More than Moore

Transistors bought per $, m

Intel

Year:
2002  04  06  08  10  12  15

Number of Transistors:
0  5  10  15  20
More than Moore – More Opportunities!

Transistors bought per $, m

Who will fuel the race for smarter devices?
More than Moore – More Opportunities!

Transistors bought per $, m

Performance/Cost (log)

CMOS Technology

New Technology

Our Challenge

[80s 90s 00s 10s 20s 30s 40s]

2002 04
Smarter Devices?

convergence of reactive control, computation, information, communication

PROCESSING, ABSTRACTING, UNDERSTANDING AS EARLY AS POSSIBLE

SMART SENSORS

INTERNET OF THINGS

(CYBER) PHYSICAL SYSTEMS

CLOUD/HPC

BIG DATA

COGNITIVE COMPUTING / DATA ANALYSIS

GLOBAL INTEGRATION OF COMMUNICATION, COMPUTATION AND REACTION

TRANSFORMING DATA INTO INFORMATION

NEW SERVICES

Fog computing
Edge computing
Stream analytics
Fast data by real-time
micro-servers and even Nano-servers (concentrator, fusion of several sensors) In different layers (Onionskin)
Smarter Devices?

convergence of reactive control, computation, information, communication

Software Crisis

CYBER) PHYSICAL SYSTEMS

SMART SENSORS

INTERNET OF THINGS

CLOUD

BIG DATA

COGNITIVE COMPUTING / DATA ANALYSIS

GLOBAL INTEGRATION OF COMMUNICATION, COMPUTATION AND REACTION

NEW SERVICES

TRANSFORMING DATA INTO INFORMATION

Fog computing
Edge computing
Stream analytics
Fast data
by real-time
micro-servers
and even
Nano-servers
(concentrator, fusion
of several sensors)
In different layers
(Onionskin)
The major cause of the software crisis is that the machines have become several orders of magnitude more powerful! To put it quite bluntly: as long as there were no machines, programming was no problem at all; when we had a few weak computers, programming became a mild problem, and now that we have gigantic computers, programming has become an equally gigantic problem.
The major cause of the software crisis is that the machines have become several orders of magnitude more powerful! To put it quite bluntly: as long as there were no machines, programming was no problem at all; when we had a few weak computers, programming became a mild problem, and now that we have gigantic computers, programming has become an equally gigantic problem.

— Edsger Dijkstra, *The Humble Programmer* [Dijkstra 72]
The New Software Crisis

FACING A NEW SOFTWARE CRISIS AND ITS COURSE OF ACTION

‘ The major cause of the software crisis is that the machines have become several orders of magnitude more powerful! To put it quite bluntly: as long as there were no machines, programming was no problem at all; when we had a few weak computers, programming became a mild problem, and now that we have gigantic computers, programming has become an equally gigantic problem. ’

— Edsger Dijkstra, The Humble Programmer [Dijkstra 72],

Correctness challenge
Performance challenge
Data challenge
Holistic and interoperability challenge
Software Crisis – Opportunities

Focus on high-productivity, high value software

➢ Higher level, reactive programming languages
➢ Correct-by-construction approaches
➢ Ubiquitous parallelism
➢ Ubiquitous distribution: elasticity, heterogeneity (edge/fog)
➢ ‘Non-functional programming’: time, resources, faults...

Invest in tools and reference platforms:
larger businesses, virtually vertical organizations, and funding agencies need to understand the urge and value in supporting a sound ecosystem of tools and platforms

Develop new computing modalities in HW and SW:
dynamicity, adaption, learning and reasoning, accuracy, trust, predictability, agile development… without throwing validation, verification, certification, quality away

And… engage into standards committees