

**MODUL Manajemen Pengetahuan**

**(KODE CSM410)**

**MODUL SESI 10**

**TOPIK KNOWLEDGE BASE**

**DISUSUN OLEH**

**Riya Widayanti**

**UNIVERSITAS ESA UNGGUL**

**TAHUN PENULISAN**

**SUBTOPIK 1 TOPIK SESI INI**

1. **Kemampuan Akhir Yang Diharapkan**

Setelah mempelajari modul ini, diharapkan mahasiswa mampu :

* 1. Memahami konsep basis pengetahuan
	2. Knowledge Rule Base
	3. Knowledge Case Base
1. **Uraian dan Contoh**

Bab ini mengkaji metode yang digunakan sistem informasi Sistem Informasi (SI) untuk menyelesaikan masalah perangkat lunak dan perangkat keras yang dihadapi oleh berbagai tenaga kerja menggunakan komputer pribadi sebagai alat pengambilan data utama mereka. Bab ini juga mengkaji evolusi yang terjadi dari hanya menggunakan teknisi help-desk dan kemampuan mereka untuk memecahkan masalah perangkat keras dan perangkat lunak, hingga penggunaan berbagai jenis basis pengetahuan untuk pemecahan masalah yang lebih cepat, lebih murah, dan lebih efisien. Metodologi yang digunakan untuk sistem berbasis pengetahuan ini meliputi sistem pakar berbasis aturan dan sistem manajemen pengetahuan berbasis kasus. Literatur tampaknya menyarankan bahwa sistem manajemen pengetahuan berbasis kasus lebih cocok untuk help desk TI daripada sistem pakar berbasis aturan.

Help desk Teknologi Informasi (TI)

Dua dekade terakhir telah menunjukkan kemajuan eksponensial dalam teknologi sistem informasi yang tersedia untuk sektor bisnis dan pemerintah negara-negara di seluruh dunia. Pertumbuhan teknologi ini menciptakan kebutuhan untuk pindah dari perpustakaan besar pengikat tiga-cincin pada masalah sistem dan solusi ke metodologi yang memungkinkan teknisi help desk untuk memberikan solusi masalah dalam hitungan menit versus jam.

Graham dan Hart (2000) dalam laporan mereka tentang keberhasilan dan kegagalan mengembangkan help desk TI terpusat di seluruh Universitas di University of Pittsburgh, menunjukkan nilai yang ditambahkan ke help desk dengan penerapan sistem manajemen pengetahuan untuk menggantikan barang kuno metode manual. Sebelum penerapan sistem berbasis pengetahuan, teknisi help desk di University of Pittsburgh, diharapkan untuk meninjau masalah yang diselesaikan di berbagai basis data, dan dapat menyelesaikan masalah di masa mendatang berdasarkan informasi yang terkandung dalam bidang solusi dari basis data ini. . Basis data berisi sekitar 150.000 panggilan masalah pada waktu tertentu sehingga secara praktis mustahil bagi teknisi untuk dapat meninjau dan mempertahankan pengetahuan ini untuk menyelesaikan masalah di masa depan.

Graham dan Hart (2000) melanjutkan bahwa universitas mengembangkan sistem help desk yang berpusat pada pengetahuan dengan menggabungkan dua paket perangkat lunak yang tersedia. Yang pertama adalah paket pemrosesan tiket untuk login dan menyimpan tiket masalah yang sebenarnya. Paket kedua menyediakan kerangka kerja untuk penciptaan basis pengetahuan yang mengorganisirnya ke dalam kategori logis menggunakan format pohon keputusan.

Graham dan Hart (2000) menyimpulkan dengan menyatakan manfaat yang diperoleh oleh universitas karena penerapan sistem yang berpusat pada pengetahuan. “Manfaatnya meliputi:

1. Peningkatan konsistensi dan keakuratan respons terhadap panggilan Help Desk.
2. Peningkatan kualitas dukungan dengan mengurangi jumlah waktu yang dibutuhkan untuk meneliti masalah.
3. Mengurangi panjang panggilan rata-rata.
4. Pengiriman alat untuk pengguna akhir untuk mencari basis pengetahuan dan menyelesaikan masalah secara mandiri.
5. Mengurangi biaya pelatihan untuk Analis Help desk yang baru. ”

Last (2003) membahas manfaat membangun perangkat lunak help desk Anda sendiri sebagai lawan membeli versi off-the-shelf. Kesimpulan terakhir adalah bahwa dengan pertumbuhan teknologi informasi dan tuntutan yang sudah ditempatkan pada berbagai sistem sentris in-house, teknisi help desk di University of Pittsburgh, diharapkan untuk meninjau masalah yang diselesaikan di berbagai database, dan dapat menyelesaikan masalah di masa depan berdasarkan pada informasi yang terkandung dalam bidang solusi dari database ini.

Halverson, Erickson, dan Ackerman (2004) meneliti konsep yang digunakan pada perusahaan besar yang menggunakan konsep menggunakan paradigma Tanya Jawab (Q&A) di mana mereka menyimpan pertanyaan (masalah) dan jawaban (solusi) dalam pengetahuan Q&A- mendasarkan. Implementasi sistem Q&A ini memungkinkan baik pelanggan in-house dan on-line mengakses kueri basis pengetahuan.

Peningkatan panggilan layanan dari 4.076 pada tahun 2000 menjadi 22.126 pada tahun 2005 di University of Rochester mendorong mereka untuk merancang sistem help desk yang memungkinkan mereka menangani peningkatan panggilan. Para perancang di University of Rochester mengembangkan sistem help desk, yang disebut System Reference (SysRef), menggunakan beberapa mesin pencari dan paket help desk. Para desainer menggunakan Intuit Corporations "Track-it" untuk mencatat masalah panggilan. Pencarian Coveo, yang mirip dengan pencarian di Google atau Yahoo, digunakan sebagai mesin pencari utama mereka. Mereka kemudian menyimpan data menggunakan Microsoft SQLServer 2000 sebagai sistem database untuk menyimpan semua data help desk. University of Rochester menyatakan bahwa karena basis data SysRef telah secara substansial menjawab kebutuhan mendesak dari lingkungan TI mereka yang berubah, mereka “... menanti-nanti periode penguatan infrastruktur yang kemudian akan mempersiapkan aplikasi untuk pertumbuhan di masa depan sambil mempertahankan konsep asli menggunakan data yang ada bila memungkinkan dan tidak 'menciptakan kembali roda' ”(Padeletti, Coltrane, dan Kline, 2005).

Penelitian menunjukkan bahwa help desk TI bergerak cepat dari sejumlah besar pengikat tiga cincin ke beberapa jenis sistem berbasis pengetahuan. Gerakan ini mengkonfirmasi sebagian tesis studi eksplorasi ini bahwa basis pengetahuan menjadi bagian penting dari help desk TI. Pertanyaan penting yang masih harus diselidiki adalah manfaat relatif CBR dan RBR untuk tugas ini.

Sistem Pakar Didefinisikan

Jackson (1999) menyatakan bahwa sistem pakar adalah program komputer yang diturunkan dari Artificial Intelligence (AI). Tujuan AI adalah untuk memahami kecerdasan dengan membangun program komputer, atau cangkang, yang menunjukkan perilaku cerdas. AI lebih lanjut memusatkan perhatian pada konsep dan metode inferensi simbolik, atau penalaran, dan bagaimana pengetahuan itu akan digunakan untuk membuat kesimpulan itu akan terwakili di dalam mesin.

Jackson (1999) lebih lanjut menyatakan bahwa sistem pakar dapat sepenuhnya memenuhi kesenjangan pengetahuan dalam suatu domain atau dapat bertindak sebagai asisten untuk membantu ahli domain dalam memecahkan masalah yang kompleks. Biasanya, sistem pakar digunakan dalam beragam bidang pengetahuan, seperti kedokteran internal, kimia organik, dan aplikasi bisnis seperti help desk TI. Tugas sistem pakar mencakup interpretasi data, diagnostik seperti kerusakan mesin, analisis senyawa kimia kompleks, konfigurasi sistem komputer, dan untuk digunakan dalam robotika.

Program komputer konvensional dapat ditulis untuk memasok beberapa pengetahuan domain yang diperlukan yang diperlukan oleh seorang individu. Namun, ada perbedaan besar dalam cara sistem pakar dapat dibedakan dari program aplikasi konvensional ini.

Pertama, sistem pakar "mensimulasikan penalaran manusia" tentang domain masalah yang diberikan tetapi tidak mensimulasikan domain itu sendiri. Kedua, sistem pakar melakukan "penalaran atas representasi" pengetahuan manusia sambil mempertahankan kemampuan untuk melakukan perhitungan numerik dan / atau pengambilan data. Ketiga, program konvensional memecahkan masalah menggunakan metodologi algoritme yang ketat sementara sistem pakar menggunakan "metode heuristik atau perkiraan" yang tidak menjamin bahwa jawabannya akan benar-benar benar dan akan berhasil. Ini "heuristik atau metode perkiraan" lebih dikenal sebagai aturan praktis (Jackson, 1999).

Jackson (1999) melanjutkan bahwa ada perbedaan antara jenis lain dari program kecerdasan buatan dan sistem pakar. Pertama, sistem pakar menangani masalah yang membutuhkan banyak keahlian manusia. Program AI lainnya terutama berurusan dengan masalah matematika abstrak dan, sebagian besar, dianggap kendaraan penelitian. Kedua, sistem pakar, karena berusaha memecahkan masalah yang berhubungan dengan keahlian manusia, harus menunjukkan tingkat kinerja yang tinggi dalam hal kecepatan dan keandalan untuk mempertahankan kegunaannya. Ini karena masalah-masalah yang berhubungan dengan keahlian manusia, misalnya menyelesaikan masalah yang berkaitan dengan kinerja komputer, membutuhkan solusi dalam jangka waktu singkat karena volume masalah yang dihadapi dalam satu hari. Program AI hanya itu, program, yang digunakan sebagai kendaraan penelitian. Akhirnya, Jackson menyarankan bahwa sistem pakar harus "mampu menjelaskan dan membenarkan solusi atau rekomendasi" untuk memberikan kepercayaan pengguna bahwa jawaban yang diambil dari sistem pakar, pada kenyataannya, benar.

Englemore dan Feigenbaum (1993) menyatakan bahwa AI berkaitan dengan konsep dan metode inferensi simbolik, atau penalaran, oleh komputer, dan bagaimana pengetahuan ini akan digunakan untuk mewakili kesimpulan ini dalam sistem komputer itu sendiri. Englemore dan Feigenbaum melanjutkan bahwa setiap sistem pakar terdiri dari dua bagian utama: basis pengetahuan; dan mesin penalaran, atau inferensi.

Luger dan Stubblefield (2002) menyatakan bahwa basis pengetahuan sistem pakar berisi pengetahuan faktual dan heuristik. Pengetahuan faktual adalah pengetahuan tentang domain tugas yang dibagikan secara luas dan, biasanya ditemukan dalam buku dan jurnal. Selanjutnya, pengetahuan faktual ini umumnya disepakati oleh para ahli dalam domain tertentu.

Pengetahuan heuristik adalah pengetahuan kinerja yang kurang ketat, lebih pengalaman, dan lebih menghakimi. Pengetahuan yang dimiliki manusia itulah yang mendasari kemampuan kita untuk merumuskan “tebakan yang baik”. Formalisasi dan organisasi pengetahuan dikenal sebagai representasi pengetahuan.

Studi eksplorasi dalam disertasi ini membahas dua paradigma terpenting yang digunakan dalam sistem pakar. Paradigma ini adalah alasan berdasarkan hukum dan kasus.

**Penggunaan saat ini dan Kemajuan dalam Sistem Pakar dan Akuisisi Pengetahuan menggunakan Rule-based Reasoning (RBR**)

Representasi pengetahuan berbasis aturan adalah salah satu jenis yang paling banyak digunakan (Englemore dan Feigenbaum, 1993). Tapi apa sebenarnya Penalaran berbasis Rule? Crossman et al. (1995) menyatakan dengan memecah ini menjadi bagian-bagian individualnya, aturan adalah representasi pengetahuan tentang mana "pola informasi yang digunakan para ahli untuk membuat keputusan dan apa keputusan yang mengikuti." Penalaran berbasis aturan menawarkan seperangkat aturan yang berantai ke diberikan kesimpulan. Cara paling populer untuk mewakili pengetahuan ini adalah dengan menggunakan aturan "jika-maka". Crossman et al. nyatakan bahwa aturan ini dapat direpresentasikan dalam hubungan logis “p → q”. Dalam hubungan ini, "p" mewakili suatu kondisi atau serangkaian kondisi dan "q" mewakili suatu kesimpulan atau serangkaian kesimpulan. Himpunan relasi / kesimpulan ini, menurut Ignizio (1991), himpunan paling umum digunakan oleh sistem pakar, dan terutama dalam sistem pakar berbasis aturan. Strategi inferensi ini lebih dikenal sebagai "modus ponens" yang berarti bahwa jika A menyimpulkan B dan A benar, maka B benar.

Crossman et al. (1995) lebih lanjut menyatakan bahwa banyak algoritma yang berbeda telah dikembangkan untuk mendukung premis dasar penalaran berbasis aturan. Ada berbagai perbedaan dalam pendekatan ini dan semuanya berada dalam domain rekayasa pengetahuan.

Salah satu contoh yang diberikan oleh Crossman et al. adalah bahwa "aturan rantai maju memfasilitasi sintesis pemrograman, sedangkan aturan rantai belakang lebih cocok untuk analisis atau pencarian."

Jika jenis penalaran yang terlibat dalam domain yang menarik melibatkan penggunaan diagram alir atau pohon, maka penggunaan aturan adalah cara terbaik untuk melanjutkan. Namun, peraturan ini tidak; mewakili fakta atau data itu sendiri; mereka lebih mewakili alasan tentang fakta atau data. Sistem berbasis aturan menggunakan mesin inferensi yang merupakan algoritme yang mengatur apa yang dapat dilakukan aturan, kapan aturan itu diaktifkan atau dipicu, dan prioritas apa yang diberikan kepada masing-masing untuk dieksekusi. Aturan dalam sistem berbasis aturan juga dapat menghibur bentuk tertentu dari penalaran tidak pasti, misalnya, menambah atau mengurangi tingkat kepercayaan saat mengevaluasi hipotesis atau menyediakan mekanisme alternatif untuk menangani jalur penalaran lain (Crossman et al., 1995).

Perkakas berbasis aturan dan sistem pakar lainnya lebih dikenal sebagai cangkang. Keuntungan terbesar yang ditawarkan oleh sistem berbasis aturan adalah mereka memungkinkan pengguna untuk melihat aturan dalam format bahasa yang hampir alami dan memberikan penjelasan mengapa penjelasan dibuat.

Luger dan Stubblefield (2002) menyatakan bahwa upaya pertama untuk membangun sistem pakar menggunakan paradigma berbasis aturan, tidak mungkin sangat berhasil. Alasan utama untuk ini adalah karena ahli domain merasa sangat sulit untuk mengekspresikan pengetahuan diam-diam dalam istilah yang dapat digunakan untuk memecahkan masalah spesifik.

Akhirnya, Luger dan Stubblefield (2002) menyimpulkan bahwa skema representasi pengetahuan yang paling banyak digunakan untuk sistem pakar adalah berbasis aturan. Biasanya, aturan itu sendiri tidak akan memiliki kesimpulan tertentu tetapi akan ada beberapa tingkat kepastian bahwa kesimpulan itu akan berlaku jika kondisinya berlaku. Ada teknik statistik yang digunakan untuk menentukan kepastian ini. Sistem berbasis aturan, apakah mereka memiliki kepastian atau tidak, biasanya mudah dimodifikasi. Kepastian ini membuatnya relatif sederhana untuk memberikan jejak bermanfaat dari penalaran sistem. Jejak-jejak ini dapat digunakan dalam memberikan penjelasan tentang apa yang dilakukannya.

Perlu dicatat bahwa alasan berbasis aturan digunakan dalam lingkungan help desk terutama sebagai peningkatan teknologi berbasis kasus (Luger dan Stubblefield, 2002).

**Penggunaan saat ini dan Kemajuan dalam Sistem Pakar dan Akuisisi Pengetahuan menggunakan Case-based Reasoning (CBR)**

Case-based reasoning (CBR) adalah paradigma Artificial Intelligence (AI) untuk pemecahan masalah dan penggunaan kembali pengetahuan yang menggunakan contoh serupa sebelumnya untuk memecahkan masalah saat ini. Selanjutnya, CBR menarik kemampuannya untuk mencari ingatannya untuk solusi dan memperoleh yang baru tanpa harus memahami prinsip-prinsip yang mendasari domainnya (Kolodner, 1993).

Namun, untuk menjelaskan cara kerja CBR, kita harus terlebih dahulu memahami makna kasus. Watson (2002, hlm. 27) mendefinisikan case sebagai:

Kasus adalah catatan pengalaman yang mengandung pengetahuan, yang bisa eksplisit dan diam-diam. Misalnya, mereka dapat menjadi kasus dalam pengertian hukum; mereka dapat berupa riwayat kasus pasien dalam pengertian medis, rincian pinjaman bank, atau deskripsi situasi pemecahan masalah peralatan.

Watson melanjutkan dengan menggambarkan apa yang terdiri dari kasus hukum, riwayat kasus medis, pinjaman bank dan catatan pemecahan masalah. Pertama, ia menyatakan bahwa deskripsi terdiri dari "masalah hukum, gejala pasien, detail pinjaman, dan masalah peralatan". Dia menyimpulkan dengan menyatakan bahwa hasil atau solusi dari masing-masing uraian ini terdiri dari "putusan atau putusan, perlakuan, hasil pinjaman, dan perbaikan teknis".

Kolodner (1993) menyatakan bahwa suatu kasus lebih lanjut dapat digambarkan sebagai laporan dari suatu peristiwa, cerita, atau beberapa catatan yang biasanya terdiri dari masalah yang menggambarkan keadaan dunia ketika kasus itu terjadi dan solusi yang menyatakan solusi yang berasal dari masalah itu. Ini berarti bahwa CBR mendapatkan solusi dari kasus-kasus sebelumnya saja dan memperoleh kasus-kasus baru untuk meningkatkan dan mengembangkan kemampuan pengambilan keputusannya.

Lebih jauh, representasi suatu kasus memiliki berbagai bentuk, seperti contoh atau bahkan cerita, selama itu dapat dikenali oleh seorang pemikir dalam domain tertentu. Secara semantik, sebuah kasus mewakili sepotong pengetahuan tertentu dan konteksnya, di mana kasus tersebut akan diambil untuk membangun solusi untuk masalah baru. Ini berarti bahwa kita dapat melihat penalaran berbasis kasus sebagai proses mengingat serangkaian kasus sebelumnya dan membuat keputusan berdasarkan perbandingan antara mereka dan situasi baru.

Kolodner (1993) melanjutkan dengan membahas kelebihan dan kekurangan menggunakan CBR dalam pengembangan pengetahuan. Kolodner menegaskan bahwa CBR memungkinkan pemikir untuk mengajukan solusi atas masalah dengan cepat, menghindari waktu yang diperlukan untuk memperoleh jawaban-jawaban itu dari awal. Sebagai contoh, seorang dokter yang mengingat diagnosis lama atau perawatan mengalami manfaat ini. Pemikir berbasis kasus, seperti halnya pemikir lainnya, harus mengevaluasi solusi yang diusulkan; memulai upaya memecahkan masalah karena dapat menghasilkan proposal dengan mudah. Hal ini tentu terungkap selama evaluasi aplikasi CBR yang disebut CASEY (aplikasi CBR pertama Kolodner), yang menunjukkan percepatan dua urutan besarnya ketika masalah telah terlihat di masa lalu.

CBR memungkinkan pemikir untuk mengajukan solusi dalam domain yang tidak sepenuhnya dipahami oleh pemikir. Banyak domain yang mustahil untuk dipahami sepenuhnya, sering kali karena banyak yang bergantung pada perilaku manusia yang tidak terduga. CBR memungkinkan asumsi dan prediksi dibuat berdasarkan apa yang berhasil di masa lalu tanpa memiliki pemahaman yang lengkap tentang masalah atau masalah.

CBR memberi pemikir sarana untuk mengevaluasi solusi ketika tidak ada metode algoritmik yang tersedia untuk evaluasi. Menggunakan kasus, dalam hal ini, sangat membantu ketika ada banyak hal yang tidak diketahui, membuat jenis evaluasi lain menjadi tidak mungkin atau paling tidak sulit. Solusi dievaluasi dalam konteks situasi serupa sebelumnya. Sekali lagi, pemikir melakukan evaluasinya berdasarkan apa yang berhasil di masa lalu (Kolodner, 1993).

Metodologi penyelesaian masalah CBR sama dengan cara manusia menyelesaikan masalah ketika ada situasi ketika seseorang menghadapi situasi atau masalah baru, orang itu akan sering merujuk pada pengalaman masa lalu dari masalah yang sama.

(Pal dan Shiu, 2004). Pal dan Shiu melanjutkan bahwa konsep CBR sangat menarik karena fakta bahwa itu sangat mirip dengan perilaku pemecahan masalah manusia dan dengan demikian akan menghilangkan tugas analisis mendalam dari domain masalah di mana sejarah tersedia. Akhirnya, Pal dan Shiu menyimpulkan bahwa penggunaan metode ini mengarah pada keuntungan bahwa CBR dapat didasarkan pada "pengetahuan dangkal" dan tidak memerlukan upaya rekayasa pengetahuan yang diperlukan oleh sistem berbasis aturan.

Bergmann et al. (2003, p. 16) menyatakan bahwa "Dibandingkan dengan sistem pakar, sistem pendukung keputusan berbasis kasus tidak bergantung pada aturan yang dipasok oleh spesialis". Bergmann et al. percaya bahwa CBR lebih merupakan "pendekatan alami" di mana teknisi help-desk (atau spesialis lainnya) tidak pernah harus menyediakan aturan diagnostik atau untuk menentukan spesifikasi formal dari setiap proses keputusan yang digunakan untuk menentukan solusi untuk suatu masalah. Sistem pendukung keputusan CBR memiliki kemampuan untuk memperoleh dan mempertahankan pengetahuan karena sistem memiliki kemampuan untuk mempelajari kasus-kasus baru.

Pal dan Shiu (2004) menyatakan bahwa proses CBR dapat diabstraksi sebagai siklus yang terdiri dari empat langkah dasar (Gambar 1, Siklus CBR): (1) Pengambilan Kasus untuk menemukan kasus yang paling mirip yang akan mengatasi masalah baru , (2) Case Reuse untuk memanfaatkan case yang diambil untuk menyelesaikan masalah, (3) Case Revisi atau mengadaptasi untuk memodifikasi case yang diambil dengan harapan akan sesuai dengan masalah baru, dan (4) Retensi Kasus untuk mempertahankan kasus yang direvisi sebagai kasus baru dalam basis kasus setelah dikonfirmasi atau divalidasi.

Penelitian lain telah menyimpulkan bahwa empat langkah dasar dalam siklus CBR sebenarnya harus menjadi proses enam langkah dengan menambahkan fase Restore and Review.



Gambar Siklus CBR Cycle

Göker dan Roth-Berghofer (1999) percaya bahwa langkah-langkah dalam siklus CBR terkandung dalam dua siklus umum; Siklus Aplikasi dan Siklus Pemeliharaan. Siklus Aplikasi, yang berisi langkah Ambil, Penggunaan Kembali, dan Revisi, dilakukan setiap kali pengguna atau teknisi help-desk menyelesaikan masalah dengan sistem dukungan help-desk berbasis kasus. Jika solusi yang dihasilkan selama siklus Reuse tidak benar dan tidak dapat diperbaiki, teknisi help-desk harus menghasilkan solusi baru. Solusi baru ini digunakan selama fase ReCyle. Semua solusi baru yang dihasilkan dengan cara ini disimpan dalam buffer dan tersedia untuk teknisi help desk sebagai kasus yang belum dikonfirmasi. Kasing yang belum dikonfirmasi ini kemudian dikirim ke Siklus Pemeliharaan untuk diproses dan dimasukkan ke pangkalan kasus. Siklus Perawatan, yang berisi langkah Retain dan Perbaiki, jarang dilakukan. Biasanya,

fase pemeliharaan dilakukan pada interval tertentu untuk memperbarui basis kasus dengan case yang belum dikonfirmasi yang terkandung dalam buffer fase ReCycle. Ada dua langkah tambahan yang telah ditambahkan Göker dan Roth-Berghofer ke siklus CBR empat fase yang diterima secara umum. Ini adalah langkah-langkah ReCycle dan Perbaiki. Langkah ReCycle digunakan sebagai persimpangan antara Aplikasi dan Pemeliharaan Siklus dan berisi kasus yang tidak dikonfirmasi dikirim oleh Siklus Aplikasi. Kasing yang belum dikonfirmasi yang diambil dari buffer ReCycle ditempatkan pada langkah Perbaiki di mana kasus tersebut diperbaiki dan ditulis berbasis kasus.

Misi utama dari langkah Perbaiki adalah untuk memastikan bahwa kasus dasar akurat. Ada lima pemeriksaan yang dinyatakan oleh Göker dan Roth-Berghofer (1999, p. 144) yang harus dilakukan sebelum kasing dapat ditambahkan ke pangkalan kasing. Ini adalah:

1. “Apakah itu merupakan alternatif yang layak yang belum ada dalam basis kasus
2. Apakah itu termasuk atau dapat dimasukkan oleh kasus yang ada,
3. Apakah dapat dikombinasikan dengan kasus lain untuk membentuk yang baru,
4. Apakah kasus baru akan menyebabkan ketidakkonsistenan, dan
5. Apakah ada kasing yang lebih baru sudah tersedia di kasing. ”

Roth-Berghofer dan Iglezakis (2001) juga percaya bahwa siklus CBR enam langkah adalah metode yang benar dan bahwa dua fase, Pemeliharaan dan Aplikasi, paling baik menggambarkan proses CBR yang benar untuk mengambil solusi dan memastikan bahwa mereka akurat dan kemudian menyimpannya. dalam basis kasus. Gambar 4 menunjukkan bagaimana fase Perawatan dan Aplikasi berinteraksi.

1. **Daftar Pustaka**
2. https://core.ac.uk/download/pdf/51097894.pdf