

Novinky v C++ a MFC pro Visual Studio 2010

Jakub Čermák
Microsoft Student Partner
jakub@jcermak.cz

Agenda

- Prvky z C++0x

- Auto
- Decltype
- Lambda
- ...

- Novinky v MFC

- Class wizard, Ribbon designer

- Concurrency Runtime

- Parallel Pattern Library
- Async agent library

Standard C++0x

• Nová verze C++, nahrazující C++03

- [3 Core language runtime performance enhancements](#) [3.1 Rvalue reference and move semantics](#)
- [3.2 Generalized constant expressions](#)
- [3.3 Modification to the definition of plain old data](#)
- [4 Core language build time performance enhancements](#) [4.1 Extern template](#)
- [5 Core language usability enhancements](#) [5.1 Initializer lists](#)
- [5.2 Uniform initialization](#)
- [5.3 Type inference](#)
- [5.4 Range-based for-loop](#)
- [5.5 Lambda functions and expressions](#)
- [5.6 Alternative function syntax](#)
- [5.7 Object construction improvement](#)
- [5.8 Explicit virtual function overrides](#)
- [5.9 Null pointer constant](#)
- [5.10 Strongly typed enumerations](#)
- [5.11 Angle bracket](#)
- [5.12 Explicit conversion operators](#)
- [5.13 Template aliases](#)
- [5.14 Unrestricted unions](#)
- [6 Core language functionality improvements](#) [6.1 Variadic templates](#)
- [6.2 New string literals](#)
- [6.3 User-defined literals](#)
- [6.4 Multitasking memory model](#)
- [6.5 Thread-local storage](#)
- [6.6 Explicitly-defaulted and deleted special member functions](#)
- [6.7 Type long long int](#)
- [6.8 Static assertions](#)
- [6.9 Allow sizeof to work on members of classes without an explicit object](#)
- [7 C++ standard library changes](#) [7.1 Upgrades to standard library components](#)
- [7.2 Threading facilities](#)
- [7.3 Tuple types](#)
- [7.4 Hash tables](#)
- [7.5 Regular expressions](#)
- [7.6 General-purpose smart pointers](#)
- [7.7 Extensible random number facility](#)
- [7.8 Wrapper reference](#)
- [7.9 Polymorphous wrappers for function objects](#)
- [7.10 Type traits for metaprogramming](#)
- [7.11 Uniform method for computing the return type of function objects](#)

auto

- Dříve
 - Automatický výběr úložiště pro proměnnou
 - Výchozí chování
 - *register* - uložení do registru
- Automatická dedukce typu proměnné
 - Podle přiřazení
- Ekvivalent *var* v C#
- Výhody
 - Složitější typy (iterátory, ...)
 - Změny v kódu (změna typu kontejneru, ...)
 - Šablony
 -

decltype

- `decltype(expression)`
- „vrací“ typ daného výrazu
- Použití: templates

decltype

```
Std::vector<decltype(a+b)> muj_vect;
```

```
template<typename TA, typename TB>  
auto secti(TA a, TB b) ->  
decltype(a+b)  
{  
return a+b;  
}
```

Lambda výrazy

- Bezejmenné funkce
- Deklarované jako výraz
- Closures
 - Předání lokálních proměnných do λ funkce
 - Hodnotou nebo odkazem
 - Buď explicitní uvedení nebo všechny referencí
- Výhody
 - Lepší využitelnost fcí z <algorithms>
 - For_each, find, erase, ...
 - Zkrácení zápisu
 - Omezení počtu funkcí a metod (IntelliSense)

Lambda výrazy

```
vector<int> v;
```

```
for_each(  
    v.begin(), v.end(),  
    [](int item) { cout << item; }  
);
```

Lambda výrazy

```
vector<int> v;
```

```
auto it = find_if(  
    v.begin(), v.end(),  
    [limit](int item) -> bool  
        { return item > limit; }  
);
```

Rvalue reference

- Lvalue vs. Rvalue
 - Pojmenovaný persistentní vs. dočasný
 - $x = 3 + 4;$
 - $((i < 3) ? i : j) = 7;$
- Move semantics
 - Přenos alokovaných zdrojů mezi objekty
 - efektivita
 - Move constructor, move assign. oper.
 - `MyClass(MyClass&& other)`
 - `string s = string("h") + "e" + "ll" + "o";`
- Perfect forwarding
 - Forwarding problem

Rvalue reference

```
template<typename TA, typename TB>  
auto secti(const TA&& a, const TB&& b) noexcept  
& { return type(a+b);  
{  
return a+b;  
}
```

Ostatní

- **Static_assert**
 - Statické testy
 - `static_assert(sizeof(void *) == 4, "64-bit code generation is not supported.");`
 - Dříve: emulace přes deklaraci pole nebo šablonami
- **Nullptr**
 - Nulový pointer, který je typu pointer
 - Srovnání: `#define NULL 0`
 - `__nullptr` – pro odstranění kolize v C++/CLI
 - `Nullptr` v C++/CLI je manager reference

Agenda

- Prvky z C++0x
 - Auto
 - Decltype
 - Lambda
 - ...
- **Novinky v MFC**
 - Class wizard, Ribbon designer
- Concurrency Runtime
 - Parallel Pattern Library
 - Async agent library

MFC designery

demo

Name

Title

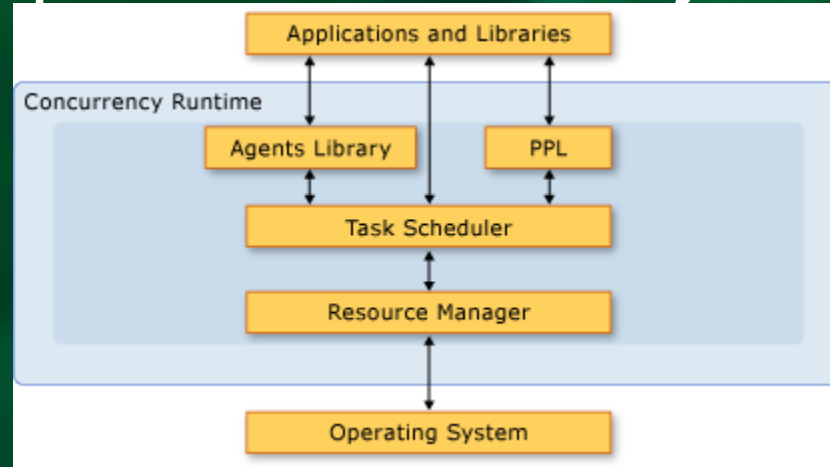
Group

Agenda

- Prvky z C++0x
 - Auto
 - Decltype
 - Lambda
 - ...
- Novinky v MFC
 - Class wizard, Ribbon designer
- **Concurrency Runtime**
 - Parallel Pattern Library
 - Async agent library

Concurrency runtime

- Podpora vláken, locků, paralelních algoritmů, paralelních kontejnerů a message



- PPL – paralelní algoritmy a kontejnery
- Agent library – in-process message passing
- Task scheduler – plánování úloh
- Resource manager – efektivní využití zdrojů

Parallel pattern library

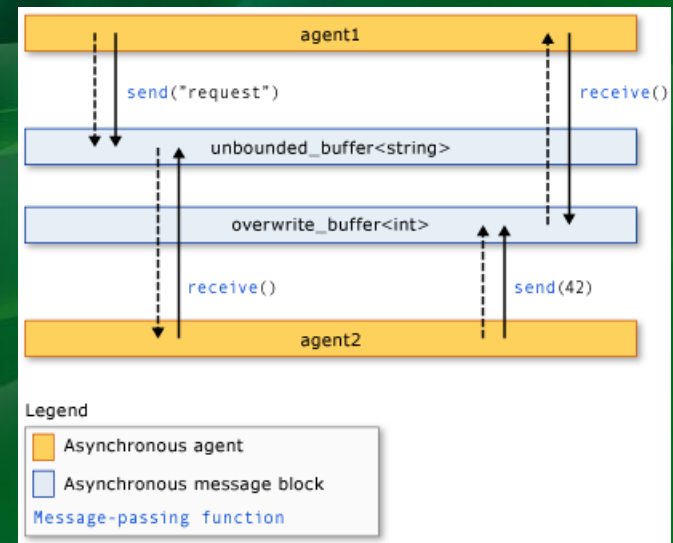
- Paralelní algoritmy
 - `parallel_for_each`, `parallel_for`
 - Strukturovaný paralelismus
- Tasks
 - Úlohy (funkce)
 - Scheduler, work-stealing queue
 - Nestrukturovaný paralelismus
- Paralelní kontejnery a objekty
 - `Concurrent_vector`, `concurrent_queue`, `combinable`
- Cancellation

PPL

demo

Asynchronous agent library

- Agent – entita zpracovávající data
- Message – data předávaná mezi agenty
- Message Passing Functions
 - Funkce posílající zprávy mezi agenty
- Podobnost s MPI – clustery



Shrnutí

- Prvky z C++0x
 - Auto
 - Decltype
 - Lambda
 - ...
- Novinky v MFC
 - Class wizard, Ribbon designer
- Concurrency Runtime
 - Parallel Pattern Library
 - Async agent library

Q & A

- Email: jakub@jcermak.cz
- <http://jcermak.cz/Prednasky.aspx>

Microsoft[®]

Your potential. Our passion.[™]

© 2007 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries. The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.

MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.