

Root.cz

Root.cz > Vývoj > Programovací jazyky > **Nástroje pro tvorbu UML diagramů**

Pavel Tišnovský - 4. 7. 2005 0:00

NÁSTROJE PRO TVORBU UML DIAGRAMŮ



Vývoj

V tomto článku budou popsány aplikace a pomocné utility, které lze použít pro tvorbu UML diagramů s případnou návazností na generování zdrojového kódu v různých programovacích jazycích. UML diagramy jsou v současné době využívány zejména pro návrh rozsáhlých informačních systémů založených na objektově orientovaném návrhu nebo pro podrobnou specifikaci procesní analýzy.

Obsah

1. Stručný popis jazyka UML [viz: #k01]
2. Princip a základní elementy jazyka UML [viz: #k02]
3. Předměty [viz: #k03]
4. Relace [viz: #k04]
5. Diagramy [viz: #k05]
6. Komerční produkty pro práci s UML [viz: #k06]
 - 6.1 Rational Rose [viz: #k061]
 - 6.2 ARTiSAN Real-time Studio [viz: #k062]
 - 6.3 I-Logix Rhapsody [viz: #k063]
 - 6.4 MetaMatrix MetaBase Modeler [viz: #k064]
 - 6.5 Together Designer Community Edition [viz: #k065]
7. Produkty šířené pod GPL a podobnými licencemi [viz: #k07]
 - 7.1 Violet [viz: #k071]
 - 7.2 Umbrello UML Modeller [viz: #k072]
 - 7.3 ArgoUML [viz: #k073]
8. Závěr [viz: #k08]

1. Stručný popis jazyka UML

Jazyk označovaný zkratkou **UML** neboli **Unified Modeling Language** je, jak již název napovídá, unifikovaný modelovací jazyk, který má, na rozdíl od převážně textově orientovaných programovacích jazyků, vlastní grafickou syntaxi (tj. pravidla pro sestavování jednotlivých elementů jazyka do větších objektů) a sémantiku (tj. jednoznačná pravidla určující jednotlivým syntaktickým výrazům jejich význam).

V současné době má jazyk UML největší význam při návrhu softwarových systémů, protože objektově orientovaný návrh každé složitější aplikace je nezbytným předpokladem pro její úspěšnou a rychlou implementaci. Pro objektově orientovaný návrh je samozřejmě možné použít různé podpůrné prostředky, zejména další odlišné typy diagramů, UML je však významné také v tom ohledu, že přesně specifikuje, co má daný diagram obsahovat, což je velmi důležité zejména při sdílení informací mezi jednotlivými analytiky

a vývojáři. Dále je již z principu UML nutné, aby vytvářené grafy měly vnitřní konzistenci a přesně danou sémantiku, což u jiných typů grafů nemusí být obecně zaručeno. UML diagramů existuje několik typů lišících podle toho, jaké se pomocí nich plánují či zpracovávají úlohy. Tyto diagramy se od sebe odlišují především repertoárem použitých značek, způsobem jejich vzájemného propojení a s nimi související sémantikou.

Mezi velké přednosti jazyka UML i na něm postavených UML diagramů patří existence otevřeného a rozšiřitelného standardu, podpora celého vývojového cyklu aplikace či jiného (ne nutně programového) systému a velká podpora pro různé aplikační oblasti. Pro širší využití jazyka UML v praxi mluví také významný fakt, že je podporován celou řadou vývojových nástrojů, ať už samostatných aplikací určených pouze pro práci s UML, nebo i integrovaných vývojových prostředí (IDE), které v některých případech dovolují provádět převod informací mezi UML diagramem a algoritmem zapsaným v programovacím jazyce (a samozřejmě i opačným směrem, ten je však z pochopitelných důvodů složitější a ne vždy uskutečnitelný).

Před vlastním popisem aplikací určených pro tvorbu UML diagramů si ozřejmíme základní principy, na kterých je celý UML založen. Podrobnější popis jazyka UML je uveden například na stránce interval.cz/clanek.asp?article=2783 [viz: <http://interval.cz/clanek.asp?article=2783>] , na kterou navazují i další díly poměrně rozsáhlého seriálu o jazyce UML a zejména o jeho implementaci i nasazení při řešení reálných problémů.

2. Princip a základní elementy jazyka UML

Celý jazyk UML je založený na třech elementech, které ale nejsou z uživatelského hlediska reprezentovány v textové podobě, ale grafickými značkami v plošném (tj. dvourozměrném) grafu. Tyto tři základní elementy jazyka UML se dle své funkce nazývají:

1. Předměty
2. Relace
3. Diagramy

V dalších kapitolách bude význam jednotlivých elementů grafického jazyka UML popsán podrobněji. Ve třetí kapitole [viz: #k03] si popíšeme význam předmětů, v kapitole čtvrté [viz: #k04] význam relací a konečně v kapitole páté [viz: #k05] význam celých diagramů jako objektů na nejvyšší hierarchii UML.

3. Předměty

Předměty jsou elementy zpracovávaného modelu, jež jsou následně členěny do několika navzájem rozdílných podkategorií. Prvním typem podkategorií jsou takzvané **strukturní abstrakce** UML modelu, například programové třídy, aplikační či objektové rozhraní, případy užití, komponenty či uzly. V diagramu UML jsou strukturní abstrakce zobrazeny jako různé převážně plošné tvary, které jsou však vždy uzavřené. Například se jedná o obdélníky, elipsy, kružnice či jednoduché zdánlivě trojrozměrné tvary, například krychle či kvádry. Každý smysluplný UML diagram by měl obsahovat alespoň dvě strukturní abstrakce, při jedné abstrakci totiž modelování ztrácí svůj hlavní smysl - popis vztahů mezi jednotlivými objekty.

Kromě strukturních abstrakcí patří mezi předměty i takzvaná **chování**, která v UML diagramu prezentují interakce, tj. vzájemné komunikace mezi jednotlivými objekty. Pomocí chování lze také modelovat stavový stroj, u něž se stavy specifikují pomocí přechodů, událostí a aktivit. Chování se v UML diagramu většinou vyznačuje pomocí různým způsobem konstruovaných a různě zakončených šípek či propojovacích čar.

Mezi předměty patří i takzvané **seskupení**, které podle potřeby modelu graficky seskupuje části diagramu na nižší úrovni. Většinou se jedná o takzvané balíčky, jež mají tvar stylizované kancelářské složky s popisem

umístěným v levé horní části obdélníku (zobrazení seskupení se však může v různých prostředích odlišovat). Seskupení nejsou v současné době v některých dále popisovaných aplikacích použitelná (tj. nelze vytvářet hierarchické diagramy), což je pro tvorbu rozsáhlejších grafů velké omezení.

Posledním a současně velmi jednoduchým typem předmětů jsou **poznámky**, které blíže specifikují vlastnosti a chování dalších elementů UML diagramu. Graficky jsou poznámky na prakticky všech typech UML diagramů většinou vyvedeny žlutě vyplněným obdélníkem s ohnutým rožkem.

4. Relace

Vzhledem k tomu, že v grafech je zapotřebí předměty různým způsobem navzájem propojovat, jsou v jazyku UML specifikovány i **relace**, tj. vztahy mezi různými předměty. V UML jsou rozeznávány následující podtypy relací (z čistě jazykového hlediska patří relace mezi základní elementy UML):

1. **Asociace** - pomocí asociací se modeluje obecná souvislost předmětů, která je však v diagramu UML přesným způsobem definovaná. Speciální variantou asociace jsou takzvané **kompozice** a **agregace**, které jsou často používány v objektově orientovaných jazycích a návrzích databází.
2. **Závislost** - použije se, pokud změna v jednom předmětu způsobí změnu v předmětu jiném, nebo mu známým způsobem poskytne požadovanou informaci.
3. **Generalizace** - pomocí generalizace se modeluje stav, kdy je jeden předmět specializací jiného předmětu. Tato relace je velmi často používána v objektově orientovaných jazycích, implementuje se většinou pomocí dědičnosti (inheritance).
4. **Realizace** - jedná se o druh vztahu, ve kterém jeden předmět představuje dohodu, za jejíž splnění je odpovědný jiný předmět. V objektově orientovaných jazycích se realizace vytváří pomocí rozhraní (interface) - samozřejmě za předpokladu, že daný OOP jazyk rozhraní podporuje.

5. Diagramy

Diagramy zachycují různé aspekty modelovaného systému, který nemusí být obecně vyjádřen pouze jedním UML diagramem, protože je možné například pomocí balíčků (packages) sdružovat více diagramů do jednoho hierarchicky organizovaného celku. Různých typů diagramů existuje velké množství, například **diagram případů použití**, **diagram tříd**, **diagram objektů**, **diagram komponent**, **diagram nasazení**, **diagram aktivit**, **stavový diagram**, **diagram spolupráce** a **sekvenční diagram**. Každý z výše vyjmenovaných typů diagramů obsahuje poněkud odlišný repertoár dostupných grafických značek (předmětů a relací).

6. Komerční produkty pro práci s UML

V této kapitole budou stručně uvedeny charakteristiky a popisy komerčních produktů, které lze použít pro práci s jazykem UML, speciálně s UML diagramy. Většinou se jedná o velmi rozsáhlé profesionální produkty, pro jejichž úspěšné a efektivní používání je zapotřebí absolvovat školení či poměrně dlouhé studium. Výsledkem nasazení těchto systémů však bývá rapidní urychlení práce, zejména prvotního návrhu systému.

6.1 Rational Rose

Pravděpodobně nejznámějším a také nejpropracovanějším produktem, který pracuje s jazykem UML, je systém **Rational Rose**, který je v současné době vyvíjen pod záštitou firmy IBM Corporation (dříve se jednalo o samostatnou firmu). Tento komerční produkt je dostupný na adrese www-306.ibm.com/software/

rational [viz: <http://www-306.ibm.com/software/rational/>] . Systém je možné provozovat buď jako samostatnou aplikaci, nebo jako integrální součást různých vývojových prostředí. V současné době patří mezi podporované programovací jazyky C, C++ a Java, k dispozici je též technologie, která provádí automatický převod UML diagramů do tříd a rozhraní výše zmíněných programovacích jazyků (v programovacím jazyku C, který není už z principu objektový, se používá poněkud jiný přístup). Největší a patrně také jedinou vážnou nevýhodou tohoto produktu je jeho vysoká cena - musíme si totiž uvědomit, že se jedná o systém, který je ve firmách nasazován plošně.

6.2 ARTiSAN Real-time Studio

Komerční systém **ARTiSAN Real-time Studio** nabízí uživatelům podporu pro modelování a vývoj aplikací, zejména se zaměřením na systémy běžící v reálném čase. Mezi významné vlastnosti tohoto rozsáhlého aplikačního balíku patří podpora týmové práce, rozšíření jazyka UML o složky umožňující definovat real-time procesy a technologie pro získání UML diagramu ze stávajících zdrojových kódů aplikací. V současnosti jsou podporovány programovací jazyky C (neobjektové rozhraní), C++ a Java. K dispozici je také filtr, který z UML diagramu vytvoří programový stavový stroj s popisem jednotlivých stavů a závislostí mezi těmito stavy. Vzhledem k podpoře týmové práce se může tento systém použít i místo utilit typu CVS apod. Jeho nevýhodou je, podobně jako u výše zmíněné aplikace **Rational Rose**, poněkud vysoká cena a delší čas na zaučení.

6.3 I-Logix Rhapsody

Systém **I-Logix Rhapsody** je vyvíjen americkou firmou I-Logix pro řešení problémů řízení a vývoje software s využitím UML diagramů. Z modelů lze automatizovaně vytvořit zdrojový kód s hlavičkami veškerých tříd spolu s jejich rozhraními. Tento systém také umožňuje správu poznámek a ručně kreslených grafů i schémat (specializovaný modul AGE: Advanced Graphics Engine). Na tomto produktu je zajímavý také fakt, že je kromě podpory programovacích jazyků C, C++ a Java podporován i programovací jazyk Ada, který se používá například v některých projektech NASA a ve vojenských institucích US Army. Předností tohoto produktu je jeho rozsáhlá funkcionality, zejména při týmové práci většího množství vývojářů a vývojových skupin.

6.4 MetaMatrix MetaBase Modeler

MetaMatrix je tvořen soustavou několika relativně nezávislých aplikací, které mezi sebou navzájem komunikují. Účelem těchto aplikací je tvorba, sdílení a prezentace různých typů informací v podnikových informačních systémech. Při návrhu těchto systémů je využit modelovací a návrhový program založený právě na jazyku UML a UML diagramech. Tento systém také zajišťuje propojení s dalšími podnikovými informačními systémy, k tomuto účelu používá technologie založené na jazyku SQL, Xquery (databázový dotaz založený na XML), SOAP atd.

6.5 Together Designer Community Edition

Systém **Together Designer** dostupný na adrese www.borland.com/together [viz: <http://www.borland.com/together/>] je vytvářen firmou Borland (pro výrobky této firmy mám určitou slabost). Mezi základní vlastnosti tohoto systému patří podpora jednoduché tvorby UML diagramů, návaznost na další vývojové nástroje této firmy (JBuilder, C++ Builder) a také možnost navázání na softwarové nástroje jiných firem (prezentovaný je například plugin ke známému vývojovému prostředí Eclipse - bližší informace viz www.eclipse.org [viz: <http://www.eclipse.org/>]). Kromě základního systému **Together Designer** jsou nabízeny i jeho modifikované verze nazvané **Together Architect** a **Together Developer**. Jednotlivé verze se

od sebe liší svou funkcionalitou, tj. zejména nabídkou jednotlivých submodulů (a samozřejmě i cenou, Borland nabízí většinu svých produktů už po delší dobu ve třech cenových kategoriích).

7. Produkty šířené pod GPL a podobnými licencemi

V této podkapitole bude uveden - pro čtenáře tohoto serveru patrně mnohem zajímavější - seznam produktů, které jsou určeny pro práci s jazykem UML a UML diagramy. Zaměřím se zejména na produkty šířené pod licencí GPL či pod licencí podobného charakteru - dále popisované aplikace jsou tedy šířeny zdarma a i jejich použití pro tvorbu prvotních dat není žádným zvláštním způsobem omezeno. Předností a současně i záporem těchto produktů je jejich menší komplexnost, což znamená nižší čas na zaučení.

7.1 Violet

Mezi aplikace určené pro tvorbu UML diagramů, které jsou šířeny pod licencí GPL, patří i na první pohled poměrně jednoduchý projekt **Violet**. Tento projekt, který je dostupný na stránce horstmann.com/violet [viz: <http://horstmann.com/violet>], slouží pouze k tvorbě UML diagramů, ovšem bez další návaznosti na zdrojový kód. Podle autora této aplikace je její nasazení vhodné například pro studenty či učitele, kteří se zabývají výukou softwarového inženýrství či objektově orientovaného programování; jeho reálné použití je však širší, zejména v případě, kdy není zapotřebí z UML diagramu generovat přímo zdrojový kód.

Ze všech možných typů UML diagramů lze v této aplikaci vytvářet **diagram tříd** (Class Diagram), **sekvenční diagram** (Sequence Diagram), **stavový diagram** (State Diagram) a **objektový diagram** (Object Diagram). Výčet typů diagramů sice není úplný (zejména při pohledu na normu jazyka UML), ale pro použití UML jako návrhového nástroje pro vývoj programových produktů se jedná o dostačující podmnožinu veškeré funkcionality jazyka UML. Samozřejmě, že mohou nastat případy, kdy je tato funkcionalita nedostačující (například může citelně chybět diagram spolupráce) - zde je nutné uvažovat o použití jiného programového produktu, například dále zmíněné aplikace **ArgoUML**.

Vytvořené UML diagramy lze uložit buď v interním formátu založeném na značkovacím jazyku XML, nebo je možné provést export diagramu do rastrového obrázku (v současné době jsou podporovány formáty PNG a JPG), nebo do vektorově orientovaného formátu PostScript. Zde mi chybí možnost exportu do dalších vektorových formátů, například DXF, WMF nebo LaTeXovského prostředí "picture". Předností této aplikace je především to, že je kompletně vytvořena v Javě, a je tak bez větších problémů přenositelná na různé platformy a operační systémy. Také práce s jednotlivými typy diagramů je poměrně jednoduchá, snad až na poněkud problematickou změnu umístění jednotlivých stavů ve stavovém diagramu. Nevýhodou této aplikace je, že není možné z UML diagramu přímo generovat zdrojový kód v některém objektově orientovaném programovacím jazyku tak, jak to nabízí prakticky všechny výše zmíněné komerční nástroje.

7.2 Umbrello UML Modeller

Softwarový projekt nazvaný **Umbrello UML Modeller**, který je dostupný na URL adrese uml.sourceforge.net/index.php [viz: <http://uml.sourceforge.net/index.php>], je určen operační systémy typu Unix a samozřejmě také pro Linux. Tento projekt je vyvíjen v prostředí KDE, v dnešní době tedy není dostupný pro další operační systémy, jež KDE nepodporují. Velkou předností tohoto projektu je rozsáhlá podpora pro mnoho programovacích jazyků, včetně PHP 5, SQL, Pythonu a dokonce i Perlu :-). Pro některé vytvořené jazykové prvky lze použít i refaktoring, který není mnohdy dostupný ani v drahých komerčních aplikacích. Vzhledem k tomu, že se jedná o aplikaci napsanou v programovacím jazyce C++, je rychlost

odezvy programu vyšší než u výše zmíněné aplikace **Violet** či níže zmíněného programu **ArgoUML** (tyto aplikace jsou napsány v Javě).

7.3 ArgoUML

Dalším programovým balíkem, který je možné použít pro práci s jazykem UML a UML diagramy, je balík nazvaný **ArgoUML**. Tato aplikace, jež je dostupná na URL adrese argouml.tigris.org [viz: <http://argouml.tigris.org/>], je vytvořena v programovacím jazyku Java, to znamená, že je zaručena její snadná přenositelnost na různé operační systémy a výpočetní platformy (samozřejmě musí být k dispozici JRE). V této aplikaci je možné vytvářet následující typy UML diagramů:

1. **Diagram tříd** (Class Diagram)
2. **Sekvenční diagram** (Sequence Diagram)
3. **Diagram použití** (Use Case Diagram)
4. **Diagram spolupráce** (Collaboration Diagram)
5. **Diagram aktivit** (Activity Diagram)

Práce s jednotlivými typy UML diagramů je velmi intuitivní, u některých diagramů lze dokonce vytvářet hierarchicky organizované poddiagramy, což některé výše zmíněné jednodušší nástroje neumožňují (například Violet). Z diagramu tříd lze vytvořit zdrojové soubory, v současné době je však podporován pouze programovací jazyk Java (chybí oblíbené C++). Mezi přednosti tohoto projektu patří jeho poměrně dobrá funkcionality, velký počet podporovaných UML diagramů a příjemné grafické uživatelské prostředí. Mezi nevýhody lze řadit pouze (prozatím) menší počet podporovaných programovacích jazyků, do kterých může být UML diagram zkonvertován. Také rychlost odezvy na uživatelský vstup je poněkud menší než u nativních aplikací, protože se jedná o aplikaci napsanou v Javě s využitím GUI knihovny Swing (ta opravdu rychlostí neoplývá).

8. Závěr

V tomto článku jsem popsal některé aplikace, které je možné použít pro interaktivní práci s jazykem UML. Samozřejmě nejsou zdaleka popsány všechny existující aplikace, všechny výše zmíněné projekty jsou však již dostatečně funkční a použitelné pro reálnou práci.



Root.cz (www.root.cz), informace nejen ze světa Linuxu. ISSN 1212-8309

Copyright © 1998–2007 Internet Info, s.r.o. Všechna práva vyhrazena. Powered by Linux.

Ti	Parf	Bateri	LCD	Hodin	Bíl	Tvor	Lev	Pneum	Nář	Zahra	DVD	Ch		povi	last	Řec
py	émy	e do	televi	ky,	é	ba	né	atiky	adí	dní	rekor	ci	lete	nné	min	ko
		Noteb	ze,	jízdni	zbo	web	pne	prodej		techn	déry	do	nky	ruč	ute	
		ooku	MP3	kola	ží	ů	u			ika		liš	y	ní		