

## PROJEKT PV178 – WINDOWS SERVICE CPUSPY

Úlohou je naprogramovať Windows službu bežiacu na pozadí medzi ostatnými službami Windows (spustite zo štart menu *services.msc*).

### ÚČEL SLUŽBY

Služba bude na pozadí monitorovať využitie procesoru jednotlivými procesmi bežiacimi na PC (teda niečo obdobné ako časť funkcionality Task Managera) a z nich vypočítavať priemerné zataženie procesora každým procesom v čase.

Napríklad každých 5 minút podá priemer využitia procesora za poslednú hodinu pre každý proces. Z konfiguračného súboru *app.config* si služba načíta spomenuté dva parametre (teda ako často poskytnúť vypočítaný priemer, a z akej periódy vypočítať daný priemer).

HINT: Trieda *ConfigurationManager* a trieda *PerformanceCounter*.

### MODULY PRE VÝSTUP

Služba bude zistené štatistiky reportovať a zaznamenávať rôznymi formami pomocou modulov. Modul je trieda implementujúca nasledovné rozhranie *ICpuAverageUsageReporter*. Služba vlastní kolekciu týchto modulov a na každom zavolá *ReportUsage* vždy keď má podľa parametra z *app.config* poskytnúť výpočet.

```
/// <summary>
/// Reporting modul contract for MEF
/// </summary>
interface ICpuAverageReporter
{
    /// <summary>
    /// Reports average CPU usage per process
    /// </summary>
    /// <param name="averageCpuUsages"></param>
    void ReportUsage(IEnumerable<ProcessAverageCpuUsage> averageCpuUsages);
}

/// <summary>
/// Reported unit of measured CPU usage
/// </summary>
public struct ProcessAverageCpuUsage
{
    /// <summary>
    /// Timestamp of reported usage
    /// </summary>
    public DateTimeOffset ReportTimestamp { get; set; }

    /// <summary>
    /// Period over which average was computed
    /// </summary>
    public TimeSpan Period { get; set; }

    /// <summary>
    /// Name of process
    /// </summary>
    public string ProcessName { get; set; }
}
```

```
    /// <summary>
    /// Avarage cpu usage in %
    /// </summary>
    public double AverageCpuUsage { get; set; }
}
```

Tieto moduly si služba načíta z adresára *plug-ins*, ktorá sa bude nachádzať v zložke kde je služba nainštalovaná. Budú tam vo forme .dll súborov. Túto funkcionality naprogramujte s využitím MEF:

- Managed Extensibility Framework Overview:  
<http://msdn.microsoft.com/en-us/library/dd460648.aspx>
- MEF Community Site  
<http://mef.codeplex.com/>.

Moduly musia zvládnuť nasledovnú situáciu: ak by užívateľ napísal plug-in, na ktorom by volanie metódy *ReportUsage* trvalo príliš dlho (napríklad zapisuje na floppy disk ☺), nebudú ostatné moduly čakať a služba bude normálne pracovať a predpokladať že modul nakoniec v rozumnom čase odpovie.

## MODULY K IMPLEMENTOVANIU

K službe patria v základe dva moduly, prvý (eventlog) sa nainštaluje priamo so službou, druhý bude dodávaný ako .dll súbor pre možné nakopírovanie do adresára *plug-ins*.

### EVENTLOG MODUL

Modul zapíše pri každom zavolaní *ReportUsage* prvých 5 procesov, ktoré najviac vyťažujú procesor. Každý tento záznam bude reprezentovaný záznamom typu *info* vo vytvorenom novom eventlogu nazvanom *CpuSpyLog*.

### XML MODUL

Modul zapíše (niekam rozumne) každú hodinu nový xml súbor v ktorom bude serializovaný posledný reportovaný údaj. Teda napríklad pri 5 minutovej perióde volania *ReportUsage* prakticky použije každý dvanásť údaj, pri dvojhodinovej perióde zapíše 2x ten istý.

HINT: Trieda *Timer*.

## NEPOVINNY BONUS

Služba sa bude dať nainštalovať pomocou jednoduchého .msi balíčku. Toto nie je nevyhnutná súčasť zadania ale môže výrazne prilepšiť pri hodnotení hotového projektu.