Building, Testing, and Deploying Software in a Cross-Platform Development Environment

Julien Jomier
julien.jomier@kitware.com

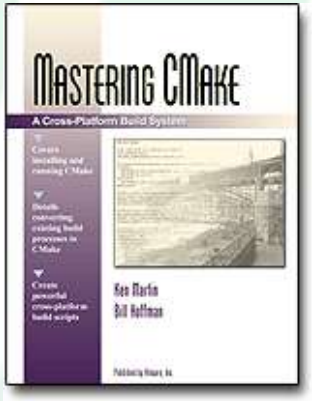
About Kitware

- Software Development Company
- Founded in 1998
- 110+ employees and growing
- Focus in
  - Scientific computing
  - Large data visualization
  - Medical image processing
  - Informatics
  - Computer vision
  - Scientific data publishing
  - Quality software processes

- **Primarily work in Open Source**

Kitware

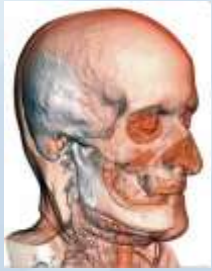
About Kitware



- Provide training and support
- Do custom development
- Sell technical books
- Host and maintain projects including:

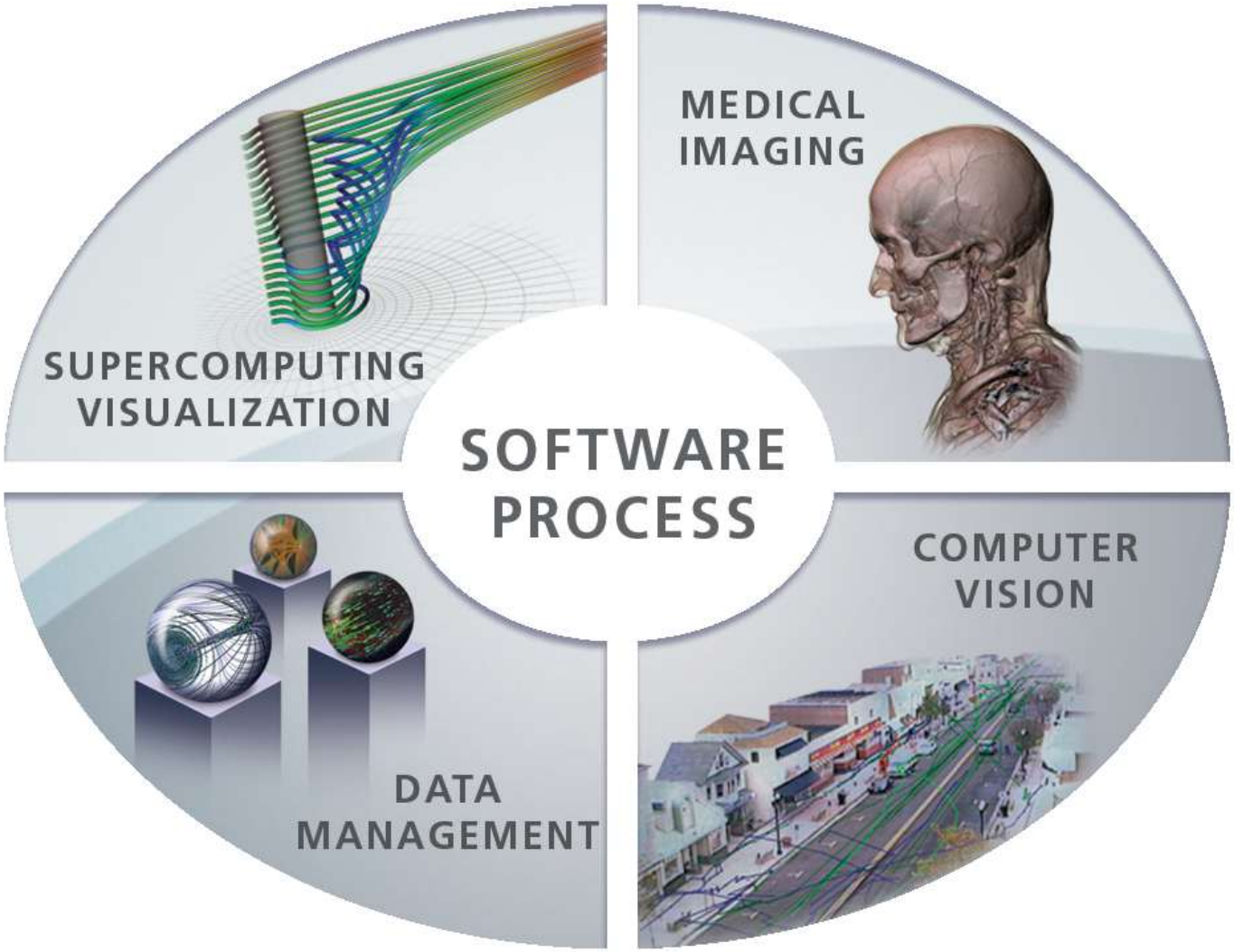
VTK	The Visualization Toolkit
CMake	A platform agnostic build system
ITK	Insight Segmentation and Registration Toolkit
ParaView	A parallel visualization application

- Primary (but not sole) developers of each project

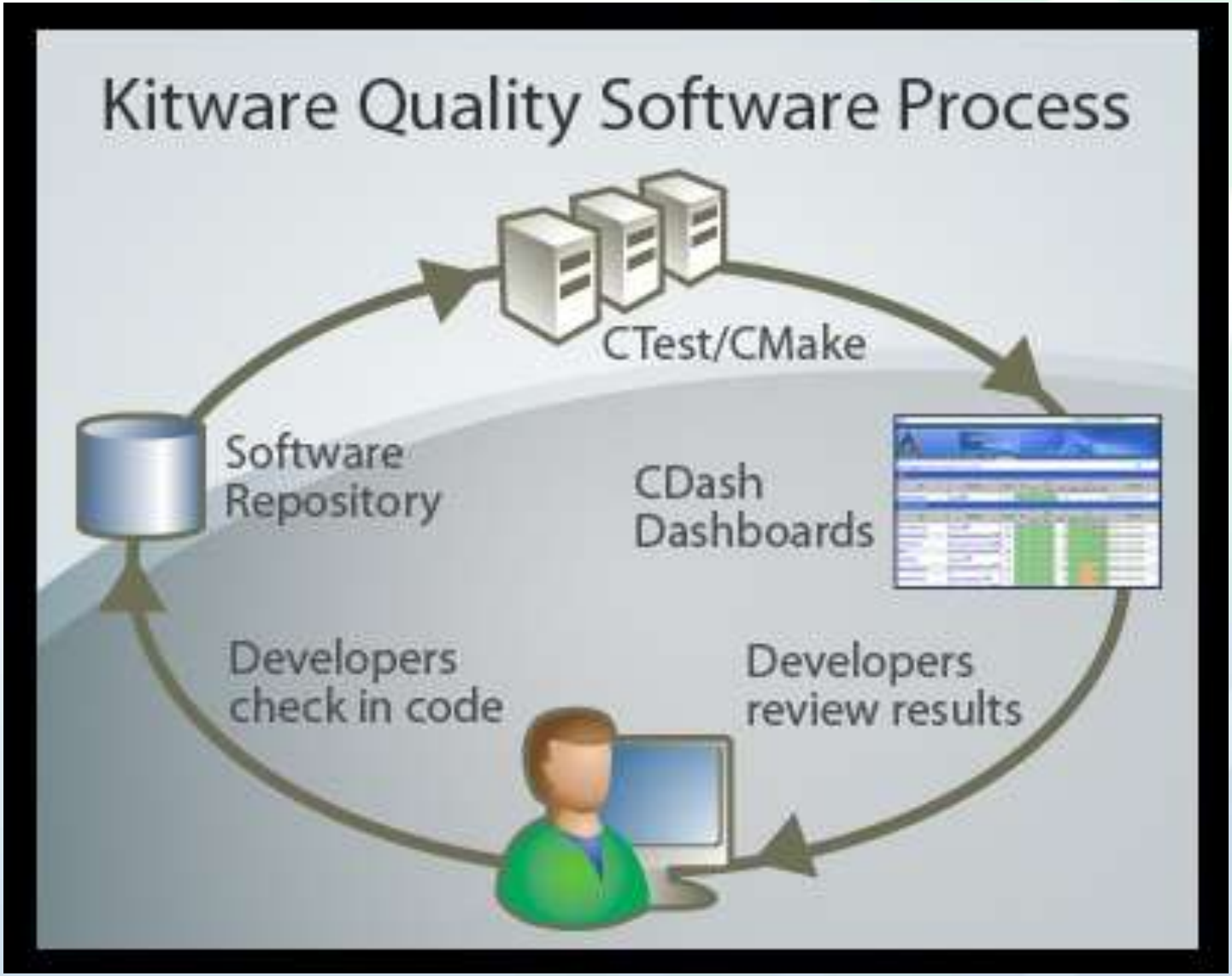


Kitware

Kitware: Core Technologies



Kitware


Kitware Quality Software Process
CTest/CMake
Software Repository
CDash Dashboards
Developers check in code
Developers review results

Kitware

Overview

- What is CMake?
- Building with CMake
- Testing with CTest/CDash
- Packaging with CPack

What is CMake?

- CMake is the **cross-platform**, **open-source build system** that lets you use the **native development tools** you love the most.
- It's a build system **generator**
- It takes **plain text files** as input that describe your project and **produces** project files or make files for use with a wide variety of **native development tools.**
- Family of Software Development Tools
  - Build – CMake
  - Test – CTest/CDash
  - Package – CPack

Kitware

CMake: History

- Built for the Insight Segmentation and Registration Toolkit (ITK) http://www.itk.org
- Funded by National Library of Medicine (NLM): part of the Visible Human Project
- Release-1-0 branch created in late 2001
- Other available tools were insufficient: pcmaker (vtk3.2), autoconf, Apache ANT, qmake (Qt), JAM
- CMake provides the combination of native build tool use and platform introspection that none of the others provide.
- CMake-2-8 branch created in September, 2009

Why CMake? It's easy, and works well

- A build system that just works
- A build system that is easy to use cross platform

Typical Project without CMake (curl)

```
$ ls
CHANGES          RELEASE-NOTES  curl-config.in  missing
CMake            acinclude.m4   curl-style.el   mkinstalldirs
CMakeLists.txt   aclocal.m4     depcomp         notes
build            docs           notes~
COPYING          buildconf      include         packages
CVS              buildconf.bat  install-sh      reconf
ChangeLog        compile        lib             sample.emacs
Makefile         config.guess   libcurl.pc.in   src
Makefile.am      config.sub     ltmain.sh       tests
Makefile.in      configure      m4              vc6curl.dsw
README           configure.ac   maketgz

$ ls src/
CMakeLists.txt   Makefile.riscos  curlsrc.dsp  hugehelp.h     version.h
CVS              Makefile.vc6     curlsrc.dsw  macos          writeenv.c
Makefile.Watcom  Makefile.vc8     curlutil.c   main.c         writeenv.h
Makefile.am      config-amigaos.h curlutil.h   makefile.amiga writeout.c
Makefile.b32     config-mac.h     getpass.c    makefile.dj    writeout.h
Makefile.in      config-riscos.h  getpass.h    mkhelp.pl
Makefile.inc     config-win32.h   homedir.c    setup.h
Makefile.m32     config.h.in      homedir.h    urlglob.c
Makefile.netware curl.rc          hugehelp.c   urlglob.h
```

Kitware

Why CMake? It's fast

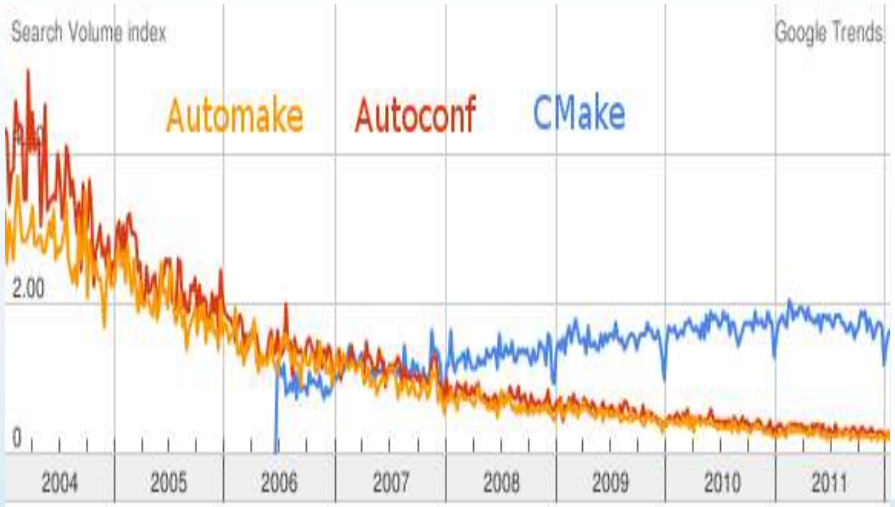
http://blog.qgis.org/?q=node/16 : "I was quite surprised with the speed of building Quantum GIS codebase in comparison to Autotools. "

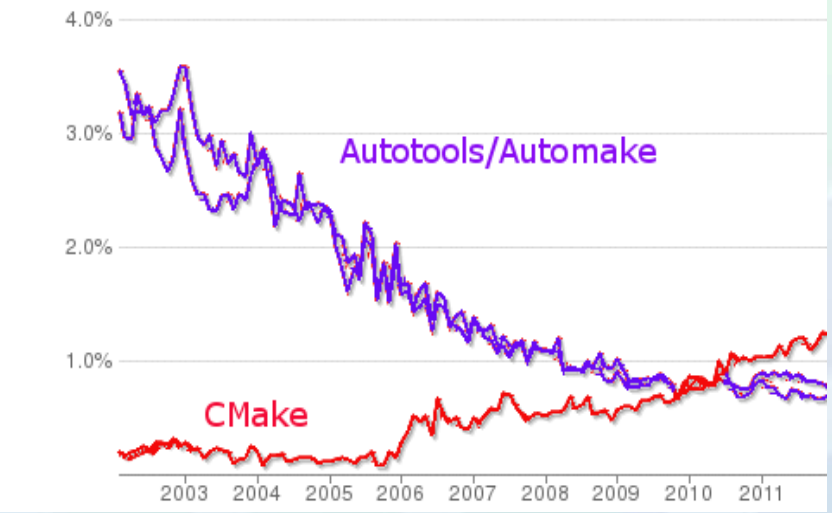
Task	CMake	Autotools
Configure	0:08	Automake: 0:41 Configure: 0:20
Make	12:15	21:16
Install	0:20	0:36
Total	12:43	22:43

Kitware

Why CMake? Everyone is using it

KDE 2006 – Tipping Point!





- Google Search Trends and Ohloh comparisons with auto*
- 1200+ downloads per day from www.cmake.org
- Major Linux distributions and Cygwin provide CMake packages
- KDE, Second Life, Boost (Experimentally), many others

Kitware

How CMake Changes The Way We Build C++

- Boost aims to give C++ a set of useful libraries like Java, Python, and C#
- CMake aims to give C++ compile portability like the compile once and run everywhere of Java, Python, and C#
  - Same build tool and files for all platforms
  - Easy to mix both large and small libraries

Kitware

Who Is Involved?

Users

- KDE
- Second Life
- ITK
- VTK
- ParaView
- Trilinos
- Scribus
- Boost (Experimentally)
- MySQL
- LLVM
- many more

Supporters

- Kitware
- ARL
- National Library of Medicine
- Sandia National Labs
- Los Alamos National Labs
- NAMIC

Kitware

ARC Centre of Excellence for Autonomous Systems- Australia

The Argo is fully automated with the onboard computers all using code built using CMake. The Helicopter is semi-automated with the pilot following a course determined by the onboard computers interacting with the Argo. Everything uses CMake.

Kitware

CMake Documentation

- Mastering CMake Book
- Web Page: www.cmake.org
- http://www.cmake.org/Wiki/CMake
- mailing list: cmake@cmake.org
  - Full reference documentation
  - http://www.cmake.org/cmake/help/documentation.html
- Ships HTML, man, and command line help

Kitware

CMake Features

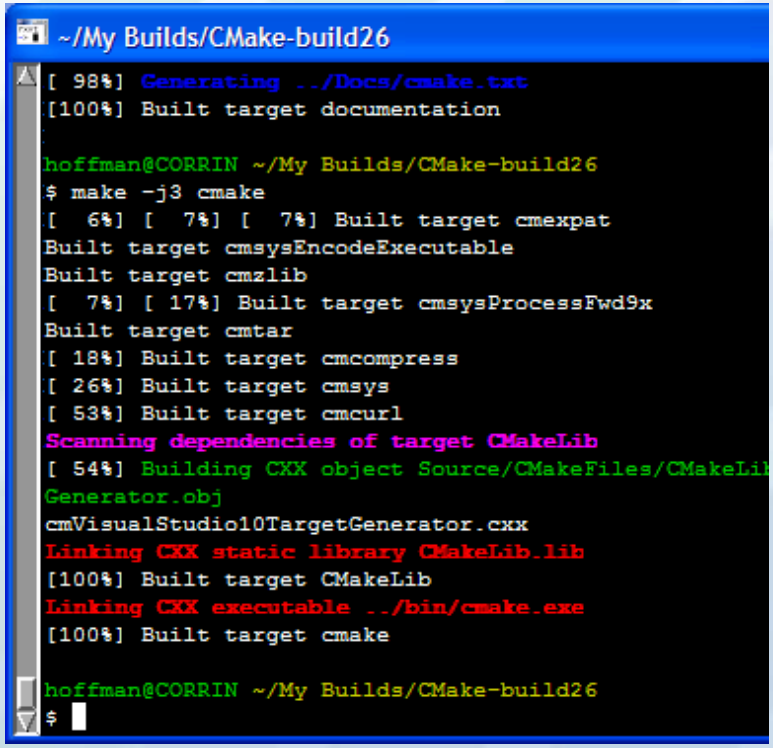
- One simple language for all platforms
  - Windows, Mac, Linux, UNIX variants
  - HPC/Embedded platforms via cross-compilation (ParaView/python)
- Generates native build systems
  - Makefiles (GNU, NMake, Borland, etc.)
  - KDevelop, Eclipse
  - Visual Studio 6,7,8,9 IDE
  - Xcode
- Out-of-source build trees leave source clean
- Interactive configuration via GUI
- Multiple configurations (Debug, Release, etc.)

CMake Features - continued

- Built-in rules for common targets
  - Executables
  - Shared Libraries / DLLs
  - Static Libraries (archives)
  - OS X Frameworks and App Bundles
- Custom rules for other targets
  - Generated Documentation
  - Generated sources and headers
- Configuration rules
  - System introspection
  - Persistent variables (options, cached results)
  - Configured header files

CMake Features - continued

- Automatic analysis
  - Implicit dependencies (C, C++, Fortran)
  - Transitive link dependencies
  - Ordering of linker search path and RPATH
- Advanced Makefile generation
  - Modular, Fast, Parallel
  - Color and progress display
  - Help targets – make help
  - Preprocessor targets – make foo.i
  - Assembly targets – make foo.s



Kitware

Input to CMake

- Simple scripting language in CMakeLists.txt file(s)
- Built-in commands for common rules
  - `add_library(MyLib MyLib.cxx)`
  - `add_executable(MyExe MyMain.cxx)`

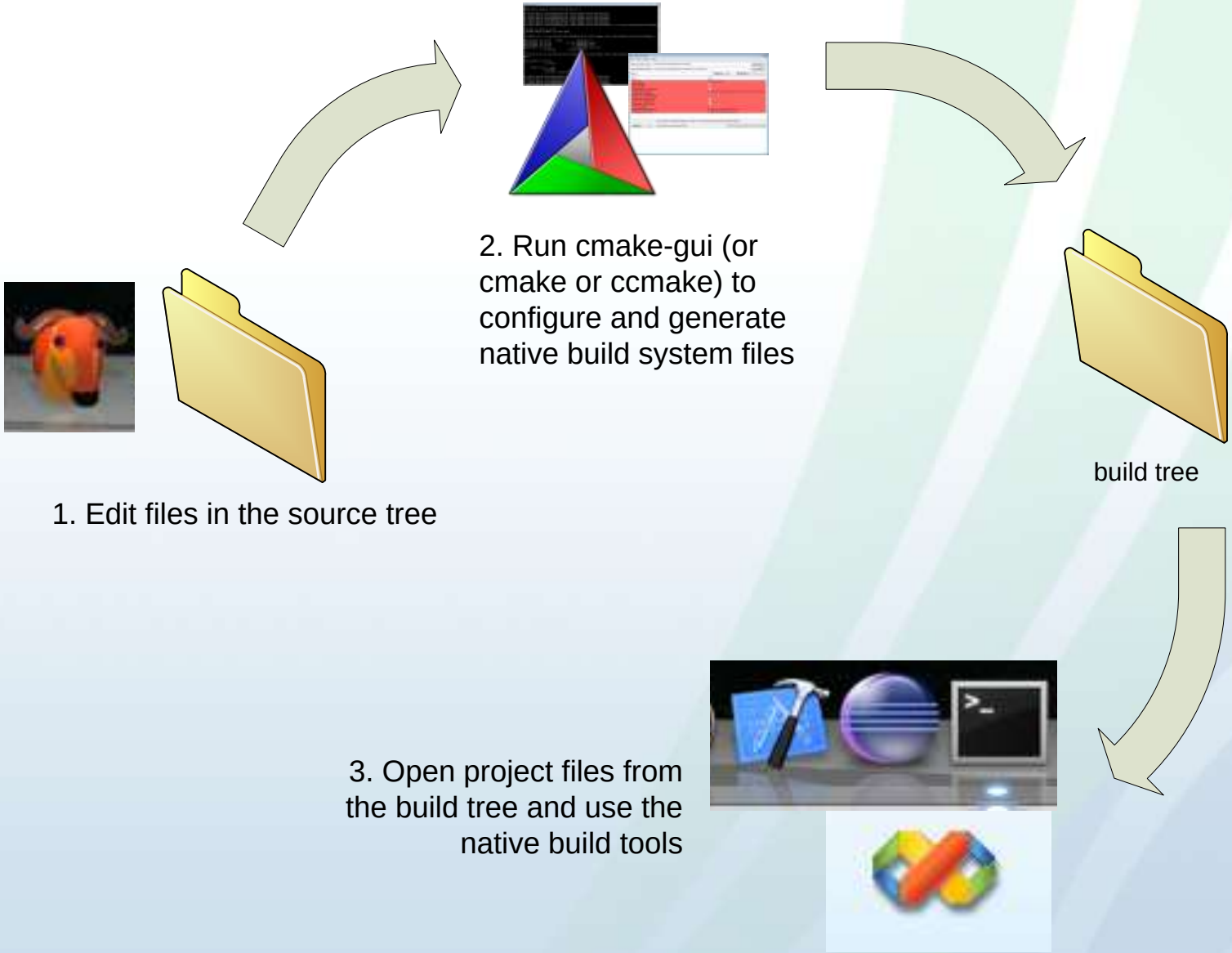
- Example project using Boost:

```
cmake_minimum_required(VERSION 2.6)
project(MyProject)
find_package(Boost REQUIRED thread signals)
include_directories(${Boost_INCLUDE_DIRS})
add_executable(MyExe MyProjectMain.cxx)
target_link_libraries(MyExe ${Boost_LIBRARIES})
```

Installing CMake

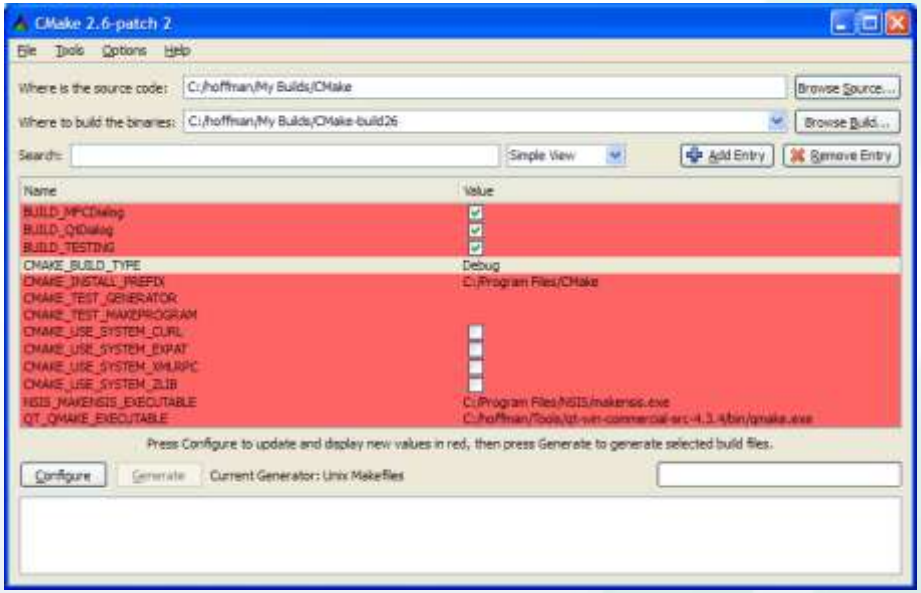
- Easy to Get CMake
  - http://www.cmake.org/cmake/resources/software.html
  - Many Linux distributions and Cygwin include CMake packages
    - apt-get install cmake
- Installing CMake from www.cmake.org
  - Windows Binary installers
  - Linux Binaries – can be installed anywhere, (don’t need root)
  - Mac
  - Other UNIX on download page
  - Source can bootstrap on any Unix platform

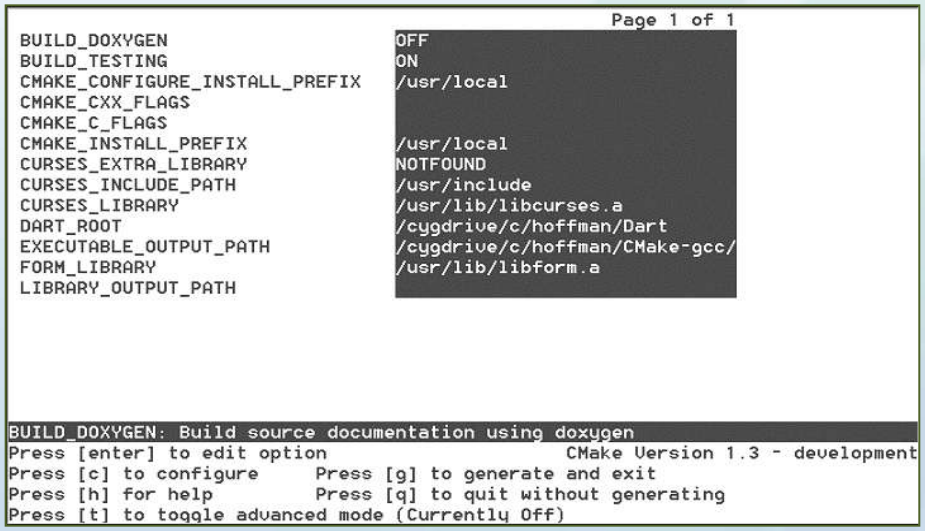
CMake Workflow



Kitware

Cache Editors: cmake-gui (qt), ccmake (curses)





Kitware

Running CMake From The Command Line

- Useful for scripted builds or for projects with no options or with options correctly set by default on the first configure

```
#CC=gcc; CXX=g++
#CFLAGS, CXXFLAGS
cd MyProjectSourceDir
mkdir ../MyProjectSourceDir-build
cd ../MyProjectSourceDir-build
cmake ../MyProjectSourceDir

(cmake -Dvar=value)
```

Kitware

CMake Scripts

- cmake –E command
  - Cross platform command line utility
  - Ex. Copy file, Remove file, Compare and conditionally copy, time etc
- cmake –P script.cmake
  - Cross platform scripting utility
  - Does not generate cmake_cache
  - Ignores commands specific to generating build environment

CTest / CDash

Kitware

Automatic Testing Benefits

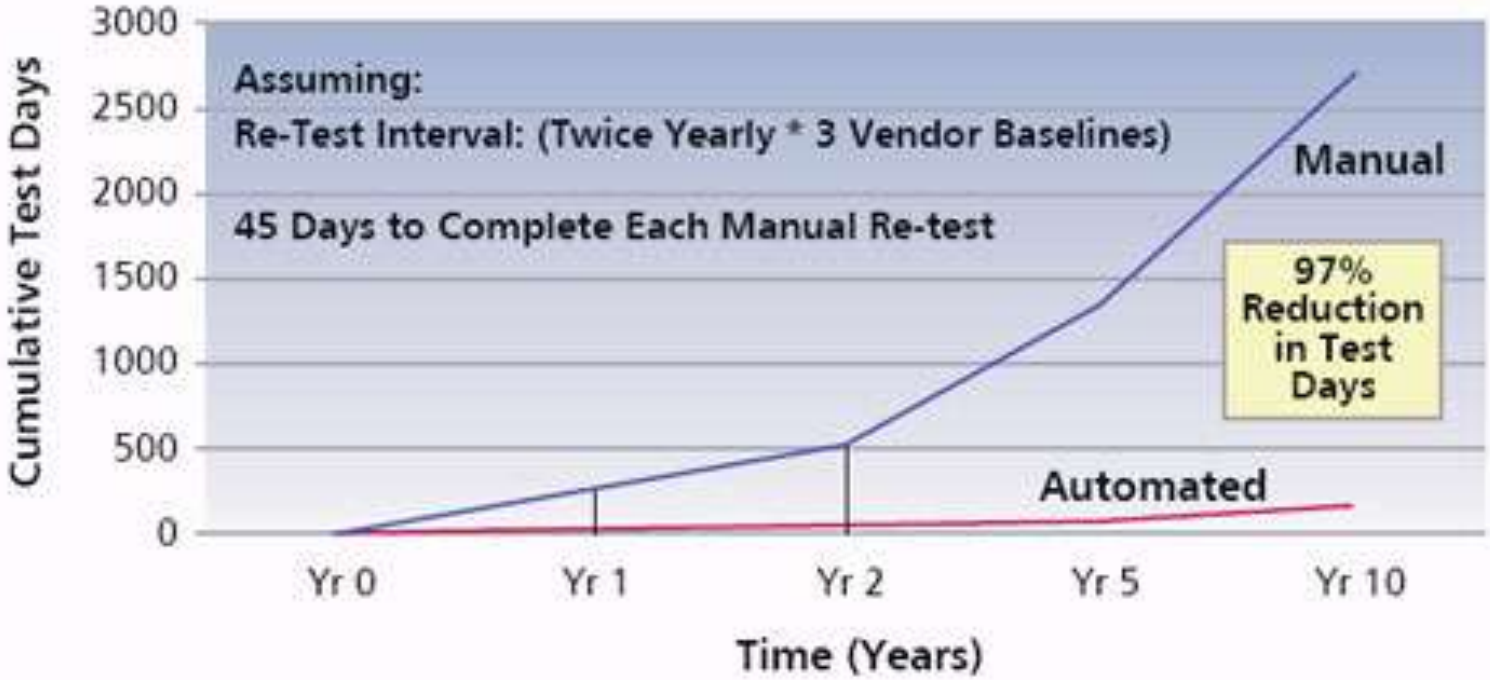


Figure 2. "Example Automated Software Testing Savings over Time"

"Automated Software Testing,"
1999, Dustin, et al, Addison Wesley

Kitware

Testing with CMake, CTest and CDash

- Testing command in CMake
  - add_test (testname exename arg1 arg2 arg3 …)
  - Executable is expected to return 0 for passed
  - Can set other test passing conditions based on output matching.
- ctest – an executable that is distributed with cmake that can run tests in a project.
  - Used for continuous integration testing
  - Client for CDash
  - Can be use for both CMake based projects and other build systems

Kitware

CDash Dashboard www.cdash.org

CDash - CMake - Mozilla Firefox

http://public.kitware.com/CDash/index.php?project=CMake

CMAKE Dashboard

DASHBOARD CALENDAR PREVIOUS CURRENT NEXT PROJECT

Nightly Changes as of 2008-02-20 21:00:00 EST

Style [Nightly Expected] [Nightly 2.4 Release] [Nightly] [Continuous] [Experimental] [Coverage] [Dynamic Analysis]

Site	Build Name	Update	Cfg	Build			Test					Build Date
				Error	Warn	Min	NotRun	Fail	Pass	NA	Min	
insight.journal.kitware	KWStyle	7	0	0	0	0						2008-02-21 02:28:33 EST

Nightly Expected [Style] [Nightly 2.4 Release] [Nightly] [Continuous] [Experimental] [Coverage] [Dynamic Analysis]

Site	Build Name	Update	Cfg	Build			Test					Build Date
				Error	Warn	Min	NotRun	Fail	Pass	NA	Min	
Titanium.IMTS.us	Linux64-Rocks-ICC-Rel	107	0	0	0	10.7	0	0	93	0	8.1	2008-02-21 10:23:00 EST
krondor.kitware	Darwin-c++	0	0	0	0	59.1	0	1	95	0	55.2	2008-02-21 09:56:00 EST
dash8.kitware	Linux64-g++332	0	0	0	0	6.3	0	0	95	0	18.3	2008-02-21 08:02:00 EST
RogueResearch3	Mac10.5-CMake-Xcode-dbg-ppc64	1	0	0	0	13.1	0	0	90	0	23.4	2008-02-21 05:15:00 EST
RogueResearch3	Mac10.5-CMake-Xcode-dbg-ppc	1	0	0	0	13	0	0	90	0	23.9	2008-02-21 04:28:00 EST

Kitware

Trilinos (Multi-Package Dashboard)
http://trilinos-dev.sandia.gov/cdash/index.php

CDash - Trilinos - Mozilla Firefox

DASHBOARD

Main Project

Project

Project	Configure			Build			Test			Last submission
	Error	Warning	Pass	Error	Warning	Pass	Not Run	Fail	Pass	
Trilinos	0	0	208	1	117	91	0	6	5227	2009-04-30 12:54:32

SubProjects

Project	Configure			Build			Test			Last submission
	Error	Warning	Pass	Error	Warning	Pass	Not Run	Fail	Pass	
Teuchos	0	0	6	0	0	6	0	0	386	2009-04-30 16:59:38
RTOp	0	0	5	0	0	5	0	0	95	2009-04-30 17:00:49
Kokkos				0	0	5	0	0	10	2009-04-30 17:01:00
Epetra				0	3	2	0	0	128	2009-04-30 17:01:14
Zoltan				0	6	0	0	0	9	2009-04-30 18:08:12
Shards	0	0	5	0	5	0	0	0	20	2009-04-30 17:02:09
Intrepid	0	0	5	0	2	3	0	0	8	2009-04-30 17:10:38

Sub Projects

Kitware

Query Filters : customize views

CDash - Trilinos - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://trilinos-dev.sandia.gov/cdash/index.php?project=Trilinos&display=project&filtercount=4&showfilte

Most Visited | Getting Started | Latest Headlines | Kitware Inc. | CDash | CDash - CMB | Free Online Broken Lin... | CMake Only Errors | CMake Bugs

CMake - Cross Platform Make | Capital District Youth Soccer League | CDash - Trilinos

Trilinos

TRILINOS Dashboard

DASHBOARD | CALENDAR | PREVIOUS | CURRENT | PROJECT

No file changed as of 2009-04-30T00:00:00 MDT

Help

[Hide Filters]

Match all of the following rules:

Build Name	is	Linux-GCC-4.1.2-MPI_DEBUG
Site	is	godel.sandia.gov
Build Time	is after	2009-04-30T10:58:53 MDT
Build Time	is before	2009-04-30T12:02:16 MDT

Apply | Clear | Create Hyperlink

Nightly

Site	Build Name	Update		Configure			Build			Test				Build Time	Labels
		Files	Min	Error	Warn	Min	Error	Warn	Min	NotRun	Fail	Pass	Min		
godel.sandia.gov	Linux-GCC-4.1.2-MPI_DEBUG	0	0	0	0	0.1	0	0	1.2	0	0	40 +39	0.7	2009-04-30T11:40:16 MDT	Rythmos
godel.sandia.gov	Linux-GCC-4.1.2-	0	0	0	0	0	0	0	0.1	0	0	7	0.1	2009-04-30T11:09:49	AztecOO

Done

Kitware

CTest Command Wrappers Output

Build Time:2009-05-04T01:53:37 MDT

Found 1 Warnings

Errors are here.

Warning while building C++ object file "CMakeFiles/Kokkos_BaseSparseSolve.dir/cxx_main.cpp.o" in target Kokkos_BaseSparseSolve.	
Source File	packages/kokkos/test/BaseSparseSolve/cxx_main.cpp
Label	Kokkos
Command	"/Users/bmpersc/bin/gcc-4.3.3/bin/g++" "-mmacosx-version-min=10.5" "-ansi" "-pedantic" "-Wall" "-Wno-long-long" "-Wwrite-strings" "-g" "-O0" "-D_GLIBCXX_DEBUG" "-I/Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/BUILD/packages/kokkos/src" "-I/Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/src" "-I/Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/test/BaseSparseSolve/../BaseSparseMultiply" "-o" "CMakeFiles/Kokkos_BaseSparseSolve.dir/cxx_main.cpp.o" "-c" "/Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/test/BaseSparseSolve/cxx_main.cpp"
Directory	/Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/BUILD/packages/kokkos/test/BaseSparseSolve
Exit Condition	0
Standard Output	
Standard Error	/Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/src/Kokkos_BaseSparseSolve.hpp: In member functio /Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/test/BaseSparseSolve/cxx_main.cpp:262: instanti /Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/src/Kokkos_BaseSparseSolve.hpp:646: warning: sugg /Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/src/Kokkos_BaseSparseSolve.hpp:693: warning: sugg /Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/src/Kokkos_BaseSparseSolve.hpp: In member functio /Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/test/BaseSparseSolve/cxx_main.cpp:287: instanti /Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/src/Kokkos_BaseSparseSolve.hpp:541: warning: sugg /Users/bmpersc/nightly/Trilinos.base/SERIAL_DEBUG/Trilinos/packages/kokkos/src/Kokkos_BaseSparseSolve.hpp:583: warning: sugg

Kitware

CDash 1.5.0 © 2009 Kitware Inc.
[report problems]

Kitware

Coverage Display GCov/Bullseye

./Source/CTest/cmCTestUpdateHandler.cxx	68.21%	45	1
./Source/cmMakefileLibraryTargetGenerator.cxx	68.48%	60	2
./Source/cmTargetLinkLibrariesCommand.cxx	69.17%	17	1
./Source/cmGetPropertyCommand.cxx	69.31%	36	2
./Source/cmExportInstallFileGenerator.cxx	69.32%	16	2
./Source/kwsys/ProcessUNIX.c	69.33%	371	11
./Source/cmVariableWatch.cxx	69.44%	8	1
./Source/cmSystemTools.h	69.64%	1	5
./Source/cmComputeLinkDepends.cxx	69.89%	78	5
./Source/CTest/cmCTestStartCommand.cxx	70.00%	12	0
./Source/cmMakefileExecutableTargetGenerator.cxx	70.83%	16	1
./Source/cmLinkLibrariesCommand.cxx	70.83%	7	0
./Source/cmMakeDepend.cxx	71.01%	44	1
./Source/CTest/cmCTestBuildCommand.cxx	71.74%	26	0
./Source/cmsys/auto_ptr.hxx	71.88%	1	1
./Source/kwsys/testCommandLineArguments.cxx	71.88%	7	1
./Source/CTest/cmCTestSVN.cxx	72.07%	57	2
./Source/cmScriptGenerator.cxx	72.34%	20	1

```
        Version:   4Revision: 1.6 $

        Copyright (c) 2002 Kitware, Inc., Insight Consortium.  All rights reserved.
        See Copyright.txt or http://www.cmake.org/HTML/Copyright.html for details.

          This software is distributed WITHOUT ANY WARRANTY; without even
          the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
          PURPOSE.  See the above copyright notices for more information.

      =========================================================================*/
      #include "cmDefinePropertyCommand.h"
      #include "cmake.h"

      // cmDefinePropertiesCommand
      bool cmDefinePropertyCommand
      ::InitialPass(std::vector<std::string> const& args, cmExecutionStatus &)
00000013 {
00000013   if(args.size() < 1)
           {
00000000   this->SetError("called with incorrect number of arguments");
00000000   return false;
           }

           // Get the scope in which to define the property.
00000012   cmProperty::ScopeType scope;
00000013   if(args[0] == "GLOBAL")
           {
00000010   scope = cmProperty::GLOBAL;
           }
00000000   else if(args[0] == "DIRECTORY")
           {
00000000   scope = cmProperty::DIRECTORY;
           }
00000003   else if(args[0] == "TARGET")
           {
00000003   scope = cmProperty::TARGET;
           }
00000000   else if(args[0] == "SOURCE")
```

```
        Coverage produced by bullseye covbr tool:
          www.bullseye.com/help/ref_covbr.html
        * An arrow --> indicates incomplete coverage.
        * An X indicates a function that was invoked, a switch label that
          was exercised, a try-block that finished, or an exception handler
          that was invoked.
        * A T or F indicates a boolean decision that evaluated true or false,
          respectively.
        * A t or f indicates a boolean condition within a decision if the
          condition evaluated true or false, respectively.
        * A N indicates a constant decision or condition.
        * The slash / means this probe is excluded from summary results.
      ...
            20  #include "cmLocalGenerator.h"
            21  #include "cmGlobalGenerator.h"
            22
X           23  bool cmCTestStartCommand
            24  ::InitialPass(std::vector<std::string> const& args, cmExecutionSta
            25  {
-->F        26    if (args.size() < 1)
            27    {
            28      this->SetError("called with incorrect number of arguments");
            29      return false;
      ...
            37    unsigned int cnt = 0;
            38
            39    const char* const* const c = this->CTest->GetSpecifiedTrack();
-->F        40    if ( c && *c && *(c+1) )
            41    {
-->P        42      if ( strcmp(c[0], "TRACK") == 0 )
            43      {
            44        cnt = atoi(c[1]);
            45        this->CTest->SetSpecifiedTrack(c[1]);
      ...
            47      }
```

Kitware

Valgrind / Purify

Dynamic analysis started on 2009-05-03 03:36:06

Site Name: dash17.kitware
Build Name: Linux-g++4.0

Name	Status	Memory Leak	Uninitialized Memory Read	Potential Memory Leak	Uninitialized Memory Conditional	Mismatched Deallocate	Freeing Invalid Memory	Invalid Pointer Read	Invalid Pointer Write	Labels
QtChart-TestBarSeriesColors	Passed		1	25						
QtChart-TestChartWidget	Passed		1	26						
Mace	Passed			2						
TestHyperOctreeContourFilter	Passed			2	1					
TestUncertaintyTubeFilter	Passed			2						
TestMultiBlock	Passed			2						
TemporalStatistics	Passed			3						
TestGenericCutter	Passed			2						
TestActorLightingFlag	Passed			2						
TestLabelPlacer	Passed			2						
TestOpacity	Passed			2						
TestTextActor3DAlphaBlending	Passed			2						
TestAreaSelections	Passed			2						
TestTranslucentImageActorDepthPeeling	Passed		2	2						
TestGenericVertexAttributesGLSLDepthPeelingPass	Passed			2						

Find: asio Next Previous Highlight all Match case

Dynamic analysis started on 2009-05-04 03:37:17

Site Name: dash17.kitware
Build Name: Linux-g++4.0
TestMultiBlockPassed

```
==3002== Memcheck, a memory error detector.
==3002== Copyright (C) 2002-2007, and GNU GPL'd, by Julian Seward et al.
==3002== Using LibVEX rev 1792, a library for dynamic binary translation.
==3002== Copyright (C) 2004-2007, and GNU GPL'd, by OpenWorks LLP.
==3002== Using valgrind-3.2.3, a dynamic binary instrumentation framework.
==3002== Copyright (C) 2000-2007, and GNU GPL'd, by Julian Seward et al.
==3002== For more details, rerun with: -v
==3002==
==3002==
==3002== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 119 from 2)
==3002== malloc/free: in use at exit: 30,294 bytes in 327 blocks.
==3002== malloc/free: 37,724 allocs, 37,397 frees, 5,207,986 bytes allocated.
==3002== For counts of detected errors, rerun with: -v
==3002== searching for pointers to 327 not-freed blocks.
==3002== checked 2,298,764 bytes.
==3002==
 ==3002== 64 bytes in 1 blocks are still reachable in loss record 15 of 34
==3002==    at 0x401DC87: realloc (vg_replace_malloc.c:306)
==3002==    by 0x62F83E5: (within /usr/lib/libX11.so.6.2.0)
==3002==    by 0x62F908E: (within /usr/lib/libX11.so.6.2.0)
==3002==    by 0x62F95F0: XrmGetStringDatabase (in /usr/lib/libX11.so.6.2.0)
==3002==    by 0x659FB22: (within /usr/lib/libXt.so.6.0.0)
==3002==    by 0x65A0ED4: _XtDisplayInitialize (in /usr/lib/libXt.so.6.0.0)
==3002==    by 0x6596DC7: XtOpenDisplay (in /usr/lib/libXt.so.6.0.0)
==3002==    by 0x437DD13: vtkXRenderWindowInteractor::Initialize() (vtkXRenderWindowInteractor.cxx:317)
==3002==    by 0x42EFDD0: vtkRenderWindow::Render() (vtkRenderWindow.cxx:265)
==3002==    by 0x441E401: vtkXOpenGLRenderWindow::Render() (vtkXOpenGLRenderWindow.cxx:1846)
==3002==    by 0x8081AB6: TestMultiBlock(int, char**) (TestMultiBlock.cxx:142)
==3002==    by 0x805B2E8: main (GraphicsCxxTests.cxx:306)
==3002==
```

Kitware

CDash Email Notification

A submission to CDash for the project CMake has failing tests.
You have been identified as one of the authors who have checked
in changes that are part of this submission or you are listed in the default contact list.

Details on the submission can be found at http://www.cdash.org/CDash/buildSummary.php?buildid=322849

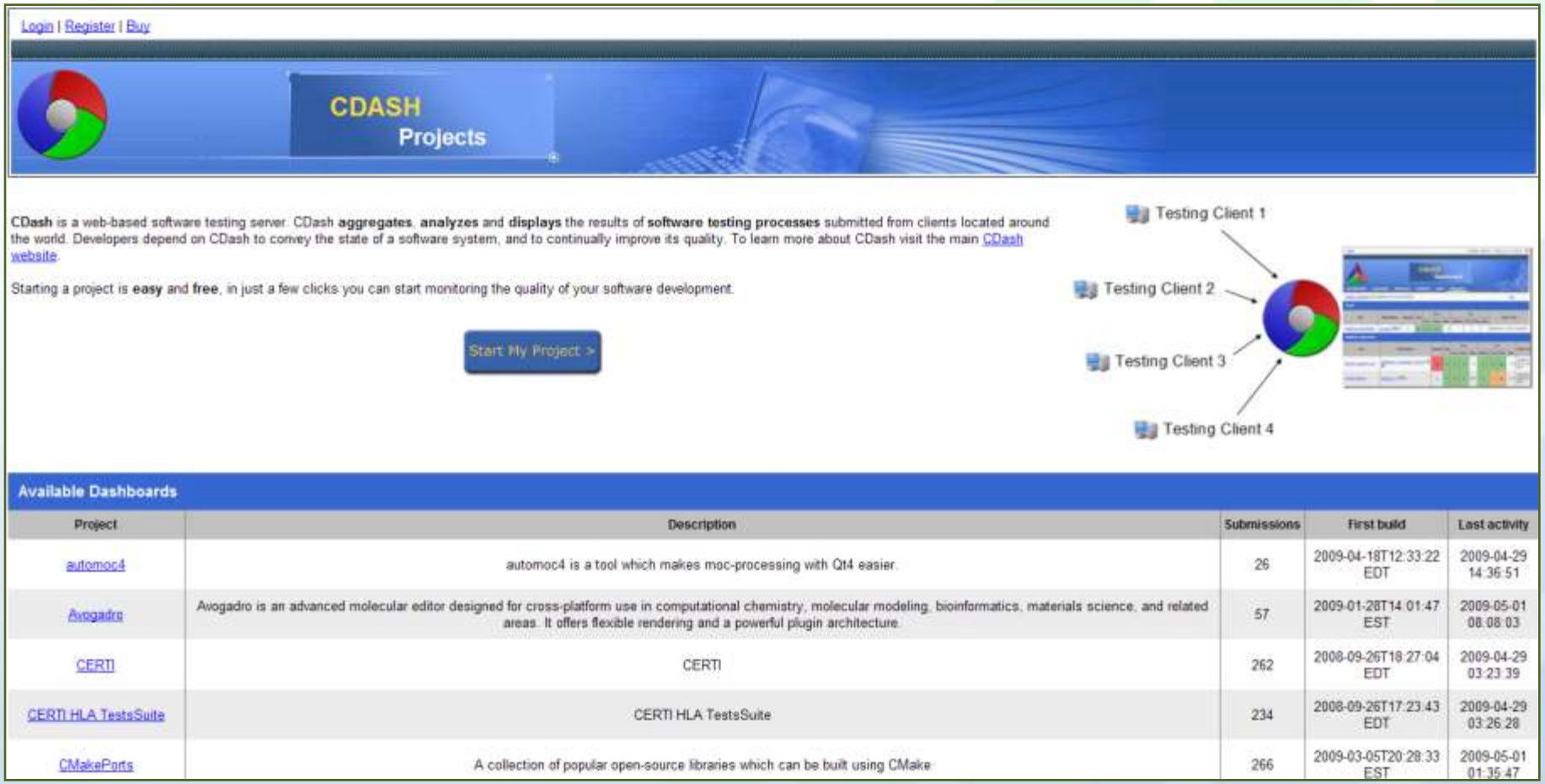
Project: CMake
Site: destiny.kitware
Build Name: HP-UX-aCC
Build Time: 2009-04-29T14:28:00 EDT
Type: Continuous
Tests failing: 85

Tests failing (first 5)
SystemInformationNew (http://www.cdash.org/CDash/testDetails.php?test=21959894&build=322849)
CommandLineTest (http://www.cdash.org/CDash/testDetails.php?test=21959897&build=322849)
FindPackageTest (http://www.cdash.org/CDash/testDetails.php?test=21959898&build=322849)
FindModulesExecuteAll (http://www.cdash.org/CDash/testDetails.php?test=21959899&build=322849)
StringFileTest (http://www.cdash.org/CDash/testDetails.php?test=21959900&build=322849)

-CDash on www.cdash.org

Kitware

Kitware Hosted CDash
http://my.cdash.org

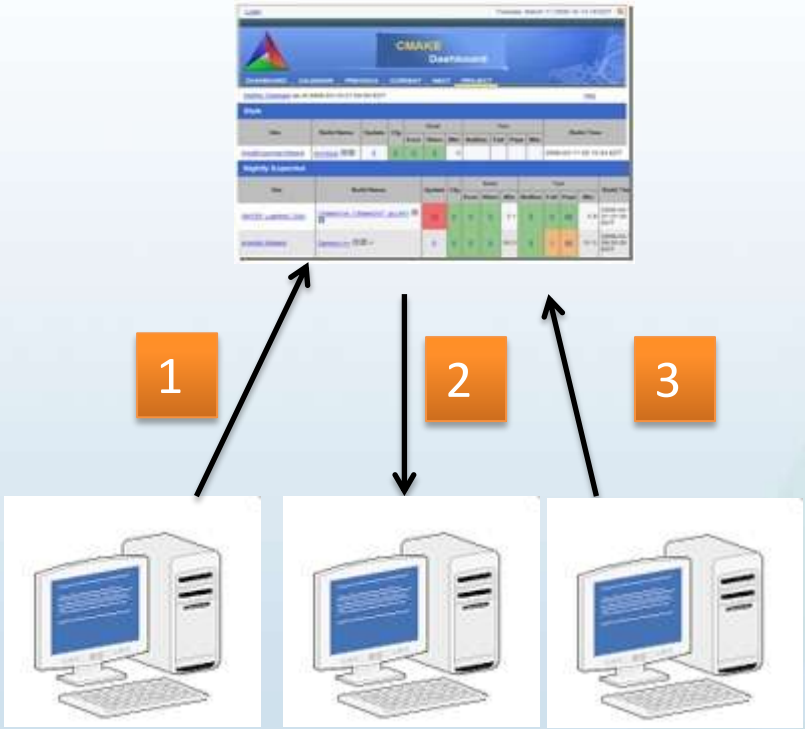


Project	Description	Submissions	First build	Last activity
automoc4	automoc4 is a tool which makes moc-processing with Qt4 easier.	26	2009-04-18T12:33:22 EDT	2009-04-29 14:36:51
Avogadro	Avogadro is an advanced molecular editor designed for cross-platform use in computational chemistry, molecular modeling, bioinformatics, materials science, and related areas. It offers flexible rendering and a powerful plugin architecture.	57	2009-01-28T14:01:47 EST	2009-05-01 08:08:03
CERTI	CERTI	262	2008-09-26T18:27:04 EDT	2009-04-29 03:23:39
CERTI HLA TestsSuite	CERTI HLA TestsSuite	234	2008-09-26T17:23:43 EDT	2009-04-29 03:26:28
CMakePorts	A collection of popular open-source libraries which can be built using CMake	266	2009-03-05T20:28:33 EST	2009-05-01 01:35:47

Kitware

CDash@Home

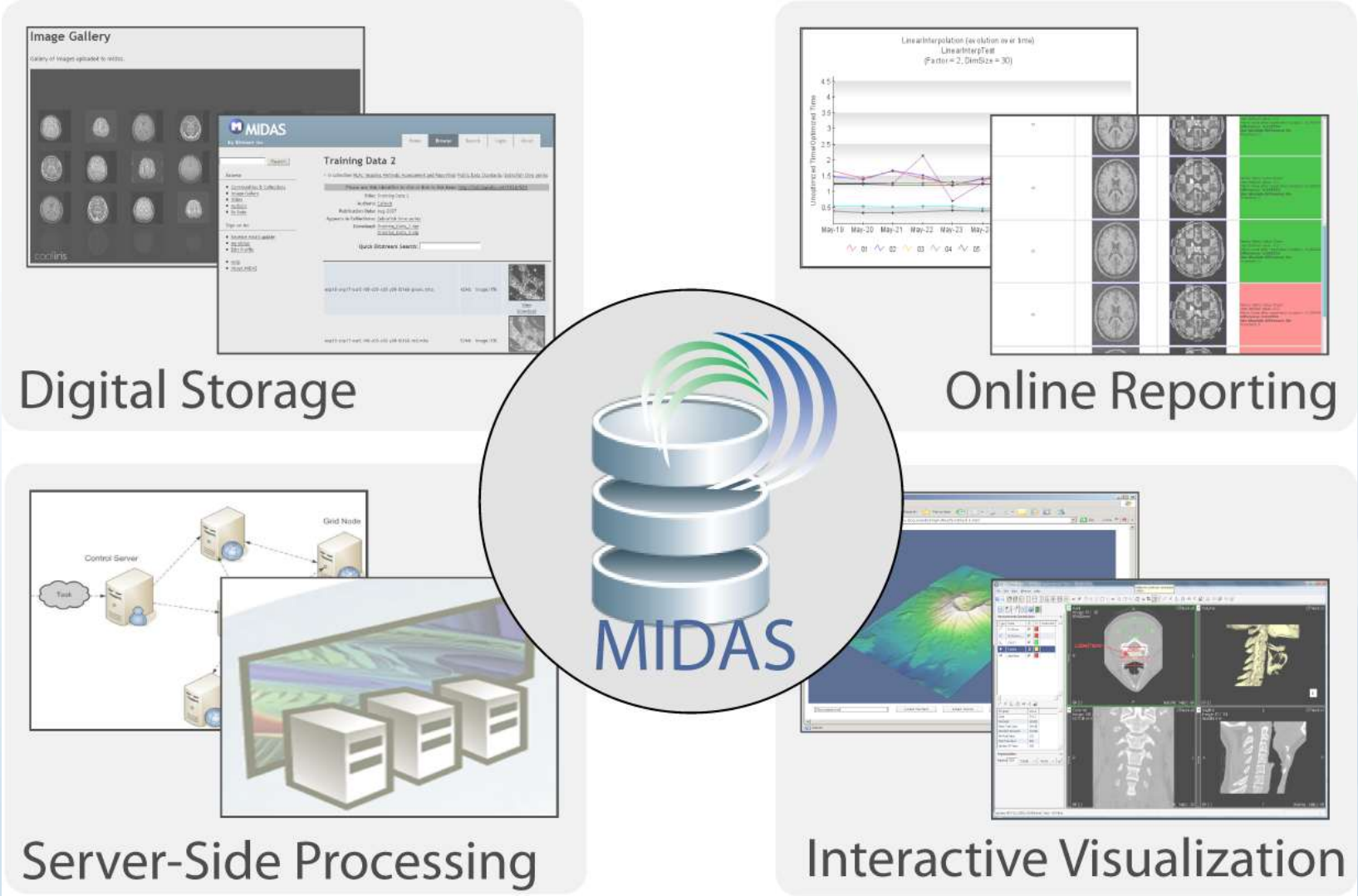
- Make use of idle machine to test code
- Allow anyone to « lend » is computer for testing



1. Site notifies CDash of its configuration and that it's ready
2. If a job is scheduled CDash sends the CTest script
3. When the testing is done CTest sends the result back to CDash.

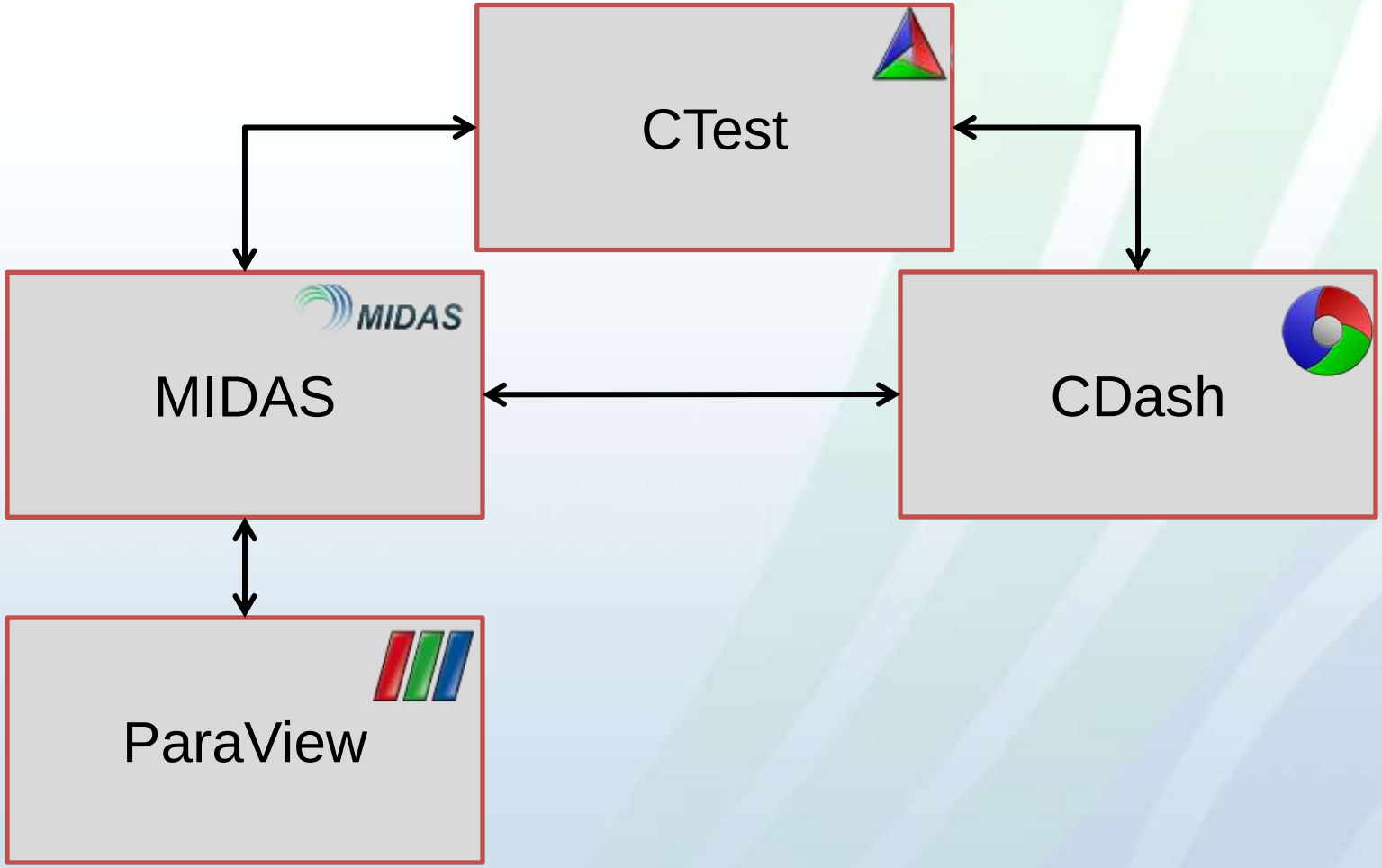
Kitware

Data Management



Kitware

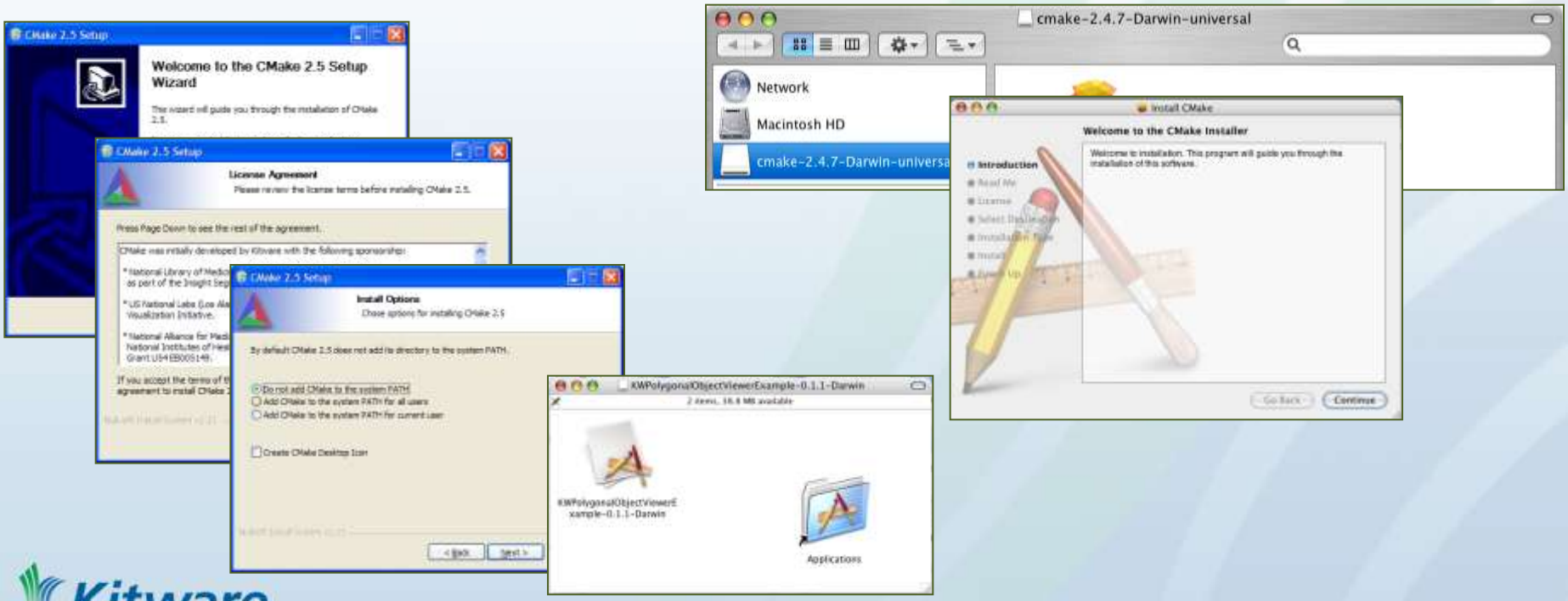
Integrating Technologies



CPack

CPack

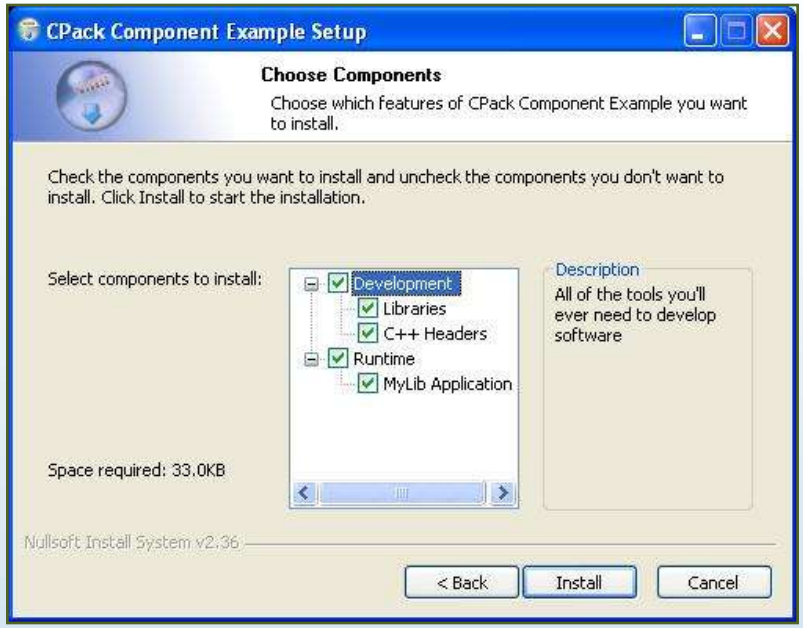
- CPack is bundled with CMake
- Creates professional platform specific installers
  - TGZ and Self extract TGZ (STGZ), NullSoft Scriptable Install System (NSIS), OSX PackageMaker, RPM, Deb

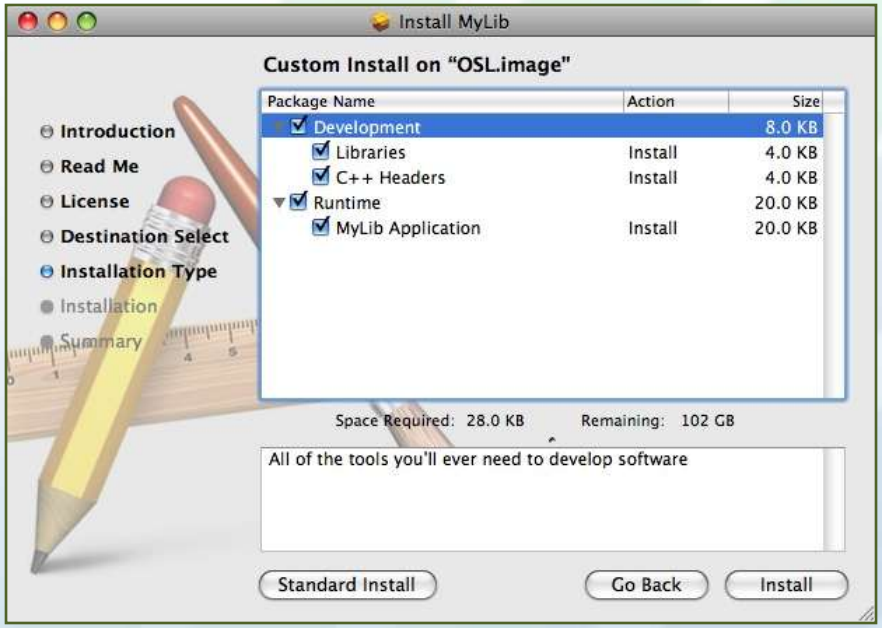


Kitware

CPack Components

http://www.cmake.org/Wiki/CMake:Component_Install_With_CPack





Kitware

Using CPack

- On Windows install command line ZIP program, and NSIS
- setup your project to work with cpack
  - set cpack option variables if needed
  - include(CPack)
  - Reuses existing project install rules
- Running cpack
  - make package (create all packages)
  - make package_source (create source package)
  - cpack -C CPackConfig.cmake -G NSIS
  - cpack -C CPackConfig.cmake -G ZIP
  - cpack -C CPackSourceConfig.cmake -G ZIP

More Information on CPack

- Mastering CMake
- http://www.cmake.org/Wiki/CMake:Packaging_With_CPack

Summary

- Build - CMake
- Test - CTest /CDash
- Deploy - CPack
- Links
  - www.kitware.com
  - www.cmake.org
  - www.cdash.org
  - www.midasplatform.org
  - julien.jomier@kitware.com

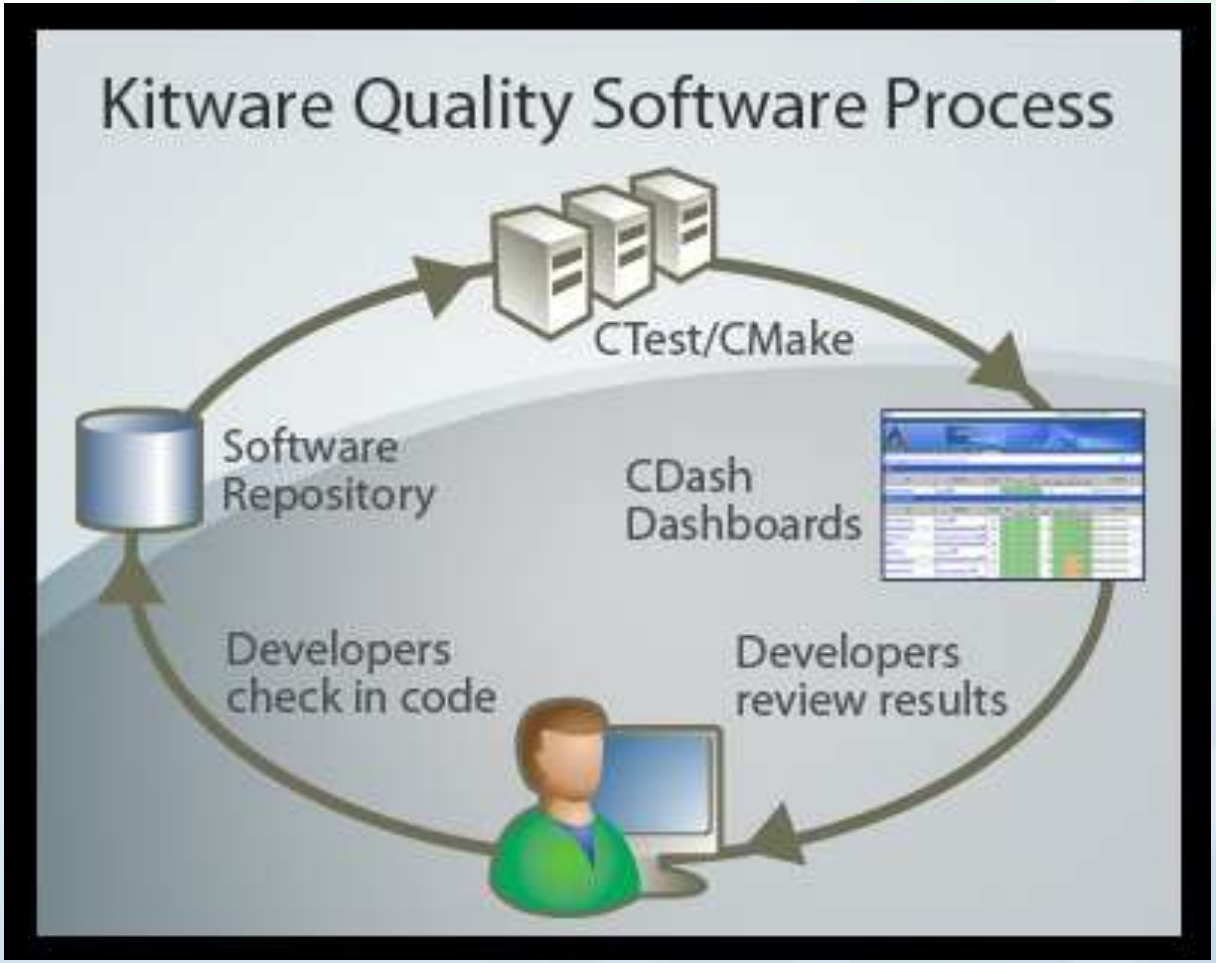
Kitware

Working together

- **Collaboration**
  - Research Grants
- **Support** on CMake (and other tools)
- **Consulting**
  - Software development
  - Infrastructure development
- **Training**
  - Courses off/on-site
  - Books (amazon.co.uk)

Kitware

Thank You



Kitware