

SISTEM BERBASIS PENGETAHUAN

RIPPLE DOWN RULES

PERTEMUAN KE - 6

PENDAHULUAN

- ❖ Suatu metode Akuisisi pengetahuan dalam pengembangan basis pengetahuan secara bertahap dari suatu sistem pakar atau sistem berbasis pengetahuan.
- ❖ Dibangun untuk mengatasi permasalahan tidak adanya penjelasan yang lengkap dari pakar mengenai keputusan yang mereka ambil.
- ❖ Pendekatan berbasis kasus (case-based)

PENDAHULUAN

- ❖ Pengklasifikasian Tunggal
- ❖ Mendapatkan pengembangan yang sederhana dan bertahap dari suatu Sistem Berbasis Pengetahuan (SBP) pada saat SBP sedang digunakan sehingga seiring dengan berjalannya waktu seorang pakar dapat melakukan evolusi pada SBP yang canggih ini sebagai tugas kecil tambahan dari tugas-tugas normal
- ❖ Pencarian konfigurasi dan heuristic dan lainnya

PENDAHULUAN

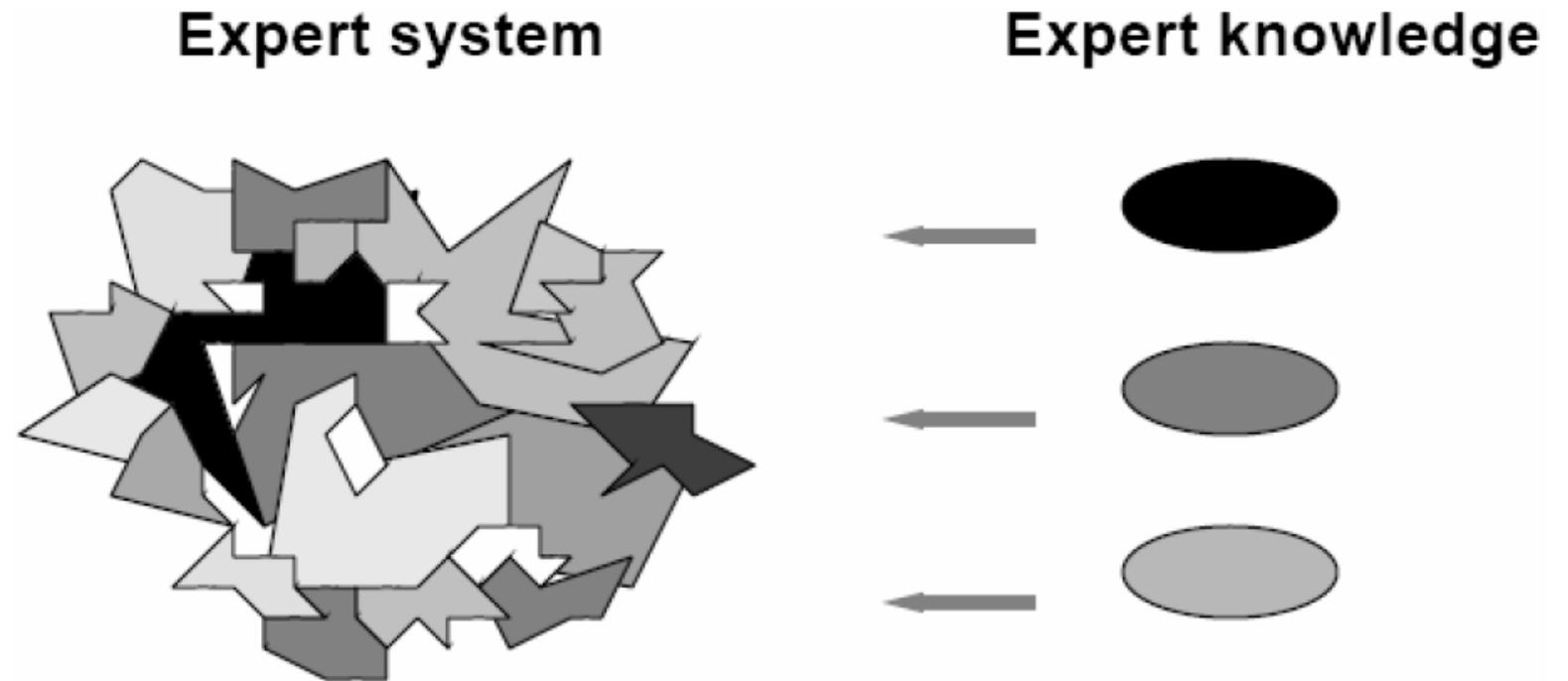
- ❖ Teori RDR menggunakan prinsip bahwa pengetahuan selalu diberikan oleh pakar untuk satu konteks tertentu dan akan bernilai benar hanya untuk konteks yang sama
- ❖ Pengetahuan baru ditambahkan jika terjadi kesalahan dalam memberikan klasifikasi suatu kasus
- ❖ Kasus yang menyebabkan penambahan pengetahuan ini akan disimpan sebagai kasus CORNERSTONE pada basis pengetahuan.
- ❖ Penambahan pengetahuan baru menjamin akurasi sistem secara bertahap karena dan tidak menyebabkan kesalahan interpretasi pada kasus-kasus sebelumnya

PENDAHULUAN

- ❖ Penambahan pengetahuan ini merupakan bagian dari proses akuisisi pengetahuan, yaitu suatu proses mentransfer pengetahuan dan mengubahnya dengan kemampuan kepakaran menjadi solusi untuk pemecahan masalah.
- ❖ Akuisisi pengetahuan secara bertahap merupakan keuntungan dari penggunaan RDR dalam suatu sistem berbasis pengetahuan, karena dalam masa pemakaianpun sistem masih terus berkembang

PENGERTIAN RDR

Adalah struktur sederhana yang ditujukan untuk menangkap, paling tidak sebagian konteks dimana suatu pengetahuan didapatkan dari seorang pakar



Representasi pengetahuan pada pakar dan dalam sistem pakar

Representasi Pengetahuan RDR

- ❖ Pengetahuan RDR direpresentasikan dalam bentuk pohon biner dengan setiap simpul menyatakan satu kaidah.
- ❖ Setiap simpul memiliki maksimal dua cabang yaitu cabang except dan cabang else.
- ❖ Kecuali pada simpul akar yang hanya memiliki satu cabang yaitu cabang except.

KOMPONEN – KOMPONEN PENGETAHUAN RDR

Simpul (node) pohon

Menyatakan kaidah dalam pengetahuan yang berisi serangkaian kondisi beserta konklusi yang dihasilkan apabila kondisi terpenuhi.

IF Klausa-1 AND klausa-2 AND... AND klausa-n THEN kelas-X

Atau

IF Klausa-1 OR Klausa-2 OR ... OR klausa-n THEN kelas-X

KOMPONEN – KOMPONEN PENGETAHUAN RDR

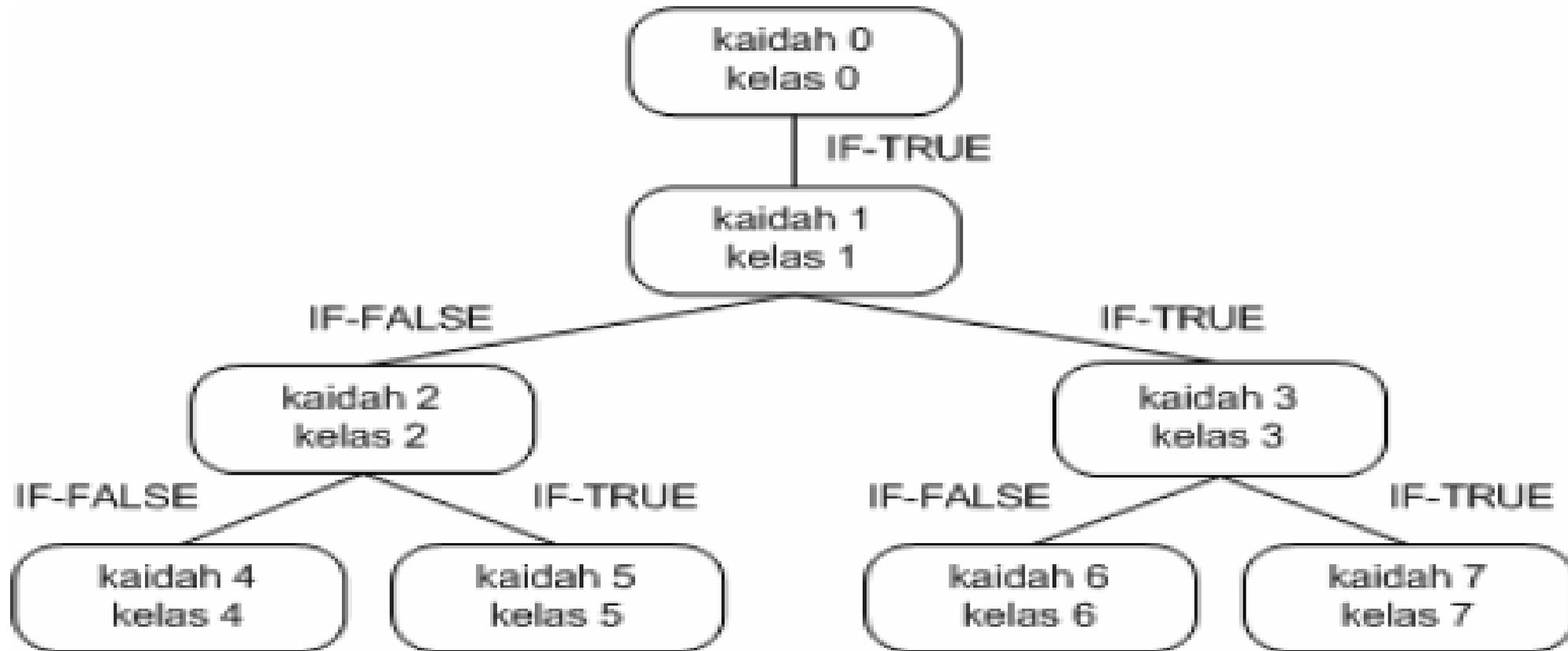
Cabang (except) pohon

- ✓ Disebut juga cabang IF –TRUE, direpresentasikan sebagai anak cabang kanan pada pohon RDR.
- ✓ Mendefinisikan suatu pengecualian positif dari kaidah pada simpul induk
- ✓ Cabang ini dieksekusi jika kondisi kaidah induk (kaidah dengan level tingkat satu tingkat atasnya) terpenuhi.

Cabang (else) pohon

- ✓ Disebut juga cabang IF-ELSE pada umumnya direpresentasikan sebagai anak cabang kiri pada pohon RDR
- ✓ Mendefinisikan suatu perkecualian negatif dari kaidah pada simpul induk.
- ✓ Akan dieksekusi jika kondisi induk tidak terpenuhi.

CONTOH



KAIDAH RDR SECARA UMUM

```
if <kondisi-1> then <konklusi-1>[because of <kasus cornerstone>]  
    except if <kondisi-2> then <konklusi-2>[because of <kasus  
    cornerstone>]  
else if <kondisi-3> then <konklusi-3>[because of <kasus cornerstone>]
```

Formulasi ini dapat dituliskan dalam pernyataan “if-then-else” biasa sehingga menjadi :

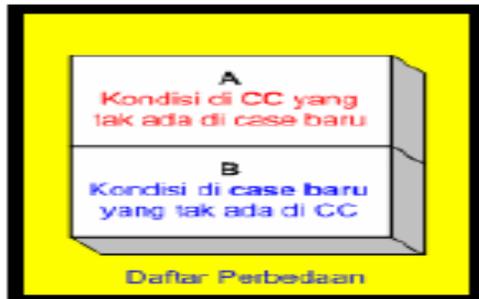
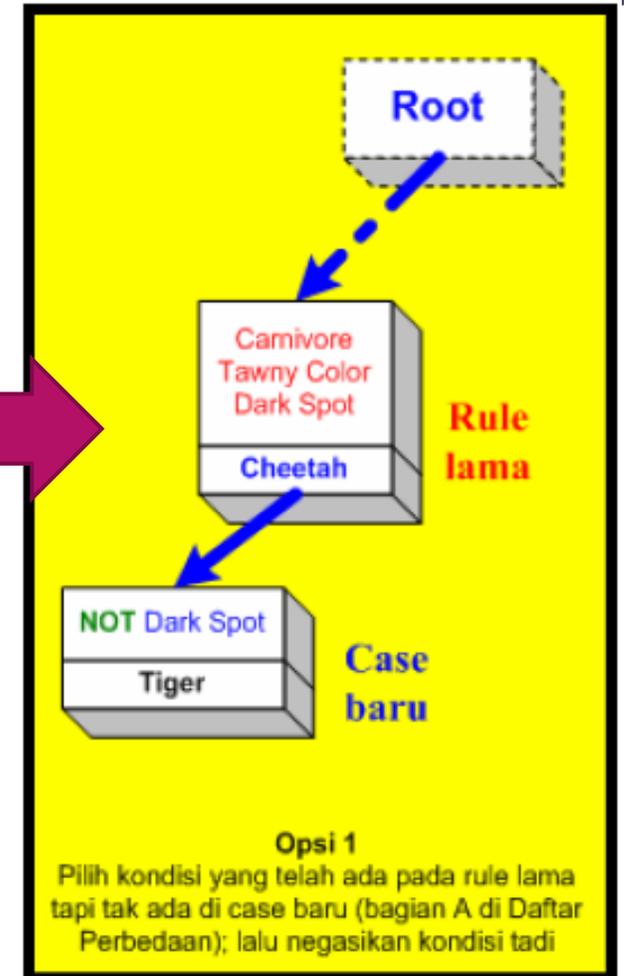
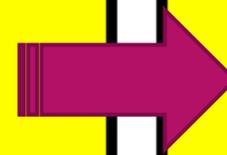
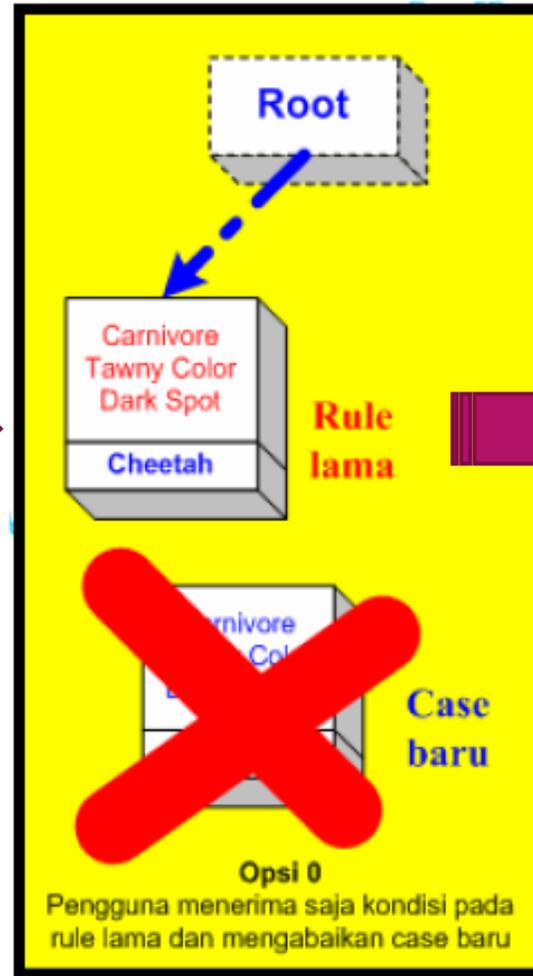
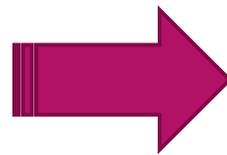
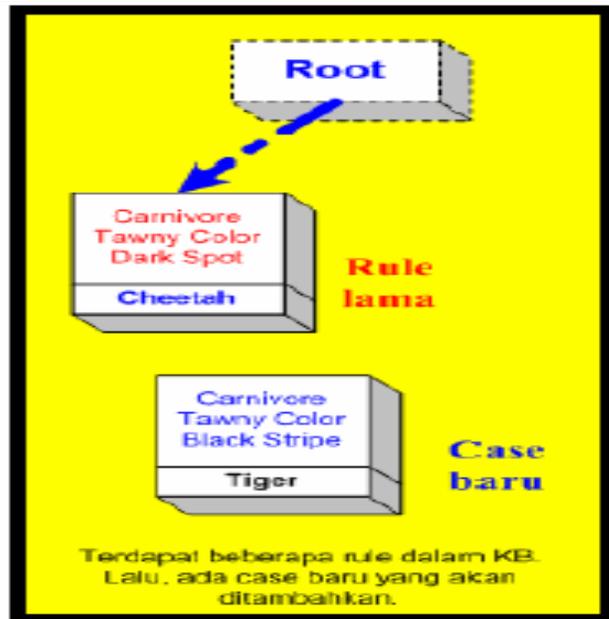
```
if <kondisi-1> then  
    if <kondisi-2> then <konklusi-2> else <konklusi-1>  
else if <kondisi-3> then <konklusi-3>
```

Apa yang dimaksud Kasus Cornerstones?

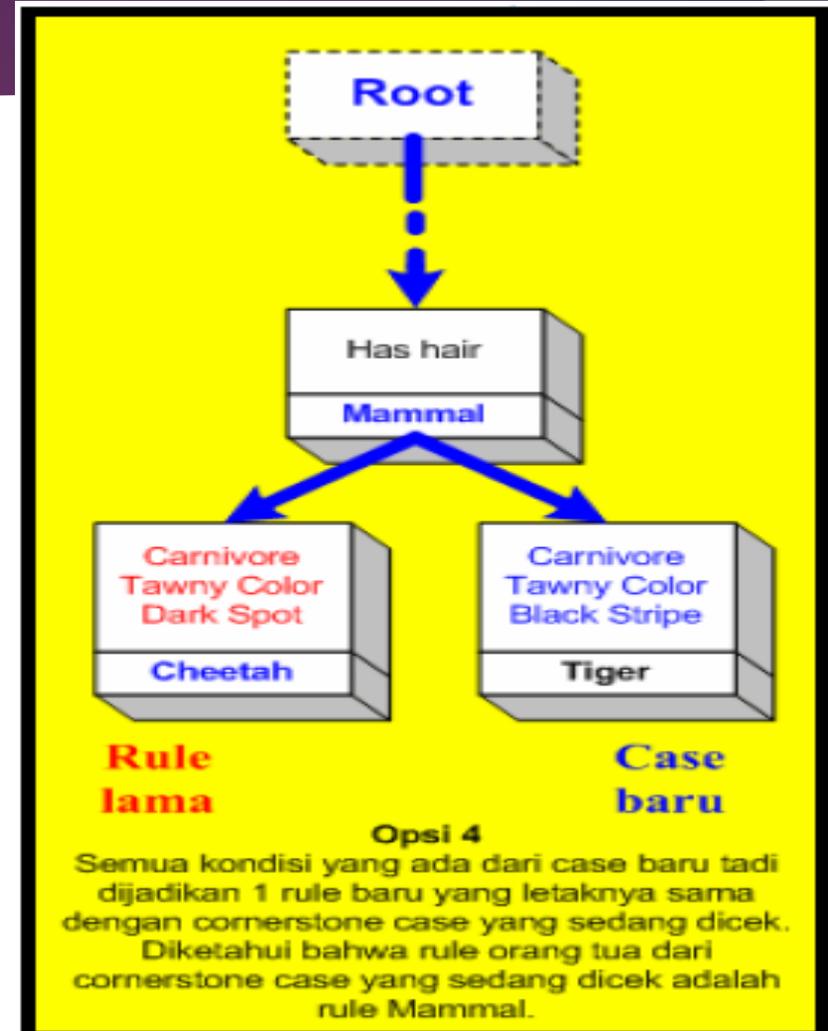
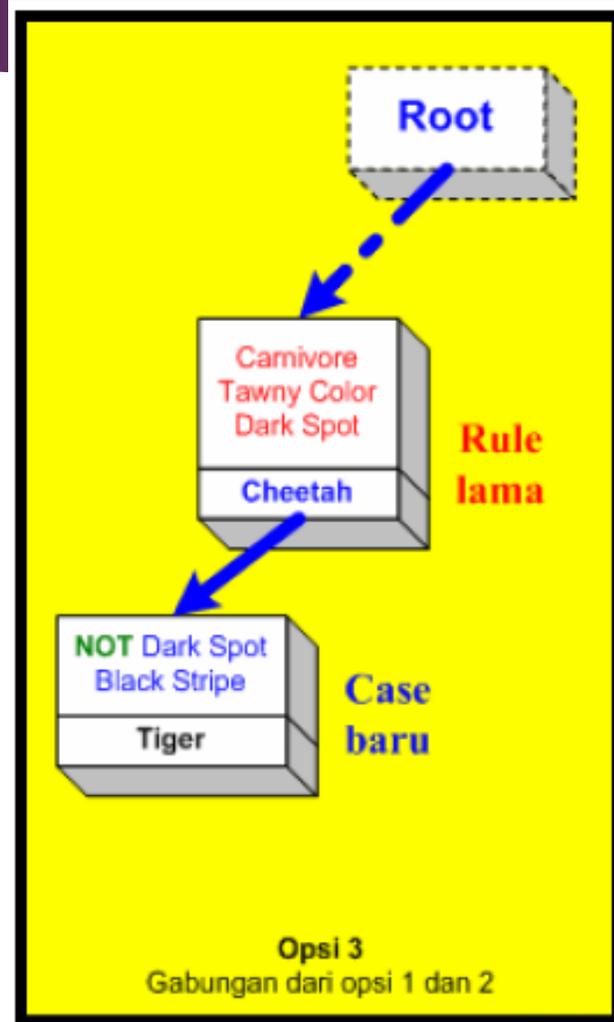
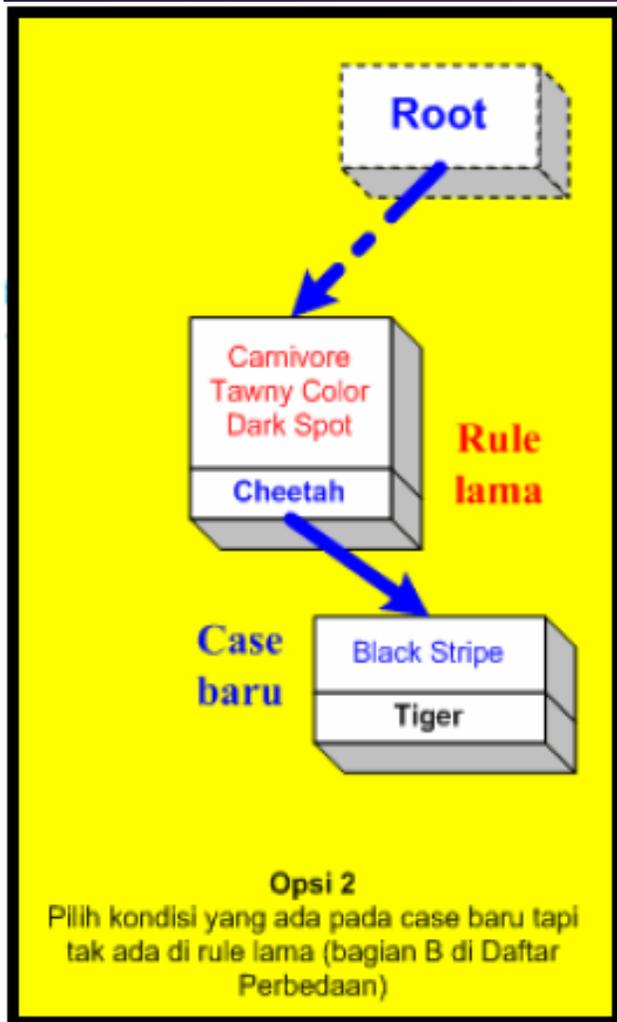
KASUS CORNERSTONE

- ❖ Basis pengetahuan RDR terdiri atas sekumpulan kaidah beserta kasus cornerstone yang terkait.
- ❖ Merupakan kasus yang menyebabkan terjadinya penambahan kaidah baru, yang disimpan ketika terjadi penambahan kaidah baru untuk mengkoreksi adanya kesalahan klarifikasi oleh sistem.
- ❖ Setiap kaidah terhubung dengan satu kasus Cornerstone kecuali kaidah pada simpul akar

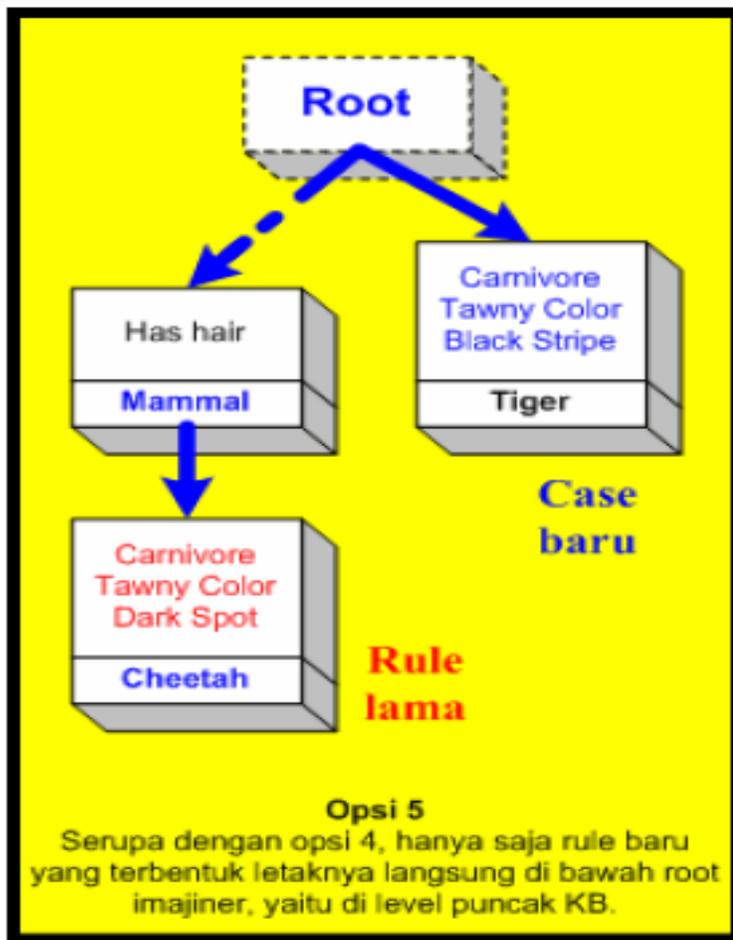
CONTOH PEMBUATAN RDR



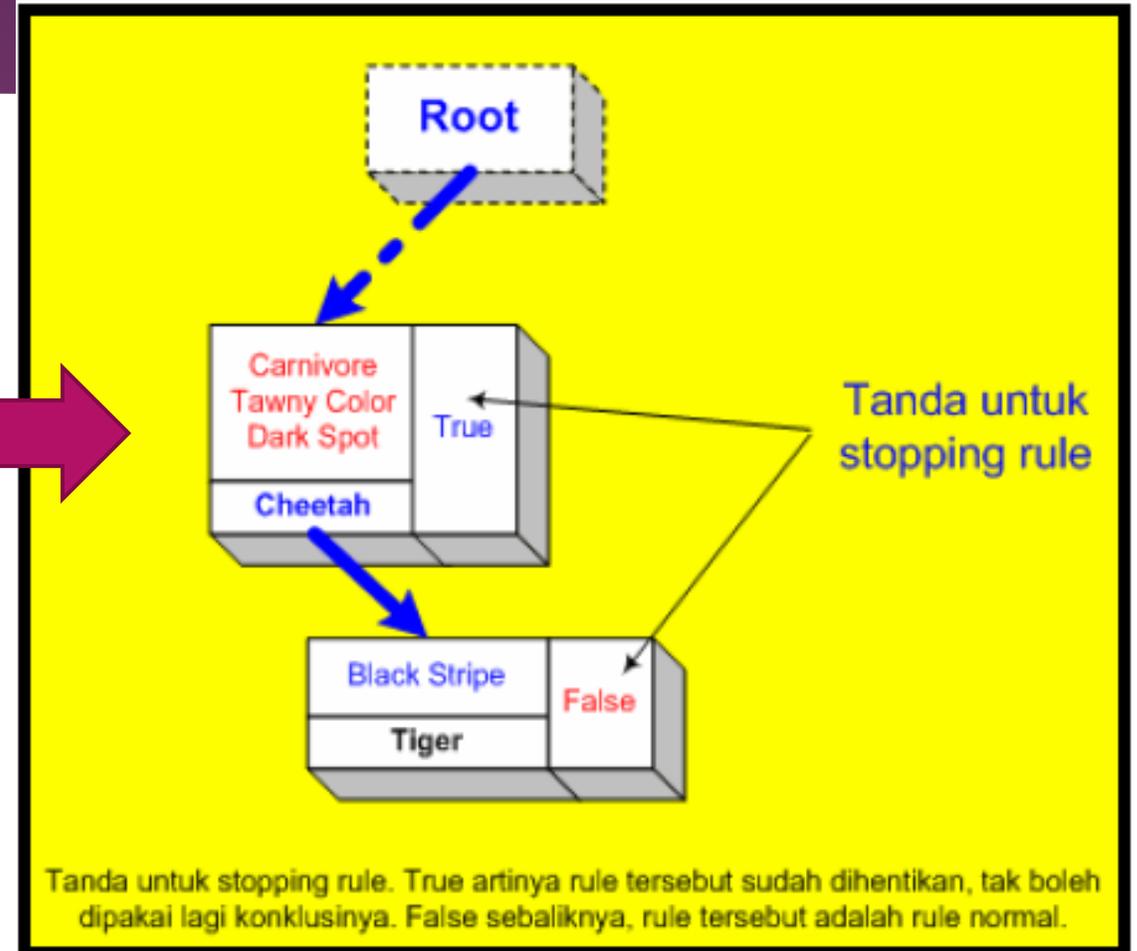
CONTOH PEMBUATAN RDR



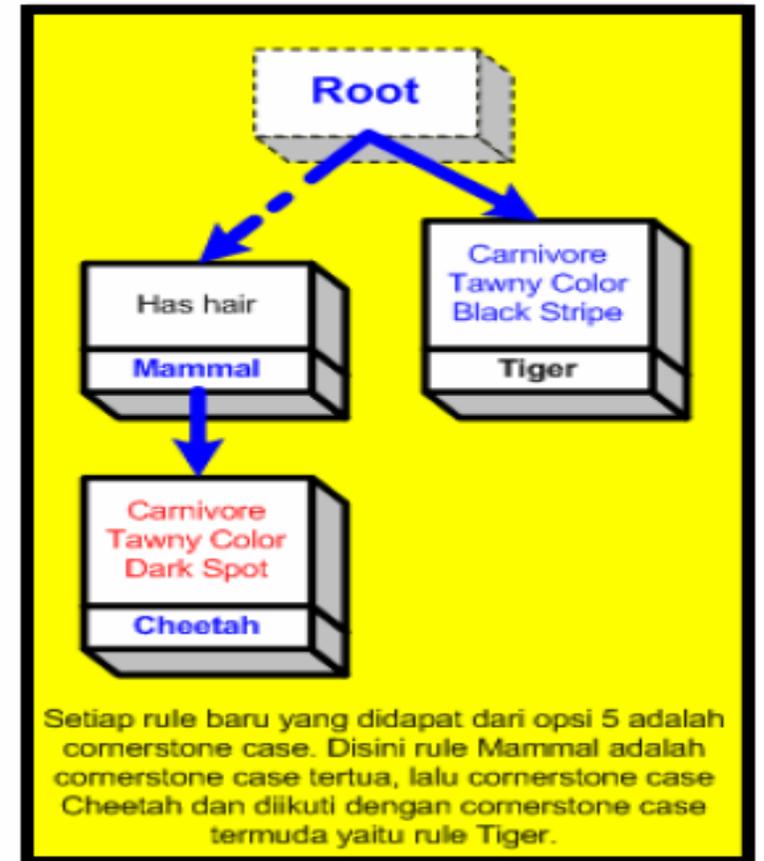
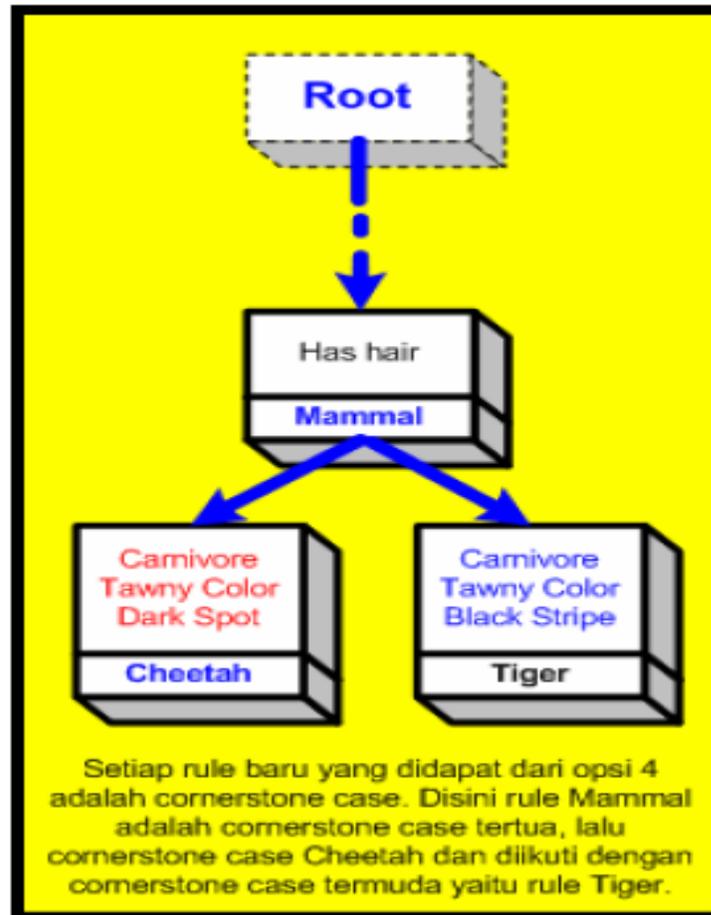
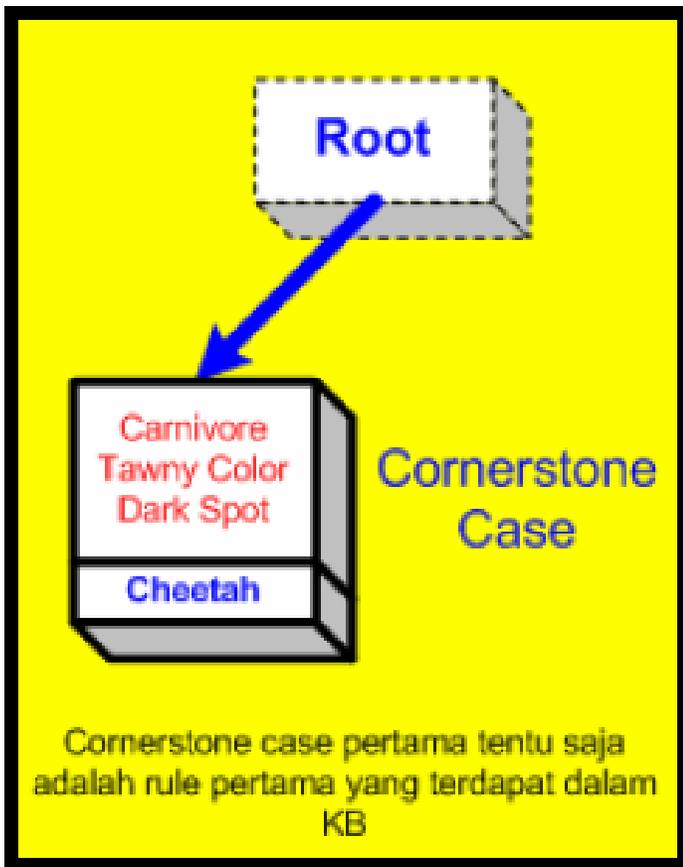
CONTOH PEMBUATAN RDR



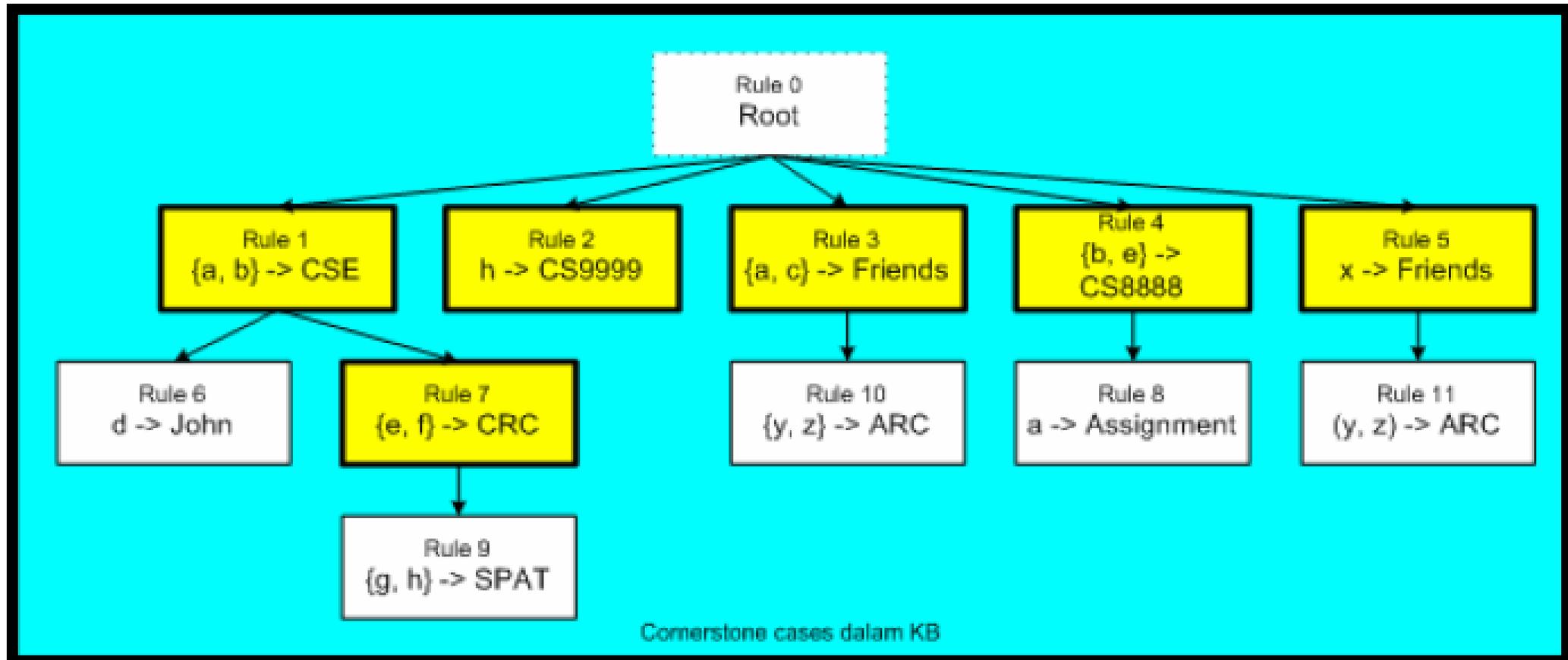
Jika suatu kondisi tidak terpakai maka berikan tanda Stopping Rule



CONTOH CORNERSTONE (CC)



CONTOH CC ke dalam Knowledge Base (KB)



PROSES INFERENSI RDR

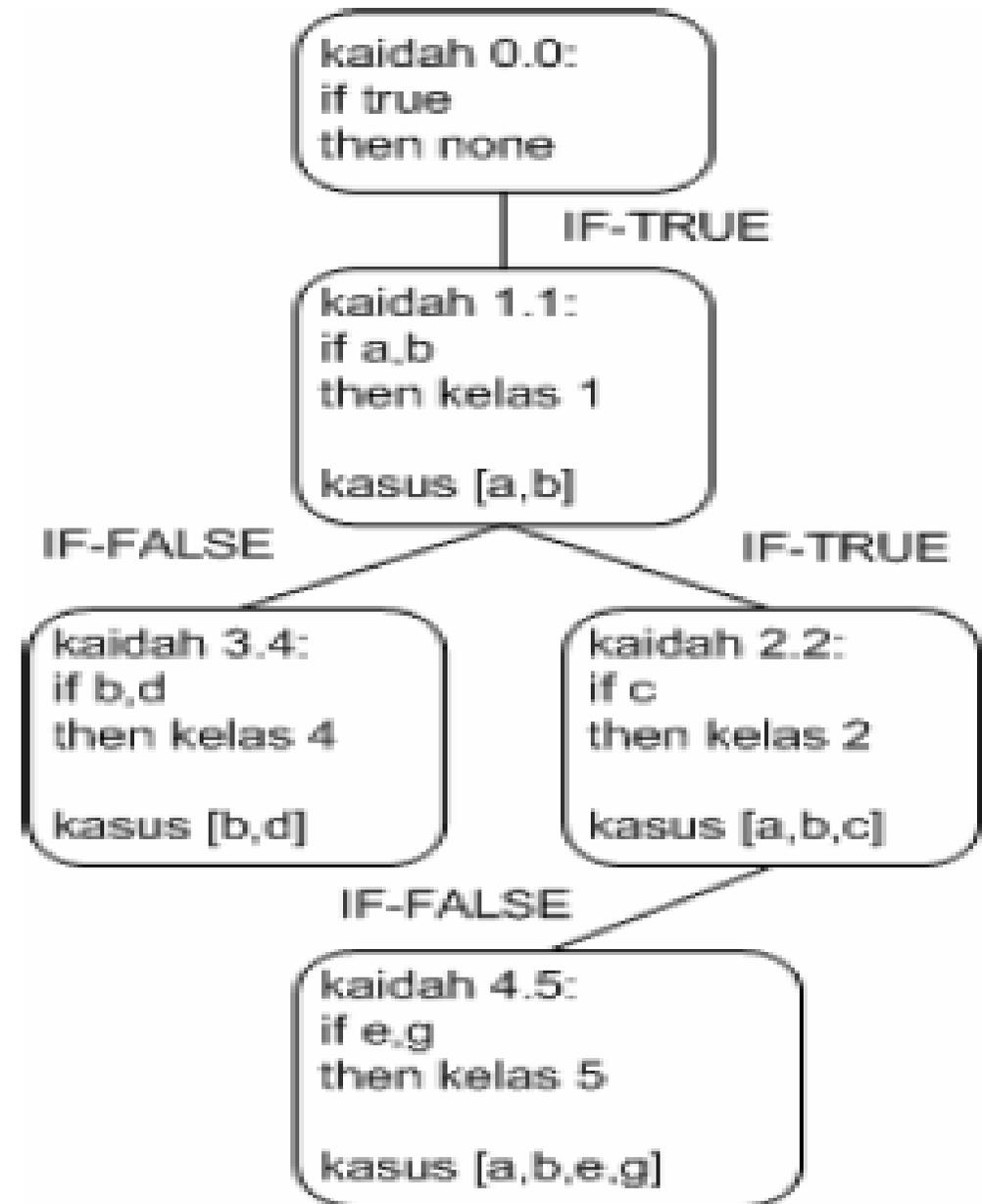
- ❖ Proses inferensi RDR dimulai pada simpul akar
- ❖ Kasus masukan akan diperiksa apakah memenuhi kaidah-kaidah dalam pohon untuk mendapatkan konklusi
- ❖ Setiap kasus masukan pasti akan memenuhi kaidah pada simpul akar
- ❖ Jika kaidah induk terpenuhi, maka konklusi dari kaidah induk akan menjadi konklusi untuk kasus terkait jika kaidah induk tidak memiliki cabang

PROSES INFERENSI RDR

- ▶ Jika kaidah induk memiliki cabang, maka penelusuran dilanjutkan ke cabang IF-TRUE untuk memeriksa pengecualian positif dari kaidah induk.
- ▶ Jika kondisi kaidah terpenuhi, maka konklusi kasus adalah konklusi kaidah sebelumnya yang dipenuhi dan penelusuran dilanjutkan ke cabang IF-ELSE untuk memeriksa pengecualian negatif dari kaidah induk
- ▶ Penelusuran berakhir jika simpul yang dievaluasi tidak memiliki cabang dan konklusi akhir sistem adalah konklusi kaidah terakhir yang dipenuhi pada jalur inferensi

CONTOH INFERENSI RDR

Jalur inferensi untuk kasus [a, b, e, f, g, i] adalah [(kaidah 0.0, none), (kaidah 1.1, kelas 1), (kaidah 2.2, kelas 1), (kaidah 4.5, kelas 5)]



Representasi RDR dalam Database Relational

Production rule (di bagian atas) dapat direpresentasikan sebagai kumpulan tuple (record) dalam tabel relasional, dimana rule sebagai sebuah objek memiliki hubungan presence (ada) dan absence (tak ada) dengan pelbagai variasi fakta dan hubungan outcome (hasil) dengan suatu interpretasi. Ini adalah representasi penyederhanaan karena tipe objek yang lain tak terindikasikan/ada disini.

RULE(42)

IF FTI is high
and T3 is high
and TSH is undetectable
and not on_t4

THEN **thyrotoxic**

ELEMENT RELATIONSHIP TABLE

owner	relationship	member
-----	-----	-----
-----	-----	-----
RULE_42	presence	FTI_high
RULE_42	presence	T3_high
RULE_42	presence	TSH_undetect
RULE_42	absence	on_t4
RULE_42	outcome	thyrotoxic
-----	-----	-----
-----	-----	-----

SUMBER REFERENSI

Azmi, Z dan Yasin, V. 2017. *Pengantar Sistem Pakar dan Metode (Introduction of Expert System and Methods)*. Jakarta: Mitra Wacana Media.

Subakti, Irfan. 2002. *Sistem Berbasis Pengetahuan (Knowledge Based System)*. Surabaya : Teknik Informatika, Institute Teknologi Sepuluh November.

Kusrini. 2006. *Sistem Pakar Teori dan Aplikasi*. Yogyakarta: Andi.