

# *Rekayasa Perangkat Lunak (Software Engineering)*

Siti Sa'uda, M.Kom.

*Software Engineering: A Practitioner's Approach, 6/e*

# Bab 1

## Perangkat Lunak dan Rekayasanya

# Peran Ganda Perangkat Lunak (PL)

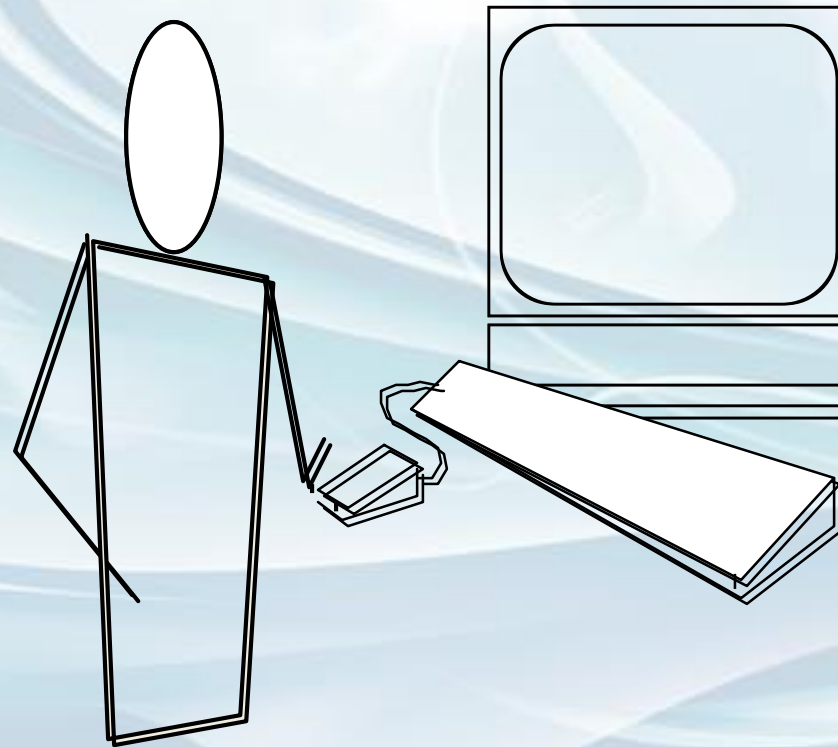
- PL adalah sebuah produk
  - Menyajikan potensi komputasi
  - Menghasilkan, mengelola, memperoleh, modifikasi, menampilkan atau mengirimkan informasi
- PL adalah alat untuk menyajikan sebuah produk
  - Mendukung atau menyediakan langsung fungsi sistem
  - Mengendalikan program lain (mis : sistem operasi)
  - Mempengaruhi komunikasi (mis : PL jaringan)
  - Membantu membangun PL lain (mis : tool PL)



# Apakah PL ?

PL adalah sekelompok item atau objek  
Yang membentuk “konfigurasi” dimana  
di dalamnya termasuk :

- program
- dokumen
- data ...



# Apakah PL ?

## Sebuah perbandingan dengan Hardware

- PL 'dirakit'
- PL tidak kadaluwarsa
- PL lebih kompleks

# Perangkat Lunak

## ❖ Definisi Perangkat Lunak

Ada beberapa definisi perangkat lunak yang pernah dikemukakan antara lain :

- **Software = Perangkat lunak**
  - ☐ Kumpulan program komputer dengan fungsi tertentu
- Perangkat lunak adalah
  1. Instruksi (program komputer) yang bila dieksekusi dapat menjalankan fungsi tertentu,
  2. Struktur data yang dapat membuat program memanipulasi informasi, dan
  3. Dokumen yang menjelaskan operasi dan penggunaan program (Pressman, 1997).
- Perangkat lunak adalah program komputer, prosedur, aturan, dan dokumentasi yang berkaitan serta data, yang bertalian dengan operasi suatu sistem komputer (IEEE, 1993).



## Karakteristik Perangkat Lunak

Perangkat lunak dibangun atau dikembangkan

Merupakan produk yang unik

Perangkat lunak tidak pernah akan rusak/aus karena selalu diperbaharui

Tidak terlihat (*invisible*)

Perangkat lunak pada umumnya dibangun sesuai keinginan

Fleksibel, sehingga mudah dimodifikasi

Dihubungkan (*linked*) dengan sistem komputer

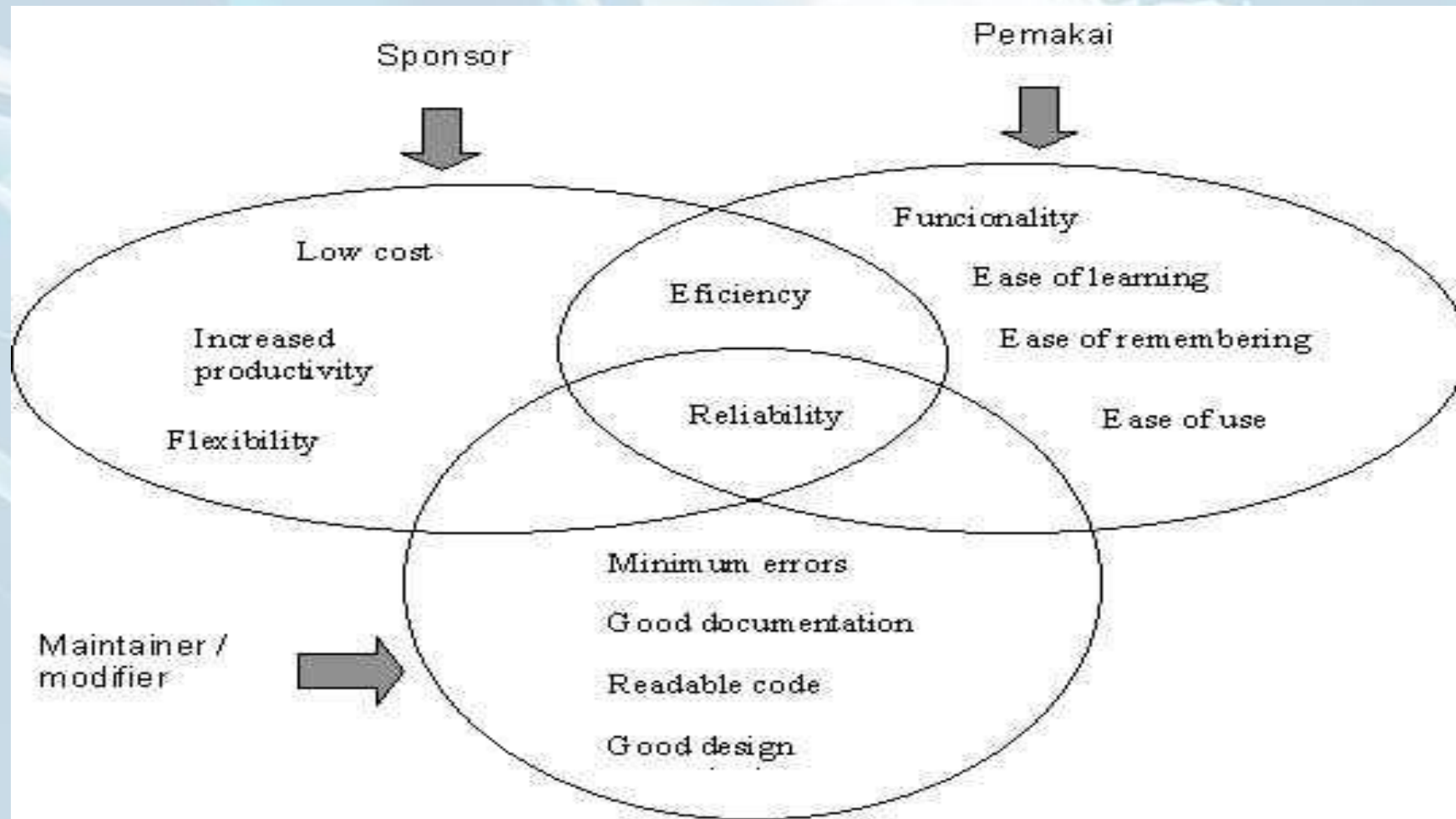
**Karakteristik  
PL**

# Pengertian RPL

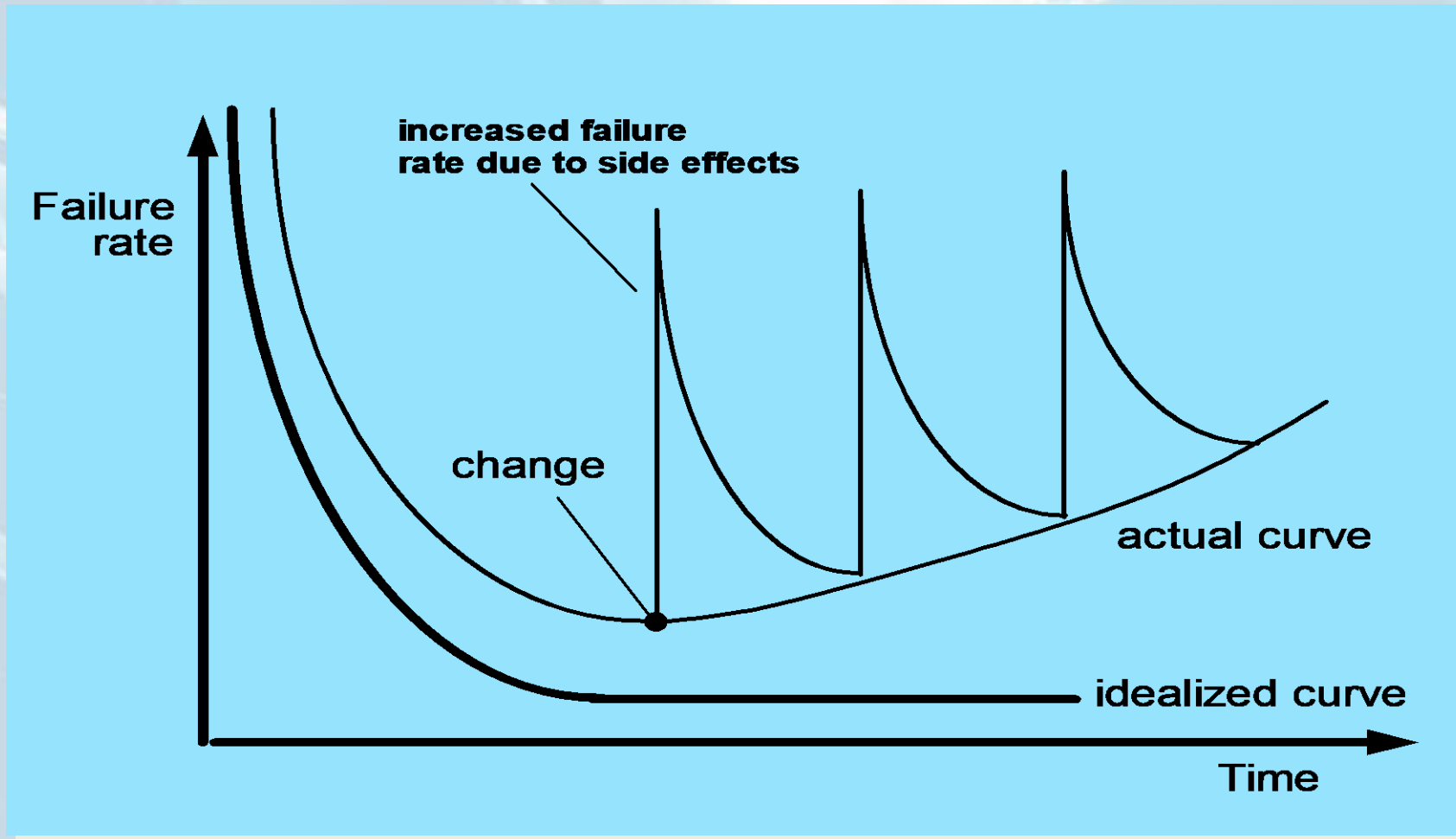
- Adalah suatu proses rancang bangun.
- Beberapa definisi tentang rekayasa perangkat lunak :
  - Pembentukan dan penggunaan prinsip rekayasa (engineering) untuk mendapatkan perangkat lunak secara ekonomis namun andal dan dapat bekerja secara efisien pada komputer (Fritz Bauer, 1968).
  - Penerapan pendekatan yang sistematis, disiplin, dan terukur untuk pengembangan, operasi, dan pemeliharaan perangkat lunak (IEEE, 1993).
  - Suatu disiplin yang mengintegrasikan proses/prosedur, metode, dan perangkat tools untuk pembangunan perangkat lunak komputer (Pressman, 97).
  - Merupakan aplikasi dari prinsip-prinsip sains untuk
    - o Mengurutkan transformasi masalah menjadi solusi yang dapat bekerja dengan baik
    - o Urutan pemeliharaan perangkat lunak tersebut sampai tidak dapat digunakan lagi (Alan M. Davis)
- Proses RPL dimulai jauh sebelum “Coding” dilakukan dan berlanjut terus setelah versi awal dari program selesai dikerjakan.







# Semakin Memburuk...?!





# Aplikasi PL

- system software
- application software
- engineering/scientific software
- embedded software
- product-line software
- WebApps (Web applications)
- AI software

# Kategori Baru PL

- Obiquitous computing — jaringan nirkabel
- Netsourcing—Web sebagai mesin komputasi
- Open source—kode program dibuka gratis kepada komunitas komputasi)
- Juga ...
  - Data mining
  - Grid computing
  - Cognitive machines
  - Software for nanotechnologies

# Perubahan PL

## *Kenapa harus berubah ?*

- PL harus ber-**adaptasi** untuk memenuhi kebutuhan lingkungan komputasi atau teknologi baru.
- PL harus di-**perbaiki** untuk mengimplementasi kebutuhan bisnis baru.
- PL harus **diperluas** untuk membuatnya dapat ber**interoperasi** dengan sistem dan database modern yang lain.
- PL harus **di-arsitek ulang** untuk membuatnya 'hidup' di tengah-tengah lingkungan jaringan.



# Evolusi PL

- **The Law of Continuing Change (1974):** Hukum Perubahan Berkelanjutan. Sistem Elektronik harus secara berkelanjutan beradaptasi jika tidak akan mengurangi tingkat kenyamanan.
- **The Law of Increasing Complexity (1974):** Hukum Peningkatan Kompleksitas : sejalan dengan pertumbuhan PL, kompleksitasnya juga akan meningkat, kecuali ada sesuatu yang dilakukan untuk mengelola atau mengurangnya.
- **The Law of Self Regulation (1974):** Hukum Regulasi Mandiri, Proses evolusi sistem elektronik adalah regulasi mandiri dengan distribusi pengukuran produk dan proses yang dekat dengan normal.
- **The Law of Conservation of Organizational Stability (1980):** Hukum Konservasi Stabilitas Organisasi. Rata-rata kecepatan aktivitas global efektif dalam sistem elektronik tidak jauh berbeda dengan kecepatan produksi.
- **The Law of Conservation of Familiarity (1980):** Hukum Konservasi Kebiasaan Sistem elektronik turut mengembangkan segala sesuatu yang berkaitan dengannya, developer, sales, user, sebagai contoh, harus tetap menguasai isi dan perilaku untuk dapat menggapai evolusi yang nyaman.
- **The Law of Continuing Growth (1980):** Hukum Keberlanjutan Pertumbuhan. Muatan fungsional sistem elektronik harus secara terus menerus meningkat untuk merawat kenyamanan pengguna.
- **The Law of Declining Quality (1996):** Hukum Penurunan Kualitas, Kualitas sistem elektronik akan selalu menurun kecuali ada usaha perawatan dan beradaptasi pada perubahan lingkungan operasional.
- **The Feedback System Law (1996):** Evolusi proses tipe elektronik membentuk sistem umpan balik multi-level, multi-loop, multi-agen, dan harus diperlakukan sebagaimana usaha-usaha serius lain untuk membuat peningkatan yang signifikan.

# Mitos PL

- Mempengaruhi manajer, pelanggan (dan stakeholder non teknis lainnya) serta para praktisi
- Dipercaya karena mereka sering mempunyai bagian-bagian kebenaran.,

*namun...*

- Sering menuju ke keputusan yang salah,

*Oleh karena itu ...*

- Tetaplah pada realita seiring dengan jalan rekayasa perangkat lunak