Objektové metody modelování

UML diagramy 2. část

Tutoriál III

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

RNDr. Zdeněk Franěk, Ph.D.

UML - Unified Modeling Language

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- UML je grafický jazyk pro vizualizaci,specifikaci, konstrukci a dokumentaci komponent softwarových systémů, ale nejen softwarových systémů
- UML nabízí standardní způsob jak psát návrhy systému, včetně konceptuálních věci jako jsou obchodní procesy a systémové funkce stejně tak jako příkazy pro programovací jazyky , databázová schémata, a znovupoužitelné softwarové komponenty.

UML definuje notaci pro následující oblasti

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- **Interakce uživatele** neboli Use Case Model , **model případů užití**. Popisuje rozhraní interakcí mezi systémem a uživateli. Jistým způsoben koresponduje s modelem požadavků (requirements model)
- **Model interakcí nebo komunikací** – popisuje jak objekty systému budou vzájemně komunikovat aby vykonaly požadovanou práci
- **Stavový model nebo dynamický model** (Dynamic Model) – Stavová schémata popisují stavy a podmínky, kterými třídy (prvky systému) projdou během času. Grafy činností popisují průběh práce, který systém bude implementovat
- **Logický model nebo model tříd** (Logical or Class Model) – popisuje třídy a objekty které vytvářejí systém
- **Fyzický model resp. komponentní model** (Component Model) – popisuje programové (a někdy i hardwarové komponenty) které tvoří systém
- **Fyzický model rozmístění** (Physical Deployment Model) – popisuje fyzickou (hardwarovou) architekturu a rozmístění komponent na této hardwarové architektuře..

Diagramy pro modelování struktury

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- **Diagramy balíčků** se používají k rozdělení modelů do logických kontejnerů nebo balíků ('packages') a popisují jejich interakce na vyšší úrovni (tj. na úrovni těch kontejnerů)
- **Diagramy tříd** nebo též **strukturální diagramy** definují základní stavební bloky modelu: typy, třídy a obecné součásti které jsou použity pro konstrukci úplného modelu
- **Diagramy objektů** ukazují jak vztahy mezi instancemi strukturálních elementů a jejich použití za běhu systému
- **Kompositní diagramy, Diagramy složených struktur** (Composite Structure diagrams) poskytují prostředky pro zobrazování vrstev struktur elementů a zaměřují se na vnitřní detaily, konstrukci a vztahy prvků a struktur
- **Diagramy komponent** se používají pro modelování struktur vyšší úrovně nebo složitějších struktur, obvykle tvořených jednou nebo více třídami a s dobře definovaným
- **Diagramy nasazení** (Deployment diagrams) ukazují fyzické rozmístění významných prvků v reálném prostředí

Diagramy pro modelování chování

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- **Diagramy případů užití** (Use Case diagrams) se používají k modelování interakce uživatele se systémem. Definují definují chování, požadavky a omezení formou scénářů a skriptů
- **Diagramy aktivit** (Activity diagrams) mají široké pole použití počínaje definováním základního toku zpracování až po zachycení rozhodovacích bodů a akcí uvnitř jakékoliv obecného procesu
- **Diagramy stavových přechodů (**State Machine diagrams) jsou podstatné pro porozumění podmínkám přechodu z jednoho stavu systému do jiného stavu které charakterizují stav modelu za jeho běhu
- **Komunikační diagramy** (Communication diagrams) znázorňují síť a posloupnost zpráv nebo komunikací mezí objekty za běhu při jejich spolupráci
- **Sekvenční diagramy** (Sequence diagrams) úzce souvisejí s komunikačními diagramy a zobrazují posloupnost zpráv předávaných mezi objekty s použitím vertikálních časových os
- **Časovací diagramy** (Timing diagrams) spojují sekvenční a stavové diagramy aby poskytly názornější pohled na stavy objektů v čase a na zprávy které způsobují změnu jejich stavu
- **Diagramy přehledu interakcí** (Interaction Overview diagrams) spojují diagramy činností a sekvenční diagramy aby umožnily interakčním fragmentům (interaction occurrence) se lépe propojit s rozhodovacími bloky a toky zpracování

Diagram užití – *use case diagram*

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- 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 (aktor).

Model případů užití

Use Case – Případ užití

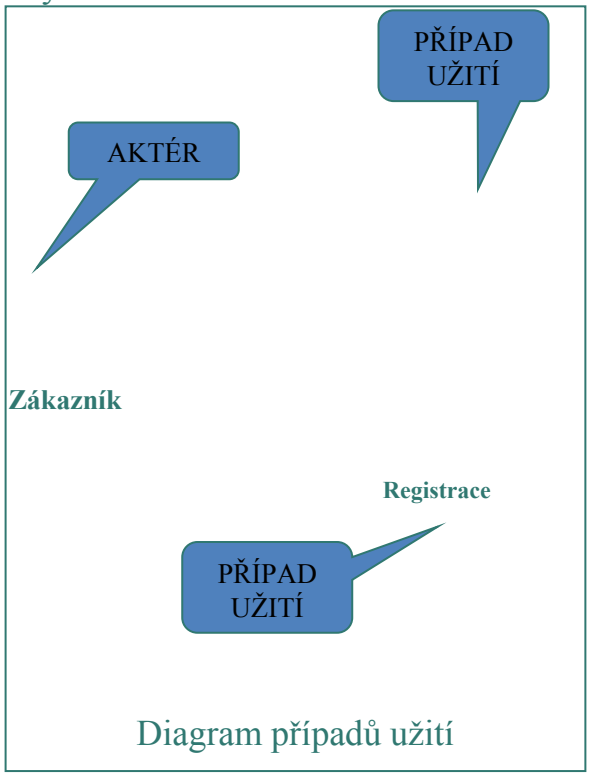
• popisuje navrhovanou funkcionalitu nového systému.

• vyjadřuje oddělenou jednotku interakce mezi uživatelem (člověk nebo stroj) a systémem

• je je samostatná jednotka smysluplné činnosti;

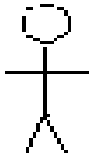
- logovat se do systému,
- registrace v systému
- vytvoření objednávky

jsou všechno případy užití.



Značky diagramu užití

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ



Aktor – uživatel

Případ užití

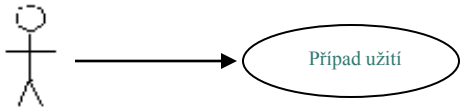
Značky diagramu užití

Aktor 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 aktora 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 aktorem a vytvářeným systémem. Případ použití určuje funkce modelu, tj. pomocí jejich popisu se definují funkční požadavky na vytvářený systém. Je to posloupnost událostí iniciovaná aktorem a specifikuje vztah mezi uživatelem a systémem. Slouží vývojářům systému jako údaj o požadavcích na systém z hlediska uživatelů.

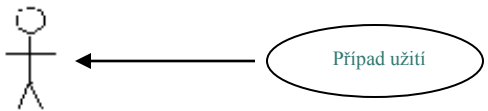
Elementární situace v diagramu užití

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

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



Impuls

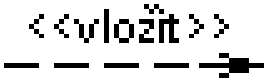


Reakce

Znázornění vztahů mezi případy užití

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- *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*.

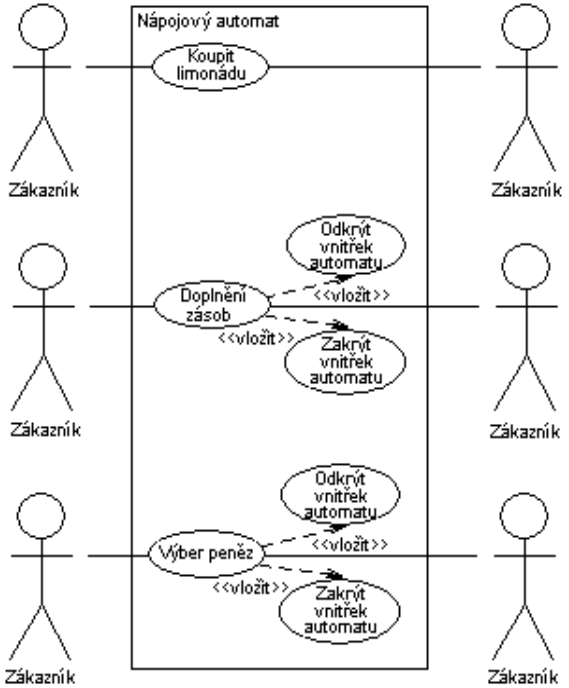


Vkládání

Rozšiřování

Příklad vkládání mezi případy užití

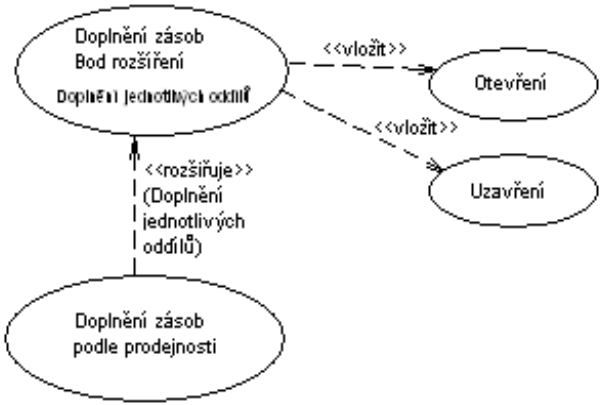
- *Nápojový automat:* Příklad se zaměřuje na případy *Doplnění zásob* a *Výběr peněz*. Oba případy začínají tím, že se automat odemkne a otevře, a končí jeho uzavřením a uzamčením. První společnou fázi lze popsat jako případ užití s názvem *Odkrýt vnitřek automatu* a závěrečnou fázi jako *Zakrýt vnitřek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



Příklad rozšíření mezi případy užití

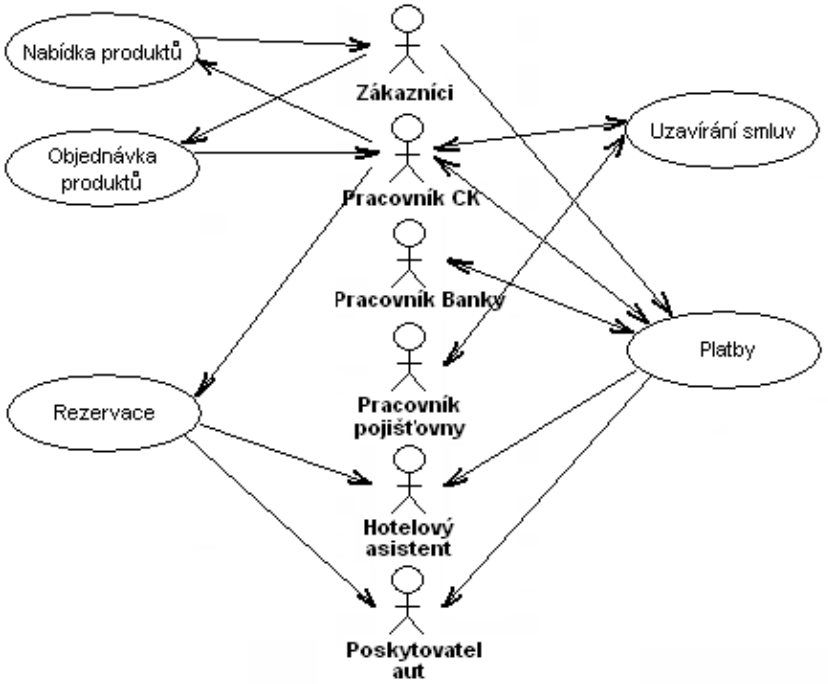
SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- *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

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ



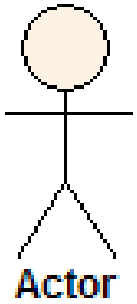
Popis případu užití obvykle obsahuje

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 mí t 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 (4) avšak vyjádřeno graficky
6. Dodatečné atributy jako implementační fáze, číslo verze, ohodnocení složitosti,stereotyp a status

Participanti, aktéry (actor)

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

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



Aktér, participant

Omezení, Požadavky a Scénáře

- **Požadavky**. Požadavky jsou formální funkční požadavky, které případ užití musí poskytovat koncovému uživatelovi. 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 vykonávaní 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)

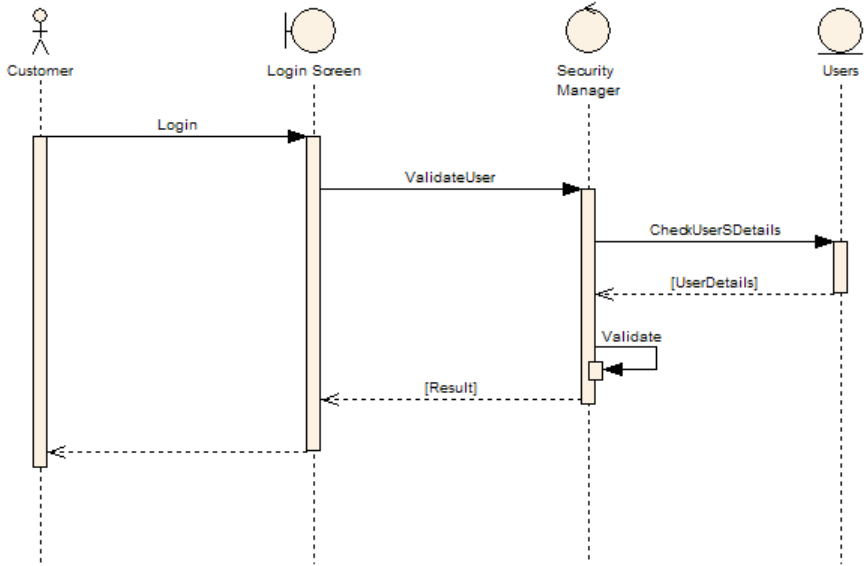
Vztahy obsahování a rozšiřování mezi případy užití

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- 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).

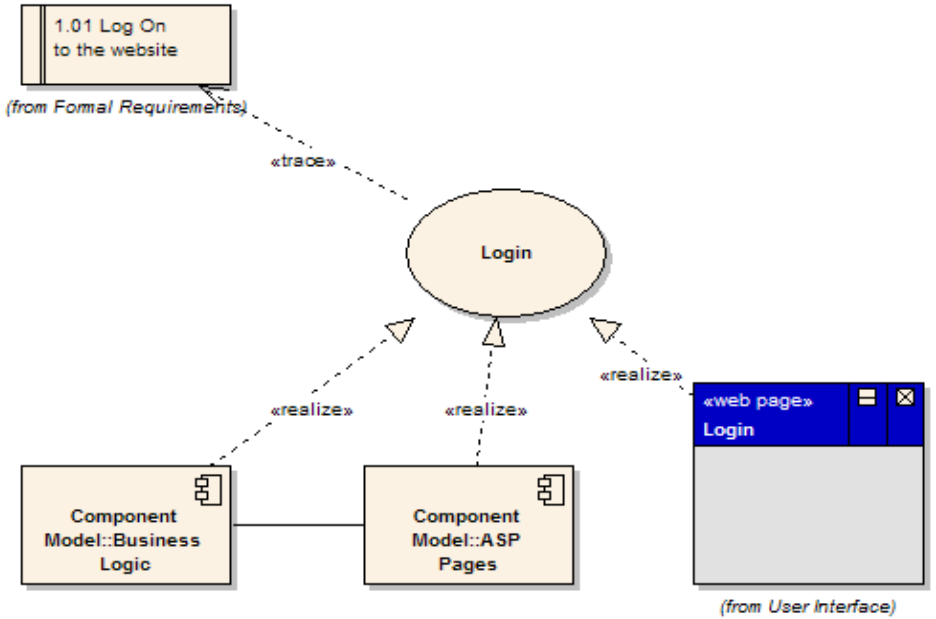
Sekvenční diagramy

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



Implementační diagramy

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



Dynamický Model

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

Vyjádření a modelování chování systému v čase. Zahrnuje podporu pro

- diagramy činností,
- stavové diagramy,
- diagramy následností a
- rozšíření včetně modelování podnikových procesů (business process modelling).

Diagramy činností

Diagram činností se používají k vyjádření dynamického chování modelu. Zobrazuje aktivity prováděné lidskými nebo systémovými aktéry a přechody mezi aktivitami včetně podmínek řídících přechody k různým aktivitám. Model též může zachytit synchronizační body v nichž se může více aktivit zahájit paralelní zpracování.

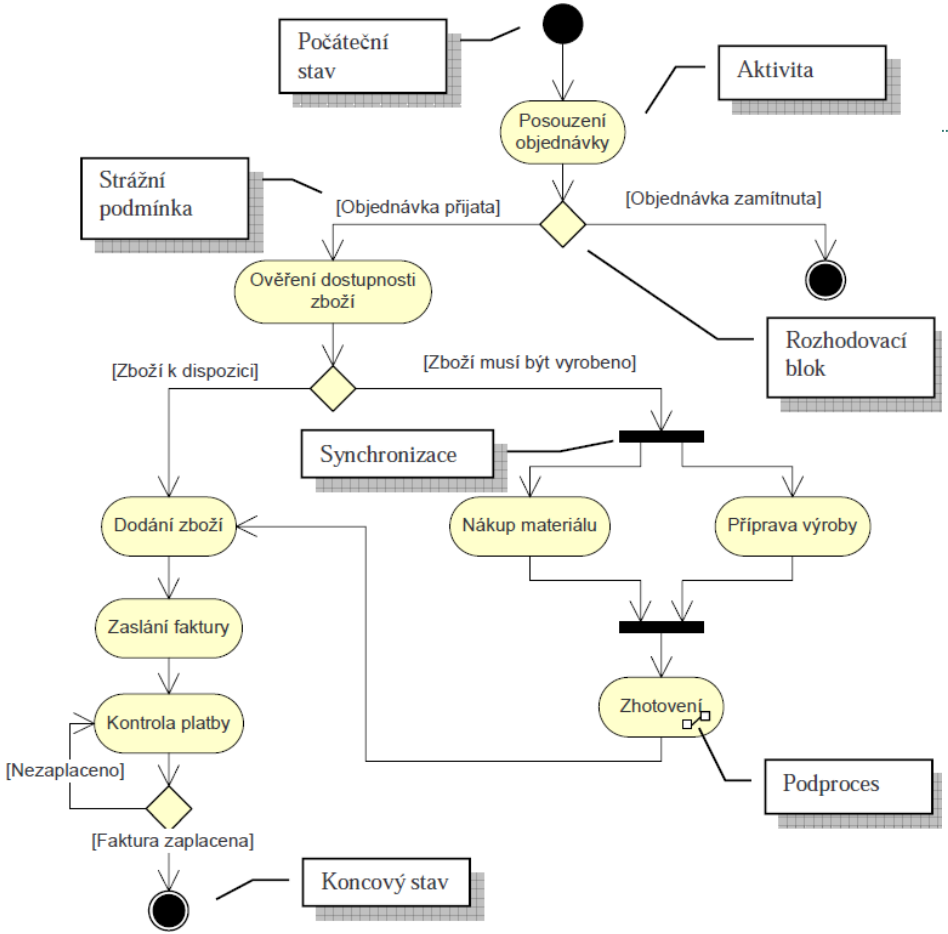
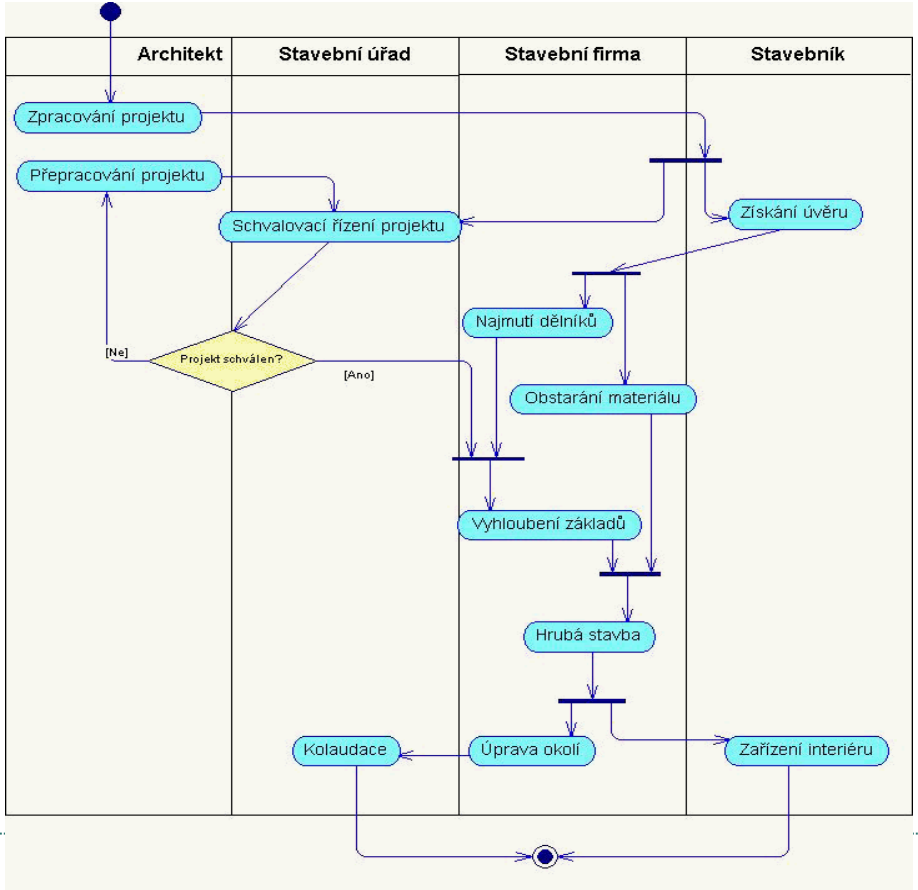


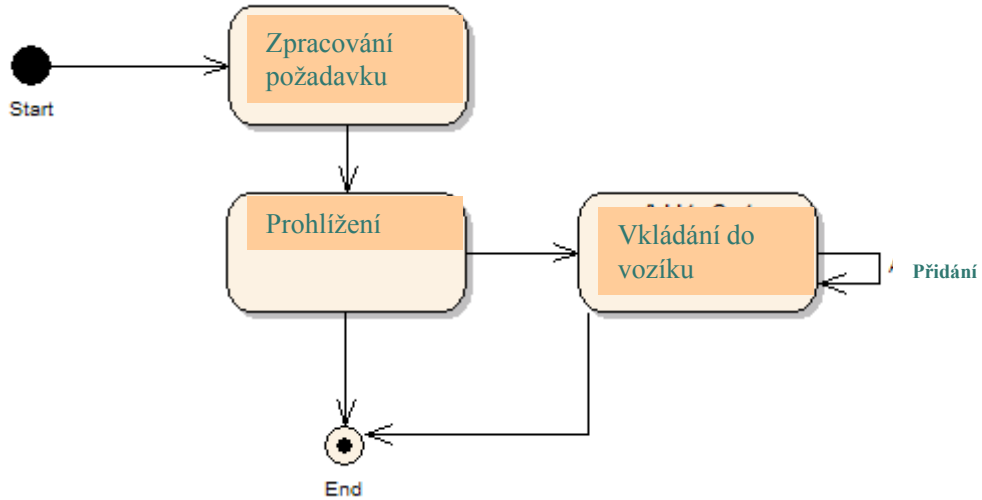
Diagram činností a plavecké dráhy

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ



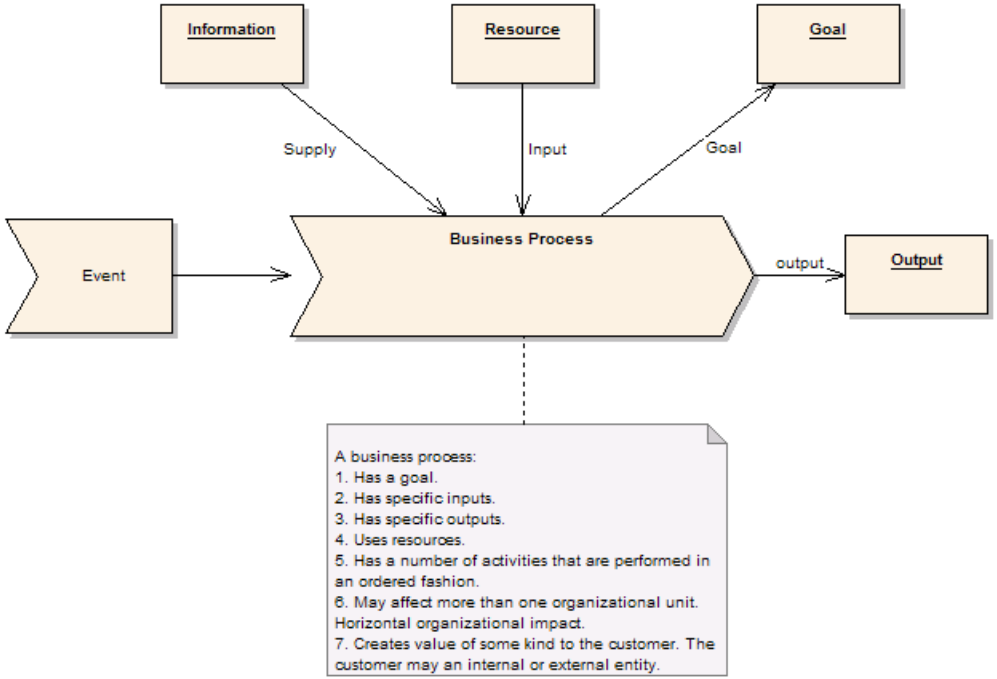
Stavové diagramy

- Stavové diagramy se používají pro znázornění detailů přechodů nebo změn stavu objektu, ke kterým může v systému dojít. Ukazují, jak objekt přechází z jednoho stavu do druhého a pravidla, kterými se řídí tyto změny. Stavové diagramy mají obvykle podmínku pro start a pro konec



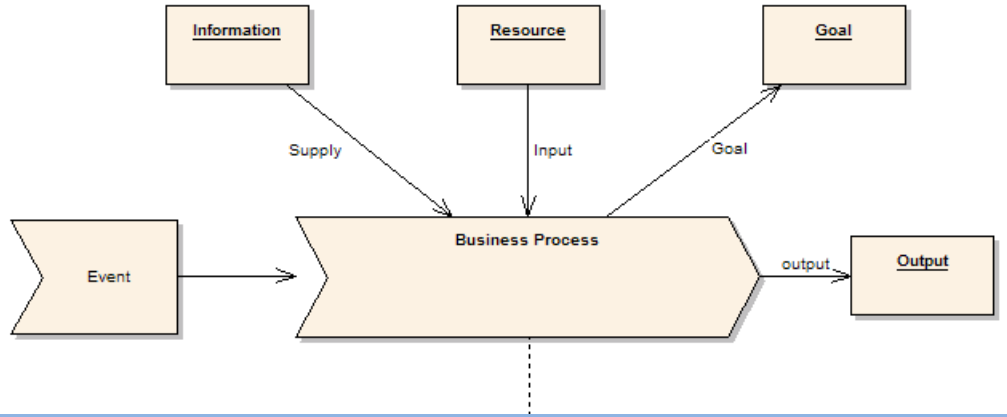
Model procesu

Model procesu je rozšíření diagramu činností užívaný pro modelování podnikového (obchodního) procesu. Tento diagram ukazuje, které cíle proces má, vstupy, výstupy , události a informace, které jsou zapojeny do procesu.



Model procesu

Model procesu je rozšíření diagramu činností užívaný pro modelování podnikového (obchodního) procesu. Tento diagram ukazuje, které cíle proces má, vstupy, výstupy , události a informace, které jsou zapojeny do procesu.



Podnikový proces:
1. Má cíl
2. Má specifické vstupy
3. Má specifické výstupy
4. Užívá zdroje
5. Skládá se z řady aktivcit které se vykonávají v určitém pořadí
6. Může ovlivnit více než jednu organizační jednotku
7. Přináší zákazníkovi nějakou hodnotu. Zákazník může být interní nebo externí entitou

Logický Model

- Logický model je statický pohled na objekty a třídy, které jsou vytvořeny při analýze a návrhu a které tvoří tzv. prostor návrhu a analýzy. Typicky Doménový model (Domain Model) je abstraktnější pohled na podnikové objekty („Bussiness“ objekty) a entity, zatímco model tříd (Class Model) je rigoróznější, přesnější a na návrh soustředěný model

Model tříd

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- Třída je standardní konstrukce UML používaná pro definování vzorů z nichž během běhu systému (software) vznikají objekty

- Třída je specifikace - objekt je instance třídy

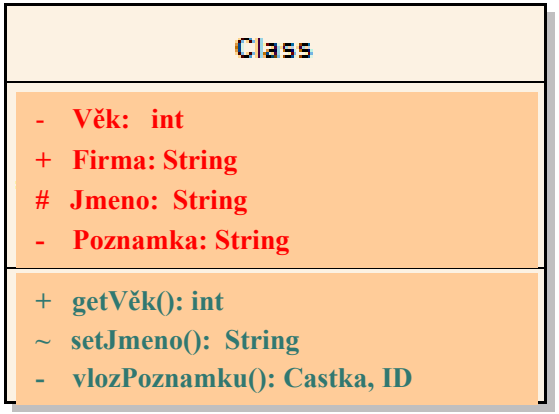
- Třídy mohou v systému
  - vznikat
  - být děděny z jiných tříd (to je, dědí veškeré chování a stav svých rodičů a k tomu přidávají svoji vlastní novou funkcionalitu)
  - mohou mít jiné třídy jako atributy
  - delegovat zodpovědnosti jiným třídám
  - a implementovat abstraktní rozhraní.

Model tříd

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- Model tříd je jádrem objektově-orientovaného vývoje a návrhu.
- Vyjadřuje jak persistentní (setrvalý) stav systému, tak chování systému.
- Třída zapouzdřuje stav (atributy) a nabízí služby (metody=procedury) pro změnu tohoto stavu (chování).
- Dobrý objektově orientovaný návrh omezuje přímý přístup k atributům třídy a nabízí služby, které manipulují s atributy třídy namísto klienta.
- Toto skrývaní dat a poskytování služeb garantuje, že aktualizace dat se děje na jednom místě a podle definovaných pravidel – u velkých systémů je údržba kódu, který má přímý přístup k datům na mnoha místech velkou zátěží.

Zobrazení třídy



třída má tři rozdílné oblasti

1. Jméno třídy (a stereotypu je-li použito)

2. Oblast atributů třídy (vnitřní datové prvky)

3. Oblast chování – jak soukromé (private) tak veřejné (public)

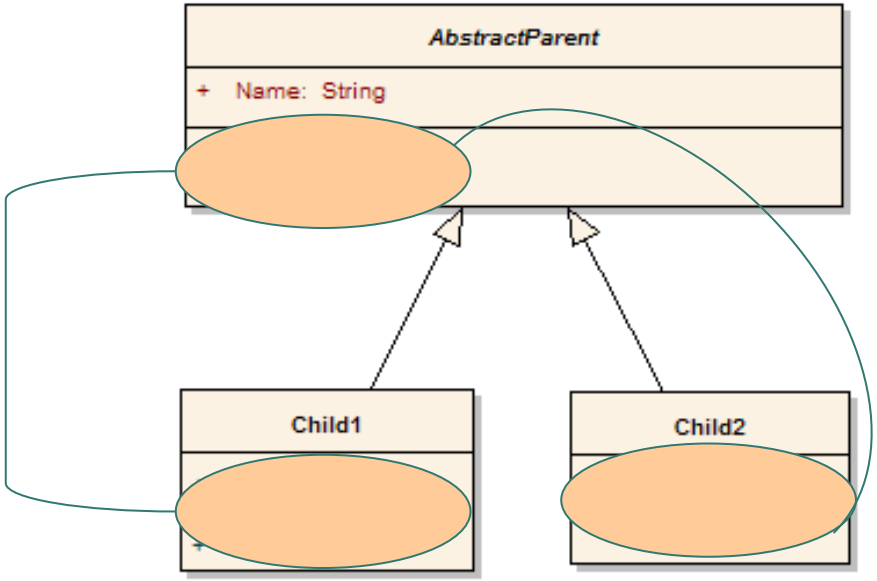
Atributy a metody mohou být poznačeny jako

- Soukromé (private), nejsou viditelné volajícím mimo třídu
- Chráněné (protected), jsou viditelné pouze dědicům třídy
- Veřejné (public), jsou viditelné všem

Dědění tříd

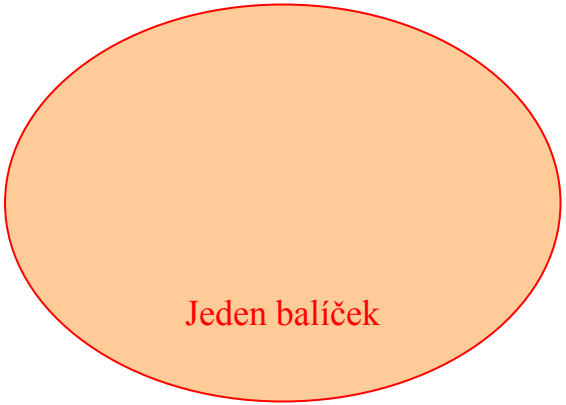
SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

abstraktní třída, je otcem dvou synovských tříd, z nichž každá dědí vlastnosti základní (rodičovské) třídy a rozšiřuje je o svoje vlastní chování.



Model tříd

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ



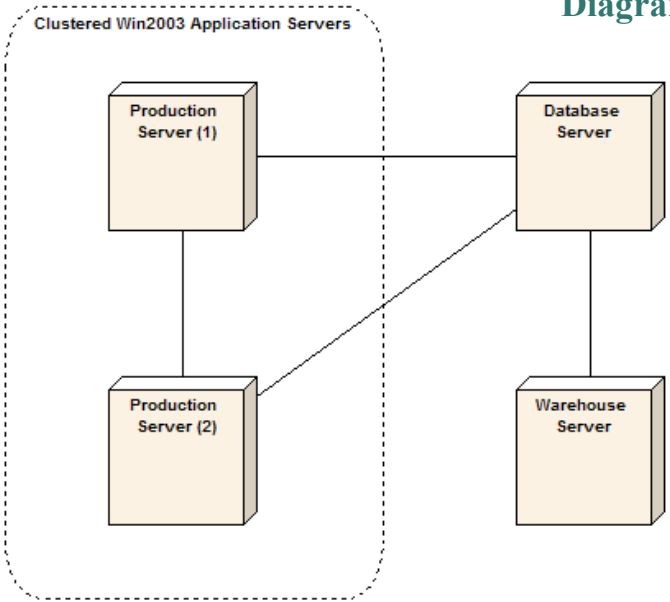
Model tříd může být soustředěn do balíčků s příbuzným chováním a stavem

Fyzický Model

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

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

Diagram nasazení (Deployment View)



Model komponent

- Model komponent ilustruje softwarové komponenty, které budou využity pro vytvoření systému.
- Ty mohou být sestrojeny z modelu tříd a napsány od základu jako nové pro tento 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

Zakreslení komponent

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

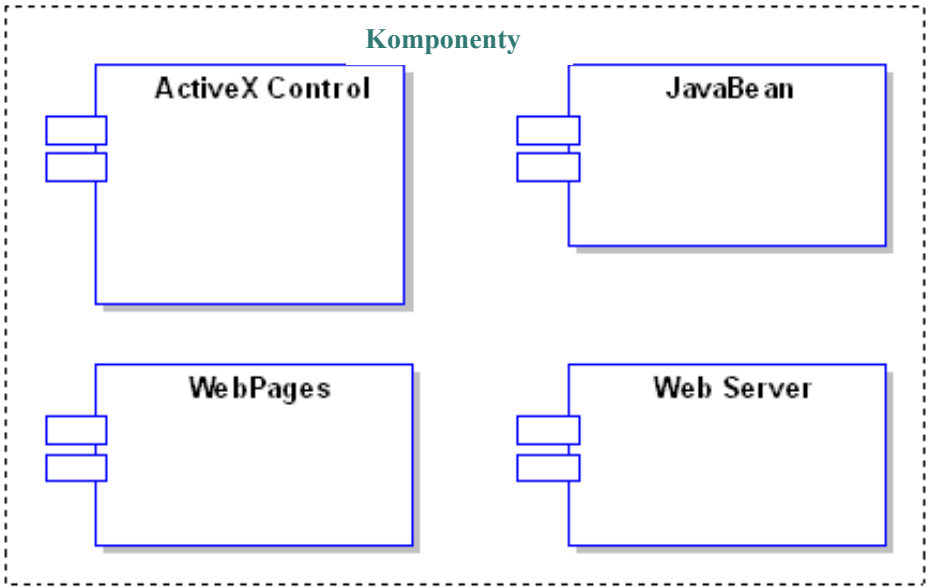


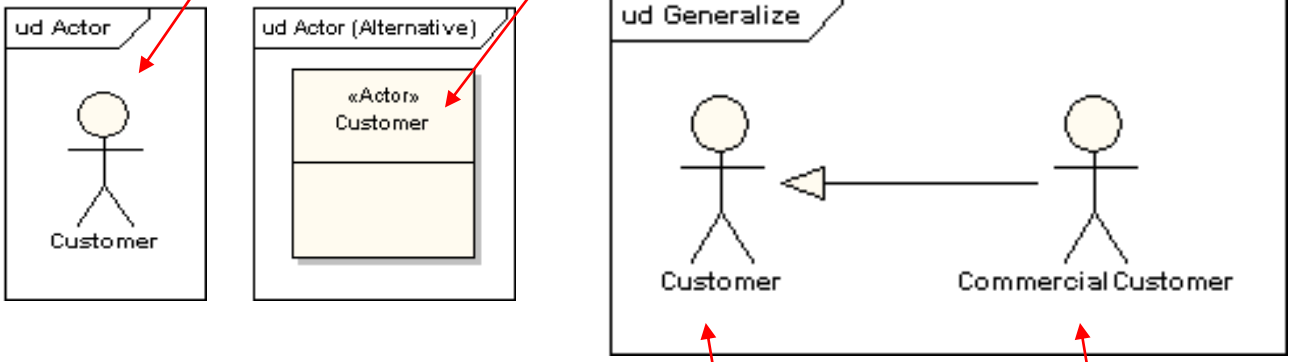
Diagram případů užití

Aktéři, participanti

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

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

Aktér je obyčejně nakreslen jako pojmenovaná schematická postava, nebo alternativně jako obdélník reprezentující třídu s klíčovým slovem **«actor»**



Aktéry mohou zobecňovat jiné aktéry

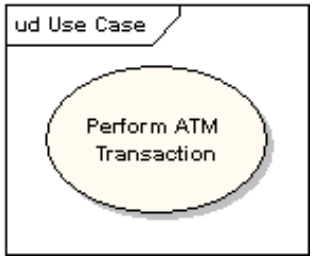
Zobecněný aktér

Specializovaný aktér

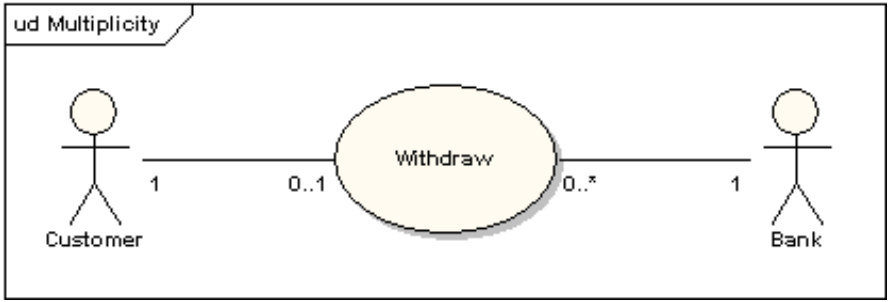
Případy užití

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

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. Konektor 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.



Definice případu užití

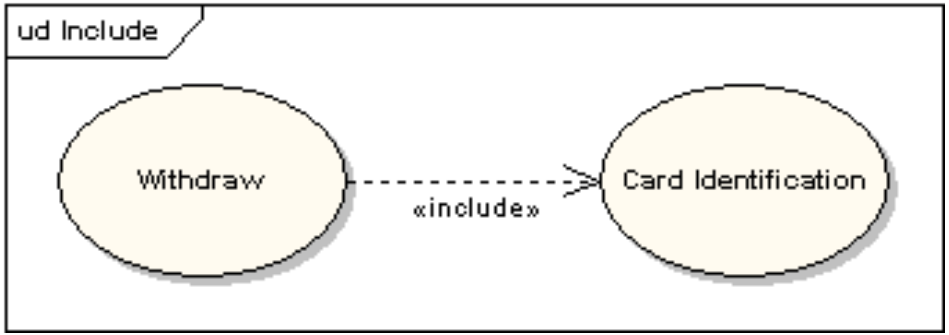
SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

- **Jméno a popis**
  - Případ 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. Požadavek je slib že, případ užití bude provádět akci nebo poskytovat nějakou hodnotu systému.
- **Omezení**
  - Omezení (constraint) je podmínka, za nichž případ užití operuje. To zahrnuje před- post- a invariantní podmínky.
- **Scénáře**
  - A Scénář je formální popis toku událostí, které se vyskytují během provádění případu užití. Definuje specifickou posloupnost událostí odehrávajících se mezi systémem a extérními aktéry. Je běžně popsán textem o odpovídá jazykovému vyjádření diagramu sekvencí

Vložený případ užití (Including Use CASE)

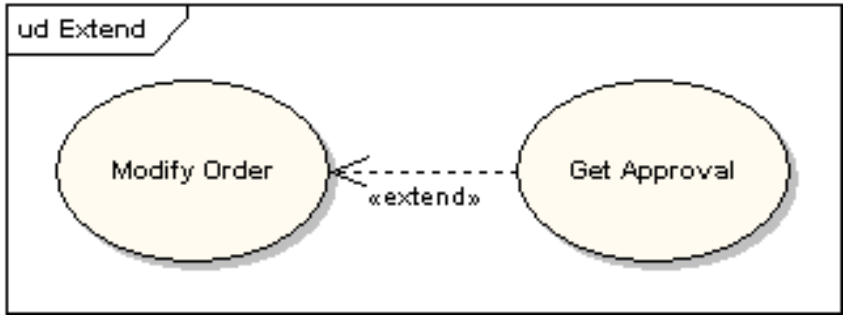
SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

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, kdy poběží základní případ užití. Jako příklad tohoto přístupu je provedení případu užití <Card Identification> jako součásti případu užití <Withdraw>



Rozšiřující případy užití (Extending Use CASE)

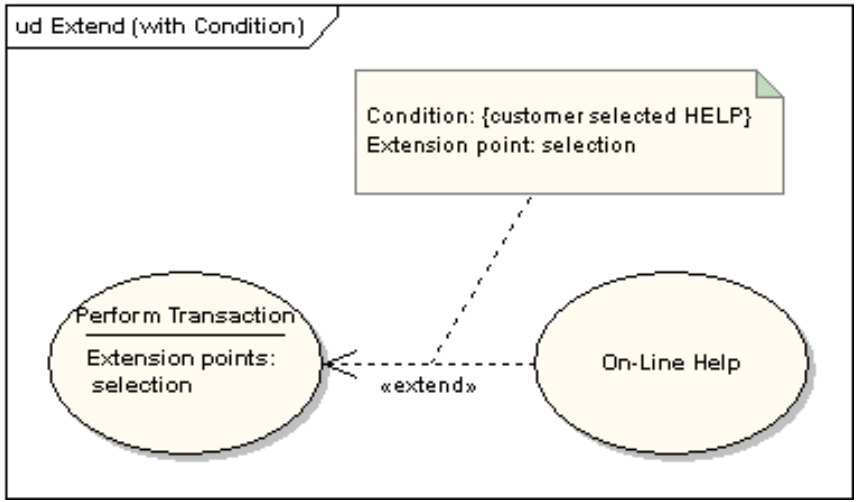
Jeden případ užití může být použit pro rozšíření chování druhého; to se typicky užívá za výjimečných okolností. Například před modifikací určitého typu objednávka zákazníka, když uživatel musí dostat souhlas od nějaké vyšší autority, může se volitelně rozšířit normální případ užití <Modify Order> případem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



Body rozšíření - Extension Points

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

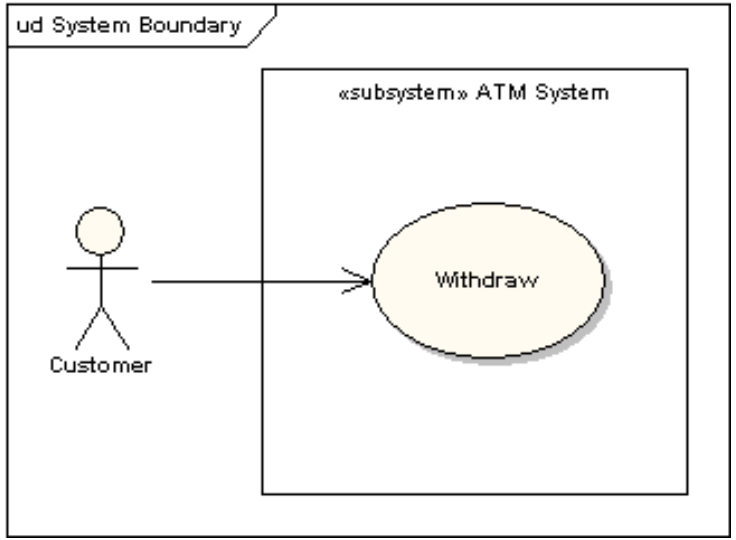
Bod kam je rozšiřující případ užití přidán je možno definovat prostřednictvím nějakého bodu rozšíření (extension point)



Hranice systému (System Boundary)

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

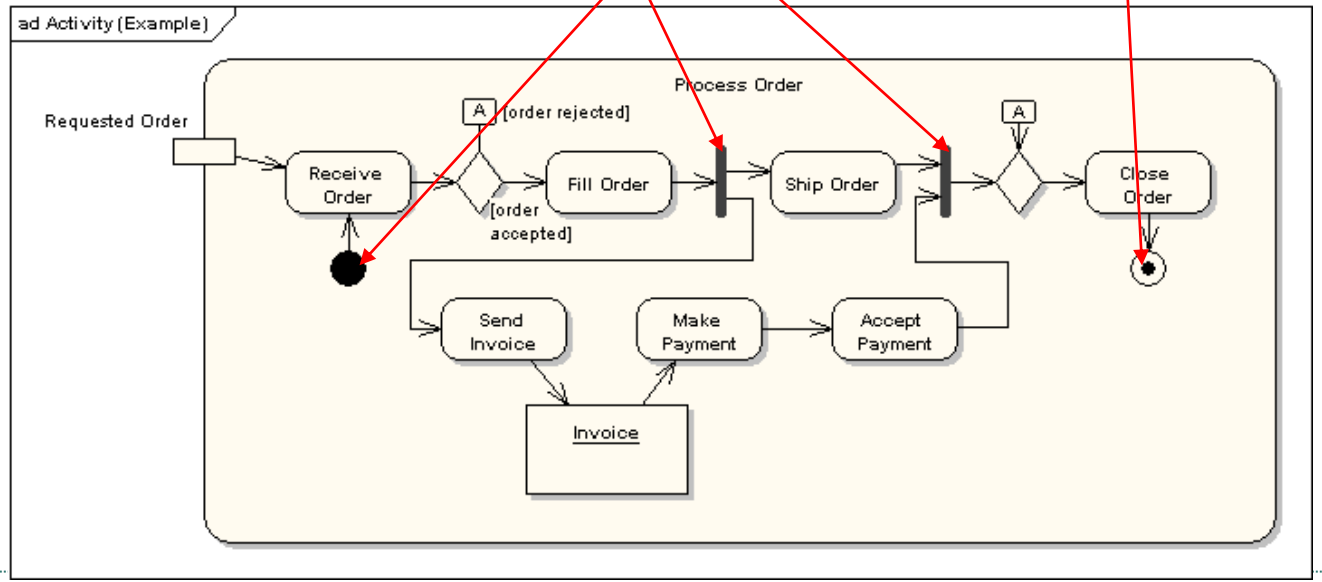
Je obvyklé zobrazovat případy užití uvnitř nějakého systému a aktéry vně tohoto systému.



Diagramy aktivit

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

Diagramy aktivit ukazují tok akcí (workflow) od startovního bodu ke koncovému bodu, přičemž zpodrobňují řadu rozhodovacích cest, které existují při postupu mezi jednotlivými událostmi vznikajících při provádění činnosti. Mohou být použity pro detailní popis situací, v nichž může dojít k paralelnímu zpracování, které se může vyskytnout při zpracování některých aktivit.

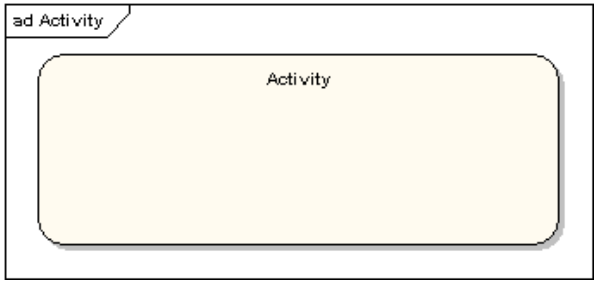


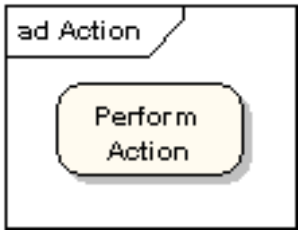
Aktivita, akce a jejich omezení

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

Aktivita je specifikace parametrizovatelných posloupností chování

Je znázorněna jako zaoblený obdélník zapouzdřující všechny činnostní prvky, kterými je tvořena aktivita





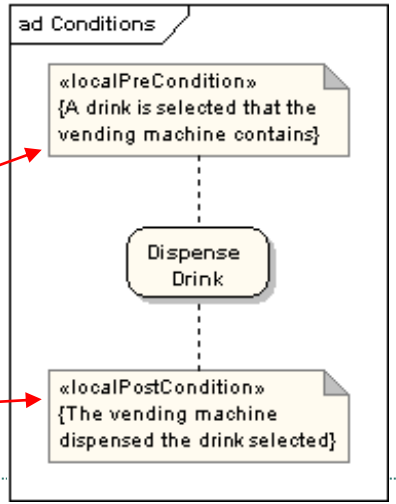
Akce představuje jeden krok uvnitř aktivity.

Akce jsou znázorněny zaoblenými obdélníky

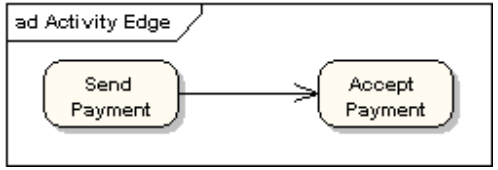
K akci může být připojeno omezení. Následující diagram ukazuje akci s lokálními

pre- a

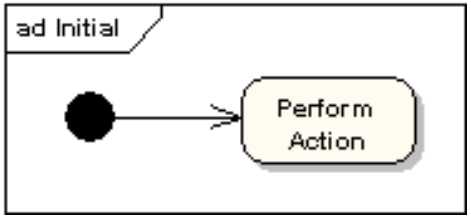
post-podmínkami



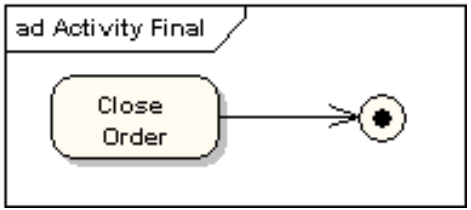
Řídicí tok,počáteční uzly,koncové aktivity a toky



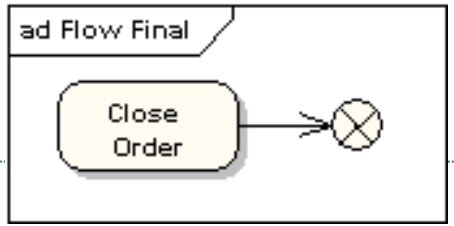
Řídicí tok ukazuje chod řízení od jedné akce k následující.
Je označen čarou s šipkou.



Počáteční nebo startovní uzel je zobrazen velkým černým bodem,



Uzel koncová aktivita je znázorněn jako kroužek s tečkou uvnitř. čarou s šipkou. uzel koncový tok označuje konec jednoho řídicího toku

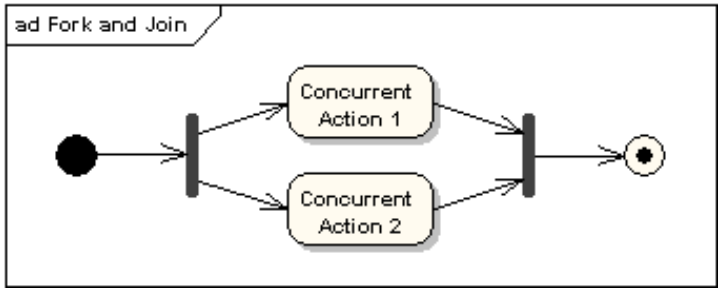


Uzel koncový tok je znázorněn jako kroužek s křížkem uvnitř. Uzel koncová aktivita označuje konec všech řídicích toků uvnitř aktivity.

Uzly paralelního rozvětvení (Fork) a spojení (Join)

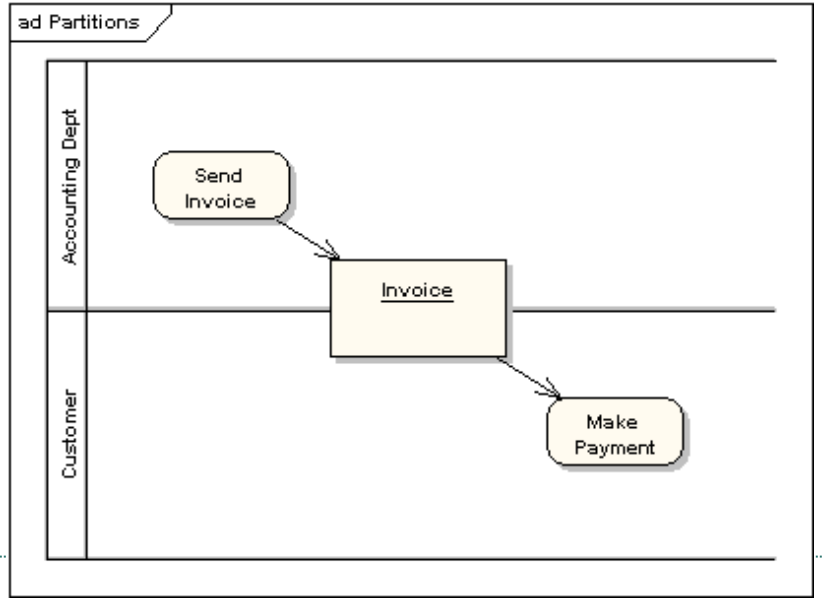
SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

Uzly rozvětvení a spojení jsou označeny shodně: buď horizontální nebo vertikální trámek (orientace je dána orientací vstupujících nebo vystupujících toků). Označují začátek a konec souběžných (paralelních) cest řízení.



Oddíly (Partition)

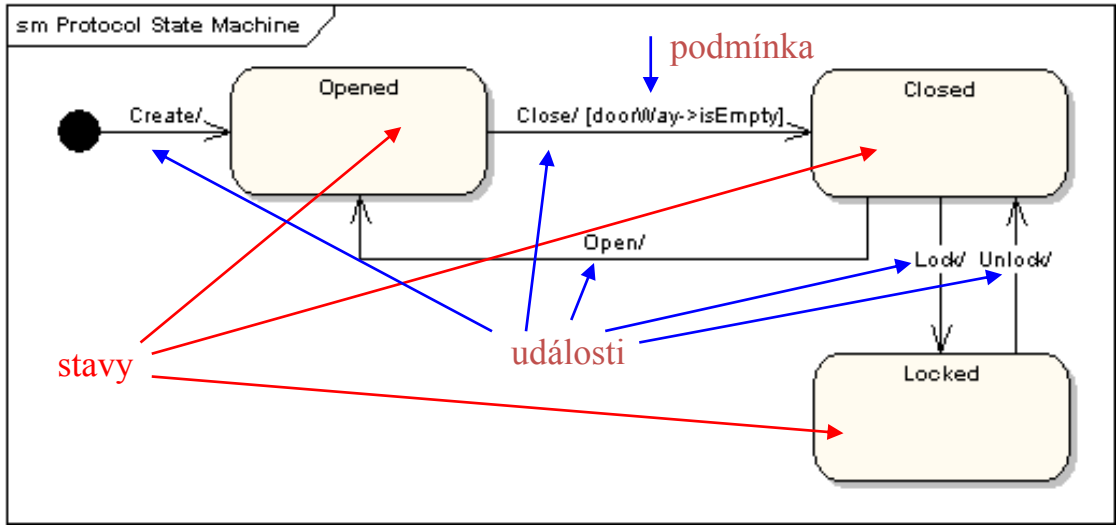
Oddíl aktivity je znázorněn jako horizontální nebo vertikální dráha (swimlane). V následujícím diagramu jsou oddíly použity pro oddělení akcí uvnitř nějaké aktivity are na ty které jsou provedeny účtárnou (Accounting Dept.) a na ty které jsou provedeny zákazníkem (Customer).



Diagramy stavových přechodů

SLEZSKÁ UNIVERZITA
OBCHODNĚ PODNIKATELSKÁ FAKULTA V KARVINÉ

Diagram stavových přechodů modeluje chování jednoho objektu tím, že specifikuje množinu stavů, kterými tento objekt prochází během svého života a množinu událostí, které způsobují přechod objektu z jednoho stavu do druhého. Následující diagram stavových přechodů ukazuje stavy kterými během svého života procházejí dveře



Děkuji za pozornost

Otázky?