

# Tools To Ease Cross-Platform C++ Development

SIMON BRAND (THEY/THEM)

C++ DEVELOPER ADVOCATE

MICROSOFT

# Goal

2

- ▶ Develop a small cross platform application live

# Goal

3

- ▶ Develop a small cross platform application live
  - ▶ Target Windows and Linux

# Goal

4

- ▶ Develop a small cross platform application live
  - ▶ Target Windows and Linux
  - ▶ Demonstrate dependency handling

# Goal

5

- ▶ Develop a small cross platform application live
  - ▶ Target Windows and Linux
  - ▶ Demonstrate dependency handling
  - ▶ Build something powerful

# Goal

6

- ▶ Develop a small cross platform application live
  - ▶ Target Windows and Linux
  - ▶ Demonstrate dependency handling
  - ▶ Build something powerful
  - ▶ ...in an hour

# Build a Compiler!

# Tools To Ease Cross-Platform C++ Development

SIMON BRAND (THEY/THEM)

C++ DEVELOPER ADVOCATE

MICROSOFT



# Write a Compiler in an Hour or Cry Trying

SIMON BRAND (THEY/THEM)

C++ DEVELOPER ADVOCATE

MICROSOFT

# Source Language

, >, < [ > - < - ] > .



[illegible]

< > + - , . [ ]





# Move pointer left





# Move pointer left



# Move pointer left







# Move pointer right



# Move pointer right



# Move pointer right







[illegible]

# Increment



0000000000000000000002000000000000000000000000000

# Increment

11/11/2016

[illegible]

# Decrement

11/11/2016

[illegible]

# Decrement

11/11/2016



# Decrement

,



# Read from stdin

,

[illegible]

# Read from stdin

# Write to stdout



[[ ]]

[illegible]

# Loop while \*cell > 0

$$[-]$$

`0000000000000000000000000000^0000000000000000000000`

# Loop while \*cell > 0

$$[-]$$
[illegible]

# Loop while \*cell > 0

$$[-]$$

$\underbrace{00000000000000000000000000000000}_{\text{padding}} \quad \overset{\wedge}{1}000000000000000000000000$

# Loop while \*cell > 0

$$[-]$$


# Loop while \*cell > 0

< > + - , . [ ]

# Target Language: x64

# Tools

40

- ▶ Visual Studio
- ▶ CMake
- ▶ WSL
- ▶ Vcpkg



# Tools

41

- ▶ Visual Studio
- ▶ CMake
- ▶ WSL
- ▶ Vcpkg

## Alternatives:

- ▶ Visual Studio Code, CLion
- ▶ Meson, Build2
- ▶ VirtualBox, Docker
- ▶ Conan, Buckaroo

# Demo: Getting Started

WSL

Windows  
Subsystem for  
Linux

# WSL

44

- ▶ Install Linux distributions easily.

- ▶ Install Linux distributions easily.
- ▶ Run Bash shell scripts and Linux command-line applications like:

- ▶ Install Linux distributions easily.
- ▶ Run Bash shell scripts and Linux command-line applications like:
  - ▶ Tools: vim, emacs, tmux
  - ▶ Languages: Javascript/node.js, Ruby, Python, C/C++, C# & F#, Rust, Go, etc.
  - ▶ Services: sshd, MySQL, Apache, lighttpd

- ▶ Install Linux distributions easily.
- ▶ Run Bash shell scripts and Linux command-line applications like:
  - ▶ Tools: vim, emacs, tmux
  - ▶ Languages: Javascript/node.js, Ruby, Python, C/C++, C# & F#, Rust, Go, etc.
  - ▶ Services: sshd, MySQL, Apache, lighttpd
- ▶ Install additional software using Linux package managers.

- ▶ Install Linux distributions easily.
- ▶ Run Bash shell scripts and Linux command-line applications like:
  - ▶ Tools: vim, emacs, tmux
  - ▶ Languages: Javascript/node.js, Ruby, Python, C/C++, C# & F#, Rust, Go, etc.
  - ▶ Services: sshd, MySQL, Apache, lighttpd
- ▶ Install additional software using Linux package managers.
- ▶ Invoke Windows applications using a Unix-like command-line shell.



WSL is not a VM!

# Installing WSL

50

- ▶ Run:
  - ▶ `Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux`
- ▶ Install distribution from the Microsoft Store

# Demo: WSL

# Vcpkg

52

► <https://github.com/microsoft/vcpkg>

# Vcpkg

53

- ▶ <https://github.com/microsoft/vcpkg>
- ▶ Package manager for C and C++ libraries

# Vcpkg

54

- ▶ <https://github.com/microsoft/vcpkg>
- ▶ Package manager for C and C++ libraries
- ▶ Runs on Windows, Linux, and MacOS

# Vcpkg

55

- ▶ <https://github.com/microsoft/vcpkg>
- ▶ Package manager for C and C++ libraries
- ▶ Runs on Windows, Linux, and MacOS
- ▶ Repository contains >1000 packages

# Vcpkg

56

- ▶ <https://github.com/microsoft/vcpkg>
- ▶ Package manager for C and C++ libraries
- ▶ Runs on Windows, Linux, and MacOS
- ▶ Repository contains >1000 packages
- ▶ Packages contributed by the Microsoft and the community



- ▶ All libraries and their dependency chains are tested

# Vcpkg

58

- ▶ All libraries and their dependency chains are tested
- ▶ You can export binary packages

# Vcpkg

59

- ▶ All libraries and their dependency chains are tested
- ▶ You can export binary packages
- ▶ It supports private libraries

# Vcpkg

60

- ▶ All libraries and their dependency chains are tested
- ▶ You can export binary packages
- ▶ It supports private libraries
- ▶ You can (and should) pin your dependency versions

# Demo: Vckpgg

- ▶ Vcpkg - <https://github.com/microsoft/vcpkg>
- ▶ WSL - <https://docs.microsoft.com/en-us/windows/wsl/install-win10>
- ▶ CMake Support in Visual Studio - <https://devblogs.microsoft.com/cppblog/cmake-support-in-visual-studio/>
- ▶ Targeting WSL from Visual Studio - <https://devblogs.microsoft.com/cppblog/c-with-visual-studio-2019-and-windows-subsystem-for-linux-wsl/>
- ▶ Effective CMake – Daniel Pfeifer - <https://www.youtube.com/watch?v=bsXLMQ6WgIk>
- ▶ Don't Package Your Libraries, Write Packageable Libraries! – Robert Schumacher - <https://www.youtube.com/watch?v=sBP17HQAQjk>