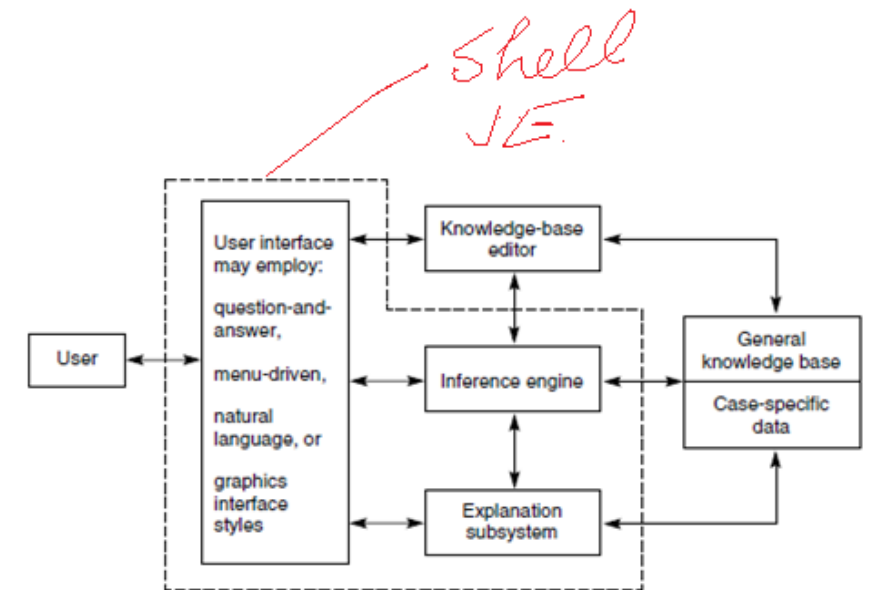


Sistemas expertos

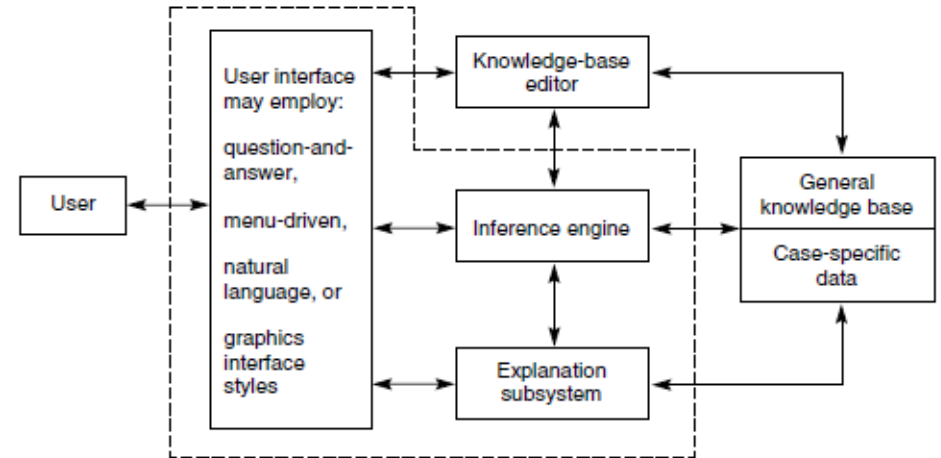
Arquitectura SE para un problema de dominio

- Interfaz de usuario: simplifica comunicacion, esconde complejidad como estructura interna de la base de reglas
 - Preguntas y respuestas
 - Manejo de menus
 - Interfaces graficas
- *La decision final va a depender de los requerimientos del usuario y de los requerimientos de la base de conocimientos y el Sistema de inferencia.*
- El ❤ del Sistema expert es la **base de conocimientos** que contiene el conocimiento de un dominio de aplicacion particular. Si es un Sistema expert basado en reglas este conocimiento esta representado en reglas **if then else**
- La **base de conocimiento** contiene conocimiento general e informacion de caso especifico



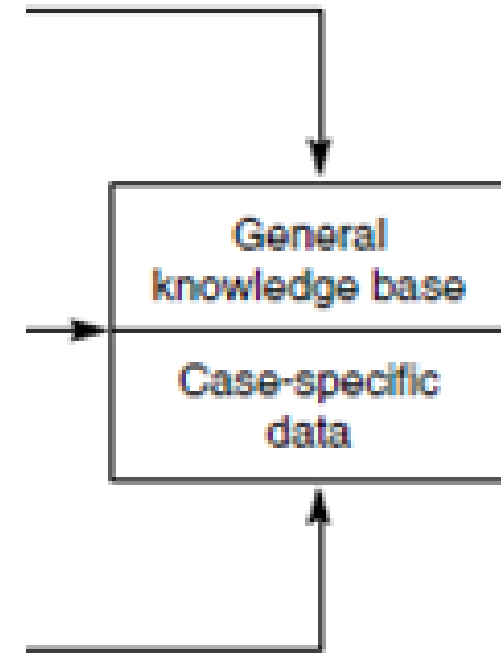
Arquitectura SE para un problema de dominio

- **La maquina de inferencias** aplica el conocimiento a la solución del problema
 - Se lo considera como un interpretador de la base de conocimientos (inference engine)



Arquitectura SE para un problema de dominio

- Data especifica de caso (Case specific data)
 - The facts
 - Conclusiones
 - Otra informacion relevante al caso en consideracion
 - Data de la instancia de un problema
 - Conclusiones parciales
 - Medidas de confianza de las conclusions
 - Finales Muertos en el proceso de busqueda
- Todo esto separado de la base de conocimiento



Arquitectura SE para un problema de dominio

- El **subsistema de explicacion** permite al programa explicar su razonamiento al usuario
- Las explicaciones incluyen justificaciones para las conclusiones del Sistema
- **Editor de base de conocimiento:** permite al programador corregir los bugs en el desempeño del programa, accediendo a la informacion provista por el subsistema de explicacion.
- NASA provee shells de sistemas expertos de CLIPS, JESS.
- Otros Shells en LISP y Prolog

Ejercicio

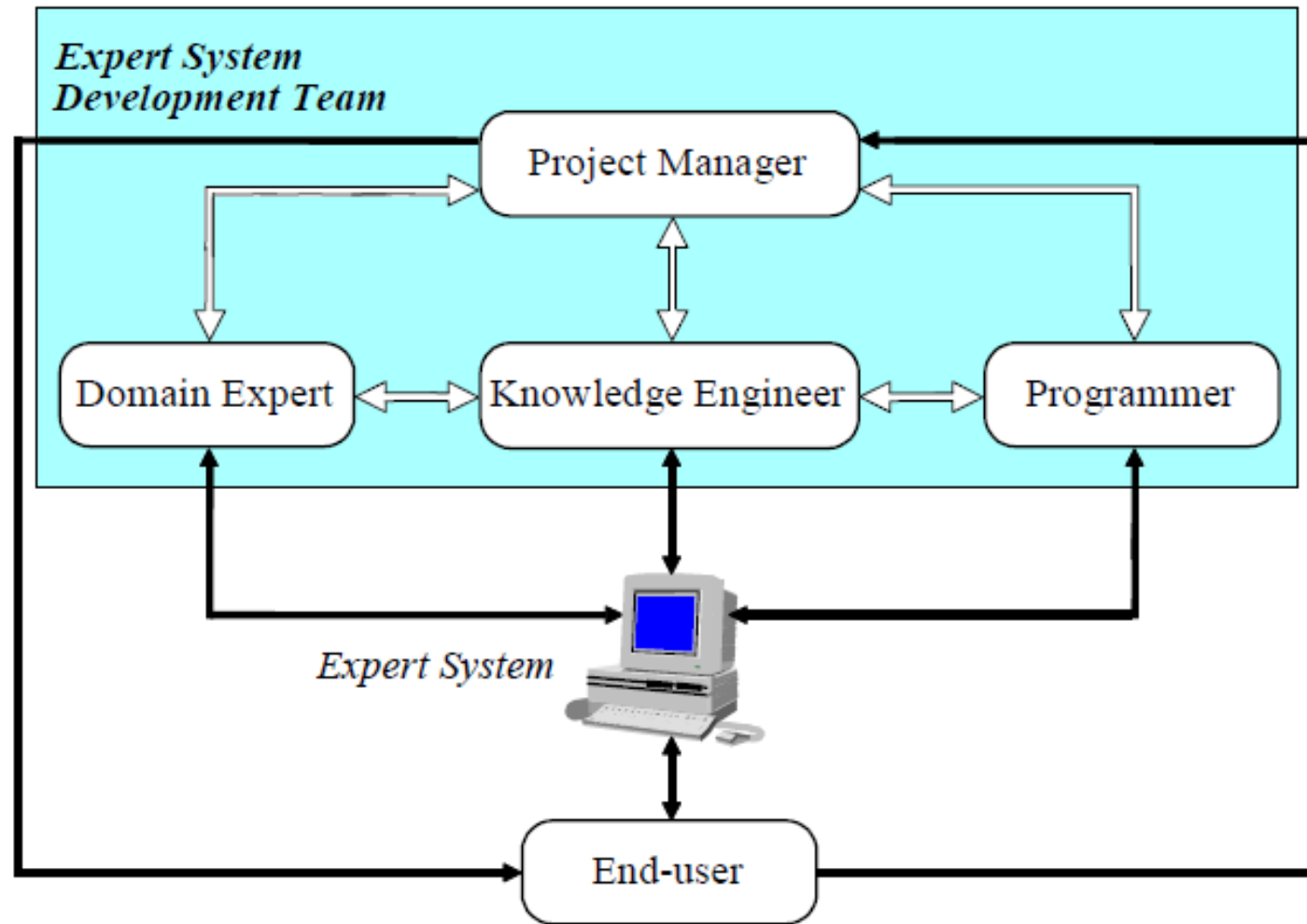
- Hagan grupos de 5
- Contesten las siguientes preguntas.
- 1. Existe un knowledge base editor en este articulo? Si existe describalo
- 2. Existe una maquina de inferencia? Si existe describala
- 3. Existe un explanation subsystem? Si existe que componente lo representa.
- 4. Como esta representada la base de conocimientos
- 5. El articulo presenta una data especifica de casos? Si es si como esta representada?
- 6. En que lenguaje se programo el SE
- 7. Donde esta implementada la maquina de inferencia
- 8. Con que se diseno la Interfaz de usuario
- 9. Describa la interfaz de usuario
- 10. Como calcularon el success rate? Que es el success rate?

Cuando se requiere que un problema sea resuelto con SE

Cuando se requiere que un problema sea resuelto con SE

- Si la **necesidad** de una solución **justifica el costo y el esfuerzo** de construirlo
- Si la **experiencia humana no está disponible** en todas las situaciones donde se necesita
- Si el **problema puede ser resuelto con razonamiento simbólico**(robots)
- Si el **dominio** del problema está **bien estructurado** y no requiere razonamiento de **sentido común**
- Si el problema no se resuelve usando métodos **computacionales tradicionales**
- Si existen **expertos cooperativos y articulados**
- Si el **tamaño y alcance** del problema es **apropiado**

Personal que se require para un SE



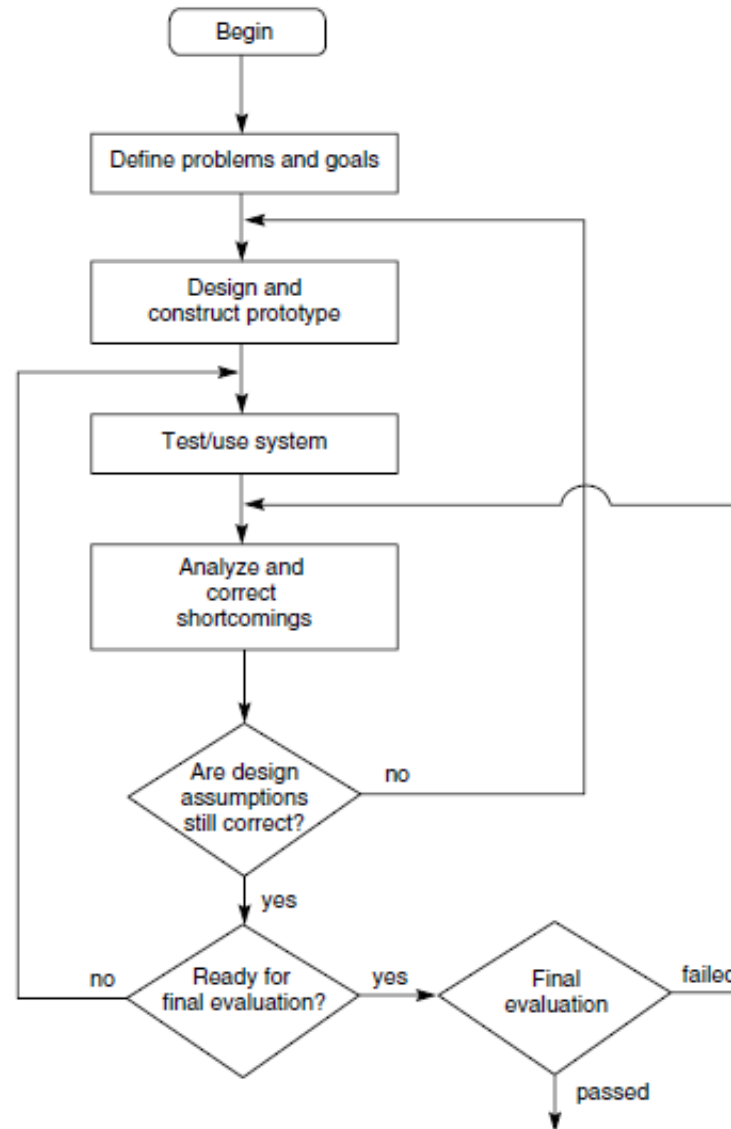
Personal para desarrollar SE

- **Ingeniero de conocimiento (+programmer):** conoce lenguaje de AI y es expert en representacion. Su trabajo es seleccionar el software y el hardware para la solucion ayudar al expert a articular el conocimiento e implementar el conocimiento en una base de conocimiento eficiente.
- **Experto de dominio:** provee el conocimiento del area del problema. **Trabaja** en el dominio por **mucho tiempo** y comprende las tecnicas de resolucion de problemas como atajos , manejo de data imprecisa, evaluacion de **soluciones** parciales. Debe deletrear las soluciones al ingeniero de conocimiento
- **Usuario final**

Como intercambia info el expert y el engineer

Eng	Gana familiaridad con el dominio del problema Se comunica con el experto
Eng y Exp	Se hacen entrevistas entre ambos
Eng	Observa expertos durante el performance de su trabajo
Eng y Exp	Extraen conocimiento
Eng	Entrega problemas ejemplo al expert Experto explica teorias usadas en la solucion Uso de video audio tapes
Eng	Es mejor que sea novato en el problema Los expertos obvian mucha informacion Un Eng novato no dejaria pasar detalles que no entiende
Eng	Disena el Sistema Selecciona forma de representacion (reglas, frames) Determina search strategies : forward, backwards, depth best Disena interfaz Entrega prototipo

Ciclo de desarrollo exploratorio



Vista de un Sistema experto

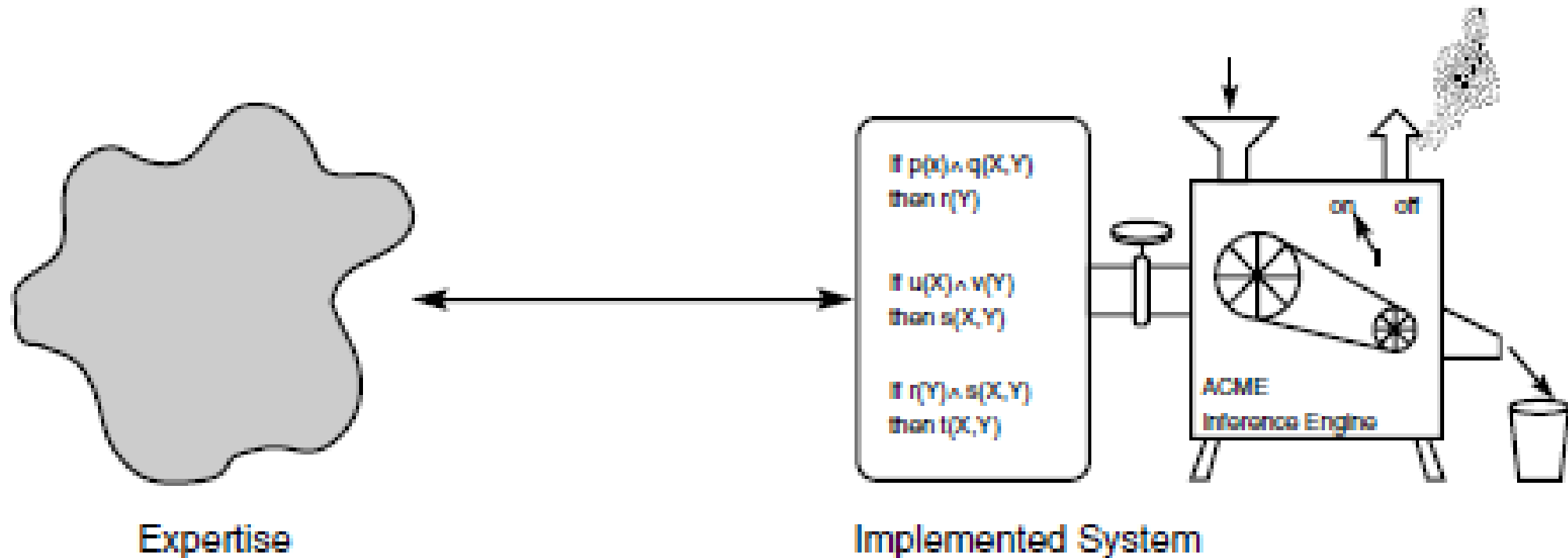
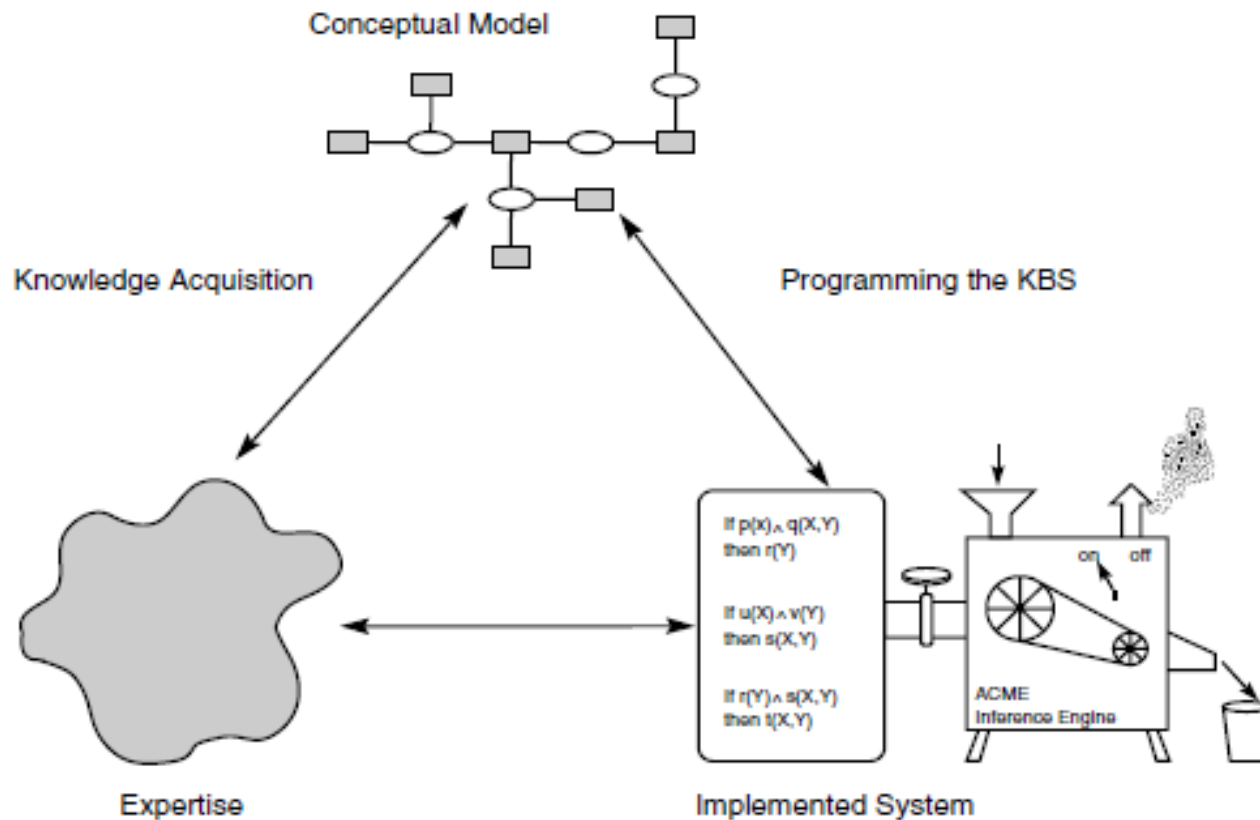


Figure 8.3 The standard view of building an expert system.

Modelo conceptual



- Modelo conceptual
- Es un modelo que esta entre el expert humano y el programa implementado
- El modelo es la concepcion del conocimiento de dominio del Ingeniero de k, que aunque es diferente del experto determina la construcción de la base de conocimiento formal

Sistemas expertos

- Basados en reglas
- Basados en razonamiento de casos
- Basados en Modelos
- Híbridos

Sistemas expertos basados en
reglas

Que es conocimiento

- Es una teoria o comprension practica de un tema o dominio
- Conocimiento es tambien la suma de lo que es actualmente conocido
- El conocimiento es poder
- Aquellos que poseen conocimiento se llaman expertos

Experto de dominio

- Cualquiera puede ser considerado un **expert de dominio**, si tiene **conocimiento profundo** (hechos y reglas) y una fuerte experiencia practica en un dominio particular
- EL area de **dominio puede ser limitada**
- En general un expert es una **persona llena de habilidades** que puede hacer cosas que otras personas no pueden.

Como se representa el conocimiento

- El proceso mental humano es interno, es demasiado complejo para ser representado como un algoritmo
- Muchos expertos son capaces de expresar su conocimiento en la forma de reglas para resolver un problema
- IF la “luz de trafico” es verde
- THEN la accion es “seguir”

- IF la “luz de trafico” es roja
- THEN la accion es “parar”

Las reglas como una tecnica de representacion

- **Rule** o regla en inteligencia artificial es la representacion de conocimiento **mas comunmente usada**
- Se la puede definir como una estructura **IF-THEN** que relaciona informacion dada o hecho en la parte IF hacia alguna accion en la parte THEN. **Una regla prove de descripcion de como resolver un problema** . Las reglas son relativamente facil de crear y comprender.
- Una regla consiste de dos partes
- **IF (antecedente, premisa, condicion)**
- **THEN (consecuente, conclusion o accion)**

Forma de una regla

- IF <antecedente> THEN <consecuente>
- Una regla puede tener multiples antecedents unidos por **CONJUNCIONES (AND)** , **DISJUNCIONES (OR)** o una combinacion de las dos
- **IF** <antecedente 1> **AND** <antecedente 2> ...**AND**<antecedente n> **THEN** <consecuente>
- **IF** <antecedente 1> **OR** <antecedente 2> ...**OR**<antecedente n> **THEN** <consecuente>

Forma de la regla

- **El antecedente de la regla incorpora 2 partes**
 - Un objeto (objeto linguistico)
 - Valor
- El objeto y su valor estan unidos por un operador
- El **operador** identifica el objeto y asigna el valor
- Operadores como **is, are, is not, are not** se usan para asignar valores simbolicos aun objeto linguistico
- Sistemas Experto pueden tambien **usar operadores matematicos** para definir a un objeto como numerico y asignarle a este el valor numerico
- IF la edad del **cliente** **< 18** AND el retiro > 1000 THEN la firma del padre es requerida.

Las reglas pueden representar relaciones, recomendaciones, directivas, estrategias y heurísticas

- **Relacion**

- IF el tanque de la gasoline esta vacio
- THEN el carro no se mueve

- **Recomendacion**

- IF la estacion del año es otono
- AND el cielo esta nublado AND la prediccion del tiempo es de alta humedad
THEN se recomienda llevar un paraguas

Las reglas pueden representar relaciones, recomendaciones, directivas, estrategias y heurísticas

- **Directiva**

- IF el carro no funciona
- AND el tanque de gasoline esta vacio
- THEN la accion es recargar gasoline en el carro

- **Estrategia**

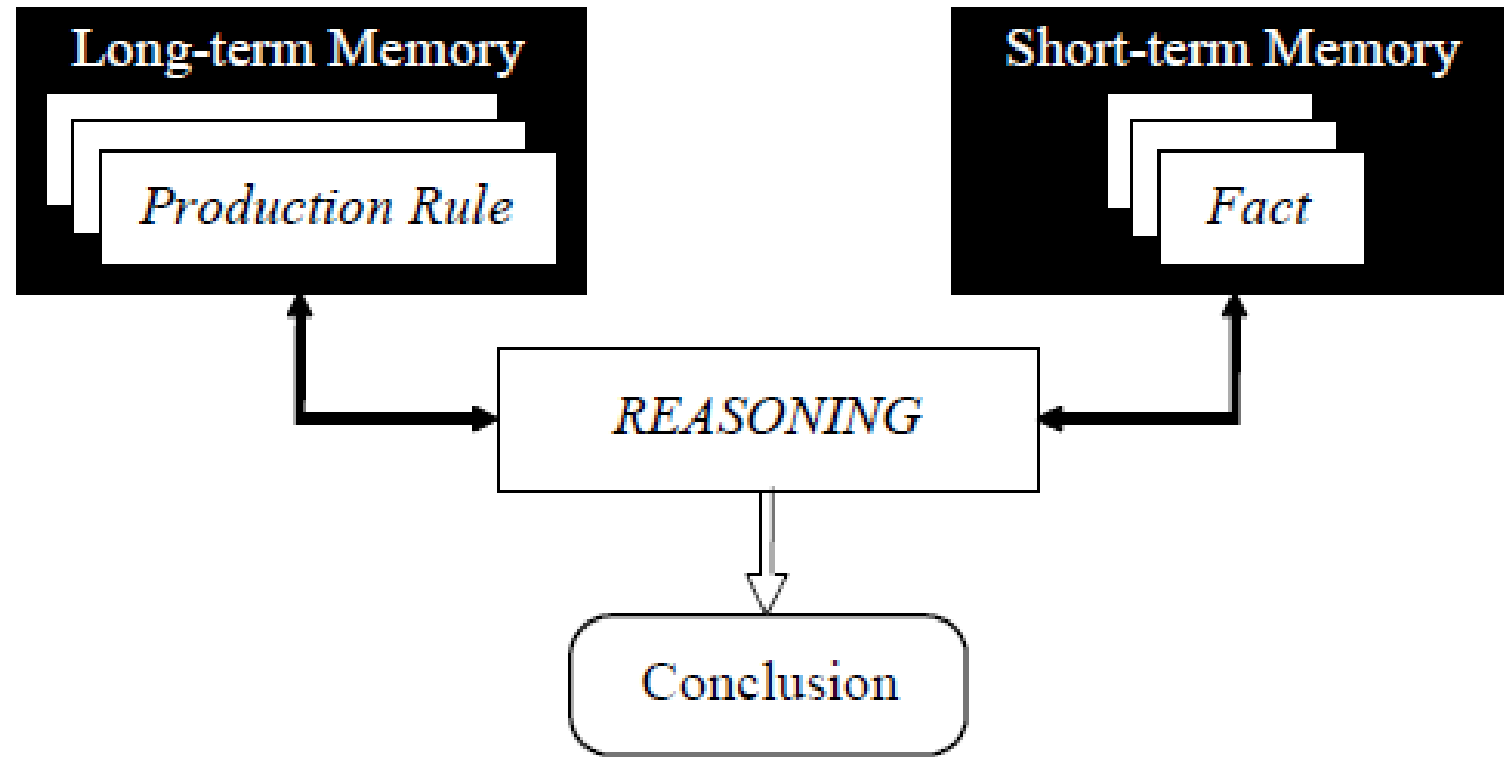
- IF el carro no funciona
- THEN la accion es chequear el tanque de la gasolina; paso 1 complete
- IF paso 1 es complete AND el tanque de la gasoline esta lleno
- THEN la accion es chequear la bateria:
- Paso 2 complete

Las reglas pueden representar relaciones, recomendaciones, directivas, estrategias y heurísticas

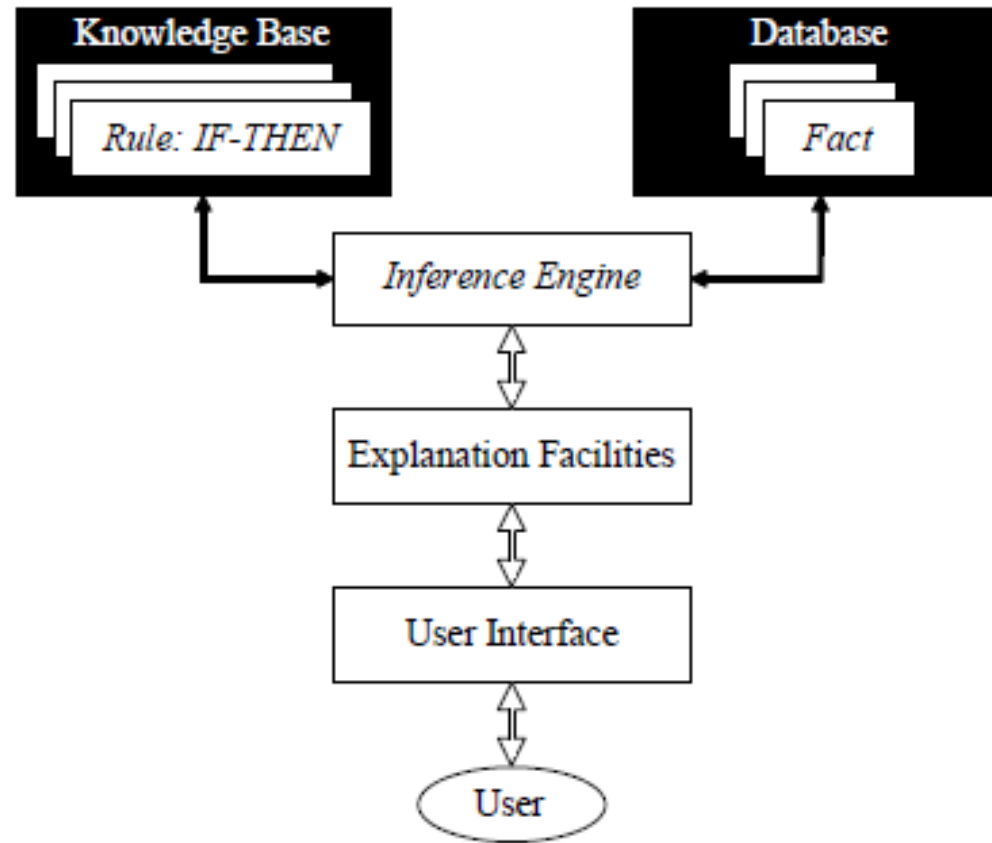
- **Heurística**

- IF el vertido es liquid
- AND el vertido ph <6
- And el olor del vertido es vinagre
- THEN el material del vertido es acido acetico

Similaridad con los problemas de sistemas de produccion



Sistemas Expertos similares a produccion



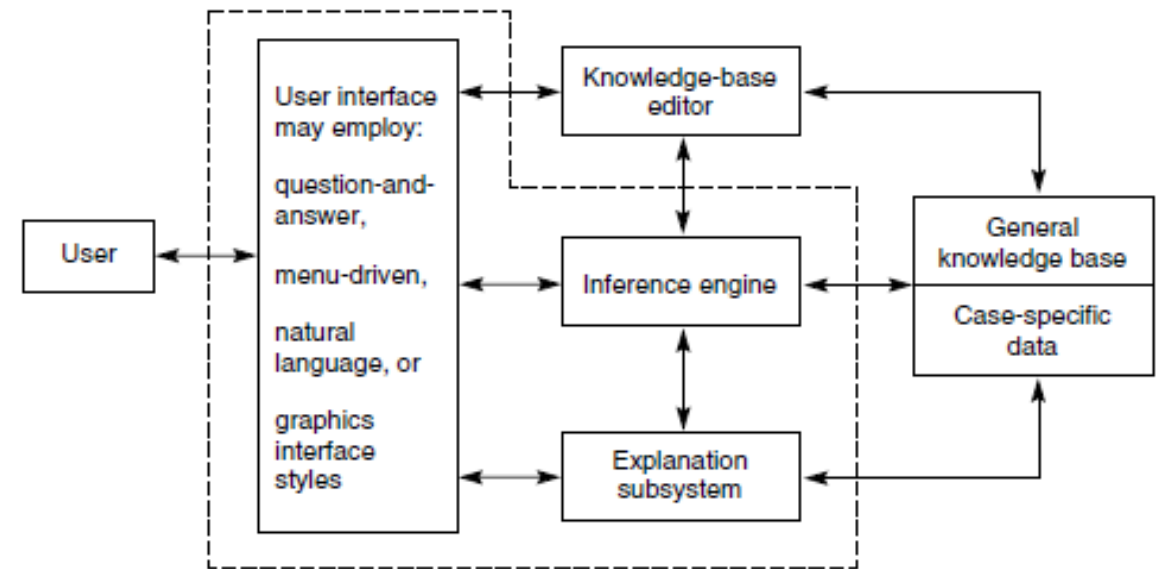
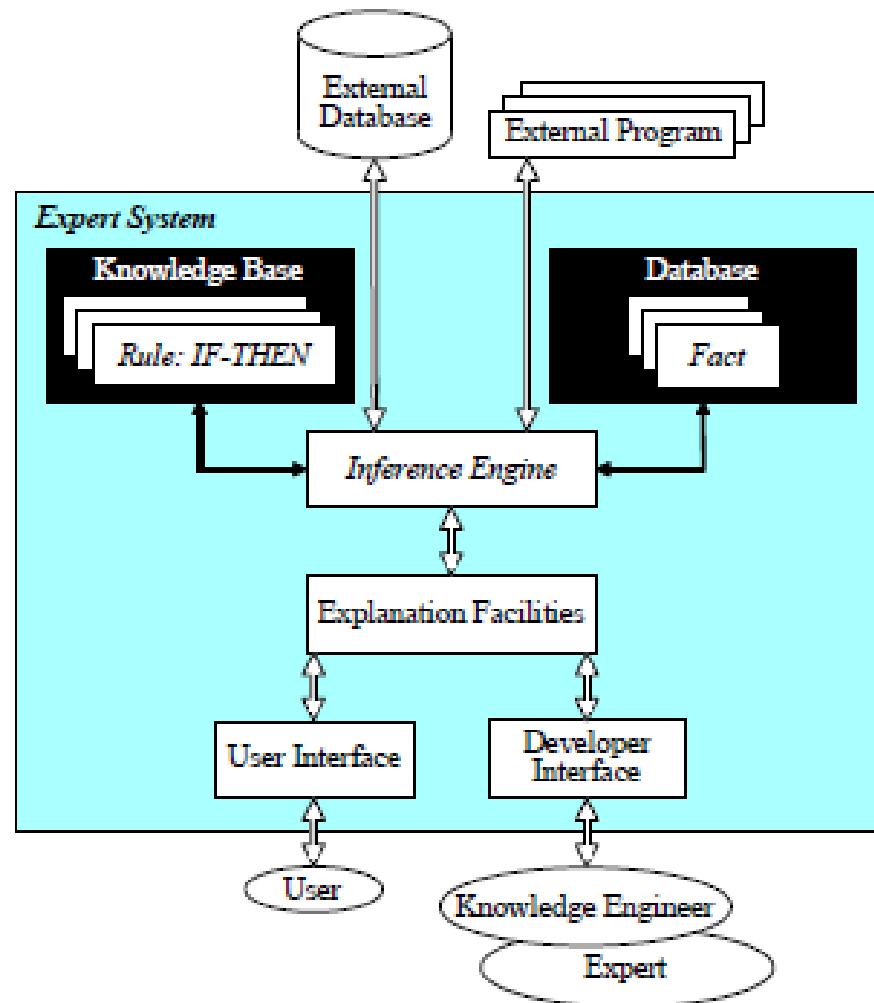
Componentes de un Sistema experto

- **La base de conocimiento** contiene el **conocimiento** util para resolver un problema. En un Sistema expert **basado en reglas** , el conocimiento se **representa con un set de reglas**. Cada regla especifica una **relacion, recomendacion, directive o estrategia o heuristica** y tienen la estructura **IF (condicion) THEN (accion)**
- Cuando la parte de la **condicion se satisface** , se **dispara la regla** y la parte de la **accion es ejecutada**.
- La base de datos incluye un set de hechos usados para emparejar con la condicion IF de las reglas almacenadas en la base de conocimiento.

Componentes de un Sistema experto

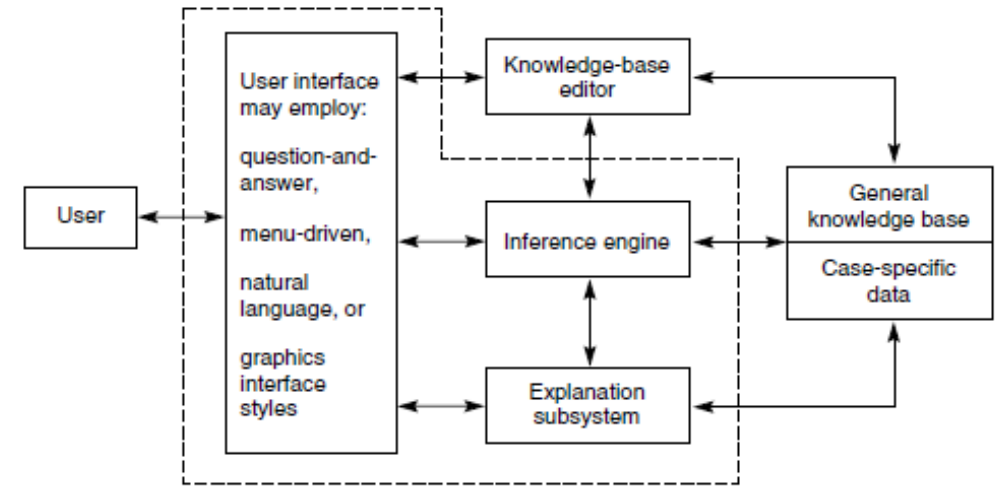
- **La maquina de inferencia** lleva a cabo el razonamiento donde los sistemas expertos alcanzan una solucion. Esto **enlaza las reglas** dadas en la base de conocimiento con los hechos provisto en la base de datos
- **El modulo de explicacion** habilita al usuario para preguntarle al Sistema expert **como una conclusion particular** is alcanzada y **por que un hecho especifico es necesario** . Un Sistema experto debe ser capaz de explicar su razonamiento y justificar su recomendacion, analisis o conclusion
- **La interfaz del usuario** es el medio de comunicacion entre un usuario buscando una solcion al un problema y el Sistema experto.

Comparacion SE anteriores y actuales

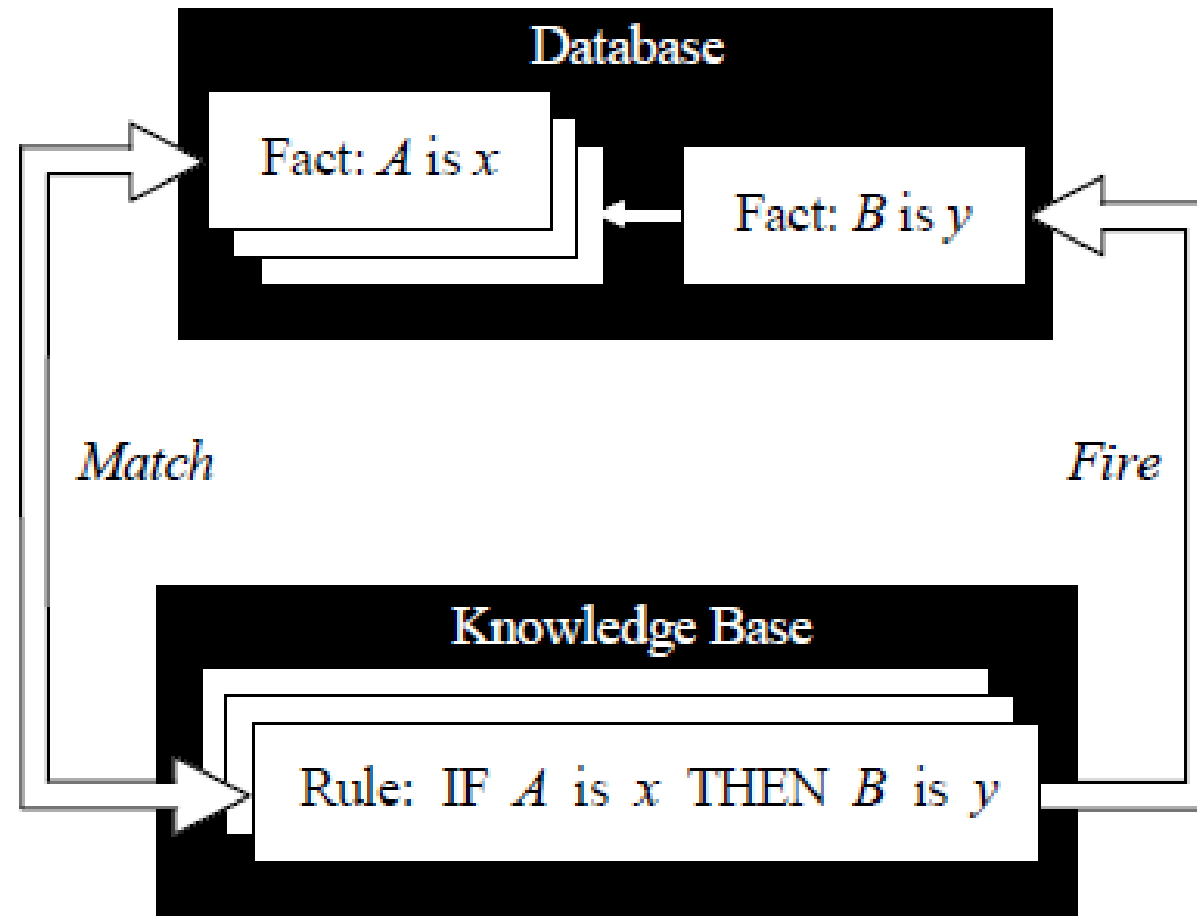


Maquina de inferencia y cadenas de inferencias

- **La maquina de inferencia** compara cada regla almacenada en el Kbase con hechos contenidos en la base general
- **Cuando la condicion IF** de una regla es igual a un **hecho** entonces la **regla se dispara y la parte then es ejecutada**
- **La similaridad de la parte if** con los hechos produce una **cadena de inferencias**. La **cadena de inferencias** indican como un Sistema expert debe aplicar las reglas para llegar a una conclusion.



Como se dispara una regla

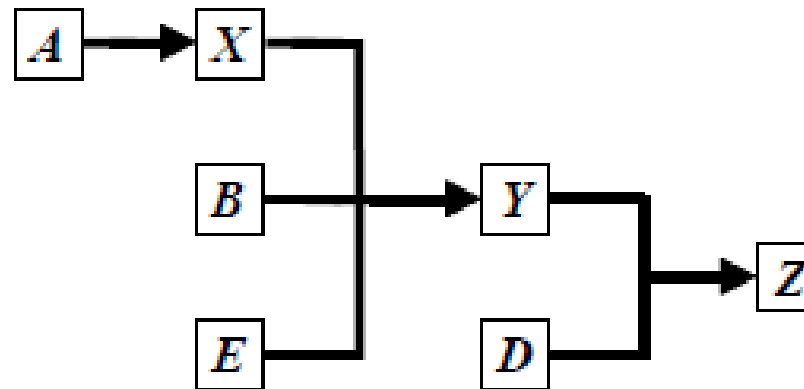


Cadena de inferencias

Rule 1: IF Y is true
AND D is true
THEN Z is true

Rule 2: IF X is true
AND B is true
AND E is true
THEN Y is true

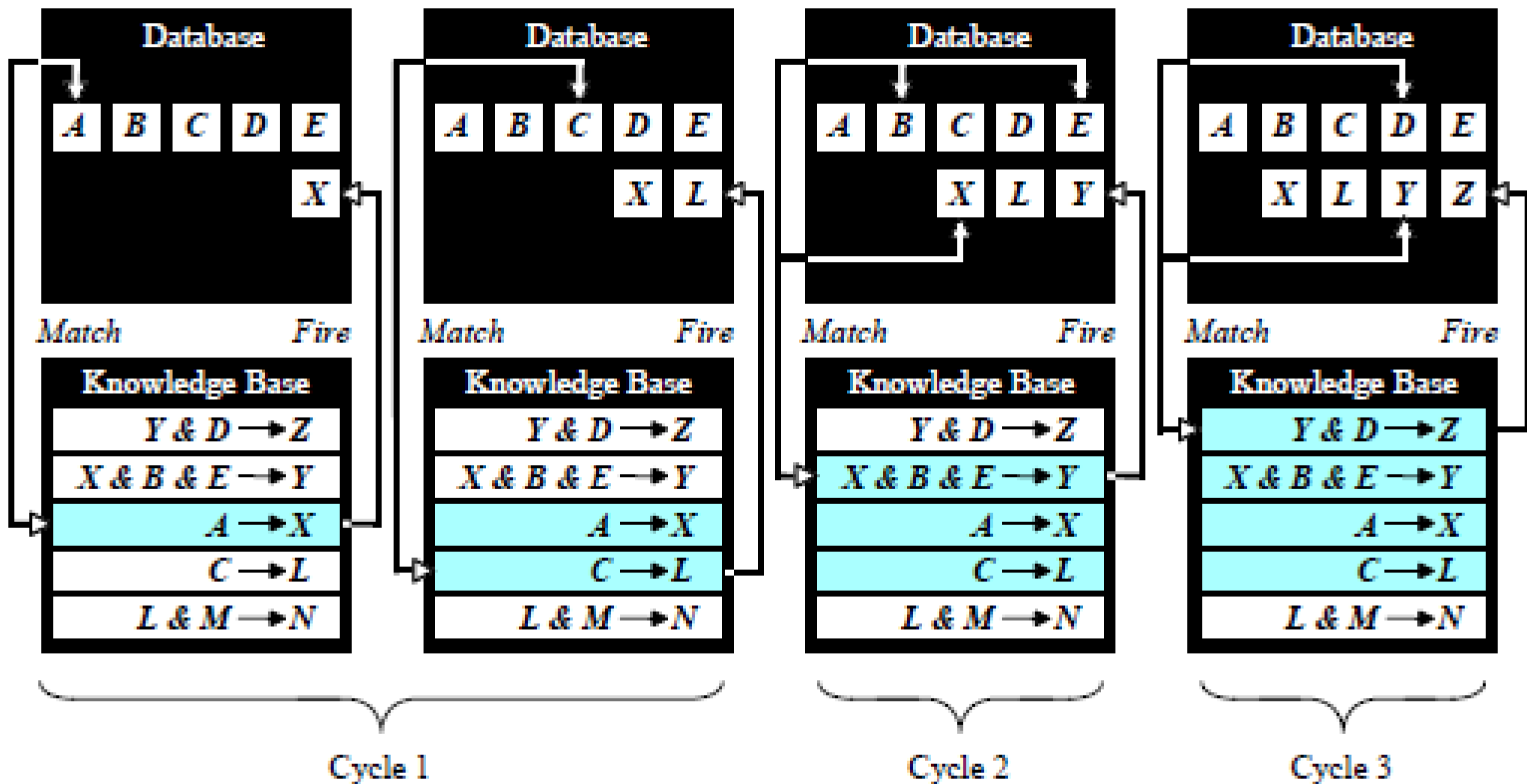
Rule 3: IF A is true
THEN X is true



Forward chaining y backward
chaining

Forward chaining

- Forward chaining es el **razonamiento manejado por datos** (data driven reasoning)
- El razonamiento **empieza de la data conocida** y procesa hacia adelante con esa data. Cada vez solo las **reglas mas altas son ejecutadas**. Cuando se disparan la regla anade un nuevo hecho en la base de datos. Caulquier regla puede **ser ejecutada solo una vez**. El ciclo de disparo de hechos iguales se detiene cuando ya no hay regla que pueda ser disparada.



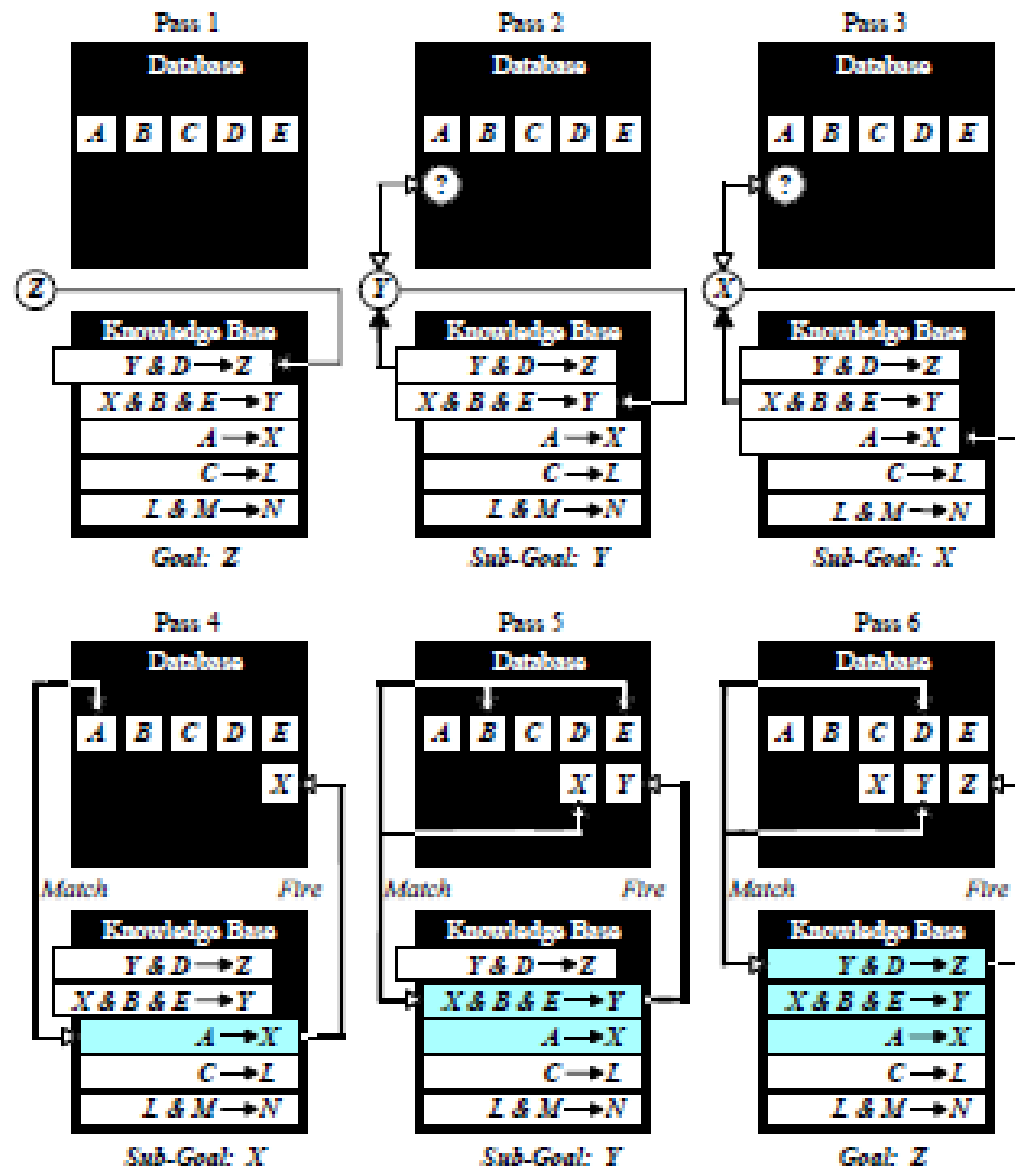
Forward Chaining

- Es una tecnica para **obtener informacion**
- Sin embargo en forward chaining **varias reglas pueden ser ejecutadas sin tener nada que ver con el objetivo buscado**
- Por lo tanto si nuestro objetivo es inferir **un hecho particular**, el forward chaining no seria eficiente.

Backward chaining

- Es el **goal driven reasoning** o razonamiento **basado en objetivos**
- Un experto tiene el objetivo (una solución hipotética) y la máquina de inferencia intenta encontrar evidencias para probarlo
- Primero la kbse es indagada para encontrar reglas que contenga la solución deseada. Esas reglas tienen el objetivo en sus secciones **Then (acción)**. Si una regla es encontrada y su if condición es igual en la base de datos entonces la regla es disparada **y el objetivo es probado.**

- **La maquina de inferencia** pone a un lado **la regla con la que esta trabajando** (se dice que se la apila (**stack**)) y se **setea un nuevo objetivo o subobjetivo** para probar la parte **if de esta regla**.
- Luego **la baseK es indagada** de nuevo para reglas que puedan probar el subobjetivo . La maquina de inferencia repite el porceso de **apilacion de las reglas** hasta que ninguna reglas es encontrada en la **Kbase para probar el subgoal**.



Como escoger entre forward o backward chaining

- **Si un expert necesita recoger informacion** y luego trata de inferir de esta lo que pueda ser inferido, se escogeria **forward chaining inference engine**
- Pero si el experto empieza **con una solucion hipotetica** y luego intenta encontrar los hechos para probarlo se **escoge el backwards chaining inference engine**

Tarea para la casa

Resolver esto en forward y en backward chaining

- Regla 1
 - IF the traffic light is green then the action is go
 - Regla 2
 - If the traffic light is red then the action is stop
 - Regla 3
 - If the traffic light is red then the action is go
-
- Cual seria el resultado en forward chaining y en backward chaining? Que regla se dispara primero, Segundo y
 - Tercero en los dos metodos. Describa lo que sucede.
 - No existen errores de escritura.

Shells para sistemas expertos basados en reglas

- <http://www.kbsc.com/rulebase.html>

Comparacion de sistemas expertos con sistemas convencionales y expertos humanos

Expertos humanos	Sistemas expertos	Programas convencionales
Uso de conocimiento en la forma de reglas de dedo gordo or heurísticas para resolver porblemas en un dominio pequeno	K procesado expresado en la forma de reglas y razonamiento simbolico para resolver problemas en un pequeno dominio	Procesa dato y usa algoritmos , una serie de operaciones definidas para reosolver problemas numericos generals
En un cerebro humano el k exite en una forma compilada	Provee una clara separacion del k y del procesamiento	No separa el conocimiento del la estrcutura de control para procesar el conocimiento
Capacidad de explicar una linea de razonamiento y proveer de detalles	Rastrea las reglas disparadas durante uns sesion de solucion de problemas y explica como una conclusion particular se alcanzo y por que cierta data especifica era necesaria	No explica como un resultado particular se btuvo y porque la informacion de entrada era necesaria

Comparacion de sistemas expertos con sistemas convencionales y expertos humanos

Expertos humanos	Sistemas expertos	Programas convencionales
Usa razonamiento inexacto y puede lidiar con informacion fuzz, incierta e incomplete	Permite el razonamiento inexacto y puede lidiar con data incomplete, incierta y fuzzy	Trabaja solamente en problemas donde la data es complete y exacta
Puede cometer errores cuando la informacion es incomplete o fuzzy	Puede cometer errores cuando la dara es incomplete o fuzzy	No proveed de ninguna solucion o una solucion errornea cuando la data es incomplete o fuzzy
Realiza la calidad de resolucion de problema a traves de anos de aprendizaje y entrenamiento practico . Es un proceso lento ineficiente y caro	Realiza la calidad del la solucion de problemas anadiendo nuevas reglas o ajustando las reglas Viejas en la base de conocimiento . Cuando nuevo conocimiento se adquiere , los cambios son faciles de lograrlos	Realiza la calidad de la resolucion de problema cambiando el codigo del programa el cual afecta el k y el procesamiento haciendo dificiles los cambios .

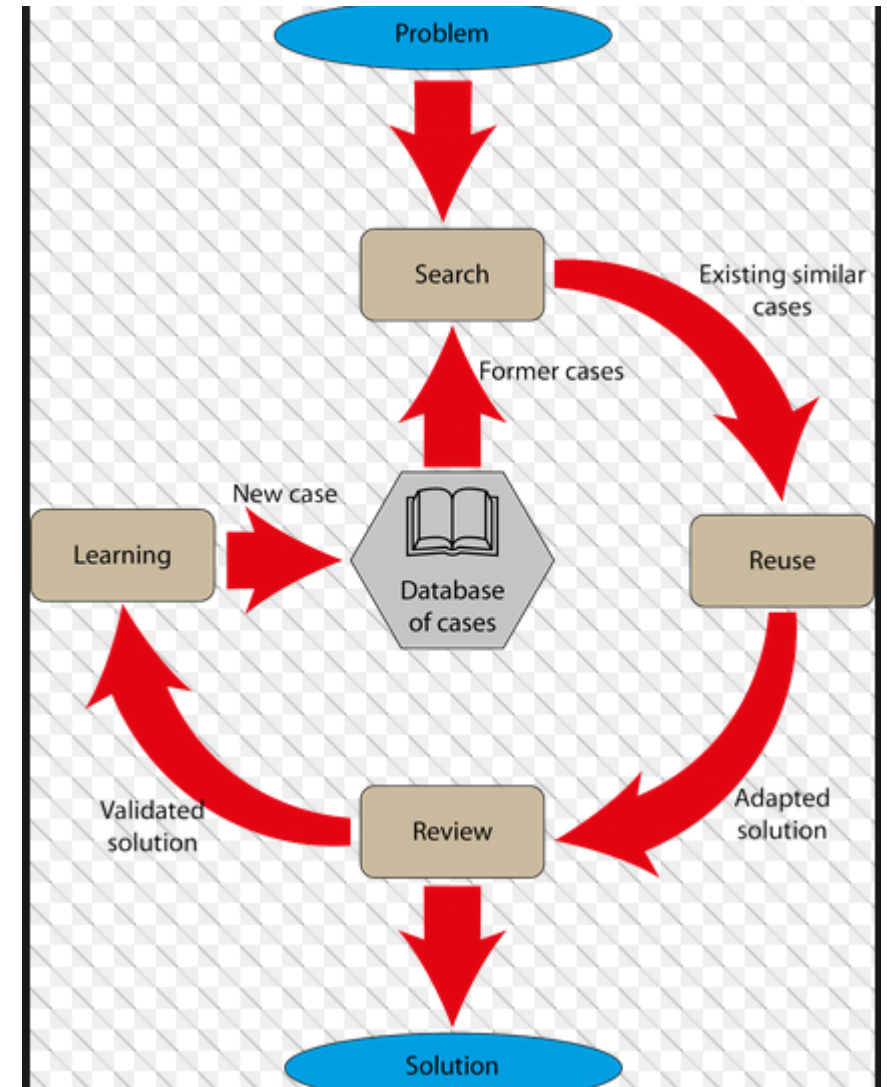
Sistemas expertos basados en casos

Que es CBR

- Case base reasoning es**recordar**(Leake 1996)
- Un **razonador basado en casos** resuelve nuevos problemas adaptando soluciones que fueron usadas para resolver viejos problemas (Riesbeck & Schank, 1989)
- Es una nueva forma de **resolver** y **aprender** (Aamodt & Plaza, 1994)
- CBR son dos cosas, la manera como **la gente usa los casos** para resolver problemas y la manera como podemos **hacer que las maquinas las usen** (Kolodner, 1993)

Sistemas expertos basados en casos

- Soporta el **reuso** de la **experiencia pasada**
- Por ejemplo en la educacion medica existe la **teoria, la anatomia, la fisiologia, las enfermedades, los casos, la experiencia de los pacientes y sus tratamientos.**
- Otro ejemplo **casos legales**
- La habilidad para razonar a partir de casos es fundamental para la **inteligencia humana.**
- El razonamiento de casos es el uso de **ejemplos de problemas pasados y sus soluciones.**
- El Case based reasoning o **razonamiento basado en casos usa una base de datos de soluciones** para direccionar nuevas soluciones de problemas.



Ejemplo de CBR

- **Sintomas**

- Eg. Engine doesn't start
- Valores medibles
- E.g. battery voltage = 6.3 V

- **Objetivo-Goal**

- Encontrar la causa por falla
 - E.g dead battery
- Y la estrategia de reparacion
 - E.g. charge battery

Casos Ejemplo

Case 1

Problem & Features

- Problem: Front light not working
- Car: VW Golf, 2.0L
- Year: 1999
- Battery voltage: 13.6V
- State of lights: OK
- State of light switch: OK

Solution

- Diagnosis: Front light fuse defect
- Repair: Replace front light fuse

Case 2

Problem & Features

- Problem: Front light not working
- Car: Passat
- Year: 2000
- Battery voltage: 12.6V
- State of lights: surface damaged
- State of light switch: OK

Solution

- Diagnosis: Bulb defect
- Repair: Replace front light

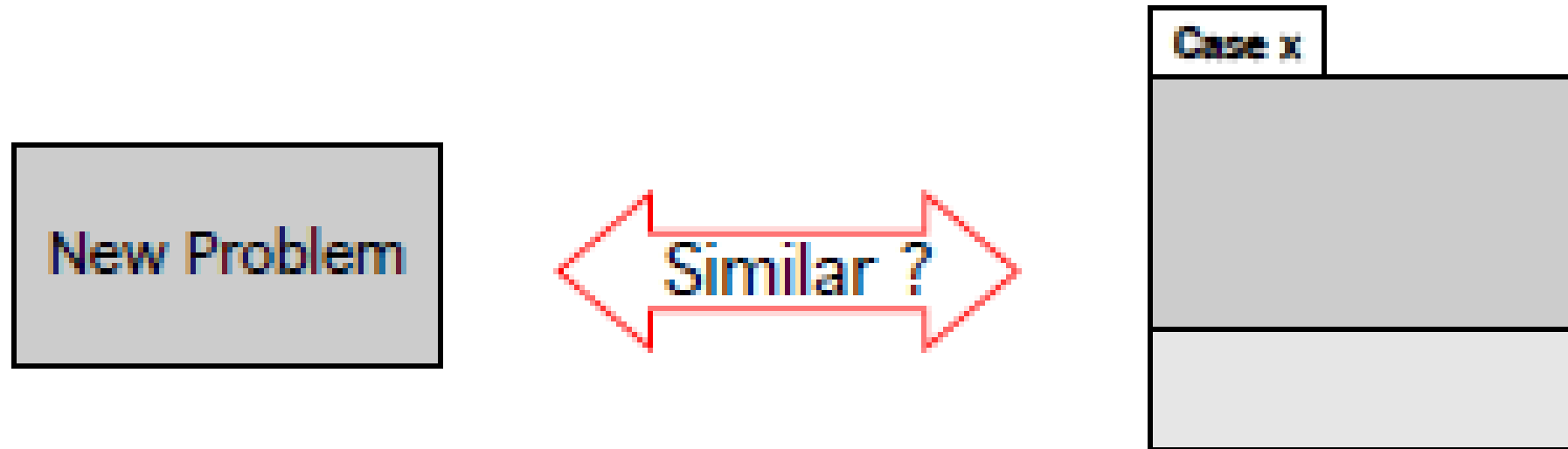
*Lo que tengo guardado en la base de K

Nuevo problema

- Y se presenta un nuevo problema
- No todas las características son conocidas
- El problema nuevo no tiene solución

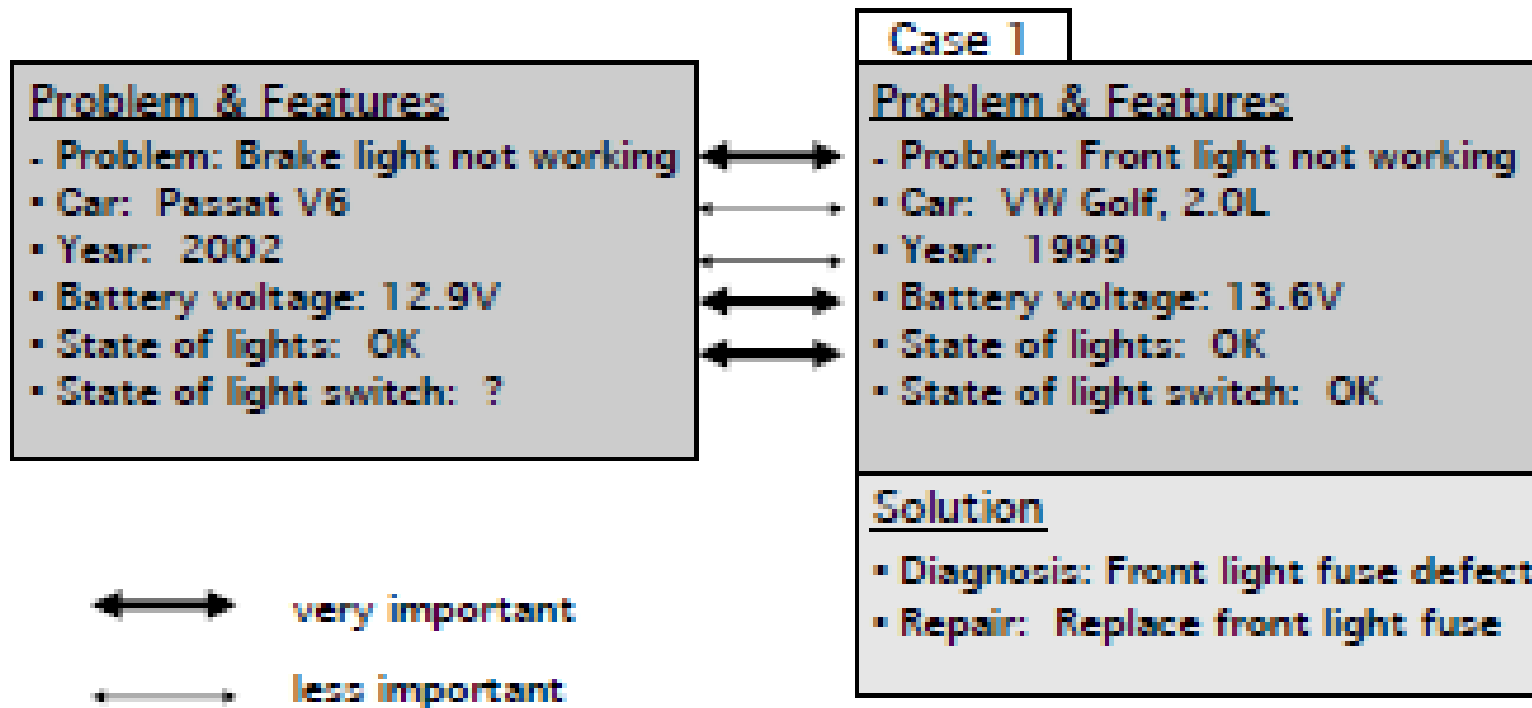
Problem & Features

- Problem: Brake light not working
- Car: Passat V6
- Year: 2002
- Battery voltage: 12.9V
- State of lights: OK
- State of light switch: ?



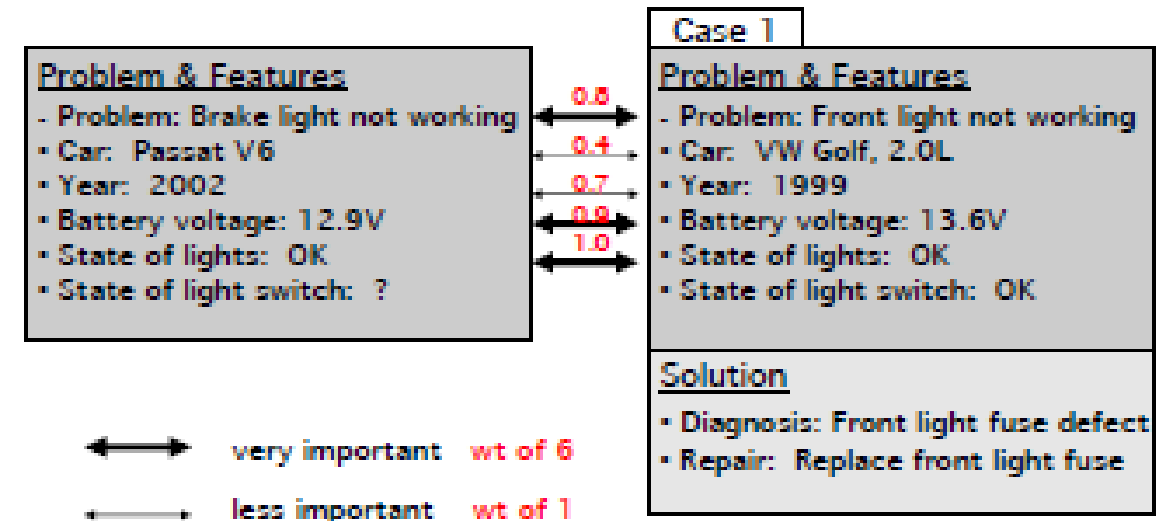
Encontrar un caso similar

Comparar caso por caso



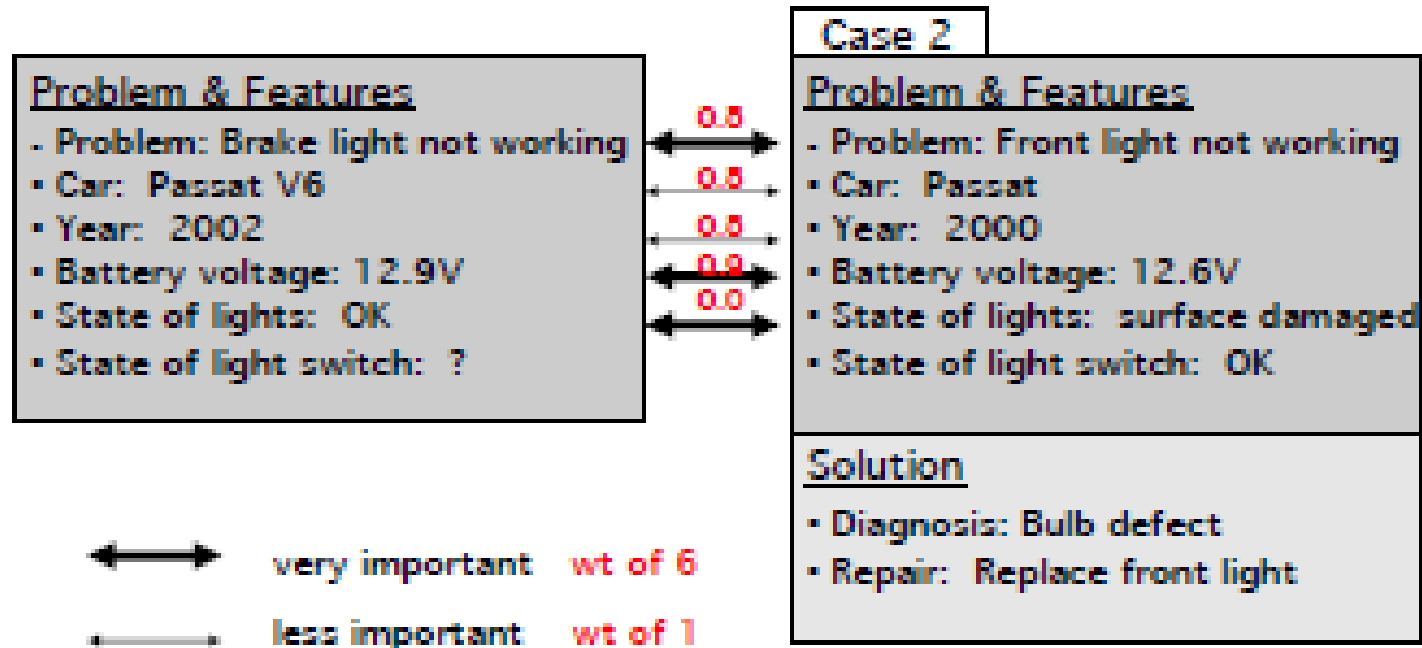
Comparamos caso 1

- Necesitamos definir cuales son las características mas importantes y luego asignarles un valor
- Por ejemplo en este caso 6 para el mas importante, 1 para el menos importante.
- Luego medimos la similaridad de características .
- Mientras mas cercano al 0 son menos similares, mientras mas cercanos al 1 son mas similares
- Identificamos el factor $1/n$ donde n es la suma de las importancias de cada factor (6+1+1+6+6). El factor es $1/20$.



Similarity by wted avg = $1/20 (6*0.8 + 1*0.4 + 1*0.7 + 6*0.9 + 6*1.0) = 0.87$

Comparamos caso 2

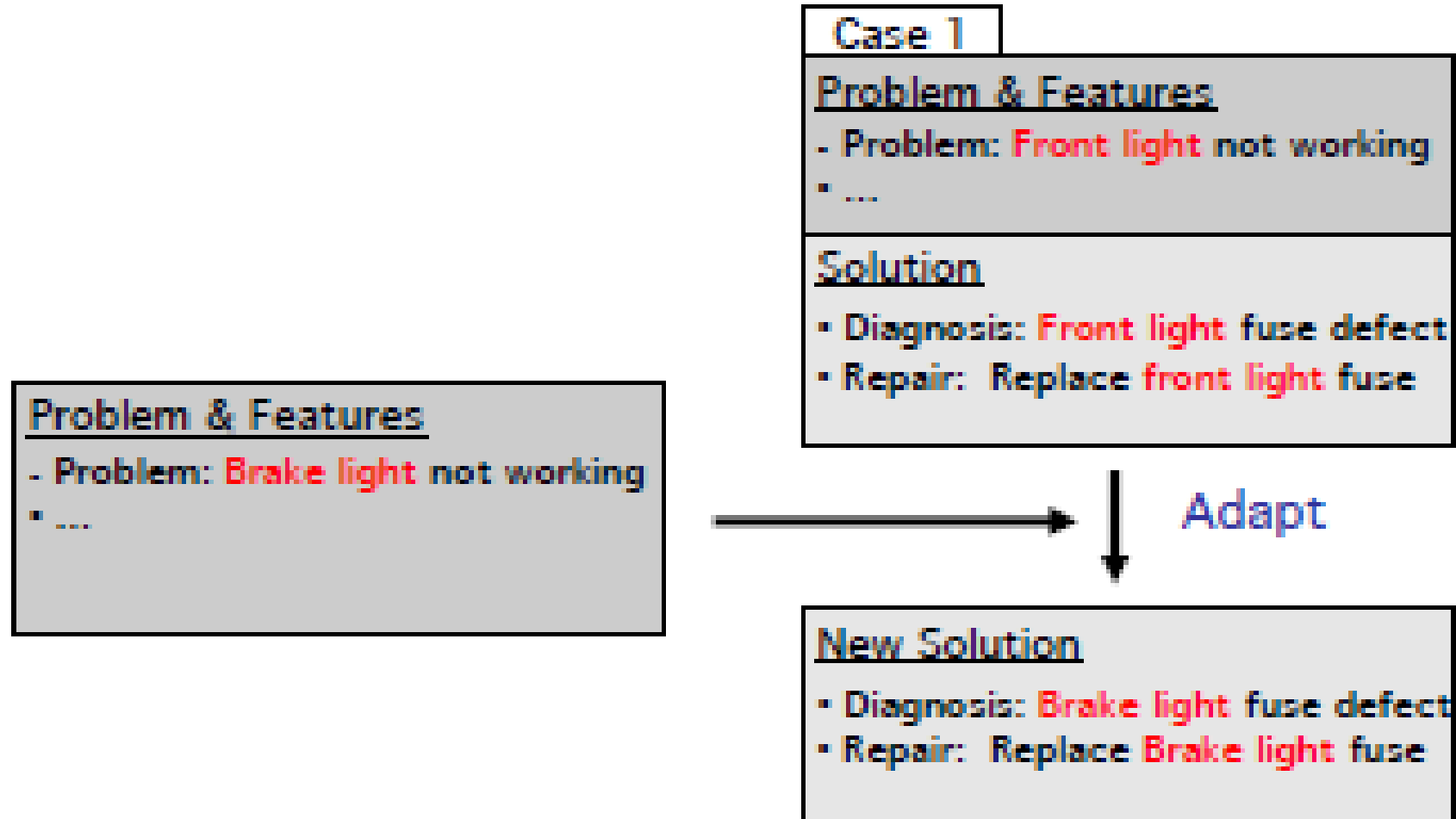


Similarity by wted avg = $1/20 (6*0.8 + 1*0.8 + 1*0.8 + 6*0.9 + 6*0.0) = 0.59$

Selección del caso

- Comparación con caso 1= 0,87
- Comparación con caso 2= 0,59
- Es más similar al caso 1

Reuso del caso



Almacenaje del nuevo caso

Case 3

Problem & Features

- Problem: Brake light not working
- Car: Passat V6
- Year: 2002
- Battery voltage: 12.9V
- State of lights: OK
- State of light switch: OK

Solution

- Diagnosis: Brake light fuse defect
- Repair: Replace brake light

Diagnostico basado en casos

- Un caso que representa una situacion de diagnostico
 - Sintomas
 - Fallas y causas
 - Valores de las características
 - Estrategias de reparacion o de accion
 - Salida: implicita falla arreglada
- Resolucion de problema
 - Almacenar la coleccion de los casos
 - Encontrar un caso similar , adaptarlo si es necesario, sugerir estrategias de reparacion
- Aprendizaje
 - Observar la salida , almacenar el nuevo caso

CBR fases

- Retrieve
- Reuse: adaptar / reparar
- Almacenar /aprender

Retrieve

- Como hago el retrieve
 - Seleccionando indices
 - Determinando las lecciones utiles
 - Describiendo circunstancias donde seria util
 - Representando esas circunstancias (caracteristicas)

Reuse: adaptar /reparar

- Adaptacion via:
 - Sustitucion
 - Ajustes de parametros
 - Busquedas locales (reemplazar frutas en una receta)

Store/Learn

- Acumulando casos
- Asignando y desasignando indices.

Tarea

- Cree tres casos con sus características (10) de un dominio particular seleccionado por usted.
- Identifique un nuevo caso sin resolver, pero que sea del mismo dominio particular
- Calcule la similitud de los casos con el caso propuesto
- Resuelva el problema utilizando los 3 casos propuestos.