

.NET, stručná informace

Josef Vochozka, FI MU

Stále častěji se v titulech knih na pultech knihkupectví zaměřených na informační technologie resp. na stránkách IT periodik objevují zmínky či přímo bouřlivé polemiky o platformě .NET. Tento článek si klade za cíl velice stručně informovat o této platformě a případnému zájemci poskytnout odkazy na další informační zdroje.

.NET (dot net) je platforma vyvinutá firmou Microsoft. Cílem platformy je zjednodušit tvorbu programových aplikací a současně zpřehlednit jejich zdrojové kódy. V neposlední řadě pak zjednodušit správu softwarových instalací na konkrétním počítači. Současně by tato platforma měla posílit vnitřní i vnější bezpečnost SW systému jako takového. Nakolik Microsoft sází na tuto platformu ilustruje i fakt, že většina vývojářských nástrojů nabízených na webových stránkách Microsoftu je nabízena již pouze pro platformu .NET.

Při úvahách o platformě .NET je třeba mít na mysli i její mohutnou podporu ze strany Microsoftu, která spočívá mimo jiné i v množství dostupné literatury i tutoriálů. Navíc je možno získat kompletní vývojové prostředí i se školením v rámci programů podporujících zavedení této platformy. Díky komplexnosti vývojového prostředí vývojářské firmy přechází nebo už přešly na .NET. Všechny potřebné vrstvy této platformy (popíšeme je dále v textu) jsou vestavěnou součástí Windows 2003 a vyšších verzí tohoto systému.

1 Nástin struktury .NETu

Základní myšlenka .NETu není nová. Vybudovat prostředí, které bude obsahovat knihovny, společné datové typy, podporu bezpečnosti a komunikací, konektivitu s databázemi - a navíc k tomu všemu bude ještě nezávislé na operačním systému, se pokoušelo a pokouší více firem a iniciativ. .NET tedy není ani první takou snahou a na základě obecných historických zkušeností můžeme říci, že není ani snahou poslední.

Zevrubný popis struktury platformy .NET rozhodně vyžaduje rozsáhlejší text než jen krátký

článek ve Zpravodaji. Návštěvník knihkupectví snadno zjistí, kolik samostatných titulů mající ve svém názvu slovo .NET je dnes běžně nabízeno. V našem informativním textu se budeme velice stručně zabývat pouze dvěma nejzákladnějšími vrstvami - první z nich je Common Language layer (CL) a druhou .NET class library.

Jazyk CL je implementace jazyka, do kterého jsou kompilovány zdrojové texty jednotlivých programů. CL je interpretován pomocí runtime systému CL. Deklarace CL jazyka se v dubnu 2000 stala z popudu Microsoftu, Hewlet Packardu a Intelu normou ECMA¹, čímž by se platforma .NET měla stát méně závislou na Microsoftu.

Autoři jazyka CL snad ani nepředpokládají, že by programátoři užívali jazyk CL přímo pro psaní zdrojového kódu (i když by to pravděpodobně bylo možné). Jazyk je primárně zamýšlen jako souhrn všech povolených operací, které může běžná aplikace potřebovat. Autoři jazyka CL vycházejí z předpokladu, že běžná aplikace nepotřebuje například formátovat disk nebo nepotřebuje nízkoúrovňové V/V operace, ale s velkou pravděpodobností bude tato aplikace potřebovat nástroje pro své zabezpečení (například šifrovací algoritmy), správu paměti, ošetření výjimek nebo například podporu svého ladění. Aplikace v .NETu by tak neměly mít přístup k těm funkcím, které mohou poškodit data jiných aplikací nebo aplikace samotné, ale na druhé straně by měly implicitně obsahovat vlastnosti zajišťující jejich vyšší bezpečnost a stabilitu.

Class library je knihovnou objektových tříd. Tyto knihovny jsou opět sdíleny všemi podporovanými jazyky. Pro snazší orientaci v knihovnách je pro volání jednotlivých objektů a jejich metod užito konceptu jmenných prostorů, jak jsou známy z xml. Zajímavý je i fakt, že všechny knihovny jsou sdíleny i mezi jednotlivými programovacími jazyky.

¹<http://www.ecma-international.org/> Mezinárodní průmyslová asociace založená v roce 1961 a zaměřená na standardizaci v informatice a komunikačních systémech. Na stránkách ECMA lze nalézt mj. i poměrně rozsáhlou prezentaci struktury CL

2 Jazyky podporující .NET

Jazyk CL, který jsme velice stručně charakterizovali v předchozím odstavci, není vyšším programovacím jazykem v pravém slova smyslu (je bližší assembleru). Zdrojové texty programů jsou psány v jazycích, které CL podporují a následně jsou do CL kompilovány. Vzhledem k tomu, že se deklarace jazyka CL stala normou ECMA, lze předpokládat, že se budou objevovat překladače do CL pro stále větší spektrum programovacích jazyků, podobně i jako implementace runtime systému CL pro další operační systémy. Jazyky podporující v současnosti .NET můžeme rozdělit na dvě základní skupiny: První z nich tvoří jazyky dodávané přímo firmou Microsoft - C, C++, C#² a Visual Basic. Druhou skupinou jsou jazyky (přesněji jejich kompilátory), které jsou dodávané třetími stranami a které kompilují do CL místo do proveditelného kódu počítače - například Python, Eiffel, Fortran. V materiálech Microsoftu se hovoří až o několika desítkách takových jazyků. Některé kompilátory, například překladač jazyka C#, jsou dodávány jak Microsoftem, tak i třetími stranami.

Protože všechny zmíněné jazyky překládají zdrojový text do jazyka CL, je zřejmé, že možnosti jednotlivých jazyků jsou omezeny možnostmi jazyka CL. Pokud chceme pracovat na vývoji konkrétní aplikace a chceme zvolit nejvhodnější jazyk, který by nejlépe vyhovoval potřebám projektu, pak je technicky zcela jedno, který jazyk zvolíme. Vlastnosti jazyků se liší v rámci vlastností CL pouze svou syntaxí a vlastnostmi vyplývajícími z definice konkrétního programovacího jazyka. V .NETu jsou si všechny jazyky rovny. Pouze jazyk C++ je výjimkou, protože je zamýšlen jako platforma pro vývoj aplikací, které vyžadují operace neposkytované jazykem CL (přímé zápisy do paměti, nízkourovňové v/v operace, ...).

3 .NET v praxi

Velice zjednodušeně můžeme práci na platformě .NET popsat následovně: Vývojář má k dispozici

²C# [sí šárp] jazyk vybudovaný na základě C++, podle některých autorů nová Java. Více informací <http://msdn.microsoft.com/vcsharp/> případně na <http://java.about.com/cs/antijava/>.

paletu programovacích jazyků, které mají v podstatě stejné vlastnosti (vyjma vlastností vyplývajících z definice jazyka, například jeho syntaxe, ...). Všechny jazyky používají stejné knihovny, stejné datové typy (pod různými synonymy), velice podobné nebo stejné prostředí pro vývoj. Zdrojový kód je kompilován překladačem zvoleného jazyka do CL. Následně je tento program interpretován pomocí runtime CL (na rozdíl od kompilace přímo do spustitelného kódu pro daný operační systém). Pro celkové zrychlení aplikace je v některých případech možnost využít i předkompilaci přímo do spustitelného kódu. Vzhledem k uložení informací o programu společně s kompilovaným kódem programu odpadá nutnost registrace programu do databáze registrů (platforma MS Windows). Instalace programu pak spočívá pouze v prostém přepírování adresáře na disk cílového počítače. Zachovány zůstávají i přípony spustitelných programů a knihoven (.exe a .dll). Před instalací prvního programu využívající runtime CL je třeba tento runtime systém na počítači nainstalovat. Instalace runtime jsou volně dostupné na stránkách Microsoftu. Z hlediska uživatele se s přechodem na novou platformu nic nemění.

3.1 Iniciativa Rotor

Rotor je aktivita, kterou lze označit za akademickou verzí platformy .NET. Rotor používá deklaraci CL jazyka pro svůj runtime CL podle standardu ECMA. Podstatné části .NETu jsou v rámci iniciativy Rotor již přeneseny pod Linux a Free BSD Unix. Pro tyto části funkcionality jsou přenositelné exe a dll soubory mezi systémy Windows, Linux a Free BSD Unix. Zájemce může vidět zdrojové texty jednotlivých objektů objektových knihoven například na stránkách <http://www.123aspx.com/rotor/default.aspx>. Na těchto stránkách je rovněž dobře patrné využití jmenných prostorů pro strukturování knihoven.

3.2 Iniciativa Mono

Mono je iniciativa společnosti Ximian (<http://www.ximian.com/>) směřující k vytvoření otevřené Linuxové implementace .NET platformy. Tento projekt ale zužuje počet podporovaných

jazyků pouze na překladač C#; jeho cílem je současně i implementace kompletních objektových knihoven. Projekt by měl umožnit vývojářům vytvářet aplikace pro .NET a spouštět je pod Windows nebo pod jakoukoli platformu, kterou podporuje Mono projekt (Linux a Unix). Více informací lze nalézt například na stránkách <http://developer.ximian.com/projects/mono/>.

4 Závěr

.NET je rozhodně velice ambiciózní záměr. O jeho úspěchu či neúspěchu rozhodne mimo jiné i to, jak velkou skupinu softwarových vývojářů osloví a jestli bude natolik flexibilní, aby pokrýval nejen současné, ale i nově vznikající potřeby vývojářů a uživatelů. Čtenáři, který by se chtěl seznámit přímo s jednotlivými vývojářskými nástroji nebo s rozsáhlou dokumentací platformy .NET, lze kromě uvedených odkazů na stránky projektů Mono a Rotor doporučit i stránky <http://www.microsoft.com/net/>. Na základě prvních zkušeností s touto platformou lze s jistou dávkou opatrnosti říct, že se rozhodně vyplatí věnovat jí pozornost. □