Ondřej Bitto

Vhodné i pro úplné začátečníky

Příkazový řádek

Windows 7 Rychlá řešení pro správu systému a konfiguraci sítě

Výuka na konkrétních příkladech

Rychlý přehled příkazů a jejich parametrů



Ondřej Bitto

Příkazový řádek Windows 7

Computer Press, a.s. Brno 2011

Příkazový řádek Windows 7

Ondřej Bitto

Computer Press, a. s., 2011. Vydání první.

Jazyková korektura: Marie Schreinerová Sazba: Petr Klíma Rejstřík: Daniel Štreit Obálka: Martin Sodomka Komentář na zadní straně obálky: Libor Pácl Technická spolupráce: Jiří Matoušek, Zuzana Šindlerová, Dagmar Hajdajová Odpovědný redaktor: Libor Pácl Technický redaktor: Jiří Matoušek Produkce: Petr Baláš

Computer Press, a. s., Holandská 3, 639 00 Brno

Objednávky knih: http://knihy.cpress.cz distribuce@cpress.cz tel.: 800 555 513

ISBN 978-80-251-3506-8 Prodejní kód: K1930 Vydalo nakladatelství Computer Press, a. s., jako svou 4053. publikaci.

© Computer Press, a. s. Všechna práva vyhrazena. Žádná část této publikace nesmí být kopírována a rozmnožována za účelem rozšiřování v jakékoli formě či jakýmkoli způsobem bez písemného souhlasu vydavatele.

Obsah

Úvod	9
Komu je kniha určena	9
Zpětná vazba od čtenářů	11
Skripty ke knize	11
Errata	11

KAPITOLA 1

Základní kroky a práce se soubory	13
Ovládání příkazového řádku a jeho ladění	13
Spuštění a ukončení příkazového řádku	14
Změna velikosti okna příkazového řádku a jeho umístění	17
Úprava a přizpůsobení příkazového řádku	20
Univerzální příkazy pro praktické použití	24
Práce s historií příkazů	25
Práce se složkami	27
Změna a výpis obsahu složky	27
Další práce se složkami	29
Práce se soubory	30
Možnosti kopírování souborů	30
Mazání a zobrazení obsahu souborů	33
Hledání a porovnání souborů	36
Robocopy: Robustní kopírování souborů	38

KAPITOLA 2

43
43
44
46

Ukončení procesů	48
Příkaz taskkill	48
Ukončení prostřednictvím příkazu tskill	51
Procesy, kterých se nemusíte bát	52
Specifická práce s procesy a pamětí	53
Spuštění procesu s definovanými oprávněními	53
Souhrn paměti s příkazem mem	54
Práce se službami	55
Správa příkazem sc	56
Změna stavu služby	57
Zjištění závislostí	58
Automatické spouštění úloh příkazem at	60
Grafická nástavba příkazů tasklist, taskkill, sc	61
Záložka Aplikace	62
Záložka Procesy	63
Záložka Výkon	66

Informace o systému a jeho správa 69

Zjištění systémových údajů	69
Příkaz ver	69
Příkaz systeminfo	71
Práce s datem a časem	74
Specifické vypnutí systému	75
Příkaz shutdown	75
Grafická nástavba příkazu shutdown	77
Správa a úprava registru	78
Jak vypadá registr uvnitř	79
Hodnoty použitelné v registru	80
Příkaz reg	81
Přidání a odebrání hodnot registru pomocí souboru	84
Tipy pro ladění systému pomocí registru	85
Změna titulku okna Internet Exploreru	85
Experimentujte s nastavením prohlížeče	86
Změna pozadí nabídky Internet Exploreru	86
Ukládání dočasných souborů	86
Úprava ikon panelu nástrojů Internet Exploreru	87

Omezení možností Internet Exploreru	87
Možnosti Internetu a jejich úprava	
Reset hesla hodnocení obsahu	89
Omezení procházení disku	90
Historie Internet Exploreru a její úprava	90
Výběr okna pouhým umístěním myši	90
Odstranění ikony zástupců	91
Změny složky s programy	91
Správa nepotřebných knihoven DLL	91
Správa automaticky spouštěných programů	91
Správa automaticky spouštěných služeb	92
Snížení prodlevy vypínání Windows	92

Správa a úprava pevného disku	
Kontrola disku příkazem chkdsk	93
Úprava diskové jednotky příkazem diskpart	96
Výpis a výběr disku	97
Operace s diskem	98
Možnosti skriptování	
Správa souborového systému s fsutil	
Výpis informací o jednotkách	102
Správa svazků	104
Správa diskových kvót	
Další použitelné příkazy	107
Jednoúčelové nástroje	
Správa virtuálních jednotek příkazem subst	
Formátování disku pomocí příkazu format	
Jak a kdy použít souborový systém UDF	
Defragmentace disku příkazem defrag	
Dopinujici nastroje label a convert	

KAPITOLA 5

Konfigurace místní sítě	119
Vlastnosti a součásti místní sítě	119
Typy sítí a jejich rozdělení	
IP adresy, jejich význam a formáty	121

Co je Ethernet a protokol TCP/IP?	
Základní nastavení protokolu TCP/IP	124
Příkaz Ipconfig	
Statistika připojení příkazem netstat	127
Správa fyzických adres a pokročilé nastavení	130
Výpisy příkazu getmac	130
Pokročilá správa příkazem arp	134
Příkazy netsh a nsloopkup	136
Příkaz lpconfig Statistika připojení příkazem netstat Správa fyzických adres a pokročilé nastavení Výpisy příkazu getmac Pokročilá správa příkazem arp Příkazy netsh a nsloopkup	

Práce se sítí a Internetem	139
Zjištění dostupnosti serveru	
Nástroj ping	140
Příkaz pathping	144
Zjišťování trasy příkazem tracert	146
Protokol FTP	
Princip činnosti	148
Přihlášení, navázání a ukončení spojení	149
Základní práce se soubory a adresáři	
Příkazy pro přenos souborů	
Další praktické příkazy	

KAPITOLA 7

Správa účtů a skupin příkazem net 159

Správa bezpečnosti pomocí net accounts	
Informace o zabezpečení a správa hesel	
Jaké volit bezpečnostní požadavky na hesla	162
Nastavení účtů pomocí net user	
Rychlý výpis účtů a jejich parametry	
Přidání nového účtu	
Zobrazení informací o účtu a další správa	168
Správa uživatelských skupin	170
Sdílení síťových prostředků	
Vytvoření a správa sdílení	
Připojení sdíleného prostředku	

Skripty příkazového řádku 179

Základní tvorba skriptů	
Formát skriptu a jeho obsah	
Vkládání komentářů do skriptů	
Příkaz echo	
Spouštění skriptů s argumenty	
Proměnné	
Zobrazení proměnných a jejich nastavení	
Matematické výrazy	
Podmíněné příkazy IF	

PŘÍLOHA A

Přehled syntaxe příkazů	191
Úprava příkazového řádku a jeho vlastností	
Color	
Prompt	
Práce se složkami a soubory	
, Cd	
Dir	
Хсору	
Del	
Attrib	
Type a more	
Find	197
Comp	
Robocopy	198
Správa procesů a služeb	
Tasklist	
Runas	
Sc	
At	
Informace o systému a jeho správa	
Systeminfo	
Shutdown	
Reg	

Práce s pevným diskem	
Chkdsk	
Diskpart	
Fsutil	
Format	
Defrag	
Convert	
Nastavení místní sítě	
lpconfig	
Netstat	
Getmac	
Arp	
Práce se sítí a Internetem	
Ping	
Pathping	
Tracert	
FTP	
Příkaz net	
Net account a net user	
Net localgroup	
Net share	

PŘÍLOHA B

Rozdíly v	příkazech	
Windows	XP a Windows 7	 21

Rejstřík 22

Úvod

Většina začínajících uživatelů používá systém Windows 7 prostřednictvím jeho grafického rozhraní, nicméně v řadě případů je praktičtější (nebo dokonce zcela nezbytné) sáhnout po příkazovém řádku. Díky této knize máte jedinečnou možnost jej zvládnout od základů až po detailní nastavení a vyladění, z každého čtenáře se tak během chvíle stane profesionální uživatel.

Kniha se i přes svůj záměr na příkazový řádek výborně hodí také pro rychlé zvládnutí systému Windows a použití kteréhokoliv počítače, principy jsou totiž v základu často stejné. Získáváte tak na českém trhu ojedinělou příručku, díky níž odhalíte všechna tabu a vyhnete se zbytečným chybám.

Komu je kniha určena

Pokud jste se s příkazovým řádkem nikdy nesetkali a rádi byste udělali první krůčky, tato publikace vás jeho použitím nenásilnou a trpělivou formou provede. Naučíte se vše od úplných základů, k nimž patří soubory se složkami a provedení základních nastavení přesně podle vašich potřeb.

Jestliže již základní práci v příkazovém řádku zvládáte, pak zjistíte, jak jej využít také k jiným činnostem než jen k jednoduchým úpravám systému či zpracování rychlých příkazů. Naučíte se konfigurovat Windows, uživatelské účty, upravit registr a mnoho dalšího. Nechybí ani detailní popis možností konfigurace a testování místní sítě či Internetu, samozřejmostí je také před-stavení tvorby dávkových souborů. Všem čtenářům kniha nabízí nepřeberné množství návodů a popis možností, o nichž byste se jinak pravděpodobně jen těžko dozvěděli.

S knihou se přístupnou formou naučíte příkazový řádek využívat ke všem každodenním úkonům, takže se stane nedílnou součástí vašich domácností i pracovního prostředí. Nebudete se muset bát, že byste někdy nějaký problém neuměli vyřešit. V knize používáme několik speciálních odstavců, které doplňují text dalšími informacemi:

🗞 Poznámka

Jde o rozšiřující informaci na okraj.



Doplňuje výklad o zajímavou a prospěšnou informaci.



Informace, kterou byste si měli zapamatovat.

Zpětná vazba od čtenářů

Nakladatelství a vydavatelství Computer Press stojí o zpětnou vazbu a bude na vaše podněty a dotazy reagovat. Můžete se obrátit na následující adresy:

redakce PC literatury Computer Press Spielberk Office Centre Holandská 3 639 00 Brno

nebo

sefredaktor.pc@cpress.cz

Skripty ke knize

Z adresy *http://knihy.cpress.cz/K1930* si po klepnutí na odkaz Soubory ke stažení můžete přímo stáhnout delší skripty uvedené v kapitolách 4 a 8.

Errata

Přestože jsme udělali maximum pro to, abychom zajistili přesnost a správnost obsahu, chybám se úplně vyhnout nelze. Pokud v některé z našich knih najdete chybu, ať už chybu v textu nebo v kódu, budeme rádi, pokud nám ji oznámíte. Ostatní uživatele tak můžete ušetřit frustrace a nám pomůžete zlepšit následující vydání této knihy.

Veškerá existující errata zobrazíte na adrese *http://knihy.cpress.cz/K1930* po klepnutí na odkaz Soubory ke stažení.

Základní kroky a práce se soubory

Příkazový řádek se vám stane neocenitelným pomocníkem při řešení celé řady problémů, které vás na cestě ve Windows mohou potkat. Oceníte jej jak v případě základního ovládání, sledování a nastavení systému, tak při nutnosti nahlédnutí systému takříkajíc pod lupou – mnoho příkazů poskytne nesrovnatelně rychlejší cestu k provedení nastavení, než jak by tomu bylo v případě použití klasického grafického rozhraní. Abyste ale zvládli příkazový řádek v celé jeho síle, nesmíte opomenout důležité první kroky, na něž se zaměří tato kapitola.

Ovládání příkazového řádku a jeho ladění

Příkazový řádek může na první pohled vypadat velice jednoduše, nicméně jeho zvládnutí představuje základ pro jakékoliv další operace, používání

příkazů a pokročilé ovládání. Důležité jsou hlavně možnosti jeho spuštění, stejně tak přizpůsobení vlastnímu použití.

Spuštění a ukončení příkazového řádku

Abyste mohli příkazový řádek využívat, musíte jej samozřejmě nejprve spustit. K tomu ve Windows 7 můžete využít různé způsoby. Jednou z nejvíce intuitivních cest je použití nabídky Start, díky které přímo zjistíte, kde je v ní odkaz obsažen:

- **1.** Levým tlačítkem myši klepněte na ikonu nabídky **Start** v levém dolním rohu obrazovky.
- Přejděte na položku Všechny programy → Příslušenství → Příkazový řádek.
- **3.** Automaticky se otevře okno příkazového řádku, v němž již můžete vkládat příkazy a ovládat z něj systém.



Nejrychlejším spuštěním příkazového řádku je stisknutí klávesové zkratky **Win + R** (případně prosté stisknutí klávesy Start) a poté vepsání **cmd**. Tím se otevře okno naprosto identické s příkazovým řádkem, stejně jako to bylo popsáno v předchozím postupu.



Obrázek 1.1: Hlavní rozhraní příkazového řádku po jeho spuštění ve Windows 7

Pokud si přejete spouštět příkazový řádek klávesovou zkratkou, vytvořte si zástupce k uvedenému spustitelnému souboru a přiřaďte mu požadovanou zkratku. Tuto zkratku pak využijete odkudkoliv k rychlému spuštění, přesný postup je následující:

- Klepněte na plochu pravým tlačítkem myši a z místní nabídky zvolte příkaz Nový → Zástupce.
- V nově otevřeném dialogu do textového pole Zadejte umístění položky vložte příkaz cmd a stiskněte tlačítko Další. Stejně stiskněte také Dokončit po následném vložení názvu zástupce.
- **3.** Na nově vytvořeného zástupce klepněte pravým tlačítkem myši a zvolte příkaz **Vlastnosti**.
- **4.** Na záložce **Zástupce** klepněte levým tlačítkem myši do pole **Klávesová zkratka** a stiskněte požadovanou klávesovou zkratku.
- **5.** Změny uložte stisknutím tlačítka **OK**, nyní již můžete příkazový řádek spouštět přiřazenou zkratkou.

Kompatibilita	Zabezpečení Podrobnosti Předchozí verze			
Obecné Zá	ástupce Možnosti Písmo Rozložení Barvy			
Př	říkazový řádek			
Typ cíle:	Aplikace			
Um ístění cíle:	system32			
<u>C</u> íl:	%windir%\system32\cmd.exe			
<u>S</u> pustit v:	%HOMEDRIVE%%HOMEPATH%			
Klávesová zk <u>r</u> atka:	Ctrl + Alt + F			
Spus <u>t</u> it:	V normálním okně 👻			
<u>K</u> omentář:	Spouští funkce pracující pouze v textovém režimi			
Qtevřít umístění souboru Změnit ikonu Upřes <u>n</u> it				

Obrázek 1.2: Nastavení klávesové zkratky pro spuštění zástupce příkazového řádku

Jakmile některá vaše operace v příkazovém řádku skončí neúspěchem s hlášením **Přístup odmítnut**, nemáte dostatečná práva pro její provedení. U pokusu o přístup k síťovým prostředkům může být "problém" v zabezpečení cílového počítače, kde například nemáte účet vůbec nebo jen s omezenými právy. Při práci ve svém vlastním počítači zase zapátrejte v nastavení Řízení uživatelských účtů, jelikož zápis nebo jiná změna ve vybraných systémových oblastech může končit právě tímto neúspěchem. Pro odstranění potíží zkuste příkazový řádek spustit příkazem **Spustit jako správce** z kontextové nabídky, která se automaticky objeví při klepnutí pravým tlačítkem myši na jeho zástupce.

Mezi nabídkou Start a přehledem aktivních úloh se v dřívějších Windows nacházel panel pro snadné spuštění programů nebo otevření vybraných umístění, který obsahoval odpovídající odkazy. Windows 7 od tohoto přístupu upustila a nabízejí novinku v podobě takzvaného připíchnutí programu. Díky tomu budete mít odkaz na příkazový řádek kdykoliv k dispozici. Nejsnazší postup připíchnutí vede přes již spuštěný příkazový řádek, v takovém případě si počínejte následovně:

- Na ikonu otevřeného příkazového řádku v hlavním panelu klepněte pravým tlačítkem myši.
- **2.** Z místní nabídky detailních příkazů vyberte možnost **Připnout tento program na hlavní panel**.
- Připíchnutí příkazového řádku na hlavní panel ověřte umístěním ukazatele myši nad jeho ikonu – namísto celistvého podbarvení se zde objeví ukazatel ve spodní části.
- **4.** Pro spuštění připíchnutého příkazového řádku na jeho ikonu kdykoliv klepněte levým tlačítkem myši.

🖗 Tip

Jakmile si na hlavní panel připíchnete více programů, zjistíte, že takto označené odkazy fungují obdobně jako zástupci v dřívějším panelu pro snadné spuštění – také jsou zde kdykoliv k dispozici. Pro případné budoucí odebrání připíchnutého příkazového řádku stačí z místní nabídky zvolit analogický příklad **Odepnout tento program z hlavního panelu**.



Obrázek 1.3: Nabídka s možností připnutí příkazového řádku na hlavní panel

Když se vydáte do zkoumání možností příkazového řádku, základem je spouštění příkazů pomocí jejich názvů. Jakmile se v příkazové řádce běžící program z libovolného důvodu neukončí, můžete jeho běh zastavit stisknutím klávesové zkratky Ctrl + C, kdy bude násilně ukončen a řízení vráceno příkazovému řádku. Když budete chtít ukončit celý příkazový řádek bez sahání na myš, zadejte jednoduchý příkaz **exit**.

🛞 Poznámka

Příkazový řádek spuštěný pomocí procesu cmd.exe běží jako samostatná instance, přímo z jeho okna můžete také spustit další, a to opětovným vložením příkazu cmd. Otevře se tím nové okno příkazového řádku.

Vybrané nástroje příkazového řádku, především specializované skripty, které řeší konkrétní problémy, mohou být časově velice náročné. Kdybyste chtěli takovéto úloze přiřadit vyšší prioritu, jinými slovy pro ni přiřadit více procesorového času, použijte k tomu standardního Správce úloh Windows. Zde pak na záložce **Procesy** klepněte pravým tlačítkem myši na řádek obsahující název procesu cmd.exe a z místní nabídky zvolte vyšší důležitost pomocí voleb v sekci **Nastavit prioritu**.

Změna velikosti okna příkazového řádku a jeho umístění

Příkazový řádek Windows 7 se zobrazuje v samostatném okně, s nímž můžete pracovat stejně, jako je tomu u jiných oken – tedy měnit jeho velikost, upravit si jeho umístění na pracovní ploše, případně jej pro možnost rychlého spuštění připnout mezi ostatní ikony hlavního panelu. Na uvedené možnosti

se v následující části textu detailně zaměříme, jelikož díky nim se vám práce s příkazovým řádkem zautomatizuje a nebude vám nijak překážet.

) Tip

Systém Windows 7 po vzoru Windows Vista nabízí možnost přehlednějšího zobrazení při přepínání mezi jednotlivými okny. Jednoduché přepínání zprostředkovává klávesová zkratka **Alt + Tab**, pomocí zkratky **Win + Tab** je nabízeno podrobnější zobrazení náhledu jednotlivých oken. Právě pomocí posledně zmíněné klávesové zkratky můžete v případě otevření většího počtu oken s příkazovým řádkem rychle poznat, ve kterém okně je který příkaz zpracováván.

Pro roztažení okna s příkazovým řádkem na výšku pracovní plochy stačí, abyste poklepali do titulní lišty levým tlačítkem myši. Tento postup je podobný také pro vrácení do původních rozměrů, kdy na vertikálně roztažené okno příkazového řádku stačí opět takto poklepat.

Při práci s větším množstvím aplikací najednou se můžete poměrně snadno v otevřených oknech ztratit. Proto systém Windows 7 nabízí některá základní uspořádání oken, s jejichž pomocí lze vždy získat přehled o všech aktuálně otevřených. Stačí, abyste pravým tlačítkem myši klepli na volnou oblast v hlavním panelu, čímž se otevře místní nabídka. Její příkazy pro uspořádání oken jsou následující:

- Okna na sebe jednotlivá okna se uspořádají do takzvané kaskády. Jsou přes sebe naskládána tak, že u dolních oken lze vidět pouze jejich titulní lišty. Tento postup využijete k tomu, abyste příkazový řádek třeba dočasně skryli, přesto však měli přehled o tom, v jaké úrovni je otevřený.
- Zobrazit okna nad sebou okna vyplní plochu obrazovky tak, že se naskládají nad sebe. Každé z oken přitom má jiný rozměr, je-li to možné. Toto je ideální postup pro uspořádání více oken příkazového řádku s delším výstupem na šířku.
- Zobrazit okna vedle sebe okna vyplní plochu obrazovky tím způsobem, že se naskládají vedle sebe. Tento příkaz s výhodou vyberte, pokud máte otevřená pouze dvě okna, z nichž jedním je příkazový řádek, a chcete je zobrazit symetricky, každé na jednu polovinu pracovní plochy.

 Zobrazit otevřená okna nebo Zobrazit plochu – přepíná mezi zobrazením oken a pracovní plochy.



Obrázek 1.4: Příkazy hlavního panelu pro uspořádání právě otevřených oken

Novinka Windows 7 oproti předchozím verzím tohoto systému představuje možnost rychlého uspořádání oken na obrazovce jen za použití přetažení k okrajům pracovní plochy. Tyto postupy se vám budou hodit i u úprav okna s příkazovým řádkem:

- Maximalizace okna uchopte okno za jeho titulek a za stisknutého levého tlačítka myši přetáhněte zcela k hornímu okraji plochy.
- Uspořádání do řady vpravo/vlevo, plná výška uchopte okno za horní okraj levým tlačítkem myši a za jeho stálého držení je přetáhněte zcela vpravo/vlevo.
- Rychlé skrytí/odkrytí všech oken kromě aktuálního jedná se o funkci s názvem Aero shake. Při uchopení okna za horní okraj a rychlém zatřepání za stále stisknutého levého tlačítka myši se všechna okna kromě tohoto aktuálního minimalizují. Pokud postup opakujete, původně minimalizovaná okna budou opět obnovena.

Úprava a přizpůsobení příkazového řádku

Pokud je vám výchozí nastavení písma příkazového řádku nepříjemné, především například hůře čitelné, můžete si změnit jeho velikost i typ následovně:

- Na titulek okna příkazového řádku klepněte pravým tlačítkem myši a vyberte příkaz Vlastnosti.
- 2. V nově otevřeném dialogu přejděte na záložku Písmo.
- **3.** Pomocí nabídky **Písmo** nyní zvolte požadovaný druh písma a prostřednictvím seznamu **Velikost** vyberte jeho velikost.
- 4. Změny uložte stisknutím tlačítka OK, projeví se okamžitě.

Nastavení písma oceníte hlavně v případě, kdy se vám většina výpisů příkazů nevejde do obrazovky okna, a přesto nechcete výstup přesměrovávat do souboru nebo schránky. Výchozí velikost písma je 8 pixelů na šířku a 12 na výšku, pro zmenšení proto zvolte například variantu 4 x 6 px nebo 6 x 8 px, samozřejmě v nabídce **Velikost**.

Možnosti Pis	mo Rozložení	Barvy
Náhled okna		Velikost
		4x 6 ▲ 6x 8 8x 8 16x 8 ■ 7x 12 8x 12 16x 12 ■ 16x 12 ■
Písmo Tr Consolas Tr Lucida Cor Bastrová písm	<u>T</u> učná písma Isole	
ndadova pisina		1
Vybrané písm C:\WINDC SYSTEM SYSTEM32 DEG DME	o:Teminal WS> dir OUS> clir 2	Každý znak je: 8 pixelů obrazovky široký 12 pixelů obrazovky vysoký

Obrázek 1.5: Nastavení vlastností písma zobrazeného v okně příkazového řádku

ු Tip

Častým problémem při praktickém používání příkazového řádku je také šířka nabízeného okna, v němž se delší výpisy nemusí správně zobrazit na jednom řádku, automaticky dojde k jejich zalomení na další. Abyste takovýmto trablům předešli, přejděte na záložku **Rozložení** a v části **Velikosti okna** navyšte hodnotu v poli **Šířka**.

Ve stejném dialogu **Vlastnosti**, zmíněném již v předchozím postupu, můžete také podrobně nastavit barvy skazového řádku, jeho pozadí i zobrazeného textu. Slouží k tomu intuitivně pojmenovaná záložka **Barvy**, kde můžete vybrat z nabízené palety, případně přesně definovat složkami červené, zelené a modré v sekci **Vybrané hodnoty barev**. Také tyto úpravy se projeví okamžitě v aktuálním příkazovém řádku.



Obrázek 1.6: Detailní konfigurace barev

Důležité

Řada uživatelů si změní barvy příkazového řádu tak, aby vypadaly co nejefektivněji, často však zapomínají na to nejdůležitější, tedy pohodlnou čitelnost. I vy upřednostněte raději snadné čtení obsahu okna s příkazovým řádkem před exotickými barvami, delší práce v něm je pak mnohem pohodlnější.

Jestliže si nechcete upravovat příkazový řádek v právě popsaném konfiguračním dialogu, ale stylově byste raději dali přednost některému specializovanému příkazu, máte možnost využít příkaz **color**. Jako jeho parametr můžete zadat přímo barvu, předává se v podobě číselné hodnoty – například vložením příkazu **color 4** změníte barevné zobrazení na červenou. Přehled předdefinovaných hodnot, jež odpovídají jednotlivým barvám, shrnuje tabulka 1.1.

0	černá	8	šedá
1	modrá	9	světle modrá
2	zelená	А	světle zelená
3	akvamarinová	В	světle akvamarinová
4	červená	С	světle červená
5	nachová	D	světle nachová
6	žlutá	Е	světle žlutá
7	bílá	F	zářivě bílá

Tabulka 1.1: Kódy barev použitelné s příkazem color

Příkaz color dovoluje kromě barvy textu (vložením jediného parametru, jak bylo popsáno) změnit i barvu pozadí, a to přidružením odpovídajícího označení barvy před barvu textu. Tak například pro nastavení žluté barvy textu na zeleném pozadí stačí využít příkaz **color 26**. Jestliže u příkazu color nevyužijete žádné doplňující parametry, obnoví se původní nastavení barev, jak bylo definováno při spuštění příkazového řádku. Tato výchozí hodnota je dána buď stávajícím oknem, přepínačem /T nebo načtením hodnoty **DefaultColor**, jež se nachází v registru systému Windows.

St. Správce: C:\\	Windows\sy:	stem32\cmd.e	xe			
C:\Users\On	C:\Users\Ondra>prompt "Ahoj, zadej prikaz: "					
"Ahoj, zade Svazek v j Sériové čí	"Ahoj, zadej prikaz: "dir *.* Svazek v jednotce C nemá žádnou jmenovku. Sériové číslo svazku je 1443-EBD5.					
Výpis adre	sáře C:\	Users\Ond	ra			
28.03.2011 28.03.2011	20:37 20:37	<dir> <dir></dir></dir>				
05.02.2011 26.02.2011	20:15		6 98 8	78 84	.jmf-resource capture-1.avi	
03.09.2009	12:40	<dir></dir>	95 1	.98	database.xml Deskton	
27.06.2011 10.06.2011	10:46 15:20	<dir> <dir></dir></dir>			Documents Downloads	
28.06.2011 05.02.2011	14:04 20:14	<dir> <dir> <dir></dir></dir></dir>			Drophox eTeks Execution	
01.05.2011 23.05.2011	11:53 18:28	<dir> <dir> <dir></dir></dir></dir>			Links Music	-

Obrázek 1.7: Příkazový řádek s pozměněnou výzvou

Ve výchozím nastavení příkazového řádku Windows 7 zobrazují cestu aktuálního adresáře, za níž můžete psát vlastní příkazy, které si přejete provádět. Jestliže si přejete toto zobrazení změnit, můžete k tomu využít jednoduchého příkazu **prompt**. Pokud za něj vložíte mezeru a libovolný text, bude se tento text zobrazovat právě namísto aktuální cesty. Výzva se může skládat nejen z běžných znaků, ale také speciálních kódů, které shrnuje tabulka 1.2.

Kód	Význam
\$A	& (ampersand)
\$B	(přesměrování)
\$C	((levá závorka)
\$D	aktuální datum
\$E	kód Escape (ASCII kód 27)
\$F) (pravá závorka)
\$G	> (symbol větší než)
\$H	backspace (smaže předchozí znak)
\$L	< (symbol menší než)
\$N	aktuální jednotka
\$P	aktuální jednotka a cesta
\$Q	= (symbol rovnosti)

Tabulka 1.2: Kódy pro vložení speciálních znaků ve výzvě příkazového řádku

Kód	Význam
\$S	mezera
\$T	aktuální čas
\$V	číslo verze systému Windows
\$_	nový řádek (CR LF)
\$\$	\$ (symbol dolaru)

Univerzální příkazy pro praktické použití

Další části textu a jednotlivé kapitoly budou popisovat praktické použití příkazů při řešení konkrétních úkolů a problémů, vždy se vám však budou hodit také univerzálně použitelné varianty – například smazání okna příkazového řádku nebo ukončení právě běžícího příkazu. Díky nim si práci v příkazovém řádku značně zjednodušíte, zpříjemníte a urychlíte. Obecně například platí, že když si kdykoli nebudete jisti s použitím různých parametrů konkrétního příkazu, můžete prakticky u všech systémových oblastí využít nápovědy volané prostřednictvím parametru?. Pokud tak potřebujete poradit například s příkazem **net**, jednoduše zadáte **net** /? a vypíšou se vám všechny související možnosti.

Podobně také například stisknutí již zmíněné klávesové zkratky **Ctrl + C** za běhu programu v řadě případů zajistí úplné přerušení a ukončení právě běžícího příkazu/programu. Někdy ale můžete potřebovat daný program pouze pozastavit, například kvůli výpisu apod. V takovém případě v příkazovém řádku za běhu daného příkazu vyzkoušejte klávesovou zkratku **Ctrl** + **Pause**, případně pouze Pause. Úspěch výsledku není zaručen, záleží na aktuálním zpracování právě běžícího programu.

Okno příkazového řádku je samozřejmě čistě textové, můžete se proto ptát, jak označit vybraný text – klasický výběr z ostatních textových částí Windows nefunguje. Postupujte proto následovně:

- 1. Klepněte do libovolné oblasti příkazového řádku pravým tlačítkem myši.
- 2. Z nově otevřené kontextové nabídky vyberte příkaz Označit.
- **3.** Přetažením myší za současně stisknutého levého tlačítka vyberte požadovanou obdélníkovou oblast.
- **4.** Výběr dokončete stisknutím klávesy **Enter**. Nyní je vybraná oblast v textové podobě uložena ve schránce.

🖓 Tip

Použití postupu pro kopírování výpisu do schránky může být občas velice zdlouhavé, proto mátek dispozici také specializovaný příkaz **clip**. Například pro umístění výpisu aktuálního adresáře do schránky stačí napsat **dir | clip**. Podrobně se možnostem přesměrování vstupů a výstupů jednotlivých příkazů bude věnovat pozdější text této knihy.

Jednou z vlastností příkazového řádku je, že uchovává provedené příkazy a výpis pro pozdější zobrazení, které můžete provést pomocí posuvníku vpravo. Často se ale dostanete do situace, kdy budete potřebovat tento výpis smazat a získat tak více volného prostoru i v rámci aktuálního zobrazení. Příkaz pro tuto činnost je jednoduchý: **cls**.

🛞 Poznámka

Stejně jako drtivá většina tuzemských uživatelů Windows 7, nejspíše také vy máte jako výchozí klávesnici nastavenou na české rozložení. To může být především u příkazového řádku nepraktické, jelikož diakritiku zde jen málo využijete. Abyste nemuseli používat přepínání přímo pomocí některých příkazů, využijte klasickou klávesovou zkratku, která funguje i v ostatních aplikacích – ve výchozím nastavení se jedná o dobře známou kombinaci levý **Alt + Shift**.

Práce s historií příkazů

Alfou a omegou práce v příkazovém řádku se často stane zápis složek a úplných cest k některým souborům. Zde je ale problém s tím, jak si je zapamatovat, navíc přepisování není zrovna pohodlné. Příkazový řádek přichází s pomocí v podobě automatického dokončování: Když například chcete přepnout do složky Program Files z aktuálního umístění C:\, stačí napsat například **Progr** a stisknout klávesu **Tab**. Systém Windows 7 automaticky volí z dostupných položek ty, které vyhovují vámi zadanému začátku.

V praxi se velice často dostanete do situace, kdy budete v příkazovém řádku potřebovat použít příkaz, který jste již jednou zadali. U jednoduchých variant, například cd, dir, cls apod., to samozřejmě není příliš zapotřebí, jiná situace ale nastává v případě, kdy se jedná o dlouhé a složité příkazy, které v sobě obsahují několik dalších. Historii použitých příkazů nejrychleji použijte automatickým doplněním části – zde stačí, když vložíte část původního příkazu a stisknete klávesu **F8**. Systém Windows 7 automaticky doplní celý příkaz podle dřívějšího zadání. To samozřejmě platí, pokud jste příkaz dříve opravdu použili.

Vybrat Správce: C:\Windows\system32\cmd.exe
C:\Users\Ondra>time /T 15:40
C:\Users\Ondra>

Obrázek 1.8: Průběh označení textu v okně příkazového řádku pro zkopírování do systémové schránky Windows 7



Předchozí text vám představil možnost automatického doplnění použitého příkazu, máme pro vás ale tip na málo známou funkci procházení šipkami historií. Pomocí kurzorových kláves nahoru a dolů můžete listovat zadanými příkazy, požadovaný spustíte klávesou **Enter**. Pro přehlednější zobrazení navíc můžete stisknout klávesu **F7**, která zajistí zobrazení přehlednější seznamové nabídky s příkazy.

```
12: time

13: date

14: cls

15: prompt

16: cls

17: prompt "Ahoj, zadej prikaz: "

18: dir *.*

19: prompt

20: cls

21: time /T
```

Obrázek 1.9: Zobrazení historie příkazů po stisknutí klávesy F7

V případě příkazového řádku je historie příkazů omezená. Historií příkazů, jak ji představily předchozí odstavce, můžete listovat pouze tak dalece, kam sahá aktuální relace příkazového řádku. Pokud tedy v okně s příkazovým řádkem provedete libovolný počet příkazů, po jeho zavření a opětovném otevření nebude v historii ani jeden zachován.

Jestliže budete potřebovat opětovně spustit poslední příkaz, případně jej lehce upravit, nemusíte používat listování kompletní historií. Právě k tomu totiž slouží rychlá funkce skrytá pod jedinou klávesou. Opětovné vyvolání posledního příkazu provedete klávesou **F3**. Příkaz poté můžete dále upravovat – provede se opět po stisku klávesy **Enter**.

Práce se složkami

Ať už si uživatel na pevném nebo připojeném disku zvolí jakoukoliv strukturu dat, ukládané soubory se vždy sdružují do složek. Jejich poslání je na první pohled jednoduché a přímočaré, nicméně jen málokdo zvládne udržet kompaktní strukturu s maximálně efektivní katalogizací dat. Ať máte složky uspořádané jakkoliv, můžete s nimi prostřednictvím příkazového řádku provádět pokročilé operace, jež vám otevřou cestu k úpravám a zobrazení statistik jednotlivých souborů.

Změna a výpis obsahu složky

Pokud se v příkazovém řádku potřebujete snadno přepnout do jiné složky, využijete k tomu příkaz **cd**. Chcete-li tak přejít do složky C:\Windows, zadáte **cd c:\windows**, v případě relativních cest samozřejmě stačí zadat název vnořené podsložky. Pokud byste se chtěli přepnout přesně o složku výš, je vypisování celé cesty zdlouhavé a hlavně zbytečné, protože existují chytré zkratky. Pro rychlé přepnutí do nadřazené složky využijte příkaz **cd**. a konečně pro přejití do nejvyšší úrovně aktuálního disku stačí zadat **cd** /.

🛞 Poznámka

V případě práce v příkazovém řádku musíte při používání příkazů pracujících se soubory a složkami rozlišovat cesty k nim, a to mezi takzvanými absolutními a relativními. Absolutní cesta určujte přesné umístění na daném disku, má tedy podobu například C:\Windows\Temp. Naproti tomu relativní cesta odkazuje na podsložky v dané složce – jste-li tak již ve složce C:\Windows, můžete se do cílové složky C:\Temp přepnout použitím příkazu **cd Temp**.

Příkazem **dir** vypíšete kompletní obsah složky, což je často praktické, jindy ale zbytečné. Proto můžete použít také masku vypisovaných dat, jež vám výpis vyfiltruje. Například všechny soubory a složky, které začínají písmenem **t**, vypíšete příkazem **dir t***. Symbol hvězdičky (asterisk) zde zastupuje

libovolný text v názvu; pokud byste chtěli zastoupit jen jeden znak, použijte namísto něj **?**.

Příkaz **dir** podporuje hned několik zajímavých parametrů, které vám dovolí přizpůsobit výpis a určit některé detaily. Zde je jejich přehled:

- /p pozastaví výpis při dosažení konce okna
- ◆ /s vypíše soubory nejen v daném adresáři, ale i ve všech podadresářích
- ◆ /w použije široký formát výpisu na sloupce

Speciálním parametrem, který se vám při práci s příkazem dir v praxi bude hodit, je /**o**, jenž zajistí automatické třídění výpisu. Obsah složky při jeho použití (samozřejmě i v kombinaci s dalšími parametry) máte možnost řadit podle názvu (N), velikosti (S), přípony (E), data a času (D) nebo s prioritou adresářů (G). Tak například vypsání obsahu aktuálně otevřeného adresáře se řazením podle přípony provedete bleskovým vložením příkazu **dir** /**O**E.

Správce: C:\\	Windows\sy:	stem32\cmd.exe	
C:\Users\On Svazek v j Sériové čí	dra\Musi ednotce slo svaz	c≻dir ∕0E C nemá žád ku je 1443	nou jmenovku. -EBD5.
Výpis adre	sáře C:\	Users\Ondr	a\Music
23.05.2011 23.05.2011 14.03.2011 07.03.2011	18:28 18:28 21:23 22:00	<dir> <dir> <dir> <dir> <dir></dir></dir></dir></dir></dir>	- 2001-toxicity Alanis Morissette - 1999 - MTV Unplugged Outicle Outicherste
27.01.2011 27.01.2011 27.01.2011	10:59	<dir> <dir> <dir></dir></dir></dir>	Audials Inditionology Audials Converted Music Audials Imported Music
27.01.2011 27.01.2011 04.05.2011	13:29 10:59 15:38	<dir> <dir> <dir></dir></dir></dir>	Audials Music Videos Audials Podcasts Audials Recorded Music
01.05.2011 08.04.2011 04.09.2010	11:46 14:33 14:23	<dir> <dir> <dir> <dir></dir></dir></dir></dir>	Audials Recorded Streams Cranberries Live David Bowie-The Best Of David Bowie 1980-198
7 18.04.2011 16.09.2010	20:35 15:57	<dir> <dir></dir></dir>	Green Day - Awesome as Fuck Hlavně že je večívek - 15 let Souboru kretén ▼

Obrázek 1.10: Výpis obsahu složky s použitím příkazu dir /oe



Tip Příkaz dir je typickým zástupcem skupiny příkazů, které vám při standardním výpisu dokážou přeplnit obrazovku svým výstupem, a tedy nedat šanci pro pohodlné přečtení výsledku. V praxi se vám proto bude hodit doplňující rozšíření o parametr **/P**, jenž zajistí zastavení výpisu po naplnění okna s příkazovým řádkem. Další výpis pak bude pokračovat až po stisknutí libovolné klávesy. Při standardním výpisu obsahu složky vám může být překážkou zobrazení souvisejících detailů, typicky například příslušného data, času nebo velikosti. Jestliže se tomu chcete vyhnout, nabízí vám příkazový řádek Windows zjednodušení v podobě takzvaného širokého výpisu složky, k čemuž slouží rozšiřující parametr /**W**. Obsah se pak vypíše v kompaktním formátu, díky němuž získáte opravdu jen výpis složek a souborů, bez jakýchkoliv podrobností. Příkaz dir podporuje několik dalších parametrů, z nichž nejzajímavější (i již zmíněné) shrnuje tabulka 1.3.

Parametr	Význam
/A	Vypíše soubory se zadanými atributy. Těmito atributy mohou být například označení adresáře (D), soubory jen pro čtení (R), soubory určené k archivování (A), skryté soubory (H) nebo třeba systémové soubory (S)
/B	Výpis holého formátu, tedy bez doplňující hlavičky a souhrnu.
/C	Slouží pro zobrazení oddělovače tisíců u velikosti souborů. Jedná se o výchozí nastavení, tento styl výpisu můžete vypnout použitím analogického parametru /-C.
/D	Formát výpisu je identický se širokým formátem (tedy jako při pou- žití parametru /W), avšak soubory budou seřazeny podle sloupce.
/L	Zapne použití malých písmen během výpisu.
/0	Seřazení výpisu souborů podle předem rozšiřujících parametrů. Můžete použit například řazení dle názvu (N), velikosti (S), přípony (E), data a času (D) nebo s prioritou adresářů (G).
/P	Pozastavení výpisu obsahu složky po zaplnění obrazovky příkazo- vého řádku.
/Q	Rozšíření výpisu o zobrazení vlastníků souborů.

Tabulka 1.3: Přehled parametrů příkazu dir

Další práce se složkami

Složku můžete smazat pomocí příkazu **rd** (nebo totožně **rmdir**). Za tento příkaz zapište název mazaného adresáře a stiskněte Enter. Dávejte při použití tohoto příkazu pozor, adresář se rovnou smaže, tedy nebudete upozorněni.

Uvedený základní příkaz pro smazání složky má nevýhodu, že nezvládne odstranit složku, která obsahuje soubory. Jednou z cest tak je následovat

obsah a smazat soubory ručně, teprve poté odebrat již prázdnou složku. Mnohem praktičtější však je využití parametru /s. Příkaz rd /s tak smaže složku předanou jako parametr i s obsaženými soubory.

Během práce s jednotlivými adresáři by vás mohlo otravovat zdlouhavé vypisování cest k jednotlivým z nich, typicky během nutnosti přepnout se do jiného. Právě proto v příkazovém řádku můžete využít dvojice příkazů **pushd** a **popd**. Prvně jmenovaný uloží aktuální adresář do mezipaměti a přejde do nového, jehož cestu zadáte, naproti tomu popd přejde právě do adresáře, který byl aktivní při použití pushd. Pokud jste tedy v příkazovém řádku v adresáři c: a zadáte příkaz pushd users, nejenže se dostanete do adresáře c:\users, ale zároveň se do paměti uloží aktuální cesta c:\. Jakmile pak kdykoliv zadáte příkaz popd, navrátíte se okamžitě do dříve uloženého adresáře c:.



🛞 Poznámka

Jak již samotný princip použití dvojice příkazů pushd a popd naznačuje, dochází opakovaným použitím příkazu pushd k přepsání dříve uložené hodnoty. Odkazy na adresáře se tedy nekumulují, v paměti může být vždy uložen pouze jeden.

Práce se soubory

Soubory jsou z pohledu koncového uživatele hlavní jednotkou ukládanou na pevný disk, zprostředkovávají totiž všechna důležitá data. Na první pohled je tak jedno, jaký souborový systém disk aktuálně používá, které fyzické uložení má k dispozici, uživatel totiž přistupuje k souborům jako k logickým jednotkám. Nejinak je tomu také v případě příkazového řádku.

Možnosti kopírování souborů

Ať jste v příkazovém řádku aktuálně v jakékoli složce, vždy si můžete její obsah nechat rychle vypsat prostřednictvím základního příkazu dir. Při jednoduchém použití získáte výpis, který všechny položky řadí abecedně, namíchá tak dohromady složky i soubory.

Při práci v příkazovém řádku budete poměrně často využívat různé operace se soubory. Abyste nemuseli dlouho tápat s těmito základními funkcemi, můžete si okamžitě osvojit příkaz pro kopírování souborů. Kopírování provedete příkazem **copy**, kterému pomocí dvou parametrů určíte zdroj a cíl, například **copy c:\test.txt c:\windows\test.txt**.



Obrázek 1.11: Rychlé zkopírování souboru s výslednou zprávou příkazového řádku

Důležité

Uživatelé se při použití příkazového řádku velice často obávají toho, aby něco nepokazili, nepoškodili systém. V řadě případů se samozřejmě jedná o odůvodněné obavy, příkaz **copy** vám však v tomto ohledu vychází vstříc: nemusíte se bát přepsání souboru. Pokud byste se pokusili přepsat soubor stejného jména, budete při základním použití příkazu **copy** včas varováni.

Rozšířenou variantou příkazu copy je **xcopy**, jenž dovoluje pracovat s celými adresářovými strukturami. Jeho základní použití je vždy stejné, kdykoliv tak můžete určit rovněž zdroj a cíl, navíc společně s doplňujícími parametry. Kromě toho ale získáváte několik dalších možností, z nichž nejvíce používané shrnuje tabulka 1.4.

Parametr	Význam
/D	Kopírování pouze těch souborů, které mají poslední datum změny novější než nastavené datum. Formát parametru je /D:m-d-r, postup- ně tedy jde o přesné určení měsíce, data a roku. Jestliže nezadá- te žádné datum, dojde pouze ke kopírování souborů, u nichž je čas zdroje novější než čas v cílovém umístění.
/P	Zajistí zobrazení dotazu před vytvořením každého z kopírovaných souborů v cílovém umístění.
/S	Zkopíruje všechny adresáře a podadresáře, avšak s výjimkou prázd- ných.
/W	Před zahájením kopírování dat vyčká na potvrzení stisknutím libovol- né klávesy.
/C	Kopírování dat ve frontě bude automaticky pokračovat i v případě, že dojde k chybě při čtení nebo zápisu některého ze zpracovávaných souborů.
/Q	Během kopírování nedojde ke zobrazení názvu právě zpracováva- ných souborů.
/L	Zobrazí soubory, které byly zkopírovány.
/H	Aktivuje automatické kopírování také těch souborů, které mají nastaveny příznaky skrytý či systémový.
/R	Zapíná možnost přepsání souborů, které jsou určené pouze pro čtení.
/Τ	Vytvoří strukturu adresářů, ale nezkopíruje do nich zdrojová data.
/U	Do kopírování zahrne pouze ty soubory, které již existují v cílo- vém umístění. Tohoto parametru s výhodou využijete v případě, kdy potřebujete aktualizovat soubory v novém umístění, například v rámci pravidelné údržby záložních kopií.
/Υ	Vypne automatické zobrazení výzvy při pokusu o přepsání souboru, který již v cílovém umístění existuje. Analogicky můžete použít také parametr /-Y, jenž zajistí, že zobrazení výzev budou opět aktivována.
/J	Kopíruje soubory pomocí vstupu/výstupu bez mezipaměti. Tato mož- nost je doporučována pro velmi velké soubory, například v řádu giga- bajtů.

Tabulka 1.4: Přehled parametrů příkazu xcopy

Mazání a zobrazení obsahu souborů

Nejen kopírováním souborů samozřejmě budete během práce s daty na disku v příkazovém řádku živi, proto byste měli zvládnout například i jednoduché přejmenování, které není vůbec o nic těžší. K přejmenování slouží příkaz **ren**, jenž si opět bere dva parametry v podobě původního názvu souboru a nového.

Abyste nemuseli přesouvání souboru provádět ručně zdlouhavou kombinací příkazů **copy** a **del**, nabízí vám příkazový řádek Windows 7 příkaz, který to provede najednou. Pokud byste si přáli soubor přesunout, sáhněte se stejnou syntaxí po příkazu **move**. Dřívější uživatelé starších Windows tento příkaz budou pamatovat jako stálici letící historií.

Nejjednodušší použití příkazu move spočívá v prostém vložení se dvěma parametry, jež zastupují zdroj a cíl přesouvaných dat. Například použitím příkazu **move C:\test.dat C:\temp** zajistíte přesunutí souboru test.dat z kořenového adresáře disku C: do cílové složky C:\temp. Příkaz move neoplývá velkým množstvím rozšiřujících parametrů, nicméně nabízená možnost /Y, resp. /-Y pro potlačení, resp. zobrazení výzev k potvrzení přepsání již existujících souborů, se vám stanou praktickým pomocníkem při urychlení přesunu, případně zamezení nechtěného přepsání dat.

```
Správce: C:\Windows\system32\cmd.exe
C:\Users\Ondra\DOCUME~1>move test.txt c:\temp\test.txt
Chcete přepsat c:\temp\test.txt? (Ano/Ne/Uše): a
Přesunuté soubory: 1
C:\Users\Ondra\DOCUME~1>
```

Obrázek 1.12: Přesunutí souboru s potvrzením přepsání cílových dat

Aby byl přehled základních operací pro nejčastější práci se soubory kompletní, nezbývá než představit také příkaz pro jejich mazání, ten dokonce můžete spustit pouze s jediným parametrem. Smazání souboru tak provedete příkazem **del** s uvedenými názvy souborů. Mějte přitom na paměti, že se automaticky provádí "skutečné" mazání, tedy odstranění bez umístění do koše, jak by tomu bylo při mazání v rámci standardního grafického rozhraní, například v Průzkumníkovi. Také příkaz del nabízí několik rozšiřujících parametrů, nejčastěji používané z nich přináší tabulka 1.5.