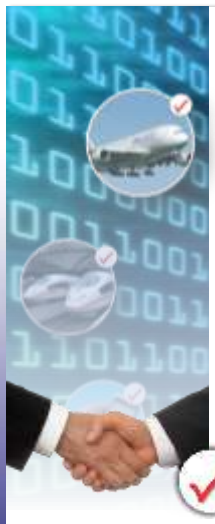
Verifysoft TECHNOLOGY

Testwell CTC++
Test Coverage Analyser for C, C++, Java and C#

Coverage on Host
On-Target Coverage for Embedded Systems

Agenda

Verifysoft TECHNOLOGY



1. Verifysoft Short Introduction
2. History of Testwell CTC++
3. Why Code Coverage?
4. Safety Standards and Code Coverage
5. Different Coverage Levels
6. Compiler Support
7. How does it work? Code Instrumentation
8. Support for Embedded Targets
9. Testwell CTC++ Packages and Qualification Kit
10. Different Reports
11. Supported Platforms/IDE and Tool Integrations
12. Live Demo

1. Verifysoft Short Introduction

Verifysoft TECHNOLOGY

Verifysoft TECHNOLOGY

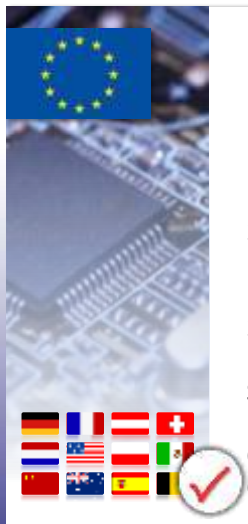
Technologiepark Offenburg
In der Spoeck 10-12
77656 Offenburg
Germany

- ✓ Phone: +49 781 127 8118-0 (Germany)
- ✓ Phone: +33 3 68 33 58 84 (France)
- ✓ Fax: +49 781 63 920-29
- ✓ Email: info@verifysoft.com

www.verifysoft.com

2. History of Testwell CTC++

Verifysoft TECHNOLOGY



1989 Start of CTC++ development by Nokia group

1992 Foundation of Testwell Oy, Tampere (Finland)
with the mission of further development of CTC++

2003 Foundation of Verifysoft Technology GmbH, Offenburg
as distributor for Testwell tools in Europe

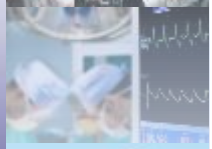
2013 Verifysoft purchased Testwell tools

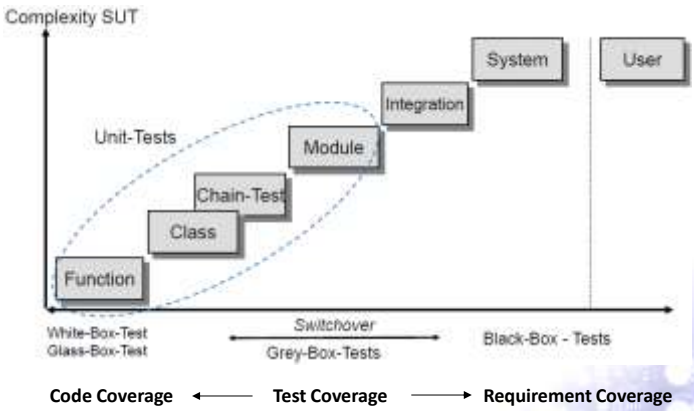
Several hundred CTC++ customers worldwide.
More than 1,000 licenses successfully in use.
Ongoing development.
Qualification-Kit
for DO-178C, IEC 61508, EN 50128, ISO 26262

3. Why Code Coverage?

Verifysoft TECHNOLOGY





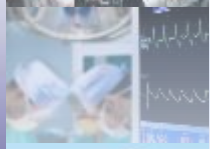



Code Coverage ← **Test Coverage** → **Requirement Coverage**

3. Why Code Coverage?

Verifysoft TECHNOLOGY






Cause-Reason-Graph	Static Testing	Back-to-Back Testing
Classification Tree Method (CTM)	Equivalent Classes Multidimensional Equivalent Classes Boundary Value Analysis Critical Value Analysis Informal Tests Smoke Tests ***Basis***	CRUD
Realtime Testing		Rare Event Testing
Load Tests		Random Testing
Recovery Tests Stress Tests		Monkeytest
		Fuzzing (Fuzz Testing)
Control Flow Oriented Testing	**Advanced**	Evolutionary Testing
		Pairwise Testing

Established test technique for critical Embedded Systems
Test-End criterion (White-Box-Tests)
Necessary to fulfill requirements of safety standards.

Code Coverage:
shows the parts of the code which have been
executed / not executed
tested / not tested

3. Why Code Coverage?

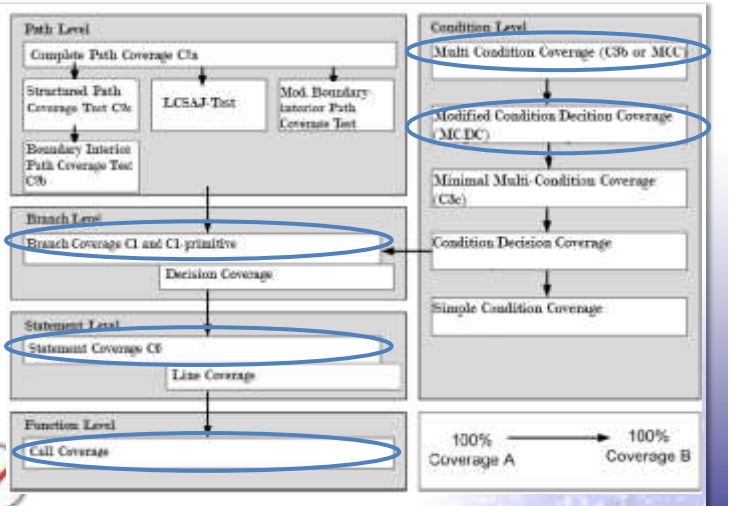
Verifysoft TECHNOLOGY

Why measure Code Coverage ?

- Write better (more adapted) tests
- Avoid redundant tests
- Know when you can stop testing
- Prove to your customers you have good quality
- Insure that your development partner delivers good quality
- Find Dead Code
- Required to obtain certifications
- Mandatory for safety critical development
  (standards DO-178C, IEC 61508, EN 50128, ISO 26262, …)

4. Safety Standards and Code Coverage

Verifysoft TECHNOLOGY



4. Safety Standards and Code Coverage

Verifysoft TECHNOLOGY

DIN EN 61508-3

General Industry

SIL: Safety Integrity Level

Method		SIL 1	SIL 2	SIL 3	SIL 4
...	...	...	...	...	...
7a	Function Coverage	++	++	++	++
7b	Statement Coverage	+	++	++	++
7c	Branch Coverage	+	+	++	++
7d	MC/DC	+	+	+	++

Table B.2 from DIN EN 61508-3

++ Highly recommended
+ Recommended

4. Safety Standards and Code Coverage

Verifysoft TECHNOLOGY

ISO 26262-6

Automotive

ASIL: Automotive Safety Integrity Level

Methods		ASIL A	ASIL B	ASIL C	ASIL D
1a	Statement coverage	++	++	+	+
1b	Branch coverage	+	++	++	++
1c	MC/DC (Modified Condition/Decision Coverage)	+	+	+	++

Table 12 (Software Unit Level), ISO 26262-6

Methods		ASIL A	ASIL B	ASIL C	ASIL D
1a	Function coverage	+	+	++	++
1b	Call coverage	+	+	++	++

Table15 (Software Architectural Level), ISO 26262-6

++ Highly recommended
+ Recommended

4. Safety Standards and Code Coverage

Verifysoft TECHNOLOGY

Aerospace

DO-178B/C

Level	Impact	Coverage Level	% of Systems	% of Software
A	Catastrophic	MC/DC, C1, C0	20-30%	40%
B	Hazardous/Severe	C1, C0	20%	30%
C	Major	C0	25%	20%
D	Minor	-	20%	10%
E	No Effect	-	10%	5%

Statement Coverage C_0, Branch Coverage C_1, Modified Condition/ Decision Coverage MC/DC

Medical Systems

IEC 62304

„... it might be **desirable** to use white box methods to more efficiently accomplish certain tests, initiate stress conditions or faults, or increase code coverage of the qualification tests." (IEC 62304, Chapter B.5.7 Software System testing)

Code Coverage Requirements: CENELEC EN 50128

Verifysoft TECHNOLOGY

Table A.21 – Test Coverage for Code

Technique / measure	Reference	SIL 0	SIL 1	SIL 2	SIL 3	SIL 4
Statement		R	HR	HR	HR	HR
Use the Coverage module to report Statement Coverage for the executed Unit Tests and/or monitored application runs – on the host, simulator and/or target platform.						
Branch		-	R	R	HR	HR
Use the Coverage module to report Decision/Branch Coverage for the executed Unit Tests and/or monitored application runs – on the host, simulator and/or target platform.						
Compound Condition		-	R	R	HR	HR
Use the Coverage module to report Condition Coverage for the executed Unit Tests and/or monitored application runs – on the host, simulator and/or target platform.						
Path		-	R	R	HR	HR
Use the Coverage module to report Path Coverage for the executed Unit Tests and/or monitored application runs – on the host, simulator and/or target platform.						

5. Different Coverage Levels

Verifysoft TECHNOLOGY

Testwell CTC++ supports **all** required **coverage levels**:

- Function Coverage
- Decision Coverage
- Statement Coverage
- Condition Coverage
- Modified Condition/Decision Coverage (MC/DC)

- Multicondition Coverage (MCC)

works together with all unit-test tools

6. Compiler Support

Verifysoft TECHNOLOGY

Testwell CTC++ works with **all compilers**

Support is available for (as of March 2014, for actual list refer to www.verifysoft.com/en_compilers.html):

Altium Tasking classic toolsets, VX-toolset toolsets, c166, cc166, ccm16c, cc51
ARM DS-5, Keil MDK-ARM
Borland/Inprise/Paradigm/Codegear bcc, bcc32, pcc, pcc32 (Paradigm)
Ceva DSP all (just use gcc settings)
Cosmic cx6805, cx6808, cx6812, cxs12x, cxs12z, cxxgate, cx6811, cx6816, cx332, cxst10, cxstm8, cxst7, cxcf, cx56K, cxppc
Freescale/Metrowerks mwccmcf, mwcceppc, mwccmcore, mwcc56800, mwcc56800e, chc12, chc08
Fujitsu/Softune fcc907s, fcc911s, fcc896s
gcc and all gcc based cross-compilers i586-mingw32msvc-gcc, x86_64-linux-gnu-gcc, m68k-palmos-coff-gcc, tricore-gcc, arm-linux-gnueabi-gcc, arm-none-eabi-gcc, arm-none-linux-gnueabi-gcc, arm-elf-gcc, arm-montavista-linux-gnueabi-gcc, pic30-gcc, pic32-gcc, avr-gcc, xc16-gcc, mlx16-gcc, thumb-epoc-pe-gcc, arm4-epoc-pe-gcc, armv-epoc-pe-gcc, powerpc-wrs-linux-gnu-e500v2-glibc_small-gcc, *-gcc, *-*-gcc, *-*-*-gcc
GHS/GreenHills/Multi ccv850, cxv850, ccmips, cxmips, ccarm, cxarm, ccthumb, cxthumb, ccppc, cxppc,gcc (GreenHill, not GNU)
Hitatchi shc, shcpp, ch38, cxrx
HI-Tech PICC (Windows and Linux) picc, picc18, picc32, dspicc, xc16-gcc, xc32-gcc,
HP HPUX CC, HP C++, aCC
IAR compilers and toolchains icc430, icc78k, icc78k0r, icc8051, iccarm, iccavr, iccavr32, icccf, icchcs12, iccdspic, iccmaxq, iccpic18, icccr16c, iccv850, icch8, iccm8k, iccm32c, iccm16c, iccr32c, iccrl78, iccrx, iccsam8, iccstm8
Intel (all platforms) icc, ic86, ic96
Java compilers Javac, jikes, ecj, gcj, kaffe
Keil c51, c166, c251, ca/ cx51, cx2, tcc / armcc
LLVM clang, clang++ / Matlab/Simulink / lcc
Metaware (both Windows and Linux host) hcac, hcarc, hcarm
Microchip MPLAB C pic30-gcc, pic32-gcc

---->

6. Compiler Support

Verifysoft TECHNOLOGY

Testwell CTC++ works with **all compilers** (continued)

→

Microsoft compilers cl on host, both 32 and 64 bit / cl for Smartphones and PocketPC / csc C# compiler / vjc J# compiler
Mono compilers dmcs, mcs, gmcs, smcs
Motorola chc12, chc08
Pathscale pathcc/pathCC
Renesas shc, shcpp, ch38, ccrx
Raisonance rc51, rcmp
Sun WorkShop compilers, javac
Symbian various compilers
TI Code Composer Studio (Windows) cl2000, cl16x, cl470, cl55, cl500, cl430
Texas Instruments Linux compilers cl2000, cl16x, cl470, cl55, cl500, cl430
Trimedia tmcc
VisualDSP++ ccblkfn, cc21k, ccts
Windriver ccarm, ccsimpc, g++simpc, g++arm, cchppa, ccsimso, ccsparc, cc68k, cc386, cc960, ccmips, ccppc

You have not seen your compiler? Contact us!
We will adapt Testwell CTC++ to your compiler within a few days and without any cost (adaptation can even be done by the customer).

Testwell CTC++ supports **all** compilers!
No unsupported compilers!

7. Code Instrumentation

Verifysoft TECHNOLOGY

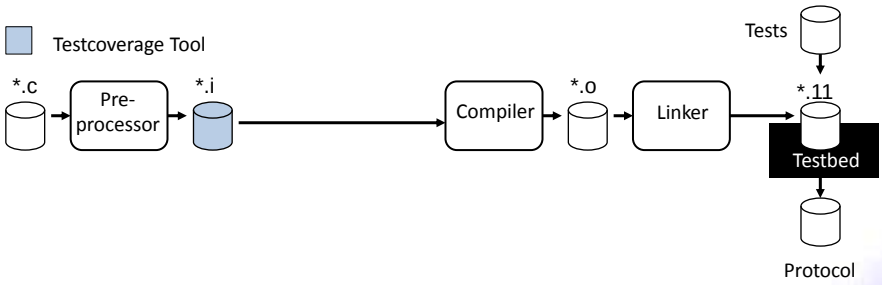
Instrumentation

- Adding of global counters (Integer-Arrays) into the source code
- Storage of information about counter instrumentation
- Increment counters with each run
- Storage of counter values
- Analysis of the counter values for reporting

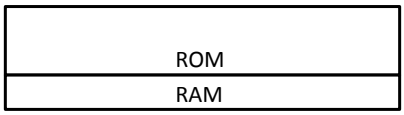
7. Code Instrumentation

Verifysoft TECHNOLOGY

Tool-Chain



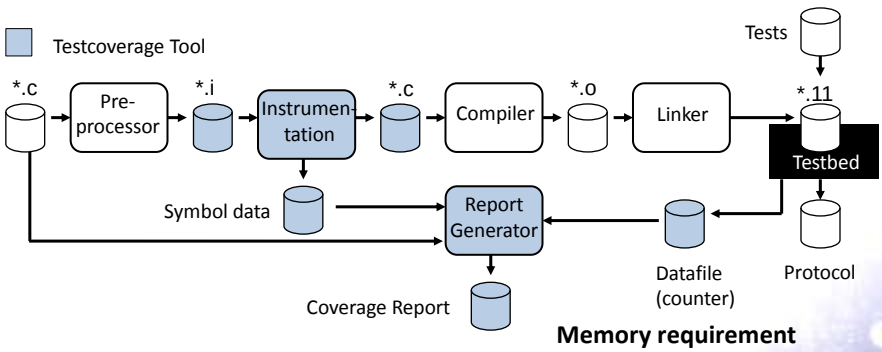
Memory requirement
without instrumentation



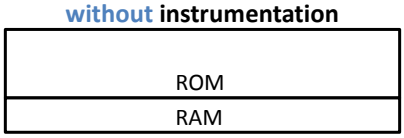
7. Code Instrumentation

Verifysoft TECHNOLOGY

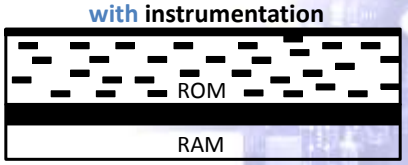
Tool-Chain



Memory requirement
without instrumentation



Memory requirement
with instrumentation



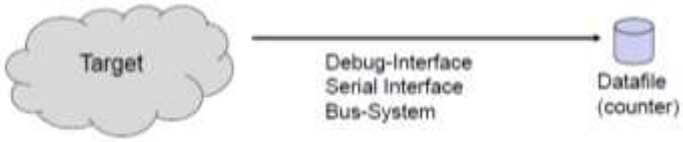
7. Code Instrumentation

Verifysoft TECHNOLOGY

- **RAM**
- **ROM**

Reason for lack of memory: **80 % RAM, 20 % ROM** (pract. experience)

- Mostly no filesystem (so counters have to be stored in memory)



- Limited amount of interfaces on the target device (transfer of datafile)
  Consider additional testing interfaces in the hardware design
  (design for test)

7. Code Instrumentation

Verifysoft TECHNOLOGY

```
int goo( int a, int b, int c)
{
  int x;

  if (((a>0) || (b>0)) && (c>0))
  {
    x = 1;
  }
  else
  {
    x = 0;
  }

  return x;
}
```

ROM-Usage

Without instrumentation:	60 Byte
Function Coverage:	67 Byte
Branch Coverage:	118 Byte
Condition Coverage:	285 Byte

Simple example with small code and big instrumentation overhead (mean 30 % of code size).

Additional RAM-Usage without Bit-Coverage

Function Coverage:	1 Integer
Branch Coverage:	4 Integer
Condition Coverage:	7 Integer

Integer: **32 Bit** (unsigned long) as default

Additional RAM-Usage using Bit-Coverage

Function Coverage:	1 Bit
Branch Coverage:	4 Bit
Condition Coverage:	7 Bit

8. Support for Embedded Targets and native projects

Verifysoft TECHNOLOGY

Testwell CTC++ is **the ideal tool for embedded targets**

- Dramatically easy to use !
- Very low instrumentation overhead on your C files
- Works with all targets
  - Host-Target add-on is provided as source code and so can be easily adapted to new targets
- … even with smallest targets and microcontrollers
- Supports all compilers/cross-compilers

8. Support for Embedded Targets and native projects

Verifysoft TECHNOLOGY

Testwell CTC++ is **the ideal tool for native projects**

- Setup and usage are straightforward
- Java and C# support on top of C & C++
- Very fast analysis
- Interfacing with MS Visual Studio IDE
- …even on large projects

9. Testwell CTC++ Packages and Qualification Kit

Verifysoft TECHNOLOGY

Host-Target add-on for embedded targets

Bit-Coverage add-on for very small embedded targets

Testwell CTC++ Host

CTC++ for Java and Android add-on

CTC++for C# add-on

You only need one code coverage tool for C, C++,Java and C#
One license covers all embedded targets and all compilers

9. Testwell CTC++ Packages and Qualification Kit

Verifysoft TECHNOLOGY

Testwell CTC++ can be used to obtain certification in automotive, railway, avionics and medical industries

Tool-Qualification Kits available

10. Different Reports

Verifysoft TECHNOLOGY

Reports in text, XML,HTML

Directory Summary
Files Summary
Functions Summary

Execution Profile
Untested Code Listing
Execution Time Listing

10. Different Reports

Verifysoft TECHNOLOGY

CTC++ Coverage Report - Directory Summary

Directory Summary | Files Summary | Functions Summary | Untested Code | Execution Profile

Symbol file(s)	: HOR.sym (Mon Feb 17 12:10:50 2014)
	: F:\ctcwork\demos\cube\HOR.sym (Fri Mar 14 09:46:50 2014)
Data file(s)	: HOR.dat (Mon Feb 17 12:13:18 2014)
	: F:\ctcwork\demos\cube\HOR.dat (Fri Mar 14 09:47:13 2014)
Listing produced at	: Wed Mar 26 14:34:47 2014
Coverage view	: Reduced to MC/DC coverage
Input listing	: STDIN
HTML generated at	: Wed Mar 26 16:34:47 2014
ctc2html v3.5 options	: -o webCTCHTML -t 75 -nsb
Threshold percent	: 75 %

(Click on header to sort)

TER % - MC/DC	TER % statement	Directory
75 % (21/28)	88 % (21/24)	.
66 % - (130/197)	77 % (215/280)	F:\ctcwork\demos\cube
67 % - (151/225)	78 % (236/304)	OVERALL

Directories	: 2
Source files	: 7
Functions	: 64
Source lines	: 903
Measurement points	: 221
TER structural	: 67 % (151/225) MC/DC
TER statement	: 78 % (236/304)

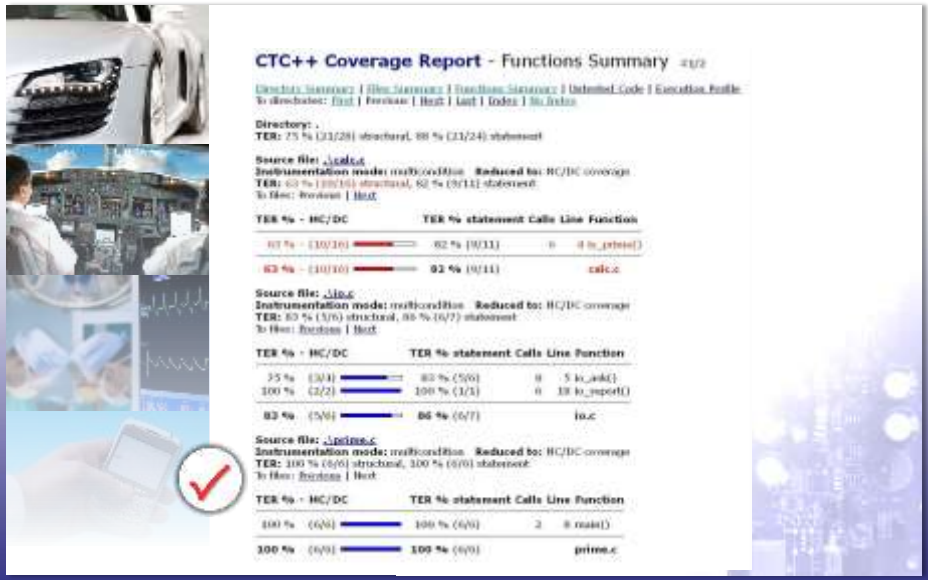
Directory Summary | Files Summary | Functions Summary | Untested Code | Execution Profile

10. Different Reports

Verifysoft TECHNOLOGY

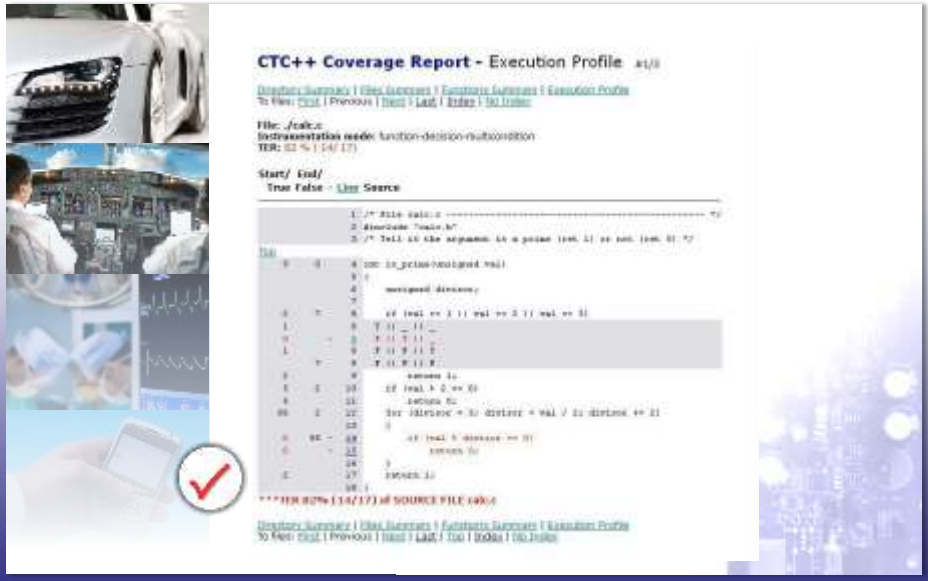
10. Different Reports

Verifysoft TECHNOLOGY



10. Different Reports

Verifysoft TECHNOLOGY

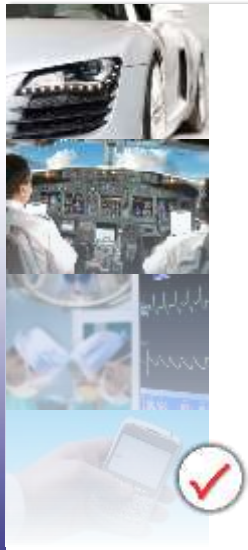


10. Different Reports

Verifysoft TECHNOLOGY

10. Different Reports

Verifysoft TECHNOLOGY



11. Supported Platforms/IDE and Tool Integrations

Verifysoft TECHNOLOGY

Supported Platforms

Windows, Linux, FreeBSD, Solaris, HP-UX

MacOSX, AIX, others on request

11. Supported Platforms/IDE and Tool Integrations

Verifysoft TECHNOLOGY

IDE-Integrations

Visual Studio v7.0 and later
IAR (all platforms)
Borland 5.02
Fujitsu Softune
Eclipse
Others on request

Works also with:
MP-LAB, Keil, …

11. Supported Platforms/IDE and Tool Integrations

Verifysoft TECHNOLOGY

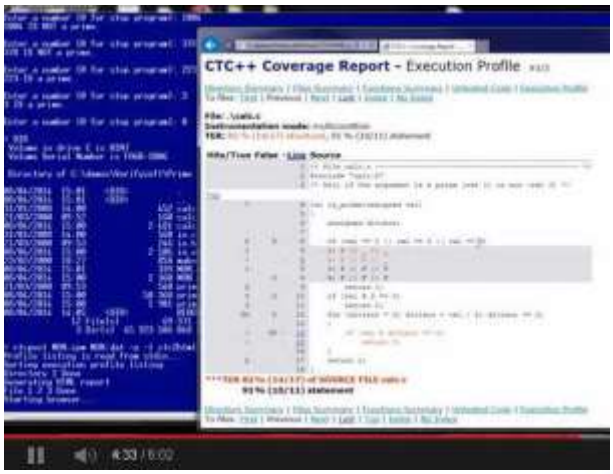
Integrations
with Tool-Chains and Testing environments
include

CATIA - AUTOSAR Builder (Dassault Systèmes)
dSpace SystemDesk
dSpace TargetLink
PikeTech TPT
QTronic TestWeaver / Silver
Imagix 4D
SonarQube
… (ask for other integrations)

Further information: www.verifysoft.com

12. Live Demo

Verifysoft TECHNOLOGY



For an online presentation, please visit
http://www.verifysoft.com/en_ctcpp_online_presentations.html

Testwell CTC++

Verifysoft TECHNOLOGY

Testwell CTC++ Test Coverage Analyser for C and C++
CTC++ add-on for Java and Android
CTC++ add-on for C#

All coverage levels
- Statement coverage
- Function coverage
- Decision/branch coverage
- Condition coverage
- Modified condition/decision cov. MC/DC coverage
- Multicondition coverage (MCC)

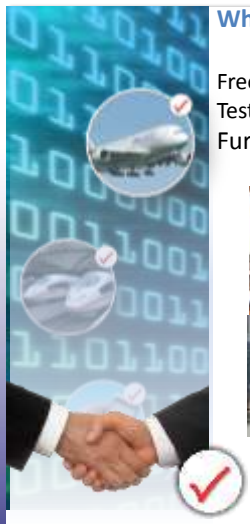
All compilers
All embedded targets!
Works with all unit test tools

Customers

Verifysoft TECHNOLOGY

Thank You

Verifysoft TECHNOLOGY



What can we do for you?

Free tool evaluation incl. support
Testwell CTC++ Training
Further information: www.verifysoft.com

Thank you for your time!

Your Verifysoft Team