

software engineering

arbeitsgruppe wirtschaftsinformatik
- managementinformationssysteme -
prof. dr. hans-knud arndt

- Softwarekrise (1):
 - Seit Mitte der 1960er Jahre identifiziertes Phänomen
 - Bei gigantischen Computern ist Programmierung ein ebenso gigantisches Problem (Edsger W. Dijkstras, 1972)
 - „Die relativ stärker steigenden Kosten der Software haben inzwischen schon dazu geführt, Hardware(-elemente) als Zugabe zu geben, um den Kauf teurer Softwarepakete anzukurbeln.“
(Vorlesungsskript Planung und Entwurf betrieblicher Informationssysteme, Prof. Dr. D. B. Pressmar, 1988)

software engineering (2)

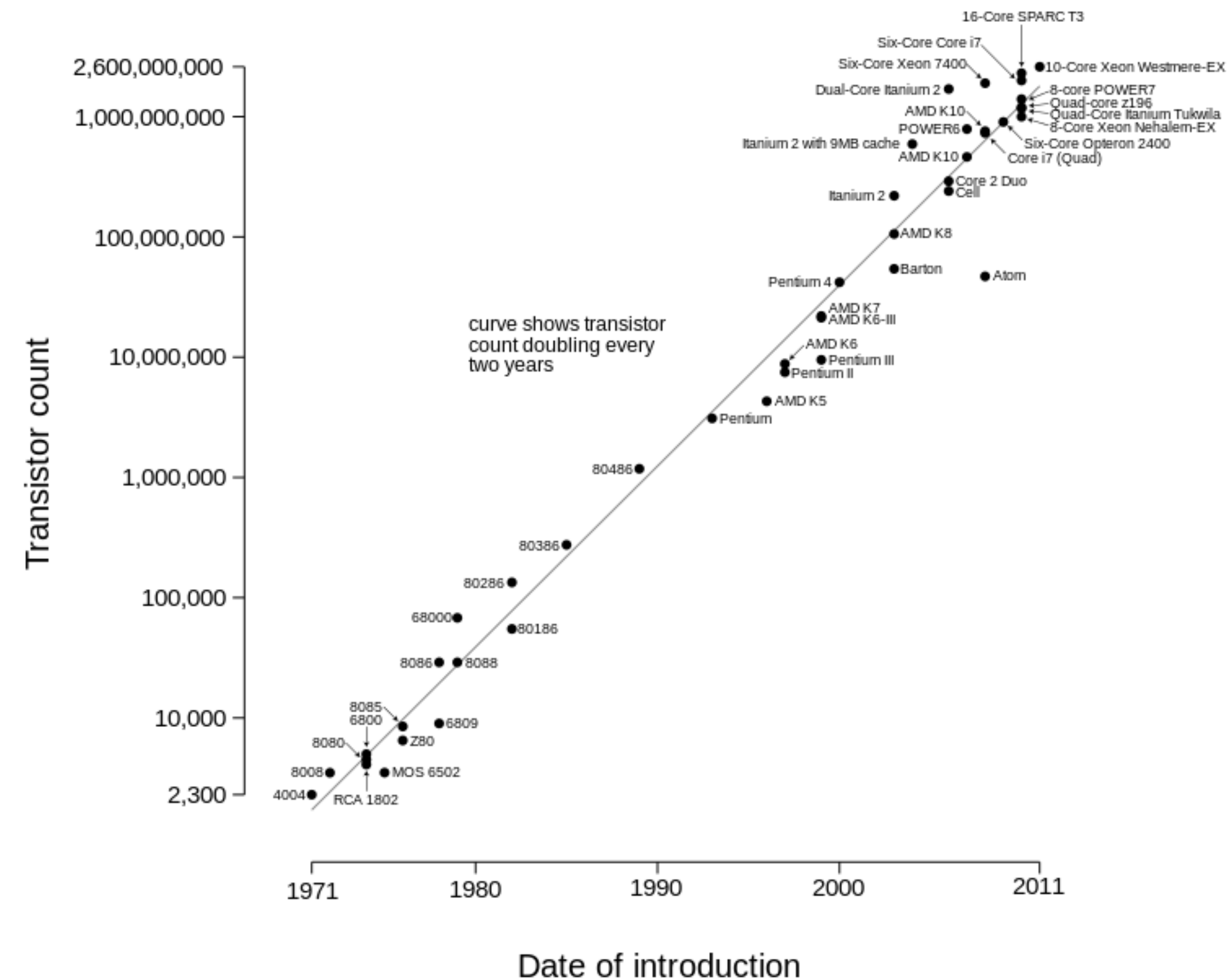
Softwarekrise (2):

Moore'sches Gesetz:

Quelle: Wgsimon, 2011:

https://de.wikipedia.org/wiki/Mooresches_Gesetz#/media/File:Transistor_Count_and_Moore%27s_Law_-_2011.svg

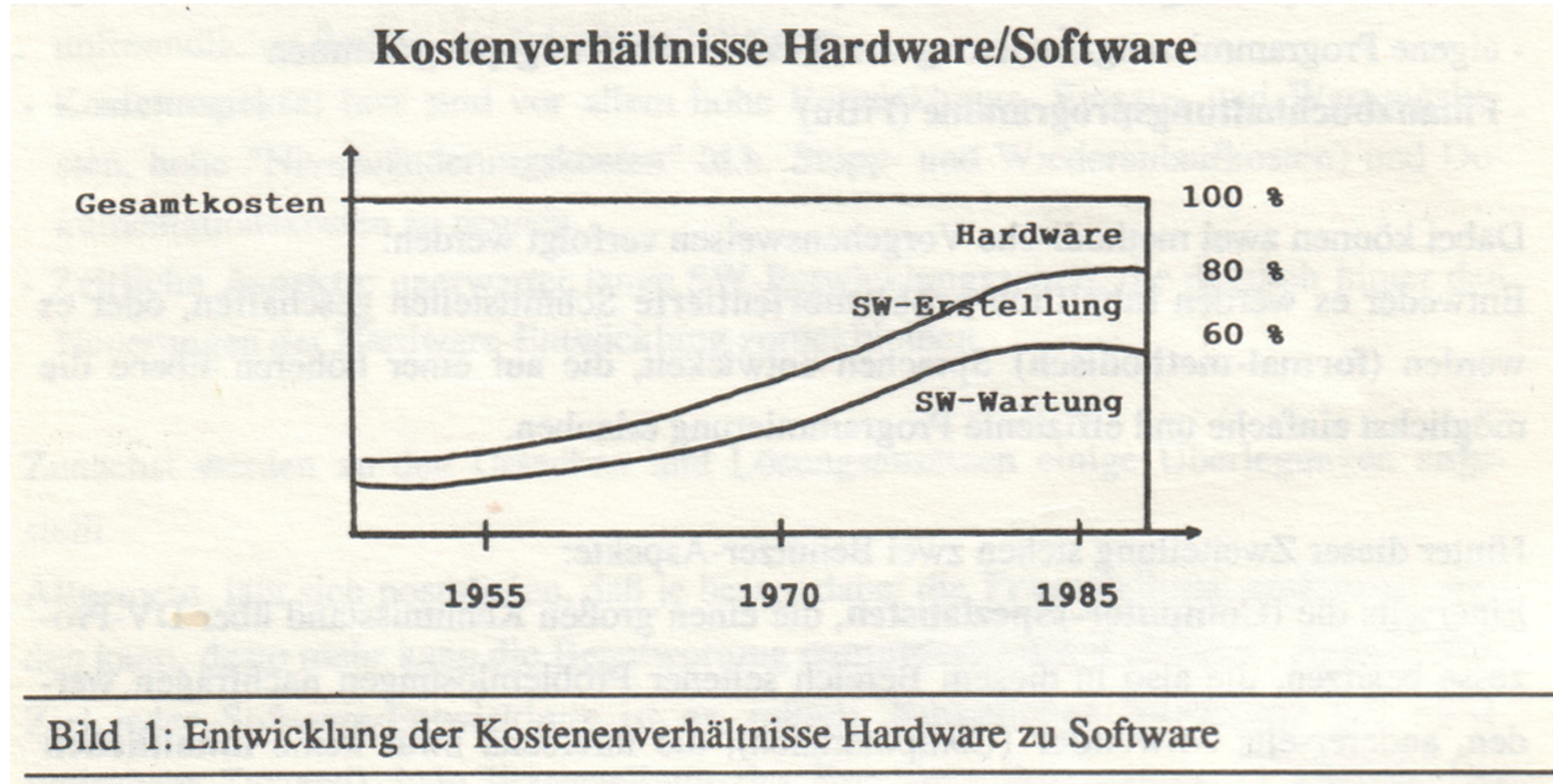
Microprocessor Transistor Counts 1971-2011 & Moore's Law



software engineering (3)

Softwarekrise (3):

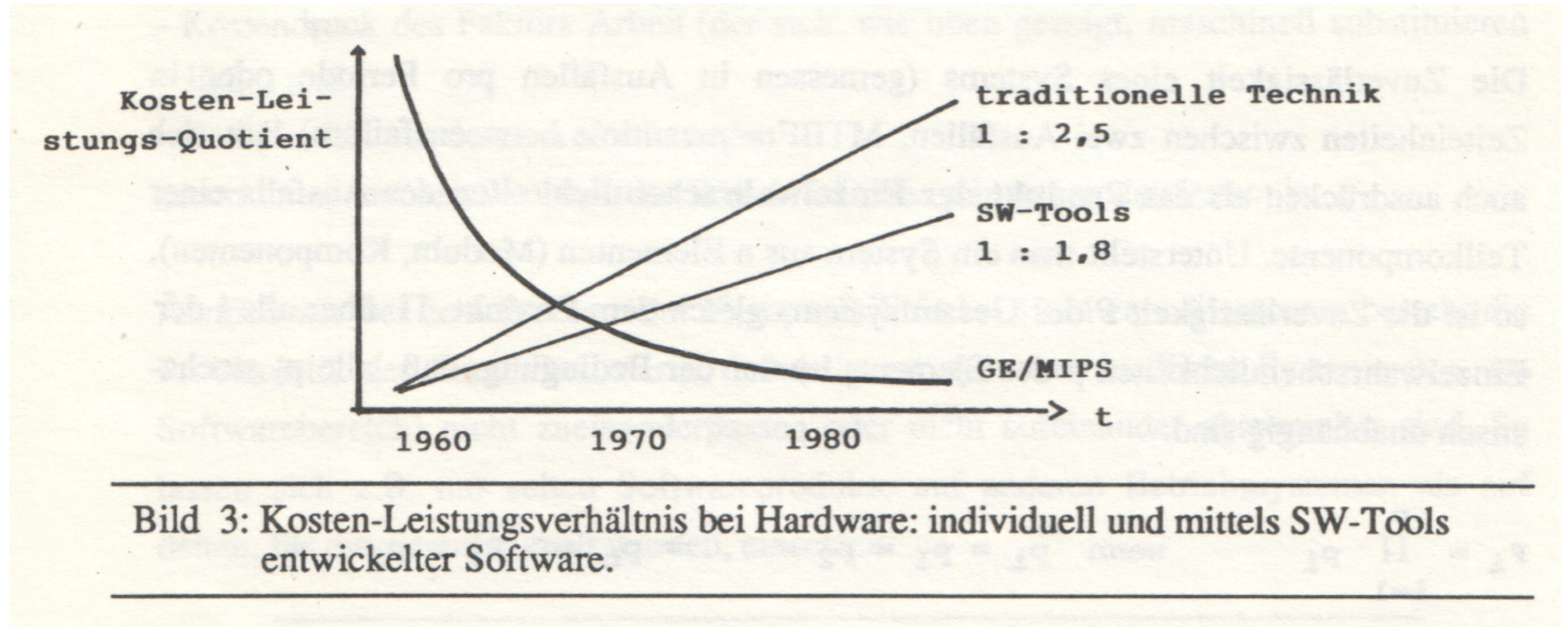
(Quelle: Vorlesungsskript Planung und Entwurf betrieblicher Informationssysteme, Prof. Dr. D. B. Pressmar, 1988)



software engineering (4)

Softwarekrise (4):

(Quelle: Vorlesungsskript Planung und Entwurf betrieblicher Informationssysteme, Prof. Dr. D. B. Pressmar, 1988)



software engineering (5)

- Softwarekrise (5):
 - Weitere Ursachen:
 - Mangelnde Qualitätssicherung
 - Mangelnde Projektplanung
 - Mangelnde Einbeziehung von Anwendern/
Kunden
 - Mangelnde Standardisierung
- Lösungsansatz: Software Engineering!

software engineering (6)

- Software Engineering (Begriff):
 - Deutsch auch: Softwaretechnik
 - Definition:

„Zielorientierte Bereitstellung und systematische Verwendung von Prinzipien, Methoden und Werkzeugen für die arbeitsteilige, ingenieurmäßige Entwicklung und Anwendung von umfangreichen Softwaresystemen.“

(Balzert 2001, S. 36)
 - Planmäßiges Vorgehen: Vorgehensmodell

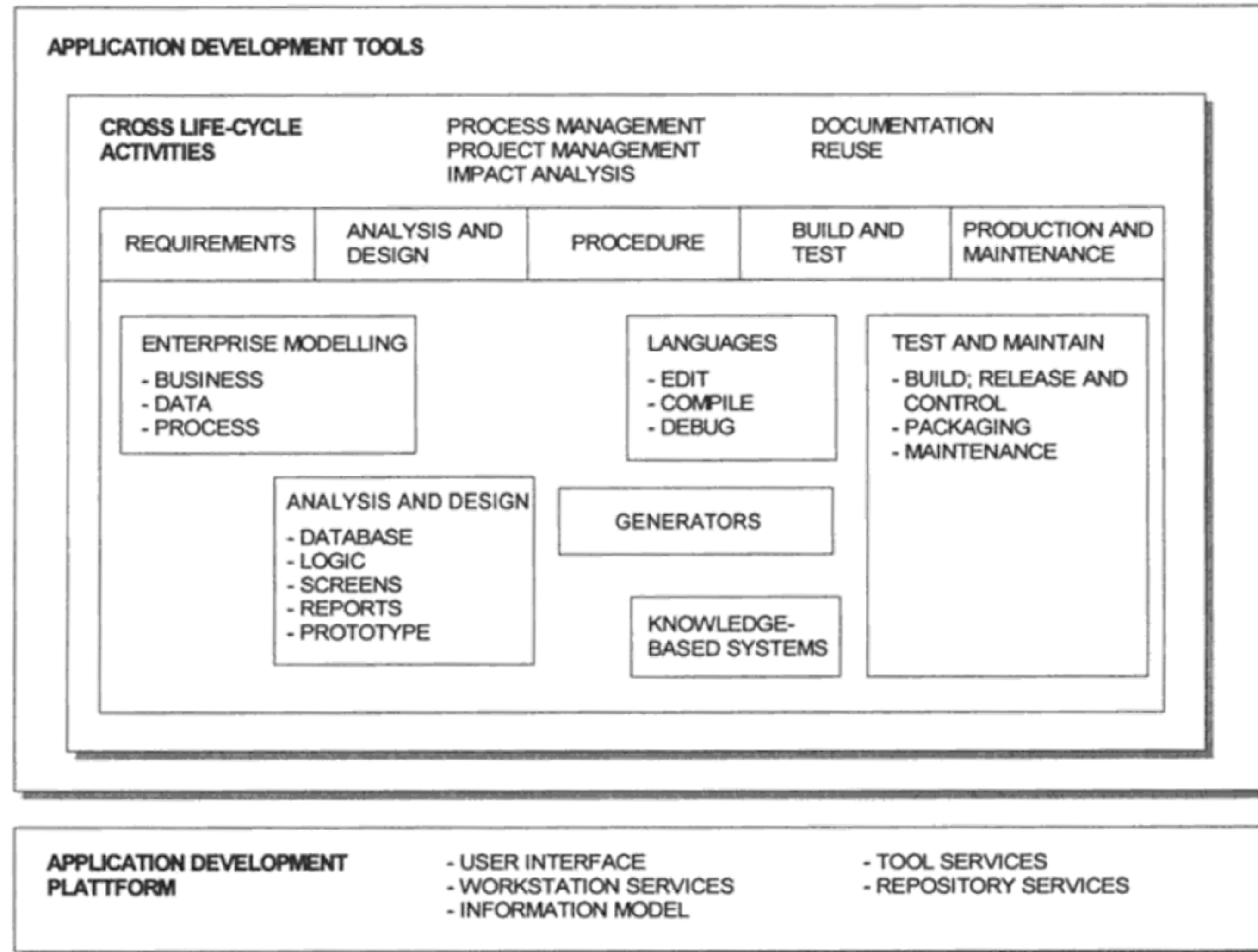
software engineering (7)

- Computer-Aided Software Engineering (CASE):
 - Deutsch: Rechnergestützte Softwareentwicklung
 - Definition:
„Entwicklung von Software durch computergestützte Software-Entwicklungswerkzeuge oder Umgebungen“
(GI 2017)
 - Strukturierte Analyse (SA) (Tom DeMarco)
 - Strukturiertes Design (SD) (Edward Yourdon und Larry Constantine)
 - Modellierungsmethoden

software engineering (8)

IBM AD/Cycle- Rahmenkonzept:

Quelle:
Mercurio/Meyers/Nisbet/Radin
In: IBM Systems
Journal, 1990



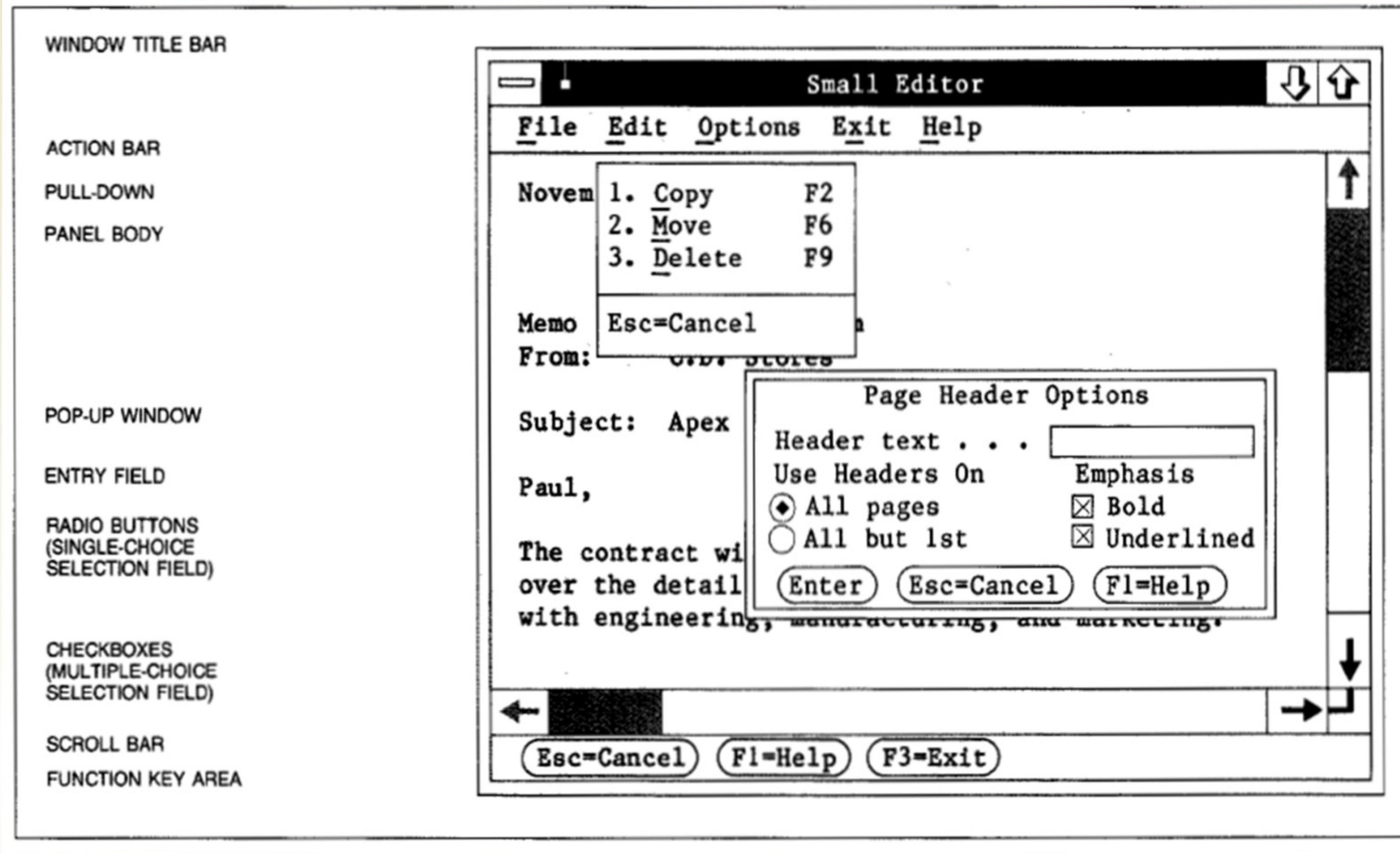
software engineering (9)

- System-Anwendungsarchitektur (SAA)
 - (Ende 1980er Jahre):
 - Standards für system- und plattformübergreifende Hard- und Softwareentwicklung
 - Vorgeschlagen von IBM
 - Regelung der Gestaltung von Benutzeroberflächen im Common User Access (CUA)-Standard
 - CUA-Standard ursprünglich in Zusammenarbeit von Microsoft und IBM für Betriebssystem Operation System 2 (OS/2) erarbeitet
 - CUA-Standard später auch von Microsoft für erste Windows-Betriebssystem-Versionen

software engineering (10)

CUA-Vorgaben
für eine
Benutzerober-
fläche (1988):

Quelle: Berry
In: IBM Systems
Journal, 1988



- DIN EN ISO Norm 9241-11

„Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten Teil 11: Anforderungen an die Gebrauchstauglichkeit — Leitsätze (ISO 9241-11:1998)“:

- Effektivität
- Effizienz
- Zufriedenstellung

software engineering (I2)

- Gebrauchstauglichkeit/Usability:
 - DIN EN ISO Norm 9241-210
„Ergonomie der Mensch-System-Interaktion – Teil 210: Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme (ISO 9241-210:2010)“
 - Definition:
„Ausmaß, in dem ein System, ein Produkt oder eine Dienstleistung durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um festgelegte Ziele effektiv, effizient und zufriedenstellend zu erreichen“

- DIN EN ISO Norm 9241-110

„Ergonomie der Mensch-System-Interaktion – Teil 110: Grundsätze der Dialoggestaltung (ISO 9241-110:2006)“:

- Aufgabenangemessenheit
- Selbstbeschreibungsfähigkeit
- Erwartungskonformität
- Lernförderlichkeit
- Steuerbarkeit
- Fehlertoleranz
- Individualisierbarkeit