Compiler Verification, More Necessary than Ever

Marcel Beemster, CTO

Solid Sands

Selection of our Customers

WIND™
AN INTEL COMPANY

DENSO

SILEXICA

HIGHTEC

ARM®

NVIDIA

SYNOPSYS®
Silicon to Software™

QUALCOMM®

Green Hills®
SOFTWARE

TEXAS INSTRUMENTS

cadence®

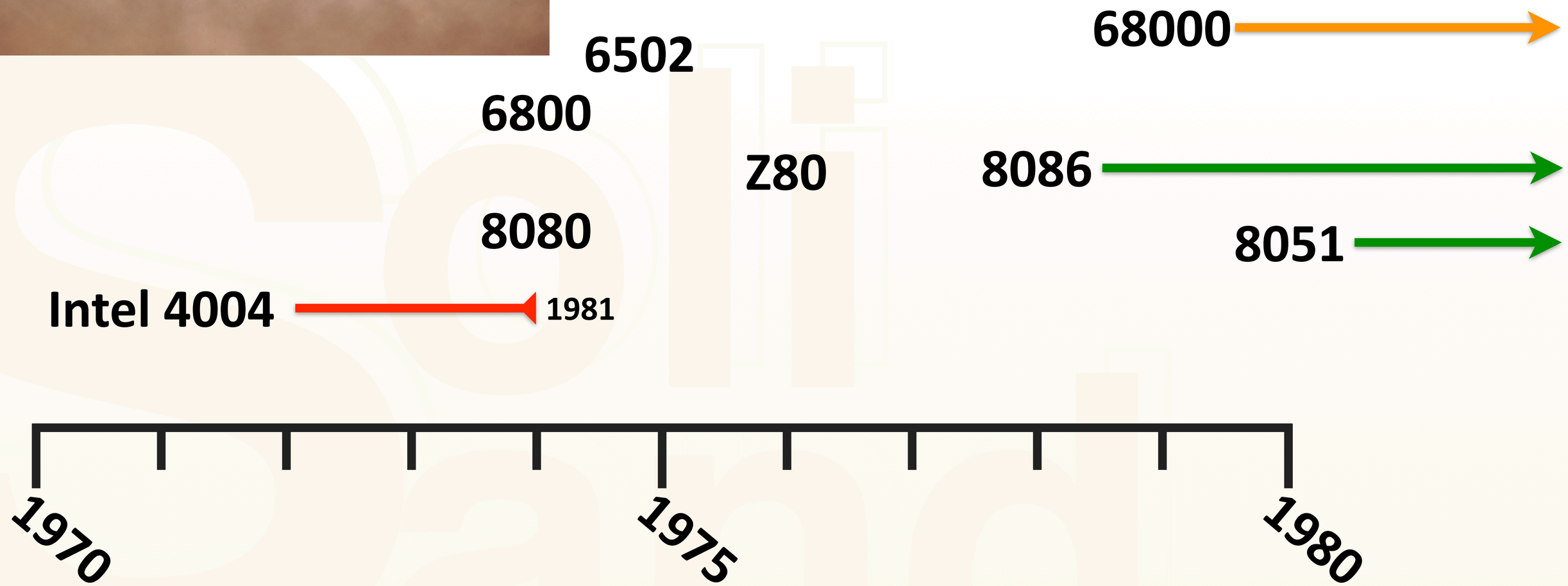
ERICSSON

Sony
Interactive
Entertainment

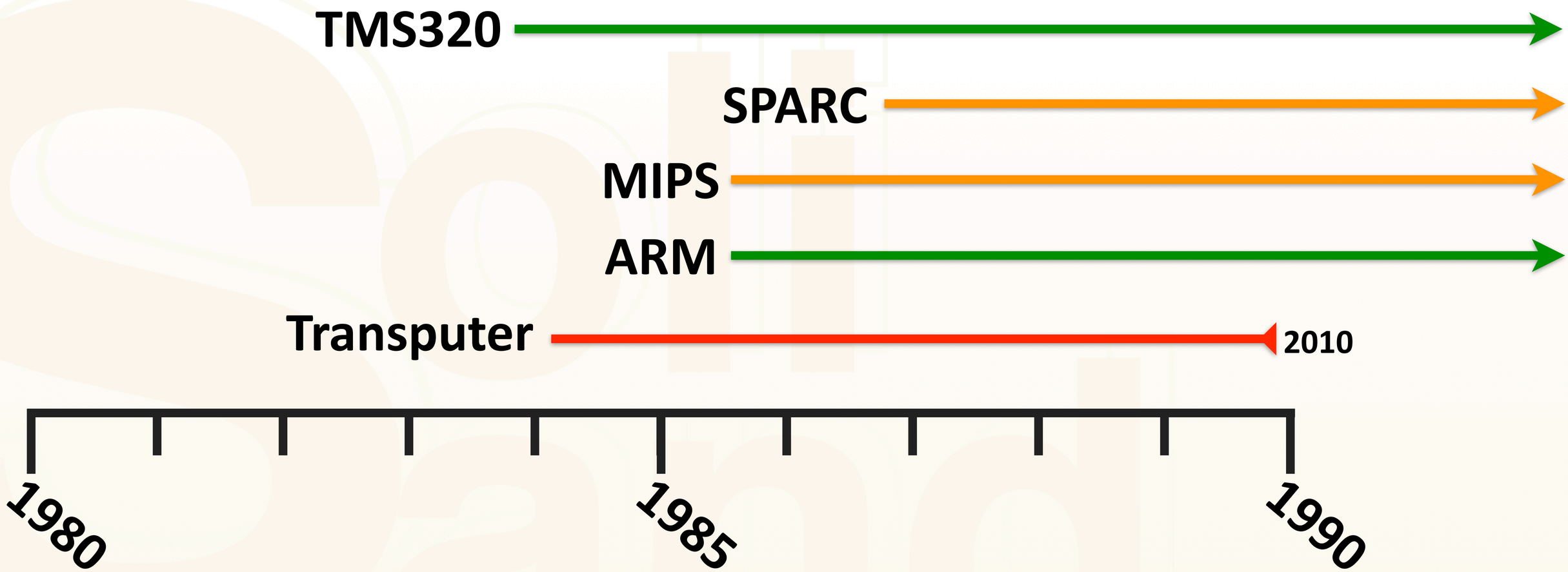
ANALOG DEVICES

CEVA®

The Seventies - CISC



The Eighties - RISC



The Nineties - DSP

StarCore

MMDSP+

REAL

SHARC

1990

1995

2000

The 2000s - VLIW/SIMD

Solid Sands

Hexagon

Itanium

2017

Trimedia

2010

GPU

2000

2005

iPhone

Crisis

2010

The 2010s

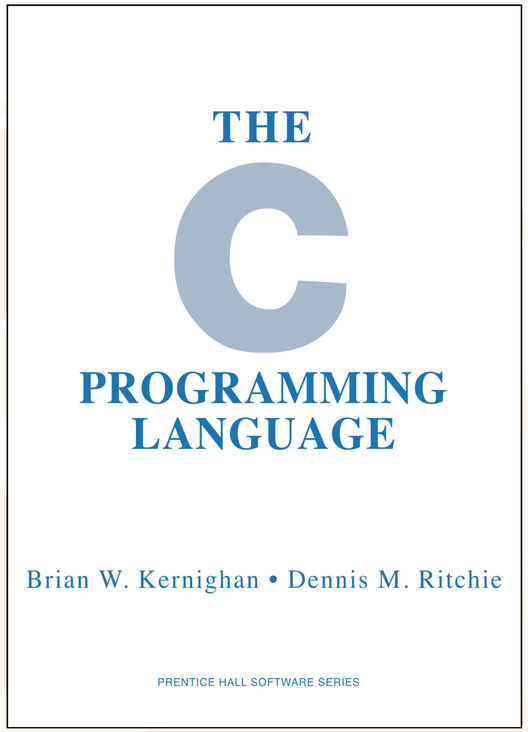
AI/ADAS
ASIP

RISC-V

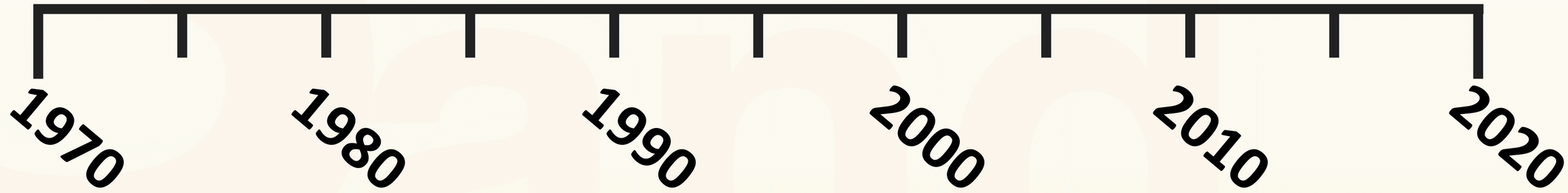
AArch64

2010 2015 2020

Constants of the IT Industry
(The Porsche 911s of Computing?)



- Moore's law
- Battery power is limited
- The C programming language



So, When is my Compiler Finished?

GCC 7.5	November 14, 2019
GCC 9.2	August 12, 2019
GCC 9.1	May 3, 2019
GCC 8.3	February 22, 2019
GCC 7.4	December 6, 2018
GCC 6.5	October 26, 2018
GCC 8.2	July 26, 2018
GCC 8.1	May 2, 2018
GCC 7.3	January 25, 2018
GCC 5.5	October 10, 2017
GCC 7.2	August 14, 2017
GCC 6.4	July 4, 2017
GCC 7.1	May 2, 2017
GCC 6.3	December 21, 2016
GCC 6.2	August 22, 2016
GCC 4.9.4	August 3, 2016
GCC 5.4	June 3, 2016
GCC 6.1	April 27, 2016
GCC 5.3	December 4, 2015
GCC 5.2	July 16, 2015
GCC 4.9.3	June 26, 2015
GCC 4.8.5	June 23, 2015
GCC 5.1	April 22, 2015
GCC 4.8.4	December 19, 2014
GCC 4.9.2	October 30, 2014
GCC 4.9.1	July 16, 2014
GCC 4.7.4	June 12, 2014
GCC 4.8.3	May 22, 2014
GCC 4.9.0	April 22, 2014
GCC 4.8.2	October 16, 2013
GCC 4.8.1	May 31, 2013
GCC 4.6.4	April 12, 2013
GCC 4.7.3	April 11, 2013
GCC 4.8.0	March 22, 2013
GCC 4.7.2	September 20, 2012
GCC 4.5.4	July 2, 2012
GCC 4.7.1	June 14, 2012
GCC 4.7.0	March 22, 2012
GCC 4.4.7	March 13, 2012
GCC 4.6.3	March 1, 2012
GCC 4.6.2	October 26, 2011
GCC 4.6.1	June 27, 2011
GCC 4.3.6	June 27, 2011
GCC 4.5.3	April 28, 2011
GCC 4.4.6	April 16, 2011
GCC 4.6.0	March 25, 2011
GCC 4.5.2	December 16, 2010
GCC 4.4.5	October 1, 2010
GCC 4.5.1	July 31, 2010
GCC 4.3.5	May 22, 2010
GCC 4.4.4	April 29, 2010
GCC 4.5.0	April 14, 2010
GCC 4.4.3	January 21, 2010
GCC 4.4.2	October 15, 2009
GCC 4.3.4	August 4, 2009
GCC 4.4.1	July 22, 2009
GCC 4.4.0	April 21, 2009
GCC 4.3.3	January 24, 2009
GCC 4.3.2	August 27, 2008
GCC 4.3.1	June 6, 2008
GCC 4.2.4	May 19, 2008
GCC 4.3.0	March 5, 2008
GCC 4.2.3	February 1, 2008
GCC 4.2.2	October 7, 2007
GCC 4.2.1	July 18, 2007
GCC 4.2.0	May 13, 2007
GCC 4.1.2	February 13, 2007
GCC 4.0.4	January 31, 2007
GCC 4.1.1	May 24, 2006
GCC 4.0.3	March 10, 2006
GCC 3.4.6	March 06, 2006
GCC 4.1.0	February 28, 2006
GCC 3.4.5	November 30, 2005
GCC 4.0.2	September 28, 2005
GCC 4.0.1	July 7, 2005
GCC 3.4.4	May 18, 2005
GCC 3.3.6	May 3, 2005
GCC 4.0.0	April 20, 2005
GCC 3.4.3	November 4, 2004
GCC 3.3.5	September 30, 2004
GCC 3.4.2	September 6, 2004
GCC 3.4.1	July 1, 2004
GCC 3.3.4	May 31, 2004
GCC 3.4.0	April 18, 2004
GCC 3.3.3	February 14, 2004
GCC 3.3.2	October 17, 2003
GCC 3.3.1	August 8, 2003
GCC 3.3	May 13, 2003
GCC 3.2.3	April 22, 2003
GCC 3.2.2	February 05, 2003
GCC 3.2.1	November 19, 2002
GCC 3.2	August 14, 2002
GCC 3.1.1	July 25, 2002
GCC 3.1	May 15, 2002
GCC 3.0.4	February 20, 2002
GCC 3.0.3	December 20, 2001
GCC 3.0.2	October 25, 2001
GCC 3.0.1	August 20, 2001
GCC 3.0	June 18, 2001
GCC 2.95.3	March 16, 2001
GCC 2.95.2	October 24, 1999
GCC 2.95.1	August 19, 1999
GCC 2.95	July 31, 1999
EGCS 1.1.2	March 15, 1999
EGCS 1.1.1	December 1, 1998
EGCS 1.1	September 3, 1998
EGCS 1.0.3	May 15, 1998
EGCS 1.0.2	March 16, 1998
gcc 2.8.1	March 2, 1998
gcc 2.8.0	January 7, 1998
EGCS 1.0.1	January 6, 1998
EGCS 1.0	December 3, 1997
2.7.2.3	August 22, 1997
2.7.2.2	January 29, 1997
2.7.2.1	June 29, 1996
2.7.2	November 26, 1995
2.7.1	November 12, 1995
2.7.0	June 16, 1995
2.6.3	November 30, 1994
2.6.2	November 12, 1994
2.6.1	November 1, 1994
2.6.0	July 14, 1994
2.5.8	January 24, 1994
2.5.7	December 12, 1993
2.5.6	December 3, 1993
2.5.5	November 27, 1993
2.5.4	November 16, 1993
2.5.3	November 11, 1993
2.5.2	November 1, 1993
2.5.1	October 31, 1993
2.5.0	October 22, 1993
2.4.5	June 20, 1993
2.4.4	June 19, 1993
2.4.3	June 1, 1993
2.4.2	May 31, 1993
2.4.1	May 26, 1993
2.4.0	May 17, 1993
2.3.3	December 26, 1992
2.3.2	November 27, 1992
2.3.1	November 1, 1992
2.3	October 31, 1992
2.2.2	June 14, 1992
2.2.1	June 9, 1992
2.2	June 8, 1992
2.1	March 24, 1992
2.0	February 22, 1992
1.42.0	September 20, 1992
1.42	September 20, 1992
1.41	August 27, 1992
1.41.0	July 13, 1992
1.40.3	October 19, 1991
1.40	June 1, 1991
1.39.1	May 4, 1991
1.39	January 16, 1991
1.38	December 21, 1990
1.37.1	March 1, 1990
1.37.0	February 28, 1990
1.37.1	February 21, 1990
1.37	February 11, 1990
1.36.4	January 30, 1990
1.36.3	January 16, 1990
1.36	September 24, 1989
1.35	April 26, 1989
1.34	February 23, 1989
1.33	February 1, 1989
1.32	December 21, 1988
1.31	November 19, 1988
1.30	October 13, 1988
1.29	October 6, 1988
1.28	September 14, 1988
1.27	September 5, 1988
1.26	August 18, 1988
1.25	August 3, 1988
1.24	July 2, 1988
1.23	June 26, 1988
1.22	May 22, 1988
1.21	May 1, 1988
1.20	April 19, 1988
1.19	March 29, 1988
1.18	February 4, 1988
1.17	January 9, 1988
1.16	December 19, 1987
1.15.3	December 18, 1987
1.15	November 28, 1987
1.14	November 6, 1987
1.13	October 12, 1987
1.12	October 3, 1987
1.11	September 5, 1987
1.10	August 22, 1987
1.9	August 18, 1987
1.8	August 10, 1987
1.7	July 21, 1987
1.6	July 2, 1987
1.5	June 18, 1987
1.4	June 13, 1987
1.3	June 10, 1987
1.2	June 1, 1987
1.1	May 24, 1987
1.0	May 23, 1987
0.9	March 22, 1987

So, When is my Compiler Finished?

GCC 7.5	November 14, 2019
GCC 9.2	August 12, 2019
GCC 9.1	May 3, 2019
GCC 8.3	February 22, 2019
GCC 7.4	December 6, 2018
GCC 6.5	October 26, 2018
GCC 8.2	July 26, 2018
GCC 8.1	May 2, 2018
GCC 7.3	January 25, 2018
GCC 5.5	October 10, 2017
GCC 7.2	August 14, 2017
GCC 6.4	July 4, 2017
GCC 7.1	May 2, 2017
GCC 6.3	December 21, 2016
GCC 6.2	August 22, 2016
GCC 4.9.4	August 3, 2016
GCC 5.4	June 3, 2016
GCC 6.1	April 27, 2016
GCC 5.3	December 4, 2015
GCC 5.2	July 16, 2015
GCC 4.9.3	June 26, 2015
GCC 4.8.5	June 23, 2015
GCC 5.1	April 22, 2015
GCC 4.8.4	December 19, 2014
GCC 4.9.2	October 30, 2014
GCC 4.9.1	July 16, 2014
GCC 4.7.4	June 12, 2014
GCC 4.8.3	May 22, 2014
GCC 4.9.0	April 22, 2014
GCC 4.8.2	October 16, 2013
GCC 4.8.1	May 31, 2013
GCC 4.6.4	April 12, 2013
GCC 4.7.3	April 11, 2013
GCC 4.8.0	March 22, 2013
GCC 4.7.2	September 20, 2012
GCC 4.5.4	July 2, 2012
GCC 4.7.1	June 14, 2012
GCC 4.7.0	March 22, 2012
GCC 4.4.7	March 13, 2012
GCC 4.6.3	March 1, 2012
GCC 4.6.2	October 26, 2011
GCC 4.6.1	June 27, 2011
GCC 4.3.6	June 27, 2011
GCC 4.5.3	April 28, 2011
GCC 4.4.6	April 16, 2011
GCC 4.6.0	March 25, 2011
GCC 4.5.2	December 16, 2010
GCC 4.4.5	October 1, 2010
GCC 4.5.1	July 31, 2010
GCC 4.3.5	May 22, 2010
GCC 4.4.4	April 29, 2010
GCC 4.5.0	April 14, 2010
GCC 4.4.3	January 21, 2010
GCC 4.4.2	October 15, 2009
GCC 4.3.4	August 4, 2009
GCC 4.4.1	July 22, 2009
GCC 4.4.0	April 21, 2009
GCC 4.3.3	Janu
GCC 4.3.2	A , 2008
GCC 4.3.1	Ju 2008
GCC 4.4	Ma 2008
GCC 4.3.0	Mar 2008
GCC	Febru
GCC 4	Octob
GCC 4.	July 18
GCC 4.2	May 13,
GCC 4.1.	February 2007
4.0.4	January 3 07
GC 1	May 24, 20
GCC 4	arch 10, 2
GCC 3.4.	rch 06, 2006
GCC 4.1.0	February 28, 2006
GCC 3.4.5	November 30, 2005
GCC 4.0.2	September 28, 2005
GCC 4.0.1	July 7, 2005
GCC 3.4.4	May 18, 2005
GCC 3.3.6	May 3, 2005
GCC 4.0.0	April 20, 2005
GCC 3.4.3	November 4, 2004
GCC 3.3.5	September 30, 2004
GCC 3.4.2	September 6, 2004
GCC 3.4.1	July 1, 2004
GCC 3.3.4	May 31, 2004
GCC 3.4.0	April 18, 2004
GCC 3.3.3	February 14, 2004
GCC 3.3.2	October 17, 2003
GCC 3.3.1	August 8, 2003
GCC 3.3	May 13, 2003
GCC 3.2.3	April 22, 2003
GCC 3.2.2	February 05, 20
GCC 3.2.1	Novem
GCC 3.2	Aug , 2002
GCC 3.	July 2002
GCC 3.	May 002
GCC 3.	Febru 20, 2002
GCC 3.	Decem
GCC 3.	October 01
GCC 3.	August 2 01
GCC 3.	June 18, 2
GCC 2.	March 16, 1
GCC 2.	October 24 9
C 2.	August 19,
2.	July 31, 1999
E	March 15, 1999
EGC	December 1, 1998
EGCS 2.1	September 3, 1998
EGCS 1.0.3	May 15, 1998
EGCS 1.0.2	March 16, 1998
gcc 2.8.1	March 2, 1998
gcc 2.8.0	January 7, 1998
EGCS 1.0.1	January 6, 1998
EGCS 1.0	December 3, 1997
2.7.2.3	August 22, 1997
2.7.2.2	January 29, 1997
2.7.2.1	June 29, 1996
2.7.2	November 26, 1995
2.7.1	November 12, 1995
2.7.0	June 16, 1995
2.6.3	November 30, 1994
2.6.2	November 12, 1994
2.6.1	November 1, 1994
2.6.0	July 14, 1994
2.5.8	January 24, 1994
2.5.7	December 12 3
2.5.6	December 3, 1
2.5.5	November 27,
	November 16, 1
	ember 11, 19
2.5.2	mber 1, 1993
2.5.1	ber 31, 1993
2.5.0	ber 22, 1993
2.4.5	e 20, 1993
	e 19, 1993
	1993
2.4.2	May 993
2.4.1	May 2 3
2.4.0	May 17,
2.3	December 26, 1992
2.2	November 27, 1992
2.1	November 1, 1992
2.3	October 31, 1992
2.2.2	June 14, 1992
2.2.1	June 9, 1992
2.2	June 8, 1992
2.1	March 24, 1992
2.0	February 22, 1992
1.42.0	September 20, 1992
1.42	September 20, 1992
1.41	August 27, 1992
1.41.0	July 13, 1992
1.40.3	October 19, 1991
1.40	June 1, 1991
1.39.1	May 4, 1991
1.39	January 16, 1991
1.38	December 21, 1990
1.37.1	March 1, 1990
1.37.0	February 28, 1990
1.37.1	February 21, 1990
1.37	February 11, 1990
1.36.4	January 30, 1990
1.36.3	January 16, 1990
1.36	September 24, 1989
1.35	April 26, 1989
1.34	February 23, 1989
1.33	February 1, 1989
1.32	December 21, 1988
1.31	November 19, 1988
1.30	October 13, 1988
1.29	October 6, 1988
1.28	September 14, 1988
1.27	September 5, 1988
1.26	August 18, 1988
1.25	August 3, 1988
1.24	July 2, 1988
1.23	June 26, 1988
1.22	May 22, 1988
1.21	May 1, 1988
1.20	April 19, 1988
1.19	March 29, 1988
1.18	February 4, 1988
1.17	January 9, 1988
1.16	December 19, 1987
1.15.3	December 18, 1987
1.15	November 28, 1987
1.14	November 6, 1987
1.13	October 12, 1987
1.12	October 3, 1987
1.11	September 5, 1987
1.10	August 22, 1987
1.9	August 18, 1987
1.8	August 10, 1987
1.7	July 21, 1987
1.6	July 2, 1987
1.5	June 18, 1987
1.4	June 13, 1987
1.3	June 10, 1987
1.2	June 1, 1987
1.1	May 24, 1987
1.0	May 23, 1987
0.9	March 22, 1987

NEVER!

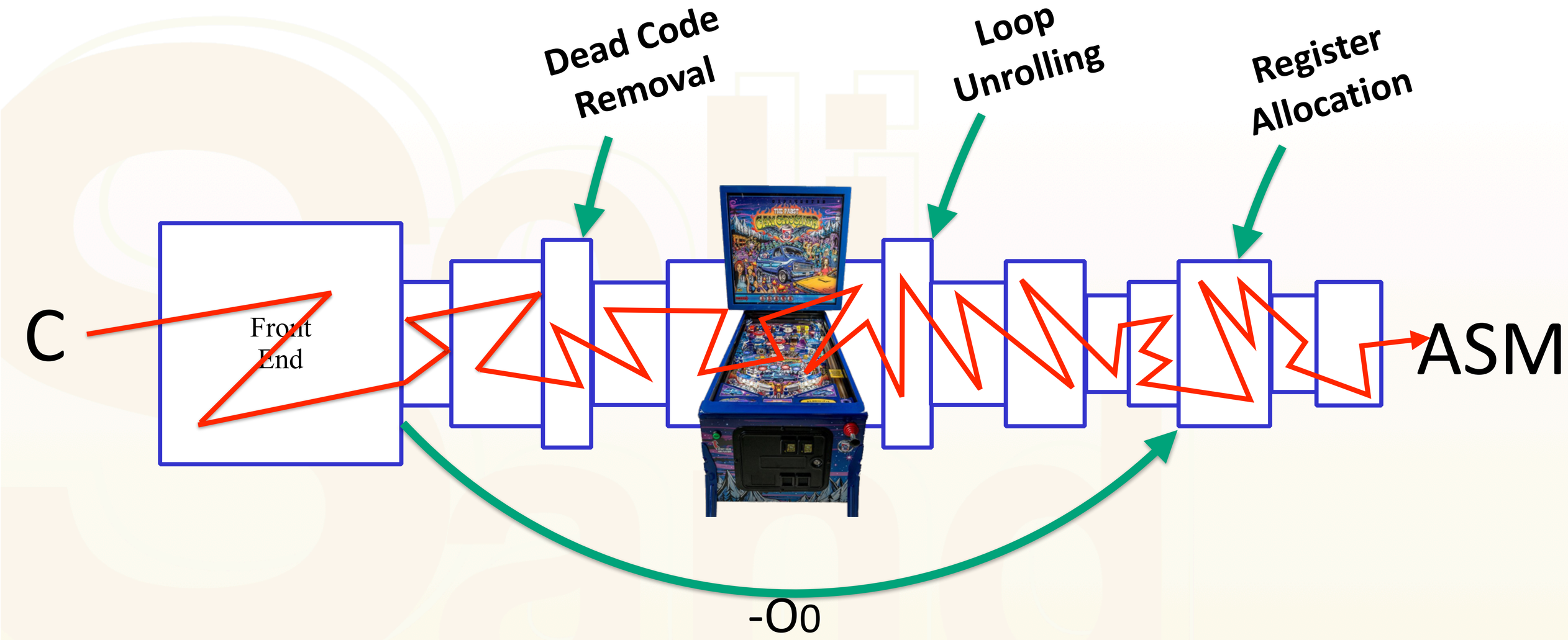
Compilers: Easy to Test in Principle, Hard to Develop

- Well defined and stable specs: ISO C/C++ std
- Single input, single out & deterministic

But need rigorous testing:

- Unstable (Pinball Effect)
- Huge in size, dev team, time-span
- Never finished
- Non-functional reqs

The Internal Structure of a Compiler



Solid Sands

cc -O1

Solid Sands

```
cc -O1
```

Solid Sands

cc -Ofast

Solid Sands

```
cc -Ofast
```

Run-Time Optimization Error SuperTest 3/5/7/tspr2388.c

```
s[0] = 42;
*(sp[0]) = -1;      /* *(sp[0]) is an alias of s[0] */
if( s[0] == 42 )    /* Incorrectly yields true */
```

- This is REALLY BAD!
- Because no optimization was specified and there is no option to turn this off
- Because it is not linked to a specific syntactical feature

Optimization Testing

```
int f(int n) {
    int total = 0;
    for (int i=0; i<n; i++){
        total += i & n;
    }
    return total;
}
```

Compile at -O2:
About 80% structural coverage at assembly with unit test:
f(999)

Needs 5 tests for maximal coverage.
Full branch coverage not possible

```
+:  test   %edi,%edi
v:  jle    0x400552 <loop+0x12>
+:  xor    %edx,%edx
+:  cmp    $0x7,%edi
>:  ja     0x400555 <loop+0x15>
-:  xor    %eax,%eax
-:  jmpq   0x400660 <loop+0x120>
-:  xor    %eax,%eax
-:  retq
+:  mov    %edi,%ecx
+:  and    $0xfffffff8,%ecx
+:  mov    $0x0,%eax
v:  je     0x400660 <loop+0x120>
+:  movd   %edi,%xmm0
+:  pshufd $0x0,%xmm0,%xmm0
+:  lea    -0x8(%rcx),%edx
+:  mov    %edx,%eax
+:  shr    $0x3,%eax
+:  bt     $0x3,%edx
>:  jb     0x4005aa <loop+0x6a>
-:  movdqa 0x17c(%rip),%xmm1
-:  pand   %xmm0,%xmm1
-:  movdqa 0x180(%rip),%xmm3
-:  pand   %xmm0,%xmm3
-:  movdqa 0x184(%rip),%xmm5
-:  mov    $0x8,%edx
-:  test   %eax,%eax
```

```
-:  jne    0x4005c0 <loop+0x80>
-:  jmpq   0x400637 <loop+0xf7>
+:  pxor   %xmm1,%xmm1
+:  movdqa 0x14a(%rip),%xmm5
+:  xor    %edx,%edx
+:  pxor   %xmm3,%xmm3
+:  test   %eax,%eax
v:  je     0x400637 <loop+0xf7>
+:  mov    %ecx,%eax
+:  sub    %edx,%eax
+:  movdqa 0x163(%rip),%xmm8
+:  movdqa 0x16a(%rip),%xmm9
+:  movdqa 0x172(%rip),%xmm6
+:  movdqa 0x17a(%rip),%xmm7
+:  nopw   %cs:0x0(%rax,%rax,1)
+:  movdqa %xmm5,%xmm2
+:  paddd  %xmm8,%xmm2
+:  movdqa %xmm5,%xmm4
+:  pand   %xmm0,%xmm4
+:  pand   %xmm0,%xmm2
+:  paddd  %xmm1,%xmm4
+:  paddd  %xmm3,%xmm2
+:  movdqa %xmm5,%xmm1
+:  paddd  %xmm9,%xmm1
+:  movdqa %xmm5,%xmm3
+:  paddd  %xmm6,%xmm3
+:  pand   %xmm0,%xmm1
```

```
+:  pand   %xmm0,%xmm3
+:  paddd  %xmm4,%xmm1
+:  paddd  %xmm2,%xmm3
+:  paddd  %xmm7,%xmm5
+:  add    $0xfffffff0,%eax
+:  jne    0x4005f0 <loop+0xb0>
+:  paddd  %xmm3,%xmm1
+:  pshufd $0x4e,%xmm1,%xmm0
+:  paddd  %xmm1,%xmm0
+:  pshufd $0xe5,%xmm0,%xmm1
+:  paddd  %xmm0,%xmm1
+:  movd   %xmm1,%eax
+:  cmp    %edi,%ecx
+:  mov    %ecx,%edx
v:  je     0x40066c <loop+0x12c>
+:  nopw   0x0(%rax,%rax,1)
+:  mov    %edx,%ecx
+:  and    %edi,%ecx
+:  add    %ecx,%eax
+:  inc    %edx
+:  cmp    %edx,%edi
+:  jne    0x400660 <loop+0x120>
+:  retq
```

Floating Point Fail on Linux Subsystem for Windows on x86


```
long double ldVar = 1.3L
assert ( ldVar + LDBL_EPSILON >= ldVar );
```

- Compiler is OK
- Windows is OK
- But integration (use case) fails

Compiler Verification, More Necessary than Ever!

- As long as there is technological progress, or new application areas, there will be new compilers
- Compilers are fundamentally unstable
- SuperTest is the best test-suite for C and C++ compilers and libraries

Thank You!

Marcel Beemster
CTO of Solid Sands
marcel@solidsands.nl