

Sistem Pakar

Definisi

Keuntungan dan kelemahan

Konsep Dasar

Bentuk dan Struktur Sistem

Basis Pengetahuan

Metode Inferensi

Ciri-ciri

Aplikasi dan Pengembangannya

Referensi

Giarrantano, J. and G.Riley – bab 1

Sri Kusumadewi – bab 5

Konsep Dasar SP

- **Definisi:**

Sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan para ahli.

- SP tidak untuk menggantikan kedudukan seorang pakar tetapi untuk memasyarakatkan pengetahuan dan pengalaman pakar tersebut.
- Menurut Efraim Turban, SP harus mengandung : keahlian, ahli, pengalihan keahlian, inferensi, aturan dan kemampuan menjelaskan.
 - Keahlian adalah suatu kelebihan penguasaan pengetahuan di bidang tertentu yang diperoleh dari pelatihan, membaca atau pengalaman.
- SP dikembangkan pertama kali oleh komunitas AI tahun 1960an. SP yang pertama adalah General Purpose Problem Solver (GPS) yang dikembangkan oleh Newel Simon.

Contoh Sistem Pakar

| Sistem Pakar | Kegunaan |
|----------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| MYCIN Dirancang oleh Edward Feigenbaum (Universitas Stanford) th '70 an | Diagnosa Penyakit |
| DENDRAL | Mengidentifikasi struktur molecular campuran yang tidak dikenal |
| XCON & XSEL Dikembangkan oleh DEC dan CMU, akhir '70 an | Membantu konfigurasi system computer besar |
| SOPHIE | Analisis sirkuit elektronik |
| PROSPECTOR Didesign oleh Sheffield Research Institute, akhir '70an | Digunakan di dalam geologi untuk membantu mencari dan menemukan deposit |
| FOLIO | Membantu memberikan keputusan bagi seorang manajer dalam hal stok broker dan investasi |
| DELTA | Pemeliharaan lokomotif listrik diesel |

Keuntungan VS Kelemahan SP

Keuntungan SP :

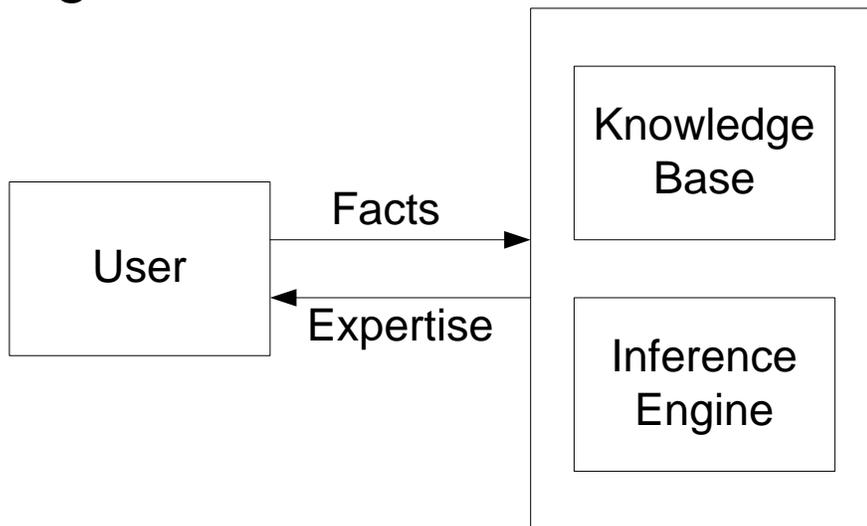
1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli
2. bisa melakukan proses secara berulang secara otomatis
3. menyimpan pengetahuan dan keahlian para pakar
4. meningkatkan output dan produktivitas
5. meningkatkan kualitas
6. mampu mengambil dan melestarikan keahlian para pakar
7. mampu beroperasi dalam lingkungan berbahaya
8. memiliki kemampuan untuk mengakses pengetahuan
9. memiliki realibilitas
10. meningkatkan kapabilitas system computer
11. memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian
12. sebagai media pelengkap dalam pelatihan
13. meningkatkan kapabilitas dalam penyelesaian masalah
14. menghemat waktu dalam pengambilan keputusan

Kelemahan :

1. biaya yang diperlukan untuk membuat dan memeliharanya sangat mahal
2. sulit dikembangkan. Hal ini erat kaitannya dengan ketersediaan pakar dalam bidangnya
3. system pakar tidak 100% bernilai benar

Bentuk dan Struktur SP (1/2)

Blok Diagram SP

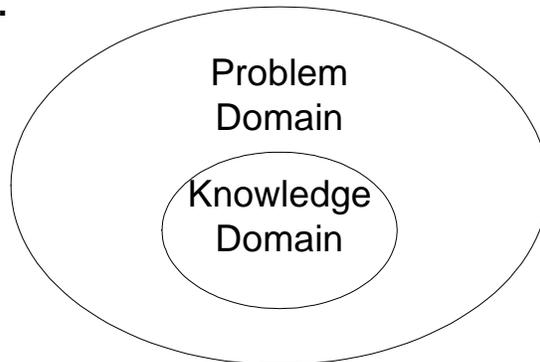


Expert System

- Bentuk pengetahuan :
 - fakta-fakta pada lingkup permasalahan tertentu
 - teori-teori pada lingkup masalah tertentu
 - prosedur-prosedur berkenaan dengan lingkup masalah tertentu
 - strategi-strategi global untuk menyelesaikan masalah
 - *meta-knowledge* (pengetahuan tentang pengetahuan)

Bentuk dan Struktur SP (2/2)

- Knowledge base (basis pengetahuan) berisi pengetahuan-pengetahuan dalam penyelesaian masalah.
 - Domain pengetahuan seorang pakar pada dasarnya adalah spesifik terhadap domain masalah.



- Inference engine (motor inferensi) bertugas untuk menganalisis pengetahuan dan menarik kesimpulan berdasarkan knowledge base.

Sistem Konvensional vs. Sistem Pakar (SP)

| Sistem Konvensional | Sistem Pakar (ES) |
|-------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| Informasi dan pemrosesan biasanya jadi satu dengan program | Basis pengetahuan merupakan bagian terpisah dari mekanisme inferensi |
| Biasanya tidak bisa menjelaskan mengapa suatu input data itu dibutuhkan atau bagaimana output itu diperoleh | Penjelasan adalah bagian terpenting dari system pakar |
| Pengubahan program cukup sulit | Pengubahan aturan dapat dilakukan dengan mudah |
| Sistem hanya akan beroperasi jika system tersebut sudah lengkap | Sistem dapat beroperasi hanya dengan beberapa aturan |
| Eksekusi dilakukan langkah demi langkah | Eksekusi dilakukan pada semua basis pengetahuan |
| Menggunakan data | Menggunakan pengetahuan |
| Tujuan utamanya adalah efisiensi | Tujuan utamanya adalah efektivitas |

Ciri-ciri dan Domain SP

- Ciri-ciri SP :
 - Memiliki fasilitas informasi yang handal
 - Mudah dimodifikasi
 - Dapat digunakan dalam berbagai jenis komputer
 - Memiliki kemampuan untuk belajar beradaptasi.

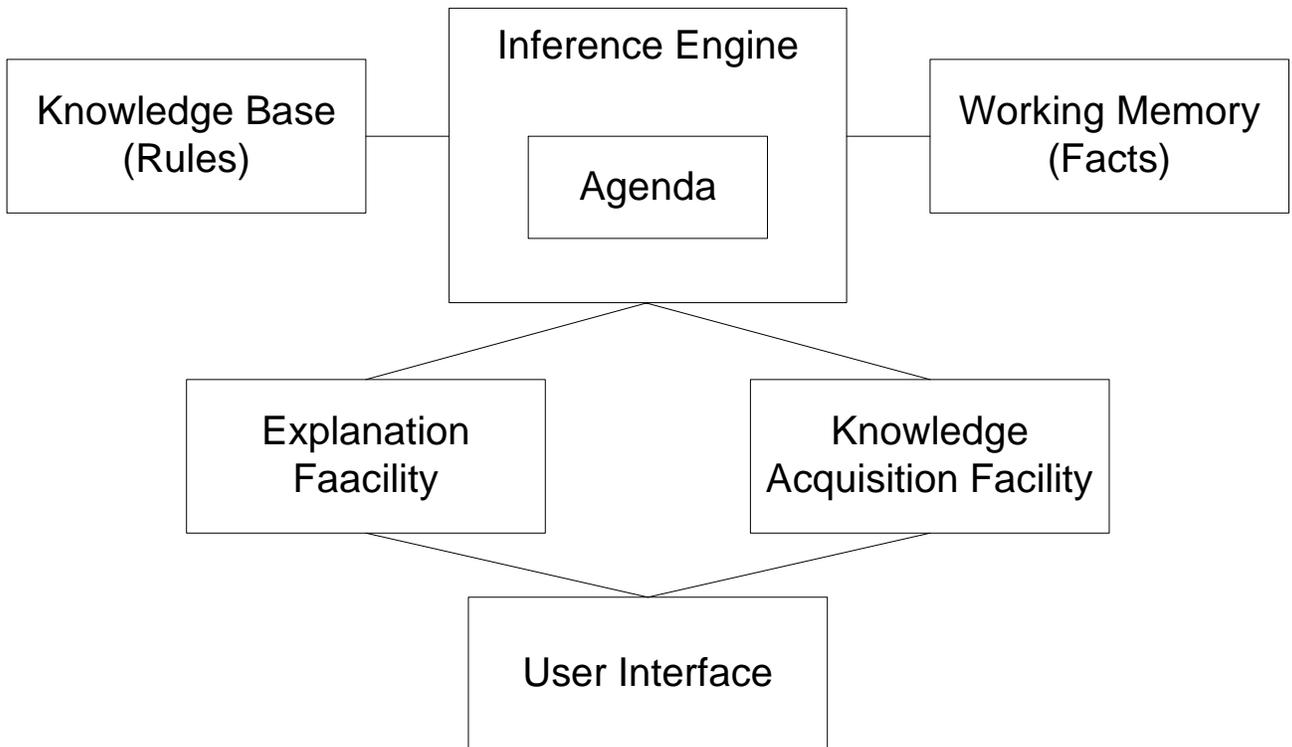
Permasalahan yang Disentuh oleh SP (Domain SP) :

- Interpretasi. Pengambilan keputusan dari hasil observasi, termasuk pengenalan ucapan, analisis citra, interpretasi sinyal, dll
- Prediksi : prediksi demografi, prediksi ekonomi, dll.
- Diagnosis : diagnosis medis, elektronis, mekanis, dll.
- Perancangan : perancangan layout sirkuit , bangunan.
- Perencanaan : perencanaan keuangan, militer, dll
- Monitoring : *computer aided monitoring system*
- Debugging : memberikan resep obat terhadap kegagalan
- Instruksi : melakukan instruksi untuk diagnosis, debugging dan perbaikan kinerja
- Kontrol : melakukan kontrol terhadap interpretasi, prediksi, perbaikan dan monitoring kelakukan sistem.

Bentuk SP

- Bentuk SP :
 - Berdiri sendiri. Sistem jenis ini merupakan s/w yang berdiri sendiri tidak tergabung dengan s/w lain.
 - Tergabung. Sistem ini merupakan bagian program yang terkandung di dalam suatu algoritma (konvensional) .
 - Menghubungkan ke s/w lain. Bentuk ini biasanya merupakan SP yang menghubungkan ke suatu paket program tertentu, misalnya DBMS.
 - Sistem mengabdikan. Sistem ini merupakan bagian dari komputer khusus yang dihubungkan dengan suatu fungsi tertentu.

Struktur SP (1/2)



Struktur SP (2/2)

- User interface (antarmuka) : mekanisme komunikasi antara user dan SP
- Explanation facility (subsistem Penjelasan) : digunakan untuk melacak respon dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif
- Working memory : database global dari fakta yang digunakan dalam prosedur
- Agenda : daftar prioritas prosedur yang dibuat oleh motor inferensi dan direkam dalam working memory
- Inference engine (motor inferensi) : program yang berisi metodologi yang digunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis pengetahuan untuk memformulasikan konklusi.
- Knowledge acquisition facility : berisi pengetahuan-pengetahuan yang dibutuhkan untuk memahami, memformulasikan dan menyelesaikan masalah.

Basis Aturan (*Rule Based*) SP (1/2)

- Pengetahuan dalam SP direpresentasikan dalam bentuk **IF-THEN** atau dalam bentuk ***Production Rules***.
- Motor inferensi menentukan aturan awal (rule antecedents) yang sesuai.
 - Sisi kiri harus cocok dengan fakta yang ada di memori kerja
- Aturan yang sesuai ditempatkan di agenda dan dapat diaktivasi
- Aturan yang terdapat di agenda dapat diaktivasi
 - Aktivasi aturan akan membangkitkan fakta baru di sisi kanan
 - Aktivasi dari satu aturan adalah bagian dari aktivasi aturan yang lain.

Basis Aturan (*Rule Based*) SP (2/2)

Contoh :

IF THEN Rules

Rule : Red_Light
IF the light is red
THEN stop

Antecedent
(left hand side)

Consequent
(right hand side)

Rule : Green_Light
IF the light is green
THEN go

Production Rules

The light is red → stop

The light is green → go

Antecedent
(left hand side)

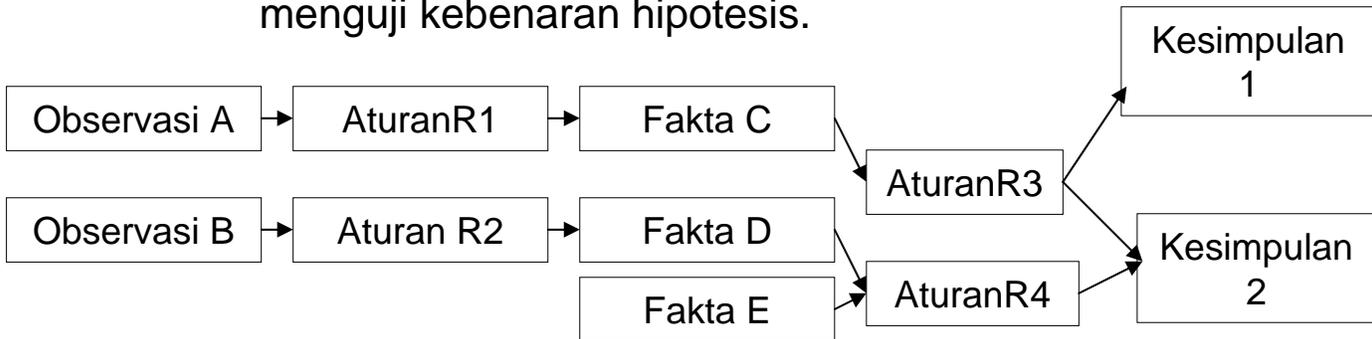
Consequent
(right hand side)

Siklus Motor Inferensi (*Inference Engine Cycle*)

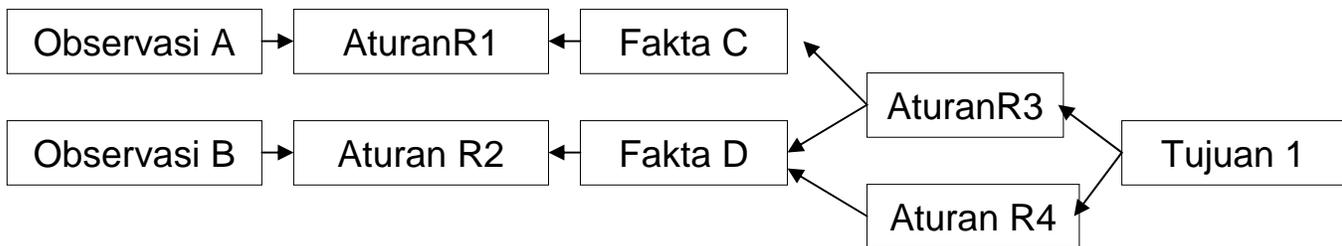
- Menggambarkan eksekusi dari aturan pada motor inferensi :
 - Conflict resolution : pemilihan aturan dengan prioritas tertinggi dari agenda
 - Execution : aksi *consequent* dari aturan yang terpilih
 - Match : pengkinian (update) agenda
- Siklus berakhir ketika tidak ada lagi aturan di agenda atau ketika ditemui perintah stop.

Forward dan Backward Chaining (1/2)

- Ada 2 cara yang dapat dilakukan dalam melakukan inferensi :
 - Forward Chaining. Pencocokkan fakta atau pernyataan dimulai dari bagian sebelah kiri (IF dulu). Dengan kata lain, penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis.



- Backward Chaining. Pencocokkan fakta atau pernyataan dimulai dari bagian sebelah kanan (THEN dulu). Dengan kata lain, penalarana dimulai dari hipotesis terlebih dahulu, dan untuk menguji kebenaran hipotesis tersebut dicari fakta-fakta yang ada dalam basis pengetahuan.



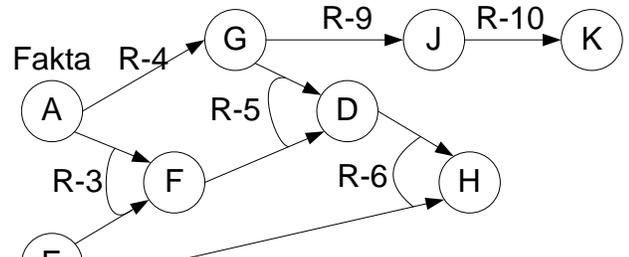
Forward dan Backward Chaining (2/2)

Contoh :

- Pada tabel di bawah ini terlihat 10 aturan yang tersimpan dalam basis pengetahuan. Fakta awal yang diberikan hanya A & F (artinya A dan F bernilai benar). Ingin dibuktikan apakah K bernilai benar (hipotesis : K) ?

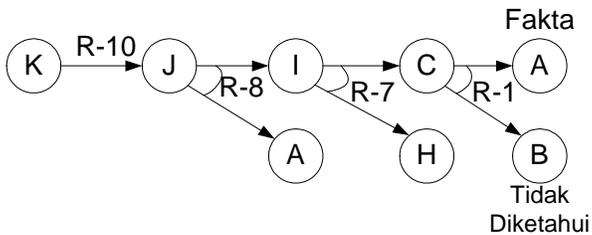
| No | Aturan |
|------|-----------------|
| R-1 | IF A & B THEN C |
| R-2 | IF C THEN D |
| R-3 | IF A & E THEN F |
| R-4 | IF A THEN G |
| R-5 | IF F & G THEN D |
| R-6 | IF G & E THEN H |
| R-7 | IF C & H THEN I |
| R-8 | IF I & A THEN J |
| R-9 | IF G THEN J |
| R-10 | IF J THEN K |

Penyelesaian dengan forward chaining

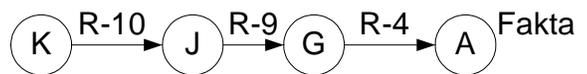


| Aturan | Fakta Baru |
|--------|--------------|
| R-3 | F |
| R-4 | G |
| R-5 | D |
| R-6 | H |
| R-9 | J |
| R-10 | K (terbukti) |

Penyelesaian dengan backward chaining



Pertama : Gagal



Kedua : Sukses