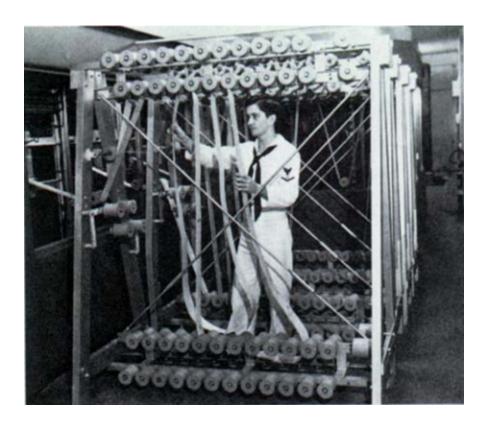
#### Once upon a time ...

A mechanic at his daily work ...



#### Immense

Mark I length 16m height 2,50m 765.000 components 650 km wires



#### WWII "Manhattan"

Calculation of implosions



#### <u>WWII – the Navy</u>



#### Salvo-calculations

#### $\underline{WWII - ENIGMA}$

#### Uncrackable German coding machine



#### Hardware

At first relais, very much of them ...



then electrotubes, very much of them



#### Hardware

Then magnetcores, very much of them



# Followed by transistors, very much of them

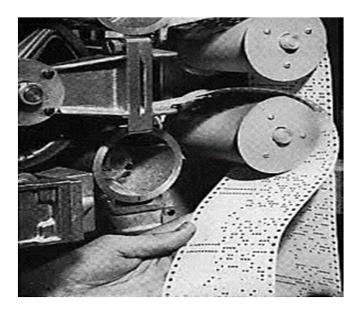


#### Data carriers

Punched tape Extremely vulnerable

Punched card

Still very vulnerable





#### Data carriers

Magnetic tape only serial access



Magnetic disk *Random access* 



#### Punched-cards sorting machine



# <u>Assembler:</u> Lowlevel Programming

Generates machine language

Varies per computer

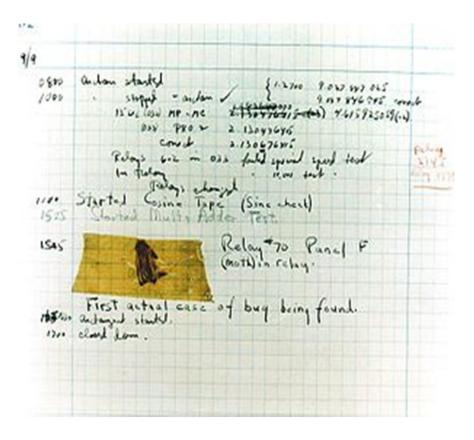
	1
Carries and Analyzamous Low-We 128	Gaunge 1. Dorner
LANTY & STATISTICS LINE IN:	
	A THE PART AND A TRACK
MARC INTER	tions
	a fra att-of
an advertision and	a bill an by and
N. M. WA	be Side large n
14 64 - 10 - 14 - 1	a Sign Mundel, Indexing NOT SIG
he is with its comments in	· Setting at, adams var: 82
	· SATE IN THAT WERE ADDRESS AND A STATE
the bit of the desired of a	in Stillaubt, odefing light R.s.
	n BC winds from here to C, brives Ligners m
as by otherses	m SZ words from black to ril, technomit, sponts m
an hit officiences	- X
	er (C - n far bell) (Croppi - Channel
an id of a state	en million media to tope, 100 publics.facts
	er Pluint Ban, bras m
	m m brau+3/1 tes. 25 pelat/24
	an Finalise legans and interliants leganse m
De Brig-Ill runfe oatst be K	ter Tupe per mei inte m
En On 100 1, record 100 000 10-02 0 1 3	0e 1(3 + e fas. n.455
Re - 30- 20 10-02-06 4	Om (K) - n fies # +50
The IP Set in 102, insufar random 2-10 0	De Mitt da vod hen e
	Cer Bits UMAS, Igane #
	n Bradgood sity, igners n
	Des Cestillowi krisibe breakpolet who hen Cestiliumi krisibe breakpolet kire
No BEARS-AR MARK	A Contrast Long Landon and
The state of the s	A STATE OF A
CONTRACT OF T	A STATE OF A
	A CONTRACTOR OF
UNAC Pase	Code
394 NON NO XE OF NA ALC DA	BREAK HAD NO. 100 LON TOP 181.
00 / 0 - 0 1 5 5 4	8 6 7 3 9
W	2 7 0 4 1
	1 3 8 3 8 4 8
3 7 7 7 7 7	
23 1 5 0 5 2	· ¥ 2 Y 2 V
	(rat)
1. 000	on bit bren sits
N. Much	144.10
1.4	party surged fill
	(ditrest last sales its
8.9	apose mirrori
	the local and the local data in the local data in the
the second se	Ends our
	$\begin{split} & \begin{array}{c} & \begin{array}{c} & \begin{array}{c} & \begin{array}{c} & \begin{array}{c} & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \end{array} \\ & \end{array} \\ & \end{array} \\ & \begin{array}{c} & \end{array} \\ & \begin{array}{c} & \end{array} \\ \\ & \end{array} \\ \\ & \end{array} \\ \\ & \end{array} \\ & \end{array} \\ \\ \\ & \end{array} \\ \\ \\ \\$

						( )		
		TIMINGS	SYS	TEM	TIMINGS	<b>IBM</b> 1401 Data Processing System Reference Card		
Key to abbreviations used La = Length of the Ad		emulas		98		INSTRUCTION FORMAT		
La = Length of the B-f			Punch a Cord	4	.0115 (L + 1) + 1/O	The IBM 1401 Data Processing System uses a variable was		
Lo = Length of Multipl		field	Read a Cord	1	.0115 (Li + 1) + 1/O	length concept, the length of an instruction can very from a		
Le = Length of Instruc			Read and Punch	5	.0115 (Lt + 1) + 1/O	to eight characters.		
Let = Length of Multipl			Select Stocker	к	$.0115 (L_1 + 1)$ .0115 (L_1 + 3)	OP CODE A- or I-ADDRESS B-ADDRESS d-CHARACTER		
La = Length of Divisor			Start Punch Feed*	:	.0115 (L + 3)	X XXX XXX X Op Cede: This is always a single character which defines the		
La = Number of signif	leant d	laits in Divisor (Excludes binh-	Stort Read Feed*	11	.0115 (4 + 1)	hosis operation halon performed. A word mark is always on		
order 0's and bla			Store A-oddress Register*	l õ	.0115 (1. + 5)	sociated with the operation code position of an instruction.		
Ly = Length of A- or B			Store B-oddress Register*	н	.0115 (L + 4)	A-Address: This always consists of three characters. It can iden- tify the units position of the A-field, or it can be used to see		
Lx = Number of chara		o be cleared ack to right-most "O" in control field	Subtract (no recomplement)		.0115 (Ly + 3 + La + La)	lett a special unit or facture (tone unit, column biogra-		
Ly = Number of chara			Subtract (recomplement)	5	.0115 (L1 + 3 + La + 4 La)	feature, disk storage, inquiry, etc.).		
1/0 = Timing for Input			Write a Line Write and Punch	1	.0115 (L1 + 1) + 1/O	I-Address: Instructions that can cause program branches use the I-address to specify the location of the next instruction		
Fm = Forms movement	times.	Allow 20 ms for first space, plus 5 ms	Write and Punch Write and Read	:	$.0115 (l_4 + 1) + 1/0$ .0115 (l_4 + 1) + 1/0	to be executed if a branch accurs,		
for each addition	al spa	•	Write, Read and Punch		.0115 (4 + 1) + 1/0	B-Address: This is a three-character storage address associated with the B-field. It usually addresses the units position of the		
T_ = Tope movement t		States and a state of the state	Zero and Add		.0115 (Li + 1 + La + La)	B-field, but in some operations, such as tape or disk record		
X = Number of fields	s inclu	ded in on operation	Zero and Subtract		.0115 (Li + 1 + La + La)	read and write, it specifies the high-order position of a record storage area.		
Add (no recomplement) Add (recomplement)	^	.0115 $(l_1 + 3 + l_2 + l_3)$ .0115 $(l_2 + 3 + l_3 + 4 l_3)$	TAPE	a data	BRATIONS	d-Character. The d-character is used to modify an operation code. It is a single alphabetic, numerical, or special charac- ter, positioned as the last character of an instruction. It can be used with instructions of any length.		
Bronch	ŝ	0115 (1 + 1)	C = Character Rate			PROCESSING OVERLAP		
Bronch if Bit Equal*	w	.0115 (L1 + 2)	729 II at 200 epi = .06 at 556 epi = .02	7 ms		A-Address		
Branch if Character Equal		.0115 (L <sub>1</sub> + 2)	729 IV et 200 cpl = .04	4 ms		The hundreds position of the A-address of a tape or input-		
Branch if Indicator On Branch if Ward Mark		.0115 (Lr + 1)	et 556 cpi = .01 7330 et 200 cpi = .13	6 ms				
and/or Zone	v	.0115 (1+ + 2)	et 556 cpi = .13	0 ms		to @. The symbol is used to signal an overlap operation with		
Clear Storage	1	.0115 (L + 1 + L)	729 Model II, Read 10.7 + 0	CN ma	= TAU interlocked	character reader, magnetic tope, paper tope, and data trans-		
Clear Word Mark	-	.0115 (L + 3)	10.5 + 0	Nms	Processing interlocked TAU interlocked	Overlag Mode		
Compare	c	.0115 (L) + 1 + La + La)	7.5 + 0	Nma	= Processing Interlocked	The following instructions are used when the system is in the		
Control Carriage		.0115 (Li + 1) + P.	729 Model IV, Read 6.8 + 0	N me	= TAU interlocked	overlap mode and card, printer or serial 1/O operations are		
Control Unit Divide (over.)*	2	$.0115 (L_1 + 1) + T_m$ .0115 (L_1 +2 +7 La Le + 8 Le)	6.7 + 0	N ms	= Processing interlocked	to be performed.		
Halt		.0115 (L1 + 2 + 7 La L6 + 8 L6)	wine 7.8 + 6	None	TAU interlocked	INSTRUCTION FUNCTION		
Load Characters to A			7330 Read 20.5 + 0	Nm	= TAU interlocked	K1 Overlap On K(0)1 Overlap On And Branch		
Word Merk	L	.0115 (L) + 1 + 2 L)	7.7 + 0	Nms	= Processing Interlocked	Ke Overlap Off		
Modify Address*	#	.0115 (L1 + 9)	Write 20.3 + 0	Nms	= TAU interlocked = Processing interlocked	K(I) Overlap Off And Branch		
Move Characters to A or B Word Mark		.0115 (4 + 1 + 2 4=)	Rewind			K[] Resot Overlap		
Move Characters and Edit	2	.0115 (L + 1 + L + L + L + L + L )	729 Model II = 1.2 mil 729 Model IV = .9 mil	nutes/r	ree!	K(1)		
Move Characters to Record	-		737 Model IV = .9 mil 7330 (High Speed) = 2.3	nutes/r	test (real	1405 TIMING		
or Word Mork*	,	.0115 (L, + 1 + 2 L)	Skip and Blank Tope			1405 TIMING		
Move Characters and Suppress Zeros	z		(add to subsequent write tim	•)		TIMINOS (Medel 2) MAX. AVO. MIN.		
" Move and Insert Zeros"	ž	.0115 (L + 1 + 3 L) .0115 (L + 1 + 2 3 L + 3 L)	729 Model II = 40.5 m 729 Model IV = 27 m			Disk to Disk 800 ms 600 ms 450 ms		
Move Numeric	2	.0115 (1 + 3)	7330 = 103 ms			Track to Track 250 ms 175 ms 100 ms Record to Record, some Track 50 ms 25 ms		
Move Zone	Ŷ	.0115 (1+ 3)	Backspace (ofter Read)		Backspace (after Write)	20 ml		
Multiply (over.)*		.0115 (L+ +3 +2 Lo +5 Le Lx +7 Lx)	729 Model II = 46 + Ch 729 Model IV = 33 + Ch	-	729 Model II = 52 + CN ms 729 Model IV = 37 + CN ms	International Business Machines Corporation		
No Operation	N	.0115 (Lt + 1)	7330 = 428 + CN ms		7330 = 435 + CN ms	Data Processing Division		
· · · · · · · · · · · · · · · · · · ·	22,2,09	CANAR CONTRACTOR	CELEVAN AND STORE	14	CONTRACTOR CONTRACTOR PROCESSION	112 East Post Road White Plains, N. Y. 03 1959, 1960, 1961 by International Business Mathines Corporation		

# <u>Compiler:</u> <u>Highlevel Programming</u>

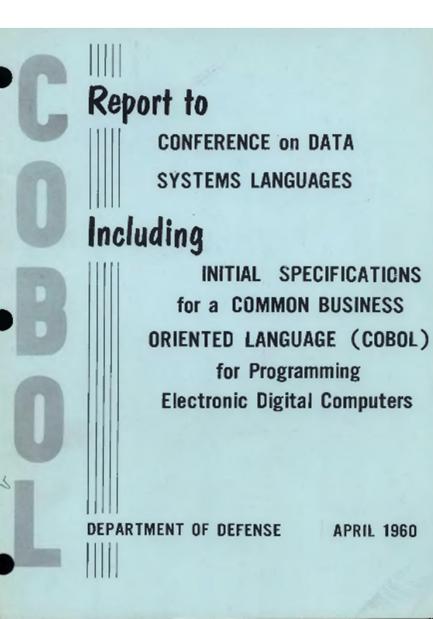
Generates assembler-code

1952 First compiler"FlowMatic"by Grace Hopper



### <u>1959: COBOL</u>

#### Thanks to DoD...



Eisenhower president



Kennedy running



#### Revolution Fidel Castro



#### No Berlin wall yet



#### Deadlock

#### NO COBOL NO BID



### First COBOL compilation

1961

A COBOL program compiled at two computers



### 1962 COBOL arrives in Europe

- in The Netherlands
- 2 teachers
- 16 students



#### 1969 Microprocessors

#### Scale-decrease



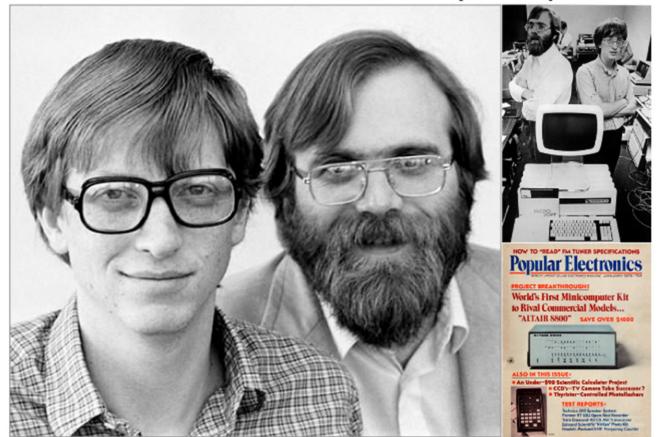
#### Used in Apollo AGC

25% COBOL





April 4, 1975: Bill Gates & Paul Allen form "Micro-soft" partnership



1975 Microsoft

#### And more ...





### And only in 1998

Google (originally Googol)



30	earch the web using Google!
10 results	Google Search (I'm feeling lucky)

#### About Google!

Stanford Search Linux Search

Get Google! updates monthly!

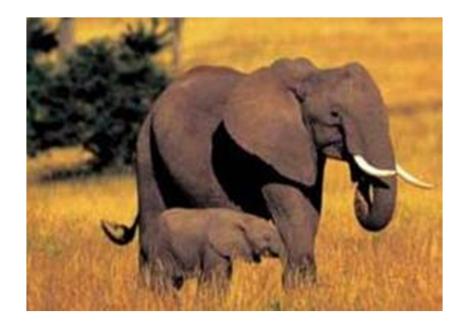
your e-mail

Subscribe Archive

Copyright @1997-8 Stanford University

#### Today's Ratio

#### COBOL : GOOGLE 7 : 1



### Surprisingly after 58 Years



COBOL Alive and kicking! 25 billion financial transactions / day (13 x / day) 70-80% in COBOL

Creditcards ATM's Tickets Banking Mobile phones Stockmarkets Taxes Insurances



KLM		rfight Pesseng MOD	ar nama	KLM KLKD
Passenger	Date JAN	UARY 6, 2013		Passenger
Fight Number	Departure Airport	Board	Seg Care	San Franciaco to Ameterdam
<b>KLM 39</b>	SFO/T2	09	10 at D10	09:40 Economy
Boarding Priority	Where's my seat?		Departing	
1234	A22	+ =	09:40 >	A22
Frequent Flyer 2073621900	Economy			105525 11

17.34	-7.89% 254.23 126,888
34.89	+2.13× 100.08 120.000 +2.13× 564.23 988.000
23.67	-11. 6x V 120.34 380.000
34.64	+23.1x▲ 893.23 128.000 +5.56x▲ 128.98 328.000
12.78	-3.67% + 432.12 758.080
13.44	+2.54× 432.24 128,080



# Impressive figures

 300 billion COBOL lines
 300.000.000.000

 5 billion new lines aeach year
 5.000.000.000

 Total investment 2 trillion
 \$ 2.000.000.000

And ... 1.500.000 COBOL developers



# Impressive figures

You just don't throw that away



#### End of WW II

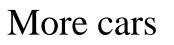


# Explosion of data

- production
- Bank accounts
- transport and logistics
- government
- trading
- insurances



#### Worldwide communications







#### More telephones







#### Worldwide communications

#### More TV's and TV-stations





FOX	(B)	амс	FX	CINN	HGTV
CN	HISTORY	<b>()</b>	<b>1</b>	FS1	tbs
[adult swim]	Tis.		<b>₽</b> %€	VICELAND	tru®.
Lifetime	travel		newşy	Bloomberg	
Univision	()iiias	POLARIS+	MAKER.	品	Oatorisión

#### New: satellites



### Travelling

Travel-agencies

#### Airlines



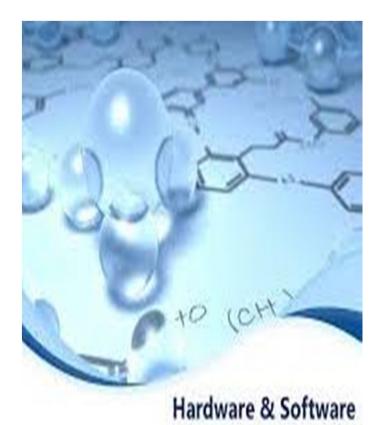
Railways

Hospitality



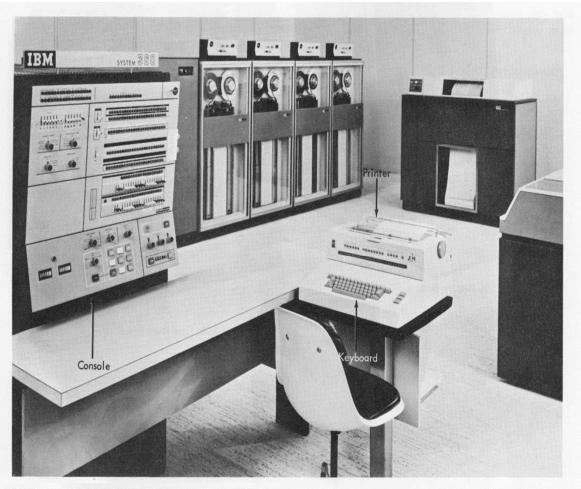
# A growing need

- new hardware
- robust software
  - . fast
  - . data-driven
  - . reliable



### Hardware

1964: IBM announces System 360



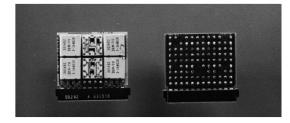
IBM System/360 Model 40 Console, Console Printer, and Keyboard

### Break through

a. New technology

b. Smaller

c. Modular







#### Software: COBOL



Fast to learn

#### Robust

#### Reliable

#### Four unique characteristics

- 1. record-structuring
- 2. mass data-processing
- 3. decimal calculating
- 4. report generating



#### <u>1981 IBM-PC</u>

- Soon very popular

- sales far beyond estimations

- PC = IBM



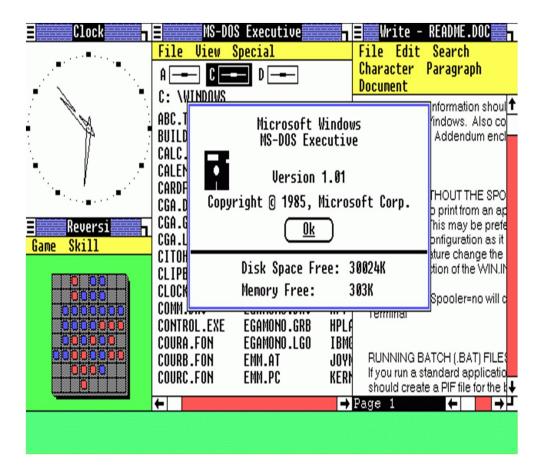
# Software for PC's 1981

#### 1<sup>st</sup> MS-DOS

Drives		Logical dri	un C: -		
HDD 80h	CIN	Logical all	ve u.		
Logical C:					
└→ Logical D:	Name	Size	Attrib	Date	Тіме
└→ Extended	MICROS~1	< <folder>&gt;</folder>		04.02.2002	
└→ Logical E:	MULTIM~1	< <folder>&gt;</folder>		14.10.2001	22:29
└→ Logical F:	PLATF0~1	< <folder>&gt;</folder>		15.03.2002	16:15
└→ Unallocated	PROGRA~1	< <folder>&gt;</folder>		14.10.2001	08:11
HDD 81h	RECYCLED			26.11.2001	
→ Unallocated	RECYCLER			07.02.2002	
HDD_82h	Темр			15.03.2002	
└→ Extended	WINNT			14.10.2001	
➡ Unallocated	WinHex	the second se		11.04.2002	
└→ Logical G:	AUTOEXEC BAT			14.10.2001	
	CONFIG SYS			14.10.2001	
	IO SYS			14.10.2001	
	MFT DAT			10.03.2002	
	MSDOS SYS			14.10.2001	
	NTDETECT COM			14.10.2001	
	boot ini			14.10.2001	
	ntldr			14.10.2001	
L PA Hala L	pagefile sys				
F1-Help Tab-Long names ENTER-Preview Ctrl+0-Copy					
Active® NTFS Reader for DOS v 1.0 2002 (C) Active Data Recovery Softwar <pre>KFREE&gt;</pre>					
CKLE7				пстр.// МММ	1112.0

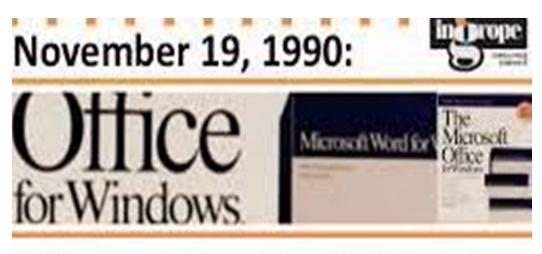
### Software for PC's

#### 1985 1<sup>st</sup> WINDOWS



### Software for PC's

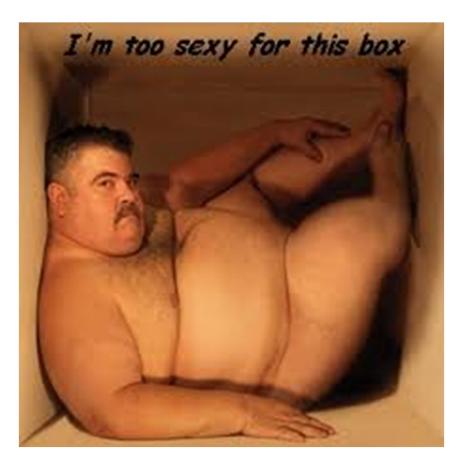
#### 1990 1<sup>st</sup> WORD, EXCEL



Microsoft Office for Windows is released otherwise known as "Office 1.0". Office 1.0 contains Word 1.1, Excel 2.0 and Powerpoint 2.0.

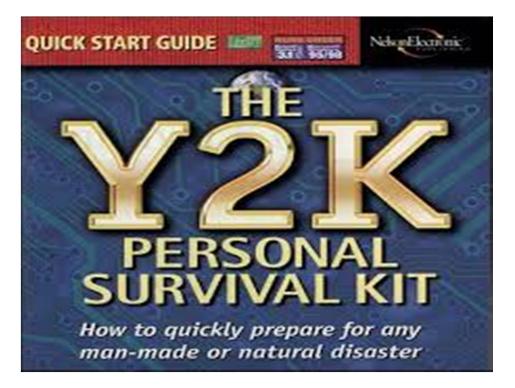
#### COBOL not sexy anymore

COBOL-scriber's image



#### <u>Y2K</u>

Catastrophic? Real problem? Hoax?



### <u>GETTING OLDER</u>

And remain yourself



#### **Developments**

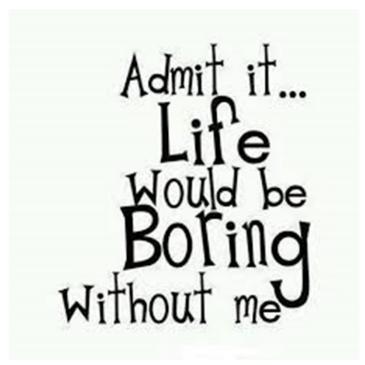
## From primitive to advanced



From simple to complex



#### Boring



#### With deep sadness ...



#### Criminal act



#### <u>Old, so ...</u>



Useless! Get out! Never again! Or ...?



#### Metric versus Imperial

Meter: light in vacuum in 1/299.792.458 sec



Inch: 2,54 cm



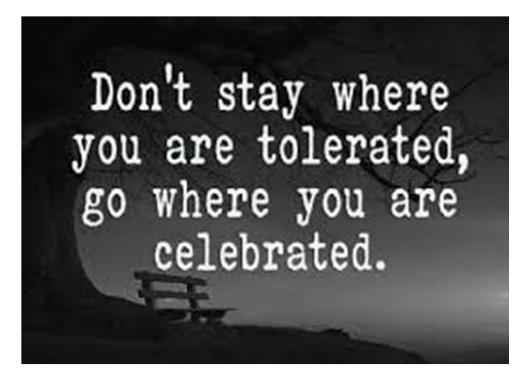
#### To switch or not to switch

- too expensive
- keep your investments
- hardly any ROI
- current system
- is efficient
- no real
- improvements



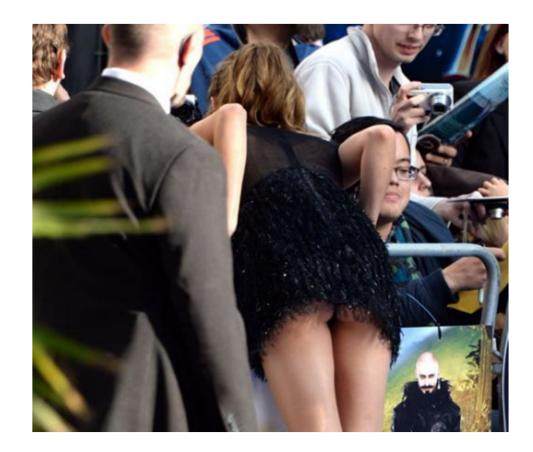
### Effective AND good

Stick to what you have





#### COBOL programmers: becoming scarce



#### Comeback

Necessary
 Business along
 with education
 authorities



Pimping

– Change position

– Improve earnings

– User groups





#### Is your system really that bad?

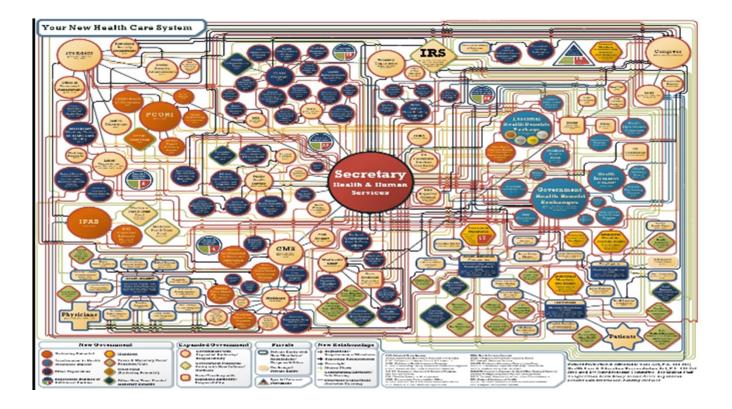


#### Performance

- . response-times
- . use of resources
- . availability
- . data management



#### Complexity = diversity \* dependency



Replacing ...



\*ADAM

#### or upgrading?





LastingCOBOL support

LifecycleCOBOL-market:long and strong



Migration

Pre-analysis:

. duplicates



- . unused routines
- . inefficiënt coding

Code-translation:

- inefficiencies being copied
- new inefficiencies
- more complex results
- increasedmaintenance



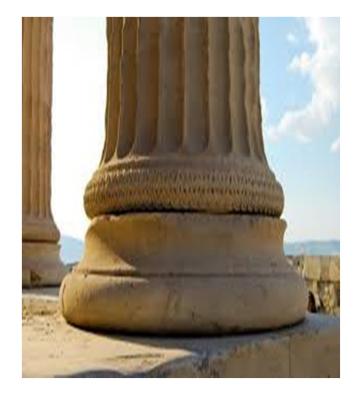
COBOL remains the fundament

- investments

committed vendors

- COBOL in 2050 still popular!

Special care: shortage of programmers



Consider migration when:

- . Change of database-systems
- . Change of functionality
- . Hardware adjustment
- . Integration of companies

```
By failing to
prepare, you are
preparing to
fail.
Benjamin Franklin
```

Migration risks:

- equal or less functionality
- maintenance problems
- increased runtime
- no added value



COBOL replaced?

In the long runNot by just one language

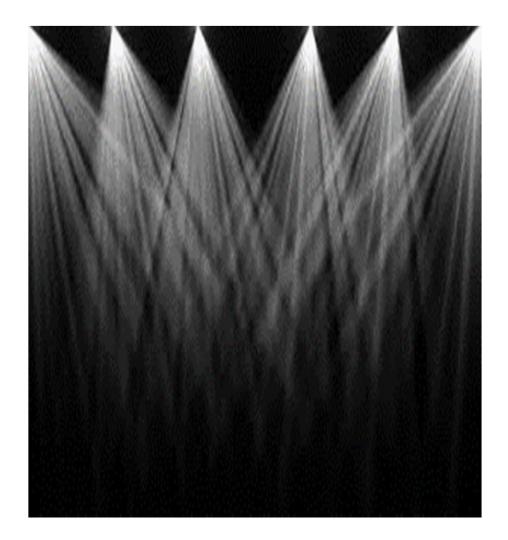


#### At the origin of IT: COBOL

- never matched
- no successor yet



#### Future



#### There is enough work to be done!

