

**CACHÉ VÝVOJÁŘŮM**

Společnost InterSystems se svou databází Caché byla oceněna prestižním oceněním Winner of Computerworld's Innovative Technology Award pro oblast „vývoj aplikací“.

[www.intersystems.cz](http://www.intersystems.cz)

**OVLÁDÁME CRM**

Rakouská firma update Software AG představí na veletrhu CRM-Expo v Kolíně nad Rýnem (hala 10. 2, stánek C 20) hybridní ovládní CRM systému. Hybridní marketing manager v. 6 může pracovat jak pomocí klienta pod MS Windows nebo MS Internet Explorer (PDA, Tablet PC, Notebook, atd.).

[www.update.cz](http://www.update.cz)

**PŘICHÁZÍ ALTUS VARIO 11**

Altus VARIO 11 je firemní systém pro řízení obchodních agend a vedení účetnictví. Je primárně určen pro segment středních firem a používá jej v celé České republice více než 700 zákazníků.

[www.altus.cz](http://www.altus.cz)

**BRAIN SYSTEM A SYSTINET**

Toto konsorcium se bude především opírat o produkty Systinet WASP (kompletní řešení pro tvorbu, nasazení, zabezpečení a správu webových služeb). Systinet WASP Server for Java a WASP Server for C++, obsahuje prostředí pro provoz webových služeb, a plně podporuje již zavedené standardy, včetně SOAP 1.1, SOAP 1.2 a WSDL 1.1.

[www.brain-systems.com](http://www.brain-systems.com)

[www.systinet.cz](http://www.systinet.cz)

**ÚŘADY LIDEM**

Lucio Stanca (italský ministr pro inovaci a technologie) a CEO společnosti Microsoft pro oblast EMEA Jean-Philippe Courtois vyzvali zástupce 46 zemí z celé Evropy a středomořského regionu (více než 300 delegátů) k diskusi na téma „Jak se průmysl IT a vlády mohou do budoucna spojit a řešit společné problémy“.

[www.microsoft.com/cze](http://www.microsoft.com/cze)

**CITRIX DO COMPAQU**

Společnost Citrix oznámila, že její klientský software pro platformu Windows CE bude začleněn do zcela nových tenkých klientů HP Compaq t5000 a t5500HP používající webové aplikace běžící v prostředí Citrix MetaFrame Access Suite.

[www.citrix.com](http://www.citrix.com)

# PŘEKLAD A SPOUŠTĚNÍ PROGRAMŮ

**VAŘÍME V JAVĚ 2**

Teď ho však musíme přeložit a spustit. Ale jak na to?

Při překladu generuje překladač Javy soubory .class obsahující bajtový kód určený pro JVM. Překladače jiných jazyků obvykle překládají zdrojový kód přímo do strojového jazyka konkrétního typu procesoru. Bajtový kód jazyka Java ale na rozdíl od strojového kódu není vázán na žádný konkrétní operační systém nebo typ procesoru. Díky tomu lze programy napsané v jazyce Java spouštět na různých platformách bez nutnosti jakýchkoli dodatečných úprav.

**PROMĚNNÁ CLASSPATH**

Předpokládejme, že náš zdrojový soubor uložíte do adresáře `c:\JavaVPraxi\prikklady`. Abyste se nemuseli přepínat do tohoto adresáře vždy, když chcete spustit nějaký zde uložený program, využijte proměnnou prostředí CLASSPATH (cesta k souborům tříd). Interpret jazyka Java kontroluje její obsah při každém spuštění. Takto by měla vypadat definice proměnné ve vašem systému, pokud jste zdrojový soubor skutečně uložili do adresáře `c:\JavaVPraxi\prikklady`:

```
CLASSPATH=.;C:\JavaVPraxi\prikklady
```

Pokud pracujete v systému Windows 9x, vložte definici do souboru autoexec.bat.

Po instalaci vývojového prostředí (JDK, Sun ONE Studio) zkontrolujte, zda můžete příkazy Java a Javac zadat z libovolného adresáře. Pokud ne, modifikujte proměnnou prostředí path. Přidejte do ní cestu k adresáři

```
c:\<adresářJavy>\bin. Java je v podstatě jakýsi systém v systému, a proto obsahuje rovněž mnoho vlastních systémových proměnných, jejichž znalost se leckdy hodí. Je pravděpodobné, že je všechny neznáte.
```

Proto dnes vytvoříme jednoduchý program, který umí zobrazit nejen seznam všech systémových proměnných používaných Javou, ale rovněž aktuální hodnoty vybrané proměnné. Uložte následující kód do souboru

**ExploreProperties.java:**

```
import java.util.*;
public class ExploreProperties {
    public static void showProperties() {
        Enumeration propNames = System.getProperties().propertyNames();
        while(propNames.hasMoreElements()){
            System.out.println(("String) propNames.nextElement());
        }
    }
    public static void showProperty(String arg) {
        System.out.println(System.getProperties().getProperty(arg));
    }
    public static void main(String[] args) {
        if (args.length > 0) {
            showProperty(args[0]);
        } else {
            showProperties();
        }
    }
}
```

Je pěkné, že máme hotový zdrojový kód.

**OBSAH DALŠÍCH DVOU DÍLŮ**

V následujícím dílu se seznámíme s výhodami seskupování tříd do balíčků. Tentokrát si vše ukážeme na malém diagnostickém programu, který umí vyhledat adresu IP skrytou za názvem DNS.

Ve 4. dílu využijeme získané znalosti k tvorbě jednoduchého programu pro posílání zpráv elektronické pošty.