

Pengantar ke Expert System¹

oleh: Gunawan²

Expert System (ES), Knowledge-Based Expert System, Knowledge-Based System (KBS), atau *sistem pakar* adalah sebuah program berbasis pengetahuan yang menyediakan penyelesaian "berkualitas pakar" untuk masalah-masalah dalam sebuah bidang yang spesifik.

ES adalah salah satu sub bidang *artificial intelligence*, mendampingi *Natural Language Processing (NLP)* atau Pengolahan Bahasa Alami, dan *Robotics* atau Robotika.

Secara umum pengetahuan di dalam ES disarikan dari para pakar pada bidang yang spesifik itu, sehingga ES cenderung menjadi "seorang" spesialis yang terfokus pada sejumlah masalah yang sempit dan terbatas. Seperti juga manusia, pengetahuan dalam ES diperoleh secara teori dan praktek (pengalaman).

Melalui ES, sistem komputer melakukan ekstraksi informasi tambahan dari user dengan memberikan sejumlah pertanyaan yang terkait dengan permasalahan selama konsultasi.

Pemakaian ES:

- ♦ oleh orang awam yang bukan pakar untuk meningkatkan kemampuan mereka dalam memecahkan masalah.
- ♦ oleh pakar sebagai asisten yang berpengetahuan.
- ♦ untuk memperbanyak atau menyebarkan sumber pengetahuan yang semakin langka.

Ciri-ciri umum ES:

- ♦ Terbatas pada bidang yang spesifik.
- ♦ Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
- ♦ Dapat mengemukakan rangkaian alasan yang diberikannya dengan cara yang dapat dipahami.
- ♦ Berdasarkan pada rule atau kaidah tertentu.
- ♦ Dirancang untuk dapat dikembangkan secara bertahap.
- ♦ Outputnya bersifat nasihat atau anjuran.
- ♦ Output tergantung dari dialog dengan user.
- ♦ Knowledge base dan inference engine terpisah.

Keuntungan pemakaian ES:

- ♦ Membuat seorang yang awam dapat bekerja seperti layaknya seorang pakar.
- ♦ Dapat bekerja dengan informasi yang tidak lengkap atau tidak pasti.
- ♦ Meningkatkan output dan produktivitas. ES dapat bekerja lebih cepat dari manusia. Keuntungan ini berarti mengurangi jumlah pekerja yang dibutuhkan, dan akhirnya akan mereduksi biaya.³

¹Edisi Revisi, dilengkapi kembali dengan sejumlah informasi dari berbagai sumber untuk kuliah Artificial Intelligence semester genap 1997/1998 (Maret-Juli 1998).

²Penulis adalah Dosen Artificial Intelligence dan Ketua Jurusan Teknik Informatika dan Komputer Sekolah Tinggi Teknik Surabaya.

³Termasuk ES yang menjadi bagian AI, "*musuh utama*" dari bidang ini adalah kaum pekerja yang merasa tersaingi dan kaum religi dari sisi etika.

- ♦ Meningkatkan kualitas. ES menyediakan nasihat yang konsisten dan dapat mengurangi tingkat kesalahan.
- ♦ Membuat peralatan yang kompleks lebih mudah dioperasikan karena ES dapat melatih pekerja yang tidak berpengalaman.
- ♦ Handal (*reliability*). ES tidak bisa lelah atau bosan, juga konsisten dalam memberi jawaban dan selalu memberikan perhatian penuh.
- ♦ Kemampuan untuk memecahkan masalah yang kompleks.
- ♦ Pemindahan pengetahuan ke lokasi yang jauh serta memperluas jangkauan seorang pakar, dapat diperoleh dan dipakai di mana saja.
- ♦ Merupakan arsip yang terpercaya dari sebuah keahlian, sehingga user seolah-olah berkonsultasi langsung dengan sang pakar, meskipun mungkin sang pakar sudah pensiun.

Sejarah ES

ES mulai dikembangkan pada pertengahan tahun 1960-an oleh *Artificial Intelligence Corporation*. Periode penelitian artificial intelligence ini didominasi oleh suatu keyakinan bahwa nalar yang digabung dengan komputer canggih akan menghasilkan prestasi pakar atau bahkan manusia super.

Suatu usaha ke arah ini adalah *General Purpose Problem-Solver (GPS)*. GPS yang berupa sebuah prosedur yang dikembangkan -- oleh Allen Newell, John Cliff Shaw, dan Herbert Alexander Simon -- dari *Logic Theorist* merupakan sebuah percobaan untuk menciptakan mesin yang cerdas. GPS sendiri merupakan sebuah *predecessor* menuju ES. GPS berusaha untuk menyusun langkah-langkah yang dibutuhkan untuk mengubah situasi awal menjadi state tujuan yang telah ditentukan sebelumnya.

Pada pertengahan tahun 1960-an, terjadi pergantian dari program serba bisa (*general-purpose*) ke program yang spesialis (*special-purpose*) dengan dikembangkannya DENDRAL oleh E. Feigenbaum dari Universitas Stanford dan kemudian diikuti oleh MYCIN. Pembuatan DENDRAL mengarah pada konklusi-konklusi berikut:

- ♦ GPS terlalu lemah untuk digunakan sebagai dasar untuk membangun ES yang berunjukkerja tinggi.
- ♦ Pemecahan masalah manusia adalah baik hanya jika beroperasi dalam domain yang sangat sempit.
- ♦ ES harus di-update secara berkala untuk informasi baru. Update semacam ini dapat efisien apabila menggunakan representasi pengetahuan berbasis rule.
- ♦ Problem yang kompleks membutuhkan pengetahuan yang banyak sekali tentang area problem.

Pada pertengahan tahun 1970-an, beberapa ES mulai muncul. Sebuah pengetahuan kunci yang dipelajari saat itu adalah "*Kekuatan dari ES berasal dari pengetahuan spesifik yang dimilikinya, bukan dari formalisme-formalisme khusus dan pola penarikan kesimpulan yang digunakannya*".

Awal 1980-an, teknologi ES yang mula-mula dibatasi oleh suasana akademis mulai muncul sebagai aplikasi komersil, khususnya XCON, XSEL (dikembangkan dari R-1 pada Digital Equipment Corp.) dan CATS-1 (dikembangkan oleh General Electric).

Beberapa perkembangan terakhir ES:

- ◆ Banyaknya peralatan yang didesain untuk mempercepat pembuatan ES sehingga dapat mereduksi biaya rekayasa.
- ◆ Penyebaran ES dalam ribuan organisasi.
- ◆ Penggabungan ES dengan komputer berbasis sistem informasi lain, terutama dengan database dan *Decision Support System (DSS)*.
- ◆ Peningkatan pemakaian ES dalam banyak pekerjaan, mulai dari bidang perbankan sampai bidang militer yang kompleks dan aplikasi ruang angkasa.
- ◆ Penggunaan teknologi ES sebagai sebuah metologi untuk mempercepat atau memperlancar pembuatan sistem informasi biasa.
- ◆ Peningkatan pemakaian dari pendekatan *Object Oriented Programming (OOP)* dalam representasi pengetahuan, khususnya untuk *frame*.
- ◆ Pengembangan sistem kompleks dengan banyak sumber pengetahuan, banyak jalur penalaran, informasi fuzzy, ataupun neural network, seperti Neuro Logic.
- ◆ Penggunaan bermacam-macam basis pengetahuan.

Alat Pengembangan ES

Sebagai tambahan, dikembangkan juga alat-alat untuk mempercepat pembuatan ES. Alat-alat tersebut termasuk:

- ◆ alat perolehan pengetahuan seperti EXPERT dan KAS
- ◆ alat-alat untuk belajar dari pakar seperti META-DENDRAL dan EURISKO
- ◆ alat pemrograman seperti EMYCIN dan AGE, termasuk kemudian ES-Shell (M1, Guru, Level-5, VP-Expert), yaitu suatu paket software yang secara khusus dibuat untuk membantu mengembangkan sebuah program ES.

Alat-alat ini mulai tersedia secara komersil mulai tahun 1983.

Level Pengoperasian ES

ES dapat dioperasikan ke dalam 3 (tiga) level, masing-masing sebagai:

- ◆ Asisten,
- ◆ Kolega, atau
- ◆ Pakar

Kategori ES

Berdasarkan tujuan pembuatannya, ES dikategorikan menjadi:

1. Interpretasi (*Interpretation*)

Bertujuan untuk menganalisa data yang tidak lengkap, tidak teratur dan data yang kontradiktif yang biasanya diperoleh melalui sensor. Contoh: analisis citra.

2. Prediksi (*Prediction*)

Bertujuan untuk memberikan kesimpulan mengenai akibat yang mungkin ditimbulkan dari sejumlah situasi yang diberikan. Contoh: *financial forecasting*.

3. Diagnosa (*Diagnosys*)

Bertujuan untuk melakukan diagnosa untuk menentukan penyebab gagalnya sistem dalam situasi kompleks yang didasarkan pada observasi terhadap gejala-gejala yang dapat diamati. Prinsipnya adalah untuk menemukan apa masalah atau kerusakan yang terjadi. Contoh: diagnosa penyakit di bidang kedokteran.

4. Disain (*Design*)

Bertujuan untuk menentukan konfigurasi yang cocok dari komponen-komponen yang ada pada sebuah sistem sehingga unjuk kerja yang memuaskan dapat diperoleh walaupun didalamnya terdapat sejumlah keterbatasan. Contoh: *layout circuit*.

5. Perencanaan (*Planning*)

Bertujuan untuk mendapatkan urutan tindakan yang harus dilakukan untuk mencapai sasaran yang ditentukan sebelumnya dari suatu kondisi awal tertentu. Contoh: lengan yang robot harus memindahkan tiga balok dengan susunan tertentu dari susunan asal yang acak.

6. Pengamatan (*Monitoring*)

Bertujuan untuk membandingkan perilaku yang diamati dalam suatu sistem dengan perilaku yang diharapkan untuk mengenal variasi perilaku yang terdapat didalamnya. Contoh: control instalasi nuklir.

7. Pelacakan dan Perbaikan (*Debugging dan Repair*)

Bertujuan untuk menentukan dan mengimplementasikan perbaikan pada kegagalan suatu sistem. Contoh: tahap ujicoba software komputer.

8. Instruksi (*Instruction*)

Bertujuan untuk mendeteksi dan memperbaiki kekurangan perilaku siswa dalam memahami bidang informasi tertentu. Contoh: program tutorial.

9. Kontrol (*Control*)

Bertujuan untuk pengaturan perilaku kerja sistem dalam suatu lingkungan yang kompleks, termasuk didalamnya adalah penafsiran, perkiraan, pengawasan dan perbaikan perilaku kerja sistem tersebut. Contoh: control terhadap proses manufacturing lengkap.

10. Klasifikasi (*Classification*).

Bertujuan untuk menentukan kategori-kategori dari sejumlah kriteria yang diberikan. Contoh: menentukan bidang pekerjaan yang cocok untuk seorang calon pegawai.

3 (tiga) Unsur Manusia dalam ES

Unsur manusia yang berpartisipasi dalam pengembangan dan pemakaian ES:

1. *Domain Expert* (para pakar)

Sebenarnya sulit untuk mendefinisikan apakah yang dimaksud dengan pakar itu. Masalahnya adalah berapa banyak keahlian yang harus dimiliki oleh seseorang sebelum ia dapat dikualifikasikan sebagai seorang pakar.

Pakar adalah seseorang yang mempunyai pengetahuan khusus, pendapat, keahlian dan metode serta kemampuan menggunakannya untuk memberikan nasihat dan memecahkan masalah. Tugas dari para pakar ini adalah menyediakan pengetahuan

bagaimana dia melaksanakan tugasnya, pengetahuan ini kemudian diserap dan diduplikasikan ke ES.

Karakteristik para pakar:

- ♦ memecahkan masalah dengan cepat dan akurat
- ♦ menjelaskan apa dan bagaimana mereka melakukannya
- ♦ mempertimbangkan apakah kesimpulan mereka sudah benar
- ♦ mengetahui ketika mereka menemui jalan buntu
- ♦ berkomunikasi dengan pakar-pakar lain.
- ♦ belajar dari pengalaman
- ♦ mengubah sudut pandang untuk disesuaikan dengan persoalan
- ♦ mampu mentransfer pengetahuan
- ♦ menggunakan peralatan seperti heuristik, sejumlah model matematika, dan simulasi-simulasi untuk mendukung keputusan mereka.

Keahlian (*expertise*) pakar dapat diperoleh dari pelatihan (*training*), membaca atau dari pengalaman. Keahlian tersebut meliputi:

- ♦ Fakta-fakta tentang area permasalahan.
- ♦ Teori-teori tentang area permasalahan.
- ♦ Aturan-aturan apa yang harus dilakukan dalam situasi permasalahan yang diberikan.
- ♦ Strategi global untuk memecahkan tipe-tipe masalah.

Keahlian yang harus dipindahkan inilah (*transferring expertise*) yang memungkinkan seorang pakar dapat membuat keputusan yang lebih baik dan lebih cepat daripada orang yang bukan pakar dalam memecahkan masalah-masalah yang kompleks.

Untuk pakar yang ingin menambah keahlian, ES bertindak sebagai kolega atau teman sejawat. Ada kalanya ES juga digunakan sebagai asisten yang dapat diandalkan.

2. Knowledge Engineer

Adalah pihak yang membuat ES. Knowledge engineer ini bertugas menyerap dan menyarikan pengetahuan yang dimiliki oleh para pakar dan mengimplementasikannya ke dalam sebuah software ES. Tugas ini cukup sulit karena seorang knowledge engineer tidak boleh memasukkan perkiraan atau perasaannya ke dalam pengetahuan yang diperolehnya. Di samping itu dia juga harus pandai mengorek pengetahuan pakar, karena adakalanya seorang pakar -- *biasanya secara psikologis* -- tidak dapat menceritakan atau menjelaskan semua keahliannya.

3. User (pemakai)

Yaitu pihak yang mempergunakan program ES. Kemampuan ES dikembangkan untuk menghemat waktu dan usaha user. ES mempunyai beberapa tipe user antara lain:

- ♦ Klien bukan pakar yang meminta nasihat, dalam hal ini ES bertindak sebagai konsultan atau penasihat.
- ♦ Siswa yang tidak tahu bagaimana harus belajar, dalam ini ES akan bertindak sebagai seorang instruktur.
- ♦ Pembuat ES yang mau memperbaiki atau menambah knowledge base, dalam ini ES akan bertindak sebagai partner.

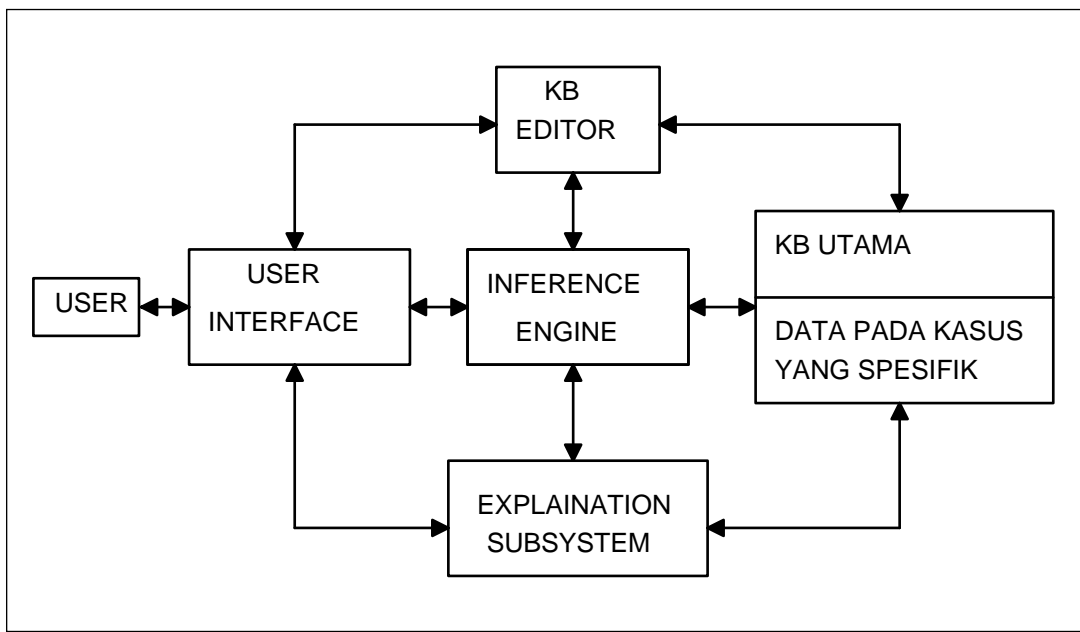
Arsitektur ES

Sebagai sebuah sistem yang selalu memiliki sejumlah komponen yang harus berinteraksi untuk mencapai tujuan, komponen utama yang harus ada dalam sebuah ES adalah:

- ◆ Knowledge Base (Basis Pengetahuan)
- ◆ Inference Engine (Mesin Penarik Kesimpulan)
- ◆ Explanation Subsystem (Subsistem Penjelas Output)
- ◆ User-Interface (Penghubung ke Pemakai).

Sedangkan komponen-komponen berikut *tidaklah mutlak* dalam sebuah ES:

- ◆ Knowledge Base Editor
- ◆ Data pada Kasus yang Spesifik



1. Knowledge Base

Knowledge base atau basis pengetahuan merupakan representasi pengetahuan dari seorang pakar yang diperlukan untuk memahami, memformulasikan dan memecahkan masalah. Terdiri dari dua elemen dasar, yaitu:

- ◆ *Fakta* yang berupa informasi tentang situasi permasalahan, teori dari area permasalahan atau informasi tentang objek.
- ◆ *Spesial heuristik* yang merupakan informasi tentang cara bagaimana membangkitkan fakta baru dari fakta yang sudah diketahui. Dalam ES berbasis rule, bagian ini berupa *rules*.

Knowledge base adalah jantung sebuah ES. Bagian ini adalah totalitas keahlian pakar yang telah disarikan dan diformat ke dalam external memory komputer. Sampai saat ini terdapat berbagai cara representasi knowledge yang telah dikenal, misalnya penyajian yang berorientasi obyek, jaringan semantik, frame, script ataupun production rules. Yang paling banyak digunakan pada pengembangan ES dengan *shell* (paket program) adalah yang berbasis production rules - biasa disebut dengan *rule* saja - sehingga ES-nya sering disebut sebagai *Rule-based Expert System*.

2. Inference Engine

Inference engine merupakan otak dari ES, bagian ini mengandung mekanisme fungsi berpikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar. Mekanisme ini akan menganalisa suatu masalah tertentu dan kemudian mencari jawaban atau kesimpulan yang terbaik. Dari fakta-fakta yang diperoleh selama proses tanya-jawab dengan user, serta rule-rule yang tersimpan di knowledge base, inference engine dapat menarik suatu kesimpulan dan memberikan rekomendasi atau saran yang diharapkan oleh user.

3. *Explanation Subsystem*

Bagian yang harus siap memberikan penjelasan saat mana user perlu mengetahui apakah alasan diberikannya sebuah solusi. Bagian ini secara konkrit membedakan sebuah ES dengan sistem aplikasi yang biasa, karena pada pemrograman konvensional tidaklah biasa sebuah sistem menyediakan informasi tambahan mengapa atau dari mana sebuah solusi diperoleh.

Bagian ini mempunyai kemampuan untuk menelusuri konklusi dan menerangkan tingkah laku ES dengan menjawab pertanyaan-pertanyaan seperti:

- ♦ Mengapa pertanyaan tersebut diajukan oleh ES?
- ♦ Bagaimana atau darimana konklusi tersebut diperoleh?
- ♦ Mengapa alternatif tersebut ditolak?

Pada sistem pakar berbasis rule, biasanya penjelasan ini dilakukan dengan cara memperlihatkan rule-rule yang digunakan. Fasilitas ini penting untuk menambah rasa percaya user pada hasil output program ES yang digunakannya.

4. *User Interface*

Merupakan bagian dari ES yang berfungsi sebagai pengendali input output. User interface melayani user selama proses konsultasi mulai dari tanya-jawab untuk mendapatkan fakta-fakta yang dibutuhkan oleh inference engine sampai menampilkan output yang merupakan kesimpulan atau rekomendasi yang dihasilkan oleh inference engine.

5. *Knowledge-base Editor*

Bagian yang digunakan untuk menambah, menghapus atau memperbaiki basis pengetahuan. Bagian ini tidaklah mutlak, karena mayoritas ES berbasis pengetahuan dalam format *text-file*, sehingga bagian ini dapat digantikan dengan berbagai word processor yang tersedia. Namun demikian bila ES dituntut untuk memiliki kemampuan *machine learning* dari pengalaman konsultasinya, bagian ini menjadi sangat vital.

6. *Data pada Kasus yang Spesifik*

Bagian ini hanya diperlukan saat mana data yang telah dimiliki pemakai - *file database* atau *spreadsheet* - diperlukan sebagai referensi untuk menarik kesimpulan.

