

Nástroje pro vývoj aplikací v závislosti na platformě a jejich vazba na CASE

Semestrální práce v rámci předmětu 4IT450 – CASE - Computer Aided Systems Engineering

Vyučující: prof. Ing. Václav Řepa, CSc.

Semestr: LS 2008/09

Tým: Bc. Adam Vopička, Bc. Jan Florian, Bc. Ondřej Píša, Bc. Kamil Jireš

Obsah

Úvod.....	3
1 Základní pojmy.....	4
1.1 Platforma.....	4
1.2 Vývojové prostředí.....	5
1.3 CASE.....	6
1.3.1 Upper CASE Tools.....	7
1.3.2 Lower CASE Tools.....	7
2 Kritéria pro porovnávání vývojových prostředí.....	7
2.1 Licence.....	8
2.2 Cena.....	8
2.3 Možnost vyzkoušení zdarma.....	8
2.4 Lokalizace.....	9
2.5 Operační systémy, pod kterými IDE funguje.....	9
2.6 Programovací jazyky, kterých IDE umožňuje vývoj.....	9
2.7 Podpora CASE.....	9
3 Popis jednotlivých prostředí dle vybraných kritérií.....	9
3.1 Eclipse.....	9
3.1.1 Aptana Studio.....	10
3.1.2 Zend Studio for Eclipse.....	11
3.1.3 CASE.....	11
3.2 NetBeans.....	12
3.2.1 CASE.....	12
3.3 ERIC IDE.....	13
3.3.1 CASE.....	13
3.4 Xcode.....	13
3.5 IntelliJ IDEA.....	15
3.6 RubyMine.....	16
3.7 Delphi.....	16
3.8 KDevelop.....	17
3.9 Microsoft Visual Studio.....	18
4 Shrnutí a porovnání platforem z hlediska nabídky nástrojů s podporou CASE.....	20
4.1 Porovnání IDE dle základních kritérií.....	20
4.2 Porovnání prostředí dle podpory CASE.....	22
4.3 Porovnání platforem z hlediska nabídky nástrojů s podporou CASE.....	22
4.4 Shrnutí.....	22
Závěr.....	24
Použité zdroje.....	25

Úvod

Tématem naší semestrální práce v rámci předmětu 4IT450 (CASE - Computer Aided Systems Engineering) je porovnání vývojových nástrojů sloužících k vývoji softwarových aplikací v závislosti na platformě a jejich vazbě na CASE. Toto zadání je dosti široké a pochopitelně je v první řadě vhodné si jej nějakým způsobem blíže vymežit a upřesnit.

Nejprve tedy blíže popíšeme, co vlastně v naší práci budeme chápat pod pojmy „platforma“, „vývojové prostředí“ a „CASE“, a to včetně příslušného rozdělení pro naše účely. Právě pojetí platformy bude největší odlišností v pojetí tohoto tématu našimi kolegy – absolventy předmětu 4IT450 v zimním semestru 2008/09, kteří byli prvními, jenž toto téma zpracovávali. V následující části dále zvolíme vhodná kritéria pro porovnání vývojových prostředí.

Při popisování jednotlivých prostředí bude kladen důraz na přímou osobní zkušenost s tím kterým produktem, což je (spolu s faktem, že to mají za úkol jiné týmy) také důvod, proč nám nepůjde o kompletní pokrytí trhu těchto produktů, jako spíš o výběr těch, které známe z praxe a které jsou dle našeho názoru prostředími nejpoužívanějšími.

Pro získávání informací budeme záměrně vzhledem k rychlému vývoji vývojových prostředí využívat elektronické zdroje, a to především webové stránky výrobců a dokumentaci jednotlivých vývojových prostředí. Komunikace uvnitř týmu a finalizace práce potom bude řešena pomocí systému Google Docs.

1 Základní pojmy

Abychom mohli přejít k určení kritérií, je vhodné vysvětlit si tři základní pojmy, se kterými budeme dále pracovat. Jsou jimi platforma, vývojové prostředí a CASE.

1.1 Platforma

Pojem „platforma“ je možné ve výpočetní technice chápat mnoha způsoby. Může to být například:

- sada instrukcí procesoru, tedy soubor možných „volání“, které může operační systém počítače použít pro komunikaci s procesorem počítače a využívání jeho funkcí,
- softwarová platforma fungující v podstatě obdobně, jen na softwarové úrovni a obvykle v rámci jednoho programovacího jazyka,
- operační systém, tedy software, který běží mezi hardwarem a aplikacemi, a uživatelům více či méně komfortně umožňuje provoz těchto aplikací.

Pro naši semestrální práci jsme si vybrali právě poslední jmenované rozdělení, a to z následujících důvodů:

- Zájem členů našeho týmu a v poslední době i veřejnosti o „alternativní“ operační systémy, tedy systémy jiné, než Microsoft Windows. Tři z nás pro běžnou práci používají některou z distribucí systém GNU/Linux (viz níže) a čtvrtý člen týmu je uživatelem systému MacOS.
- Možné ekonomické úspory při využití operačních systémů, jejichž legální využívání je zdarma. Tyto úspory jsou pochopitelně reálnější ve střednědobém až dlouhodobém horizontu, protože přechod na takové systémy samozřejmě znamená náklady na instalaci, školení a administraci. V některých firmách či veřejných institucích nicméně k těmto přechodům již dochází. To také souvisí s tím, že uživatelská stránka těchto systémů prošla zejména zhruba v posledních třech letech bouřlivým vývojem a nejde tedy jen o operační systémy pro hrstku nadšenců.
- Možnost snadnějšího začlenění do existující podnikové infrastruktury.
- Již v úvodu zmíněná snaha o odlišný přístup než měl předchozí tým zpracovávající toto téma.

Vzhledem k našim zkušenostem a možnostem jsme se rozhodli zaměřit se na následující operační systémy:

- **Microsoft Windows** – Patrně nejrozšířenější operační systém na světě, který většině uživatelů není třeba blíže představovat. I přes řadu neduhů se již desetiletí drží na prvním místě v počtu

uživatelů. V současnosti nejpoužívanější verze nesou označení 2000, XP, Vista a 2008, přičemž první tři jmenované existují v několika mutacích určených pro domácí či firemní prostředí a poslední je používána výhradně na serverech.

- **GNU/Linux** – Operační systém vycházející ze systémů unixového typu je postavený na open-source jádru Linux¹. Podle tohoto jádra také bývá obvykle zkráceně nazýván.

Určitým specifikem GNU/Linuxu je to, že neexistuje jedna určitá verze od jednoho výrobce – vzhledem k otevřenosti celého systému si jej v podstatě každý uživatel s patřičnými technickými schopnostmi může upravit k obrazu svému. Zhruba od poloviny devadesátých let tak vzniká nepřehledné množství tzv. distribucí. Ty se liší jak množstvím nabízeného software (obvykle mají zabudovaný tzv. balíčkovací software umožňující uživateli jednoduše software nainstalovat), tak uživatelskou přívětivostí, nároky na uživatele a tím nepřímo i cíleným využitím. Nejpoužívanějšími distribucemi jsou v současnosti dle serveru DistroWatch.com² Ubuntu, openSUSE, Fedora, Debian a Mandriva³. Kromě těchto distribucí, jejichž využívání nevyžaduje po uživateli placení licenčních poplatků, existují ale i komerční distribuce jako Red Hat Enterprise Linux, které za určitý poplatek nabízejí jako přidanou hodnotu obvykle technickou podporu.

- **Mac OS** – Mac OS (v aktuální verzi Mac OS X 10.5.6) je operační systém svázaný s počítači firmy Apple⁴, která jej nedovoluje (legálně) provozovat mimo své počítače i přes to, že několik posledních let tyto počítače využívají hardwarovou architekturu x86 (stejně jako např. naprostá většina „obyčejných“ osobních počítačů). Jedná se o uživatelsky přívětivý operační systém postavený na UNIXovém základu⁵.

Ještě dodejme, že všechny tři zmíněné operační systémy existují (pochopitelně dle verze / distribuce) v 32-bitových i 64-bitových verzích.

1.2 Vývojové prostředí

Vývojovým prostředím v našem kontextu budeme rozumět software, který programátorovi umožňuje komfortnější vývoj aplikací, než kterého dosáhne využitím jednoduchého textového editoru.

1 Aktuální verze jádra je 2.6, viz The Linux Kernel Archives [online]. [2009] [cit. 2009-05-03]. Dostupný z WWW: <<http://www.kernel.org/>>.

2 DistroWatch.com [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://distrowatch.com/dwres.php?resource=major>>.

3 Ubuntu [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.ubuntu.com/>>. / OpenSUSE.org [online]. 2008 [cit. 2009-05-03]. Dostupný z WWW: <<http://cs.opensuse.org/>>. / Fedora Project [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://fedoraproject.org/>>. / Debian [online]. 2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.debian.org/>>. / Mandriva [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www2.mandriva.com/>>.

4 Apple [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.apple.com/>>.

5 Apple - Mac OS X Leopard - Technology - UNIX [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.apple.com/macosx/technology/unix.html>>.

Takový software je obvykle označován zkratkou IDE (Integrated Development Environment), tedy „integrované vývojové prostředí“. Mezi obvyklé funkce takových prostředí patří například⁶:

- editor zdrojového kódu umožňující zvýrazňování syntaxe,
- doplňování zdrojového kódu při psaní dle syntaxe,
- zjednodušená kompilace zdrojového kódu,
- debugger, tedy nástroj umožňující hledání chyb v programu,
- nástroje pro vytváření uživatelských rozhraní,

a další.

1.3 CASE

V případě CASE nástrojů je žádoucí začít definicí pojmu, co vlastně „CASE“ (Computer-Aided Software Engineering) znamená. Jedna z mnoha definic je sice stručná, zato možná nejvýstižnější:

„CASE is the use of computer-based support in the software development process.“⁷

Jedná se tedy o soubor nástrojů, které automatizují činnosti spojené s vývojem software. Funkcionalita CASE v sobě zahrnuje činnosti pokrývající celý životní cyklus programu, tedy analýzu, design, samotné programování i tvorbu závěrečné dokumentace. Existují⁸ klíčové myšlenky, kvůli kterým CASE vzniklo a je snaha o to, aby vývoj software (jako disciplína) dosáhl:

- plné zakomponování počítačové asistence do vývoje a správy software,
- inženýrský přístup k této disciplíně.

Je potřeba říci, že v současné době je díky dostupným IDE obou cílů dosaženo.

Jednotlivé CASE nástroje v IDE je možné strukturovat z mnoha možných pohledů, nicméně základní dělení je podle fáze životního cyklu aplikace, ve které jednotlivé nástroje nastupují (nejedná se o striktně vymezené hranice, ve kterých se použití těchto nástrojů drží, nýbrž pouze o nástup použití). Z tohoto pohledu rozeznáváme Upper CASE Tools a Lower CASE Tools. V následující části jsou uvedeny konkrétní typy nástrojů. Vybrané nástroje jsou pouze reprezentativním vzorkem, neboť není možné vytvořit komplexní seznam CASE nástrojů. Valná většina z nich je běžnou součástí každého vývojového prostředí – každá možnost určité akce je do jisté míry zautomatizovaná a

6 NOURIE, Dana. Getting Started with an Integrated Development Environment (IDE) [online]. 2005 [cit. 2009-05-03]. Dostupný z WWW: <<http://java.sun.com/developer/technicalArticles/tools/intro.html>>.

7 What is a CASE Environment? [online]. c2007 [cit. 2009-05-05]. Dostupný z WWW: <http://www.sei.cmu.edu/legacy/case/case_what.html>.

8 Computer-aided software engineering - Wikipedia, the free encyclopedia [online]. 2009 [cit. 2009-05-05]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Computer-aided_software_engineering>.

nespočívá pouze v činnosti programátora – uživatele IDE.

1.3.1 Upper CASE Tools

Tyto nástroje přicházejí na úplném počátku životního cyklu aplikace. Podporují tedy převážně strategické plánování, analýzu a modelování na konceptuální úrovni. Do úrovně samotného designu vůbec nezasahují. Mezi tyto nástroje patří především jazyk UML a konceptuální modelování. Lze sem ještě zahrnout konfigurační management - opět ovšem na konceptuální úrovni. Nejčastěji používanými druhy modelu jsou:

- data flow diagram
- ER diagram (entity relationship)
- logická schémata

1.3.2 Lower CASE Tools

Lower CASE nástroje se koncentrují spíše na tzv. back-end aktivity životního cyklu vývoje software a přímo podporují řadu akcí, které uživatel při vývoji v IDE vyvolává. Namátkou se jedná o:

- debugging,
- refactoring,
- generování kódu,
- systémy kontroly verzí,
- testování,
- dynamická kontrola.

Již zde je možné vidět vzájemné propojení jednotlivých nástrojů a jejich spolupráce napříč celým životním cyklem aplikace. Například z vhodně vytvořených ER-diagramů je možné nechat vygenerovat kód pro vytvoření namodelovaných objektů. Při pokročilé práci a potřeby editace nám refactoring velmi usnadní práci při provádění globálních změn v projektu, vše se dynamicky kontroluje z hlediska věcné správnosti a k jednotlivým krokům se na požadavek uživatele vytvoří potřebná dokumentace.

2 Kritéria pro porovnávání vývojových prostředí

Abychom mohli systematicky porovnávat IDE, je třeba zvolit základní kritéria. Ta jsme vybrali následovně:

2.1 Licence

Licenci rozumíme to, jaká práva a povinnosti vyplývají z využívání konkrétního IDE. Základní rozdělení licencí může být následující:

- **Komerční licence** – Software je povoleno využívat pouze za úplatu. Částka se obvykle liší podle verze produktu, počtu licencí (dle počtu pracovních stanic) atd.
- **Freeware** – Software, jehož využívání (a často i distribuce) je zdarma. Nejsou však dostupné zdrojové kódy.
- **Open-source** – Software s otevřenými (volně dostupnými) zdrojovými kódy. Existuje mnoho (možná až přespříliš) licencí, ze kterých si tvůrce softwaru může vybírat, a samozřejmě je zde i možnost vytvoření licence vlastní. Tyto licence obvykle nějakým způsobem upravují využívání software, distribuci odvozeného software atd. V současnosti nejpoužívanější open-source licencí je GNU General Public License ve verzi 2⁹ (aktuální verze je nicméně 3¹⁰), dále lze jmenovat např. Eclipse Public License¹¹

Kromě těchto tří případů je možné i jejich kombinování, například možnost použití pro nevydělečné nebo studijní účely zdarma, ale povinnost za software platit při komerčním využití.

2.2 Cena

Cena samozřejmě bezprostředně souvisí s komerční licencí. Zde nás bude zajímat kromě cen různých verzí (pokud v daném případě existují) také možnost množstevních, studentských či jiných slev.

2.3 Možnost vyzkoušení zdarma

U komerčních prostředí, stejně jako obvykle u jiného komerčního software, bývá dobrým zvykem poskytnout uživatelům verzi k vyzkoušení. Taková verze (demoverze, trial verze) bývá obvykle nějakým způsobem omezená. Může jít čistě o licenční řešení (omezení je podchyceno pouze v licenci), častěji se ale můžeme setkat s řešením technologickým omezením zabudovaným přímo v software.

Pokud jde o způsob omezení, je časté omezení některých funkcí (např. debugger nebo ukládání projektu), nebo časové omezení (např. že lze software využívat pouze 30 dní a po uplynutí této doby je

9 GNU General Public License Version 2 [online]. 1991 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.gnu.org/licenses/old-licenses/gpl-2.0.txt>>.

10 GNU General Public License Version 3 [online]. 2007 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.gnu.org/licenses/gpl-3.0.txt>>.

11 Eclipse Public License [online]. [2004] [cit. 2009-05-03]. Dostupný z WWW: <<http://www.eclipse.org/legal/epl-v10.html>>.

nutno software přestat používat / odinstalovat nebo zakoupit).

2.4 Lokalizace

Lokalizací máme na mysli překlad do českého jazyka, a to jak překlad uživatelského rozhraní, tak i např. dokumentace.

2.5 Operační systémy, pod kterými IDE funguje

Toto jsme již nastínili v sekci věnující se pojmu platforma. Kromě našich tří vybraných operačních systémů ale samozřejmě může vývojové prostředí podporovat i provoz pod jinými operačními systémy, ať už nativně¹², nebo interpretovaně¹³. Z dalších operačních systémů můžeme jmenovat například FreeBSD, OpenBSD, (Open)Solaris, BeOS atd.

2.6 Programovací jazyky, kterých IDE umožňuje vývoj

Nabízený programovací jazyk (či velmi často jazyky) je jedním ze základních charakteristik IDE.. Je to ostatně faktor, který často hraje při výběru IDE hlavní roli – vývojář má obvykle zkušenosti s některým jazykem (některými jazyky), nebo je konkrétní jazyk vhodný pro určitý projekt. Potom je pochopitelně nesmyslné, aby se programátor přizpůsoboval nabídce jazyků určitého IDE.

2.7 Podpora CASE

Vzhledem k tématu práce je pro nás důležitý i pohled z hlediska CASE. Pro ten budeme využívat poznatky uvedené v kapitole 1.3.

3 Popis jednotlivých prostředí dle vybraných kritérií

3.1 Eclipse

Eclipse¹⁴ je šířen pod licencí EPL¹⁵ a díky tomu, že je založen na Javě, běží na jakékoliv platformě podporované Java Virtual Machine. Samozřejmostí je Linux (32-bit i 64-bit), Windows, MacOS, Solaris a další.

IDE bylo původně určeno pouze pro vývoj v Javě, ale postupně se díky svému systému pluginů transformovalo na komfortní prostředí pro většinu známých jazyků. Prostředí jako takové je zdarma,

¹² Přímou zdrojový kód je zkompileován přímo pro daný operační systém.

¹³ Zdrojový kód je přeložen do „mezikódu“, který je možné spustit za pomoci určitého prostředí (například Java Runtime Environment / Java Virtual Machine) na libovolném operačním systému, pod kterým funguje toto prostředí.

¹⁴ Eclipse.org home [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.eclipse.org/>>.

¹⁵ Eclipse Public License - Wikipedia, the free encyclopedia [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Eclipse_Public_License>.

stejně tak jako velká řada pluginů. Na jeho základě (Eclipse Framework) jsou postaveny i komerční produkty, ať už jako samostatná řešení, pluginy, nebo obojí. Dnes tedy můžeme v Eclipse vyvíjet nejen v Javě, ale využít i podporu pro C, C++, C#, Python, Perl, Ruby, PHP, XHTML, CSS, JavaScript, Lua, XML, SQL, LISP, Adobe AIR, iPhone a mnohé další jazyky. V současnosti je k dispozici téměř 1200 pluginů¹⁶, a to nejen pro různé programovací jazyky, ale i například pro softwarové modelování či přístup k databázovým systémům.

Prostředí je celkem dobře přizpůsobitelné a svižné. Někomu vadí klasický „Java syndrom“ aplikace, projevující se delším startem a náročností na operační paměť. Platforma má vestavěný systém pro správu pluginů a systém aktualizací, použitelný i v případě, že se jedná o produkty třetích stran. Jak v základu, tak při využití pluginů prostředí oplývá pěknou řádkou zajímavých funkcí. Od doplňování slov (autocompletion), zvýrazňování syntaxe a code folding (skrývání některých částí kódu, například v rámci jedné funkce), podporu dokumentace kódu, inteligentního vyhledávání a nahrazování i pomocí regulárních výrazů v rámci celého projektu, nástrojů na refactoring kódu, přes podporu řady verzovacích systémů (CVS, SVN, Bazaar), sestavovacích nástrojů, generátorů grafického uživatelského rozhraní, prohlížeč tříd až po debugger a testovací nástroje.

O oblibě prostředí zajisté vypovídá téměř půl milionu stažení¹⁷ poslední verze 3.4.2, které ovšem nezahrnuje stažení z repozitářů¹⁸ na Linuxu, *BSD, Solarisu a podobných operačních systémech. Mezi velké přednosti určitě patří také možnost mít „vše pod jednou střešou“ ve smyslu jednotného vývojového prostředí pro více jazyků.

Jak jsme již zmínili, pokud něco bylo a je devizou Eclipse, tak tím je obrovská rozšiřitelnost za pomoci pluginů:

3.1.1 Aptana Studio

Aptana¹⁹ je plugin / samostatné IDE, které se specializuje převážně na vývoj webových projektů. Od toho se také odvíjí podpora HTML, CSS, JavaScriptu (včetně frameworků jako jQuery, Prototype, script.aculo.us, YUI, dojo, Ext JS, MooTools atd.), PHP, Pythonu, Ruby on Rails. Mimo vyjmenované jazyky také Aptana poskytuje nástroje pro mobilní platformy jako iPhone či Nokia a navíc stále populárnější multiplatformní Adobe AIR. „Spolka“ i populární plugin PyDev²⁰, který je nicméně stále dostupný i mimo Aptanu. Mezi vlastnosti vyčnívající nad standardní výbavu patří

16 Eclipse Plugin Central :: Eclipse plugins resource center and marketplace for Eclipse and Plugin Ecosystem [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.eclipseplugincentral.com/>>.

17 Eclipse Downloads [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.eclipse.org/downloads/>>.

18 Repozitář je místo (typicky na Internetu), ze kterého je možné stahovat softwarové balíčky. Mechanismus umožňuje automatické stahování dalších balíčků na nichž je požadovaný závislý. Repozitáře bývají digitálně podepsány, aby se zamezilo možnosti distribuce škodlivého software. Tento systém distribuce software je hojně využíván u Linuxových distribucí.

19 Aptana [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.aptana.com/>>.

20 Pydev [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://pydev.sourceforge.net/>>.

JavaScript debugger a celkem pěkně zpracovaný nástroj pro zasílání dotazů, prohlížení výsledků a struktur tabulek na SQL serverech. Produkt je volně ke stažení. K 1.5. 2009 bylo IDE staženo více než 3 700 000 krát.

3.1.2 Zend Studio for Eclipse

Zend Studio²¹ bylo dříve samostatným produktem, ale od řady 6 je založeno na Eclipse. Toto IDE je komerčním řešením pro vývoj PHP aplikací. Nabízí skvělé prostředí těžící ze spousty vlastností již obsažených v Eclipse, které ale dále rozšiřuje. Integruje doplňování slov přímo s PHP manuálem nebo s dokumentací vytvořenou podle PHPDoc u vlastních objektů, metod a funkcí zahrnutých v projektu. Programátor si může prostředí plně přizpůsobit „vlastním návykům“ psaní kódu – umístění závorek, mezer atd. To je potom zohledňováno nejen při psaní nového kódu, ale je možné (pomocí klávesové zkratky) nechat již stávající kód přeformátovat. IDE již při psaní kódu zvýrazňuje syntaktické chyby a upozorňuje na deklarace vedoucí k chybám. Komentáře @TODO a @FIXME jsou u projektu uvedeny v seznamu a je možné jednoduše jimi procházet. Vývojář si může vytvářet i vlastní šablony kódu a IDE obsahuje i jednoduchou databázi snippetů (malých, znovupoužitelných kousků kódu).

Výborná je i podpora debuggerů. Dříve protěžovaný Zend Debugger je nyní doplněn i podporou XDebugu. Samozřejmostí je podpora Zend serveru a Zend Frameworku. IDE s balíčkem roční podpory stojí 399 USD.

3.1.3 CASE

Eclipse Foundation nabízí mimo klasického IDE i balík určený přímo pro modelování – Eclipse Modeling Project²². Eclipse Modeling Framework (EMF) je modelovací framework a nástroj na generování kódu pro sestavovací nástroje založené na strukturovaném datovém modelu. Modely lze odesílat do relačních databází pomocí CDO a tím jednoduše řešit jejich úpravy a následnou distribuci. Framework obsahuje i nástroje pro ověření integritních omezení apod. Je možné vytvářet následující modely dle UML 2.1: Activity diagram, Class diagram, Component Diagram, Composite Structures diagram, Deployment diagram, Profile diagram, StateMachine diagram a Use Case diagram.

V budoucnu má být implementován i framework pro transformaci modelů (Model-to-model transformation). Generátory kódu jsou obsluhovány pomocí M2T – Model-to-Text translation.

21 Zend Studio - The Leading PHP IDE from the PHP Experts – Zend.com . 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://www.zend.com/products/studio/>.

22 Eclipse Modeling Project [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://www.eclipse.org/modeling/>.

3.2 NetBeans

NetBeans je IDE, jehož prapředek Xelfi byl vyvinut²³ jako studentský projekt na Matematicko – fyzikální fakultě Karlovy univerzity a později odkoupen firmou Sun, která umožnila jeho šíření jako open-source software. Zvláště v poslední době se mezi vývojáři těší větší a větší popularitě. Tento trend přisuzujeme přidání podpory jazyků masově používaných na vývoj webových aplikací. Jde zejména o PHP, Ruby (Ruby on Rails), Python a základ v podobě HTML, CSS a JavaScriptu. Díky nedávnému nákupu Sunu firmou Oracle se ale množí názory, že by tímto krokem mohly být NetBeans v budoucnosti ohroženy.

IDE lze stáhnout přímo pro jednotlivé známější jazyky, nebo jako balík s podporou jazyků / technologií Java SE, Web & Java EE, SOA, Ruby, C/C++, PHP, HTML, JavaScript, CSS, Apache Tomcat a dalších. NetBeans podobně jako Eclipse poskytují platformu pro rozšíření IDE o další funkcionalitu pomocí pluginů.²⁴ Prostředí je založené na Javě, čímž je zaručena podpora napříč platformami (jmenovitě minimálně Windows / Linux / Solaris / MacOS). Bonusem je lokalizace do značného množství jazyků, včetně češtiny.

Prostředí je v základu vybaveno klasickými pomůckami – zvýrazňování syntaxe, code folding, autocompletion, výbornou podporou verzovacích systémů (CVS, SVN, Mercurial, Clearcase), kterou nad konkurenci vyvyšuje především možnost vidět úpravy oproti HEAD revizi jako podbarvené řádky přímo v kódu. Příjemná je možnost nahrávání a spuštění maker, samozřejmostí jsou „prohlížeče“ struktur tříd a jejich prvků, pomůcky pro kompilaci a nasazení software, podpora refactoringu kódu a debugger. Velice užitečná je také možnost online chatu mezi vývojáři. Vývojáři díky této funkci nemusí opouštět IDE, když potřebují něco rychle probrat. Povedená je také podpora lokalizace projektů.

3.2.1 CASE

NetBeans je možné pomocí UML pluginů²⁵ rozšířit o pěknou řádku zajímavých funkcí. Plugin umožňuje jak reverse engineering (tedy vytvoření diagramu z již existujícího kódu), tak forward engineering (vygenerování zdrojového kódu na základě diagramu). Programátoři tedy mohou vygenerovat Java kód přímo z UML navrženého analytiku. Vygenerovaný kód lze kdykoliv nechat přegenerovat na základě změn v UML. Takto vytvořený kód lze i sloučit s kódem již existujícím. Systém generování kódu je založen na šablonách a díky tomu lze jeho chování do značné míry ovlivnit.

23 NetBeans - Wikipedia, the free encyclopedia [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: http://en.wikipedia.org/wiki/Netbeans#Early_history.

24 Plugin Portal Main Page [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://plugins.netbeans.org/PluginPortal/>.

25 NetBeans IDE - Unified Modeling Language (UML) Plugin [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://www.netbeans.org/features/uml/>.

Pomocí GUI lze vytvářet Activity diagram, Class diagram, Sequence diagram, State diagram, a Use Case diagram. Prostředí podporuje drag & drop.

3.3 ERIC IDE

Eric IDE²⁶ je vývojové prostředí pro Python a Ruby, kompletně napsané v Pythonu a založené na multiplatformní knihovně Qt²⁷. Jeho aktuální verzi tedy není problém spustit na jakékoliv platformě podporující aktuální verze Pythonu, a knihoven PyQt, QScintilla. IDE jako takové je distribuováno pod GPL a je ke stažení (aktuální verze je 4.3.2) na SourceForge²⁸ nebo pomocí repozitářů. K dispozici jsou překlady do několika světových jazyků a velmi překvapivě i češtiny.

Prostředí je velmi svižné a hodí se tak jako „rychlý editor“ stejně jako na vývoj rozsáhlejších projektů. Funkcionalitu lze rozšiřovat pomocí systému pluginů²⁹. IDE lze tak snadno rozšířit například o podporu oblíbených webových frameworků založených na Pythonu jakými jsou například Django, Pylons, TurboGears, nebo o podporu verzování pomocí CVS a SVN.

Po instalaci Eric nabízí spoustu zajímavých funkcí. Přes základ v podobě zvýrazňování syntaxe, autocompletion, code foldingu, zvýrazňování párových závorek a syntaktických chyb a prohlížeč tříd přes generování dokumentace a API kódu, správu TODO komentářů, nahrávání maker, refactoring kódu, testovací nástroje až po debugery pro oba jazyky (u Pythonu je zajímavá podpora i vícevláknových aplikací), integraci CORBA, průvodce a generátory grafického uživatelského rozhraní pro Qt a překlad aplikací.

Za velmi užitečné funkce (pro Python) považují zejména kontrolu indentation (odsazení) kódu v celém projektu a generátor regulárních výrazů.

3.3.1 CASE

Eric sice umí vygenerovat z kódu primitivní Class / Package / Import / Application diagramy, ale ty jsou pro reálné využití de-facto bezcenné.

26 The Eric Python IDE [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://eric-ide.python-projects.org/>>.

27 Qt (čte se jako anglické „cute“) je grafická knihovna, běžící napříč OS platformami i některými mobilními zařízeními. Mezi populární a masově rozšířený software používající tuto knihovnu jsou například KDE, Google Earth, Opera, Skype, VirtualBox a mnoho dalších. V roce 2008 byla firma Trolltech vyvíjející tuto knihovnu odkoupena společností Nokia. (Qt - A cross-platform application and UI framework [online]. c2008-2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.qtsoftware.com/products/>>.)

28 SourceForge.net: Eric Integrated Development Environment: Files [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://sourceforge.net/project/showfiles.php?group_id=119070&package_id=233329>.

29 SourceForge.net: Eric Integrated Development Environment: Files [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://sourceforge.net/project/showfiles.php?group_id=119070&package_id=249666&release_id=548540>.

3.4 Xcode

Xcode³⁰ je balík nástrojů, které slouží primárně k vývoji aplikací pro platformu Mac OS X v jakékoliv její modifikaci (v praxi se jedná hlavně o plnohodnotný operační systém Mac OS X a jeho upravenou verzi určenou pro prostředí mobilních telefonů iPhone), nicméně se dá také velmi dobře použít pro tvorbu webových aplikací a služeb. Xcode je přiložený na DVD u každého operačního systému Mac OS X 10.5, není však přímou součástí běžné instalace systému. Další možnost, jak jej získat, je pomocí programu ADC (Apple Developer Connection), kde se po registraci zpřístupní možnost volného stažení celé sady. Dále je třeba zmínit, že Xcode verze 3 spolupracuje pouze se systémem 10.5.x, tedy s „Leopardem“, starší verze systému podporovány nejsou a je potřeba Xcode nižší verze.

Balík se skládá z těchto komponent: Xcode IDE, Interface Builder, Instruments, Dashboards, the WebObjects framework a kompletní dokumentace všech uvedených. Xcode obsahuje modifikovanou verzi GNU Compiler Collection (GCC), díky čemuž je možné vytvářet aplikace kompatibilní s oběma doposud používanými architekturami v počítačích Macintosh (tzv. universal binary build), tzn. je možné spuštění vytvořeného softwaru jak na platformě PowerPC, tak na Intel-based x86. Dále je možné vytvářet jak 32-bitové, tak 64-bitové aplikace. Z podporovaných jazyků jsou mezi hlavními C, C++, Fortran, Objective-C, Objective-C++, Java, AppleScript, Python a Ruby.

Hlavním výhodou tohoto nástroje je především jeho výkon. I když se pracuje s velkým množstvím objektů, vyhledává se v rozsáhlých zdrojových kódech nebo se pracuje s velkými objemy dat, uživatel prakticky nepozná nějaký pokles výkonu. Dále je velkým kladem „vizualizace“ kódu, kdy je možné pracovat s diagramy jako s objekty projektu, které následně podle nastavených pravidel generují finální kód. Za zmínku dále stojí poměrně kvalitně vyřešený refactoring, který opravdu spolehlivě mění požadované komponenty zdrojového kódu bez následné změny chování celkové aplikace. Customizace vlastní práce uživatele (nastavení automatického doplňování, zvýrazňování atd.) je samozřejmostí.

Unikátem tohoto IDE je spolupráce s aplikací Automator, která je součástí operačního systému Mac OS X. Automator je prostředí, ve kterém je možné „neprogramátorskou“ cestou zautomatizovat určité činnosti. Výsledek je velmi podobný jednoduchým skriptům či primitivním aplikacím, nicméně čas pro jeho vytvoření je mnohem kratší. Tvorba probíhá převážně metodou drag & drop a za použití speciální syntaxe založené na běžném jazyce (anglickém). Tyto samostatně vytvořené úkoly (task) je možné v Xcode načíst a ten je automaticky transformuje do použitého kódu.

Vzhledem k velmi kvalitně zpracovaným online video-tutorialům je dle mého názoru Xcode jasným favoritem pro zájemce o vývoj aplikací pro tuto platformu. V důsledku zobrazení struktury a

30 Xcode [online]. c2009 [cit. 2009-05-04]. Dostupný z WWW: <<http://developer.apple.com/tools/xcode/index.html>>.

objektů aplikace do diagramů je vývoj daleko přístupnější programátorům amatérům, kteří by se při složitějším projektu v kódu ztráceli. Nicméně je potřeba zdůraznit, že tento produkt není vhodný pro pokročilou práci v jazyce Java, neboť IDE neobsahuje takovou zásobu knihoven jako ostatní produkty a jejich přidávání či instalace dodatečných pluginů je poměrně náročná činnost.

3.5 IntelliJ IDEA

První verze IDEA³¹ vznikla roku 2001, nicméně hojně využívaným IDE se stalo až po uvolnění verze 5.0 v roce 2005, kde již byla plná podpora J2EE a zaměření na front-end aplikace na bázi XML, HTML, CSS a JavaServer Pages. Současná verze 8.1 podporuje následující jazyky: Java, JavaScript/Flex, HTML / XHTML / CSS, XML / XSL, Ruby / JRuby, Groovy, SQL, FreeMarker / Velocity. Dále je integrovaná podpora UML a designeru grafického uživatelského rozhraní na bázi Swing, což je widget toolkit (soubor prvků GUI, které zobrazují informace editovatelné uživatelem) pro Javu (součást Sun Microsystems' Java Foundation Classes). Dále také IDEA nabízí integraci s open source projekty jako je CVS, Subversion, Apache Maven nebo JUnit.

Výrobce JetBrains zvolil licenční politiku zaměřující se na veškeré možné skupiny zákazníků. Pro open-source projekty je tento nástroj zdarma. Pro komerční účely jsou k dispozici verze pro jedince či pro organizaci v ceně 249 / 599 USD. Pro studenty je určité zajímavá akademická licence, která nemůže být využita pro komerční účely za 99 USD.

Stejně jako například u Eclipse se v důsledku zvětšující se komunity uživatelů na internetu objevuje stále více pluginů rozšiřující možnosti použití. Výborná je i interoperabilita s Eclipse, je totiž možnost importovat, exportovat a synchronizovat projekty mezi těmito IDE. IDEA také podporuje zálohovací formát Eclipse. V tom spočívá velká výhoda při větších projektech, kdy může každý vývojář používat nástroje, na které je zvyklý a není zde zábrana pro tvorbu tzv. mixed-IDE týmů.

Z podpory CASE nástrojů stojí především za zmínku podpora UML. IDEA obsahuje funkci tvorby class diagramů, které umožňují provádění elementárních operací na úrovni tříd nebo o hierarchii výše - objektů. To následně velmi usnadňuje především navigaci v kódu rozsáhlých projektů. Kvalitně zpracovaná je i dynamická analýza kódu, která současně se záznamem kódu ověřuje jeho správnost z hlediska syntaxe i obsahu (ve smyslu relevantnosti použitých objektů). Celkem překvapující jsou možnosti customizace (přizpůsobení) vlastního IDE. Počínaje běžnou změnou fontu (velikost, barva) v jednotlivých typech oken, až po design rámců, menu položek, průvodců, panelu nástrojů atd. U průvodců customizace nekončí pouze změnou jejich vzhledu, nýbrž je možné vytvářet či modifikovat nové i stávající, a to buď určováním pořadí původních akcí, nebo za použití pokročilejších funkcí vytvoření nových automatizovaných činností. Ve výsledku je tedy možné

31 IntelliJ IDEA [online]. c2000-2009 [cit. 2009-05-04]. Dostupný z WWW: <<http://www.jetbrains.com/idea/>>.

za použití těchto customizací a případně dalších pluginů poměrně snadno rozšířit funkcionalitu a použití IDEI úplně novým směrem.

3.6 RubyMine

Z jazyků Ruby a Ruby on Rails se díky jejich vzrůstající popularitě stává téměř standardně používaný programovací jazyk. Proto se ve firmě JetBrains rozhodli vytvořit IDE zaměřené speciálně na tuto technologii. V roce 2008 byla veřejnosti představena beta verze RubyMine³², verze 0.x, nicméně toto vydání si nepřipravilo vhodnou půdu pro zaujetí veřejnosti. Hlavní výtkou byla především opravdu pomalá odezva programu na činnost programátora. Teprve před velmi krátkou dobou (duben 2009) byla vydána plnohodnotná verze IDE RubyMine verze 1.0, ve které byla většina nedostatků betaverze odstraněna.

RubyMine je podobně jako IDEA zdarma pro vývojáře open source projektů. Pro komerční účely je stanovena jednotná cena jak pro organizace, tak pro jednotlivce, a sice 90 USD na uživatele.

RubyMine je IDE postavené na JetBrains' IDEE a zahrnuje v sobě nástroje pro co nejefektivnější práci s jazykem Ruby a Ruby on Rails, dynamickou analýzu kódu a refactoring jak pro projekty v Ruby, tak pro webové aplikace na bázi Ruby on Rails. Pro vývoj webových aplikací je rovněž implementována editace HTML, CSS a JavaScriptu. Kromě této hlavní funkcionality je také přidáno: Version Control System (systém kontroly verzí s podporou SubVersion, CVS, Perforce a Git), Unit Test Framework (testování), Graphical Ruby Debugger, Model Dependency Diagram a rozhraní pro tvorbu vlastních pluginů pro Javu nebo Ruby.

Je zajímavé, že Model Dependency Diagram není založen na standardu UML, ale používá vlastní vizuální syntaxi. Dalším postřehem je již jen jedno negativum³³, kterým je stále slabá optimalizace výkonu, která byla tolik vyčítána betaverzi. Poměrně často, při otevření projektu a následné jeho editace, probíhá indexace, která dokáže výkon počítače velmi zpomalit (pozn. IDE bylo testováno v Mac OSX 10.5.6, CPU 2.16 GHz Intel Core 2 Duo, 2 GB 667 MHz DDR2 SDRAM).

3.7 Delphi

Prostředí Delphi³⁴ je na trhu aktuálně nabízeno ve verzi Delphi 2009. Z historického pohledu se jedná o produkt společnosti Borland, v současné době je tento produkt nicméně vyvíjen a prodáván dceřinou společností firmy Embarcadero Technologies, firmou CodeGear. Lokalizovanou nabídku včetně cen je možné najít přímo na webových stránkách českého zastoupení Embarcadero

32 JetBrains RubyMine [online]. c2000-2009 [cit. 2009-05-04]. Dostupný z WWW: <<http://www.jetbrains.com/ruby/index.html>>.

33 Pozn. autora: znalostí jazyka Ruby nebo Ruby on Rails mám naprosté minimum, tudíž jsem byl při zkoušení IDE poněkud omezen.

34 Delphi [online]. 1993-2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.embarcadero.com/products/delphi/>>.

Technologies³⁵.

Prostředí se zaměřuje výlučně na operační systém Microsoft Windows. Pro ostatní operační systémy je k dispozici pouze prostřednictvím emulace / virtualizace OS Microsoft Windows.

Prostředí Delphi je možno použít pro vývoj za využití programovacího jazyka Object Pascal. Je k dispozici i verze, která zároveň nabízí možnost psaní kódu v jazyce C/C++. Během návrhu grafického rozhraní je k dispozici RAD (Rapid Application Development), což je systém pro vizuální návrh grafického uživatelského rozhraní. Samotný grafický návrh se provádí „skladbou“ výsledného celku z komponent, které jsou k dispozici v okolních nabídkách. Jednotlivé komponenty se aplikují tažením myši, respektive chytnutím na zmiňované nabídce a přesunutím na cílové místo. Samotné komponenty nemusí být přímo vizuální (rámečky, tlačítka, textová pole a další), ale mohou být čistě funkční, například komponenta pro zajištění FTP spojení, spojení s databází a další podobné.

Během vývoje aplikací lze využívat již předpřipravených komponent. Systém umožňuje tvorbu vlastních, případně stažení a doinstalování dalších komerčních, či volně šiřitelných částí z prostředí sítě internet.

Nejlevnější cena jedné licence je 25 100 Kč. Tato cena je bez předplatného (bez přístupu k aktualizacím). V případě zahrnutí předplatného se cena navyšuje na 40 800 Kč. Chceme-li si vyzkoušet toto prostředí, zkušební verzi můžeme získat na webu výrobce³⁶.

Během vývoje produktové řady Delphi se autoři pokusili vytvořit i produkt pro prostředí OS Linux. Ten se nazýval Kylix. Ačkoli firma Borland značnou dobu podporovala vývoj tohoto produktu, podpora již zanikla a projekt byl ukončen. Poslední verze byla Kylix 3.

3.8 KDevelop

KDevelop³⁷ je integrovaným vývojovým prostředím primárně pro UNIXové systémy, konkrétně pro prostředí KDE. Dostupnou verzi lze provozovat i pod dalšími systémy, jako například Mac OS, nebo Microsoft Windows (s využitím software Cygwin³⁸).

Funkcionalita prostředí může být ovlivněna pomocí zásuvných modulů (pluginů). To nabízí uživatelům možnost přidávat nové funkce i bez zásahu do zdrojového kódu systému. Pro vývoj samotný nabízí podporu správy projektů s využitím automake nebo qmake (jedná se o systémy

35 CodeGear/DatabaseGear : Produkty : Delphi & C++Builder [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <<http://www.embt.cz/cs/produkty/19-delphi-c-builder>>.

36 Download a Trial or Free Version for Delphi [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <<http://downloads.embarcadero.com/free/delphi>>.

37 KDevelop – IDE – domovská stránka [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <<http://www.kdevelop.org/>>.

38 Cygwin je projekt, který nabízí soubor knihoven (soubory poskytující služby programům – nezbytné pro jejich správnou funkčnost) rozšiřující možnosti operačního systému Windows schopnost spouštět některé programy primárně určené pro operační systém Linux. (Cygwin [online]. [2008] [cit. 2009-05-01]. Dostupný z WWW: <<http://www.cygwin.com/>>.)

používané pro automatický překlad zdrojových kódů s minimální nutností zásahu uživatele).

Během vývoje software programátor jistě uvítá podporu verzovacích systémů (systémy napomáhající evidovat změny prováděné během vývoje a v případě zanesení chyb do zdrojových kódů nabízí možnost návratu do stavu, kdy byla provedena „nevhodná úprava“). KDevelop nabízí podporu všech běžně rozšířených systémů. Jmenovitě bychom mohli jmenovat CVS, Perforce, Subversion nebo Clearcase. Po promítnutí změn do zdrojového kódu nabízí barevné odlišení zjištěných změn od původního stavu. Nabízí samozřejmě i rozhraní (API) pro rozšíření podpory dalších systémů pro správu verzí. K dispozici jsou také šablony pro jednotlivé typy projektů – KDE applications, Kcontrol modules, KOffice parts, GNOME a další. Pro vytvoření dokumentace je k dispozici nástroj doxygen. Samozřejmostí u této kategorie produktů je RAD nástroj pro rychlý návrh grafického uživatelského rozhraní.

Samotné prostředí jako takové nedisponuje kompilátorem, ale využívá externí aplikace. Tento fakt ovšem není nijak limitující, neboť systém Linux jako takový je v podstatě na obdobném systému přístupu postaven – ne každá aplikace obsahuje vše, ale co jsou „obecné“ funkcionality, využívá externí „poskytovatele“ služby (označováno jako závislosti).

V současné době je k dispozici ve verzi 3.5.4-1. Jedná se o produkt distribuovaný pod licencí GPL, z čehož plyne nulová cena pořízení softwaru.

Tento produkt nabízí možnost programování s využitím jazyků Ada, Bash, C/C++, Fortran, Haskell, Java, Pascal, Perl, PHP, Python, Ruby, SQL, XUL (neoficiálně).

3.9 Microsoft Visual Studio

Posledním produktem, který se objevuje v tomto přehledu, je nástroj firmy Microsoft³⁹. Jedná se o komerční produkt. Vzhledem k výrobci asi není zářezující, že je určen právě pro prostředí Microsoft Windows, Windows Mobile, .NET, Microsoft Silverlight apod. Aktuální verzí je Microsoft Visual Studio 2008.

Samozřejmostí je podpora verzovacích systémů – pro příklad jsme vybrali Subversion a Visual SourceSafe. Co se do přístupu autorů k produktu týče, byl zvolen „modulový“ přístup. Tato volba nám přináší možnost rozšíření funkcionalit pomocí rozšiřujících balíčků – modulů. Dále se produkt dělí na:

- **Editor kódu** – V tomto pohledu nezbyvá než konstatovat schopnost zvýrazňování textu (syntaxe) a nabízení automatické dokončování (využívá se technologie IntelliSense). Další příjemnou funkcí je sbalování bloků kódu.

³⁹ Microsoft Visual Studio: *Přehled* [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <<http://www.microsoft.com/cze/msdn/produkty/vstudio/default.aspx>>. / Microsoft Visual Studio - Wikipedie [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Microsoft_Visual_Studio>.

- **Debugger** – Tato součást systému nabízí možnosti sledování provádění námi vytvářené aplikace. Můžeme v libovolném okamžiku sledovat stav proměnných, nebo krokovat provádění kódu.
- **GUI aplikace** – Základní nabídkou pro tvorbu rozhraní pro aplikace s grafickým uživatelským rozhraním jsou Winforms designer a WPF designer. V obou případech se jedná o grafický návrh – umístění prvku se provádí přímo tahem myši na své místo, kde daný prvek požadujeme v aplikaci.
- **WEB designer** - Ve spojení s ASP .NET nabízí možnost navrhovat webové formuláře. Samotný návrh probíhá opět myší. Visual Studio „popisuje“ požadovanou strukturu pomocí HTML, CSS a JavaScript.
- **Třídy a data** – Designer tříd umožňuje UML návrh jednotlivých tříd. Zároveň je chopen designer vygenerovat příslušné kódy z daného návrhu.
- **Mapování** – Designer mapování slouží pro zobrazení mapování mezi databázovými schématy.

Produkt je nabízen zdarma ve verzi Express edition, nebo v dalších licenčních balících, kde cena, v závislosti na zvoleném balíku, začíná na hodnotě 310 EUR. V případě zkušební verze (trial) je možné vyzkoušet produkt ve verzi Pro po dobu 90 dní.

Prostředí nabízí možnost programování v jazycích Visual Basic, C#, C++.

4 Shrnutí a porovnání platforem z hlediska nabídky nástrojů s podporou CASE

4.1 Porovnání IDE dle základních kritérií

Název	Výrobce	Licence	Cena	Aktuální verze ⁴⁰	Možnost demoverze	Lokalizace
Eclipse	Eclipse Foundation	EPL	-	3.4.2	-	Ne
NetBeans	Sun Microsystems	CDDL	-	6.5.1	-	Ne
ERIC IDE	Detlev Offenbach	GPL	-	4.3.2	-	Ano
Xcode	Apple	Komerční	-	3.1.2	Ne	Ne
IntelliJ Idea	JetBrains	Komerční (kromě open source SW vývoje pro nekomerční použití)	různá pro různé typy zákazníků / zdarma	8.1	Ano	Ne
RubyMine	JetBrains	Komerční / free pro open source projekty	90 EUR / zdarma	1.0	Pouze screencasty ⁴¹	Ne
Delphi	CodeGear (dceřinná společnost firmy Borland)	Komerční	25100,- / 40800 CZK (bez / s předplatným)	2009	Ano	Ano
KDevelop		GPL	-	3.5.4-1	-	Ano
Microsoft Visual Studio	Microsoft	Komerční	0 EUR (Express edition) / od 310 EUR	2008	Ano (Express edition)	Ano

Tabulka 1: Porovnání IDE dle základních kritérií

První ze základních kritérií pro porovnání IDE jsou znázorněny v tabulce 1. Jak je vidět, popisovaná vývojová prostředí jsou dostupná pod mnoha licencemi, přičemž poměr mezi open-source a komerčními je přibližně jedna ku jedné. Většina komerčních prostředí potom nabízí možnost vyzkoušení zdarma.

Problémem pro českého uživatele neznalého angličtiny (což, pravda, můžeme zvlášť u programátora považovat za výrazný nedostatek) může být fakt, že jen pět z devíti prostředí je

⁴⁰ V době psaní práce.

⁴¹ Vizuální záznam práce se softwarem.

lokalizováno.

Název	Windows	Linux	Mac OS	Další OS	Hlavní programovací jazyky
Eclipse	Ano	Ano	Ano	OS podporující JVM	Java
NetBeans	Ano	Ano	Ano	OS podporující JVM	Java SE, JavaFX, Java, Ruby, C/C++, PHP
ERIC IDE	Ano	Ano	Ano	OS podporující jazyk Python a knihovnu Qt	Python, Ruby
Xcode	Ne	Ne	Ano	Ne	C, C++, Fortran, Objective-C, Objective-C++, Java, AppleScript, Python, Ruby
IntelliJ Idea	Ano	Ne (lze zprovoznit)	Ano	Ne	Java, JavaScript/Flex, HTML/XHTML/CSS, XML/XSL, Ruby/JRuby, Groovy, SQL, FreeMarker/Velocity
RubyMine	Ano	Ano	Ano	Ne	Ruby, HTML/CSS, JavaScript
Delphi	Ano	Ne	Ne	Ne	Delphi (Objektový Pascal), nabízena varianta umožňující i C++
KDevelop	Ano (Cygwin)	Ano	Ano	Ne	Ada, Bash, C/C++, Fortran, Haskell, Java, Pascal, Perl, PHP, Python, Ruby, SQL
Microsoft Visual Studio	Ano	Ne	Ne	Ne	Visual Basic, C#, C++

Tabulka 2: Porovnání IDE dle dalších základních kritérií

Tabulka 2 se již věnuje podporovaným operačním systémům. Pět z popisovaných prostředí je možno provozovat na všech zkoumaných operačních systémech, v případě KDevelop je ovšem pro provoz mimo UNIXové systémy nutné ještě využít služeb již zmíněného software Cygwin.

Zároveň může být zajímavé, že pouze dvě prostředí jsou fixována jen na jeden operační systém (Xcode a Microsoft Visual Studio), přičemž v obou případech (což už takové překvapení není) jde o prostředí pocházející z dílny výrobce daného operačního systému.

Z hlediska podporovaných jazyků naprosto převládají nástroje umožňující vývoj v několika jazycích, přičemž nejzastoupenějšími jsou C, C++ a Java.

4.2 Porovnání prostředí dle podpory CASE

Název	Generování kódu	Automatické dokončování	Dynamická analýza	Podpora UML	Generování dokumentace	Refactoring
Eclipse	ano	ano	ano	ano	ano	ano
NetBeans	ano	ano	ano	ano	ano	ano
ERIC IDE	ne	ne	ne	ne	ne	ne
Xcode	ano	ano	ano	ne (vlastní grafická syntaxe)	ano	ano
IntelliJ Idea	ano	ano	ano	ano	ano	ano
RubyMine	ne	ano	ano	ne	ano	ne
Delphi	ne	ne	ne	ano	ano	ne
KDevelop	ne	ne	ne	ne	ano	ne
Microsoft Visual Studio	ano	ne	ne	ano	ne	ne

Tabulka 3: Porovnání podpory CASE v jednotlivých IDE

Z tabulky 3 vyplývá, že nelze nalézt konkrétní CASE nástroj, který by se dal považovat za standard mezi IDE. V některých případech může být absence určitého CASE nástroje zapříčiněna technologickou podstatou vývojového prostředí - pro některé typy programovacích jazyků zkrátka mohou nástroje ztrácet smysl. Navíc, pokud nějaké IDE nemá určitý CASE nástroj ve své „základní výbavě“, většinou není problematické tuto funkcionalitu implementovat skrz nějaký plugin. Obecně platí, že hojně používaná IDE disponují velkou zásobou CASE nástrojů už v základní verzi, právě díky velké komunitě uživatelů, kteří se z velké části podílejí na zdokonalování daného IDE od samého počátku.

4.3 Porovnání platform z hlediska nabídky nástrojů s podporou CASE

Jestliže máme porovnat platformy mezi sebou, můžeme říci, že je situace relativně vyrovnaná. Je to především díky tomu, že většina vývojových prostředí funguje na větším množství operačních systémů. Uživatel má tedy vcelku velkou možnost rozhodnout se ne podle operačního systému, ale podle dalších, pro programátora důležitějších, kritérií.

4.4 Shrnutí

Na trhu s vývojovými prostředími je velké množství nabízených produktů a je jasné, že jsou to právě CASE nástroje, které určují konkurenční výhodu daného produktu (IDE) nad ostatními. Samozřejmě první volbou uživatele při rozhodování, které IDE používat, je vybraná technologie a

potřebná funkcionalita, kterou musí IDE podporovat. V dnešní době je ovšem běžný stav, kdy jednotlivé IDE podporují více jazyků a technologií, a tak je možné na různorodé projekty používat jediný systém, případně obohacený o několik pluginů. Další rozhodující faktor však nebývá cena, jak je tomu u jiných produktů zvykem, ale míra pohodlí a komfortu práce s daným IDE, k jejíž zvýšení mohou velkou měrou přispět právě CASE nástroje.

Důležité je také zmínit možnost využití virtualizace. Současný výkon osobních počítačů již v podstatě bez problémů pomocí moderních virtualizačních prostředků (například VMware Player⁴² nebo VirtualBox⁴³) umožňuje provozovat kompletně virtualizovaný (hostovaný) operační systém uvnitř systému jiného (hostitele), a to při zachování plného uživatelského komfortu. Nezáleží tedy už tolik na tom, jaký operační systém používáme jako „hlavní“ systém – lze jednoduše a s vcelku vysokou úrovní integrace provozovat i potřebné vývojové nástroje primárně určené pro systém jiný⁴⁴.

42 VMware Player [online]. c2009 [cit. 2009-05-05]. Dostupný z WWW: <<http://www.vmware.com/products/player/>>.

43 VirtualBox [online]. c2009 [cit. 2009-05-05]. Dostupný z WWW: <<http://www.virtualbox.org/>>.

44 Pochopitelně je ale však stále potřeba zaplatit příslušné licenční poplatky.

Závěr

V naší semestrální práci jsme se snažili o pohled na vývojové nástroje a jejich návaznost na CASE z hlediska platformy.

Jako platformu jsme v našem případě z rozličných důvodů (naš zájem a zkušenosti, možnost úspor, snaha odlišný přístup k tématu, možnost začlenění do existující podnikové infrastruktury) zvolili operační systém. Jako vývojové nástroje jsme potom určili integrovaná vývojová prostředí.

Dále jsme se pokusili o stanovení základních kritérií pro porovnání těchto prostředí, a dále o popis vybraných prostředí se zaměřením na vlastní zkušenosti.

Při závěrečném shrnutí a srovnání jsme narazili na pozitivní zjištění, že je nabídka vývojových nástrojů s podporou CASE napříč platformami pestrá, a to i z hlediska licenčního, z hlediska podpory programovacích jazyků a dalších.

Věříme, že má tato práce smysl a najde svého čtenáře. Mohla by například sloužit jako základní podklad pro výběr vývojového prostředí, nebo naopak operačního systému. Každopádně bychom případným zájemcům o tuto problematiku rádi doporučili i ostatní seminární práce vytvořené v rámci předmětu 4IT450⁴⁵.

45 Práce vzniklé v rámci semináře CASE na KIT VŠE [online]. [2008] [cit. 2009-05-05]. Dostupný z WWW: <<http://panrepa.org/CASE/>>.

Použité zdroje

- Apple [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.apple.com/>>.
- Apple - Mac OS X Leopard - Technology - UNIX [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.apple.com/macosx/technology/unix.html>>.
- Aptana [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.aptana.com/>>.
- CodeGear/DatabaseGear : Produkty : Delphi & C++Builder [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <<http://www.embt.cz/cs/produkty/19-delphi-c-builder>>.
- Computer-aided software engineering - Wikipedia, the free encyclopedia [online]. 2009 [cit. 2009-05-05]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Computer-aided_software_engineering>.
- Cygwin [online]. [2008] [cit. 2009-05-01]. Dostupný z WWW: <<http://www.cygwin.com/>>.
- Debian [online]. 2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.debian.org/>>.
- Delphi [online]. 1993-2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.embarcadero.com/products/delphi/>>.
- DistroWatch.com [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://distrowatch.com/dwres.php?resource=major>>.
- Download a Trial or Free Version for Delphi [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <<http://downloads.embarcadero.com/free/delphi>>.
- Eclipse Downloads [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.eclipse.org/downloads/>>.
- Eclipse Plugin Central :: Eclipse plugins resource center and marketplace for Eclipse and Plugin Ecosystem [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.eclipseplugincentral.com/>>.
- Eclipse Modeling Project [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.eclipse.org/modeling/>>.
- Eclipse Public License [online]. [2004] [cit. 2009-05-03]. Dostupný z WWW: <<http://www.eclipse.org/legal/epl-v10.html>>.
- Eclipse Public License - Wikipedia, the free encyclopedia [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Eclipse_Public_License>.

- Eclipse.org home [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: [<http://www.eclipse.org/>](http://www.eclipse.org/).
- Fedora Project [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: [<http://fedoraproject.org/>](http://fedoraproject.org/).
- GNU General Public License Version 2 [online]. 1991 [cit. 2009-05-03]. Dostupný z WWW: <http://www.gnu.org/licenses/old-licenses/gpl-2.0.txt>>.
- GNU General Public License Version 3 [online]. 2007 [cit. 2009-05-03]. Dostupný z WWW: <http://www.gnu.org/licenses/gpl-3.0.txt>>.
- IntelliJ IDEA [online]. c2000-2009 [cit. 2009-05-04]. Dostupný z WWW: <http://www.jetbrains.com/idea/>>.
- JetBrains RubyMine [online]. c2000-2009 [cit. 2009-05-04]. Dostupný z WWW: <http://www.jetbrains.com/ruby/index.html>>.
- KDevelop – IDE – domovská stránka [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <http://www.kdevelop.org/>>.
- Mandriva [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <http://www2.mandriva.com/>>.
- Microsoft Visual Studio: Přehled [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: <http://www.microsoft.com/cze/msdn/produkty/vstudio/default.mspx>>.
- Microsoft Visual Studio - Wikipedie [online]. [2009] [cit. 2009-05-01]. Dostupný z WWW: http://cs.wikipedia.org/wiki/Microsoft_Visual_Studio>.
- NetBeans - Wikipedia, the free encyclopedia [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: http://en.wikipedia.org/wiki/Netbeans#Early_history>.
- NetBeans IDE - Unified Modeling Language (UML) Plugin [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://www.netbeans.org/features/uml/>>.
- NOURIE, Dana. Getting Started with an Integrated Development Environment (IDE) [online]. 2005 [cit. 2009-05-03]. Dostupný z WWW: <http://java.sun.com/developer/technicalArticles/tools/intro.html>>.
- OpenSUSE.org [online]. 2008 [cit. 2009-05-03]. Dostupný z WWW: <http://cs.opensuse.org/>>.
- Plugin Portal Main Page [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://plugins.netbeans.org/PluginPortal/>>.

- Práce vzniklé v rámci semináře CASE na KIT VŠE [online]. [2008] [cit. 2009-05-05]. Dostupný z WWW: <<http://panrepa.org/CASE/>>.
- Pydev [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://pydev.sourceforge.net/>>.
- Qt - A cross-platform application and UI framework [online]. c2008-2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.qtsoftware.com/products/>>.
- SourceForge.net: Eric Integrated Development Environment: Files [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://sourceforge.net/project/showfiles.php?group_id=119070&package_id=233329>.
- SourceForge.net: Eric Integrated Development Environment: Files [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <http://sourceforge.net/project/showfiles.php?group_id=119070&package_id=249666&release_id=548540>.
- The Eric Python IDE [online]. 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://eric-ide.python-projects.org/>>.
- The Linux Kernel Archives [online]. [2009] [cit. 2009-05-03]. Dostupný z WWW: <<http://www.kernel.org/>>.
- Ubuntu [online]. c2009 [cit. 2009-05-03]. Dostupný z WWW: <<http://www.ubuntu.com/>>.
- VirtualBox [online]. c2009 [cit. 2009-05-05]. Dostupný z WWW: <<http://www.virtualbox.org/>>.
- VMware Player [online]. c2009 [cit. 2009-05-05]. Dostupný z WWW: <<http://www.vmware.com/products/player/>>.
- What is a CASE Environment? [online]. c2007 [cit. 2009-05-05]. Dostupný z WWW: <http://www.sei.cmu.edu/legacy/case/case_what.html>.
- Xcode [online]. c2009 [cit. 2009-05-04]. Dostupný z WWW: <<http://developer.apple.com/tools/xcode/index.html>>.
- Zend Studio - The Leading PHP IDE from the PHP Experts – Zend.com . 2009 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.zend.com/products/studio/>>.