

## BAB VI

### SISTEM PAKAR (*EXPERT SYSTEM*)

*Expert System* yaitu program-program yang bertingkah laku seperti manusia pakar/ahli(*human expert*).

Sistem pakar atau sistem berbasis pengetahuan adalah yang paling banyak aplikasinya dalam membantu menyelesaikan masalah-masalah dalam dunia nyata.

Contoh aplikasi dari program ini antara lain yaitu :

- Delta dari *General Electric* untuk konsultasi kerusakan lokomotif.
- Prospector dari *Stanford Research Institute* untuk penaksiran prospek mineral.
- Xycon dari *Digital Equipment Corp's* untuk mengkonfigurasi bagian-bagian komputer.
- Mycin dikembangkan pada Universitas Stanford (1970), untuk menolong para ahli dalam mendiagnosa bakteri penyakit tertentu.

Program kecerdasan tiruan ini dapat dilakukan dengan menggunakan suatu program paket, yaitu alat pengembangan sistem aplikasi pengetahuan (*knowledge system application development tool*) seperti:

- VP-Expert
- PC PLUS
- GURU
- JESS(*Java Expert System Shell*) Version 5.0  
<http://herzberg.ca.sandia.gov/jess/FAO.html>
- EXSYS, dan lain-lain.

Atau dengan menggunakan bahasa untuk pemrograman kecerdasan tiruan seperti :

- PROLOG (*Programming Logic*)
- WIN-PROLOG 4.040 (*Windows-Programming Logic*)  
<http://www.lpa.co.uk/web386/8f922bf1.zip>

- LISP(*Lisp Processing*)
- CLIPS(*C Language Integrated Production System*)

<http://www.ghgcorp.com/clips/download/source/>

Perangkat-perangkat lunak ini dapat dijalankan dengan komputer pribadi (PC), sehingga pengembangan untuk aplikasi kecerdasan tiruan dapat dilakukan dengan mudah dan dengan biaya yang murah.

Sistem pakar adalah program komputer yang :

- Menangani masalah dunia nyata, masalah yang kompleks yang membutuhkan interpretasi pakar.
- Menyelesaikan masalah dengan menggunakan komputer dengan model penalaran manusia dan mencapai kesimpulan yang sama dengan yang dicapai oleh seorang pakar jika berhadapan dengan masalah.

Komputer berbasis sistem pakar adalah program komputer yang mempunyai pengetahuan yang berasal dari manusia yang berpengetahuan luas(pakar) dalam domain tertentu, di mana pengetahuan di sini adalah pengetahuan manusia yang sangat minim penyebarannya, mahal serta susah didapat. Di sini keahlian dari manusia dimasukkan ke pengetahuan tersebut untuk menyelesaikan masalah, seperti yang dilakukan manusia.

Walaupun sistem pakar dapat menyelesaikan masalah dalam domain yang terbatas berdasarkan pengetahuan yang dimasukkan ke dalamnya, tetapi sistem pakar tidak dapat menyelesaikan masalah yang tidak dapat diselesaikan manusia. Oleh sebab itu keandalan dari sistem pakar terletak pada pengetahuan yang dimasukkan ke dalamnya.

Kondisi-kondisi dimana sistem pakar dapat membantu manusia dalam menyelesaikan masalahnya, antara lain:

- Kebutuhan akan tenaga ahli(pakar) yang banyak, tetapi pakar yang tersedia jumlahnya sangat terbatas.
- Pemakaian pakar yang berlebihan dalam membuat keputusan, walaupun dalam suatu tugas yang rutin.

- ❑ Pertimbangan kritis harus dilakukan dalam waktu yang singkat untuk menghindari hal-hal yang tidak diinginkan.
- ❑ Hasil yang optimal, seperti dalam perencanaan atau konfigurasi.
- ❑ Sejumlah besar data yang harus diteliti oleh pakar secara kontinu.

Dengan menggunakan sistem pakar dalam membantu memecahkan masalah, didapat beberapa keuntungan:

- Sifatnya yang permanen.
- Mudah untuk ditransfer atau direproduksi.
- Mudah didokumentasikan.
- Menghasilkan keluaran yang konsisten.
- Biaya yang murah.
- Dapat digunakan untuk 24 jam sehari
- Dapat dibentuk semenjak ada keterbatasan dari manusia pakar
- Sulit mendapatkan seorang yang *expert*/pakar sehingga sistem pakar dapat menggantikan tugas tersebut.
- Pengetahuan pada sistem pakar mudah disimpan dan dicopy
- Pengetahuan yang ada tidak mudah hilang
- Selalu membentuk opini terbaik dalam batas pengetahuannya.

Kerugian *Expert System*:

- Kurang personalitinya
- Tidak dapat menyelesaikan masalah yang membutuhkan intuisi

*Expert System* terdiri dari 2 bagian yang harus dimilikinya:

1. *Knowledge Base*
2. *Inference Engine*

*Knowledge Base*

Sebuah database yang menyimpan informasi pengetahuan tertentu dan aturan-aturan tentang subyek tertentu.

*Knowledge base* terdiri 2 bentuk:

1. Obyek: kesimpulan yang didefinisikan oleh kelompok aturan.
2. Atribut: kualitas tertentu di mana bersama-sama dengan aturan membantu mendefinisikan obyek.

Jadi *knowledge base* dapat diartikan : daftar dari obyek-obyek dengan kelompok-kelompok aturan dan atribut.

### *Inference Engine*

*Inference Engine* adalah bagian dari sistem pakar yang mencoba menggunakan informasi yang diberikan untuk menemukan obyek yang sesuai.

Kategori *inference engine*:

1. *deterministic*
2. *probabilistik*

Dasar untuk membentuk *inference engine*:

1. *Forward Channing*
2. *Backward Channing*
3. *Rule Value*

### Metoda *Forward-Channing*

Kadang disebut: *data-driven* karena *inference engine* menggunakan informasi yang ditentukan oleh user untuk memindahkan keseluruhan jaringan dari logika 'AND' dan 'OR' sampai sebuah terminal ditentukan sebagai obyek.

Bila *inference engine* tidak dapat menentukan obyek maka akan meminta informasi lain. Aturan(*Rule*) di mana menentukan obyek, membentuk Path(lintasan) yang mengarah ke Obyek. Oleh karena itu, hanya satu cara untuk mencapai obyek adalah memenuhi semua aturan.

### Metoda *Backward Chaining*

Merupakan kebalikan dari *forward chaining* di mana mulai dengan sebuah hipotesa(sebuah obyek) dan meminta informasi untuk menyakinkan atau mengabaikan *backward chaining inference engine* sering disebut: '*Object – Driven/Goal-Driven*'

Karena sistem mulai dengan obyek dan mencoba untuk memverifikasi obyek.

#### Metoda *Rule-Value*

Merupakan pendahulu dari *forward* atau *backward channing inference engine system*, karena metoda ini membutuhkan informasi yang mempunyai kepentingan terbesar. Metoda ini umumnya membuktikan mekanisme *backward channing*.

Kesulitan dari sistem '*Rule Value*':

1. Dalam situasi yang nyata, *knowledge base* sering menjadi besar jumlah kemungkinan kombinasi, sehingga tidak dapat menentukan informasi yang sesuai dan menghapus ketidakpastian pada keadaan yang ada.
2. Sistem ini membutuhkan *knowledge base* yang berisi tidak hanya standard informasi atribut obyek melainkan juga nilai penentu yang membuat bentuk *knowledge base* lebih sulit.

#### Pembentukan *Expert System(ES)* Tujuan Umum

Metoda yang dipakai: *Backward channing* karena sistem tersebut memperbolehkan mekanisme inferensi menggunakan berbagai *knowledge base*.

Mekanisme *inference*: campuran antara obyek dan atribut.

Spesifikasi yang harus ada dalam mekanisme inferensi:

1. Sistem pakar tidak boleh membutuhkan atribut yang sama lebih dari satu kali.
2. Sistem pakar segera menolak dan memindahkan untuk melewati suatu obyek yang tidak membutuhkan atribut yang perlu diketahui.
3. Sistem pakar harus dapat melaporkan mengapa perintah mengikuti alur alasan.

Sistem pakar adalah program AI dengan basis pengetahuan (*knowledge base*) yang diperoleh dari pengalaman/pengetahuan pakar/ahli dalam memecahkan persoalan pada bidang tertentu dan didukung mesin inferensi/*inference engine* yang melakukan penalaran/pelacakan terhadap sesuatu/fakta-fakta yang diberikan oleh *user*/pemakai, dicocokkan/*matching* dengan fakta-fakta dan aturan/kaidah yang ada di basis pengetahuan setelah dilakukan pencarian, sehingga dicapai kesimpulan.

Sistem pakar memecahkan persoalan yang secara normal dipecahkan dengan keahlian/kepakaran manusia

Sistem pakar adalah perangkat lunak komputer yang memiliki basis pengetahuan (*knowledge base*) untuk domain tertentu & juga memerlukan untuk mengeksplotasikan satu atau lebih mekanisme penalaran atau pemikiran / pertimbangan (*inference*) yang menyerupai seorang pakar dalam memecahkan masalah.

Kemampuan ES: memecahkan masalah-masalah praktis pada saat sang pakar berhalangan.

Basis pengetahuan (*knowledge base*) merupakan bukan pengetahuan formal (**bukan *text book***), tetapi harus berupa pengalaman bekerja seorang pakar pada sebuah bidang tertentu disiplin ilmu tertentu.

ES : ***Learning by doing for knowledge*** (pengetahuan yang berasal dari belajar dari pengalaman)

### 6.1. Ciri-ciri Sistem Pakar

Ciri-ciri sistem pakar :

1. Terbatas pada domain keahlian tertentu
2. Dapat memberikan penalaran untuk data yang tidak pasti.
3. Dapat mengemukakan rangkaian alasan-alasan yang diberikannya dengan cara yang dapat dipahami.
4. Berdasarkan pada kaidah/ ketentuan/ *rule* tertentu.
5. Dirancang untuk dapat dikembangkan secara bertahap.

6. Pengetahuan & mekanisme penalaran (*inference*) jelas terpisah.
7. Keluarannya bersifat anjuran.
8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai dituntun oleh dialog dengan *user*.

**Keuntungan sistem pakar :**

1. Membuat seorang yang awam bekerja secara seperti layaknya seorang pakar.
2. Meningkatkan produktifitas akibat meningkatnya kualitas hasil pekerjaan, disebabkan meningkatnya efesiensi kerja.
3. Menghemat waktu kerja.
4. Menyederhanakan pekerjaan.
5. Merupakan arsip yang terpercaya dari sebuah keahlian.
6. Memperluas jangkauan, dari keahlian secara pakar.

Dimana ES yang telah disahkan akan sama saja artinya :

- seorang pakar yang tersedia dalam jumlah besar.
- dapat diperoleh & dipakai dimana saja

**6.2. Tipe-tipe sistem pakar**

Tipe-tipe sistem pakar berdasarkan struktur program, ada 3(tiga) tipe :

**A. Program Mandiri**

Sistem pakar yang murni dan berdiri sendiri, artinya program utamanya tanpa mengandung *subroutine* yang memakai algoritma konvensional.

**B. Program Terkait**

Sistem pakar yang dikelilingi program lainnya, artinya sebuah *subroutine* yang akan dipanggil oleh program utama.

Misalnya memiliki *subroutine* untuk:

- perhitungan matematik
  - pembuatan grafik
  - keperluan lainnya
- 
- algoritma konvensional

### C. Program Terhubungkan

Sistem pakar merupakan program yang dapat berhubungan dengan paket program lainnya, misalkan:

- spreadsheet (lotus, excel, quatro pro, dan sebagainya)
- *Database Management System* (dBase III+, foxbase, dan sebagainya)
- atau pembuat Grafik.

## 6.3. Komponen Sistem Pakar

Sebuah program sistem pakar terdiri atas komponen-komponen sebagai berikut :

### a. Basis Pengetahuan (*knowledge base*)

- inti program sistem pakar
- merupakan representasi pengetahuan (*knowledge representation*) dari seorang pakar.
- Tersusun atas fakta yang berupa objek dan kaidah/ ketentuan (*rule*) yang merupakan informasi tentang cara bagaimana membangkitkan fakta baru dari fakta yang sudah diketahui.

*Facts list* (daftar fakta-fakta) berisikan hasil observasi dan sesuatu kenyataan yang dibutuhkan selama pengolahan

Bagian yang mengandung semua fakta-fakta, baik fakta awal pada saat sistem mulai beroperasi maupun fakta-fakta yang didapatkan pada saat pengambilan kesimpulan.

### b. Mesin Inferensi (*Inference Engine*)

- Bagian yang mengandung mekanisme fungsi berpikir dan pola-pola penalaran sistem yang digunakan seorang pakar.
- Mekanisme ini akan menganalisa sesuatu masalah tertentu dan selanjutnya mencari jawaban / kesimpulan yang terbaik.
- Memilih pengetahuan yang relevan dalam rangka mencapai kesimpulan.



- Memulai pelacakannya dengan mencocokkan kaidah (*rule*) dalam basis pengetahuan dengan fakta-fakta yang ada dalam *facts list* disimpan dalam Basis Pengetahuan di harddisk.

Ada dua teknik penalaran (*inference*):

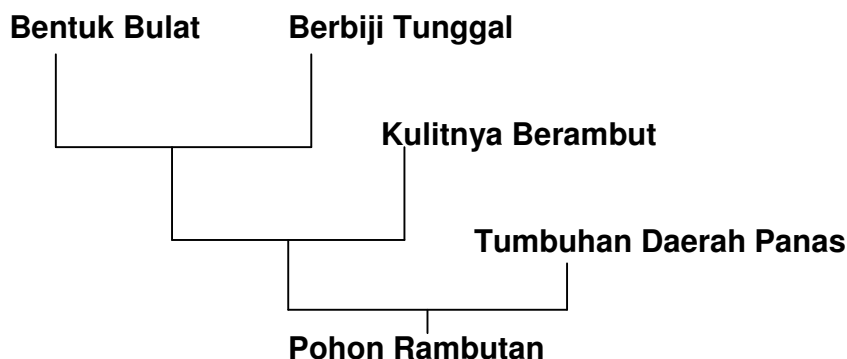
- Pelacakan ke belakang (*backward channing*) yang memulai penalarannya dari kesimpulan(*goal*), dengan mencari sekumpulan hipotesa-hipotesa yang mendukung menuju fakta-fakta yang mendukung sekumpulan hipotesa-hipotesa tersebut.
- Pelacakan ke depan (*forward channing*) memulai dari sekumpulan fakta-fakta (data) dengan mencari kaidah yang cocok dengan dugaan/hipotesa yang ada menuju kesimpulan.

Kedua teknik penalaran dipengaruhi oleh tiga macam teknik penelusuran (*searching*):

- *Depth-First Search*
- *Breadth-First Search*
- *Best-First Search*

1) *Forward Channing*

Contoh 1:



Contoh 2: F.C. menggunakan implikasi

Implikasi 1: p1: a college professor teaches in the summer

q1 : the professor can't do anything but teach

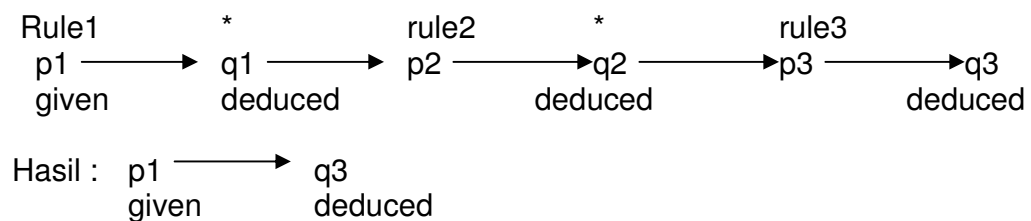
Implikasi 2: p2: the professor can't do anything but teach (= q1)

q2 : the professor does not have time to do research

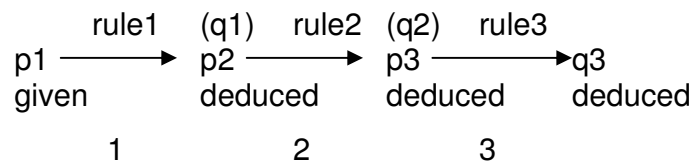
Implikasi 3: p3: the professor does not have time to do research (= q2)

q3: the professor is unhappy

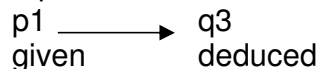
Basis pengetahuan (*knowledge base*) dapat dituliskan dengan menggunakan variabel substitusi :



Tanda \* adalah *symbolic matching approach*. Atau dapat digambarkan sebagai berikut:



Gambar di atas memperlihatkan bentuk dari *forward channing* yang akan menghasilkan :

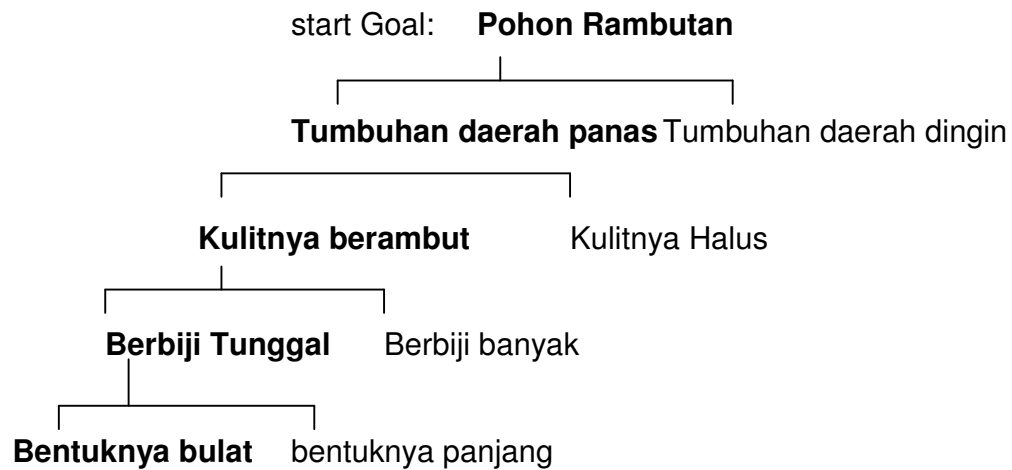


Basis aturannya(*rule base*):

Step1: ((p1 → p2) p1 = True) → (p2 = True)

Step1: ((p2 → p3) p2 = True) → (p3 = True)

Step1: ((p3 → q3) p3 = True) → (q3 = True)

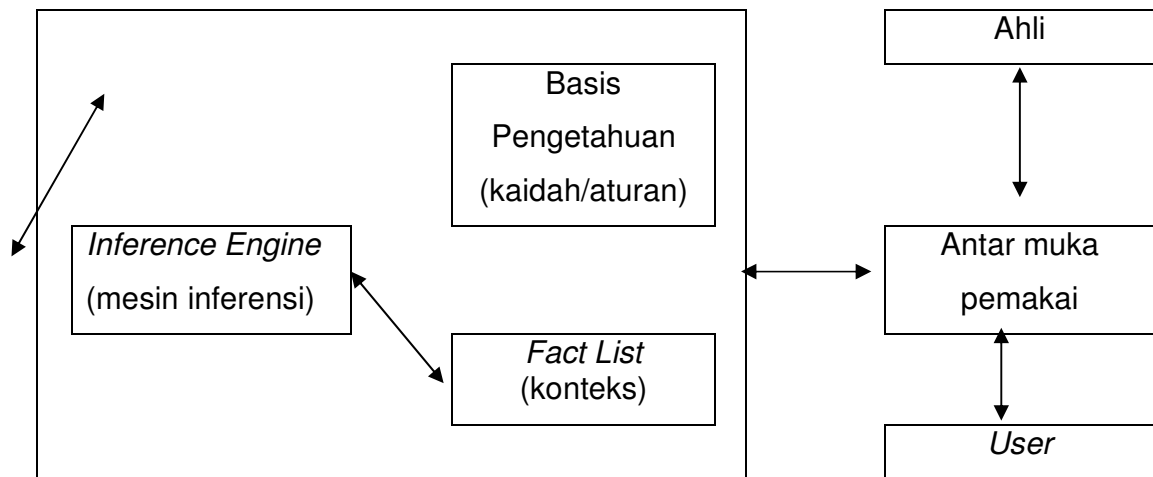
2) *Backward Channing*c. **Antar Muka Pemakai(*User Interface*)**

- Bagian penghubung antara sistem pakar dengan pemakai
- Akan terjadi dialog antara program dan pemakai
- Program akan mengajukan pertanyaan-pertanyaan dan jawaban berbentuk “ya”/”tidak”, berbentuk panduan menu(*menu driven*), pernyataan-pernyataan bahasa alami(*natural language*), dan *graphics Interface style*. Program sistem pakar akan mengambil kesimpulan berdasarkan jawaban-jawaban dari pemakai tadi.

d. **Development Engine**

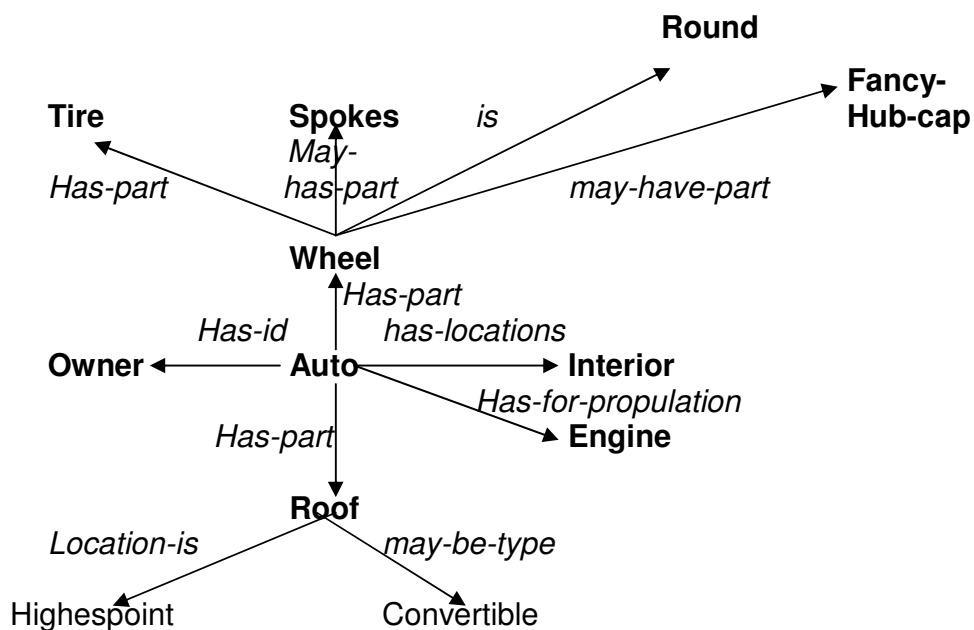
Bagian dari sistem pakar sebagai fasilitas untuk mengembangkan mesin inferensi dan penambahan basis pengetahuan yang akan dilakukan oleh *knowledge engineer*(harus punya keahlian dalam mengerti bagaimana pakar menerapkan pengetahuan mereka dalam memecahkan masalah, mampu mengekstrasi penjelasan(*knowledge acquisition*) mengenai pengetahuan dari pakar), bila si pakar menemukan pengetahuan dan aturan yang baru dari pengalaman ia bekerja.

Struktur sistem pakar dapat dirinci sebagai berikut:



Contoh Representasi pengetahuan dari *Automobile*:

Gambar Jaringan semantik dari auto:



Representasi fact list dari auto:

|      |                     |          |
|------|---------------------|----------|
| Auto | has-part            | wheels   |
|      | has-id              | owner    |
|      | has-locations       | interior |
|      | has-part            | root     |
|      | has-for-propulation | engine   |

|       |               |               |
|-------|---------------|---------------|
| wheel | has-part      | tire          |
|       | may-has-part  | spokes        |
|       | may-have-part | fancy-hub-cap |
|       | is            | round         |
| roof  | locations-is  | highes-point  |
|       | may-be-type   | convertible   |

Dari segi bahasa yang digunakan :

- LISP , maka harus ditentukan terlebih dulu dengan tegas apakah ES yang akan dibangun akan menggunakan teknik pelacakan ke belakang / teknik pelacakan ke depan.
- Prolog , seperti turbo prolog, maka lebih baik menggunakan pelacakan ke belakang.

#### 6.4. Kemampuan Tambahan Diperlukan ES

*Untuk lebih meningkatkan kemampuan ES, diperlukan komponen-komponen tambahan:*

- fasilitas penjelasan
  - untuk menjelaskan bagaimana prosesnya sampai kesimpulan-kesimpulan tersebut diperoleh.
  - dengan cara memperlihatkan *rule – rule* yang digunakan
- kemudahan memodifikasi (K Bs)
  - dikarenakan ilmu pengetahuan berkembang
  - kemampuan seorang pakar bertambah terus
- kompatibilitas
  - dapat dijalankan pada berbagai jenis komputer.
- kemampuan belajar
  - kemampuan ES untuk menambah sendiri pengolahannya, selama interaksi dengan pemakainya.

### 6.5. Klasifikasi sistem pakar

Klasifikasi sistem pakar berdasarkan kegunaannya:

a. Diagnosis :

- ❖ Digunakan untuk merekomendasikan:
  - obat untuk orang sakit ,kerusakan mesin, kerusakan rangkaian elektronik.
- ❖ Menemukan apa masalah/kerusakan yang terjadi.
- ❖ Menggunakan pohon keputusan (*decision tree*) sebagai representasi pengetahuannya.

b. Pengajaran :

- ❖ Digunakan untuk pengajaran, mulai dari SD s/d PT.
- ❖ Membat diagnosa apa penyebab kekurangan dari siswa, kemudian memberikan cara untuk memperbaikinya.

c. Interpretasi

- ❖ Untuk menganalisa data yang tidak lengkap, tidak teratur, dan data yang kontradiktif.  
Misal: untuk interpretasi citra

d. Prediksi :

- ❖ Contoh : - Bagaimana seorang pakar meteorologi memprediksi cuaca besok berdasarkan data-data sebelumnya.
- ❖ Untuk peramalan cuaca.
- ❖ Penentuan masa tanam.

e. Perencanaan :

- ❖ mulai dari perencanaan mesin-mesin s/d manajemen bisnis.
- ❖ Untuk menghemat biaya, waktu & material, sebab pembuatan model
- ❖ sudah tidak diperlukan.
- ❖ contoh : - sistem konfigurasi komputer.

f. Kontrol

- ❖ Digunakan untuk mengontrol kegiatan yang membutuhkan presisi waktu tinggi.
- ❖ Misal : pengontrolan pada industri-industri berteknologi tinggi.

## Kategori dari aplikasi sistem berbasis pengetahuan

| Category       | Problem Addressed   | Types of Systems  |
|----------------|---|---|
| Diagnosis      | Infers system malfunctions from observations                | Medical, electronic, financial analysis, auditing, machine repair     |
| Monitoring     | Compares observations in order to identify variations       | Management control, nuclear power, plant regulation                   |
| Debugging      | Prescribes remedies for malfunctions                        | Computer Software   |
| Repair         | Executes a plan to administer a prescribed remedy           | Automobile, computer, telephone                                       |
| Instruction    | Diagnoses, debugs, and corrects student behaviors           | Tutorial, remedial  |
| Control        | Interprets, predicts, repairs, and monitors system behavior | Air traffic control, battle management, manufacturing process control |
| Prediction     | Infers likely consequences of given situations              | Weather forecasting, crop estimation, financial forecasting           |
| Interpretation | Infers situations descriptions from sensor data             | Speech understanding, image analysis, surveillance, mapping           |
| Design         | Configures objects within situation constraints             | Circuit layout, budgeting, automatic program generation               |
| Planning       | Develops guidelines for action                              | Strategic planning, process scheduling, military planning             |
| Classification | Prescribes categories for given sets of criteria            | Planning, scheduling, layout, remedial, audit                         |

Beberapa sistem pakar yang terkenal :

- Mycin , dirancang oleh Edward Short life of Standford University dalam tahun 70-an.
- Dendral , merupakan produk peneliti di Universitas Standford.
  - dengan menggunakan pengetahuannya struktur molekular dan kimia.
  - Berusaha mengidentifikasi struktur molekul campuran yang tak dikenal.

➤ XCON & XSEL

XCON dikenal sebagai RI dalam tingkat prototype dini.

- Membantu konfigurasi sistem komputer besar.
- Dikembangkan bersama dengan digital Equipment Corporation (DEC) & para peneliti, Carnegia Mellon University (CNU).
- Membantu melayani order langganan sistem computer DEC VAX II/780 le dalam sistem spesikasi final yang lengkap.

XSEL : ES DEC-CMU lainnya.

- Dirancang untuk membantu karyawan bagian penjualan dalam memilih komponen sistem VAX (dengan pengetahuannya sistem komputer VAX II/780).
- Pengetahuan yang ada dalam XSEL membantu untuk memilih konfigurasi yang dikehendaki, kemudian XSEL memilih CPU, memori, peripheral, dalam menyarankan paket *Software* tertentu yang paling tepat dengan konfigurasinya.
- Bisa merancang *Layout* dasar suatu sistem kompeter.

➤ Prospector

- Membantu ahli geologi dalam mencari & menemukan hasil tambang di bumi.
- Berisi taxonomi berbagai macam mineral & batu-batuan.
- Berisi bentuk-bentuk kandungan mineral/ batu-batuan yang diperoleh dari para ahli geologi.
- Mengevaluasi areal dalam bentuk pertanyaan dan menentukan apakah jenis batu-batuan tertentu potensial itu terdapat di tempat tertentu.



- Contoh aplikasi sistem berbasis pengetahuan komersial (*Commercial Knowledge Base System Applications*) yang lainnya :
  - ❖ ACE (AT & T)  
Digunakan untuk memberikan laporan trouble-shooting dan analisa untuk perawatan kabel telepon.
  - ❖ AS/ASQ (Arthur Young)  
Digunakan untuk membantu dalam prosedur auditing
  - ❖ AUDITOR(University of Illinois)  
Memilihkan prosedur audit untuk memverifikasi rekening pendapatan sebuah perusahaan.
  - ❖ AUTHORIZER'S ASSISTANT( American Express)  
Membantu dalam meninjau penipuan kartu kredit.
  - ❖ BUSINESS PLAN(Sterling Wentworth Corp.)  
Membantu pegawai professional dan pemilik bisnis tentang semua aspek yang menyangkut perencanaan keuangan.
  - ❖ CASH VALUE(Heuros Ltd.)  
Mendukung perencanaan proyek modal.
  - ❖ COMPASS(GTE Corp.)  
Troubleshoots tidak berfungsinya sirkuit telepon
  - ❖ CONCEPT(Tyashare)  
Memproduksi model-model dari pasar yang disenangi konsumen
  - ❖ DELTA(GE)  
Membantu mendiagnosa dan memperbaiki kereta api listrik diesel.
  - ❖ EL
    - Digunakan untuk menganalisa sirkuit elektronik yang terbuat dari transistor, dioda & resistor.
    - Bekerja melalui diagram skematik dari sirkuit yang telah dimasukkan dalam komputer & EL menentukan karakteristik sirkuit, nilai voltage dan strum yang ada pada semua titik sirkuit.
    - Sangat baik untuk rekayasa rancangan & bantuan belajar operasi sirkuit elektronik & rancangan.

- ❖ EXPERT TAX(Coopers & Lybrand)  
Memberikan bimbingan menghitung pajak.
- ❖ FIN PLAN(Wright Patterson Air Force Base)  
Mendukung perencanaan keuangan pribadi.
- ❖ FINANCIAL ADVISOR(Palladian)  
Memberikan bimbingan keuangan pada proyek, produk dan penggabungan serta akuisisi.
- ❖ FOLIO(University of Stanford),  
Membantu manajer investasi portfolio memutuskan sasaran investasi kliennya dan memilihkan portfolio yang terbaik yang dibutuhkan .
- ❖ SOPHIE  
Untuk membantu mahasiswa belajar memecahkan masalah atau kesulitan sirkuit elektronik.(Dengan cara mensimulasikan sirkuit & masalahnya).
- ❖ GENESIS(Intelli Corp.)  
Membimbing insiyur genetic dalam menganalisa molekul DNA.
- ❖ INTELLIGENT SECRETARY(Nippon T & T)  
Menangani jadwal dari personel dalam sebuah perusahaan.
- ❖ TRADER'S ASSISTANT(A.D. Little)  
Membimbing pedagang sekuritas dalam mengakses pasar modal.