

An Overview of Software Evolution

CPRE 556: Lecture 3

Software Evolution

- What is it?
- How important is it?
- What to do about it?

An early history of software engineering

- The following slides provide a condensation of the ideas of Robert L. Glass in his book "In the Beginning: Recollections of Software Pioneers" about the history of software engineering.

The Pioneering Era (1955-1965)

- New computers were coming out every year or two.
- Programmers did not have computers on their desks and had to go to the "machine room".
- Jobs were run by signing up for machine time. Punch cards were used.
- Computer hardware was application-specific. Scientific and business tasks needed different machines.

The Pioneering Era (1955-1965)

- High-level languages like FORTRAN, COBOL, and ALGOL were developed.
- No software companies were selling packaged software.
- Academia did not yet teach the principles of computer science.

The Stabilizing Era (1965-1980)

- Came the IBM 360.
- This was the largest software project to date.
- The 360 also combined scientific and business applications onto one machine.

The Stabilizing Era (1965-1980)

- Programmers had to use the job control language (JCL) to tell OS what to do.
- PL/I, introduced by IBM to merge all programming languages into one, failed.
- The notion of timesharing emerged.
- Software became a corporate asset and its value became huge.
- Academic computing started in the late 60's.
- Software engineering discipline did not yet exist.

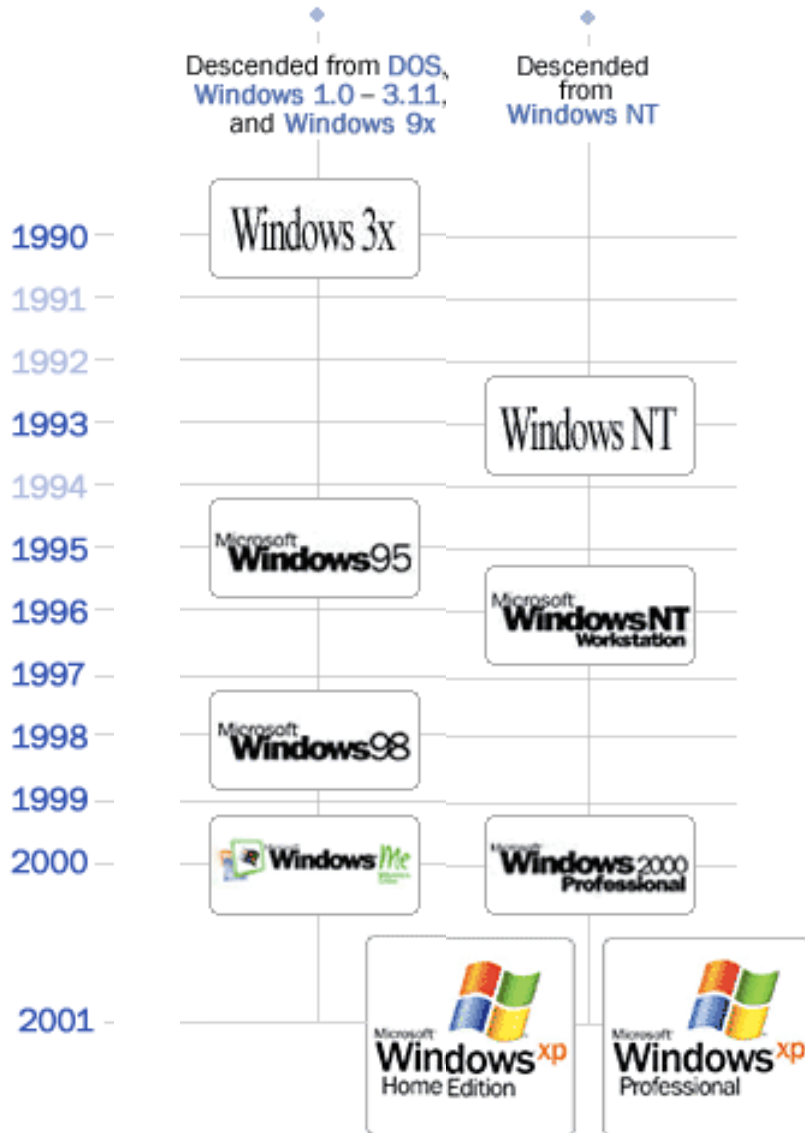
The Stabilizing Era (1965-1980)

- High-hype disciplines like Artificial Intelligence emerged..
- Structured Programming burst on the scene.
- Standards organizations became control battle grounds. .
- Programmers still had to go to the machine room.

The Micro Era (1980-Present)

- The price and size of computers shrunk. Programmers could have a computers on their desks.
- The JCL got replaced by GUI.
- The most-used programming languages today are between 15 and 40 years old. The Fourth Generation Languages never achieved the dream of "programming without programmers".

History of Microsoft Windows



November 10, 1983,
Microsoft announced
Microsoft Windows®, an
extension of the MS-DOS®
operating system.

This marked the beginning of
the graphical user interface
(GUI) era at Microsoft.

Beginning of Linux

Linus Torvalds was a second year student of Computer Science at the University of Helsinki.

From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)

Newsgroups: comp.os.minix

Subject: What would you like to see most in minix?

Summary: small poll for my new operating system

Message-ID: <1991Aug25.205708.9541@klaava.Helsinki.FI>

Date: 25 Aug 91 20:57:08 GMT

Organization: University of Helsinki

Hello everybody out there using minix -I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things). I've currently ported bash(1.08) and gcc(1.40),and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)
Linus (torvalds@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT portable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-).

Operating System/360

1965-1972

- The not-unexpected passing away of OS/360, in its 21st release, was announced in every major city of the world on August 2, 1972 by its guardian, IBM.
 - The offspring first saw the light of day in December 1965 and the birth announcement recorded a weight of 64K. It rapidly became apparent that OS, in spite of its unusual size, was more than normally subject to childhood diseases. For a long period, this weak and sickly baby hovered close to death despite almost continuous transformations and major transplants of several vital organs. Many experts are of the opinion that the huge weight of OS at birth contributed greatly to its early ill health. OS is survived by two lineal descendants, OS/VS1 and OS/VS2. It will be mourned by its many friends and particularly by the over 10,000 system programmers throughout the world who owe their jobs to its existence.

History of Cobol

- COBOL (Common Business Oriented Language) was developed under the auspices of the U.S. Department of Defense. The initial specifications for COBOL were presented in April of 1960.
- COBOL has emerged as the leading data processing language in the business world.
- ISO and ANSI committees have completed the latest revision of COBOL in 2002.

History of Java

- **January 15, 1991**
 - "Stealth Project" (as named by Scott McNealy) brainstorming meeting in Aspen with Bill Joy, Andy Bechtolsheim, Wayne Rosing, Mike Sheridan, James Gosling and Patrick Naughton.
- **February 1, 1991**
 - Gosling, Sheridan, and Naughton begin work. Naughton focuses on "Aspen" graphics system, Gosling on programming language ideas, Sheridan on business development.
- **June 1991**
 - Gosling starts working on the "Oak" interpreter, which, several years later (following a trademark search), is renamed "Java."
- **May 23, 1995**
 - Sun formally announces Java and HotJava at SunWorld '95.

How important is it?

- Windows and Solaris expand well into the range of 30 to 50 million lines of code.
- Project managers have learned to devote as much time to combing the tangles out of legacy code as to adding new code.
- While the microchip performance has increased dramatically, software's inability to scale at even linear rates has gone from dirty little secret to industry-wide embarrassment.

Legacy Cobol Code

- 200 billion lines representing 60% of world's software and it processes 85% of global business data.
- The Gartner Group estimates
 - more than ten thousand mainframes running about 200 billion lines of Cobol code
 - the cost of manually rewriting is between \$6-20 per line.
- Global investment in COBOL technologies, staff, and hardware: \$5 trillion.

Source: "India tops the charts in the legacy conversion market", Express
1/10/2006
1/10/2006

Lecture Notes – Copyright © 2006
S. C. Kothari, All rights reserved

Topics to be covered

- Analyzing and maintaining software quality: inspections, defect identification and analysis.
- Program comprehension: extracting analyzing, organizing, and visualizing program artifacts.