

# TOPICS

**The Nature and History of Software  
Development**

**Problems with Software Development**

**Software Engineering Paradigms and  
Technology**

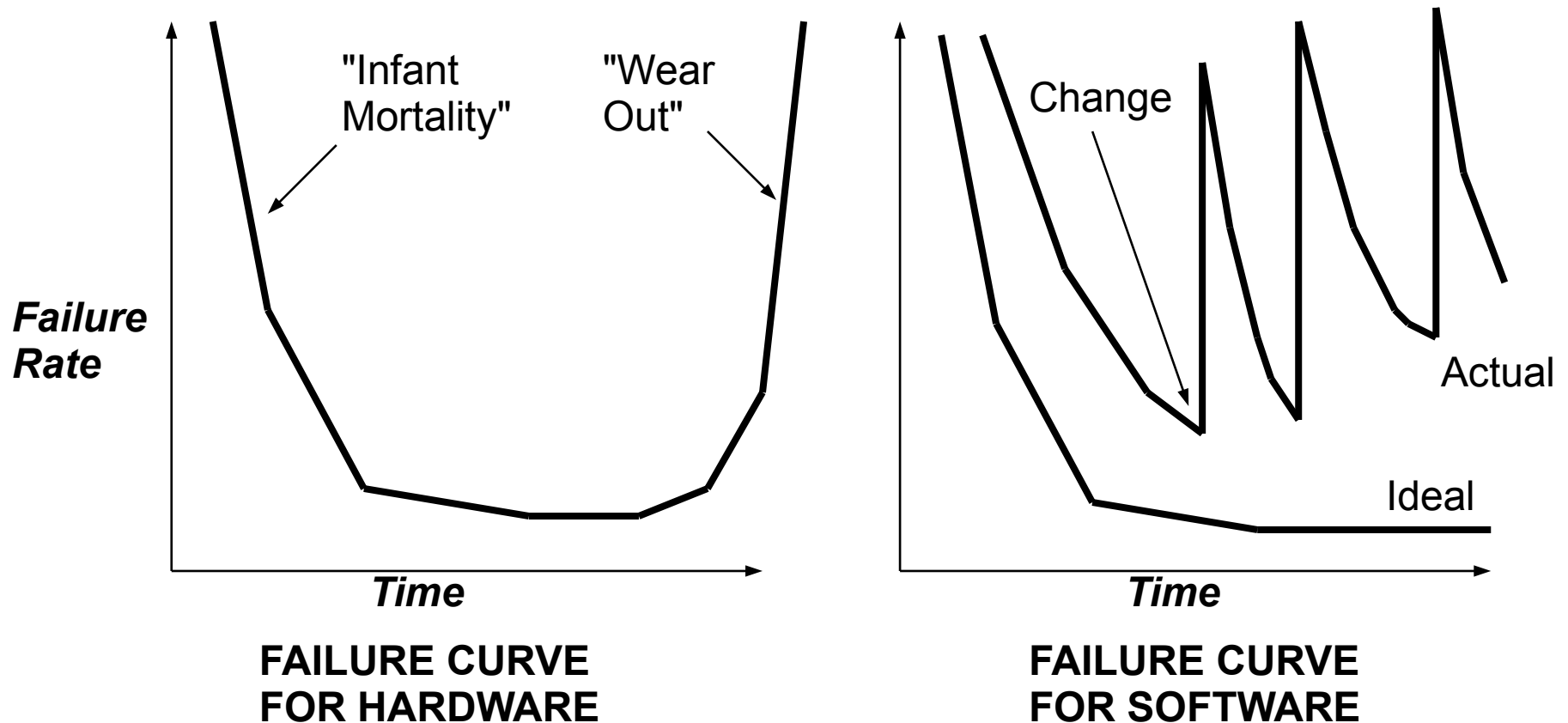
# **THE NATURE OF SOFTWARE**

- Characteristics of Software**
- Failure Curves for Hardware and Software**
- Software Components**
- Software Configuration**
- Software Application Areas**

# Characteristics of Software

- ❑ **Software is *programs, documents, and data.***
- ❑ **Software is developed or engineered; it is not manufactured like hardware.**
- ❑ **Software does not wear out, but it does *deteriorate.***
- ❑ **Most software is custom-built, rather than being assembled from existing components.**
- ❑ **Software is a *business opportunity.***

# Failure Curves for Hardware and Software



# Software Components

- ❑ Software programs, or software systems, consist of *components*.
- ❑ A set of components which comprise a logical unit of software is called a *software configuration item*.
- ❑ Reuse and development of reliable, trusted software components improves software *quality* and *productivity*.
- ❑ Computer language forms:
  - ❑ Machine level (microcode, digital signal generators)
  - ❑ Assembly language (PC assembler, controllers)
  - ❑ High-order languages (FORTRAN, Pascal, C, Ada, ...)
  - ❑ Specialized languages (LISP, OPS5, Prolog, ...)
  - ❑ Fourth generation languages (databases, windows apps)

# Software Configuration

**Software  
Project  
Plan**

**Software  
Requirements  
Specification**

**Software  
Design**

**User  
Documents**

**Software  
Test Plan and  
Procedures**

**Data  
Structures  
and  
Dictionary**

**Code**



# **Software Development Activities**

## **Planning Activity**

**Software Project Plan**

## **Requirements Definition Activity**

**Software Requirements  
Specification**

**Software Test Plan and  
Procedures**

**Data Structures and Dictionary**

**User Documents**

## **Design Activity**

**Software Design Documents**

**Software Test Plan and Procedures**

**Data Structures and Dictionary**

## **Coding and Testing Activity**

**Code**

**Software Test Plan and Procedures**

## **Delivery and Maintenance Activity**

**User Documents**

**Others as needed**

# Software Application Domains

## System

compilers

editors

Operating Systems

## ● Real Time

machine control

auto controls

## Business

databases

stock management

## ● Personal Computer

all non-realtime above

## Embedded

appliance control

FPGA programs

auto controls

## Engineering and Scientific

simulation

computer-aided design

"number crunching"

## Artificial Intelligence

expert systems

neural networks



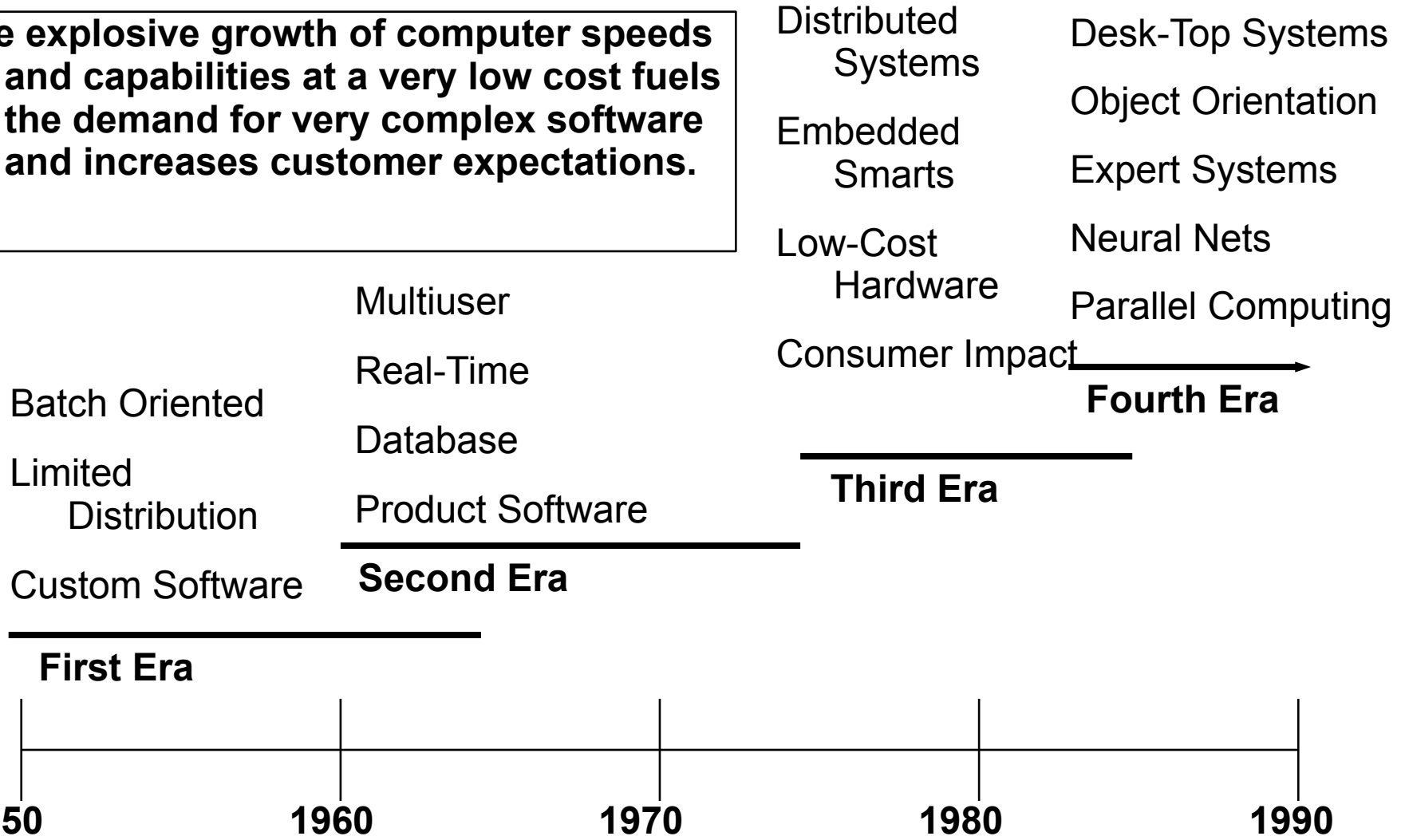
# HISTORY OF SOFTWARE DEVELOPMENT

 **Role of Software**

 **Industrial View**

# Role of Software

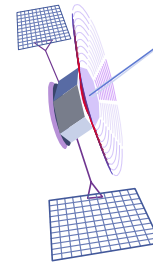
**The explosive growth of computer speeds and capabilities at a very low cost fuels the demand for very complex software and increases customer expectations.**



# Role of Software, Continued

**Where Do We Go From Here?**

- Parallel computing to extend speed of computation**
- Object-oriented methods of software design**
- Software frameworks evolve to handle larger and multiprogram systems**
- Heavy dependence on graphics interfaces**
- Artificial intelligence and neural computing become useful**
- National computing motivates huge software systems**
- Advanced programming languages**



# Industrial View



**Why does it take so long to finish a working software system?**

**Why are development costs so high?**

**Why can't we find all software errors before software is delivered?**

**How can we measure the progress of software development?**

**How can we survive in the global economy?**