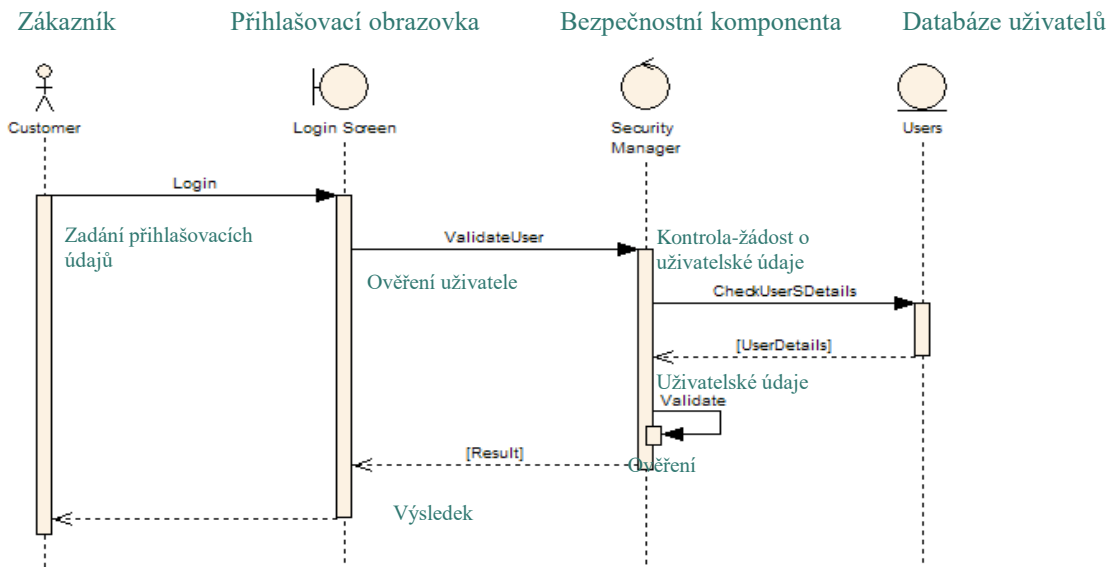


Zobrazení interakcí objektů v čase. Tyto typicky ukazují uživatele nebo aktéra, objekty a komponenty, se kterými interagují v průběhu provedení případu užití. Jeden sekvenční diagram typicky vyjadřuje samostatný scénář toků událostí nějakého případu užití.

## Přihlašování uživatele do systému:

Zákazník přes přihlašovací obrazovku zadá přihlašovací údaje, ty zpracuje bezpečnostní komponenta, která si „vytáhne“ údaje z databáze uživatelů, ověří přihlašovací údaje a výsledek (kladný nebo záporný) zobrazí na obrazovku pro zákazníka.

ČAS běží horizontálně





**Děkuji za pozornost.**

**Otázky?**

---