

VÝKONOVÉ PROFILOVÁNÍ NA PLATFORMĚ .NET BC. JAN VRATISLAV

KATEDRA POČÍTAČŮ, FEL, ČVUT

VÝKONOVÉ PROFILOVÁNÍ

- Vyhledání míst v programech vhodných pro optimalizaci
- Sleduje se četnost volání a trvání jednotlivých částí programu

```
😩 🚳 🚱 🐉 🗑 💆
🖃 🤰 🏲 99,96% Thread #1397392 - 2 625,7 ms
  System.Windows.Forms.Application.Run... - 2 625,7* n
      🖹 № 95,92% OnPaint - 2519,5* ms - 46 calls - Demo.Floatio
             🗐 🔌 95.92% FastPaint - 2519.4*ms - 46 calls - Demo.I
                🕑 🚃 54,92% SetClip - 1 442,7* ms - 47 calls -

☑ ■ 35,54% SetClip - 933,5* ms - 46 calls - 9

                3,41% DrawString - 89,7 ms - 93 calls - System

○ 0,22% AddEllipse - 5,7 ms - 2 024 calls - Syste

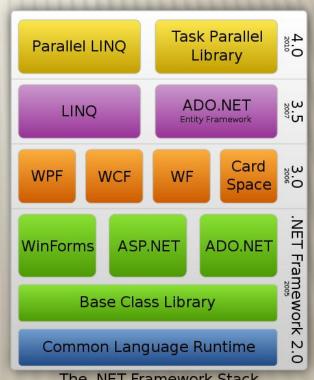
                🗐 🙀 0,02% get_FPS - 0,5 ms - 46 calls - Demo.Float
                · 🕙 0,02% get_Black - 0,5 ms - 46 calls - System.D
      🖻 🤰 0,06% MyTimer OnTick - 1,5 ms - 6 calls - Demo.DemoForm.
i ≥ 1,0 ms
```

JetBrains dotTrace Profiler

PLATFORMA MICROSOFT .NET



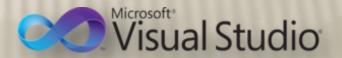
- Softwarový framework pro vývoj a spouštění programů
- Základem je běhové prostředí CLR
- Knihovny s funkcemi



The .NET Framework Stack

CÍLE PRÁCE

- Analýza principů výkonového profilování
- Rešerše stávajících komerčních a open source řešení
- Implementace a porovnání dvou metod profilování
- Integrace do vývojového prostředí Visual Studio 2010



TRACING A SAMPLING PROFILOVÁNÍ

Dva rozdílné způsoby sběru výkonnostních dat

Tracing (sledovací) Sampling (vzorkovací) mód mód Exaktní (každou Stochastický (jen významné metody) metodu) Přesný Pouze trendy Výkonnostní nároky Velmi rychlý

CO SE MĚŘÍ A VYHODNOCUJE

Měřené údaje se liší pro každý mód

Tracing (sledovací) mód

Počet volání metody

User + kernel time metody

Celkový "běžný" čas metody Sampling (vzorkovací) mód

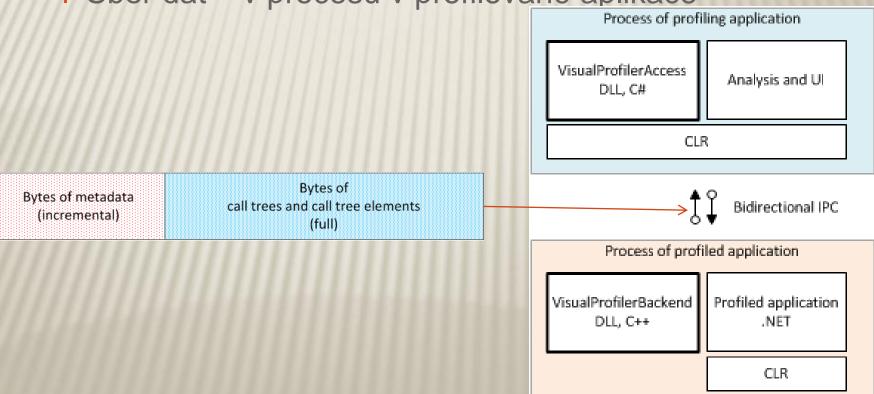
Výskyty metody na vrcholu zásobníku

Čas běhu vláken

KONSTRUKCE PROFILERU

- Dvě samostané části
 - + Vyhodnocení dat samostatný proces

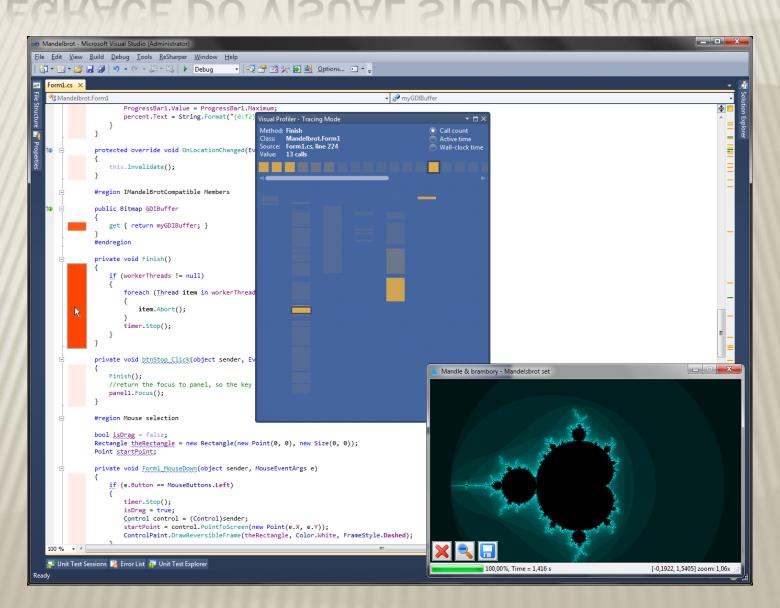
+ Sběr dat – v procesu v profilované aplikace



INTEGRACE DO VISUAL STUDIA 2010

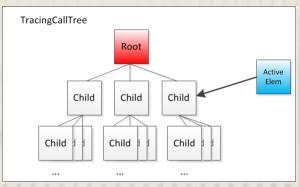
- Instalační balíček
- Příkazy pro spouštění profileru v obou módech
- Okno s výsledky
- Barevné zvýraznění "problémového" zdrojového kódu

INTEGRACE DO VISUAL STUDIA 2010

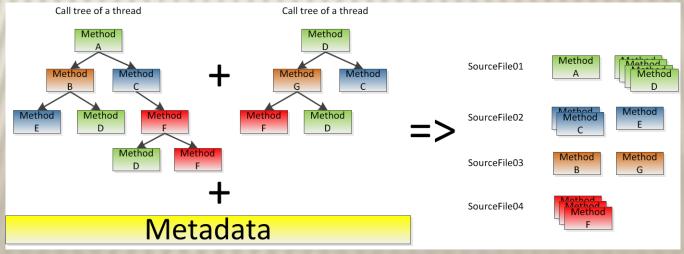


NĚKTERÉ VNITŘNÍ DATOVÉ STRUKTURY

Stromy hierarchie volání



Sloučení stromů a výpočet výsledných dat



OPTIMALIZACE

- Selektivní profilování
- Posílání metadat dat jen jednou
- Několikanásobně zrychlení a snížení paměťové náročnosti záznamu dat (přechod od zásobníků ke stromům)
- Využití lokálních vláknových proměnných pro přístup k datovým strukturám

VIDEO UKÁZKA

http://www.youtube.com/watch?v=_IRkdeGG

<u>tJo</u>



NĚKOLIK POUŽITÝCH TECHNOLOGIÍ

 Assembler, C++, COM, ATL, Profiling API, Win32 API, Named pipes, .NET, C#, Ninject, Linq, NUnit, Moq, WPF, XAML, Visual Studio 2010 Extension API, VSIX packages, MEF...

ZÁVĚR A PROSTOR PRO OTÁZKY

- Děkuji za pozornost.
- Rád nyní zodpovím Vaše otázky.