
Mastering CMake

Sixth Edition

Ken Bill
Martin & Hoffman

With contributions from:

Andy Cedilnik, David Cole, Marcus Hanwell, Julien Jomier, Brad King, Robert Maynard,
Alex Neundorff

Published by Kitware Inc.

Join the CMake community at www.cmake.org

Contents

| | | |
|-------------------------|--|------------------|
| <u>CHAPTER 1</u> | <u>WHY CMAKE?</u> | <u>1</u> |
| | The History of CMake | 3 |
| | Why Not Use Autoconf? | 3 |
| | Why Not Use JAM, qmake, SCons, or ANT? | 4 |
| | Why Not Script It Yourself? | 4 |
| | On What Platforms Does CMake Run? | 4 |
| | How Stable is CMake? | 5 |
| <u>CHAPTER 2</u> | <u>GETTING STARTED</u> | <u>7</u> |
| | Getting and Installing CMake on Your Computer | 7 |
| | <i>UNIX and Mac Binary Installations</i> | 7 |
| | <i>Windows Binary Installation</i> | 7 |
| | Building CMake Yourself | 8 |
| | Basic CMake Usage and Syntax | 8 |
| | Hello World for CMake | 10 |
| | How to Run CMake? | 11 |
| | <i>Running CMake's Qt Interface</i> | 11 |
| | <i>Running the ccmake Curses Interface</i> | 14 |
| | <i>Running CMake from the Command Line</i> | 16 |
| | <i>Specifying the Compiler to CMake</i> | 16 |
| | <i>Dependency Analysis</i> | 17 |
| | Editing CMakeLists Files | 18 |
| | Setting Initial Values for CMake | 18 |
| | Building Your Project | 19 |
| <u>CHAPTER 3</u> | <u>KEY CONCEPTS</u> | <u>21</u> |
| | Main Structures | 21 |
| | Targets | 24 |
| | Source Files | 25 |
| | Directories, Generators, Tests, and Properties | 26 |
| | Variables and Cache Entries | 27 |
| | Build Configurations | 32 |
| <u>CHAPTER 4</u> | <u>WRITING CMAKELISTS FILES</u> | <u>33</u> |
| | CMake Language | 33 |
| | Basic Commands | 34 |
| | Flow Control | 34 |
| | Regular Expressions | 41 |

| | |
|--|--|
| Checking Versions of CMake | 43 |
| Using Modules | 44 |
| <i>Using CMake with SWIG</i> | 45 |
| <i>Using CMake with Qt</i> | 46 |
| <i>Using CMake with FLTK</i> | 47 |
| Policies | 48 |
| <i>Updating a Project For a New Version of CMake</i> | 51 |
| Linking Libraries | 54 |
| <i>Specifying Optimized or Debug Libraries with a Target</i> | 56 |
| Advanced Linking | 57 |
| Object Libraries | 58 |
| Shared Libraries and Loadable Modules | 59 |
| Shared Library Versioning | 64 |
| Installing Files | 66 |
| <i>Installing Prerequisite Shared Libraries</i> | 76 |
| Advanced Commands | 81 |
| <u>CHAPTER 5</u> | <u>SYSTEM INSPECTION</u> |
| | 85 |
| Using Header Files and Libraries | 85 |
| System Properties | 87 |
| Finding Packages | 92 |
| Built-in Find Modules | 93 |
| How to Pass Parameters to a Compilation? | 95 |
| How to Configure a Header File | 97 |
| Creating CMake Package Configuration Files | 99 |
| CMake Package Registry | 102 |
| <u>CHAPTER 6</u> | <u>CUSTOM COMMANDS AND TARGETS</u> |
| | 105 |
| Portable Custom Commands | 105 |
| Using add_custom_command on a Target | 107 |
| <i>How to Copy an Executable Once it is Built?</i> | 108 |
| Using add_custom_command to Generate a File | 108 |
| <i>Using an Executable to Build a Source File</i> | 109 |
| Adding a Custom Target | 110 |
| Specifying Dependencies and Outputs | 113 |
| When There Isn't One Rule For One Output | 113 |
| <i>A Single Command Producing Multiple Outputs</i> | 113 |
| <i>Having One Output That Can Be Generated By Different Commands</i> | 114 |
| <u>CHAPTER 7</u> | <u>CONVERTING EXISTING SYSTEMS TO CMAKE</u> |
| | 115 |
| Source Code Directory Structures | 115 |
| Build Directories | 117 |

| | |
|--|-----|
| Useful CMake Commands When Converting Projects | 119 |
| Converting UNIX Makefiles | 120 |
| Converting Autoconf Based Projects | 121 |
| Converting Windows Based Workspaces | 122 |

CHAPTER 8 CROSS-COMPILING WITH CMAKE 125

| | |
|---|-----|
| Toolchain Files | 126 |
| <i>Finding External Libraries, Programs and Other Files</i> | 128 |
| System Inspection | 130 |
| <i>Using Compile Checks</i> | 131 |
| Running Executables Built in the Project | 133 |
| Cross-Compiling Hello World | 136 |
| Cross-Compiling for a Microcontroller | 140 |
| Cross Compiling an Existing Project | 143 |
| Cross-Compiling a Complex Project - VTK | 145 |
| Some Tips and Tricks | 147 |

CHAPTER 9 PACKAGING WITH CPACK 149

| | |
|--|-----|
| CPack Basics | 149 |
| <i>Simple Example</i> | 150 |
| <i>What Happens When CPack.cmake Is Included?</i> | 151 |
| <i>Adding Custom CPack Options</i> | 152 |
| <i>Options Added by CPack</i> | 153 |
| CPack Source Packages | 154 |
| CPack Installer Commands | 155 |
| CPack for Windows Installer NSIS | 156 |
| <i>CPack Variables Used by CMake for NSIS</i> | 156 |
| <i>Creating Windows Short Cuts in the Start Menu</i> | 162 |
| <i>Advanced NSIS CPack Options</i> | 162 |
| <i>Setting File Extension Associations With NSIS</i> | 163 |
| <i>Installing Microsoft Run Time Libraries</i> | 164 |
| <i>CPack Component Install Support</i> | 164 |
| CPack for Cygwin Setup | 174 |
| CPack for Mac OS X PackageMaker | 177 |
| CPack for Mac OS X Drag and Drop | 179 |
| CPack for Mac OS X X11 Applications | 181 |
| CPack for Debian Packages | 183 |
| CPack for RPM | 184 |
| CPack Files | 184 |

CHAPTER 10 AUTOMATION & TESTING WITH CMAKE 185

| | |
|--------------------------------------|-----|
| Testing with CMake, CTest, and CDash | 185 |
|--------------------------------------|-----|

| | |
|---|-----|
| How Does CMake Facilitate Testing? | 186 |
| Additional Test Properties | 187 |
| Testing Using CTest | 189 |
| Using CTest to Drive Complex Tests | 191 |
| Handling a Large Number of Tests | 192 |
| Managing Test Data | 194 |
| Producing Test Dashboards | 196 |
| <i>Adding CDash Dashboard Support to a Project</i> | 199 |
| <i>Client Setup</i> | 203 |
| Customizing Dashboards for a Project | 205 |
| <i>Dashboard Submissions Settings</i> | 205 |
| <i>Filtering Errors and Warnings</i> | 206 |
| <i>Adding Notes to a Dashboard</i> | 208 |
| Setting up Automated Dashboard Clients | 209 |
| <i>Settings for Continuous Dashboards</i> | 213 |
| <i>Variables Available in CTest Scripts</i> | 215 |
| <i>Limitations of Traditional CTest Scripting</i> | 215 |
| Advanced CTest Scripting | 216 |
| <i>Extended CTest Scripting</i> | 216 |
| Setting up a Dashboard Server | 221 |
| <i>CDash Server</i> | 221 |
| <i>Advanced Server Management</i> | 223 |
| <i>Build Groups</i> | 226 |
| <i>Email</i> | 228 |
| <i>Sites</i> | 229 |
| <i>Graphs</i> | 230 |
| <i>Adding Notes to a Build</i> | 231 |
| <i>Logging</i> | 232 |
| <i>Test Timing</i> | 232 |
| <i>Mobile Support</i> | 233 |
| <i>Backing up CDash</i> | 233 |
| <i>Upgrading CDash</i> | 234 |
| <i>CDash Maintenance</i> | 235 |
| Subprojects | 236 |
| <i>Using ctest_submit with PARTS and FILES</i> | 239 |
| <i>Splitting Your Project into Multiple Subprojects</i> | 240 |

CHAPTER 11 PORTING CMAKE TO NEW PLATFORMS AND LANGUAGES245

| | |
|------------------------------|-----|
| The Determine System Process | 245 |
| The Enable Language Process | 246 |
| Porting to a New Platform | 248 |
| Adding a New Language | 250 |
| Rule Variable Listing | 251 |
| <i>General Tag Variables</i> | 251 |

| | |
|---|------------|
| <i>Language Specific Information</i> | 252 |
| Compiler and Platform Examples | 252 |
| <i>Como Compiler</i> | 252 |
| <i>Borland Compiler</i> | 253 |
| Extending CMake | 254 |
| <i>Creating a Loaded Command</i> | 254 |
| <i>Using a Loaded Command</i> | 255 |
| CHAPTER 12 TUTORIALS | 259 |
| A Basic Starting Point (Step 1) | 259 |
| <i>Adding a Version Number and Configured Header File</i> | 260 |
| Adding a Library (Step 2) | 262 |
| Installing and Testing (Step 3) | 264 |
| Adding System Introspection (Step 4) | 266 |
| Adding a Generated File and Generator (Step 5) | 267 |
| Building an Installer (Step 6) | 271 |
| Adding Support for a Dashboard (Step 7) | 272 |
| CHAPTER 13 APPENDIX A - VARIABLES | 273 |
| Variables That Change Behavior | 273 |
| Variables That Describe the System | 280 |
| Variables for Languages | 283 |
| Variables That Control the Build | 288 |
| Variables That Provide Information | 293 |
| CHAPTER 14 APPENDIX B - COMMAND LINE REFERENCE | 302 |
| CMake Command Line Options | 302 |
| <i>Name</i> | 302 |
| <i>Usage</i> | 302 |
| <i>Description</i> | 302 |
| <i>Options</i> | 303 |
| CMake Generators | 308 |
| CTest Command Line Options | 312 |
| CPack Command Line Options | 317 |
| CPack Generators | 317 |
| CHAPTER 15 APPENDIX C – GENERATOR EXPRESSIONS | 318 |
| CMake Generator Expressions | 319 |

| | | |
|--|--|-------------------|
| <u>CHAPTER 16</u> | <u>APPENDIX D – LISTFILE COMMANDS</u> | <u>321</u> |
| Current Commands | | 321 |
| Compatibility Commands | | 399 |
| <u>CHAPTER 17</u> | <u>APPENDIX E – SELECTED MODULES</u> | <u>407</u> |
| Standard CMake Modules | | 407 |
| <u>CHAPTER 18</u> | <u>APPENDIX F - PROPERTIES</u> | <u>583</u> |
| CMake Properties - Properties supported by CMake, the Cross-Platform Makefile Generator. | | 583 |
| Properties of Global Scope | | 583 |
| Properties on Directories | | 587 |
| Properties on Targets | | 591 |
| Properties on Tests | | 615 |
| Properties on Source Files | | 617 |
| Properties on Cache Entries | | 620 |
| <u>CHAPTER 19</u> | <u>APPENDIX G – CMAKE POLICIES</u> | <u>623</u> |
| <u>CHAPTER 20</u> | <u>INDEX</u> | <u>635</u> |