

13

dan Sistem Pakar

Pokok Bahasan:

- Definisi kecerdasan buatan
- Sistem pakar
- Database pengetahuan (Knowledge base)
- Penalar (Inference engine)
- Bahasa pemrograman sistem pakar (Development engine)

Pendahuluan

Bagian dari sistem informasi berbasis komputer yang paling banyak menarik perhatian para ilmuwan komputer adalah sistem pakar (*expert system*) yang juga merupakan bagian dari kecerdasan buatan (*artificial intelligence*) atau AI. Tidak seperti halnya SPK, sistem pakar memiliki potensi untuk memperluas kemampuan manajer dalam memecahkan masalah manajer melebihi kemampuan normalnya.

Sistem pakar terdiri dari empat bagian utama yaitu: bagian penghubung sistem dengan pemakai (*interface*), *database* pengetahuan (*knowledge base*), bagian penyimpul (*inference engine*), dan bahasa pemrograman untuk membangun sistem pakar (*development engine*). *Knowledge base* dengan menggunakan aturan-aturan yang telah ditentukan mengekspresikan logika permasalahan dengan bantuan sistem pakar .

Inference engine dengan menggunakan penalaran, yang serupa dengan penalaran manusia mengolah isi dari *data* pengetahuan atau *knowledge base* yang ada dalam sistem pakar. *Development engine* merupakan bahasa pemrograman atau (*prewritten inference engine*) yang juga disebut *shell* sistem pakar (*expert system shell*). Metode *Prototyping* merupakan metode yang paling cocok diterapkan untuk pengembangan sistem pakar. Bagian paling penting dalam sistem pakar, seperti halnya untuk semua subsistem CBIS, adalah bagian penghubung antara sistem pakar dengan pemakai yang disebut sebagai *interface*.

13.1 Definisi Kecerdasan Buatan

AI adalah aktifitas penyediaan mesin seperti komputer yang memiliki kemampuan menampilkan perilaku yang dianggap cerdas seperti yang terjadi pada manusia

Selama lima belas tahun terakhir ini telah terjadi peningkatan minat para pakar teknologi informasi untuk menggunakan komputer dalam aplikasi kecerdasan buatan. Aktifitas didalam bidang kecerdasan buatan (*artificial intelligence*) atau AI, adalah aktifitas penyediaan mesin seperti komputer yang memiliki kemampuan menampilkan perilaku yang dianggap cerdas seperti yang terjadi pada manusia. AI menunjukkan aplikasi komputer tercanggih saat ini yang mencoba agar komputer memiliki penalaran menyerupai beberapa jenis penalaran manusia.

Sejarah Kecerdasan Buatan (AI)

AI bukan merupakan terobosan baru dalam teknologi komputer. Cikal bakal AI mulai ada dua tahun setelah *General Electric* memasang komputer pertama aplikasi bisnis pada tahun 1956. Istilah kecerdasan buatan pertama kali dikemukakan oleh John McCarthy sebagai tema dalam suatu konferensi yang diadakan di Dartmouth College. Pada tahun yang sama program komputer AI pertama, yang disebut *Logic Theorist*, diumumkan dan ini telah mendorong para peneliti untuk mengembangkan program lain yang disebut *General Problem Solver (GPS)* dengan maksud untuk memecahkan segala jenis masalah umum akan tetapi mengalami kesulitan.

Penelitian AI berlanjut, tapi tertinggal dari aplikasi komputer yang kurang ambisius seperti MIS dan SPK. Namun, dengan berlalunya waktu, penelitian yang tekun terus memperluas batas-batas penggunaan komputer untuk tugas-tugas yang biasanya memerlukan kecerdasan manusia.

Bidang-bidang Kecerdasan Buatan (AI)

Penerapan AI dalam bidang bisnis dikenal sebagai sistem pakar (*expert system*)

Penerapan AI dalam bidang bisnis dikenal sebagai sistem pakar (*expert system*). Sistem pakar adalah program komputer yang beroperasi menyerupai pengetahuan seorang pakar dalam bentuk *heuristic*. Berbagai aturan yang ada didalam sistem pakar memungkinkan sistem ini untuk berfungsi secara konsisten seperti seorang manusia pakar (ahli). Sistem pakar dapat memberi nasehat kepada pemakai cara memecahkan masalah. Sistem pakar dapat berfungsi sebagai konsultan, karena itu kegiatan menggunakan disebut konsultasi pemakai berkonsultasi dengan sistem pakar untuk mendapatkan nasehat.

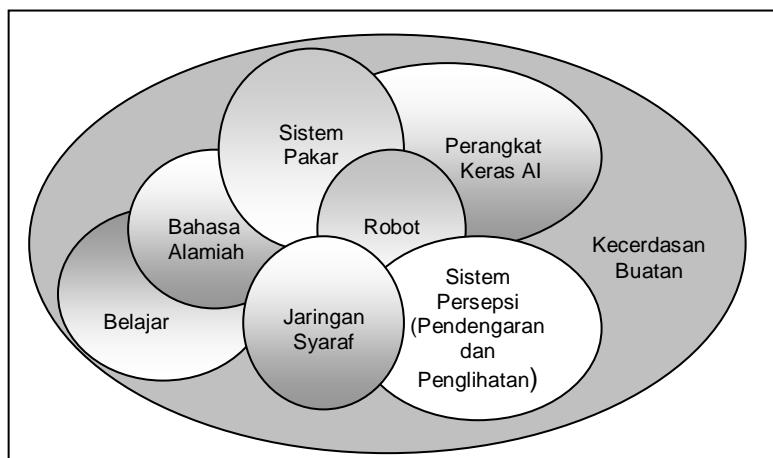
Selain sistem pakar, bidang yang mencakup kecerdasan buatan meliputi:

- **Jaringan syaraf (*neural network*)** adalah model sistem syaraf manusia yang disederhanakan tapi masih tetap menunjukkan kemampuan seperti belajar, generalisasi dan abstrak-

- si. Dengan kemampuan ini dimungkinkan model jaringan syaraf untuk mempelajari perilaku manusia.
- **Sistem persepsi (perceptive systems)** adalah sistem yang menggunakan citra visual (penglihatan) dan sinyal suara untuk mengintruksikan komputer atau alat lain, seperti robot.
 - **Belajar (learning)** adalah semua kegiatan yang memungkinkan komputer dan alat lainnya untuk memperoleh pengetahuan tambahan dari apa yang selama ini telah dimasukkan kedalam memori oleh pabrik komputer atau programer.
 - **Robotik (robotics)** merupakan sistem komputer yang mengendalikan alat-alat sehingga alat-alat tersebut dapat meniru aktivitas manusia.
 - **Perangkat keras kecerdasan buatan (AI hardware)** merupakan peralatan fisik yang dapat membantu aplikasi AI. Seperti perangkat keras yang digunakan untuk menyimpan sistem *database* pengetahuan (*knowledge based system*), sistem jaringan syaraf (*neural network*) dan digunakan juga untuk membantu mempercepat perhitungan, sebagai retina dan telinga elektronik.
 - **Pengolah bahasa alamiah** yang memungkinkan para pemakai dengan bantuan komputer dapat berbicara, menulis atau menterjemahkan dalam salah satu bahasa seperti, Jerman, Spanyol, Cina, Inggris, Indonesia, Perancis dan lain-lain.

Bidang-bidang kecerdasan buatan ini dapat dilihat pada gambar 13.1. Perhatikan bahwa terdapat beberapa area yang tumpang tindih. Tumpang tindih menunjukkan bahwa suatu bidang dapat memanfaatkan bidang yang lain. Tiap area ini memiliki potensi untuk digunakan dalam masalah bisnis, tapi sejauh ini yang paling diterima adalah bentuk sistem pakar.

Gambar 13.1 Bidang-bidang kecerdasan buatan



Daya Tarik Sistem Pakar

Beberapa tugas tertentu membutuhkan pengetahuan khusus sehingga diperlukan para pakar. Sayangnya, tidak setiap manajer mampu membayar staf spesialis atau memanggil konsultan tiap kali timbul suatu masalah. Konsep sistem pakar didasarkan pada asumsi bahwa pengetahuan pakar dapat diambil atau disimpan dalam komputer dan kemudian digunakan oleh orang lain saat dibutuhkan.

Perbedaan Sistem Pakar dengan SPK

Sistem pakar sangat serupa dengan SPK dalam hal keduanya memberikan dukungan sangat tinggi terhadap pemecahan masalah pemakainya. Namun, kedua sistem CBIS itu berbeda dalam dua hal.

- **Perbedaan Pertama** - SPK menunjukkan kebiasaan, gaya kemampuan, atau rutinitas manajer dalam menghadapi setiap permasalahan yang harus dipecahkan. Sistem pakar dipihak lain memberikan kesempatan untuk membuat keputusan yang melebihi kemampuan manajer. Contoh keputusan untuk melakukan investasi baru dapat dilakukan dengan bantuan sistem pakar.
- **Perbedaan kedua** - Antara sistem pakar dan SPK adalah kemampuan sistem pakar untuk menjelaskan alur penalarannya sampai timbul pemecahan tertentu. Sering kali penjelasan mengenai cara pemecahan masalah lebih berharga dari pemecahan masalah itu sendiri.

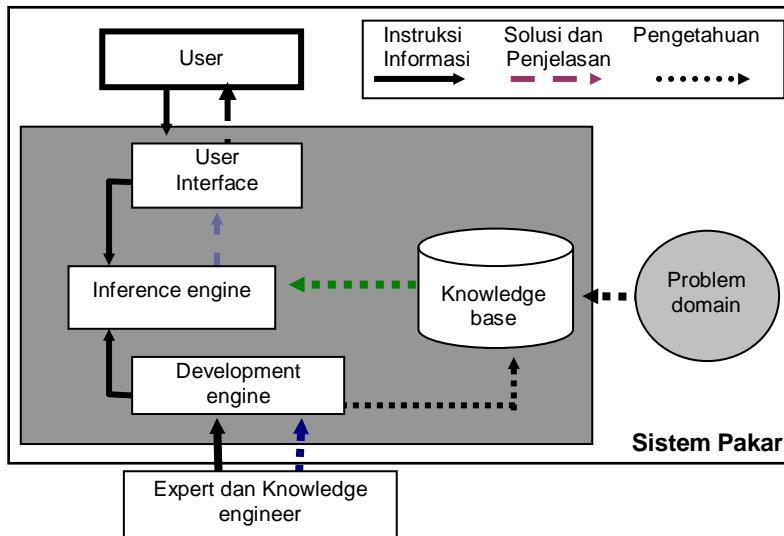
13.2 Sistem Pakar

Model sistem pakar pada Gambar 13.2 terdiri dari empat bagian utama, yaitu :

- **User interface (penghubung antara sistem pakar dengan pemakai)** - memungkinkan pemakai untuk berinteraksi dengan sistem pakar. Bisa dalam bentuk layar monitor, *mouse* dan *keyboard*
- **Knowledge base (database pengetahuan)** - menyimpan pengetahuan untuk masalah-masalah tertentu yang akan diselesaikan.
- **Inference engine (Penalar)** - memberikan kemampuan menalar yang menafsirkan isi *database pengetahuan* (*knowledge base*)

- **Development engine (Bahasa pemrograman)** yang digunakan oleh para pakar dan analisis sistem dalam membuat sistem pakar.

Gambar 13.2 Model sistem pakar



Sumber :Mc Leod

User Interface

User interface memungkinkan manajer untuk memasukkan instruksi dan informasi kedalam sistem pakar dan menerima informasi dari sistem pakar. Instruksi yang dimasukan berbentuk parameter (nilai tertentu) yang memungkinkan sistem pakar melakukan proses penalaran.

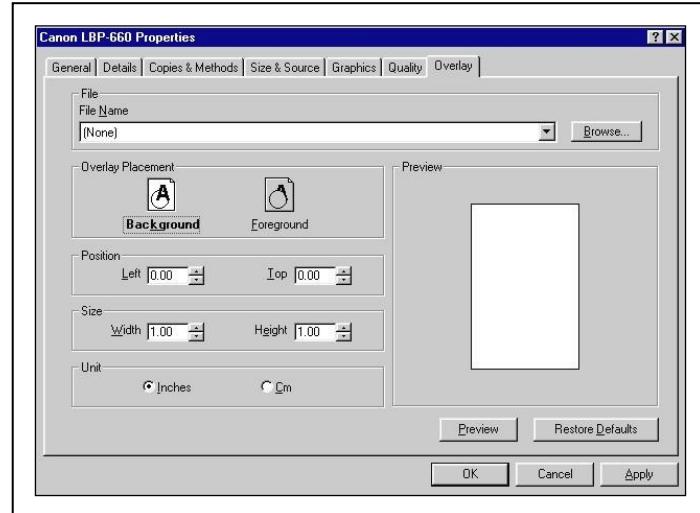
Input Sistem Pakar

Manajer dapat menggunakan empat cara memasukan *data/informasi*, yaitu melalui:

- **Menu** - Misalnya seperti menu yang terdapat dalam aplikasi excel atau lotus123
- **Perintah (commands)** - seperti perintah yang terdapat pada DBMS umumnya seperti foxpro, clipper, Dbase V. Sebagai contoh misalnya LIST FILES ON A: LIKE PAYROLL.*),
- **Bahasa alamiah (natural language)** - seperti bahasa-bahasa yang digunakan pada bahasa generasi keempat (4GL) seperti SQL : SELECT NAMA FROM KARYAWAN dan lain-lain, WHO ARE THE EMPLOYEES IN OUR NEW OFFICE WITH A COLLEGE EDUCATION.

Interface yang dapat disesuaikan atau *customized interfaces*. Interface ini menunjukkan adanya aspek tertentu dari masalah yang harus dipecahkan berbeda dari yang biasanya sehingga harus disesuaikan. Gambar 13.3 menunjukkan contoh bagaimana *interface* dapat disesuaikan bagi masalah yang sedang dipecahkan.

Gambar 13.3 Interface yang dapat dimodifikasi



Output Sistem Pakar

Sistem pakar dirancang untuk memberikan saran pemecahan masalah. Pemecahan masalah yang ditawarkan dilengkapi dengan penjelasan. Ada dua jenis penjelasan :

- **Penjelasan atas pertanyaan** - Manajer mungkin memerlukan penjelasan saat sistem pakar melakukan penalaran. Mungkin sistem pakar akan meminta manajer untuk memasukkan sejumlah informasi. Manajer menanyakan mengapa informasi itu diperlukan, dan sistem pakar akan menjelaskannya.
- **Penjelasan atas pemecahan masalah** - Setelah sistem pakar memberikan suatu pemecahan masalah, manajer dapat meminta penjelasan bagaimana pemecahan masalah tersebut didapat. Sistem pakar selanjutnya akan menampilkan langkah-langkah penalaran yang menuju kepada pemecahan masalah tersebut.

Walau cara kerja sistem pakar mungkin rumit, *user interface*nya sangat mudah dioperasikan. Para manajer yang biasa menggunakan komputer tidak akan menemui kesulitan menggunakannya.

13.3 Database Pengetahuan (Knowledge Base)

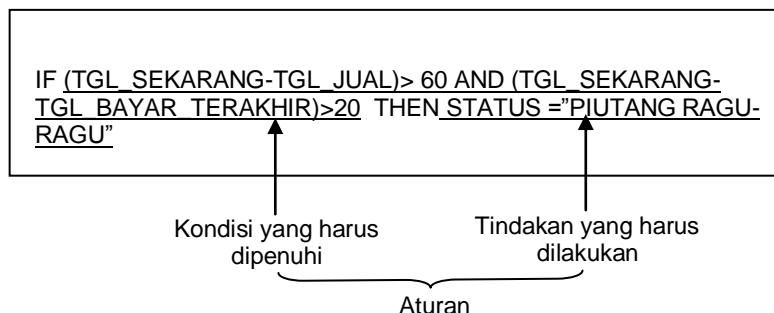
Knowledge base berisi fakta-fakta yang menggambarkan area permasalahan dan teknik bagaimana fakta-fakta tersebut dapat sesuai atau cocok satu sama lain secara logis. Disini digunakan istilah problem domain untuk menggambarkan area permasalahan.

Knowledge Base berisi fakta-fakta yang menggambarkan area permasalahan dan teknik bagaimana fakta-fakta tersebut dapat sesuai atau cocok satu sama lain secara logis

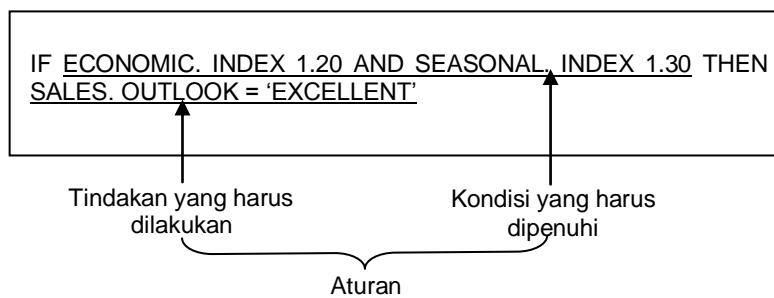
Aturan

Teknik penyajian pengetahuan yang paling populer (*knowledge representation technique*) yang paling populer adalah dengan menggunakan aturan. Aturan menentukan apa yang harus dilakukan dalam situasi tertentu, dan terdiri dari dua bagian: yaitu kondisi yang mungkin benar mungkin tidak dan tindakan yang harus diambil jika kondisinya benar. Contoh aturan adalah :

Gambar 13.4 Kondisi dan perintah



Gambar 13.5 Contoh aturan yang lain



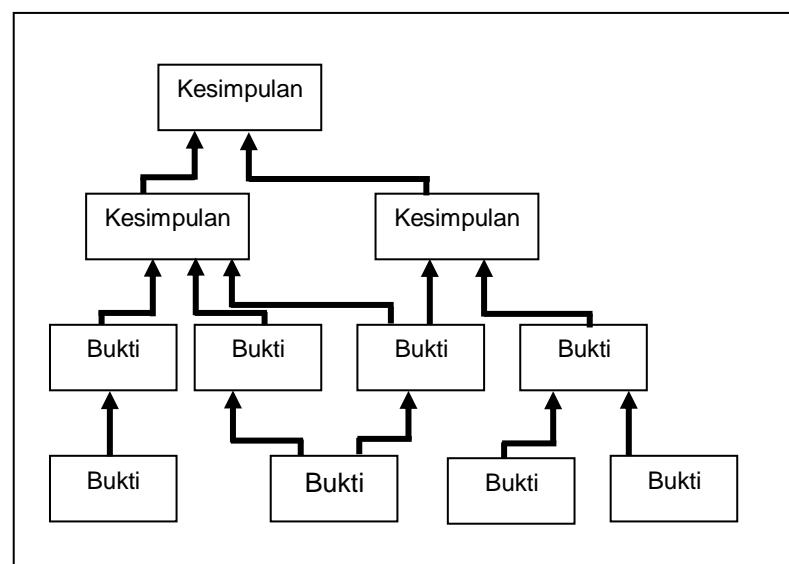
Semua aturan yang ada di dalam sistem pakar tersebut disebut perangkat aturan (rule set). Perangkat aturan dapat bervariasi tergantung kepada kompleksnya permasalahan.

Jaringan Aturan

Aturan yang ada dalam suatu perangkat aturan masing-masing tidak terkait secara fisik, tapi memiliki hubungan logis yang dapat digambarkan dengan diagram hirarki seperti terlihat pada pada gambar 13.6. Aturan pada bagian bawah hirarki menyediakan bukti bagi aturan pada tingkat atas. Bukti tersebut memungkinkan aturan pada tingkat atas untuk menghasilkan kesimpulan.

Tingkat atas mungkin hanya memiliki satu kesimpulan seperti tampak pada gambar 13.6. Istilah variabel sasaran (*goal variabel*) digunakan untuk menunjukkan suatu solusi. Dimana solusi tersebut dapat berupa nilai perhitungan, suatu objek, tindakan yang harus dilakukan, atau suatu rekomendasi yang lainnya.

Gambar 13.6 Perangkat aturan yang menghasilkan kesimpulan akhir



Sebagai contoh, jika sistem pakar menasehati manajer puncak apakah akan memasuki bidang baru?. Jawabnya hanya satu nilai: Ya atau Tidak harus dimasukan kedalam variabel tunggal MARKET . DECISION.

Hirarki pada tingkat atas mungkin pula berbentuk sistem ganda yang memberikan lebih dari satu kemungkinan solusi. Sebagai contoh misalnya sistem pakar membuat rekomendasi mengenai strategi terbaik yang perlu diikuti dalam menghadapi persaingan yang terus meningkat. Sistem pakar mungkin memberikan pilihan diantara berbagai kemungkinan strategi seperti meningkatkan kualitas produk perusahaan, lebih banyak memasang iklan, atau menurunkan harga.

Memilih Aturan

Kesulitan utama dalam menggunakan aturan yang menggambarkan pengetahuan adalah memilih aturan-aturan itu dari dari *database* pengetahuan (*knowledge base*) secara efisien. Sering terjadi pemecahan masalah hanya memerlukan satu subset aturan dari aturan secara keseluruhan.

13.4 Penalar (Inference Engine)

Inference engine adalah bagian dari sistem pakar yang melakukan penalaran terhadap isi *database* pengetahuan (*knowledge base*) berdasarkan urutan tertentu.

Selama konsultasi, *inference engine* menguji aturan-aturan yang ada dalam *database* pengetahuan (*knowledge base*) satu demi satu dan ketika kondisi tertentu terpenuhi maka tindakan tertentu diambil. Ada dua cara penalaran yang dilakukan oleh penalar (*inference engine*) untuk menguji suatu aturan, yaitu: penalaran maju (*forward reasoning*) dan penalaran mundur (*reverse reasoning*).

Inference engine adalah bagian dari sistem pakar yang melakukan penalaran terhadap isi *database* pengetahuan (*knowledge base*) berdasarkan urutan tertentu

Penalaran Maju

Penalaran maju di sebut juga sebagai *forward chaining*. Pada penalaran ini aturan-aturan diuji satu demi satu dalam urutan tertentu. Caranya dilakukan dengan memasukan suatu aturan kedalam seperangkat aturan berdasarkan urutan tertentu, atau dapat juga suatu urutan lain yang di tentukan oleh pemakai. Ketika satu aturan diuji, sistem pakar berusaha mengevaluasi apakah kondisinya benar atau salah .

Evaluasi Aturan

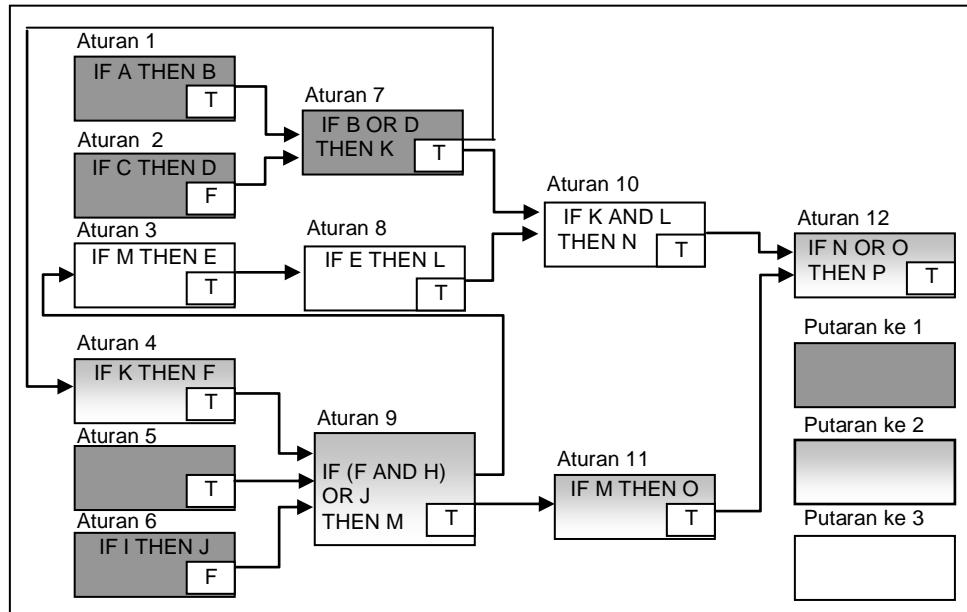
Jika kondisinya benar suatu aturan akan ditampilkan dan aturan berikutnya diuji. Saat kondisinya salah aturan tersebut tidak ditampilkan dan aturan berikutnya diuji. Suatu aturan bisa saja tidak dapat dievaluasi benar atau salahnya karena kondisinya mencakup satu atau lebih variabel yang tidak memiliki nilai akibatnya aturannya tidak diketahui dan jika aturannya tidak diketahui maka aturan tersebut tidak ditampilkan dan aturan berikutnya diuji .

Proses Penalaran Berulang (Iterative)

Proses pengujian aturan dilakukan satu demi satu seterusnya sampai seluruh set aturan yang ada dalam *database* pengetahuan terbaca. Sering kali diperlukan lebih dari satu kali putaran (looping) pembacaan *data* pengetahuan untuk memberikan suatu

nilai tertentu pada variabel sasaran. Mungkin informasi yang diperlukan untuk mengevaluasi satu aturan tertentu diperoleh dari aturan lain yang diuji kemudian. Sebagai contoh, setelah aturan kesebelas ditampilkan misalnya, aturan kelima dapat dievaluasi kembali pada putaran berikutnya. Pembacaan aturan ini akan terus berlanjut selama masih mungkin ada aturan yang dapat ditampilkan. Ketika tidak ada lagi aturan yang dapat ditampilkan, proses penalaran akan berhenti. Sebagai contoh penalaran maju dapat dilihat pada gambar 13.7 dibawah ini.

Gambar 13.7 Proses penalaran maju



Segi empat menunjukkan aturan. Garis yang menghubungkan menggambarkan ketergantungan secara logis. Sebagai contoh aturan empat tidak dapat muncul sampai aturan tujuh muncul. Huruf-huruf digunakan untuk menyederhanakan kondisi dan tindakan. Pada aturan 1, misalnya jika kondisi A terpenuhi, maka tindakan B dilakukan. Kondisi A misalnya PENJUALAN.TAHUN.INI>PENJUALAN.TAHUN. LALU dan TINDAKAN B mungkin PASAR='MENINGKAT'. begitu, pula dalam aturan 2 ,jika kondisi D ada, tindakan D diambil.

Perlu anda catat bahwa sebagian kondisi hanya memiliki satu variabel dan yang lainnya memiliki dua. Ketika suatu kondisi ditentukan oleh beberapa variabel maka hubungan antar variabel tersebut dapat menggunakan kata AND atau OR. Pada aturan 7, jika salah satu kondisi B atau (OR) D terpenuhi maka aturan tersebut akan muncul. Kalau dilihat pada aturan 10, kondisi K dan (AND) L harus keduanya harus terpenuhi agar aturan bisa dimunculkan.

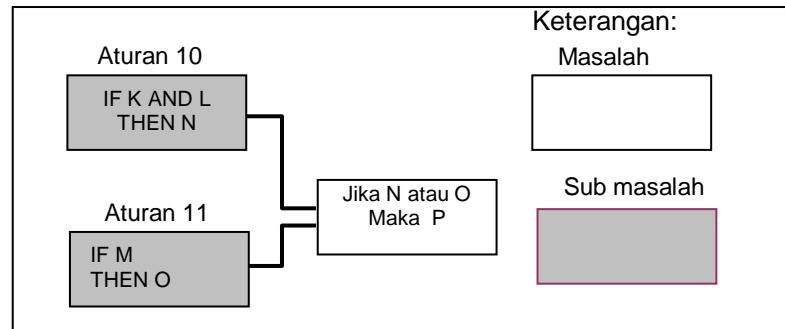
Sistem pakar ini merupakan contoh bagaimana menghitung ramalan penjualan (P) pada aturan 12, dimana P disini berperan sebagai variabel sasaran. Kita asumsikan bahwa manajer memberikan nilai bagi variabel kondisi A, C, G, dan I seperti terlihat pada kotak segi empat, pada aturan 1, 2, 5, dan 6 sebelum aturan-aturan tersebut diproses (dievaluasi) oleh penalar (*inference engine*). Manajer memberikan suatu nilai benar untuk variabel A pada aturan 1, nilai salah untuk variabel C pada aturan 2 nilai benar untuk variabel G pada aturan 5, dan nilai salah untuk variabel I pada aturan 6.

Keterangan yang muncul menunjukkan aturan yang dapat di evaluasi pertama kali pada putaran pertama. Keterangannya bisa benar atau salah. Disini aturan muncul selama tiga kali, dimana huruf T (benar) dan F(salah) yang muncul merupakan hasil evaluasi. Pada putaran ke empat, tidak ada aturan yang muncul dan penalaran berhenti. Jika aturan 12 muncul, nilai yang muncul pada variabel sasaran P menunjukkan ramalan penjualan. Jika sistem pakar tidak dapat menampilkan aturan 12 ini berarti tidak cukup informasi untuk memecahkan masalah. Pada contoh ini, aturan 12 muncul baik untuk putaran kedua maupun putaran ketiga. Kondisi M dinyatakan benar pada putaran kedua, sedangkan kondisi K dan L dinyatakan benar pada putaran ketiga. Anda mungkin bertanya mengapa penalaran tidak berhenti pada saat pertama kali nilai pada variabel sasaran muncul. Jawabannya adalah kemunculan selanjutnya mungkin dapat memberikan solusi lebih baik.

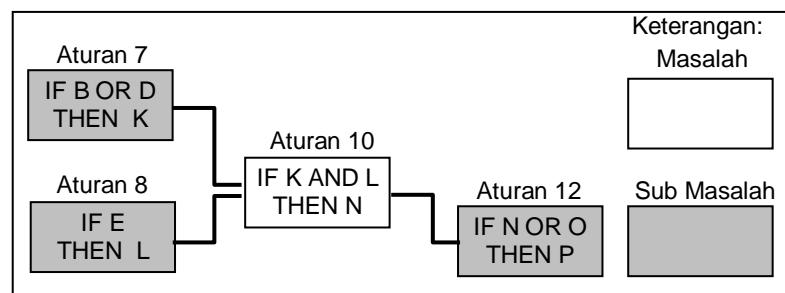
Penalaran Mundur

Penalaran ini disebut sebagai *backward chaining*, penalar akan memilih aturan dan menganggapnya sebagai masalah yang harus dipecahkan. Dengan menggunakan perangkat aturan yang sama seperti penjelasan sebelumnya, aturan 12 merupakan objek permasalahan, karena aturan itu memberikan nilai pada variabel sasaran P. Penalar kemudian mengevaluasi aturan 12 tapi menyadari bahwa aturan 10 atau aturan 11 harus dievaluasi terlebih dahulu. Karena itu, aturan 10 dan 11 menjadi submasalah aturan 12 seperti tampak pada gambar 13.7. Penalar kemudian memilih salah satu dari submasalah untuk dievaluasi, dan submasalah yang terpilih tersebut kemudian menjadi masalah baru.

Gambar 13.8 Masalah dan submasalah



Gambar 13.9 Submasalah menjadi masalah baru



Membandingkan Penalaran Maju dan Mundur

Proses penalaran mundur lebih cepat dari penalaran maju karena penalaran mundur selain tidak harus menguji semua aturan secara berulang-ulang. Penalaran mundur sangat sesuai jika:

- Terdapat variabel sasaran berganda (*multiple goal variables*)
- Terdapat banyak aturan.
- Semua atau hampir semua aturan tidak harus diuji dalam proses mencapai pemecahan

Berapa penalaran dirancang untuk melakukan penalaran baik penalaran maju ataupun mundur dan pemakai bisa memilih mana yang akan digunakan .

Bagaimana Penalar Menangani Ketidakpastian

Tidak pernah ada kepastian 100 persen dari informasi yang dihasilkan oleh sistem pakar. Ketidak pastian itu berlaku untuk seluruh aturan yang ada dalam sistem pakar. Sistem pakar menggunakan faktor kepastian (*certainty factor*), atau CF, untuk menunjukkan derajat ketidak pastian. CF menyerupai probabilitas dan berkisar

antara 0 yang berarti tidak pasti sampai dengan 100 yang berarti pasti. Sistem pakar mengingatkan pemakainya terhadap kesimpulan yang memberikan kepastian kurang dari 100%. Sebagai contoh misalnya ada empat aturan yang mengarah kesatu kesimpulan, tiga diantaranya memberikan kepastian 100%, namun aturan keempat memberikan CF 80, ini berarti bahwa kepastian terhadap tindakan tersebut adalah 80%. Sistem pakar menelusuri berbagai aturan CF sepanjang proses penalaran dan menunjukkan bahwa derajat kepastian bagi variabel sasaran pada akhir konsultasi, sebagai berikut:

Target Penjualan = Rp 12,450,500.000, CF = 0.80

Contoh diatas hanyalah salah satu cara untuk menangani ketidakpastian. Para manajer bisa menggunakan cara lain yang lebih cocok.

13.5 Bahasa Pemrograman Sistem Pakar (Development Engine)

Komponen utama keempat dari sistem pakar adalah *software* pembuat sistem pakar, yang digunakan untuk menciptakan sistem pakar. Pada dasarnya proses ini meliputi pembuatan seperangkat aturan yang dapat dilakukan dengan dua pendekatan dasar: Menggunakan bahasa pemrograman dan *shells* sistem pakar.

Bahasa Pemrograman

Sistem pakar dapat dibuat dengan menggunakan bahasa pemrograman apa pun, namun yang paling banyak digunakan adalah Lisp dan Prolog. Lisp dikembangkan pada tahun 1959 oleh John McCarthy, dan pengembangan prolog dimulai oleh Alain Colmerauer pada University of Marseilles tahun 1972.

Untuk beberapa tahun, Lisp mengalami popularitas yang besar di Amerika, dan prolog dipilih oleh pemakai Eropa dan Jepang. Pemilihan secara geografis ini tidak berlangsung lama. Prolog akhirnya memperoleh dukungan dari masyarakat Amerika, sedangkan LIPS cukup banyak dipakai di Eropa.

Sistem pakar dapat dibuat dengan menggunakan bahasa pemrograman apa pun, namun yang paling banyak digunakan adalah *Lisp* dan *Prolog*

Shells Sistem Pakar

Salah satu sistem pakar pertama kali adalah Mycin yang dibuat oleh Edward Shortliffe dan Stanley Cohen dari Universitas Stanford dengan bantuan seorang ahli kedoktoran bernama Stanton Axline, Mycin dibuat saat itu untuk mendiagnosa penyakit menular tertentu.

Ketika Mycin berhasil mencapai sukses, para pengembang mencari alternatif lain untuk mengimplementasikannya. Mereka menemukan bahwa penalar pada Mycin dapat digunakan untuk menangani masalah lain dengan mengganti *database* pengetahuan (*knowledge base*) Mycin. Dengan *database* pengetahuan lain Mycin dapat menangani masalah lain. Temuan ini melahirkan era baru dalam membangun sistem pakar dan ditandai dengan dikembangkannya *shell* sistem pakar dalam bentuk prosesor siap pakai yang telah diisi dengan *database* pengetahuan dan disesuaikan dengan. Dengan teknologi ini *shell* dapat memproduksi sistem pakar lebih cepat dan lebih mudah dioperasikan daripada pemrograman.

Saat ini kebanyakan aplikasi dari sistem pakar untuk masalah bisnis menggunakan *shell*. *Shell* komersil pertama adalah KEE yang merupakan kependekan dari *Knowledge Engineering Environment*. KEE dirancang untuk digunakan pada komputer yang menggunakan bahasa mesin *Lips*. Namun dengan diperkenalkannya *processor* yang lebih cepat saat ini menyebabkan sebagian besar pengembangan berpindah dari mesin khusus *Lips* ke *platform* yang baru, dan pengembangan pun mulai bercabang ke komputer yang memiliki prosesor banyak.

Peranan Sistem Analis

Knowledge engineer (seperti sistem analis) adalah orang yang merancang sistem pakar bersama dengan satu atau beberapa orang pakar

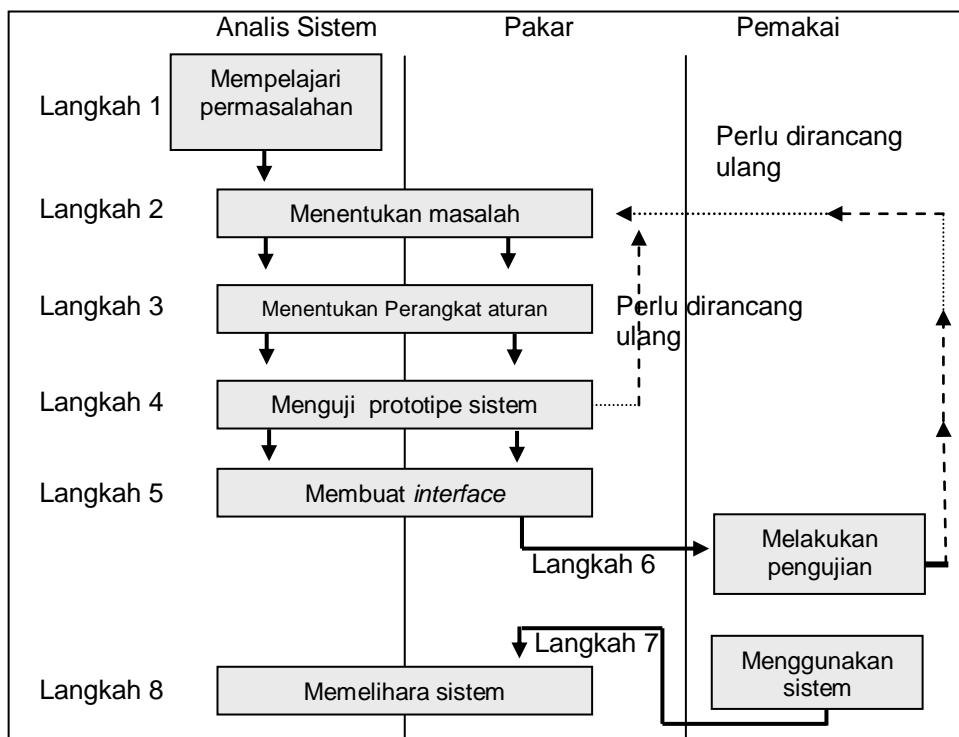
Knowledge engineer (seperti sistem analis) adalah orang yang merancang sistem pakar bersama dengan satu atau beberapa orang pakar. Orang ini harus :

- Mengerti bagaimana seorang pakar menerapkan pengetahuan mereka dalam memecahkan masalah.
- Mampu mencerna penjelasan mengenai pengetahuan yang diberikan oleh seorang pakar. Proses mencerna keahlian disebut *knowledge acquisition* dengan berbagai pendekatannya.

Pengembangan Sistem Pakar Dengan Menggunakan Metode Prototyping

Penjelasan tentang metode prototyping akan dijelaskan lebih detail pada bab 16. Dalam pengembangan sistem informasi manajemen biasanya dilakukan oleh sistem analis dan para manajer suatu perusahaan. Dalam pengembangan sistem pakar melibatkan pihak ketiga yaitu seorang pakar. Gambar 13.10 menggambarkan bagaimana analisis sistem bekerja sama dengan pakar dan partisipasi dari pemakai mengembangkan model sistem pakar.

Gambar 13.10 Metode prototyping dalam pengembangan sistem pakar



- **Mempelajari permasalahan** - Analis sistem mempelajari bidang permasalahan dan menentukan apakah pekerjaan pengembangan akan dilanjutkan. Jika ya, analisis harus memiliki cukup pengetahuan tentang bidang permasalahan tersebut.
- **Menentukan permasalahan** - Analis sistem dan pakar bersama-sama menentukan permasalahan yang harus dipecahkan.
- **Tentukan perangkat aturan** - Analis sistem dapat bertanya kepada pakar tahap-tahapan bagaimana memecahkan suatu masalah. Tahapan-tahapan tersebut akan merupakan dasar untuk membuat aturan.
- **Uji sistem prototype** - Prototype tergantung kepada test yang dilakukan. Tahap dua sampai empat terus berulang sampai dirasakan cukup.
- **Membuat *interface*** - Ketika developer merasa bahwa sistem telah bekerja sesuai dengan yang diharapkan, tahap selanjutnya adalah menambahkan *interface*.

- **Lakukan test dengan user** - test pertama yang dilakukan dengan *user* adalah untuk mendidik *user* bagaimana memasukkan *input* dan menginterpretasikan *output*. Ketika *user* merasa *familier* dengan sistem. Kalau hasil test tidak sukses maka tahap dua sampai enam diulang.
- **Penggunaan sistem** - Bila test dengan pemakai berjalan sukses maka sistem pakar dapat dipakai untuk aktivitas sehari-hari.
- **Memelihara sistem pakar** - Seperti halnya semua aplikasi komputer perlu dijaga agar sistem komputer tersebut selalu *up-to-date* maka hal tersebut berlaku juga untuk sistem pakar.

Pengembangan sistem pakar biasanya lebih lama dibandingkan dengan pengembangan aplikasi komputer lain. Walaupun terjadi tapi seharusnya tidak terjadi menghabiskan waktu dua atau tiga tahun hanya untuk menentukan perangkat aturan.

Keuntungan Sistem Pakar Bagi Manajer

Dengan sistem pakar diharapkan manajer dapat membuat keputusan lebih baik

Dengan sistem pakar diharapkan manajer dapat membuat keputusan lebih baik karena memiliki lebih banyak alternatif dengan tingkat logika yang lebih tinggi dan memiliki lebih banyak waktu untuk mengevaluasi sehingga dapat diperoleh konsistensi untuk keputusan yang dicapai.

Contohnya, manajer keuangan yang biasanya hanya mampu menelusuri kenerja tiga puluh saham karena banyaknya *data* yang harus dipertimbangkan dengan bantuan sistem pakar dapat menelusuri 300 saham. Dengan kemampuan mempertimbangkan lebih banyak maka peluang investasi dan kemungkinan untuk memilih alternatif terbaik meningkat. Penerapan logika yang lebih tinggi bagi manajer yang menggunakan sistem pakar, manajer seperti menerapkan logika seorang pakar yang sangat ahli dibidangnya.

Dengan memperoleh nasehat dari sistem pakar secara lebih cepat maka akan lebih banyak waktu yang tersedia untuk menimbang kemungkinan hasil sebelum tindakan dilakukan. Manajer akan mendapatkan solusi konsisten untuk setiap masalah sama yang dihadapi karena Setelah penalaran diprogram dalam komputer.

Keuntungan Sistem Pakar Bagi Perusahaan

Harapan dalam penerapan sistem pakar:

- **Kinerja perusahaan dan pengendalian yang lebih baik** karena para manajer memiliki kemampuan yang lebih luas dalam memecahkan masalah.

- **Mempertahankan pengetahuan perusahaan** yang selama ini dimiliki sehingga tersedia untuk pegawai yang baru atau kurang berpengalaman

Hal yang Tidak Menguntungkan dari Sistem Pakar

Ada dua ciri sistem pakar yang membatasi kemampuannya:

- Sistem pakar tidak dapat menangani pengetahuan yang tidak konsisten.
- Sistem pakar tidak dapat menerapkan penilaian dan intuisi yang kita kenal sebagai unsur penting dari pemecahan masalah saat kita membahasnya dalam pendekatan sistem pada Bab 2.

Kendala besar dalam menerapkan sistem pakar dalam masalah bisnis adalah struktur masalahnya. Agar sistem pakar dapat digunakan, masalah harus sangat terstruktur dan kita melihat bahwa sebagian besar masalah bisnis cocok dengan bentuk ini.

Rangkuman

Aktifitas didalam bidang kecerdasan buatan (artificial intelligence) atau AI, adalah aktifitas penyediaan mesin seperti komputer yang memiliki kemampuan menampilkan perilaku yang dianggap cerdas seperti yang terjadi pada manusia.

Penerapan AI dalam bidang bisnis dikenal sebagai sistem pakar (expert system). Sistem pakar adalah program komputer yang beroperasi menyerupai pengetahuan seorang pakar dalam bentuk 'heuristic'. Berbagai aturan yang ada didalam sistem pakar memungkinkan sistem ini untuk berfungsi secara konsisten seperti seorang pakar (ahli).

Konsep sistem pakar didasarkan pada asumsi bahwa pengetahuan pakar dapat diambil atau disimpan dalam komputer dan kemudian digunakan oleh orang lain saat dibutuhkan.

Knowledge engineer (seperti sistem analis) adalah orang yang merancang sistem pakar bersama dengan satu atau beberapa orang pakar. Metode pengembangan yang paling sesuai untuk sistem pakar adalah dengan menggunakan metode prototyping, dan disini akan terlibat pihak ketiga yaitu pakar.

Harapan dalam penerapan sistem pakar adalah kinerja perusahaan dan pengendalian yang lebih baik dan mempertahankan pengetahuan perusahaan yang selama ini dimiliki. Yang tidak menguntungkan adalah sistem pakar tidak dapat menangani pengetahuan yang tidak konsisten dan sistem pakar tidak dapat menerapkan penilaian dan intuisi yang kita kenal sebagai unsur penting dari pemecahan masalah.

Soal

1. Apakan yang dimaksud dengan kecerdasan buatan (AI)?
2. Apakah yang dimaksud dengan sistem pakar?
3. Sebutkan cabang kecerdasan buatan selain sistem pakar?
4. Sebutkan empat bagian dari model sistem pakar?
5. Sebutkan keuntungan dan hal yang tidak menguntungkan dari sistem pakar

Tugas

1. Kosep sistem pakar sudah lama muncul, mengapa perkembangannya belum begitu pesat dalam dunia bisniss?
2. Buat dan jelaskanlah model sistem pakar? Mengapa perlu problem domain?
3. Jelaskan yang dimaksud dengan penalaran maju dan mundur?
4. Apa yang dimaksud dengan knowledge base apa bedanya dengan database?
5. Konsep sistem pakar sudah lama muncul, mengapa perkembangannya belum begitu banyak dalam aplikasi di dunia bisnis?