

# Obsah

---

<b>Úvod</b>	<b>7</b>
Předchozí znalosti	7
Tato kniha	8
Systém Windows a programování	8
Požadavky na systém	10
Předběžný software	11
Ukázky kódu	11
Podpora této knihy	11
Otázky a komentáře	11
Poznámka redakce českého vydání	12
Web autora	12
Zvláštní poděkování	12

## Část I

### Kód

---

<b>1</b>	<b>Aplikace a okno</b>	<b>15</b>
<b>2</b>	<b>Základní štětce</b>	<b>33</b>
<b>3</b>	<b>Koncepce obsahu</b>	<b>53</b>
<b>4</b>	<b>Tlačítka a další ovládací prvky</b>	<b>71</b>
<b>5</b>	<b>DockPanel a StackPanel</b>	<b>93</b>
<b>6</b>	<b>Ukotvení a mřížka</b>	<b>109</b>
<b>7</b>	<b>Panel Canvas</b>	<b>133</b>
<b>8</b>	<b>Vlastnosti závislosti</b>	<b>143</b>
<b>9</b>	<b>Směřované vstupní události</b>	<b>157</b>
<b>10</b>	<b>Vlastní prvky</b>	<b>183</b>
<b>11</b>	<b>Prvky s jedním podřízeným prvkem</b>	<b>201</b>
<b>12</b>	<b>Vlastní panely</b>	<b>231</b>
<b>13</b>	<b>Ovládací prvek ListBox</b>	<b>253</b>
<b>14</b>	<b>Hierarchie nabídek</b>	<b>283</b>
<b>15</b>	<b>Panely nástrojů a stavové řádky</b>	<b>309</b>
<b>16</b>	<b>Ovládací prvky TreeView a ListView</b>	<b>333</b>
<b>17</b>	<b>Tisk a dialogová okna</b>	<b>363</b>
<b>18</b>	<b>Program Notepad Clone</b>	<b>399</b>

**Část II**  
**Markup**

<b>19</b>	<b>Jazyk XAML</b>	<b>441</b>
<b>20</b>	<b>Vlastnosti a atributy</b>	<b>469</b>
<b>21</b>	<b>Prostředky</b>	<b>503</b>
<b>22</b>	<b>Okna, stránky a navigace</b>	<b>523</b>
<b>23</b>	<b>Datové vazby</b>	<b>579</b>
<b>24</b>	<b>Styly</b>	<b>609</b>
<b>25</b>	<b>Šablony</b>	<b>631</b>
<b>26</b>	<b>Vstupy dat, pohledy na data</b>	<b>681</b>
<b>27</b>	<b>Grafické objekty</b>	<b>717</b>
<b>28</b>	<b>Geometrie a cesty</b>	<b>739</b>
<b>29</b>	<b>Grafické transformace</b>	<b>773</b>
<b>30</b>	<b>Animace</b>	<b>809</b>
<b>31</b>	<b>Bitmapy, štětce a kreslení</b>	<b>881</b>

<b>Rejstřík</b>	<b>917</b>
-----------------	------------

*Při vývoji moderních digitálních počítačů se předpokládalo, že tato zařízení usnadní a urychlí složité a časově náročné výpočty. Dnes se při nasazení počítačů většinou využívá jejich schopnost ukládat velké objemy informací a přistupovat k nim a tato vlastnost se považuje za jejich základní rys. Oproti tomu jejich matematické funkce, tj. schopnost provádět aritmetické výpočty, v mnoha případech téměř ztrácejí význam.*

*Niklaus Wirth, Algorithms + Data Structures = Programs (1976)*

*jdu v cestu dálnou...  
v Aviljonské údolí,  
kde žádných není dešťů, sněhu, krup,  
a nikdy vítr hlučně nevane.  
Však nivy hluboké a překrásné  
tam jsou a štěpných stromů zahrady  
a doly stinné s letním mořem kol –  
tam zhojím já se z rány bolestné...*

*Tennyson, Idyly královské (Idylls of the King), přeložil F. Krsek*



# Úvod

---

V této knize se naučíte používat grafický subsystém Microsoft Windows Presentation Foundation (WPF) při psaní programů, které fungují v systému Microsoft Windows. Může se jednat buď o běžné samostatné aplikace systému Windows (které se v současnosti často označují jako *klientské* aplikace), nebo o uživatelská rozhraní distribuovaných aplikací. WPF se považuje za primární rozhraní API (application programming interface) systému Microsoft Windows Vista, ale aplikace WPF můžete spouštět i v systému Microsoft Windows XP s aktualizací Service Pack 2 nebo v systému Windows Server 2003, když nainstalujete platformu Microsoft .NET Framework 3.0.

Pomocí rozhraní WPF se sice píše programy, které se někdy označují jako „běžné aplikace systému Windows“, avšak rozhodně nejde o staré známé programy pro Windows. Grafický subsystém WPF poskytuje nový vzhled, nové principy přizpůsobení ovládacích prvků, nové grafické funkce (včetně animace a 3D) a nové programovací rozhraní.

WPF v praxi obsahuje *dvě* související programovací rozhraní. Programy WPF lze psát kompletně pomocí jazyka C# nebo jiného programovacího jazyka, který vyhovuje specifikaci CLS (Common Language Specification) platformy .NET. WPF kromě toho zahrnuje zajímavý nový značkovací jazyk založený na XML, který se nazývá XAML (Extensible Application Markup Language). V některých případech lze v jazyku XAML vytvořit celé programy. Obecně se však aplikace skládají z kódu i značek. Pomocí jazyka XAML můžete definovat uživatelské rozhraní a grafické prvky aplikací (včetně grafiky a animace) a v procedurálním jazyku vytvořit kód k obsluze událostí uživatelského vstupu.

## Předchozí znalosti

Tato kniha předpokládá, že již máte zkušenosti s programovacím jazykem C# a předchozími verzemi platformy .NET Framework. V opačném případě si přečtěte mou krátkou knihu s názvem *.NET Book Zero: What the C or C++ Programmer Needs to Know about C# and the .NET Framework* (Úvodní příručka platformy .NET: Co by měl programátor v C nebo C++ vědět o jazyku C# a platformě .NET Framework). Kniha je k dispozici zdarma a můžete si ji přečíst nebo stáhnout na následující stránce mého webu:

<http://www.charlespetzold.com/dotnet>

Pokud s programováním začínáte, doporučuji vám, abyste se s jazykem C# nejdříve seznámili při psaní konzolových programů, což jsou programy ve znakovém režimu, které lze spouštět v okně příkazového řádku. Tento přístup jsem zvolil ve své knize *Programming in the Key of C#: A Primer for Aspiring Programmers* (Programování v tónině C#: Úvod pro začínající programátory; Microsoft Press, 2003).

## Tato kniha

Programy pro systém Windows vytvářím už od roku 1985 a grafický subsystém WPF znamená nejzásadnější pokrok v programování pro Windows, jaký jsem během své kariéry zažil. Vzhledem k tomu, že podporuje dvě velmi odlišná programovací rozhraní, však subsystém WPF zároveň značně komplikoval psaní této knihy. Po mnoha úvahách nad koncepcí knihy jsem se rozhodl, že každý programátor ve WPF musí mít solidní základy manuálního programování. Z tohoto důvodu se v první části této knihy dozvíte, jak psát kompletní programy WPF v jazyku C#.

Některé funkce WPF vyžadují rozšíření vlastností a událostí platformy .NET. Je důležité těmto rozšířením porozumět, zejména při práci s jazykem XAML. Proto jsem některé kapitoly první části vyhradil novým koncepcím vlastností závislosti a směřovaných vstupních událostí.

Druhá část této knihy se zaměřuje na jazyk XAML. Ukážeme si, jak lze vytvářet malé aplikace výhradně v jazyku XAML a také jak lze kód XAML kombinovat s kódem C# při psaní větších a komplikovanějších aplikací. Jednou z prvních úloh druhé části je vytvoření programátorského nástroje s názvem XAML Cruncher, který mi učení jazyka XAML značně usnadnil. Doufám, že stejně pomůže i vám. Jazyk XAML se primárně používá k tvorbě grafických prvků aplikací. V druhé části této knihy se proto soustřeďuje popis většiny grafických funkcí.

V dlouhodobém výhledu se pravděpodobně většina kódu XAML bude generovat pomocí interaktivních návrhových programů a jiných programátorských nástrojů. Při vývoji svých aplikací budete s těmito návrhovými nástroji časem nepochybně pracovat také. Podle mého názoru je však pro všechny programátory ve WPF klíčové, aby uměli psát kód XAML „ručně“. V této knize si ukážeme, jak na to. Možná se shodneme v tom, že programování v jazyku XAML – zejména při použití nástroje, jako je XAML Cruncher – může být velmi zábavné.

Moje knihy se podobají spíše učebnicím než referenčním příručkám. Čtenáři mých předchozích knih již vědí, že jsou tyto knihy nejužitečnější, když se čtou plynule počínaje první kapitolou. Při seznamování s jazyky C# a XAML je zásadně důležité, abyste se naučili psát jejich kód. Stejně jako v případě žonglování či hraní na hoboj se ani programování nelze naučit jen čtením knih. Lidé se obvykle s programovacím rozhraním seznámí rychleji a podrobněji, když sami kód zapisují. Pokuste se takto postupovat hlavně v prvních kapitolách a neváhejte zkoušet různé varianty kódu. V úvodních kapitolách navrhuji mnoho takových pokusů.

## System Windows a programování

Společnost Microsoft vydala první verzi svého systému Windows na podzim roku 1985. Od té doby systém Windows průběžně aktualizuje a zdokonaluje. Nejzásadnější změnou byly systémy Microsoft Windows NT (1993) a Windows 95 (1995), kdy systém Windows přešel od 16bitové k 32bitové architektuře.

V prvních verzích systému Windows bylo možné psát programy pro tento systém prakticky pouze jediným způsobem, který byl založen na přístupu k rozhraní Windows API pomocí programovacího jazyka C. Během dalších let se programování systému Windows přizpůsobily mnohé další jazyky, včetně jazyků Microsoft Visual Basic a C++. V případě jazyků založených na jazyku C nabízí společnost Microsoft v současnosti čtyři přístupy, jak tvořit aplikace systému Windows. Podrobnosti uvádí následující tabulka.

## Psaní aplikací pro systém Windows pomocí jazyka založeného na jazyku C

(hledisko společnosti Microsoft)

Dostupnost od roku	Jazyk	Rozhraní
1985	C	Rozhraní Windows API
1992	C++	Knihovna MFC (Microsoft Foundation Class)
2001	C# nebo C++	Windows Forms (součást platformy .NET Framework 1.0)
2006	C# nebo C++	Windows Presentation Foundation (WPF)

Nemám v úmyslu vám říkat, který jazyk nebo rozhraní byste při psaní aplikací pro Windows měli používat. Toto rozhodnutí musíte přijmout sami s ohledem na konkrétní programátorský úkol a prostředky, které máte k dispozici.

Rozhodně lze doporučit platformu .NET. Programy vytvořené pro platformu .NET se obvykle kompilují do zprostředkujícího kódu (jazyka MSIL neboli Microsoft Intermediate Language), který se pak překládá do nativního kódu při spuštění aplikace. Tento řízený kód umožňuje chránit operační systém před chybami programů, které by mohly poškodit uživatelské prostředí, a poskytuje také programům platformy .NET teoretickou nezávislost na platformě.

Možná vám není jasné, proč společnost Microsoft vydala dvě odlišné platformy .NET pro psaní klientských aplikací systému Windows. Domnívám se, že rozhraní Windows Forms bude při vývoji aplikací systému Windows i nadále hrát významnou roli. Zejména s ohledem na rozšíření platformy .NET Framework 2.0 obsahuje rozhraní Windows Forms bohatší škálu výkonných ovládacích prvků a standardních dialogových oken než subsystém WPF. Programování na platformě Windows Forms je také v mnoha ohledech snazší než ve WPF. Programování Windows Forms je tématem dvou mých knih. *Programming Windows with C#* (Programování systému Windows v jazyku C#; Microsoft Press, 2001) je rozsáhlá a komplexní kniha, která podrobně popisuje první verzi rozhraní Windows Forms. Kniha *Programming Microsoft Windows Forms* (Programování rozhraní Microsoft Windows Forms; Microsoft Press, 2005) je oproti tomu kratší a přehlednější. Oproti předchozí knize doplňuje přehled funkcí rozhraní Windows Forms 2.0 a začínajícího programátora ve Windows Forms dokáže rychle uvést do problematiky i samostatně.

Rozhraní WPF sice zatím neposkytuje plnou paletu ovládacích prvků a standardních dialogových oken, která je k dispozici ve Windows Forms, ale klade pevné základy pro rozsáhlý budoucí vývoj. Rozhraní WPF se vyplatí zvolit hlavně v případech, kdy potřebujete značně upravovat grafický vzhled svých aplikací.

# Požadavky na systém

Chcete-li pracovat s touto knihou, potřebujete:

- systém Windows Vista, Windows XP s aktualizací Service Pack 2 nebo Windows Server 2003,
- vývojové prostředí Microsoft Visual Studio 2005 Standard Edition, Visual Studio 2005 Professional Edition, Microsoft Visual C# 2005 Express Edition nebo novější verzi prostředí Visual Studio (Verze Visual C# 2005 Express Edition neobsahuje kolekci rastrových obrázků a ikon, které se v této knize používají. Pomocí verze Visual C# Express Edition nelze kompilovat projekty, které obsahují soubory XAML. Jedná se o zásadní omezení, které nedovoluje použít Visual C# Express Edition s druhou částí této knihy.),
- platformu .NET Framework 3.0 (dodává se jako součást systému Windows Vista),
- sadu .NET Framework 3.0 SDK (Software Development Kit),
- počítač, který umožňuje uvedený software spustit<sup>1</sup>.

Všechny verze platformy .NET Framework a sad SDK jsou k dispozici na webu společnosti Microsoft:

<http://msdn2.microsoft.com/en-us/windowsvista/aa904955.aspx>

Z této stránky můžete přejít na stránku pro stažení nebo instalaci balíčku .NET Framework Redistributable Package (také se označuje jako Runtime Components):

<http://www.microsoft.com/downloads/details.aspx?FamilyId=10CC340B-F857-4A14-83F5-25634C3BF043>

Balíček Runtime Components není nutný, používáte-li systém Windows Vista. Potřebujete také sadu SDK, kterou si můžete stáhnout jako obraz ISO k vypálení na CD:

<http://msdn.microsoft.com/en-us/netframework/aa569263.aspx>

Nakonec se neobejdete ani bez rozšíření prostředí Visual Studio 2005, která podporují programování v subsystému Windows Presentation Foundation:

<http://www.microsoft.com/downloads/details.aspx?FamilyId=F54F5537-CC86-4BF5-AE44-F5A1E805680D>

Teoreticky lze aplikace WPF kompilovat a spouštět i bez prostředí Visual Studio. Sada SDK obsahuje program pro příkazový řádek s názvem MSBuild, který dokáže sestavit aplikace WPF ze souborů projektu C# (.csproj). Prostedí Visual Studio však programování ve WPF značně usnadňuje a podle názoru některých programátorů je vzhledem k současnému vývoji v podstatě nezbytné.

---

<sup>1</sup> **Poznámka českého vydavatele:** Kromě výše uvedeného lze samozřejmě využít i Windows Server 2008, Visual Studio 2008 a platformu .NET 3.5. Nové verze těchto nástrojů jsou zpětně kompatibilní s kódy starších verzí. VS 2008 navíc obsahuje subsystém Windows Presentation Foundation, takže u něj není nutné níže uvedené rozšíření Visual Studia 2005.



## Předběžný software

Tato kniha byla kontrolována a testována pomocí verze CTP (Community Technology Preview) platformy .NET Framework 3.0 z června 2006. Očekávám, že kniha bude plně kompatibilní s ostrými verzemi platformy .NET Framework 3.0 a sady SDK, ale pokud se vyskytnou jakékoli změny nebo opravy, najdete je v článku znalostní báze Microsoft Knowledge Base, který je uveden v části „Podpora této knihy“ dále v této úvodní kapitole.

## Ukázky kódu

Všechny popisované ukázky kódů si můžete stáhnout z webové stránky této knihy:

<http://knihy.cpress.cz/k1456>

ze záložky Soubory ke stažení.

## Podpora této knihy

Autor knihy se maximálně snažil o to, aby kniha ani doprovodný obsah neobsahovaly žádné chyby. Opravy a změny se budou postupně objevovat v článku znalostní báze Microsoft Knowledge Base.

Nakladatelství Microsoft Press poskytuje podporu knih a doprovodného obsahu na následující webové stránce:

<http://www.microsoft.com/learning/support/books>

## Otázky a komentáře

Máte-li připomínky, dotazy nebo nápady týkající se této knihy nebo doprovodného obsahu, nebo pokud nenajdete odpověď na své otázky na dříve uvedených adresách, obraťte se na společnost Microsoft Press e-mailem:

[mspinput@microsoft.com](mailto:mspinput@microsoft.com)

nebo poštou:

*Microsoft Press*

*Attn: Applications = Code + Markup Editor*

*One Microsoft Way*

*Redmond, WA 98052-6399*

Berte prosím na vědomí, že na těchto adresách není k dispozici odborná pomoc k softwarovým produktům společnosti Microsoft.

## Poznámka redakce českého vydání

I nakladatelství Computer Press, které pro vás tuto knihu přeložilo, stojí o zpětnou vazbu a bude na vaše podněty a dotazy reagovat. Můžete se obrátit na následující adresy:

Computer Press  
redakce počítačové literatury  
Holandská 8  
639 00 Brno

nebo

*knihy@cpress.cz.*

Další informace a případné opravy českého vydání knihy najdete na internetové adrese <http://knihy.cpress.cz/K1456>. Prostřednictvím uvedené adresy můžete též naší redakci zaslat komentář nebo dotaz týkající se knihy. Na vaše reakce se srdečně těšíme.

## Web autora

Informace specifické pro tuto knihu najdete také na této stránce mého webu:

<http://www.charlespetzold.com/wpf>

Z úvodní stránky mého webu také vedou odkazy na informace o mých dalších knihách, můj blog a různé články.

## Zvláštní poděkování

Chci poděkovat své agentce Claudette Moore ze společnosti Moore Literary Agency, která mi pomohla dát tuto knihu dohromady, a také mým redaktorům ve společnosti Microsoft Press: Valerie Woolley (projektový redaktor), Kenn Scribner (technický redaktor) a dalším lidem uvedeným na stránce o autorských právech této knihy, kteří udělali maximum, aby můj text zdokonalili po jazykové stránce a opravili chyby.

Hodně mi také dala komunita blogerů – z nichž mnozí pracují ve společnosti Microsoft – kteří píší o programování ve WPF a nepřímo se na této knize podíleli. Dále mi pomohli lidé, kteří mi posílali odkazy. Nejvíce přispěli (v abecedním pořadí): Chris Anderson, Pete Blois, Aaron Cornelius na webu [www.wiredprairie.us](http://www.wiredprairie.us), Pablo Fernicola, Jessica Fosler, Henry Hahn, Karsten Januszewski, Chuck Jazdzewski, Nick Kramer, Rahul Patil, Rob Relyea, Greg Schechter, Tim Sneath, David Teitlebaum a Shawn Van Ness.

Samozřejmě i s touto skupinou špičkových poradců jsem se mohl dopustit různých přešlapů. Za všechny chyby a omyly, které v této knize zůstaly, odpovídám výhradně já.

Zejména chci poděkovat své ženě Deirdre, která mi nabídla slova moudrosti, věty soucitu, odstavce trpělivosti, kapitoly dobroty a velkou dlouhou knihu lásky.

Charles Petzold  
New York City a Roscoe, New York  
září 2005 – červen 2006