

Oktany pro váš počítač

Programovací nástroj Delphi společnosti Borland procházel v celé své vývojové linii, tzn. od verze 1 až po verzi 7, mnoha změnami. Žádné změny však nebyly tak radikální jako změny v aktuálně představené "oktanové" verzi.

ADO.NET, MS .NET Framework, ASP.NET, C#... Který programátor či vývojář by se v poslední době neseťkal - ať již na internetu, na pultech prodejen knih nebo v reálu na počítači - s některým z těchto pojmů? Z technologií programování v sobě zahrnují asi to nejlepší, co v současnosti existuje (však se také na internetu objevil článek s podezřením, že .Net je produktem firmy Borland...).

Ve své podstatě je celý systém .NET skutečně impozantní a prospěšnou myšlenkou. Do systému zapadají i nové programovací jazyky C#, VB a C++, nově i J#. Ale právě zde je určité slabé místo celé myšlenky. Zatímco C# je jazykem zcela novým, integrace technologií .NET do dalších jazyků vede k jejich natolik podstatným změnám, že prakticky narušuje kontinuitu kompatibility s předchozími verzemi. Portovat starší projekty pod .NET v těchto jazycích je velmi problematické. Zní to možná jako laciná reklama, ale skutečností je, že na takové problémy můžete v případě Delphi zapomenout.

JSOU VĚCI, KTERÉ VYŘEŠÍ ZA VÁS...

První a nejpodstatnější skutečností, kterou je třeba si uvědomit, je fakt, že verze Delphi 8 je určena jako vývojové prostředí nad systémem Microsoft .NET Framework. Vygenerovaný kód je mezikódem (IL), určeným pro zpracování v tomto prostředí. Delphi 8 tedy již nevytváří klasické programy pro Windows, ale umožňuje vám beze změn, nebo jen s minimálními změnami, přenést dříve vytvořené aplikace (pro Win32) na platformu .NET. Množství úprav, které budete muset při převodu provést, závisí na různých okolnostech, jako je např. přímé využití API, "vzdálenost" od verze 8 nebo použití komponent třetích stran (pro zajímavost - na poslední konferenci firmy Borland byla bez jakékoliv změny kompilována a spuštěna na platformě Delphi for .NET demonstrační aplikace, která byla součástí 16bitové Delphi 1!).

Druhým důležitým faktorem je to, že Delphi v dnešní podobě není jen obyčejným vývojovým prostředím, které vám umožní zapsat a přeložit zdrojový kód. Je to systém, který vám interaktivně pomůže od formulace požadavků a analýzy problému (CaliberRM) přes model a návrh struktury programu a dat pomocí standardů UML (ECO) až po vlastní zápis programu, včetně všech dnes obvyklých vymožeností automatického doplňování kódu. Navíc máte k dispozici knihovnu typů či vzorů úseků programu, kterou můžete sami rozšiřovat. Na překlad pak navazuje, kromě již klasických ladicích metod, nástroj pro otestování a vyhledání úzkých míst v programu (Optimizeit Profiler) a nakonec nástroje pro distribuci (k dispozici je opět známý InstallShield Express) a údržbu programu. Většina výše zmíněných nástrojů je při návrhu samozřejmě obousměrná, například změny provedené do modelu se promítnou ihned do zdrojového kódu a naopak.

Pro jednotlivé výše zmíněné etapy existují alternativní nástroje, plnicí obdobné funkce, ale integrace těchto nástrojů do jednotného vývojového řetězce s jednotným designem a ovládáním je v takovémto rozsahu unikátní. Navíc celý výše zmíněný proces podporuje týmovou spolupráci. K dispozici je Microsoft Common Source Code Control API (SCC API), propojující StarTeam, CVS, ClearCase a Visual SourceSafe. Dosažitelnost jednotlivých částí tohoto vývojového řetězce nástrojů pochopitelně závisí i na verzi zakoupeného produktu, který přichází v obvyklé trojici Professional, Enterprise a nejvyšší verzi Architect.

VÝVOJOVÉ PROSTŘEDÍ DELPHI A JEHO SOUVISLOST S .NET

Pokud jste pracovali v prostředí Visual Studio .NET, jistě objevíte mnoho paralel ve vzhledu a uspořádání vývojových prostředí Visual Studio IDE Delphi 8. Jestliže jste naopak pracovali v prostředí Delphi, asi si budete muset chvíli zvykat na nový vzhled (viz obr. nahoře - všimněte si, jak jsou umístěny nevizuální komponenty, což je specialita aplikací nad .NET).

IDE si ale samozřejmě můžete opět konfigurovat podle svých představ. Nejdůležitější změna (především co se týče vzhledu) je v paletě komponent. Ta je označena jako paleta nástrojů a je nyní proměnlivá podle toho, co v daném okamžiku v prostředí provádíte. Budete-li psát kód, budete mít v tomto okně k dispozici typy a úryvky kódu. Pokud budete navrhovat formulář, bude v tomto okamžiku okně nabídka komponent. Nabídka komponent se bude dále lišit podle druhu aplikace, kterou budete vytvářet. Na obrázku vpravo vidíte, z kterých základních typů aplikací a vzorů lze vybírat. Nabídka komponent na paletě nástrojů se pak liší například pro aplikaci ASP a aplikaci VCL Forms. Výběrem typu

aplikace (Windows Forms, Web Forms, VCL Forms...) současně určujete, zda bude aplikace vytvářena nad množinou komponent odpovídajících původním VCL-komponentám Delphi, nebo nad komponentami prostředí .NET. Výsledný program bude samozřejmě vždy určen pro prostředí .NET. Součástí Visual Studia .NET je nástroj PEVerify.exe, kterým si můžete tuto skutečnost ověřit.

Navrhujete-li ASP-aplikaci, zobrazuje se trojice záložek se stránkou asp, kódem na pozadí a formulářem. Přizpůsobí se rovněž paleta komponent, takže obsahuje jen komponenty přípustné pro daný typ aplikace. Té se v tomto okamžiku přizpůsobí i systém code insight. Po přesunu kurzoru do závorek tagu na stránce asp-kódu se objeví v okně code insight všechny přípustné parametry tagu. Po umístění webkomponenty na záložce design se v objectinspektoru objeví množina vlastností a událostí obdobně jako při práci s komponentami na klasickém formuláři. Poklepáním na řádek v záložce Events se vygeneruje na stránce s kódem na pozadí kostra procedury pro obsluhu příslušné události. Doba vytvoření stránky asp je tak redukována na minimum.

Na spodním okraji editačního okna jsou tlačítka, kterými můžete nahrávat posloupnost stisků kláves a uložit ji ve formě makra pro pozdější použití.

JAZYK A EDITACE KÓDU

Stejně jako ve Visual Studiu .NET dochází v editoru ke skrývání kódu. Kód automaticky vygenerovaný editorem a označený dvojicí direktiv {\$REGION} a {\$ENDREGION} i jednotlivé procedury a funkce jsou označeny znaky +/- a poklepáním lze odpovídající bloky skrýt nebo zobrazit. Novou direktivou je rovněž {\$UNSAFECODE ON} (resp. OFF). Ta umožňuje nebo zakazuje psát v odpovídajícím bloku neřízený kód (pokud použijete neřízený kód, neprojde výsledný program testem programu PEVerify.exe). Podstatným způsobem je zjednodušena práce s webovými službami.

Základem pro přístup k datům je v .NET knihovna tříd ADO.NET. To se pochopitelně projevilo i v rozšíření způsobů přístupů k datům v Delphi 8. Samozřejmě zůstaly zachovány stávající způsoby a komponenty pro přístup k datům prostřednictvím BDE. Nově je k dispozici BDP.NET provider, zajišťující přístup k datům pod .NET.

V rámci něho je to přístup prostřednictvím TADONETConnectoru (jako potomka komponenty DataSet). Dále je to .NET verze komponent dbExpress a komponent IBX pro přístup k InterBase a nakonec je to DataSnap.NET založený na modelu DCOM. V rámci IDE lze zobrazit okno Model View. V něm můžete vidět strukturu projektu a poklepáním na prvek struktury se namísto editačního okna objeví plocha s UML diagramem odpovídajícího objektu (viz obr. uprostřed). Do diagramu lze doplňovat další prvky z palety a změny se zpětně promítají do kódu programu. Podobně jako model kódu fungují i datové diagramy. V případě struktur dat jde flexibilita až do takových detailů, že po úpravě struktury tabulky (přidání či ubrání polí, změně formátu pole apod.) se změna promítne do již existující databáze.

VYŠLAPANOU CESTOU

Pokud jste programovali a vytvářeli projekty v Delphi a nyní se chystáte přejít na platformu .NET, máte proti jiným prostředkům cestu podstatně ulehčenu jednak díky zachování kompatibility, jednak díky zapouzdření toho, co poskytuje .NET, do objektů Delphi. Tak lze navrhovat a programovat aplikace takřka stejně jako ve verzích pro klasické Windows. Neznamená to, že byste nemuseli vědět nic o .NET, ale přechod na tuto platformu je mnohem plynulejší. Významná je i skutečnost, že díky kumulaci nástrojů podporujících celý proces vývoje aplikace se doba vývoje mnohonásobně zkrátí.

Zcela bezbolestný ale přechod na novou verzi není. Jedním ze slabších míst je použití komponent třetích stran. Problém jejich použití sice nastává s každou novou verzí, ale je zřejmé, že předpokladem masového vývoje komponent třetích stran pro verzi 8 je masové rozšíření platformy .NET. A to samozřejmě nebude tak rychlé jako v případě dřívějších verzí Delphi.

To je ovšem z hlediska toho, co v tomto systému získáváte, zcela zanedbatelné. V Delphi 8 máte k dispozici unikátní nástroj, jenž vám bezproblémově pomůže přejít k nové platformě tempem, kterým můžete snadno získat náskok před konkurencí. Prostě ty oktany jsou, stejně jako v případě benzínu, vidět...

Jiří Ventluka

POPIS PRODUKTU

Borland Delphi 8 Architect - předběžná verze

Vývojářský produkt pro systém Microsoft .NET Framework

Minimální systémové požadavky Pentium II, 450 MHz, MS Windows XP/ 2000, 128 MB RAM, 700 MB diskového prostoru

Zapůjčil Borland, s. r. o., Praha

Předpokládaná cena cca 16 100 Kč bez DPH

