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

Perangkat Lunak dan Rekayasanya

Oleh :

RUSTAM EFFENDI ,S. Kom., M.Kom.

Pustaka Wajib

1. Pressman, R. S., Software Engineering: A Practitioner's Approach, 8th Edition, McGraw-Hill, 2008
2. Sommerville, I., Software Engineering 8th Edition, Addison-Wesley, 2007.
3. Stephen R. Schach: Object-Oriented and Classical Software Engineering, 7th Edition, Prentice Hall, 2001

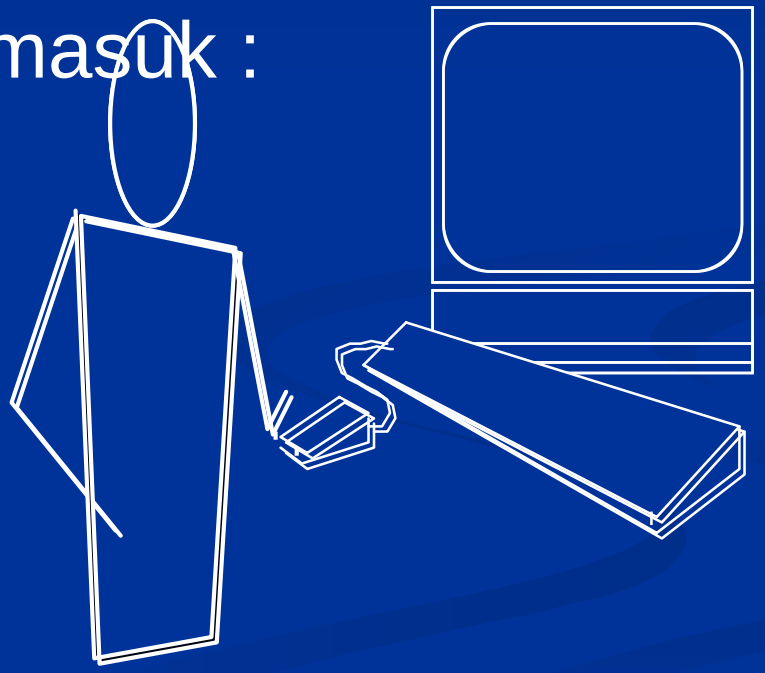
Peran Ganda Perangkat Lunak (PL)

- 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 RPL ?

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

- program
- dokumen
- data ...



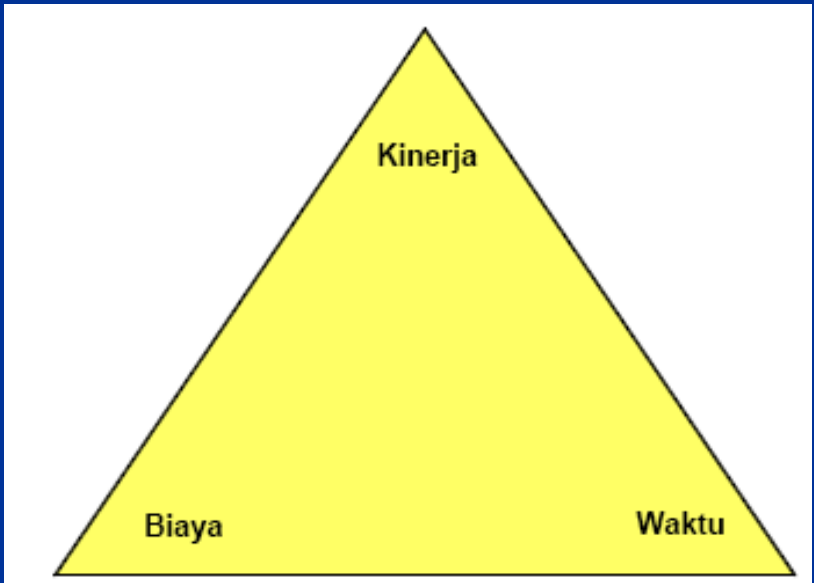
Pengertian secara Umum

- Rekayasa Perangkat Lunak suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal yaitu analisis kebutuhan pengguna, menentukan spesifikasi dari kebutuhan pengguna, desain, pengkodean, Implementasi, pengujian sampai pemeliharaan sistem setelah digunakan.

Rekayasa Perangkat Lunak ≠ Program

- **Rekayasa Perangkat lunak** adalah seluruh perintah yang digunakan untuk memproses informasi.
- **Rekayasa** Perangkat lunak dapat berupa program atau prosedur.
- **Program** adalah kumpulan perintah yang dimengerti oleh komputer sedangkan
- **prosedur** adalah perintah yang dibutuhkan oleh pengguna dalam memproses informasi (O'Brien, 1999).

TUJUAN RPL



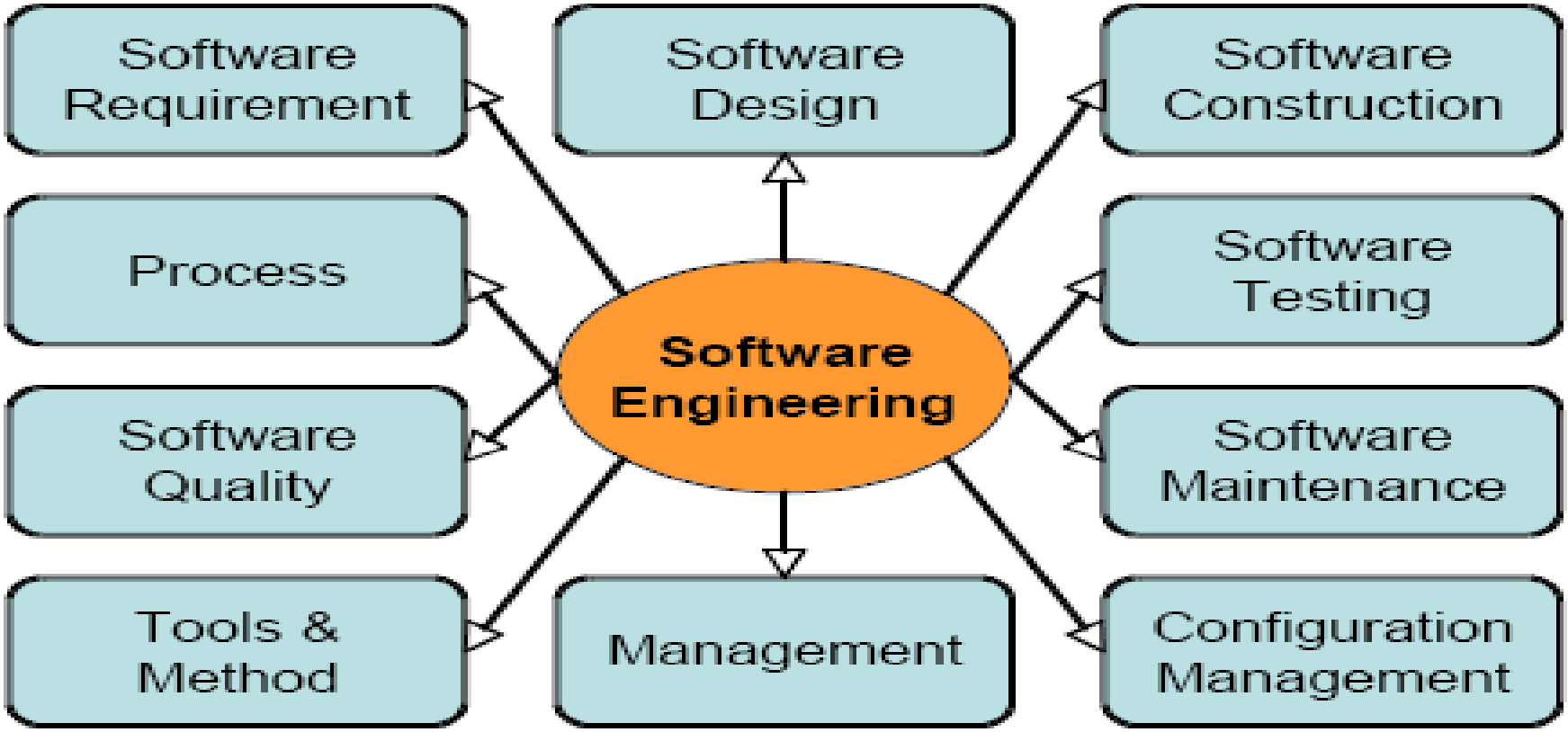
Gambar 1.2. Tujuan RPL.

1. Memperoleh biaya produksi perangkat lunak yang rendah.
2. Menghasilkan perangkat lunak yang kinerjanya tinggi, andal dan tepat waktu.
3. Menghasilkan perangkat lunak yang dapat bekerja pada berbagai jenis platform.
4. Menghasilkan perangkat lunak yang biaya perawatannya rendah.

Apakah PL ?
Sebuah perbandingan dengan Hardware

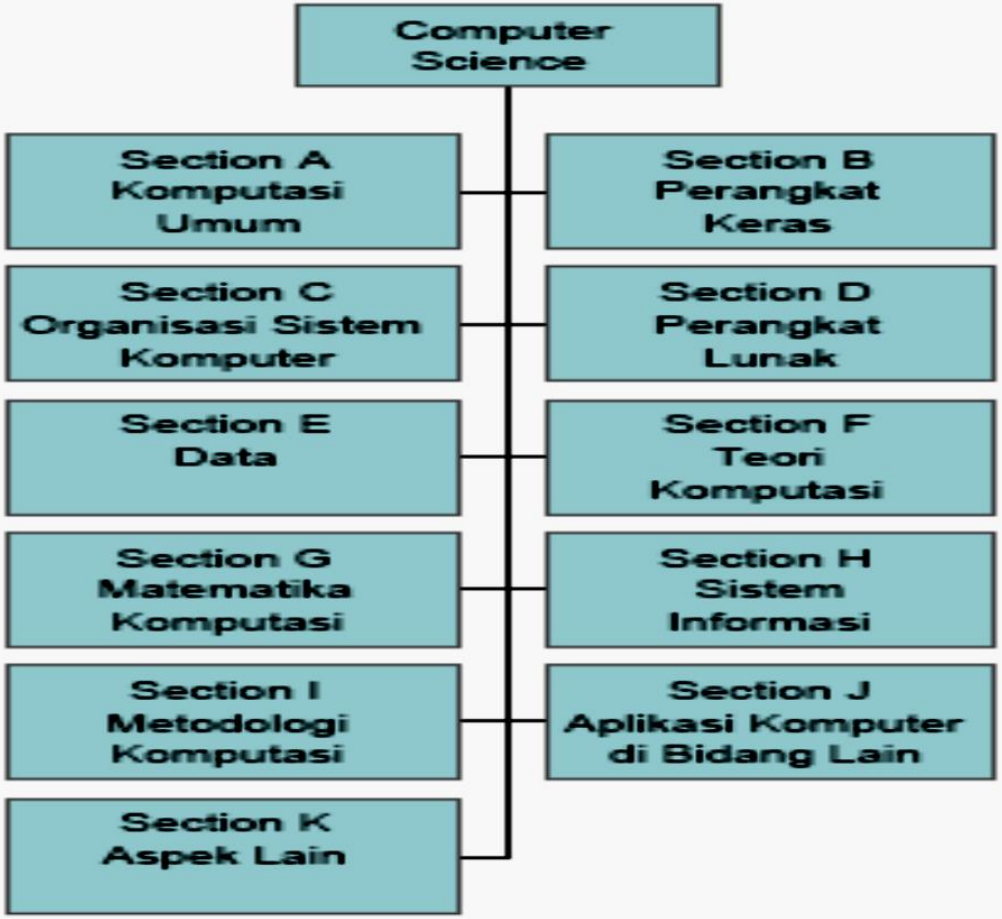
- PL ‘dirakit’
- PL tidak kadaluwarsa/ter update
- PL lebih kompleks

RUANG LINGKUP

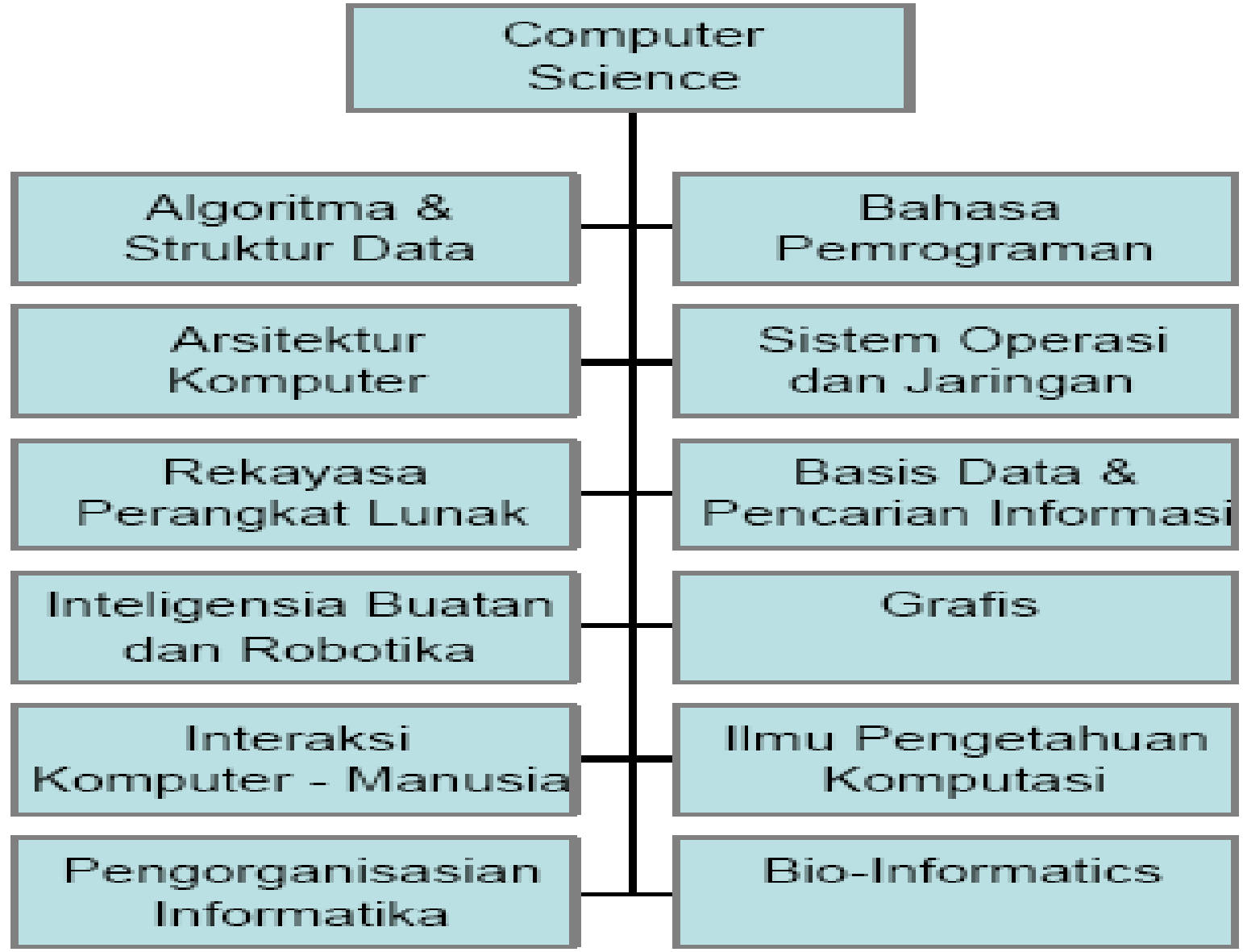


Gambar 1.3. Ruang lingkup RPL (Abran et.al., 2004).

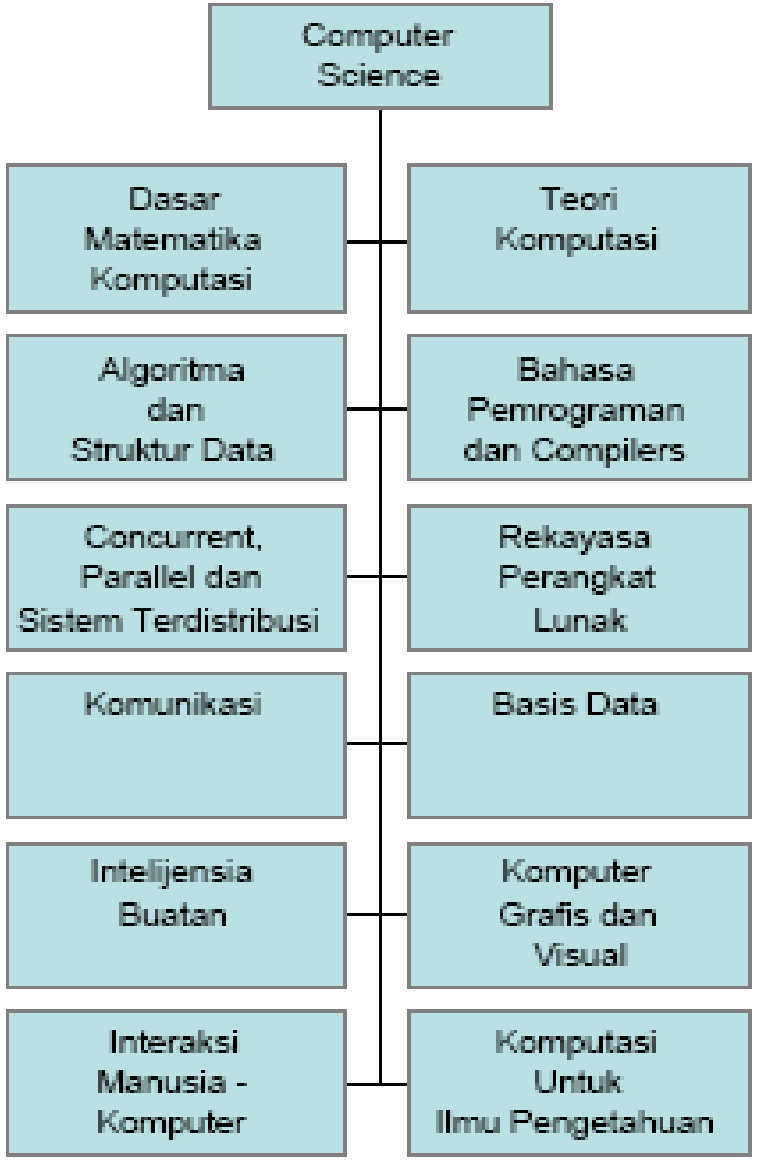
RPL Bagian dari Disiplin Ilmu Komputer



Gambar 1.4. Klasifikasi disiplin ilmu komputer menurut ACM (1998).

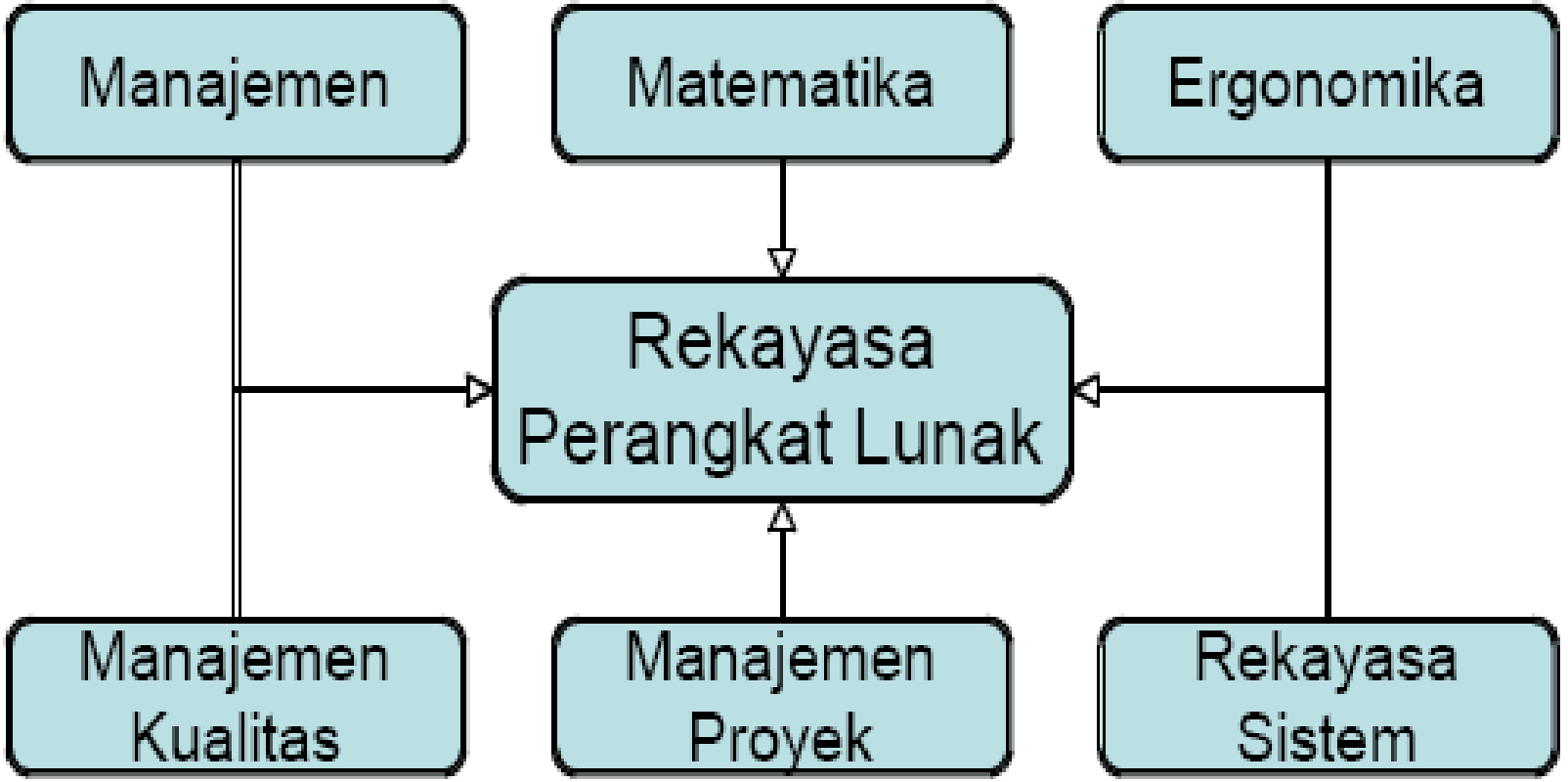


Gambar 1.5. Klasifikasi disiplin ilmu komputer menurut Denning (2000).



Gambar 1.6. Klasifikasi disiplin ilmu komputer menurut Wikipedia (2007).

POSISI RPL



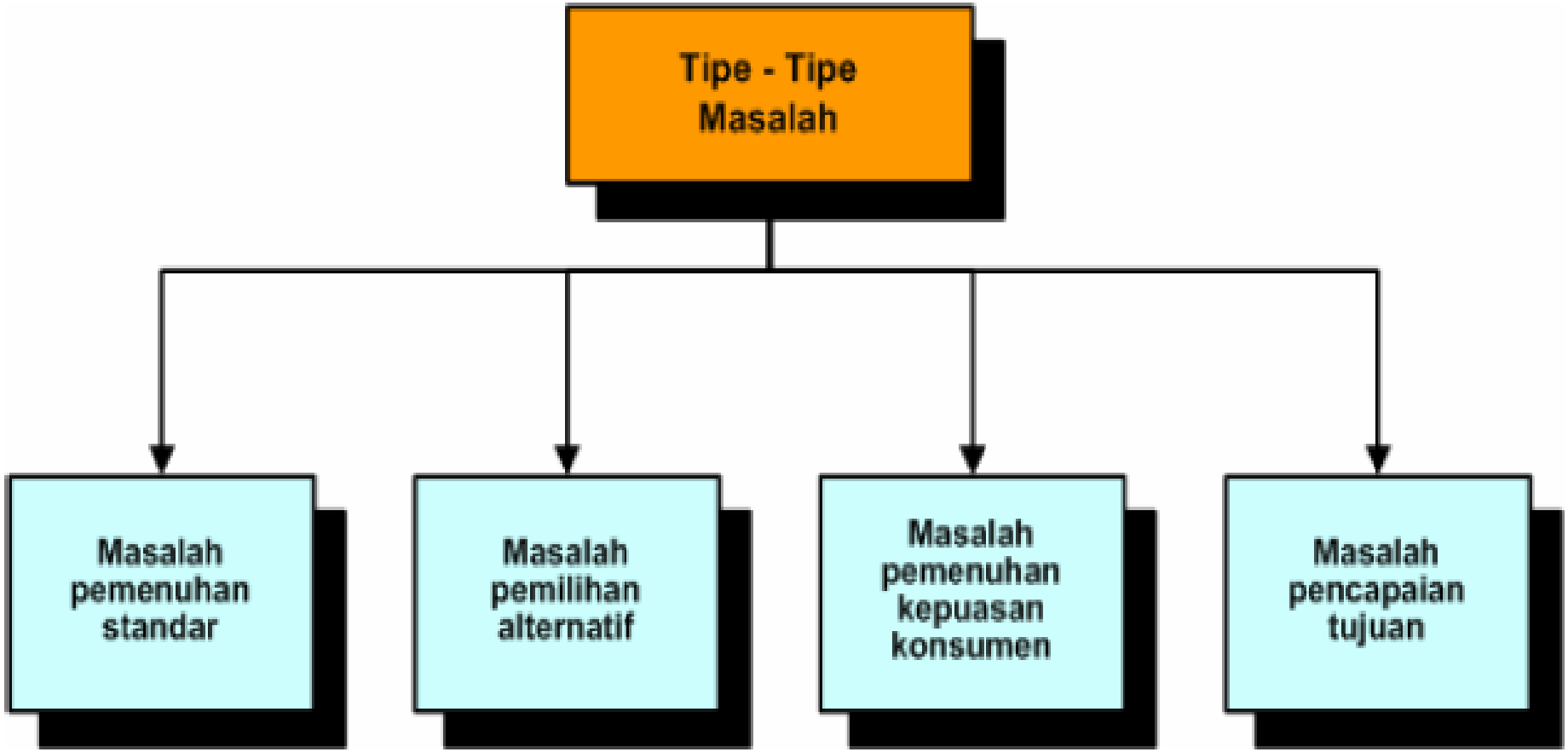
Gambar 1.7. Keterkaitan RPL dengan bidang ilmu lain.

SEJARAH RPL

1940an	Komputer pertama yang membolehkan pengguna menulis kode program langsung
1950an	Generasi awal interpreter dan bahasa macro Generasi pertama compiler
1960an	Generasi kedua compiler Komputer mainframe mulai dikomersialkan Pengembangan perangkat lunak pesanan Konsep Software Engineering mulai digunakan
1970an	Perangkat pengembang perangkat lunak Perangkat minicomputer komersial
1980an	Perangkat Komputer Personal (PC) komersial Peningkatan permintaan perangkat lunak
1990an	Pemrograman berorientasi obyek (OOP) Agile Process dan Extreme Programming Peningkatan drastis kapasitas memori Peningkatan penggunaan internet
2000an	Platform interpreter modern (Java, .Net, PHP, dll) Outsourcing

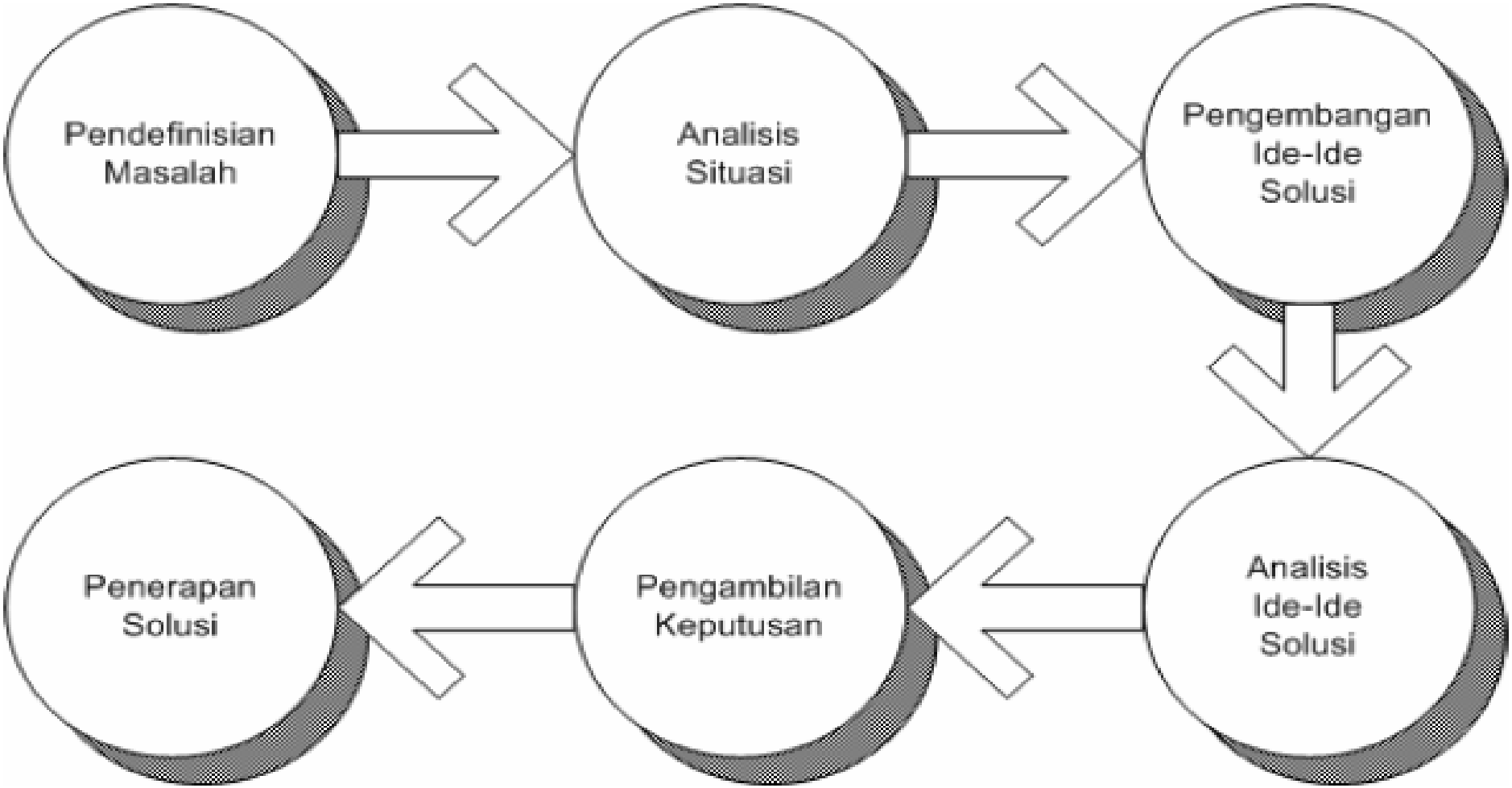
Gambar 1.8. Perkembangan RPL.

TIPE MASALAH



Gambar 1.11. Tipe-tipe masalah (Deek et al, 2005).

SIKLUS SOLUSI RPL



Gambar 1.12. Proses pemecahan masalah (diadopsi dari Deek et al, 2005)

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 berinteroperasi dengan sistem dan database modern yang lain.
- PL harus di-arsitek ulang untuk membuatnya ‘hidup’ di tengah-tengah lingkungan jaringan.

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