

CAPÍTULO 4

METODOLOGÍA DE DESARROLLO DE LOS SISTEMAS EXPERTOS

4.1 Factores y condiciones que favorecen la implantación de los sistemas expertos.

4.1.1 Cuestionamientos para determinar las probabilidades de éxito de un sistema experto, según González & Dankel.

El problema de un sistema experto puede aparecer en etapas muy tempranas, en la decisión de hacerlo o no hacerlo.

González & Dankel (1993) ofrecieron una serie de cuestionamientos que permiten detectar el probable éxito de un proyecto de desarrollo de sistemas expertos; ellos planteaban que un sistema experto debe ser, ante todo, necesario y posible. Para llegar a conocer esa información, era necesario plantear las siguientes preguntas:

- a) ¿La aplicación del sistema experto es sobre un problema cuya solución está basada en el

Se muestra a continuación la tabla 4, que ilustra la secuencia del proceso de inferencia con este método, para nuestro ejemplo.

**Tabla 5.
SECUENCIA DEL PROCESO DE INFERENCIA MEDIANTE FORWARD CHAINING.**

Supongamos que tenemos que resolver la siguiente cuestión: ¿Si tengo una impresora HP_LaserJet, qué suministro requiero?				
Secuencia	Regla	Objetivo	Memoria de trabajo	Acción
	Rule 1 If impresora = laser Then calidad = alta.	modelo como antecedente	modelo = HP_LaserJet	No aplica al objetivo; a la siguiente regla.
	Rule 2 If impresora = laser Then precio = alto.	modelo como antecedente	modelo = HP_LaserJet	No aplica al objetivo; a la siguiente regla.
	Rule 3 If impresora = inkjet Then calidad = alta.	modelo como antecedente	modelo = HP_LaserJet	No aplica al objetivo; a la siguiente regla.
	Rule 4 If impresora = inkjet Then precio = alto.	modelo como antecedente	modelo = HP_LaserJet	No aplica al objetivo; a la siguiente regla.
1	Rule 5 If modelo = HP_LaserJet Then impresora = laser.	modelo como antecedente	modelo = HP_LaserJet	Agrega a la memoria de trabajo un hecho nuevo.
	Rule 6 If modelo = HP_DeskJet Then impresora = inkjet.	modelo o impresora como antecedente	modelo = HP_LaserJet; impresora = laser	Si aplica al objetivo pero la evaluación falla.
2	Rule 7 If impresora = laser Then suministro = torner .	modelo o impresora como antecedente	modelo = HP_LaserJet; impresora = laser	Se encuentra suministro como consecuencia, y puede deducirse de los valores que se tienen en la memoria de trabajo. suministro = torner Fin de la búsqueda.
	Rule 8 If impresora = inkjet Then suministro = Cartucho_tinta.			

4.5.2.2.2 Backward Chaining

En Backward chaining la deducción está basada en los objetivos, específicamente en los elementos que componen la conclusión. La inferencia comienza buscando aquellas reglas en donde la solución del problema actúe como consecuencia.

En backward chaining no se tienen valores de inicio en la memoria de trabajo, por lo que sería recomendable procurar la obtención de ciertos datos con anterioridad a otros; algunos programas con sofisticados mecanismos de inferencia permiten que el desarrollador especifique prioridades de ejecución a las reglas. No obstante a ello, la filosofía del método persiste, ya que las prioridades sólo determinan un nuevo orden de evaluación de reglas, y no una nueva lógica de trabajo. La figura 26 ilustra la forma en que trabaja el método de encadenamiento hacia atrás (*backward chaining*).

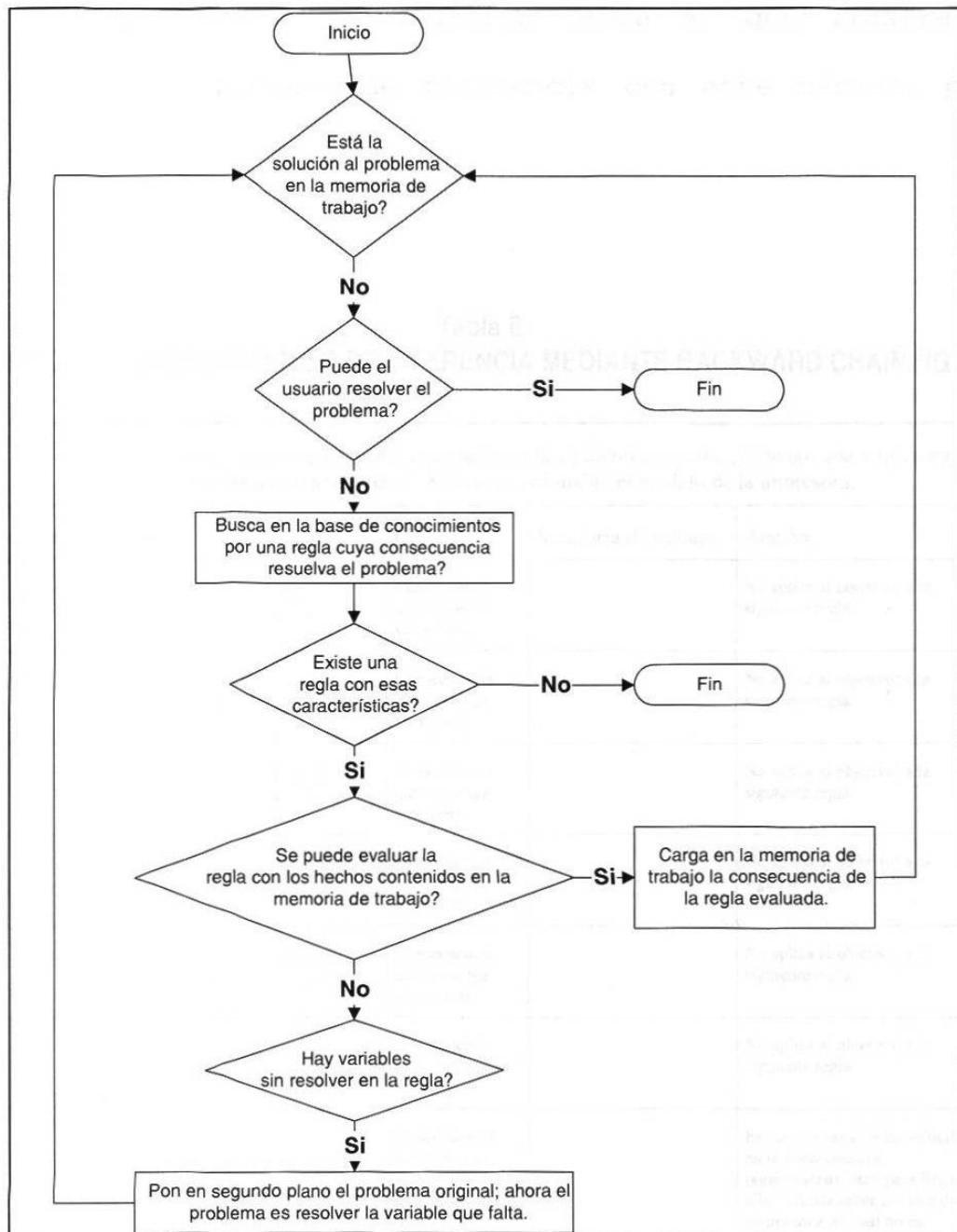


Figura 26: Forma en que trabaja el método de encadenamiento hacia atrás (backward chaining).

Se muestra a continuación tabla 5, que ilustra la secuencia del proceso de inferencia con este método, para nuestro ejemplo.

**Tabla 6.
SECUENCIA DEL PROCESO DE INFERENCIA MEDIANTE BACKWARD CHAINING.**

Supongamos que tenemos que resolver la siguiente cuestión: ¿Si tengo una impresora, qué suministro requiero? Sólo es cuestionable el modelo de la impresora.				
Secuencia	Regla	Objetivo	Memoria de trabajo	Acción
	Rule 1 If impresora = laser Then calidad = alta.	Consecuencia que contenga suministro		No aplica al objetivo; a la siguiente regla.
	Rule 2 If impresora = laser Then precio = alto.	Consecuencia que contenga suministro		No aplica al objetivo; a la siguiente regla.
	Rule 3 If impresora = inkjet Then calidad = alta.	Consecuencia que contenga suministro		No aplica al objetivo; a la siguiente regla.
	Rule 4 If impresora = inkjet Then precio = alto.	Consecuencia que contenga suministro		No aplica al objetivo; a la siguiente regla.
	Rule 5 If modelo = HP_LaserJet Then impresora = laser.	Consecuencia que contenga suministro		No aplica al objetivo; a la siguiente regla.
	Rule 6 If modelo = HP_DeskJet Then impresora = inkjet.	Consecuencia que contenga suministro		No aplica al objetivo; a la siguiente regla.
1	Rule 7 If impresora = laser Then suministro = toner .	Consecuencia que contenga suministro		Encuentra variable de solución en la consecuencia (suministro), pero para llegar a ella necesita saber el valor de impresora , el cual no es cuestionable. Suspende el proceso de inferencia para encontrar suministro . Ahora el objetivo es encontrar impresora .
	Rule 8 If impresora = inkjet Then suministro = Cartucho_tinta.			

Secuencia	Regla	Objetivo	Memoria de trabajo	Acción
	Rule 1 If impresora = laser Then calidad = alta.	Consecuencia que contenga impresora.		No aplica al objetivo; a la siguiente regla.
	Rule 2 If impresora = laser Then precio = alto.	Consecuencia que contenga impresora.		No aplica al objetivo; a la siguiente regla.
	Rule 3 If impresora = inkjet Then calidad = alta.	Consecuencia que contenga impresora.		No aplica al objetivo; a la siguiente regla.
	Rule 4 If impresora = inkjet Then precio = alto.	Consecuencia que contenga impresora.		No aplica al objetivo; a la siguiente regla.
1.1	Rule 5 If modelo = HP_LaserJet Then impresora = laser.	Consecuencia que contenga impresora.		<p>Encuentra variable de solución en la consecuencia (impresora), pero para llegar a ella necesita saber el valor de modelo, el cual es cuestionable.</p> <p>El usuario selecciona como valor correcto "HP_LaserJet".</p> <p>La regla cumple por verdadero, entonces agrega los hechos a la memoria de trabajo.</p> <p>Regresa al control de la inferencia a la variable que tratábamos de resolver desde un principio: suministro.</p>
	Rule 6 If modelo = HP_DeskJet Then impresora = inkjet.			
	Rule 7 If impresora = laser Then suministro = torner .			
	Rule 8 If impresora = inkjet Then suministro = Cartucho_tinta.			

Supongamos que tenemos que resolver la siguiente cuestión: ¿Si tengo una impresora, qué suministro requiero? Sólo es cuestionable el modelo de la impresora.

Secuencia	Regla	Objetivo	Memoria de trabajo	Acción
	Rule 1 If impresora = laser Then calidad = alta.			
	Rule 2 If impresora = laser Then precio = alto.			
	Rule 3 If impresora = inkjet Then calidad = alta.			
	Rule 4 If impresora = inkjet Then precio = alto.			
	Rule 5 If modelo = HP_LaserJet Then impresora = laser.			
	Rule 6 If modelo = HP_DeskJet Then impresora = inkjet.			
1	Rule 7 If impresora = laser Then suministro = torner .	Consecuencia que contenga suministro	impresora = laser	Regresa a la regla en donde necesitábamos el valor de impresora ; ya que lo tenemos, podemos resolver nuestro problema. suministro = torner
	Rule 8 If impresora = inkjet Then suministro = Cartucho_tinta.			

4.5.3 Diferencias entre Backward y Forward Chaining.

Paul Harmon & Rex Maus hacen notar que existen elementos en Backward chaining que no están presentes en Forward chaining:

- En Backward chaining no existen observaciones o hechos sino hasta que son solicitados por el sistema experto. En consecuencia, al iniciar la inferencia la memoria de trabajo está vacía. Un dato importante es que no todos los datos son cuestionables dentro de la inferencia.
- En Backward chaining la inferencia siempre comienza con la evaluación de las consecuencias (objetivos); cuando se evalúa un objetivo, el mecanismo de inferencia siempre busca en tres lugares, y en el siguiente orden:

a) **Memoria de trabajo.**- Contiene las cosas que el sistema experto "sabe" actualmente.

b) **El usuario.**- El usuario puede interactuar con el sistema experto para darle pistas (hechos cuestionables) que lo ayuden a resolver el problema. Por

lo general, se hará la pregunta (ASK), misma que ya tiene clasificadas las respuestas (CHOICES) válidas para el sistema. La respuesta "No se" debe estar presente dentro de las posibilidades de cada hecho concreto.

c) **Consecuencia de las reglas.**- Busca primero en la reglas que contienen la probable solución del problema dentro de su consecuencia. (THEN VariableAResolver = ...). En este caso, la primera regla en evaluar (en orden de arriba abajo) será la que contenga la variable buscada.

4.6 Representación esquemática del conocimiento.

4.6.1 Fórmula del conocimiento.

Para poseer el conocimiento se debe saber el significado de los elementos que lo componen, adicional al propósito que tiene el conocerlo, y saber cómo puede ser de utilidad. Si tuviéramos que representar gráficamente al conocimiento, podría ser tal y como lo ilustra la figura 27:

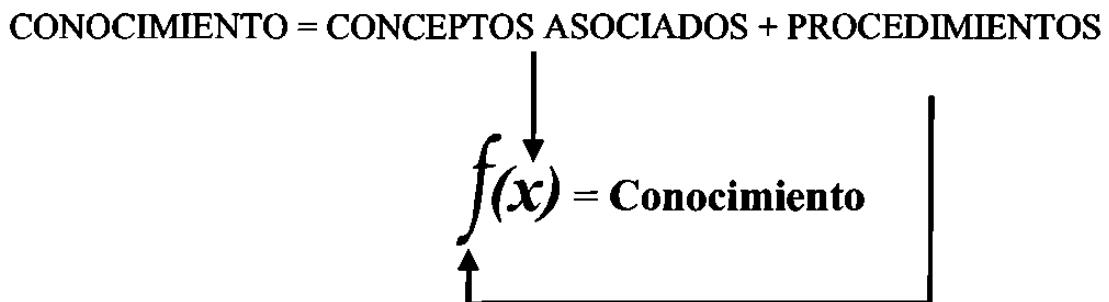


Figura 27: Fórmula del conocimiento.

Para que una persona sea considerada experta, debe conocer los conceptos asociados a una rama específica del conocimiento y saber cómo utilizarlos para obtener un determinado resultado.

4.6.2 Conocimientos y habilidades del experto.

Un experto siempre conoce:

- Un plan de cómo llegar a la solución de un problema
- Los conceptos asociados al problema, en su individualidad y como parte de un entorno (rol de los conceptos)
- Métodos específicos para calcular o derivar conceptos asociados desconocidos, relativos al problema.
- Reglas de juicio y criterio personal para evaluar la evidencia obtenida en el análisis del problema y concluir en una probable solución.

4.6.3 Tipos de conocimiento en base a su naturaleza.

Como se puede apreciar, aquellas cosas que debe conocer el experto tienen diferente naturaleza; de ahí surge la necesidad de tipificar el conocimiento, derivándose los tipos que se muestran en la figura 28, propuestos por Mora Tabares (1993):

Estratégico: Etapas para la solución de un problema	Taxonómico: Clasificación de los conceptos
Procedural: Métodos para el cálculo o derivación de valores	Evaluativo: Reglas para evaluar la evidencia y concluir una solución.

Figura 28: Tipos de conocimiento en base a su naturaleza.

Para poder tratar las particularidades de cada uno de estos tipos, nos valdremos de un ejemplo corto, mismo que será abordado sólo para fines explicativos.

Suposición:

Suponga que un analista financiero tiene la tarea de interpretar la situación financiera de una compañía a través del análisis de los estados financieros e información adicional.

a) Conocimiento estratégico

Muestra las etapas, en términos generales, para la solución del problema. A este nivel no se particulariza. Un ejemplo de dicho conocimiento se ilustra en la figura 29:

- 1.- Obtener estados financieros, indicadores financieros del sector y otros datos de tipo cualitativo y cuantitativo.
- 2.- Realizar cálculos de indicadores financieros de la empresa.
- 3.- Evaluar la situación de liquidez, su estructura de capital, operaciones y rentabilidad de manera aislada.
- 4.- analizar la evidencia conjunta para determinar la situación financiera de la empresa.
- 5.- Identificar riesgos generados por la obtención de resultados parciales en contra aunque existan resultados parciales a favor.

Figura 29: Conocimiento estratégico.

b) Conocimiento taxonómico.

Muestra la clasificación de los conceptos involucrados en el problema. Permite una definición formal y elimina discrecionalidades. Un ejemplo de dicho conocimiento se ilustra en la figura 30:

```
(CONCEPTO razones_financieras
  (ES UN TIPO DE (indicadores_financieros))
  (SE CLASIFICAN EN (razones_de_liquidez, razones_de_rentabilidad,
                      razones_de_operación, razones_de_estructuras_de_capital))
  (SE USAN EN (proceso_de_evaluación_financiera))
)

(CONCEPTO razones_de_liquidez
  (ES UN TIPO DE (razones_financieras))
  (SE CLASIFICAN EN (razón_circulante, prueba_del_ácido))
  (SE USAN EN (proceso_de_evaluación_de_liquidez))
)
(CONCEPTO razón_circulante
  (ES UN TIPO DE (razones_de_liquidez))
  (SE CLASIFICAN EN (Sin_clasificación))
  (SE CALCULA CON (fórmula_para_calcular Razón_circulante))
  (DATOS REQUERIDOS (activo_circulante, pasivo_circulante))
  (RANGO DE VALORES OK (1.2,...,3.5))
  (RANGO DE VALORES PROBLEMATICOS (0.1,...,1.1))
  (RANGO DE VALORES EXTRÁÑOS (3.6,...,10.0))
)
(CONCEPTO activo_circulante
  (ES UN TIPO DE (activo))
  (SE CLASIFICAN EN (efectivo, clientes, inventario, otros_activos_circulantes))
  (SE CALCULA CON (formula_para_calcular activo_circulante))
)
```

Figura 30: Conocimiento taxonómico.

c) Conocimiento procedural.

Muestra los métodos para el cálculo o derivación de valores. Un ejemplo de dicho conocimiento se ilustra en la figura 31:

```
PROCESO fórmula_para_calcular_razón_circulante (razón_circulante NUMERICO) {
    razón_circulante = activo_circulante / pasivo_circulante
}

PROCESO formula_para_calcular_activo_circulante (activo_circulante NUMERICO) {
    activo_circulante = efectivo + clientes + inventario +
        otros_activos_circulantes
}
```

Figura 31: Conocimiento procedural.

d) Conocimiento evaluativo.

Reglas para evaluar la evidencia y concluir una probable solución al problema. Un ejemplo de dicho conocimiento se ilustra en la figura 32:

```
(SI la razón de liquidez es menor que 1.0 ENTONCES CONCLUYA (
    (la situación de liquidez es muy crítica)
    (ESCRIBIR lista_de_riesgos "Existen serios problemas de liquidez")
)

(SI la razón de liquidez esta entre 1.0 y 1.5 ENTONCES CONCLUYA (
    (la situación de liquidez es aceptable)
    (ESCRIBIR lista_de_riesgos "Poco riesgo en liquidez")
)

(SI la razón de liquidez esta entre 1.6 y 3.5 ENTONCES CONCLUYA (
    (la situación de liquidez es muy favorable)
    (ESCRIBIR lista_de_riesgos "No existe riesgo en liquidez")
)

(SI la razón de liquidez es mayor que 3.5 ENTONCES CONCLUYA (
    (la situación de liquidez es extraña)
    (ESCRIBIR lista_de_riesgos "Cuestionas sobre la alta liquidez")
)
```

Figura 32: Conocimiento evaluativo.

4.7 Manejo y cálculo de la incertidumbre.

4.7.1 Factores de certeza.

Uno de las máximas capacidades de los sistemas expertos es la de manejar la incertidumbre, mediante los denominados factores de certeza o confianza (CNF /

Certainty Factor). Este rasgo diferencial despegó a los sistemas expertos de otros tipos de herramientas para encontrar solución a problemas, tales como la programación lineal u otros métodos de tipo matemático.

Para cada diferente herramienta de solución de problemas, es necesario determinar el escenario sobre el cual el problema requiere ser resuelto; todos tienen datos de entrada, siguen un proceso y proporcionan una salida esperada. ¿Qué sucede si alguno de los elementos determinantes de la solución del problema no se conoce con exactitud?; la respuesta es igual para casi todas las herramientas: los datos son insuficientes, y las herramientas por sí mismas no disponen de los mecanismos adecuados para generar el dato que requieren.

Los sistemas expertos no caen dentro de ese universo de herramientas, ya que manejan diferentes tipos de problemas, entre los que destacan aquellos que incluyen elementos de alta discrecionalidad, en los que un factor determinante para la solución de un problema no es conocido o es presentado en forma vaga. Supóngase que para solución de un problema se requiera conocer las dimensiones de una determinada cosa; suponga que las dimensiones son desconocidas, o bien se proporciona como respuesta al cuestionamiento de la dimensión un calificativo, tal como

"grande", lo cual es vago para aquellos planteamientos que requieren exactitud; en tal caso, la herramienta deberá ser capaz de determinar por sí misma los datos omitidos o vagos, haciendo uso de la heurística y datos estadísticos de conocimiento generalmente aceptado, o bien extraídos de un acervo de casos. No se debe olvidar que dentro de las respuestas que un usuario puede manifestar puede estar el "no se".

No debe confundirse a los factores de certeza como producto de la estadística; el concepto obedece un poco más a la seguridad en un conocimiento para la toma de decisiones; la consideración estadística de los factores de certeza llevaría a complicar el uso de los sistemas expertos debido a la demanda de información en muchos casos no disponible o fácilmente comprobable por el usuario. Derivado de esto, existen 2 tipos de factores de certeza.

- a) **CNFe (Factor de certeza del Experto).**- Es el que un experto sugiere en la conclusión. En caso de omisión de CNFu, este factor es considerado como plenamente válido.
- b) **CNFu (Factor de certeza del Usuario).**- Es el que el usuario especifica al responder a un cuestionamiento.

"grande", lo cual es vago para aquellos planteamientos que requieren exactitud; en tal caso, la herramienta deberá ser capaz de determinar por sí misma los datos omitidos o vagos, haciendo uso de la heurística y datos estadísticos de conocimiento generalmente aceptado, o bien extraídos de un acervo de casos. No se debe olvidar que dentro de las respuestas que un usuario puede manifestar puede estar el "no se".

No debe confundirse a los factores de certeza como producto de la estadística; el concepto obedece un poco más a la seguridad en un conocimiento para la toma de decisiones; la consideración estadística de los factores de certeza llevaría a complicar el uso de los sistemas expertos debido a la demanda de información en muchos casos no disponible o fácilmente comprobable por el usuario.

Derivado de esto, existen 2 tipos de factores de certeza.

- a) **CNFe (Factor de certeza del Experto).**.- Es el que un experto sugiere en la conclusión. En caso de omisión de CNFu, este factor es considerado como plenamente válido.
- b) **CNFu (Factor de certeza del Usuario).**.- Es el que el usuario especifica al responder a un cuestionamiento.

4.7.2 Diversidad de apreciaciones y conocimientos que hacen necesario el uso de los factores de certeza.

Veamos el siguiente ejemplo; supóngase que un sistema experto tiene entre su base de conocimiento lo siguiente.

REGLA 1:

If animal_pone_huevos <> "no" and

animal_vuela = "sí" then

Animal = "ave", CNF 95%

Endif

En esta regla podemos observar que el experto, probablemente un biólogo, estimó que si el animal pone huevos y vuela, existe un 95% de probabilidades de que el animal sea "Ave". Obviamente, dentro de su conocimiento tiene bien diferenciado lo que puede catalogarse como un "huevo", y lo que considera "poner un huevo". Se debe prestar atención como la regla no hace una igualación directa en "animal_pone_huevos" a sí, pregunta por una respuesta diferente a "no", con lo que se hace más dinámica

la regla para aceptar un "no se" como respuesta y determinar conclusión de todas formas.

Supóngase que el usuario del sistema experto lo utiliza; las variables "animal_pone_huevos" y "animal_vuela" no están resueltas en la memoria de trabajo. Al evaluarse las reglas, el sistema cuestiona al usuario:

RESPUESTAS 1:

¿El animal pone huevos?: "sí"

¿El animal vuela?: "sí", CNF 60%

En nuestro ejemplo, el usuario sabe que el animal pone huevos y contesta que "sí" a la pregunta de que si el animal vuela, con un 60% de factor de certeza; esto puede ocurrir por diversas razones: tal vez el usuario no sabe a ciencia cierta cuántas de las aves vuelan, y estima que sólo vuela un 60% de ellas. Como se puede ver, el factor de certeza puede ser diferente entre el experto y el usuario.

Otra diferencia que puede darse con respecto a la seguridad de la información proporcionada al sistema experto tiene que ver con las reglas mismas que están contenidas en la base de conocimiento.

Supóngase que en lugar de las respuestas brindadas en RESPUESTAS 1, el usuario contestara lo siguiente:

RESPUESTAS 2:

¿El animal pone huevos?: "no se"

¿El animal vuela?: "sí", CNF 60%

Con esta respuesta, la REGLA 1 se resuelve favorablemente, pero puede no ser un dato exacto; para ello, otra regla dentro de la base del conocimiento puede complementar la conclusión:

REGLA 2:

```
If animal_pone_huevos <> "no se" and  
    animal_vuela = "sí" then
```

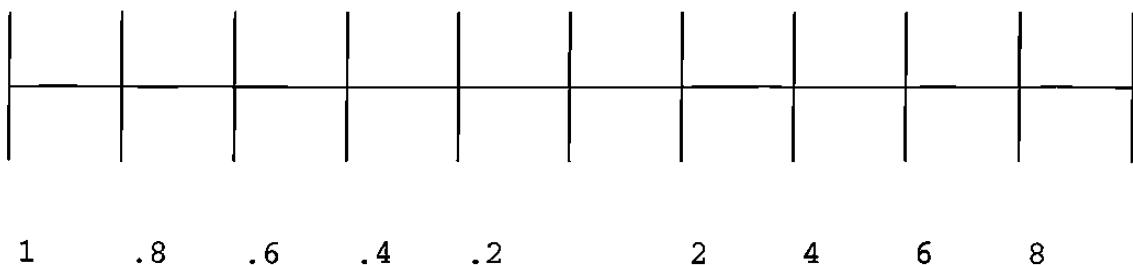
Animal = "ave", CNF 35%

Endif

El hecho de no saber si el animal pone huevos o no, compromete seriamente la certeza de que el animal sea un ave, ya que son muchos los insectos que vuelan.

4.7.3 Escala de factores de certeza desarrollada en la universidad de Stanford.

Para poder resolver estos predicamentos en cuanto a la certeza, la Universidad de Stanford desarrolló una escala de factores de certeza, que es como sigue:



Donde:

- 1 Definitivamente No
- .8 Muy poco probable
- .6 Probablemente No
- .4 Tal vez No
- .2 a .2 Ignorados por poco significativos
- .4 Tal vez
- .6 Es probable
- .8 Casi cierto
- 1 Definitivamente Sí

4.7.4 Cálculos que resuelven controversias en relación a los grados de certeza.

Asimismo, se definieron algunos cálculos muy sencillos para la solución de controversias relativas con los factores de certeza, aplicables a las diferencias entre Expertos - Usuarios, así como diferencias entre reglas. La simpleza de los cálculos las hace más comprensibles y sirven para el propósito para el que fueron creadas; como se citaba, un cálculo muy complejo no tiene mucho sentido.

a) Diferencia de certeza entre CNFe y CNFu.

Para la solución de diferencias entre usuarios y expertos, se emplea la siguiente fórmula:

$$(CNFe * CNFu) / 100 = CNF$$

Un detalle a tomar en cuenta es que las respuestas "no se" no deben considerarse para el cálculo de certeza, ya que CNFu es cero.

En nuestro ejemplo sería, en el caso de CNFe de REGLA 1 con CNFu en RESPUESTAS 1, variable animal_vuela:

(95 * 60) / 100 = CNF

57 = CNF

En caso de que ambas respuestas tuvieran un CNFu significativo, se realizaría la operación para cada uno de los CNFu y se determinaría un promedio.

En el caso de nuestro ejemplo, la certeza está distribuida en dos condiciones, las cuales comparten parte de la certeza de la conclusión; por concordar la segunda respuesta con la condición (la condición evalúa si animal_vuela = "sí" y la respuesta es "sí") se considera un CNFu de 1, y el cálculo sería:

animal_pone_huevos

(1 * 60) / 100 = CNF

60 = CNF

animal_vuela

(95 * 60) / 100 = CNF

57 = CNF

CNF Final para la conclusión:

Promedio(60 + 57) = CNF Final

58.5 = CNF final

Se puede comprobar que si las respuestas son totalmente ciertas para las condiciones que se evalúan, es decir CNFu siempre es 1, la aplicación de este cálculo arrojará la misma cifra que CNFe, es decir, CNFe se sostiene como válido.

b) Certeza complementaria entre reglas.

En el caso de que se detallaba, en donde una respuesta "no se" lleva a una regla a cumplir por verdadero, para después poder ajustarse en su certeza mediante la aplicación de otra regla, la fórmula que se aplica es la siguiente:

$$\text{CNF1} + \text{CNF2} - ((\text{CNF1} * \text{CNF2}) / 100) = \text{CNF}.$$

En nuestro ejemplo sería, en el caso de CNF de REGLA 1 con CNF en REGLA 2, originada por la aplicación de RESPUESTAS 2:

Regla 1:

$$(CNF_e * CNF_u) / 100 = CNF_1$$

$$(95 * 60) / 100 = CNF_1$$

$$57 = CNF_1$$

Regla 2:

animal_pone_huevos

Respuesta "no se", $CNF_u = 0$

No se toma en cuenta para promedio.

animal_vuela

$$(35 * 60) / 100 = CNF$$

$$21 = CNF$$

CNF2: Es igual al único CNF que interviene en el cálculo.

$$21 = CNF_2$$

Al aplicar la fórmula para el complemento de la confianza entre reglas, tendríamos lo siguiente:

$$\text{CNF1} + \text{CNF2} - ((\text{CNF1} * \text{CNF2})/100) = \text{CNF}.$$

$$57 + 21 - ((57 * 21)/100) = \text{CNF}.$$

$$78 - (11.97) = \text{CNF}.$$

$$66.03 = \text{CNF}.$$

Tanto en el cálculo para determinar la certeza entre $\text{CNF}_e - \text{CNF}_u$, y entre reglas que se complementan, se tiene un mismo comportamiento, que es la reducción de la certeza, derivada de la falta de certeza que un experto plantea en sus conclusiones, así como de la falta de certeza manifestada por el usuario al momento de responder a las preguntas que el sistema experto le plantea.

conocimiento humano, y dicha solución se repite continuamente?.

- b) ¿El conocimiento que se utiliza para la solución del problema es de naturaleza heurística, es decir, no es predominantemente algorítmica?.
- c) ¿El conocimiento y la experiencia utilizada para la solución del problema permanece constante, es decir, no cambia periódicamente?.
- d) ¿Hay experiencia involucrada en la solución al problema?.
- e) ¿La experiencia que resuelve el problema está bien entendida y aceptada?.
- f) ¿Son los datos de entrada para la solución al problema siempre completos y correctos, es decir, el usuario puede contestarlos en todo momento?.
- g) ¿Existe un experto dispuesto, capaz y disponible para la solución de un proyecto?.

Si cualquiera de las preguntas planteadas según el modelo de González & Dankel es negativa, lo más probable es que el desarrollo e implantación de un sistema experto sea un fracaso, por lo que se recomienda mejor no proceder con su realización.

4.1.2 Condiciones que favorecen la implantación de un sistema experto con altas probabilidades de éxito.

De acuerdo a lo definido en el marco teórico y a la información proporcionada al aplicar el modelo de González & Dankel, se concluye que para implantar un sistema experto con altas posibilidades de éxito en una organización, se deben cumplir las siguientes condiciones:

- La organización ha madurado en cuestión de **informática**.- Se cumple cuando la organización no tiene necesidades más apremiantes relativas a la obtención de información para poder operar. Si la organización no se encuentra en la sexta etapa de evolución planteada por Richard Nolan, lo más probable es que fracase.
- La organización reconoce al conocimiento como **recurso**.- Se cumple cuando en la organización se considera al conocimiento como algo que se puede aprovechar en el cumplimiento eficiente de los objetivos de negocio. Si en la organización no se reconoce al conocimiento como recurso, lo más probable es que ni siquiera se conozca dónde se

encuentra el conocimiento, y no se sabrá cómo manejarlo y administrarlo.

- **La organización tiene acceso a expertos.**- Se cumple cuando la participación de expertos, propios o externos, se tiene asegurada; con las contribuciones de estos, es posible mantener el conocimiento útil y actual. En caso de que no se cuenten con expertos, la efectividad de las sugerencias y comportamientos del sistema experto tendrían un margen de error tan amplio que lo harían inoperante.
- **Los problemas que busca resolver requieren de experiencia para ser resueltos.**- Se cumple cuando el proceso para alcanzar la solución de un problema a través de la utilización del conocimiento ha sido comprobado, es decir, no es nuevo. Que se hayan experimentado resultados positivos en la aplicación de un proceso para la solución de un problema asegura que las alternativas de solución están probadas y son conocidas, lo que facilita que puedan ser reproducidas mediante un sistema.
- **Las soluciones a los problemas están basadas en el conocimiento heurístico y no algorítmico.**- Se

cumple cuando la solución del problema que se busca resolver es fruto de la experiencia, y variables que participan en la solución no siempre son confiables y explícitas.

4.1.3 Factores críticos de éxito para los sistemas expertos.

Los factores que determinan el éxito o fracaso de los sistemas expertos son los siguientes:

- Rasgos diferenciativos de los sistemas expertos.
- Factores tecnológicos involucrados en proyectos de sistemas expertos.
- Naturaleza informativa de los sistemas expertos.

A continuación se desarrollará por qué los elementos anteriormente citados.

- a) Rasgos diferenciativos de los sistemas expertos.
- **Están basados en el conocimiento.- Se obtiene el éxito cuando:** Se tiene la cultura suficiente para

entender al conocimiento, las formas de cómo adquirirlo, como representarlo, como utilizarlo en procesos de inferencia y como mantenerlo. **Se fracasa cuando:** No entendemos bien al conocimiento y queremos darle un trato similar a la información.

- **Participantes en la elaboración y mantenimiento.-**

Se obtiene éxito cuando: La organización que desarrolla sistemas expertos es lo suficientemente sólida como para conseguir al personal adecuado para que participe en su proyecto de sistemas expertos, además de poder soportar los costos y el manejo de personal altamente especializado y sensible. **Se fracasa cuando:** No se tiene el poder económico, administrativo y de organización para conseguir y mantener un equipo de especialistas trabajando en armonía.

- **Límites de éxito.- Se obtiene el éxito cuando:** Los problemas que se pretende solucionar con el sistema experto es limitado, y su solución es posible determinarla con la cantidad de conocimiento limitado y disponible. **Se fracasa cuando:** La solución excede el límite de éxito, es decir, cuando la solución de este puede

encontrarse en conocimiento no conocido por el sistema experto.

b) Factores tecnológicos involucrados en proyectos de sistemas expertos.

- **Hardware.- Se obtiene éxito cuando:** Como parte de la planeación del desarrollo e implantación de un sistema experto se determina con exactitud qué necesidades de procesamiento, compatibilidad, enlace y desempeño se tendrán en el corto y en el largo plazo, a efectos de conseguir el equipo e infraestructura adecuada. **Se fracasa cuando:** No se conocen las necesidades reales de procesamiento, compatibilidad, enlace y desempeño para el corto y en el largo plazo, lo que limita la amplitud del proyecto o incrementa significativamente los costos, debido al establecimiento de mecanismos para emular las condiciones óptimas.
- **Software.- Se obtiene éxito cuando:** Como parte de la planeación del desarrollo e implantación de un sistema experto se determina con exactitud qué

necesidades de crecimiento, compatibilidad, desempeño y flexibilidad tendrá nuestro sistema experto en el corto y en el largo plazo, a efectos de conseguir la plataforma de desarrollo adecuada. **Se fracasa cuando:** No se conocen las necesidades reales de crecimiento, compatibilidad, desempeño y flexibilidad que tendrá nuestro sistema experto en el corto y en el largo plazo, de tal forma que se elija una plataforma de desarrollo muy grande o muy limitada.

- **Métodos de representación del conocimiento.-** Se obtiene éxito cuando: Se selecciona el método de representación del conocimiento más adecuado para el tipo de conocimiento que deseamos representar, y que además es soportado por la plataforma de desarrollo. **Se fracasa cuando:** El método de representación del conocimiento no es soportado por la plataforma de desarrollo, en cuyo caso debe realizar una conversión o adecuación, o bien comprar otra plataforma de desarrollo.

c) Naturaleza informativa de los sistemas expertos.

- **Proceso de desarrollo sistemático.-** Se obtiene éxito cuando: No se olvida que los sistemas expertos son una especie de sistema de información más elaborado, y que como tal debe ser sometido a un proceso formal de desarrollo bien documentado.
Se fracasa cuando: Se considera a los sistemas expertos como una forma de sistemas muy aparte de los convencionales, y se dejan de realizar validaciones y fases importantes que aseguran la calidad de todo desarrollo.

4.2 Ciclo administrativo de desarrollo de los sistemas expertos.

En virtud de que existen muchas empresas que no tienen conocimiento de los sistemas expertos, por ser una tecnología nueva y no probada, es necesario dejar en claro antes de su implantación todas sus características, requerimientos y beneficios; partiendo de ahí, podemos afirmar que la labor del desarrollador de sistemas expertos tiene dos variantes:

a) **Introducción y convencimiento.-** Este tipo de labor se presenta cuando se desarrolla por primera vez en la organización un sistema experto; la atención será fijada en el proyecto a fin de vigilar si es conveniente la inversión, entre otros aspectos.

b) **Desarrollo de sistemas expertos.-** Este tipo de labor se presenta cuando la organización ya tiene conocimiento de los sistemas expertos, y por tanto los esfuerzos pueden ser dedicados a su elaboración, sin distraer los recursos en labores del convencimiento.

En este punto del capítulo trataremos la labor de introducción y convencimiento, y en el siguiente se tratará ya más a fondo el desarrollo de los sistemas expertos desde un punto de vista técnico.

La labor de introducción y convencimiento es importante porque de ella depende el futuro de los sistemas expertos en la organización. Si no se realiza adecuadamente esta labor, probablemente los sistemas expertos no sean considerados como posible solución a las problemáticas en el corto plazo, particularmente en aquellas organizaciones muy sensibles al riesgo.

El objetivo de las fases de desarrollo de sistemas expertos al inicio es minimizar los riesgos inherentes a todo proyecto que implica la introducción de nueva tecnología en una organización.

Tal y como propone Mora Tabares (1993), podríamos ilustrar las fases administrativas de desarrollo de sistemas expertos al inicio como se muestra en la figura 11.

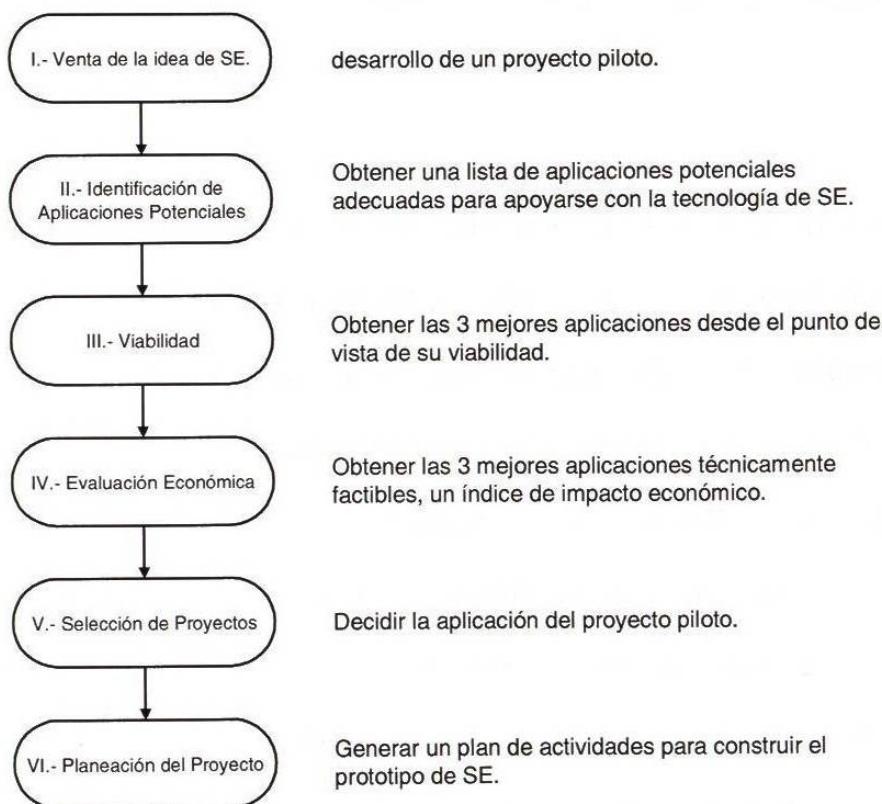


Figura 11: Fases administrativas de desarrollo de sistemas expertos al inicio.

A continuación se desarrollará cada una de las fases anteriormente referidas.

4.2.1 Venta de la idea de los sistemas expertos.

a) Identificación de participantes.

Primeramente se debe identificar a los participantes indispensables en todo proyecto exitoso relacionado con la tecnología, que son:

- **Promotor.-** Es la persona que pertenece o tiene acceso a los niveles de la organización con capacidad de decidir la realización de los proyectos de la organización. Generalmente es una persona que cree en los beneficios de los sistemas expertos aún antes de que las presentaciones correspondientes sean hechas. Entiende a grandes rasgos las características de los sistemas expertos y los beneficios que es posible obtener de ellos. El personal adecuado para esta función es un Director o Gerente con influencia en la toma de decisiones; la ausencia de promotor dificulta enormemente la aceptación de un proyecto de desarrollo

de sistemas expertos, ya que por lo general las organizaciones tienen siempre varios proyectos por realizar, en cuyo caso optarían por aquellos que si son adecuadamente promovidos.

- **Clientes.-** Son las personas a las que va a beneficiar el sistema experto, quienes van a operarlos (usuarios) y quienes se auxiliarán de las conclusiones del mismo para el desempeño de su trabajo. Si un sistema experto no tiene un grupo de clientes constante, no tiene caso su desarrollo. Si no hay cliente, no hay venta (y por tanto, no hay compra).
- **Patrocinador.-** Es la persona que está dispuesta a pagar el precio del sistema experto; en este caso puede ser la propia organización, o alguien que muy probablemente se vea beneficiado con el uso o desarrollo del sistema experto. Por tratarse de tecnología nueva, es posible que los mismos desarrolladores de herramientas y plataformas para el desarrollo de sistemas expertos patrocinen algunos proyectos con la finalidad de exhibir las capacidades y virtudes de su producto, así como sus niveles de soporte; son comunes las alianzas entre casas de software y organizaciones fuertes, casi siempre transnacionales, con esa finalidad. Si nadie patrocina el desarrollo del sistema experto este nunca va a tener

éxito; primeramente, la falta de patrocinador pone en evidencia la falta de voluntad para trabajar con ese tipo de sistemas, y segundo, los sistemas expertos requieren para su desarrollo de tecnología de punta, que por lo general es costosa.

- **Técnico.-** Es la persona que va a desarrollar el sistema experto, al mismo tiempo que es el indicado para resolver dudas de tipo técnico, vitales para la decisión al momento de seleccionar hardware, software y métodos de representación del conocimiento. La presencia del técnico siempre reconforta, pues es el que tiene las respuestas que pueden aclarar el futuro de los sistemas expertos en el corto y en el largo plazo.
- **Coordinador del proyecto.-** Es la persona que se encarga de coordinar la comunicación e identificación de los demás participantes. Es una especie de promotor a bajo nivel, es decir, se encarga de identificar la aplicación potencial antes que nadie, analiza y entrevista a los posibles clientes para evaluar los beneficios potenciales, identifica a quién puede interesarle que los beneficios se obtengan y cuestiona a los técnicos a fin de determinar las limitantes, posibilidades y pronósticos de éxito. El coordinador de proyecto se encarga de informar al promotor, y conoce los recursos

necesarios, el tiempo y las condiciones requeridas para la realización del proyecto.

b) Presentación inicial a la alta gerencia.

Una vez que se tengan identificados a todos los participantes, es posible dar a conocer a los sistemas expertos; en este momento, seguramente el promotor ya se encargó de crear el interés y la curiosidad por conocer a los sistemas expertos en el patrocinador; los clientes saben de la posibilidad de contar con una herramienta nueva para el desarrollo de su trabajo y están a la expectativa; el técnico también está a la expectativa ante un posible incremento de su carga de trabajo.

Cuando esas condiciones están presentes, el coordinador de proyecto debe convocar a una presentación gerencial, que preferentemente debe contar con los siguientes atributos:

Presentación realizada por personal de alto nivel.-

La presentación debe ser presentada y precedida de manera ejecutiva por el promotor; se hace esto porque seguramente él es la persona de más alto rango que apoya de manera

entusiasta al proyecto. El hecho de que personal de alto rango esté a cargo de la reunión posiblemente infundirá respeto hacia el proyecto por parte de aquellos participantes de más bajo nivel que aún tengan dudas; este punto pone en claro, antes de analizar los pros y contras del proyecto, que este cuenta con respaldo organizacional.

Exposición del coordinador del proyecto.- Una de las funciones del promotor es saber el momento en el que debe pasar el control de la reunión al personal idóneo. Se recomienda que el coordinador del proyecto exponga el resto de la presentación gerencial, ya que por las características del rol que juega, es el más informado de todos los aspectos administrativos y técnicos involucrados en el proyecto; el coordinador del proyecto es quien posee las respuestas que llevarán a la aceptación o rechazo de la ejecución del proyecto.

Definir de sistemas expertos.- Se deberá definir a los sistemas expertos de manera teórica, ya que a nadie le gusta hablar un tema que no conoce.

Enumeración de las características de los sistemas expertos.- Se deben listar a grandes rasgos las características de los sistemas expertos. Esta explicación debe ser breve, ya que el punto medular de la

presentación es la exposición de los beneficios y la imposibilidad de obtenerlos con los medios actuales.

Distinción de la información y el conocimiento.- Esta parte tiene como objetivo dejar en claro que el conocimiento y la información son diferentes; aquí se deberá realizar la distinción de los sistemas de información y los sistemas expertos en cuanto a su propósito. De ser posible, se debe identificar un área de oportunidad que no pueda ser cubierta por un sistema de información y si por un sistema experto, a fin de exponer el caso.

Análisis de beneficios de los sistemas expertos.- Se deben enumerar los principales beneficios de los sistemas expertos; esta parte es medular para la presentación, ya que las inversiones siempre esperan beneficios, y en ese orden de ideas, este punto puede llevar a la aceptación o rechazo del proyecto. De entre los mínimos beneficios a resaltar se tienen los siguientes:

- Preservan el conocimiento valioso para la organización.
- Facilitan la distribución y uso del conocimiento.
- Promueven la consistencia en la toma de decisiones.

- Agilizan el proceso de toma de decisiones.

Lista de aplicaciones típicas por área organizacional.- Es la ilustración de las aplicaciones más típicas en las cuales las empresas de todo el mundo utilizan sistemas expertos. Una referencia adecuada puede ser muy valiosa en la medida que sitúa nuestros problemas particulares que tenemos como organización, con soluciones que otras organizaciones han adoptado para sobrellevarlas. Para darse una idea, le recomendamos utilizar la siguiente tabla ilustrativa (tabla 4), propuesta por Paul Harmon & Rex Maus (1998):

Tabla 4.
DOMINIOS Y PROBLEMAS MAYORMENTE SOLUCIONADOS POR LOS SISTEMAS EXPERTOS.

Dominio	Problemas
Alta administración: Ejecutivos Planeación estratégica Asistentes Consultores	Necesidad de reducir la complejidad organizacional. Necesidad de dar seguimiento a los incrementos en el volumen de información. Necesidad de tener acceso a expertos y consultores para obtener consejos.
Operación: Servicios de manufactura Operación de equipo complejo Exploración de energía Control de calidad Control de Inventarios	Proporcionar elementos para incrementar y mejorar la coordinación en la organización, la calendarización y la administración. Necesidad de revisar de manera rápida sistemas complejos para la rápida toma de decisiones. Necesidad de dar seguimiento y controlar equipo complejo. Necesidad de anticipar resultados de eventos dinámicos y complejos.
Servicios de soporte: Relaciones públicas Legal Recursos humanos y entrenamiento Procesamiento de datos Construcción y mantenimiento Investigación y desarrollo	Necesidad de capacitar personal para el manejo de tareas complejas. Necesidad de comunicar y explicar nuevos procesos complejos. Necesidad de examinar y explicar políticas para la toma de decisiones. Necesidad de controlar y reducir costos de desarrollo y mantenimiento de software.
Finanzas: Administración de inversiones. Contabilidad Administración financiera Auditoría Contraloría	Necesidad de revisar los sistemas complejos existentes Necesidad de herramientas inteligentes de planeación financiera, orientadas a los objetivos. Necesidad de pronosticar situaciones futuras en base a condiciones inciertas.
Mercadotecnia: Ventas Publicidad Investigación de mercados Servicio al cliente Pedidos	Necesidad de asistencia de expertos para examinar preguntas acerca de la situación del mercado. Necesidad de asistencia a las ventas, a través de programas que proporcionen conocimiento de los productos y que asustan en la elaboración de propuestas a los clientes en base a sus características.
Automatización de oficinas: Procesamiento de palabras Administración de datos	Necesidad de incrementar la productividad en el manejo, llenado, comunicación, recuperación y distribución de información y datos en la oficina.
Servicios profesionales: Consultores administrativos Abogados Físicos Contadores	Necesidad de dar seguimiento e incrementar el volumen de información. Necesidad de disponer de los servicios de expertos para obtener consejo. Necesidad de generar reportes inteligentes.

Tal vez pueda resultar tedioso ver la totalidad de los conceptos contenidos en la tabla; se recomienda seleccionar para la presentación aquellos rubros que sabemos son de interés de la organización. Al hacer referencia a las aplicaciones típicas y ejemplos de la funcionalidad de los sistemas expertos, es importante dirigir la ponencia con las siguientes herramientas y consideraciones:

- **Lista de casos comerciales exitosos.**— Se recomienda hacer referencia a casos exitosos de implantación de sistemas expertos; como mínimo deben relacionarse los datos de la empresa, el problema que resolvió, los beneficios adicionales que obtuvo y las fechas de realización del proyecto de sistemas expertos. Se recomienda que los ejemplos sean lo más cercano posible a la organización, tanto en tiempo como en relación. Una empresa de éxito en un país y una industria diferente, hace tres años es un buen ejemplo; una empresa que compite con nosotros actualmente es mejor. Cabe aclarar que la existencia de sistemas expertos en la competencia cambia la orientación de la presentación, pues la búsqueda

ya no es sobre la innovación tecnológica, sino sobre la supervivencia competitiva.

- **Presentación de un demostrativo o video, en caso de existir.-** Sólo si se dispone de un buen material de calidad, que sea breve y que consideremos que favorecerá la opinión de las personas con respecto a los sistemas expertos.

Muchas veces una imagen dice más que mil palabras.

Aplicaciones sugeridas para la organización.-

Proposición de aplicaciones que la organización podría explotar mediante los sistemas expertos. En este punto se pueden hacer notar las áreas de oportunidad que se tienen, y de ser posible, se deben respaldar las proposiciones con cifras la relativas a costos en dinero y en tiempo que estamos dejando de recuperar o ahorrar por la falta de herramientas tecnológicas adecuadas.

Limitantes y costos de los sistemas expertos.-

Deberán aclararse los límites de los sistemas expertos, esto a fin de que no se piense en los mismos como una panacea; asimismo, se debe hacer mención del costo que tienen. Este es un punto muy importante, ya que los beneficios ya fueron expuestos; con el conocimiento de los costos y beneficios implícitos con el desarrollo de los

sistemas expertos, se puede poner en la balanza de manera muy práctica si son convenientes o no. Al momento de detallar pros y contras de los sistemas expertos se debe hacer hincapié en los costos de las áreas de oportunidad y como los sistemas expertos pueden ser la solución a menos costo.

No se debe ocultar información relevante de las limitantes de los sistemas expertos, ya que las falsas expectativas pueden ser muy perjudiciales en el proceso de desarrollo.

Dentro de las limitantes de los sistemas expertos que no debe pasar por alto son las siguientes:

- No aplican para tareas que demanden el uso de los sentidos.
- No aprenden automáticamente.
- No aplican para tareas de sentido común.
- Interacción similar a la encontrada en otros sistemas de cómputo.

Sesión de preguntas y respuestas.- Una sesión de preguntas y respuestas resulta muy provechosa porque nos dice qué tanta atención e interés se tuvo en la presentación. Es muy importante la seguridad al

contestar, pues respuestas mal manejadas pueden dar la impresión de que la organización no está preparada para los sistemas expertos. Se recomienda no mentir ni ocultar información relevante en las respuestas, ya que, por ser un proyecto de sistemas expertos por primera vez, fija un precedente difícil deuitar.

4.2.2 Identificación de aplicaciones potenciales.

Después de vender la idea, hay que dar seguimiento a las áreas de oportunidad detectadas a fin de seleccionar el proyecto que más asegure el éxito. En el caso del primer sistema experto es indispensable un éxito rotundo, a fin de que sirva de muestra y pueda ser usado como referencia para la difusión de la tecnología. Existen consideraciones relativas a la organización, al experto, a la tarea a sistematizar y a los usuarios que influyen en la decisión de qué proyecto seleccionar para su desarrollo. Cabe hacer notar que nos referimos al usuario del sistema experto y no al cliente del sistema experto; esto se debe a que el cliente puede ser alguna persona que nunca interactúa con el sistema experto, y sin embargo puede obtener beneficios de las conclusiones proporcionadas por

el mismo sistema. Una aplicación es potencial en una organización si:

Relativo a la **Organización**:

- Acepta el impacto económico derivado del desarrollo, derivado de la adquisición y uso de nueva tecnología.
- Acepta que debe dedicar recursos, capacitarlos y formarlos hacia el nuevo esquema de trabajo.

Relativo al **Experto**:

- Existe un experto reconocido en la organización.
- El experto usa razonamiento lógico y heurístico (no habilidades físicas).

Relativo a la **Tarea a sistematizar**:

- El conocimiento que requiere para ser llevado a cabo es representable de manera simbólica.
- Su ejecución requiere de cierto grado de consistencia y contribuye marginalmente a la obtención de utilidades.
- Es de naturaleza intelectual, basada en el razonamiento, y no en habilidades físicas.

Relativo a los Usuarios:

- Aceptan que deberán dedicar tiempo a su capacitación.
- Aceptan la complejidad relativa de la tarea.
- Al finalizar esta fase, deberá contar con una lista de aplicaciones potenciales que pueda someter a evaluación para determinar su viabilidad.

4.2.3 Análisis de viabilidad.

Mora Tabares (1993) sugiere una sencilla forma de comparar la viabilidad de diferentes proyectos, mediante un análisis que nos arrojará información en referencia a qué tan preparados estamos para abordar el desarrollo de un sistema experto para una propuesta específica.

Los grados de calificación son:

- **5: Alto grado de viabilidad.** Los recursos son existentes y disponibles, se cuenta con el apoyo de la organización y las condiciones son propicias para comenzar.

- 3: **Grado medio de viabilidad.** Las condiciones son propicias, pero la existencia o disponibilidad del recurso no está confirmado; se requieren esfuerzos adicionales para comenzar.
- 1: **Grado bajo de viabilidad.** Prácticamente se requiere adquirir, contratar o implantar los recursos, pues son nuevos para la empresa; adicional a esto se requiere crear las condiciones propicias para iniciar.

La mecánica es la siguiente: Existen factores a considerar para determinar si las condiciones son propicias o no para el desarrollo de un sistema experto; un ejemplo podría ser la existencia de un experto cuyo desempeño es claramente superior con respecto a la ejecución de una tarea.

Si en la organización existe un experto con un desempeño claramente superior que pueda colaborar con el proyecto de desarrollo de sistemas expertos, el grado de viabilidad aplicable será 5; si en la organización existe un experto con un desempeño claramente superior, pero está dedicado a otros proyectos y puede dedicar sus conocimientos al proyecto de desarrollo de sistema experto de forma no dedicada, o bien se tiene alguien que puede

colaborar plenamente con el proyecto de desarrollo del sistema experto pero que aún y cuando es el más experimentado su desempeño no es claramente superior, el grado de viabilidad aplicable será 3; si en la organización no hay un experto para la tarea y las condiciones para conseguir alguno no son satisfactorias, el grado de viabilidad aplicable será 1.

Se debe tener una tabla en donde se evalúe la viabilidad en base a los factores relacionados con la organización, el experto, la tarea a sistematizar y el usuario. En esa sencilla matriz se realizará una suma de puntos, determinándose qué proyectos son viables y en qué factores. No hay factores más relevantes que otros, asimismo, no hay puntuaciones mínimas para considerar un proyecto como viable. La decisión del proyecto con más viabilidad depende de la posición de la organización, sus recursos y sus objetivos más prioritarios. Existe la posibilidad que el proyecto menos viable esté relacionado con una tarea de misión crítica que requiere atención inmediata; tal vez ese proyecto sea seleccionado antes que otros con más viabilidad que aplican sobre tareas no prioritarias para la organización.

La tabla de análisis de viabilidad puede lucir como muestra la figura 12:

Instrucciones: Para cada aplicación propuesta califique su grado de certeza, donde 5: Alto grado de certeza
3: Grado medio de certeza y 1 Grado bajo de certeza.

	ID de las Aplicaciones propuestas				
Factores:					
EXPERTO					
Su desempeño es claramente superior					
Su ausencia afecta a la organización					
Su actitud hacia un posible SE es cooperativa					
Es comunicativo					
Está familiarizado con las computadoras					
Calificación promedio del experto					
TAREA					
Se basa en el uso de la inferencia simbólica					
Tiene identificadas claramente las entradas y salidas					
Demandó el uso de conocimiento generalmente estable					
Usa conocimiento muy particular de la organización					
Causa costos significativos por retrasos y errores					
Su ejecución exige personal altamente calificado					
Calificación promedio de la tarea					
USUARIOS					
Están interesados en el uso de las nuevas tecnologías					
Cuentan con apoyo de la alta gerencia					
Son realistas y sus expectativas son alcanzables					
Son claramente afectados por la tarea					
Reconocen el costo de no tener un SE					
Calificación promedio del usuario					
Evaluación técnica de la aplicación:					

Figura 12: Matriz para evaluar la viabilidad de los proyectos de sistemas expertos.

4.2.4 Evaluación Económica.

La evaluación económica se encarga de cuantificar cuánto cuesta no atender un área de oportunidad y cuánto cuesta el desarrollo de un sistema experto que la atiende,

a efectos de confrontarlos para determinar la conveniencia del desarrollo.

La temporalidad es muy importante; considérese para efectos del cálculo la vida útil estimada del sistema experto, para poder calcular el costo no atender un área de oportunidad en el mismo período.

El método de costeo a utilizar pueden ser cualquiera en el que ya se tenga experiencia; se deberá cuantificar el costo de oportunidad de la tarea a sistematizar, atendiendo básicamente los siguientes rubros:

- Cuánto cuestan las demoras
- Cuánto cuestan los errores
- Cuánto se gasta en asesorías especializadas

Se deberá también estimar el costo de desarrollo de un sistema experto considerando como mínimo los siguientes rubros:

- Shell o lenguaje.
- Equipo de cómputo
- Asesoría especializada

- Sueldos (programadores, ingenieros del conocimiento, expertos dedicados, etc.)
- Mantenimiento del sistema.

Los beneficios se cuantificarán determinando los costos de oportunidad que se evitan en una temporalidad dada, o bien los beneficios proporcionados por el sistema, en caso de que el uso de éste represente un producto por sí mismo.

Finalmente se puede realizar un análisis de costo - beneficio, en donde se comparan los beneficios multiplicados por número de temporalidades dadas como tiempo de vida estimado y efectivo del sistema experto, contra el costo total de su desarrollo y mantenimiento.

A continuación la figura 13 muestra el ejemplo de una tabla comparativa de costos y beneficios para un proyecto dado; en el ejemplo, hay un proceso que se sistematiza mediante un proyecto de sistemas expertos. El proceso está compuesto por las tareas A, B, C, D Y E; mensualmente cada tarea tiene un costo de oportunidad y una cantidad solventada (ahorrrada) por mes. Como las cantidades no son exactas siempre, se hace mención de que la cantidad es promedio.

Temporalidad: Tareas involucradas en el proceso:	Mensual A, B, C, D, E	COSTO DE OPORTUNIDAD	SOLVENTADO MEDIANTE S.E.
Costo de oportunidad tarea A	N\$	150,000.00	N\$ 150,000.00
Costo de oportunidad tarea B	N\$	20,000.00	N\$ 15,000.00
Costo de oportunidad tarea C	N\$	35,000.00	N\$ 35,000.00
Costo de oportunidad tarea D	N\$	48,000.00	N\$ 2,000.00
Costo de oportunidad tarea E	N\$	2,388.00	N\$ 2,388.00
Costo de oportunidad mensual promedio:	N\$	255,388.00	N\$ 204,388.00
COSTO DE DESARROLLO			
		Tiempo de desarrollo:	2 meses
Shell (3 licencias)	N\$	18,000.00	
3 Computadoras	N\$	42,000.00	
Sueldo de Experto	N\$	50,000.00	
Sueldo del Ingeniero del conocimiento	N\$	25,000.00	
Programador	N\$	15,000.00	
Honorarios del Asesor Externo	N\$	38,000.00	
Costo total de desarrollo	N\$	188,000.00	
COSTO BENEFICIO			
Tiempo de vida estimado del sistema experto:		6	
Beneficios del periodo	N\$	1,226,328.00	
Vs. costo total de desarrollo.	N\$	188,000.00	
Contribución marginal (reducción de costos estimado)	N\$	1,038,328.00	

Figura 13: Tabla comparativa de costos - beneficios al utilizar un sistema experto.

Se calcula el costo total de desarrollo del sistema experto y se estima que su tiempo de vida estimado es de 6 meses. Si se calcula el costo de oportunidad (monto de lo solventado) multiplicado por el tiempo de vida del sistema se obtienen los beneficios de desarrollar un sistema experto; si los beneficios del período exceden el costo total de desarrollo, es recomendable desarrollar el sistema.

4.2.5 Selección del proyecto.

Se procede a seleccionar el proyecto dependiendo de las prioridades del negocio; se deberá optar por el que sea más rentable (justificación económica) y más viable.

En caso de que los factores de selección no coincidan en un mismo proyecto, se deberá optar por el que mejor siga la línea del negocio. Cabe aclarar que la selección del proyecto debe contar con el apoyo moral y económico de la alta gerencia, o patrocinador en su caso, mismos que pueden variar de un proyecto a otro.

4.2.6 Planeación del proyecto

Se procede a establecer tareas, secuencia de ejecución, responsables, costos asignados, etc. Se sugiere controlar el proyecto mediante el uso de gráficas de Gantt, para lo cual se puede utilizar software especializado para el control de proyectos. Para poder realizar la planeación de manera adecuada, es importante determinar el alcance del proyecto, el alcance del presupuesto, y disponibilidad del personal experto, así como de la atención de las tareas que pueden ser ejecutadas simultáneamente.

4.3 Ciclo técnico de desarrollo de los sistemas expertos.

4.3.1 Ingeniería del conocimiento.

Para poder comprender el ciclo de desarrollo de los sistemas expertos, es necesario entender muy bien las funciones del ingeniero del conocimiento. No todos los sistemas expertos requieren la presencia del ingeniero del conocimiento, ya que las herramientas y la amplitud de la tarea a sistematizar pueden ser muy sencillas. La figura 14 ilustra la relación existente entre la aplicación a desarrollar y la necesidad de ingeniero del conocimiento, propuesta por Paul Harmon & Rex Maus (1988).

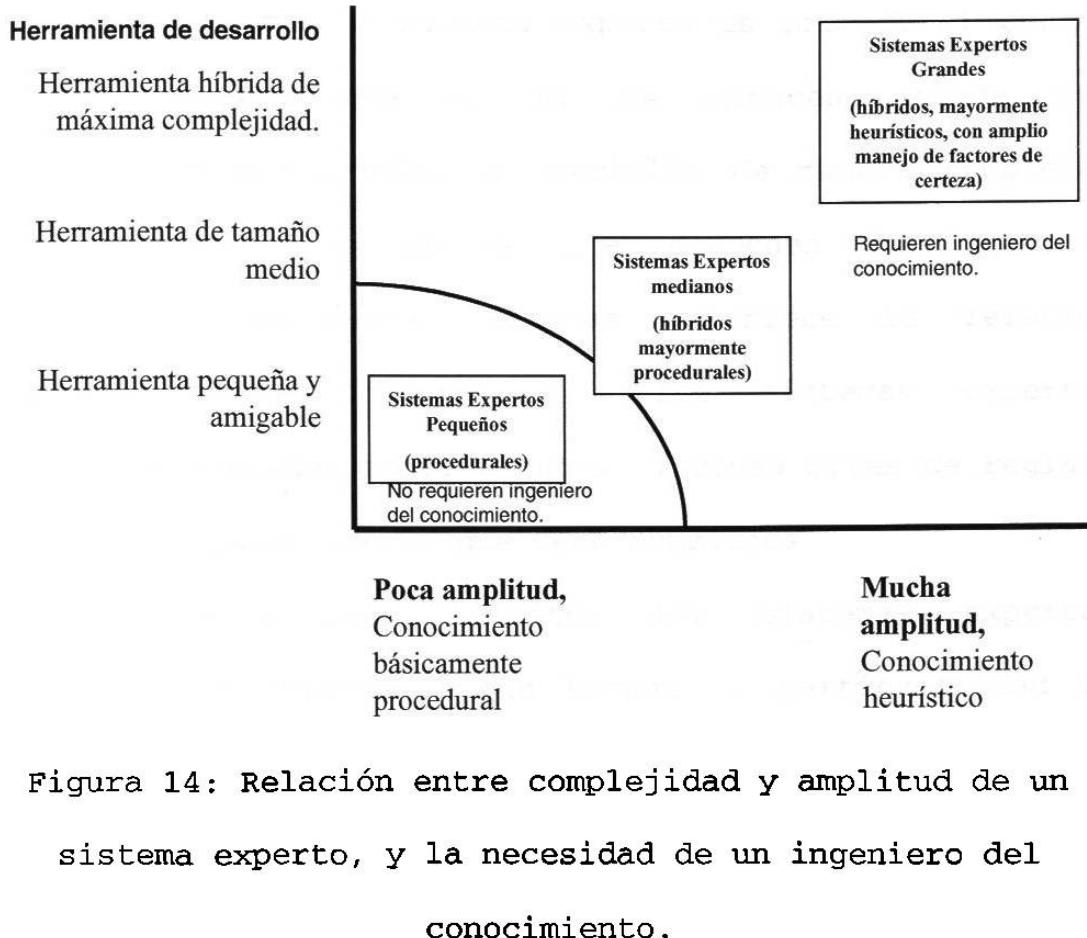


Figura 14: Relación entre complejidad y amplitud de un sistema experto, y la necesidad de un ingeniero del conocimiento.

Las herramientas de desarrollo basadas en reglas (rule-based tools) son cada día más amigables en su interfase, de tal forma que cualquier persona puede aprender fácilmente a introducir conocimiento en ellas de manera correcta. La decisión de utilizar estas herramientas u otras más complejas depende de la naturaleza y complejidad del conocimiento que requiere ser formulado, modelado y codificado.

Se dice que un sistema experto es procedural cuando el conocimiento que en él se almacena puede ser representado por árboles de decisión de manera completa; casi siempre, las reglas que componen la base de conocimiento de estos sistemas establece la relación existente entre términos. Los sistemas expertos procedurales pueden tener cientos, incluso miles de reglas, pero eso no quiere decir que sean complejos.

Por otro lado, existen los sistemas expertos denominados heurísticos, que tienen la particularidad de poder trabajar con evidencia incompleta, de gran complejidad. Estos sistemas expertos necesariamente hacen manejo de factores de certeza, hacen uso continuo de bases de datos y de casos, así como mecanismos de inferencia que involucran tecnología compleja como lógica difusa y redes neuronales.

Cuando los sistemas expertos son pequeños, lo más probable es que sean procedurales y una herramienta de desarrollo basado en reglas sea suficiente; si son de mediano tamaño, seguirán basándose en reglas pero la heurística comenzará a aparecer, por lo que comienza a ser necesaria la labor del ingeniero del conocimiento. Los sistemas expertos grandes y complejos, definitivamente

requieren de la participación de uno o varios ingenieros del conocimiento para realizar sus tareas.

4.3.1.1 Actividades desarrolladas por el ingeniero del conocimiento.

Las actividades que deben realizar los ingenieros del conocimiento son las siguientes:

a) **Adquisición del conocimiento.**- Consiste en todas las actividades relacionadas con obtener el conocimiento de los expertos. Esto incluye entrevistas y sesiones de trabajo que permitan unificar los objetivos, definir términos, analizar procedimientos y criterios que llevarán a una conclusión, eliminando la vaguedad e inconsistencia que se pueda presentar en los procesos. Dentro de este trabajo se incluye el estudio de casos y la carga lenta y de manera fiel el conocimiento del experto en la base de conocimiento. El ingeniero del conocimiento nunca deja de adquirir el conocimiento, ya que este puede estarse puliendo de manera continua.

b) **Modelación del conocimiento.**- Consiste en organizar el conocimiento adquirido por los expertos. En este proceso se elaboran los árboles de decisión y redes de

elementos que permitan definir claramente el escenario que constituye el conocimiento. Los expertos proporcionan hechos, términos, reglas, objetos, criterios, todo de forma conjunta; el ingeniero del conocimiento debe determinar la naturaleza de cada uno de los elementos proporcionados por el experto, debe generar grupos y sub grupos de elementos, debe analizar la dependencia entre ellos, formular el conocimiento de forma tal que sea posible analizarla de manera lógica.

c) **Codificación del conocimiento.**- Proceso por medio del cual se introducen los hechos, reglas, objetos y dependencias entre los diferentes elementos en un sistema experto. La codificación puede realizarse en lenguajes apropiados tales como LISP, PROLOG, Small Talk u otro, y para ello se requieren habilidades de programación; también se puede utilizar una herramienta de desarrollo se le deberá especificar el conocimiento en la forma en que lo requiere, y generalmente es considerablemente más fácil que utilizar un lenguaje.

4.3.1.2 Proceso de ingeniería del conocimiento

Para los proyectos grandes, difícilmente un solo ingeniero del conocimiento desarrolla tres actividades, ya

que es tedioso y cansado. Usualmente la división del trabajo da buenos resultados en la ingeniería del conocimiento, porque asegura una revisión continua entre el desempeño de los ingenieros involucrados; esto es posible porque la labor de la ingeniería del conocimiento no detiene, ya que siempre se puede estar perfeccionando un desarrollo. Paul Harmon & Rex Maus (1988) sugieren una serie de fases para desarrollar sistemas expertos, con un enfoque basado en tareas y la interacción del recurso humano en el desarrollo de estas. En la siguiente gráfica (figura 15) podemos ver el proceso de ingeniería del conocimiento, como una tarea continua.

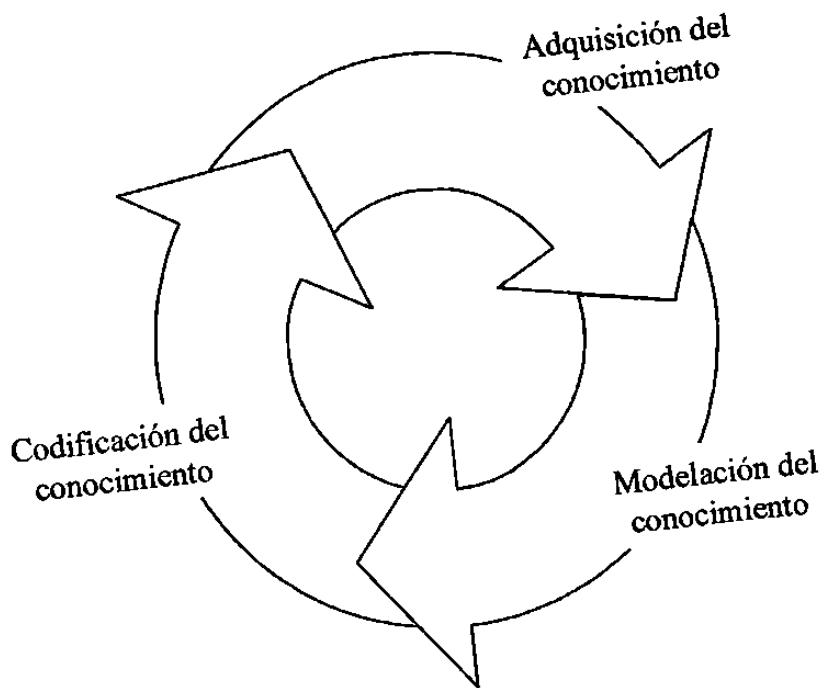


Figura 15: Ingeniería del conocimiento como tarea.

El ingeniero del conocimiento está prácticamente en todos los momentos del desarrollo, involucrado de una manera activa. El desarrollo de un sistema experto está íntimamente ligado con la función de ingeniería del conocimiento.

4.3.2 Fases del desarrollo de un sistema experto.

El desarrollo de sistemas expertos involucra una comunicación exhaustiva entre usuarios, expertos y personal de desarrollo, tal y como lo ilustra la figura 16.



Figura 16: Fases del desarrollo de sistemas expertos y su relación con los usuarios y expertos.

Tal comunicación determina que las fases para su desarrollo estén fuertemente orientadas en ese sentido. Las 7 fases de desarrollo de sistemas expertos son:

- a) Análisis preliminar.
- b) Análisis de tareas.
- c) Desarrollo del prototipo.
- d) Desarrollo del sistema.
- e) Pruebas de campo.
- f) Implementación.
- g) Mantenimiento.

Cada una de ellas tiene sus particularidades y metas; en la figura 17 se ilustra de manera gráfica las fases y sus objetivos.



Figura 17: Fases de desarrollo de los sistemas expertos y su objetivo.

A continuación se detallan cada una de las fases.

a) Análisis preliminar.

Consiste en analizar detenidamente los diferentes problemas que pueden ser solucionados por sistemas expertos, a fin de seleccionar alguno para el desarrollo. Las tareas en esta fase son ejecutadas por administradores y líderes de equipo, ya que un proyecto de desarrollo de sistemas expertos es igual a cualquier otro proyecto de inversión.

Como cualquier otro proyecto, debe existir un equipo de trabajo que desarrolle el análisis preliminar, en donde se determinará si un experto es el apropiado para basar el proyecto, si el proyecto es costeable, qué hardware y software se requerirá para su desarrollo, qué personal va a ser asignado al proyecto. El administrador a cargo de un proyecto de sistemas expertos tendrá que planear, identificar los patrocinadores corporativos del proyecto y los clientes del mismo, elaborar informes para administrar las expectativas, entre otras tareas de rutina propias de un proceso de desarrollo de sistemas. En esta fase los productos son la selección de un proyecto, la justificación del mismo, la autorización a su realización, y el establecimiento de los objetivos sobre los que se va a medir el éxito o el fracaso.

b) Análisis de tareas.

Consiste en determinar cómo serán alcanzados los objetivos; en esta fase se deben definir los criterios para determinar el éxito o fracaso del proceso de desarrollo. Se deberán definir las tareas a ejecutar, asociando a ellas la cantidad de conocimiento que se desea capturar en el sistema experto en cada una, estableciendo parámetros mínimos de funcionalidad y contenido que permitan la comparación y la evaluación. Realizan esta tarea los administradores o líderes de proyecto, trabajando en conjunto con el ingeniero del conocimiento.

c) Desarrollo de prototipo.

Se desarrolla una versión limitada del sistema experto para demostrar su factibilidad y funcionalidad. Se deben definir los conceptos más significativos del conocimiento involucrado en el problema que resuelve el sistema experto, así como el flujo de información necesario para ilustrar el proceso de solución del problema.

Tanto los conceptos como los flujos de información deben ser representados por el prototipo, que valiéndose de estos como herramienta, también propone el hardware y software que se espera utilizar, los formatos para la formulación del conocimiento que se pretende seguir, y se detectan las posibles restricciones.

Esta es la última fase en la que se realizan análisis y obtención de información como propósito básico; es posible que en otras fases se obtenga información, pero no es su propósito. En esta fase se hace el refinamiento del sistema; se obtiene del experto su método de razonamiento, mismo que se documenta y que será el corazón del sistema.

Cuando el prototipo está liberado, constituye el modelo del sistema experto completo; en esta versión se hacen ajustes y se enfrentan los problemas que con seguridad se presentarán en el desarrollo de todo el sistema.

d) Desarrollo del sistema.

Se toma como base el prototipo y se le agrega la totalidad del conocimiento recabado en las fases anteriores; asimismo, se depuran los procesos de razonamiento expuestos por el experto. Todo esto se

codifica de tal forma que se obtenga buen rendimiento por parte del sistema, una interfase cuidadosa y amigable, y un comportamiento estable. Esta tarea es realizada particularmente por el ingeniero del conocimiento; el producto de esta fase es obtener los programas sin errores que generen los resultados correctos al aplicarles unos datos de entrada comprobables, y siguiendo las especificaciones del diseño.

e) Pruebas de campo.

Consiste en instalar el sistema en un medio ambiente similar al del usuario, a fin de comparar los resultados y la funcionalidad del sistema ya en el campo. La primera fase es verificar si no existen errores obvios; después, es evaluado el funcionamiento del sistema contra el experto, a fin de ver las diferencias en rapidez y precisión. Se considera como bueno que un sistema experto y el experto mismo coincidan en un 80% a un 90% en las conclusiones que generan ante un mismo problema. La mayoría de las tareas de esta fase son ejecutadas por el ingeniero del conocimiento.

f) Implantación.

Se instala el sistema en el medio ambiente de trabajo del usuario; esto involucra colocar la aplicación en hardware diverso y capacitar en el uso del sistema, con la finalidad de eliminar dudas, fomentar su uso y promover su aceptación. La implantación debe ser bien organizada, ya que es una fase muy sensible donde los problemas son más organizacionales y psicológicos que técnicos; se recomienda tener bien documentado el sistema, brindar capacitación dedicada e integral, y dar excelente soporte. Un sistema experto, por más bueno que sea, es inútil si la gente decide no utilizarlo. Esta fase requiere el apoyo de la administración de la compañía, personal del proyecto y el ingeniero del conocimiento.

g) Mantenimiento.

Es la fase final del proceso; esta fase nunca se concluye, ya que se sostiene durante todo el ciclo de vida del sistema.

Esta fase permite actualizar el conocimiento y los programas que componen el sistema; consiste en corregir o adecuar el sistema a fin de que sea actual y cumpla con las expectativas que de él se tienen. Esta fase involucra revisión del sistema y acciones tomadas en base a los resultados obtenidas de la misma revisión.

4.4 Reglas y tipos de razonamiento.

Un experto puede divagar en su conocimiento para llegar a conclusiones nuevas, y podría esperarse que los sistemas expertos hicieran lo mismo, y sin embargo no lo hacen; la explicación a ello se mencionó en los inicios del presente trabajo en donde se diferenciaba el conocimiento de la imaginación.

Los programas de cómputo, hasta en sus más flexibles manifestaciones, no pueden desprenderse de su rigidez lógica; el comportamiento estable de un programa de cómputo es producto de la estructura lógica de su funcionamiento, de la secuencia de ejecución que el diseñador planeó que tuviera.

Aun y cuando se pretenda dar a los sistemas expertos la riqueza y flexibilidad de posibilidades de la mente humana y su forma de razonar, el conocimiento y las formas

de razonamiento requieren ser estructurados para poder ser interpretados por la computadora.

4.4.1 Reglas.

La forma de representar la estructura lógica del conocimiento es, por excelencia, la utilización de *reglas*. La mayoría de los sistemas expertos están basados en reglas, por lo que son muy conocidos como RBS (*Rule Based Systems*).

Los sistemas basados en reglas representan el conocimiento mediante un conjunto sentencias del tipo:

"If **x** then **y**"

Según Waterman y Hayes-Roth (1978), en el conjunto de sentencias que conforman el conocimiento, se va formando una cadena de condiciones antecedentes que, una vez aplicadas producen una consecuencia. Las consecuencias derivadas por un conjunto de reglas involucradas en un proceso de inferencia son añadidos a la base del conocimiento como nuevos hechos (*facts*). Algunas de las características primordiales de las reglas son:

- Cada regla actúa como una condición del tipo IF THEN.
- Cada regla tiene una o más *condiciones de antecedente* (IF) y una o más *condiciones subsecuentes* (THEN), llamadas también *consecuencias o conclusiones*.
- Las reglas pueden variar en complejidad, teniendo una condición de antecedente o muchas, ligadas por medio de operadores booleanos ("and", "or", "less than", "greater than", etc.).
- Se debe procurar que las conclusiones derivadas de la aplicación de una regla actúen como elementos para la solución de condiciones de antecedente de otras de aplicación posterior.
- Se dice que se aplica una regla cuando el lenguaje o Shell considera una regla para intentar generar, en conjunto con la memoria de trabajo, una conclusión.

4.4.2 Representación gráfica de las reglas.

Gráficamente las reglas se representarían como lo muestra la figura 18.

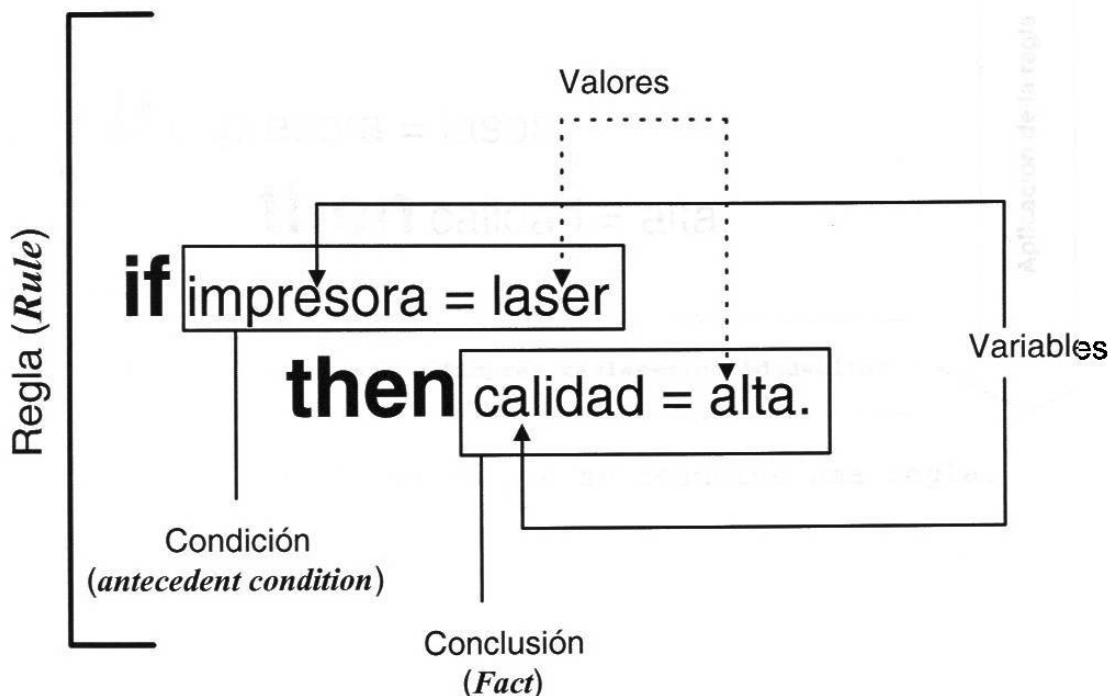


Figura 18: Diagrama estructural de una regla.

Al ser aplicada una regla, tiene el siguiente comportamiento: primeramente se evalúa una condición a la que se le llama "antecedente"; si la condición se resuelve por verdadero, se lleva a cabo una asignación de valor a una variable, lo que se conoce como conclusión; la variable y el valor adquirido se almacenan en la memoria de trabajo,

formando lo que se llama un hecho (*fact*). La forma de trabajo de una regla se ilustra en la figura 19.

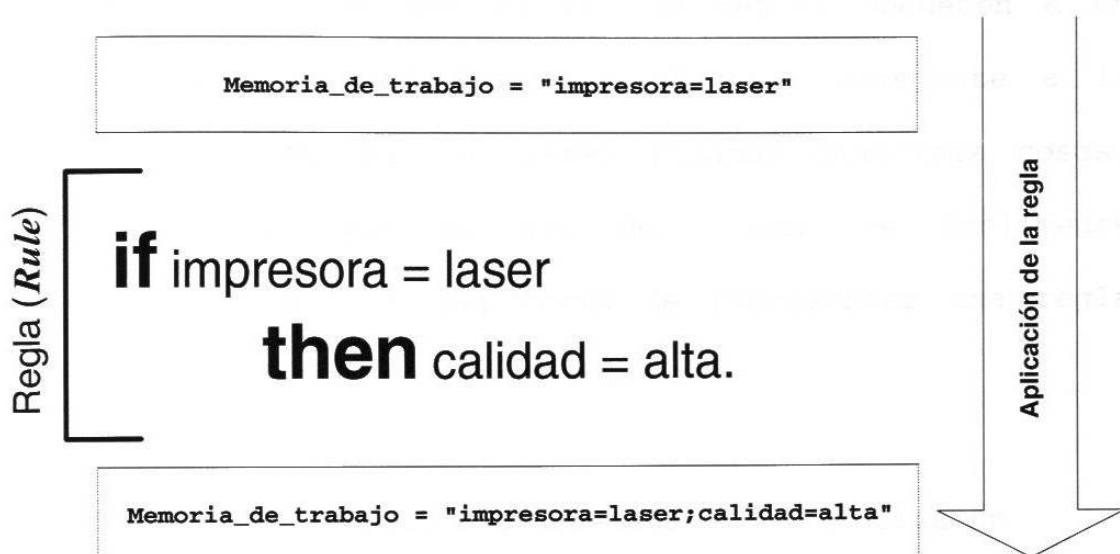


Figura 19: Forma en que se resuelve una regla.

4.4.3 Ventajas del uso de las reglas.

Algunas ventajas de representar al conocimiento en base a reglas, son las siguientes:

- El esquema de reglas permite descomponer problemas complejos en componentes más simples, los cuales al ser resueltos, contribuyen a la solución del problema original y más extenso.

- Los sistemas basados en reglas pueden contener miles de reglas, característica por la cual son los más ampliamente aceptados.
- Se entiende que el uso de reglas obedecen a un modelo causal (causa - efecto), semejante a la forma en que los seres humanos deducimos cosas, por lo que el uso de reglas es fácilmente comprensible; una forma de representar una regla considerando su causalidad es:

Condición antecedente → Conclusión

Recordemos que la *inferencia* es obtener elementos desconocidos a partir de los conocidos; considerando eso, las condiciones antecedentes evaluarán siempre los hechos conocidos para llevarnos a los desconocidos, en este caso, las conclusiones.

4.4.4 Tipos de razonamiento.

Cómo se logrará evaluar e interpretar las reglas, depende del *método de razonamiento* que se utilice. Un

método de razonamiento es el mecanismo por medio de los cual se extraerán los elementos desconocidos de los elementos ya conocidos.

Existen varios tipos de razonamiento utilizados por los sistemas expertos; por tipo de razonamiento nos referimos precisamente a la estructura lógica que se lleva a cabo para concluir una cosa a partir de datos base. Mora Tabares sugiere la existencia de 3 tipos de razonamiento utilizados por los sistemas expertos; los tipos de razonamiento básicos utilizados para llevar a cabo la inferencia en las herramientas de cómputo son los siguientes:

- Deducción
- Abducción
- Inducción

El mecanismo de inferencia de un sistema experto no necesariamente hace uso de manera exclusiva de uno de estos tipos de razonamiento; en ocasiones se podrá utilizar mezclas entre los tipos para obtener la inferencia más acertada o productiva. A continuación se detallarán los tres tipos de razonamiento:

a) Deducción

En este método, la evaluación recae sobre la condición antecedente; si dicha condición no cumple por verdadero, no se tiene acceso a la conclusión y el proceso de razonamiento continúa evaluando las demás reglas. Un ejemplo puede verse en la figura 20.

Conocimiento:	$P(x) \Rightarrow Q(x)$
Datos requeridos:	$P(x)$
Conclusiones:	$Q(x)$ es 100% cierto
Ejemplo:	
Conocimiento: si oferta = aumenta entonces precio = baja fin	
Datos requeridos:	oferta = aumenta
Conclusiones:	el precio baja en todos los casos cuando la oferta aumenta.

Figura 20: Deducción.

La deducción es un ejemplo clásico del uso de reglas definitivas; la certeza en la deducción es 100% cierta en todos los casos (a menos que se especifique un factor de incertidumbre), ya que la conclusión es directamente determinada por la condición antecedente.

b) Abducción

En este método, la evaluación recae sobre la conclusión, y no sobre la condición antecedente. Un ejemplo puede verse en la figura 21.

Conocimiento:	$P(x) \Rightarrow Q(x)$
Datos requeridos:	$Q(x)$
Conclusiones:	$P(x)$ es posiblemente cierto

Ejemplo:

Conocimiento:	si empleado = incapacitado entonces asiste_a_trabajar = no fin
---------------	---

Datos requeridos:	asiste_a_trabajar = no
-------------------	-------------------------------

Conclusiones:	si el empleado no asiste a trabajar probablemente está incapacitado
---------------	--

Figura 21: Abducción.

c) Inducción

En este método las conclusiones son ciertas para un determinado dato porque no es relevante al resultado. Un ejemplo se puede ver en la figura 22.

Conocimiento:	P(a), P(b), P(c)...
Datos requeridos:	P(x)
Conclusiones:	P(x) es cierto para cualquier valor de x
Ejemplo:	
Conocimiento:	
<pre>si costo = baja y producto = SPLE512 entonces se_puede_bajar_precio = si si costo = baja y producto = MSA2452 entonces se_puede_bajar_precio = si si costo = baja y producto = CMP2562 entonces se_puede_bajar_precio = si</pre>	
Datos requeridos:	producto = ?
Conclusiones:	para cualquier producto el precio se puede bajar si el costo baja

Figura 22: Inducción.

La inducción se da cuando el historial de una condición antecedente sugiere un resultado; generalmente se da cuando la condición antecedente incluye una condición dominante, que determina el resultado sin necesidad de las restantes condiciones. En este caso, las restantes condiciones son determinantes en la ejecución de la regla, pero no en el conocimiento; en nuestro ejemplo, todo parece indicar que para cualquier producto, si el costo baja el

precio se puede bajar. La condición antecedente puede incluir una parte que marque una excepción.

La inducción es muy utilizada para razonamientos con alto grado de incertidumbre para situaciones similares pero no iguales; es muy común su uso en estadística. Un ejemplo de ello son los análisis deportivos, en donde se muestra el patrón seguido por un equipo jugando en cierta ciudad o frente a cierto contrincante o cosas así; si no existiera alto grado de incertidumbre y las situaciones evaluadas fueran las mismas, sería posible determinar con mayor certeza una conclusión utilizando programación lineal o probabilidad y estadística.

4.4.5 Razonamiento simbólico

Los sistemas expertos no comprenderían los razonamientos de forma declarativa, es por ello que se basan en variables que al ser igualadas a valores adquieran el carácter de símbolos. Como se puede observar, los tipos de razonamiento utilizados por los sistemas expertos son de tipo simbólico; en ellos se utilizan, además de operadores lógicos (.and., .or.), aritméticos (+, -, /, *) y relacionales (>, <, >=, <=, <>) se definen y utilizan

variables y constantes que el lenguaje o Shell no reconoce; toma valores sugeridos por nosotros.

Para el lenguaje o Shell, colocar una variable llamada "aceptar_proyecto" o llamada "x" es lo mismo. Sea cual sea el nombre de la misma, significa lo que nosotros queramos que signifique, lo cual lo hace un símbolo.

a) Características esenciales del conocimiento simbólicamente representado.

Antes de que los sistemas expertos puedan razonar simbólicamente el conocimiento, este debe estar representado en la computadora de tal manera que:

- La información sea almacenada y recuperada eficientemente.
- Que el conocimiento refleje de manera fiel un fenómeno.
- Que la computadora pueda entender el fenómeno, es decir, que pueda manejar la información de manera sustancialmente significativa.

b) Inexactitudes en la interpretación del conocimiento simbólicamente representado.

La representación del conocimiento no siempre es sencilla, ya que algunas cosas que para la percepción humana son evidentes, para la computadora puede ser muy complejo. De entre las mayores problemáticas que se tienen para formular el conocimiento, se tienen las siguientes:

- En ocasiones representar un fenómeno representa un problema de afirmaciones y excepciones que se contraponen. Se corre el riesgo de dar significados ambiguos o equivocados.
- Si alguien ajeno al fenómeno ve la representación simbólica del mismo, puede ver una realidad en esencia igual, y en forma diferente, a lo que llamamos el efecto Picasso; este efecto se caracteriza por la perdida en la observación de los detalles determinantes que forman la idea con respecto a algo.

- Una representación descriptiva del conocimiento puede representar fielmente un fenómeno, pero para la computadora eso no es conocimiento.
- El texto (secuencia de caracteres) no tienen significado sustantivo.
- La computadora no entiende el conocimiento expresado en forma declarativa y por tanto no puede manejarlo en una forma significativa.
- Una representación basada en modelos matemáticos del conocimiento puede ser interpretado y manipulado por la computadora, pero por lo general simplifica en demasía el fenómeno de tal forma que la representación no resulta ser fiel.

4.5 Definición de reglas y mecanismos de inferencia.

Ya vimos que las reglas son un elemento fundamental para la formulación del conocimiento a utilizar en los sistemas expertos; adicional a los conceptos, las reglas ilustran los procesos de razonamiento que el experto realiza en su mente para llegar a una conclusión, emula los caminos que sigue la información, recopilando restricciones y afirmaciones, definiendo nuevas, hasta el resultado final

que constituye la solución a un problema particular. El objetivo básico de las reglas es concluir algo a partir del cumplimiento de una condición.

4.5.1 Variables objetivo.

Lo primero que hay qué hacer para la definición efectiva de reglas es encontrar los objetivos (hechos determinables), a los que también podemos llamar **variables**. El siguiente ejemplo nos puede ayudar a comprender mejor este y otros conceptos que manejaremos en el camino.

Suposición:

Suponga que usted posee una base de conocimiento que tiene las particularidades generales relativas a las impresoras de uso para el hogar.

¿Cuáles serían sus variables? Esta pregunta se contesta si sabemos qué es lo que el usuario desearía **conocer**.

Algunas variables podrían ser:

- **impresora**: tipo de impresora de la que se trate.

- **calidad**: tipo de calidad de impresión.
- **precio** : precio de la impresora; puede ser un monto o un calificativo de relatividad.
- **modelo** : modelo de la impresora.
- **suministro**: tipo de consumible de impresión que requiere.

De cada variable por sí misma puede derivarse un árbol de decisión que nos apoye en la creación de reglas; así mismo, hay variables más generales que otras, por tanto, dichas variables serán las candidatas idóneas para ser origen del árbol. La figura 23 muestra una estructura de árbol, en donde X es una variable más general que Y o Z.

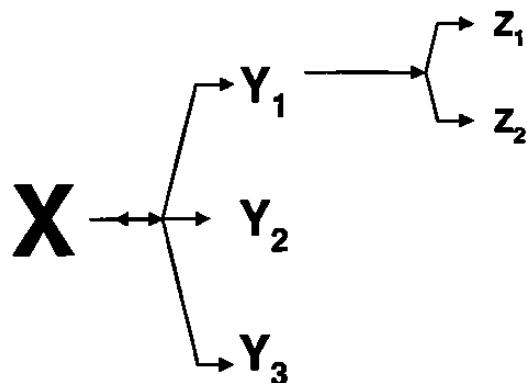


Figura 23: Estructura de árbol existente entre variables generales y particulares.

En el caso de la elaboración de reglas debe considerarse esto como una jerarquía, es decir es más fácil saber si una impresora es laser o de inyección de tinta que saber si el suministro es cartucho de tinta o torner.

Por lo general el proceso de inferencia realiza una especialización; casi siempre se procede de conocimientos generales a obtener otros más particulares.

Algunas reglas aplicables al ejemplo que se plantea pudieran ser las que se ilustran en la figura 24:

Rule 1	Rule 5
if impresora = laser	if modelo = HP_LaserJet
then calidad = alta.	then impresora = inkjet.
Rule 2	Rule 6
if impresora = laser	if modelo = HP_DeskJet
then precio = alto.	then impresora = inkjet.
Rule 3	Rule 7
if impresora = inkjet	if impresora = laser
then calidad = alta.	then suministro = torner.
Rule 4	Rule 8
if impresora = inkjet	if impresora = inkjet
then precio = alto.	then suministro = cartucho_tinta.

Figura 24: Ejemplo de reglas.

4.5.2 Métodos de inferencia utilizados por los sistemas expertos.

4.5.2.1 Búsqueda de variable objetivo como propósito de los sistemas expertos.

El problema que un sistema experto debe pretender resolver debe basarse en un requerimiento de información que haga referencia a una de las variables definidas como elementales para la definición del problema, por ejemplo:

Si tengo una impresora HP_DeskJet,

¿Qué suministro requiero?

Si nos damos cuenta, SUMINISTRO es una de las variables que podemos utilizar como objetivo, y puede ser algo que el usuario del hogar quisiera saber en un momento dado, pero que por quedar fuera de su área de especialización, no conoce. Se tienen dos opciones: conseguir el consejo de un experto en impresoras, o bien ejecutar un sistema experto que en base a ciertas preguntas concluya lo que necesitamos saber.

4.5.2.2 Métodos de inferencia.

Existen dos métodos de inferencia utilizados con los RBS:

- Forward Chaining (encadenamiento hacia delante)
- Backward Chaining (Encadenamiento hacia atrás)

Los métodos de inferencia determinan la secuencia de evaluación de las reglas, así como el tratamiento de la memoria de trabajo y la forma de evaluación de las condiciones antecedentes. Estos métodos se detallarán a continuación:

4.5.2.2.1 Forward Chaining

En Forward chaining, la deducción está basada en los **datos conocidos (hechos/facts)**; el objetivo buscado no participa activamente en el proceso de inferencia; en esta estrategia, el mecanismo de inferencia sólo verifica en el contenido de la memoria de trabajo al principio del proceso y después de resolver cada una de las reglas, para ver si la variable objetivo ha sido resuelta.

Con este método, la memoria de trabajo puede contener hechos predefinidos antes de comenzar el proceso de inferencia, de tal suerte que si la solución del problema, es decir, si la variable que buscamos ya se encuentra con algún valor en la memoria de trabajo, no será necesario iniciar el proceso de inferencia; el proceso de inferencia se detiene cuando la variable objetivo ha sido resuelta. La figura 25 muestra la forma de trabajo con forward chaining.

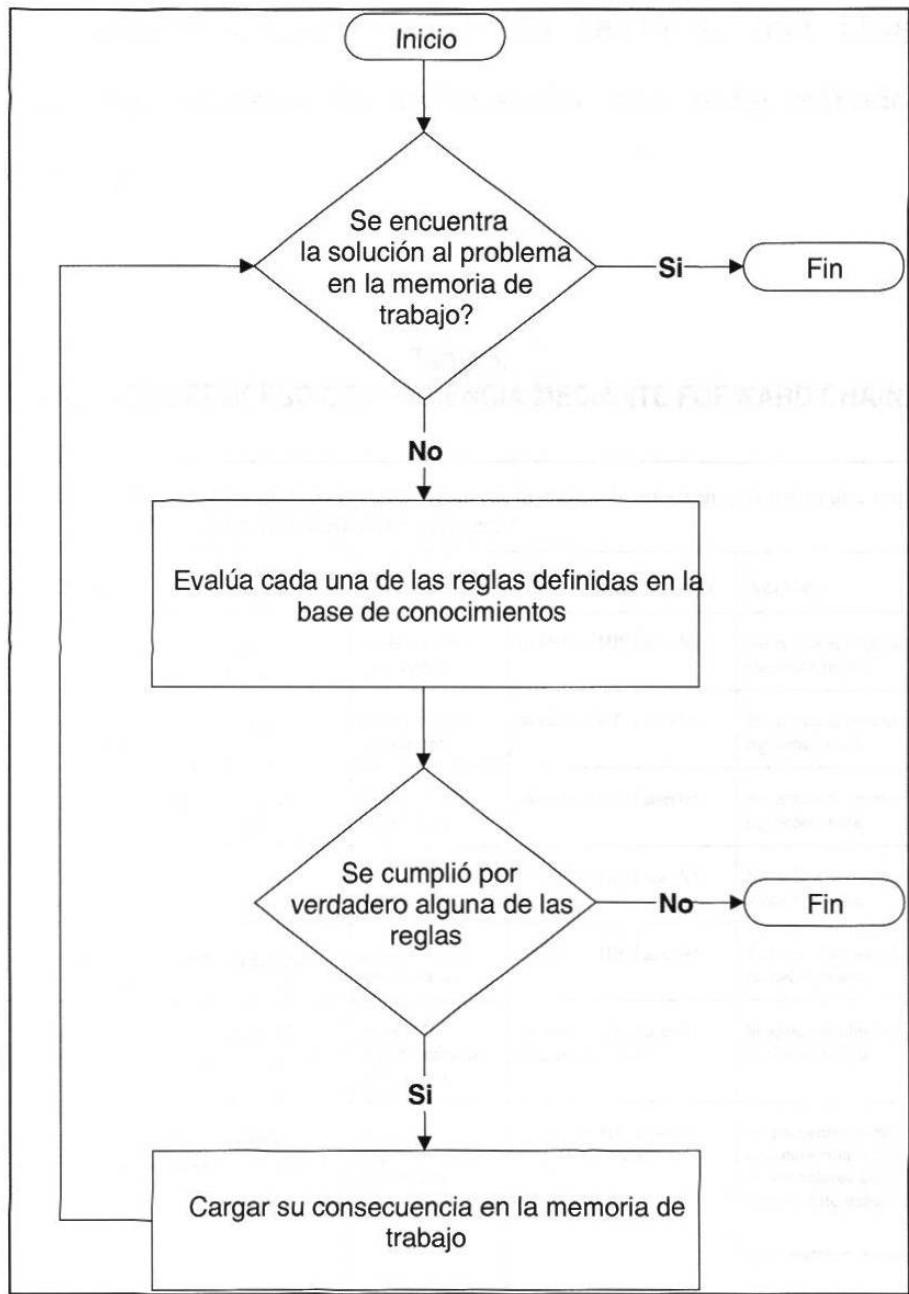


Figura 25: Forma en que trabaja el método de encadenamiento hacia delante (forward chaining) .