

REKAYASA SISTEM KOGNITIF BERBASIS MULTI-AGEN: PENDEKATAN PENALARAN BERBASIS KASUS

Nur Ulfa Maulidevi⁽¹⁾, dkk

Abstract: Cognitive system modeling first introduced by psychology researchers. Unfortunately, the model has not been sufficient in supporting computer based problem solving. For that reason, artificial intelligence tries to propose a computational model of cognitive system. The main purpose of the computational model is to support human in solving complex problems, especially problems that involve large number of data, uncompleted data, and problem solving that requires systematic approach as human does. This research proposes an engineering of such multiagent based cognitive system, which employs case based reasoning as imitation of human reasoning to maintain the knowledge base.

Keywords: Cognitive System, Engineering, Multiagent, Case-based Reasoning, Knowledge

Rekayasa Sistem Kognitif atau dalam bahasa Inggris dikenal dengan istilah *Cognitive System Engineering*, memiliki dua pendekatan yang berbeda. Pendekatan pertama adalah bagaimana membuat sebuah aplikasi perangkat lunak (*software*) dengan memanfaatkan ilmu kognitif pada setiap tahap rekayasanya (Kushwana, 2006a, 2006b; Roth, 2002), sehingga sistem yang dibuat dapat digunakan dengan interaksi yang memudahkan pengguna dan tujuan dibangunnya sistem tersebut dapat dicapai dengan optimal. Pendekatan ke dua adalah bagaimana melakukan proses rekayasa untuk suatu sistem kognitif, yaitu sistem yang memiliki kemampuan kognitif. Pendekatan ke dua belum banyak dibahas dalam penelitian dengan topik rekayasa sistem

kognitif. Makalah ini mengusulkan suatu rekayasa untuk sistem dengan kemampuan kognitif.

Dasar dari sistem kognitif adalah hasil penelitian dalam bidang psikologi kognitif. Terdapat empat besar pendekatan untuk memahami *cognitive psychology*, namun untuk penelitian ini pendekatan yang digunakan adalah pendekatan dari *cognitive science*, yang memanfaatkan model komputasi untuk memahami proses kognisi manusia (Eyesenck, 2003). Dalam psikologi kognitif modern, makna psikologi kognitif adalah mencari analogi untuk memahami cara kerja otak.

Analogi yang paling mendekati cara kerja otak adalah (Eyesenck, 2003; Parkin, 2000) memandang otak sebagai suatu sistem pemroses informasi

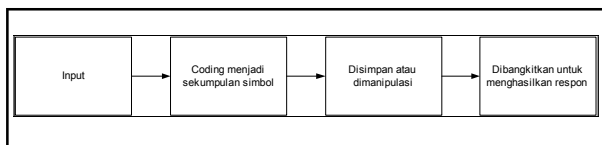
⁽¹⁾ Nur Ulfa Maulidevi, Program Studi Teknik Elektro, Sekolah Teknik Elektro dan Informatika, Insitut Teknologi Bandung

⁽²⁾ Jaka Sembiring, Program Studi Teknik Elektro, Sekolah Teknik Elektro dan Informatika, Insitut Teknologi Bandung

⁽³⁾ Hana Wijaya, Fakultas Psikologi, Universitas Padjadjaran Bandung

⁽⁴⁾ Adang Suwandi Ahmad, Program Studi Teknik Elektro, Sekolah Teknik Elektro dan Informatika, Insitut Teknologi Bandung

(*information processing system*). Dalam analogi ini dikenal dua istilah yang penting yaitu *coding* dan *channel capacity* (Parkin, 2000). *Coding* artinya adalah simbol-simbol dari stimulus, dan *channel capacity* diartikan sebagai kapasitas ruang. Kapasitas ini penting karena manusia menerima informasi dalam jumlah atau ruang yang terbatas. Proses yang terjadi dalam otak manusia dengan analogi ini dapat dilihat pada Gambar 1. Dengan pemikiran bahwa cara kerja otak manusia adalah suatu sistem pemroses informasi, maka psikologi kognitif memiliki banyak kesamaan dengan cara kerja komputer. Proses yang diacu di sini adalah proses antara munculnya stimuli hingga menghasilkan suatu respon.



Gambar 1 Proses Berpikir Manusia

Dipertegas dalam (Konar, 2000), kognisi diartikan sebagai kemampuan dari aktivitas mental yang berhubungan dengan abstraksi informasi dari skenario dunia nyata (*real world*), representasinya, serta penyimpanannya dalam memori untuk kemudian dapat diambil dari memori secara otomatis. Hal yang penting di sini adalah persepsi, sebagai suatu bentuk informasi/pengetahuan dalam *level* rendah yang ditangkap dari lingkungan. Berhubungan dengan otak manusia, media *input* adalah sensor manusia yaitu panca indra, untuk kemudian memerlukan integrasi dari memori untuk memahami dan menalar sesuai konteks dunia nyata.

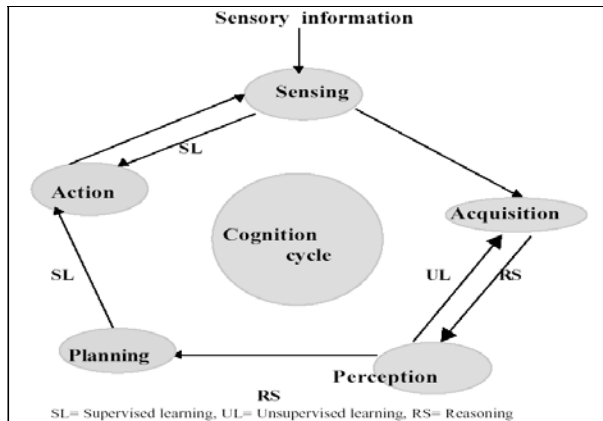
Dalam melakukan proses kognisi, terdapat status-status (*states*), di mana tiap status adalah kemampuan otak yang dikatakan 'cerdas'. Sub proses

yang terlibat dalam siklus kognisi adalah (Konar, 2000):

- a. Tanggapan (*Sensing*)
Status ini mengacu pada penerimaan, transformasi sinyal, *pre-processing* dan ekstraksi fitur dari informasi yang diterima.
- b. Akuisisi
Status ini melakukan perbandingan antara respon pada *Short Term Memory (STM)* dengan informasi yang sudah tersimpan pada *Long Term Memory (LTM)*. Informasi, yang nantinya disebut dengan pengetahuan, yang tersimpan pada *LTM* dapat berubah sewaktu-waktu, sesuai dengan pembelajaran yang dilakukan oleh manusia.
- c. Persepsi
Status ini melakukan proses representasi pengetahuan sedemikian sehingga saat diperlukan, otak dapat dengan cepat mengakses informasi atau pengetahuan yang diinginkan dari *LTM*. Model komputasi yang banyak digunakan untuk status ini adalah *semantic net*.
- d. Perencanaan (*planning*)
Status ini menentukan langkah-langkah yang perlu dilakukan dari tahap awal hingga mencapai tujuan yang diinginkan. Status ini juga mencari pengetahuan yang sesuai dengan persoalan yang ingin diselesaikan berdasarkan masukan dari lingkungan. Status ini hanya menentukan langkah-langkah, namun tidak sampai pada eksekusi dari langkah-langkah tersebut.
- e. Aksi
Berdasarkan urutan langkah-langkah yang diberikan pada status sebelumnya, status ini melakukan eksekusi sesuai dengan jadwal yang ditentukan. Status ini memanfaatkan *supervised learning* karena berusaha memperbaiki aksi

berikutnya sesuai dengan rangsangan *input* dan juga respon dari lingkungan.

Keterhubungan dari kelima status mental tersebut dapat dilihat pada Gambar 2.



Gambar 2 Status Mental dan Keterhubungan pada Proses Kognisi (Konar, 2000)

Salah satu perkembangan dalam *Distributed Artificial Intelligence* adalah sistem multiagen. Kumpulan agen cerdas, bekerja sama mengorganisasikan diri sendiri untuk mencapai tujuan sistem. Karakteristik dari sistem multiagen adalah: (1) agen memiliki keterbatasan informasi dan kemampuan dalam menyelesaikan suatu persoalan; (2) tidak ada kontrol global dalam sistem tersebut; (3) data bersifat desentralisasi; dan (4) komputasi bersifat asinkronous. Berdasarkan karakteristik tersebut, sistem multiagen sesuai digunakan untuk persoalan yang cukup kompleks, sehingga pemanfaatan agen tunggal akan meningkatkan resiko keterbatasan sumberdaya. Adanya kegagalan satu bagian akan berakibat fatal pada seluruh sistem. Keunggulan lain dalam pemanfaatan sistem multiagen adalah dalam hal efisiensi komputasi, reliabilitas, ekstensibilitas, ketangguhan, responsif, fleksibel, dan kemudahan dalam perawatan (Sycara, 1998; Weiss, 1999). Salah satu sistem yang cukup kompleks adalah sistem kognitif, karena sistem

tersebut merupakan suatu tiruan sistem pemrosesan informasi yang didalamnya terdapat tanggung jawab untuk melakukan penangkapan informasi dari lingkungan (persepsi), pembelajaran, penalaran, pembuatan keputusan, komunikasi, dan melakukan aksi.

Dengan model yang sedang berkembang saat ini, persoalan yang muncul adalah bagaimana memodelkan rekayasa suatu sistem kognitif berbasis multiagen, sehingga pengetahuan dapat berkembang dan dapat bersifat reaktif terhadap persepsi yang ditangkap dari lingkungan. Artinya, sistem dapat menyelesaikan persoalan secara sistematis sebagaimana manusia menyelesaikan persoalan. Pengetahuan menjadi kunci dalam sistem kognitif, termasuk di dalamnya adalah bagaimana pengetahuan dapat berkembang sesuai dengan persepsi lingkungan.

Tujuan penelitian ini adalah mengusulkan suatu bentuk rekayasa sistem kognitif, yaitu rekayasa untuk sistem yang mampu mengorganisasikan dirinya sendiri dalam rangka mengembangkan pengetahuannya, dengan penekanan pada penalaran berbasis kasus. Pengetahuan tersebut pada proses selanjutnya dijadikan acuan dalam penyelesaian persoalan dan pengambilan keputusan.

METODE

Kaitan yang sangat erat antara disiplin psikologi dengan kecerdasan tiruan (Haugeland, 1997; Konar, 2000; Satchl, 2006), membuat pemodelan komputasi untuk cara kerja otak banyak dilakukan dalam disiplin kecerdasan tiruan. Berdasarkan kelima status mental tersebut, pendekatan dari kecerdasan tiruan untuk membuat model komputasi dari sistem kognitif dibuat.

Dalam kecerdasan tiruan, ketiga tahap awal yaitu *sensing*, *acquisition*, dan *perception* umumnya

tergabung sebagai satu tahapan awal yang menerima rangsangan/*input*/persepsi dari lingkungan. Namun, status *perception* dalam kecerdasan tiruan terbagi menjadi dua bagian besar. Sebagian awal termasuk dalam pemrosesan rangsangan/*input* dari lingkungan dan dikenal dengan istilah *perception*, dan sebagian lain adalah proses untuk merepresentasikan *input* tersebut dalam suatu pengetahuan yang tersimpan pada basis pengetahuan (*LTM*). Pengetahuan tersebut dapat berubah sejalan dengan waktu karena manusia atau sistem pada kecerdasan tiruan melakukan pembelajaran secara terus menerus.

Tahapan selanjutnya adalah pemanfaatan pengetahuan yang ada untuk penyelesaian suatu persoalan. Dengan melakukan penalaran (*reasoning*) dapat ditentukan pengetahuan mana yang dapat digunakan untuk menyelesaikan persoalan, dan berdasarkan pengetahuan tersebut ditentukan satu atau urutan aksi untuk menyelesaikan persoalan.

Salah satu model komputasi untuk meniru penalaran manusia adalah penalaran berbasis kasus (Pal, 2004) atau dikenal juga dengan istilah *Case-Based Reasoning (CBR)*. Proses yang terjadi pada *CBR* adalah sebagai berikut. Persoalan diberikan kepada *CBR*, kemudian *case-based reasoner* mencari kasus-kasus yang ada pada basis kasus untuk menemukan kasus yang memiliki kemiripan dengan persoalan yang sedang dihadapi (*retrieve*). Jika kasus yang ditemukan sama persis dengan persoalan yang sedang dihadapi, maka *reasoner* akan mengembalikan solusi dari kasus tersebut sebagai jawaban dari persoalan (*reuse*). Jika kasus yang ditemukan tidak sama persis, maka proses adaptasi berlangsung (*revise*). Proses adaptasi dilakukan dengan mengidentifikasi perbedaan dari kasus yang ditemukan dengan persoalan yang dihadapi, kemudian

hasil identifikasi tersebut digunakan untuk melakukan modifikasi terhadap solusi kasus yang ditemukan. Persoalan yang dihadapi dan hasil modifikasi solusinya kemudian ditambahkan pada basis kasus setelah divalidasi (*retain*). Oleh karena itu, proses *CBR* sering juga disebut dengan "4 re-". Pendekatan inilah yang dimanfaatkan dalam mengelola basis pengetahuan untuk sistem kognitif berbasis multiagen.

Hasil penelitian dari pakar bidang psikologi dimanfaatkan untuk membuat model komputasi siklus kognisi pada kecerdasan tiruan (Haugeland, 1997; Mitchell, 1997; Konar, 2000, 2005; Russel, 2003), dan diimplementasikan pada *intelligent agent* (agen cerdas). Penelitian ini tidak melihat model proses secara rinci dalam suatu sistem kognitif, namun lebih pada suatu sistem secara utuh, dan bagaimana tahapan rekayasa suatu sistem kognitif.

Dalam sistem kognitif, proses kognisi manusia dilakukan secara paralel dan terdistribusi. Artinya, banyak hal yang dilakukan otak manusia saat dihadapkan pada suatu persoalan, dan manusia mampu melakukan beberapa *tasks* dalam otak secara paralel untuk kemudian disatukan kembali dalam rangka menyelesaikan persoalan yang dihadapi. Artinya, perlu adanya kumpulan agen yang masing-masing melaksanakan suatu *tasks* untuk kemudian dikomunikasikan hasil pengerjaan *task* tersebut, dan hal tersebut digunakan dalam rangka mencapai tujuan utama atau penyelesaian persoalan yang dihadapi (Koning, 2003).

Berdasarkan analogi tersebut, penelitian ini berusaha mengajukan model rekayasa sistem kognitif yang tidak lagi bersifat *stand-alone system*, tapi merupakan persoalan kompleks, sehingga memerlukan pendekatan penyelesaian untuk sistem yang kompleks. Oleh karena itu, proses rekayasa yang diusulkan

memang terbatas pada suatu sistem yang dapat direpresentasikan sebagai suatu sistem kognitif dan didalamnya memanfaatkan status kognisi seperti halnya manusia dalam menyelesaikan suatu persoalan.

Pada saat berbicara mengenai rekayasa, dalam hal ini adalah rekayasa perangkat lunak, maka terdapat dua hal utama yang perlu diperhatikan dalam membangun suatu perangkat lunak. Dua hal tersebut adalah *modeling language* dan *software process* (Luck, 2004). *Modeling language* adalah deskripsi dari model yang memberikan definisi elemen-elemen yang ada pada model lengkap dengan sintaks tertentu dan makna semantiknya. Sedangkan *software process* mendefinisikan aktivitas pembangunan, keterhubungan antar aktivitas, serta cara-cara aktivitas yang ada dalam proses dilakukan.

Tiga tahapan utama dalam proses rekayasa perangkat lunak adalah analisis, desain, dan implementasi (Luck, 2004). Dalam makalah ini akan diusulkan suatu *software process* untuk sistem kognitif berbasis sistem multiagen, dengan fokus pada ke tiga tahap tersebut yaitu analisis, desain, dan implementasi.

Dalam setiap rekayasa perangkat lunak tentu saja diperlukan analisis kebutuhan dari sistem yang akan dibangun. Dalam pendekatan rekayasa sistem kognitif ini, analisis kebutuhan telah diperoleh sebelum rekayasa sistem kognitif dilakukan. Proses mendapatkan kebutuhan ini di luar cakupan rekayasa sistem kognitif, sehingga rekayasa sistem kognitif dapat menjadi bagian dari suatu rekayasa sistem yang lebih besar. Kebutuhan yang telah didefinisikan sebelumnya akan menjadi masukan utama dalam rekayasa sistem kognitif, karena dari sini dapat ditentukan tujuan (*goals*) dari sistem kognitif. Analisis

kebutuhan dapat dilakukan *on the fly* saat rekayasa sistem kognitif akan dilakukan, baik secara formal mengikuti metodologi tertentu ataupun secara non formal. Satu hal yang penting dan harus dilakukan dalam rangka keberhasilan proses rekayasa adalah, analisis kebutuhan tersebut harus dituliskan dalam suatu dokumen agar dapat diacu ulang. Jika terjadi perubahan maka perubahan dan versi terakhir dokumen kebutuhan tersebut juga didokumentasikan.

HASIL DAN PEMBAHASAN

Usulan Rekayasa Sistem Kognitif

Metode dalam pengembangan aplikasi berbasis agen, memiliki dua sudut pandang. Ke dua sudut pandang tersebut harus dilakukan dalam tahap rekayasa perangkat lunak berbasis agen. Ke dua sudut pandang tersebut adalah sudut pandang eksternal dan sudut pandang internal. Sudut pandang eksternal adalah sudut pandang di luar agen dalam sistem yang akan dibangun. Dalam sistem multiagen sudut pandang ini melakukan tahapan rekayasa terhadap agen yang akan dibentuk mencakup tujuan, tanggung jawab, layanan (*services*) yang diberikan oleh tiap agen dalam sistem, informasi yang diperlukan serta disimpan oleh tiap agen, serta interaksi yang dilakukan dengan agen lain dalam sistem tersebut. Sudut pandang internal adalah sudut pandang yang melihat dalam diri tiap agen, elemen apa saja yang diperlukan mencakup tingkat keyakinan, tujuan, serta perencanaan dalam mencapai tujuan agen tersebut.

Analisis

Dalam tahap analisis terdapat dua sudut pandang yang digunakan, dan masing-masing sudut pandang berupa suatu siklus yang dapat berulang untuk perbaikan analisis sistem.

Sudut Pandang Eksternal

Dalam Sudut Pandang Eksternal, proses rekayasa yang terdapat di dalamnya adalah:

Penentuan Tujuan dari Sistem Kognitif secara Keseluruhan

Penentuan tujuan dari sistem kognitif tidak terlepas dari hasil analisis kebutuhan yang telah dilakukan sebelumnya. Dalam penentuan tujuan perlu ada tingkatan atau hirarki. Hirarki tersebut dimulai dari penentuan tujuan utama sistem, untuk kemudian diturunkan menjadi tujuan spesifik yang harus dicapai, hingga ke tingkat yang cukup rinci yang dapat diselesaikan oleh suatu agen. Tujuan dari suatu *level* dapat menjadi prasyarat dari pencapaian tujuan di *level* yang lain.

Penentuan Peran (dikaitkan dengan status kognisi) Dalam Rangka Mencapai Tujuan Sistem Kognitif

Dari tujuan yang telah didefinisikan di tahap awal, ditentukan peran yang perlu ada dalam sistem kognitif, terkait dengan status kognisi yang telah dipaparkan pada bagian pertama. Namun dalam penentuan peran ini, pemetaan peran dan status kognisi tidak harus berupa pemetaan satu ke satu. Mungkin saja dalam suatu peran meliputi beberapa status kognisi.

Identifikasi Layanan atau Fungsi yang ada pada setiap Peran

Setelah peran yang perlu ada pada sistem kognitif didefinisikan, maka fungsionalitas dan layanan tiap peran tersebut perlu diturunkan. Sebagai contoh, untuk peran yang mencakup status *planning*, fungsi untuk mengambil status awal dan status akhir yang diinginkan harus ada. Setelah itu fungsi untuk membangkitkan *steps* dalam rangka membentuk *plan* juga perlu ada.

Identifikasi Interaksi antar Peran Berdasarkan Layanan yang Diberikan

Setelah setiap peran didefinisikan, maka ditentukan interaksi antar peran dalam sistem kognitif. Dalam tahap ini juga perlu didefinisikan bahwa interaksi antar tahap tertentu akan mencapai *goal* pada *level* tertentu. Dalam setiap interaksi yang didefinisikan, perlu juga ada ketentuan bentuk interaksi, misal bahasa untuk interaksi dan bagaimana format pesan untuk komunikasi yang akan dilakukan. Pada saat ini, analisis dari sudut pandang internal agen mulai dapat dilakukan.

Dalam interaksi ini juga perlu didefinisikan ontologi untuk komunikasi, terutama ketika diperlukan penambahan agen nantinya saat implementasi oleh *provider* yang berbeda dari pengembang awal. Dalam tahap ini, analisis terhadap kebutuhan *fault-tolerance* juga perlu didefinisikan. Salah satu keunggulan kognisi manusia adalah mampu mencari alternatif atau celah saat proses penyelesaian persoalan mengalami hambatan karena satu atau beberapa hal tidak sesuai dengan situasi pada umumnya. *Fault-tolerance* di sini dapat memanfaatkan penalaran berbasis kasus, tentunya analisis mengenai pembentukan basis kasus juga perlu dilakukan, sejalan dengan analisis mengenai ontologi komunikasi. Kasus yang pernah ditemui untuk *domain* tersebut perlu dimasukkan dalam basis kasus bersama dengan solusinya. Saat sistem nanti menemui persoalan baru, maka dengan *self-organized* sistem mampu memberikan solusi untuk persoalan tersebut. Perlu diingat bahwa analisis basis kasus nantinya tidak akan diterapkan pada sistem kognitif secara global, namun dibuat untuk tiap-tiap agen.

Perbaikan Struktur Agen/ Peran dalam Sistem Kognitif

Setelah interaksi antar peran telah didefinisikan, struktur hirarki dari agen dapat diperbaiki, terutama karena analisis dari sudut pandang internal juga telah dilakukan setelah tahap Identifikasi Interaksi antar Peran Berdasarkan Layanan yang Diberikan. Jika ditemui adanya beberapa peran dengan sejumlah fungsi atau layanan yang sama, maka dapat dibentuk peran baru, dan peran yang telah ada didefinisikan dengan lebih spesifik serta memanfaatkan layanan dari peran yang lebih umum. Peran ini yang nantinya diimplementasikan sebagai kelas-kelas agen.

Sudut Pandang Internal

Sedangkan dari sudut pandang internal, proses rekayasa untuk tahap analisis yang dilakukan terdiri atas:

Penentuan *Goal* dari Agen Tunggal

Setelah tahap analisis tahap Identifikasi Layanan atau Fungsi yang ada pada setiap Peran telah dilakukan, maka tujuan setiap agen perlu didefinisikan. Artinya, sesuai dengan perannya, setiap agen harus mencapai suatu status agar tujuan sistem secara keseluruhan, dan aktivitas agen tersebut tidak boleh menjadi penghambat dari tercapainya tujuan sistem. Agar dapat diketahui aktivitas apa yang harus dilakukan oleh agen dalam rangka mencapai *goal*, maka *initial mental state* dari agen dan *event* yang dapat mengaktifkan agen juga harus didefinisikan di sini. Artinya, agen akan mulai melakukan aktivitas saat dipicu adanya suatu *event* yang terjadi. Sedangkan *initial state* berguna untuk menentukan aktivitas yang harus dilakukan agen dalam rangka mencapai *goal*. Saat agen telah mencapai *goal*, tentunya harus didefinisikan suatu *event* atau dalam bentuk protokol

komunikasi yang menunjukkan status ini agar agen lain yang berhubungan dapat melakukan aktivitas dalam rangka mencapai tujuan sistem.

Penentuan Proses Pencapaian Tujuan

Setelah *initial state* dan *goal* didefinisikan pada langkah Penentuan *Goal* dari Agen Tunggal maka langkah selanjutnya adalah menentukan proses yang harus dilakukan oleh agen dari *initial state* hingga mencapai *goal*. Dalam proses mencapai *goal* agen, maka agen mungkin perlu melakukan komunikasi dengan agen lain selama proses pencapaian tujuan.

Penentuan Perilaku dari tiap Agen yang Terlihat oleh Agen Lain dalam Sistem

Sifat alami dari sistem multiagen adalah adanya informasi yang bersifat terdistribusi. Oleh karena itu, tidak semua informasi atau pengetahuan yang dimiliki suatu agen akan juga tersimpan di agen lain. Karena tidak semua informasi di dalam agen diketahui oleh agen lain, maka perlu adanya suatu analisis dari sudut pandang internal agen untuk mendefinisikan informasi apa yang akan tampak oleh agen lain dalam sistem kognitif. Informasi tersebut dapat berupa layanan yang dapat diberikan oleh agen, serta *event* yang dapat memicu aktifnya agen tersebut atau *event* yang dapat memicu aktifnya agen yang lain. Informasi lain yang cukup penting adalah apabila terdapat kegagalan pada agen, maka *event* apa yang akan dibangkitkan sehingga sistem mengetahui adanya kegagalan untuk kemudian agen-agen mengorganisasikan diri sendiri dalam menyelesaikan kegagalan agar tujuan sistem tetap dapat tercapai. Hal ini sangat terkait dengan hasil analisis terhadap kebutuhan *fault tolerance* pada tahap Identifikasi

Interaksi antar Peran Berdasarkan Layanan yang Diberikan. Analisis pada tahap ini dapat dilakukan secara paralel dengan analisis untuk interaksi antar peran, karena perilaku agen yang terlihat oleh agen lain juga menentukan komunikasi antar agen tersebut.

Setelah analisis perilaku yang tampak oleh agen lain dilakukan, maka tahap desain dilakukan. Pada prakteknya, langkah-langkah rekayasa pada tahap analisis dapat berulang kembali jika pada suatu langkah dihasilkan analisis yang ternyata harus melakukan perubahan pada hasil langkah sebelumnya.

Desain

Dalam tahap desain juga terdapat dua sudut pandang yang digunakan, yaitu sudut pandang eksternal dan sudut pandang internal. Penomoran pada tahap ini melanjutkan pada tahap sebelumnya agar jelas urutan antara tahap analisis dan desain.

Sudut Pandang Eksternal

Sudut pandang eksternal pada tahap desain terdiri atas langkah-langkah sebagai berikut.

Pembentukan Kelas-kelas Agen

Pada tahap ini, kelas agen didefinisikan berdasarkan hasil langkah Perbaikan struktur agen/ peran dalam sistem kognitif diagram keterhubungan antar agen akan terbentuk, dan interaksi antar kelas pun didefinisikan dalam bentuk pertukaran pesan yang harus dilakukan antar agen. Di sinilah desain komunikasi antar agen, terutama bagian sintaks harus didefinisikan sesuai dengan semantik hasil dari tahap analisis. Dengan adanya kelas-kelas agen serta keterhubungannya, akan terbentuk arsitektur sistem kognitif dengan gambaran proses-proses yang terjadi di dalamnya.

Pembangunan *Knowledge Sharing* serta Representasi Pengetahuan yang Digunakan pada Sistem

Knowledge Sharing dimanfaatkan karena sistem kognitif selalu memanfaatkan pengetahuan dan menghasilkan pengetahuan hasil pembelajaran. Bagian ini sebenarnya tidak terlepas prosesnya dengan desain pada tahap. Pembangunan Basis Pengetahuan Lokal Dalam proses *self-organized*, basis kasus lokal digunakan secara bersama (*knowledge sharing*) antar agen, untuk menyelesaikan persoalan. Penjelasan lebih rinci mengenai keterkaitan *knowledge sharing* dengan basis kasus tiap agen dijelaskan pada tahap. Pembangunan Basis Pengetahuan Lokal

Sudut Pandang Internal

Sudut pandang internal pada tahap desain terdiri atas langkah-langkah sebagai berikut.

Pembentukan Struktur di Dalam tiap Kelas Agen

Struktur dari tiap kelas agen didefinisikan pada langkah ini. Struktur ini meliputi atribut dari agen yang dapat berupa pengetahuan atau informasi yang diperlukan dan dihasilkan oleh agen; *method* dalam agen yang mencakup fungsi-fungsi (*tasks*) yang dilakukan oleh kelas agen tersebut; serta layanan atau informasi apa saja yang ditampilkan untuk agen lain dalam rangka interaksi. Tentu saja definisi di sini harus sesuai dengan desain interaksi baik berupa *knowledge sharing* atau pertukaran pesan dengan protokol komunikasi tertentu.

Pembangunan Basis Pengetahuan Lokal

Karena penelitian ini berfokus pada pemanfaatan proses penalaran manusia, maka basis pengetahuan yang digunakan adalah basis kasus, di

mana isinya terdiri atas pasangan kasus dan solusi. Dalam tahap ini dijelaskan juga keterkaitan antar basis kasus dalam upaya *knowledge sharing* antar agen di sistem kognitif.

Basis kasus pada tiap agen terdiri atas bagian kasus dan bagian solusi. Representasinya tergantung pada model konseptual yang digunakan dan kasus yang digunakan untuk implementasi pada komputer. Salah satu contoh yang sederhana adalah dengan representasi status dan atribut untuk bagian kasus. Sedangkan bagian solusi direpresentasikan dengan ruang status, ruang aksi, fungsi probabilitas transisi status, dan fungsi *reward*. Fungsi *reward* diadaptasi dari metode *unsupervised learning*. Metode ini dipilih karena pada penelitian ini diinginkan agen-agen dalam sistem dapat belajar sendiri tanpa membutuhkan suatu 'supervisor', agar *self-organized* tampak pada sistem kognitif yang dibangun. Fungsi *reward* berupa pemetaan terhadap *reward* yang diperoleh, jika dilakukan suatu aksi pada status tertentu. Fungsi inilah yang nantinya berperan untuk menentukan aksi atau kasus apa yang dipilih untuk menyelesaikan persoalan. Fungsi ini juga dimanfaatkan saat perawatan basis kasus.

Organisasi pengindeksan basis kasus untuk proses *retrieval* berdasarkan pada kesamaan kasus. Pengukuran kesamaan ini dilakukan dengan teknik *nearest neighbor*. Indeks basis kasus akan berubah saat proses modifikasi terhadap basis kasus terjadi. Fungsi *reward* digunakan di sini untuk menjaga kemitakhiran indeks basis kasus.

Dalam sistem kognitif yang diusulkan, tidak terdapat basis kasus global, sehingga pemutakhiran basis kasus terjadi pada masing-masing agen. Pengetahuan baru pun ditambahkan pada basis

kasus agen yang bersangkutan. Karena terdapat lebih dari satu basis kasus, maka perlu didefinisikan aksi global sebagai vektor-vektor dari aksi individual (tiap agen). Status yang dicatat pada suatu proses yang melibatkan lebih dari agen adalah status global, dan *reward* yang didefinisikan adalah *reward* global sebagai konsekuensi dari digunakannya status global. Agen-agen dalam sistem berkolaborasi untuk menggabungkan aksi mereka dalam menyelesaikan suatu persoalan melalui sekumpulan aksi yang direpresentasikan dalam format suatu vektor. Fungsi *reward* menghubungkan vektor aksi tersebut dengan transisi untuk status global berikutnya. Tujuan yang ingin dicapai adalah mencari vektor aksi yang memaksimalkan *reward* yang diperoleh. Solusi yang ditemukan akan digunakan untuk memutakhirkan indeks dari basis kasus tiap agen.

Fase adaptasi berlangsung jika terdapat kasus baru yang harus ditambahkan pada basis kasus. Jika basis kasus kosong, maka kasus baru tersebut otomatis ditambahkan pada basis kasus. Jika basis kasus tidak kosong, maka perlu dilakukan suatu proses untuk memeriksa apakah kasus tersebut perlu ditambahkan sebagai kasus baru pada basis kasus yang sudah ada. Pada tahap ini dilakukan kembali pengukuran kesamaan kasus baru dengan kasus yang sudah ada pada basis kasus. Pengukuran ini melibatkan suatu angka ambang batas, yang dijadikan patokan dalam penambahan kasus baru. Kasus baru ditambahkan jika ukuran basis kasus masih di bawah batas besarnya ukuran basis kasus, dan pengukuran kesamaan masih berada di bawah ambang batas. Jika kasus baru ditambahkan, maka proses perawatan basis kasus perlu dilakukan.

Proses perawatan terdiri atas dua tipe, yaitu tipe kuantitatif dan kualitatif. Perawatan kuantitatif, memastikan jika kinerja sistem secara keseluruhan masih sesuai dengan yang diharapkan. Jika terdapat kasus baru yang ditambahkan, proses pemutakhiran bisa dilakukan apabila hal tersebut tidak menambah waktu belajar bagi sistem. Bagian solusi dari tetangga terdekat kasus baru pada basis kasus harus diinisialisasi kembali karena sudah tidak valid.

Perawatan basis kasus yang kedua adalah perawatan kualitatif. Perawatan kualitatif melakukan evaluasi solusi berdasarkan kemanfaatannya untuk kasus baru. Salah satu pendekatan untuk menilai kemanfaatan adalah dengan mencoba setiap gabungan vektor aksi minimal satu kali pada tiap kasus di basis kasus. Hal ini dapat dipandang sebagai proses pelatihan bagi solusi kasus tersebut.

Dengan melakukan dua perawatan tersebut, maka basis kasus diharapkan selalu mutakhir dan dapat digunakan untuk menyelesaikan persoalan yang dihadapi sistem dengan kinerja yang tetap terjaga dengan baik.

Sama seperti pada tahap analisis, tahap desain juga berupa siklus berulang sehingga suatu langkah dapat diperbaiki jika ternyata pada langkah berikutnya dihasilkan suatu desain yang memerlukan penyesuaian kembali terhadap langkah sebelumnya. Namun sangat diharapkan pada proses rekayasa sistem kognitif ini, saat tahap desain dilakukan, tahap analisis sudah stabil sehingga tidak banyak perubahan yang dilakukan jika memang ada.

Implementasi

Pada tahap ini hasil desain diimplementasikan. Seharusnya pada tahap ini hasil analisis dan desain dapat diimplementasikan dalam bahasa atau ling-

kungan pemrograman apapun yang mendukung pemrograman berbasis agen. Oleh karena itu, jika suatu bagian diimplementasikan pada lingkungan *Windows* dan bagian lain diimplementasikan pada lingkungan *Linux*, keduanya dapat berinteraksi selama mengikuti desain komunikasi yang telah didefinisikan. Hal ini dapat dilakukan jika pengembangan sistem memanfaatkan suatu *middleware* yang mendukung interoperabilitas antar sistem operasi. Telah banyak tersedia kaskas pemrograman yang memanfaatkan *middleware* dan menyediakan transparansi bagi pengembang sehingga memudahkan tahap implementasi.

Keterhubungan antara ketiga tahap rekayasa yaitu analisis, desain, dan implementasi dapat dilihat pada Gambar 3. Pada gambar tersebut penomoran sesuai dengan penomoran langkah yang telah dijelaskan sebelumnya pada makalah ini.

SIMPULAN

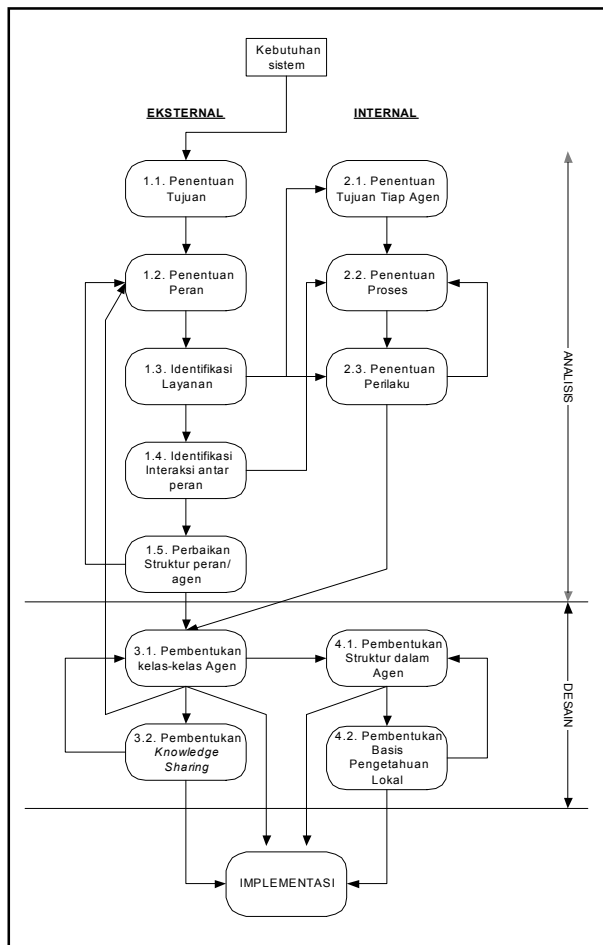
Sistem kognitif sebagai suatu pendekatan penyelesaian persoalan yang mengambil analogi dari proses kognisi manusia makin berkembang dewasa ini. Usulan pemanfaatan aspek kognitif makin diminati dari berbagai disiplin ilmu dalam ranah ilmu komputer antara lain rekayasa perangkat lunak, kecerdasan tiruan, serta sistem terdistribusi.

Usulan rekayasa sistem kognitif ini hanya mencakup *software process* dan tidak membahas *modeling language* sebagai bagian dari suatu metodologi rekayasa. Pertimbangan yang melandasinya adalah sudah banyak tersedia *modeling language* untuk rekayasa sistem yang berbasiskan agen. Karena bagian terkecil suatu sistem multiagen adalah agen, maka notasi atau *modeling language* untuk agen dapat dimanfaatkan dalam sistem

multiagen. Satu hal yang menjadi penting saat melakukan rekayasa berbasis multiagen adalah aspek komunikasi serta pemrosesan pengetahuan sebagai sarana agar agen dalam sistem mampu mengorganisasikan dirinya sendiri dalam rangka mencapai tujuan sistem seperti yang terlihat pada Gambar 3. Salah satu pendekatan pemrosesan pengetahuan adalah dengan memanfaatkan penalaran berbasis kasus, sebagai suatu teknik penalaran dari bidang kecerdasan tiruan yang memiliki kemiripan dengan proses penalaran manusia.

Penelitian selanjutnya yang dilakukan adalah memanfaatkan rekayasa sistem kognitif untuk mengembangkan suatu sistem kognitif yang banyak

berkembang di institusi pendidikan maupun institusi lain berbasis pengetahuan. Salah satu contohnya adalah *Collaborative Knowledge Building (CKB)*, sebuah bentuk sistem kognitif yang berupaya membangun suatu pengetahuan dari komunitas agen dengan latar belakang pengetahuan masing-masing. *Review* beberapa model untuk *CKB* dapat dilihat di (Maulidevi, 2006). Sedangkan pemanfaatan *CKB* sendiri bervariasi mulai dari dunia hiburan, industri pabrikan, institusi layanan jasa, insitusi pendidikan hingga institusi pemerintahan. Oleh karena itu, diharapkan usulan ini memberikan dampak positif untuk pendekatan pemanfaatan sistem kognitif dalam rangka mendukung aktivitas manusia yang memanfaatkan teknologi komputasi.



Gambar 3 Daur Rekayasa Sistem Kognitif

RUJUKAN

Eyesenck, MW. 2003. *Principles of Cognitive Psychology, 2nd edition*. East Sussex UK: Psychology Press Ltd.

Haugeland, J, ed. 1997. *Mind Design II: Philosophy, Psychology, Artificial Intelligence*. Cambridge: MIT Press.

Konar, A. 2000. *Artificial Intelligence and Soft Computing: Behavioral and Cognitive Modeling of Human Brain*. Florida: CRC Press LCC.

Konar, A, and Jain, L. 2005. *Cognitive Engineering: A Distributed Approach to Machine Intelligence*. London: Springer-Verlag Limited.

Koning, JL, and Ling, CX. 2003. Cognitive Agent and Multiagent Interactions, *Journal of Cognitive System Research*, vol. 4 pp. 167-168. Elsevier Science B.V.

Kushwana, DK, Singh, RK, and Misra, AK. 2006a. *Cognitive Web Based Software Development Process: Towards a more Reliable Approach*, ACM SIGSOFT Software Engineering Notes, Vol. 31 Nb. 4.

Kushwana, D.S., and Misra, A.K. 2006b. *Cognitive Complexity Metrics and Its Impact on Software Reliability Based on Cognitive Software Development Model*, ACM SIGSOFT Software Engineering Notes, Vol. 31 Nb. 2.

Luck, M, Ashri, R., and D’Inverno, M. 2004. *Agent-Based Software Development*. Norwood, Artech House Inc.

- Mitchell, TM. 1997. *Machine Learning*. McGraw-Hill.
- Maulidevi, NU, and Ahmad, AS. 2006. *A Review on SharedPlan Model for Collaborative Knowledge Building*, Proceeding of International Symposium on Communication and Information Technologies, IEEE CNF DOI: 10.1109/ISCIT.2006.339837, pp. 731 – 736, Thailand.
- Pal, SK, and SCK, Chiu. 2004. *Foundations of Soft Case-Based Reasoning*. John Wiley & Sons, Inc.
- Parkin, AJ. 2000. *Essential Cognitive Psychology*. East Sussex UK: Psychology Press Ltd.
- Roth, EM, Patterson, ES, and Mumaw, RJ. 2002. *Cognitive Engineering: Issues in User-Centered System Design*. To appear in: Roth, E. M., Patterson, E.S. & Mumaw, R. J. *Cognitive Engineering: Issues in User-Centered System Design*. In J. J. Marciniak (Ed.), *Encyclopedia of Software Engineering*, 2nd Edition. New York: Wiley-Interscience, John Wiley & Sons
- Russel, S, and Norvig, P. 2003. *Artificial Intelligence: A Modern Approach*. New Jersey: Prentice Hall.
- Stachl, G. 2006. *Group Cognition: Computer Support for Building Collaborative Knowledge*, Cambridge: MIT Press.
- Sycara, K. 1998. *Multiagent Systems*. Article in AI Magazine Summer 1998, American Association for Artificial Intelligence.
- Weiss, G . 1999. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. Cambridge: MIT Press.