



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE HIDALGO

INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍA

**CENTRO DE INVESTIGACIÓN EN TECNOLOGÍAS DE
INFORMACIÓN Y SISTEMAS**

***CONSULTAS A BASES DE DATOS BASADAS EN
ESTRUCTURAS DE REPRESENTACIÓN DEL CONOCIMIENTO***

T E S I S

presentada en opción al Grado de Maestro en Ciencias Computaciones

Autor: Marco Antonio Pérez Bustos.

Director de Tesis: M. en C. María de los Ángeles Alonso

Tutor: Dr. Argelio V. de la Cruz Rivera.

**Pachuca de Soto Hgo., julio de 2003.
México.**





Universidad Autónoma del Estado de Hidalgo
Centro de Investigación en Tecnologías de
Información y Sistemas



Oficio No. CITIS-0472/2003

**Lic. en Comp. Marco Antonio Pérez Bustos
PRESENTE.**

Por este conducto le comunico que el jurado asignado para la revisión de su trabajo de tesis titulado "**Consultas a Bases de Datos Basadas en Estructuras de Representación del Conocimiento**", que para obtener el grado de Maestro en Ciencias Computacionales fue presentado por usted, ha tenido a bien, en reunión de sinodales, autorizarlo para impresión.

A continuación se anotan las firmas de conformidad de los integrantes del Jurado:

PRESIDENTE: Dr. Ornar López Ortega.

PRIMER VOCAL: Dr. Gustavo Núñez Esquer.

SECRETARIO: Dra. Aurora Pérez Rojas

PRIMER SUPLENTE: M. en C. Mariano J. Pozas Cárdenas.

SEGUNDO SUPLENTE: M. en C. Ma. De los Ángeles Alonso
Lavernia

ATENTAMENTE
"AMOR, ORDEN Y PROGRESO"
Pachuca, Hgo. a 9 de Julio de 2003.

Dr. Roberto A. Hernández Gómez
Coordinador de la Maestría en Ciencias Computacionales.

c.c.p. M. en D. Adolfo Pontigo Loyola.- Director de Control Escolar
c.c.p M. en C. Raúl García Rubio.-Director del ICBI
c.c.p. Archivo

A mis primeros
maestros: Socorro y
Salomón.

Sentir gratitud y no expresarla es como envolver un regalo y no darlo.

William Arthur Ward.

Doy gracias a Dios, por iluminar mi mente y mi camino en todo momento.

Expreso mi gratitud a todas aquellas personas que siempre confiaron en mí y que me ayudaron a concluir este trabajo, en especial a:

Mis padres, por todo lo que me han dado y por alentarme siempre.

A mis hermanos, por su comprensión y apoyo durante todo este tiempo.

A mi maestra María de los Ángeles, por creer en mí y darme la oportunidad de trabajar juntos para que este trabajo tuviera éxito.

Al Dr. Argelio V., por todo el conocimiento que compartió conmigo y por orientarme durante el desarrollo de este trabajo.

A mis compañeros de generación, por todos los días que compartimos, las vivencias, los desvelos, las emociones y los buenos momentos.

A mis amigos, por las palabras de aliento que siempre recibí de ellos.

A mis revisores de tesis, por sus recomendaciones y tiempo que dedicaron a este trabajo.

Gracias a todos ustedes por ayudarme a alcanzar una meta mas en mi vida.

Marco Antonio.

Resumen

El hombre continuamente obtiene información de los datos recopilados a partir de la observación de objetos, fenómenos o entes que lo rodean. Las organizaciones utilizan esta información para apoyar la toma de decisiones. Frecuentemente se llega a una conclusión a partir de análisis matemáticos de los datos contenidos en una base de datos y de un razonamiento lógico por parte del tomador de decisiones. Proporcionando así, resultados al aplicar su experiencia sobre alguna situación en particular, que puede ser representada en una base de conocimiento. Sin embargo, dentro de las aplicaciones computacionales existentes, comúnmente los datos y la experiencia son utilizados por separado.

El medio ambiente para la programación del conocimiento "HARies" es un sistema de propósito general, que se utiliza para el desarrollo de aplicaciones inteligentes en cualquier dominio del conocimiento, donde la experiencia juega un papel fundamental.

Este ambiente cuenta con diversas estructuras que permiten representar múltiples situaciones de la vida real, no obstante, no contaba con la posibilidad de realizar consultas a bases de datos, de donde se puede obtener una gran cantidad de información que apoye la solución de problemas.

Durante el desarrollo del presente trabajo se planteó como objetivo fundamental, implementar diversas estructuras para la representación del conocimiento, que permitieran la integración de Bases de Datos y Bases de Conocimiento, con la finalidad de abordar la solución de problemas específicos, donde se requiera utilizar la información contenida en ambas fuentes de información.

Para ello, se adicionaron al medio ambiente HARies nuevas estructuras para la representación del conocimiento, lo cual involucró el desarrollo de sus respectivos sistemas para la adquisición y procesamiento, necesarios para definir los parámetros requeridos durante una consulta específica y para procesar el conocimiento acerca de la información que se puede obtener de una base de datos, que será utilizada dentro de la base de conocimiento.

En cuanto a la consulta a fuentes de datos, se creó un módulo que se encarga de la comunicación con bases de datos desarrolladas en diversos sistemas manejadores de bases de datos, proporcionando información a la base de conocimiento a partir de los parámetros contenidos en cada una de las estructuras implementadas para ello.

Dichas estructuras se han definido como variable extracción de datos y variable cuantificador. La primera es la encargada de obtener datos que serán asignados a estructuras contenidas en una base de conocimiento y la segunda para realizar la cuantificación de registros dentro de la base de datos que cumplen determinadas propiedades. Estas estructuras han sido utilizadas para el desarrollo de diversas aplicaciones en medicina, educación, industria y servicios, resultados que también son presentados para ilustrar el potencial de las estructuras que se implementaron.

Contenido

Introducción	I
Objetivo General	VI
Objetivos Específicos	VI
Capítulo I: <i>Revisión Bibliográfica</i>	1
1.1. Introducción	1
1.2. Técnicas de IA	2
1.2.1. Redes Neuronales Artificiales	2
1.2.2. Algoritmos Genéticos	3
1.2.3. Minería de Datos.....	4
1.2.4. Sistemas Basados en Conocimiento	5
1.3. Representación del Conocimiento	6
1.3.1. Reglas de producción.....	8
1.3.2. Redes Semánticas	9
1.3.3. Frames	10
1.3.4. Scripts	10
1.3.5. CYC.....	11
1.4. Adquisición y Procesamiento del Conocimiento	11
1.5. Integración de Datos y Conocimiento	12
Capítulo II: <i>Estructuras de Representación del Conocimiento para la Manipulación de Bases de Datos desde Bases de Conocimiento</i>	14
2.1. Introducción	14
2.2. Lenguaje para la Programación del Conocimiento. Medio Ambiente HArries	14
2.3. Manipulación de BDs desde HArries	19
2.4. Estructuras para la Representación del Conocimiento de HArries que se Utilizan en la Propuesta	19
2.4.1. Estructura Proposición.....	20
2.4.2. Estructura Variable	21
2.5. Estructuras para la Manipulación de BD	22
2.5.1. Estructura Variable Extracción de Datos	23
2.5.1.1. Ejemplo.....	25

2.5.2. Estructura Variable Cuantificador.....	26
2.5.2.1. <i>Cuantificador es en el Cálculo de Predicados</i>	26
2.5.2.2. <i>Cuantificadores Parametrizados</i>	27
2.5.2.2.1. <i>Cuantificador Existencial Parametrizado</i>	28
2.5.2.2.2. <i>Cuantificador Universal Parametrizado</i>	30
2.5.2.3. <i>Relación con la Teoría de Tipos</i>	31
2.5.2.4. <i>Tratamiento de Incertidumbre en los Cuantificadores</i>	31
2.5.2.5. <i>Variable Cuantificador</i>	34
2.6. Ventajas de la Incorporación de las Estructuras para la Manipulación de BDs.....	35

Capítulo III: *Diseño y Funcionamiento de las Variables Extracción de Datos y Cuantificador dentro del Sistema HAriesA*..... 36

3.1. Introducción.....	36
3.2. Sistema para la Adquisición de Conocimiento HAriesA.....	36
3.3. Diseño del Módulo para la Adquisición de la Variable Extracción de Datos.....	39
3.4. Funcionamiento del Módulo para la Adquisición de la Variable Extracción de Datos.....	42
3.5. Diseño del Módulo para la Adquisición de la Variable Cuantificador.....	44
3.6. Funcionamiento del Módulo para la Adquisición de la Variable Cuantificador.....	48

Capítulo IV: *Diseño y Funcionamiento de las Variables Extracción de Datos y Cuantificador dentro del Sistema HAriesC*..... 51

4.1. Introducción.....	51
4.2. Sistema Consultante HAriesC.....	51
4.3. Diseño del Módulo para el Procesamiento de la Variable Extracción de Datos.....	53
4.4. Funcionamiento del Módulo para el Procesamiento de la Variable Extracción de Datos.....	55
4.5. Diseño del Módulo para el Procesamiento de la Variable Cuantificador.....	56
4.6. Funcionamiento del Módulo para el Procesamiento de la Variable Cuantificador.....	58

Capítulo V: <i>Ambiente de Comunicación con ODBC</i>	60
5.1. Introducción	60
5.2. Ambiente de comunicación	60
5.2.1. Interfaz para la Adquisición	60
5.2.2. Interfaz para la Ejecución.....	62
5.3. Implementación del Ambiente de Comunicación	63
5.3.1. Comunicación Vía ODBC	64
5.3.1.1. <i>Fuentes de Datos</i>	64
5.3.1.2. <i>Tipos de Datos</i>	66
5.3.2. Diseño y Funcionamiento del Módulo de Comunicación en el Sistema de Adquisición	67
5.3.3 Diseño y funcionamiento del Módulo de Comunicación en el Sistema Consultante.....	69
5.3.3.1. <i>Diseño del Módulo para la Comunicación con ODBC</i>	71
5.3.4. Ventajas del Uso del Ambiente de Comunicación.....	73
 Capítulo VI: <i>Ejemplos de Aplicación de las Variables Extracción de Datos y Cuantificador</i>	 74
6.1. Introducción	74
6.2. Aplicaciones Construidas	74
6.2.1. Aplicaciones en Medicina	74
6.2.2. Aplicaciones en Educación	75
6.2.3. Aplicaciones en los Servicios	75
6.2.4. Aplicaciones en la Industria	76
6.3. Prototipo de un Sistema para el Pronóstico de Pacientes Enfermas de Cáncer Cérvico Uterino	77
6.4. Prototipo de un Sistema para la Evaluación del Riesgo de SIDA	82
6.5. Sistemas Educativos	84
6.6. Prototipo de un Sistema para el Control de la Calidad en los Servicios Ofrecidos en el Museo "El Rehilete"	86
6.7. Prototipo de un Sistema para el Control del Consumo de Agua	88
6.8. Prototipo de Sistema para el Control de Autos Robados	89
6.9. Sistema para Pronósticos en Pozos de Petróleo	91
6.10. Ejemplo para la consulta a una BD de Alumnos	94
Conclusiones.....	101
Trabajos Futuros	101
Bibliografía	102
Apéndice A. Teoría de Tipos	108

Índice de Figuras

Capítulo II

Figura 2.1 Pantalla principal del sistema HAriesA	16
Figura 2.2 Adquisición del conocimiento	17
Figura 2.3 Pantalla principal del sistema HAriesC	18
Figura 2.4 Puesta a punto del conocimiento	18
Figura 2.5 Consulta del conocimiento	19
Figura 2.6 Perspectiva general para la consulta de un sistema inteligente en HAries	20
Figura 2.7 Formas para la representación del conocimiento en HAries.....	23
Figura 2.8 Tabla a consultar	25
Figura 2.9 Funcionamiento de los cuantificadores parametrizados.....	28
Figura 2.10 Esquema para el cálculo del valor de certidumbre de la proposición asociada a la variable.....	33

Capítulo III

Figura 3.1 Caso de Uso: Manipulación de las estructuras tipo variable en el sistema HAriesA	37
Figura 3.2 Jerarquía de clases de las estructuras del tipo variable en HAriesA.	38
Figura 3.3 Caso de Uso: Edición de la variable extracción de datos.....	40
Figura 3.4 Diagrama de clases de la variable extracción de datos	41
Figura 3.5 Diagrama de Actividad: Proceso de creación o edición de la variable extracción de datos	43
Figura 3.6 Interfaz de Adquisición de la variable extracción de datos.....	44
Figura 3.7 Caso de Uso: Edición de la variable cuantificador	45
Figura 3.8 Diagrama de clases de la variable cuantificador	47
Figura 3.9 Diagrama de Actividad: Proceso de creación o edición de la variable cuantificador.....	49
Figura 3.10 Interfaz para la adquisición de la variable cuantificador.....	50

Capítulo IV

Figura 4.1 Jerarquía de clases de las variables en HAriesC	52
Figura 4.2 Caso de Uso: Procesamiento de la variable extracción de datos.....	53
Figura 4.3 Diagrama de Clases de la variable extracción de datos (VarDatosBD).....	54
Figura 4.4 Diagrama de Actividad: Procesamiento de la variable extracción de datos.....	55

Figura 4.5 Caso de Uso: Procesamiento de la variable cuantificador.....	57
Figura 4.6 Diagrama de clases de la variable cuantificador (VarCuantificador). 58	
Figura 4.7 Diagrama de Actividad: Procesamiento de la variable cuantificador. 59	

Capítulo V

Figura 5.1 Interfaz para la conexión con la BD para la variable cuantificador. ...	62
Figura 5.2 Interfaz para la conexión con la BD durante la ejecución o procesamiento de las variables	63
Figura 5.3 Administrador de orígenes de datos ODBC.....	65
Figura 5.4. Caso de Uso: Definición de la conexión con la BD.....	67
Figura 5.5. Diagrama de clases: Clase ConexionBD dentro de HAriesA	68
Figura 5.6. Interacción durante la consulta del Módulo de Comunicación	70
Figura 5.7. Diagrama de clases : Clase ConexionBD dentro de HAriesC.....	71
Figura 5.8. Diagrama de clases de la DLL para la conexión con la BD.....	72

Capítulo VI

Figura 6.1 Adquisición de la variable cuantificador requerida para evaluar a las pacientes de la Jurisdicción Tula con un resultado positivo	81
Figura 6.2 Un resultado obtenido por el Prototipo de Sistema para el Pronóstico de Pacientes Enfermas de Cáncer Cérvico Uterino	81
Figura 6.3 Un resultado proporcionado por el Prototipo de Sistema para la Evaluación del Riesgo de SIDA.....	83
Figura 6.4 Un resultado de la evaluación en el tema "Introducción a la Trigonometría".....	86
Figura 6.5 Un resultado de la evaluación de los servicios que ofrece el museo ..	87
Figura 6.6 Un resultado otorgado por el Prototipo de Sistema para el Control del Consumo de Agua.....	89
Figura 6.7 Un resultado brindado por el Sistema para el Control de Autos Robados.....	90
Figura 6.8 Ejemplo de una pantalla de conclusiones brindada por el Sistema para Pronósticos en Pozos de Petróleo	91
Figura 6.9 Variable extracción de datos para obtener los datos relacionados con el tipo de crudo	92
Figura 6.10 Ejemplo de los datos extraídos de una BD	93
Figura 6.11 Tabla <i>Calificaciones</i> de la BD <i>Alumnos</i>	95
Figura 6.12 Ejemplo de los datos obtenidos para un alumno determinado	95
Figura 6.13 Tabla <i>Alumnos</i> de la BD <i>Alumnos</i>	97
Figura 6.14 Ejemplo de los resultados que proporciona el sistema.....	97
Figura 6.15 Ejemplo de los resultados que proporciona el sistema.....	100
Figura 6.16 Variables utilizadas para la consulta a la BD de Alumnos.....	100

Introducción

El desarrollo tecnológico de los últimos años, ha impulsado el trabajo del hombre en su afán de encontrar soluciones a problemas que enfrenta cotidianamente, adquiriendo tal importancia las computadoras, que simplemente no se concebiría un mundo globalizado sin las mismas. Por esta razón, día a día se investigan, desarrollan y utilizan nuevas técnicas computacionales con el objetivo de mejorar las actividades que desempeña el hombre.

Dentro de las actividades que realiza el hombre en las organizaciones, se encuentran el diagnóstico, pronóstico y toma de decisiones, entre otras. Estas tareas generan una gran cantidad de datos como resultado directo de sus operaciones cotidianas, como lo son: inventarios, manejo de nómina, formularios que contienen información acerca de los clientes, estadísticas obtenidas, registros para el control de facturación, garantías otorgadas, seguimiento de reclamos y devoluciones, por mencionar algunos.

Sin embargo, la gran cantidad de datos acumulados representa una riqueza en información que frecuentemente no es aprovechada en su totalidad, debido a que el conocimiento implícito en los datos no es fácil de descubrir de forma manual por el hombre.

Para lograr un mejor aprovechamiento de la información, se desarrolla una gran cantidad de aplicaciones computacionales que apoyan la toma de decisiones, principalmente las Bases de Datos (BD), las cuales han sido utilizadas en negocios, medicina, ingeniería, política y muchas otras áreas. Éstas tienen la tarea de proporcionar las bases para fundamentar las alternativas elegidas para la solución de problemas en determinado momento, además de proveer un mejor y más rápido acceso a la información necesaria.

Una BD permite el almacenamiento de las propiedades que caracterizan a un problema dado, la cual es de gran utilidad en los sistemas para la contabilidad, el manejo de la información, el soporte en la toma de decisiones, sistemas estadísticos y operacionales en general.

Sin embargo, el esfuerzo que se tiene que emplear para la obtención de una gran cantidad de datos, es muy difícil de comparar en relación con la información o conocimiento que se puede destilar de la BD. Por ejemplo, al realizar una consulta determinada se debe recurrir, en la mayoría de los casos, al análisis de ésta para que sirva de base en una toma de decisiones, resultando necesaria la experiencia de los administradores.

La mayoría de estos sistemas computacionales, poseen la característica principal de ser, esencialmente, de naturaleza transaccional. Puesto que la información proveniente de estos sistemas no puede ser utilizada sin un tratamiento previo, los administradores de las organizaciones utilizan su conocimiento y experiencia para tomar decisiones, logrando así los objetivos y metas que se han trazado.

Algunas BD son tan grandes que los administradores del sistema no siempre saben que información ha sido representada o qué tan relevante podría ser para responder a determinadas preguntas, haciéndose evidente una vez más la necesidad de utilizar la experiencia de los administradores para el procesamiento de la información obtenida a partir de los datos.

La necesidad de utilizar la gran cantidad de información contenida en las BD y la aplicación de la experiencia humana representada mediante Bases de Conocimiento (BC), para la solución de problemas específicos dentro de un área del conocimiento, conlleva a considerar la alternativa de unificar ambas herramientas (BD y BC) para lograr resultados mejor fundamentados y, con esto, servir a los fines para los cuales han sido definidas dichas aplicaciones.

Por otro lado, si analizamos el caso de los tomadores de decisiones dentro de las organizaciones, necesitan de los datos generados por las actividades en las que se encuentran involucrados, para desempeñar sus actividades de la mejor manera posible, resultando que la experiencia necesariamente requiere de datos para proporcionar resultados satisfactorios.

De aquí que se pueda asegurar que no sólo se necesitan datos como fuente de información en el proceso de búsqueda de soluciones de distintos problemas, sino que además se requiere de la experiencia y el conocimiento de los expertos que se desempeñan dentro del dominio de aplicación.

Queda claro que dentro de las organizaciones, los datos almacenados en las BD y la experiencia de los tomadores de decisiones, son fuentes de información que deben tomarse en cuenta para la solución de problemas, ya que ninguna puede ni debe ser despreciada en el momento de realizar la búsqueda de soluciones.

La integración de la información contenida en los datos y el conocimiento acerca de un dominio específico, se ha contemplado para desarrollar métodos y sistemas computacionales que sean capaces de brindar respuestas a problemas, en donde se involucre la toma de decisiones, con la finalidad de reducir las limitantes en cuanto al uso de la información que se encuentra almacenada en las BD y el uso de la experiencia de los tomadores de decisiones.

Una de las disciplinas de la computación que proporciona gran variedad de técnicas que permiten manipular conocimiento y datos es la inteligencia artificial (IA).

La IA es la disciplina que estudia cómo hacer que las computadoras hagan las cosas que hasta el momento el hombre hace mejor [52]. Esta área del saber cuenta con una variedad de técnicas que, de forma general, están encaminadas a simular el comportamiento humano en las máquinas, proporcionando las herramientas para obtener resultados a partir de conocimientos específicos sobre un problema dado, aportando técnicas para representar y procesar el conocimiento. Entre éstas, se puede mencionar a los sistemas basados en conocimiento (SBC), aprendizaje automático, redes neuronales artificiales, algoritmos genéticos y lógica difusa.

La idea básica de la IA es que la computadora puede ser programada para desarrollar tareas que sean consideradas como inteligentes. Es decir, se trabaja con la idea de desarrollar sistemas que realicen funciones como las llevan a cabo los especialistas dentro de determinadas áreas. Por ejemplo, dentro de las organizaciones, un sistema desarrollado con técnicas de IA puede proveer la misma asistencia a un administrador como la realizaría un consultor gerencial.

Los SBC suministran una variedad de métodos para codificar y procesar el conocimiento en sistemas computacionales, con el objetivo de proporcionar herramientas para resolver problemas empleando conocimiento.

Un SBC es un programa en donde se representa y se realiza inferencia a partir del conocimiento de un tema especializado, con una visión para resolver problemas o dar consejos. Éste se distingue por simular el razonamiento humano sobre un dominio del problema, razonar sobre representaciones del conocimiento humano y resolver problemas por heurística o métodos de aproximación [35].

En una BC se representa el conocimiento del dominio para la solución de problemas, donde la experiencia juega un papel importante y decisivo para la solución de éstos, resolviéndose de esta forma el área del conocimiento que no abarcan las BD.

Se puede considerar que una BC es el ente físico utilizado para almacenar el conocimiento obtenido a partir de las experiencias proporcionadas por expertos humanos dentro de un área específica. Ésta representa un depósito de las estructuras del conocimiento disponibles para un SBC [53].

Dado que el conocimiento es primordial para definir como "inteligente" el comportamiento de los sistemas computacionales, la representación del conocimiento es uno de los temas principales de investigación dentro de los SBC, que trata con los problemas de representar, mantener y manipular conocimiento acerca de un dominio de aplicación. Resultando así que la representación, adquisición y el procesamiento del conocimiento, son las actividades fundamentales dentro de los SBC.

Los SBC cuentan con estructuras para la representación del conocimiento que se encuentran separadas de los mecanismos para ejecutar el razonamiento, resultando necesarias ambas componentes para completar el procesamiento en una sesión de consulta con el sistema. Una forma para la representación del conocimiento consta de una estructura propia para la definición del conocimiento y de una máquina de inferencia encargada de realizar su procesamiento.

En los SBC los métodos para representar el conocimiento incluyen aproximaciones clásicas como son: la lógica de predicados, reglas de producción, redes semánticas, frames y scripts [52].

Para proveer de conocimiento a los SBC, se realiza la adquisición del conocimiento, que se refiere a las técnicas mediante las cuales los sistemas computacionales pueden obtener el conocimiento que necesitan para llevar a cabo sus tareas, esto va desde la interacción con el experto hasta su incorporación a la BC.

Los sistemas encargados de la inferencia y ejecución del razonamiento sobre el conocimiento adquirido dentro de un SBC, realizan el procesamiento de las estructuras contenidas en la BC durante su ejecución, con la finalidad de obtener las respuestas para la cual han sido desarrollados. Dichos subsistemas son conocidos como "Máquinas de Inferencia".

Una máquina de inferencia sirve para el procesamiento y como mecanismo de control en el SBC; es una parte esencial del sistema, así como un factor principal en la determinación de la efectividad y eficacia de dichos sistemas [34].

Podemos concluir que los SBC son sistemas que simulan el razonamiento humano en las computadoras, y tienen la capacidad de representar, adquirir, mantener y procesar el conocimiento. Proporcionan además, una variedad de métodos para obtener resultados a partir del conocimiento sobre un dominio de aplicación.

Por otra parte, el aprendizaje automático es otra de las técnicas dentro de la IA que permite la integración de datos y conocimiento. Sin embargo, éste se ha enfocado al desarrollo de algoritmos autónomos para la adquisición de conocimiento desde los datos y para la organización y compilación del conocimiento [58]. El campo del aprendizaje automático pretende construir programas computacionales que mejoren automáticamente con el uso de la experiencia.

Las redes neuronales artificiales son programas de IA, capaces de simular algunas de las funciones de aprendizaje del ser humano [33]. Sin reglas convencionales, una red neuronal artificial obtiene ciertos conocimientos analizando gran cantidad de datos, para determinar reglas de comportamiento. En base a estas reglas, puede realizar predicciones sobre nuevos casos. El conocimiento en una red neuronal se encuentra distribuido a través de la estructura de la red, haciendo muy difícil separar los conceptos individuales en sus regiones específicas, puesto que pueden representar conceptos diferentes y variados [46].

Los algoritmos genéticos son procedimientos matemáticos aplicables a problemas de optimización, que utilizan como metáfora la teoría de la evolución de Darwin, operando esencialmente en un ciclo simple de selección y reproducción, involucrando una recombinación y mutación del material genético de las soluciones. Una población de posibles soluciones se genera al azar, se avalúan con respecto a un objetivo y las más aptas se combinan entre sí para producir nuevas soluciones. El ciclo se repite hasta llegar a una solución aceptable o al determinarse el óptimo de una función. Se pueden encontrar soluciones aproximadas a problemas de gran complejidad computacional mediante un proceso de evolución simulada.

La lógica difusa suministra un conjunto de técnicas matemáticas para la representación y tratamiento de datos que no tienen una precisión definida y concreta. La lógica difusa tiene una importancia significativa en lo que se refiere al procesamiento de datos, ya que en muchas áreas del conocimiento es necesario asociar valores de verdad (grados de veracidad o falsedad), más amplios que los significados de falso o verdadero, esto con el objetivo de describir el grado de certidumbre asociado a los resultados que proporciona el sistema que emplea lógica difusa, en términos de los datos de entrada que le han sido

proporcionados. Con esta técnica se hace posible la representación del conocimiento inexacto, logrando así traducir el conocimiento de un dominio de la misma manera de cómo se utiliza en un lenguaje usual determinado.

Otra de las principales áreas de las ciencias computacionales, que no forma parte de la IA, utilizadas para el procesamiento de la información es la minería de datos [61].

La minería de datos proporciona algoritmos para desarrollar sistemas de búsqueda de conocimiento, tratando de determinar relaciones y patrones de comportamiento en bases de datos, de una forma inteligente y automática. Esta técnica consiste en un proceso interactivo donde el progreso es definido por descubrimiento, a través de métodos automáticos o manuales [60], resultando ser más útil en escenarios de exploración de datos en donde no existen nociones predeterminadas acerca de que resultados se obtendrán. Dentro de la minería de datos se utilizan con frecuencia diversas técnicas de IA, como método de búsqueda del conocimiento [37].

Existen trabajos dirigidos hacia la integración de bases de conocimiento con bases de datos. Algunos de estos proponen la inclusión de los datos en las bases de conocimiento, por medio de estructuras que permitan incorporar datos como parte de la misma. Por ejemplo, se ha propuesto una metodología unificada, que representa los datos, la información y el conocimiento de una forma homogénea, así como también las relaciones entre ellas [28].

En otros trabajos se obtiene conocimiento a partir de los datos contenidos en una fuente específica, definiendo relaciones y patrones entre estos, proporcionando resultados a partir del conocimiento encontrado en los depósitos de datos, entre ellos se encuentran una variedad de sistemas desarrollados con técnicas de minería de datos [36].

También se ha planteado la integración de datos a partir de sistemas que contienen conocimiento acerca de las relaciones que existen entre fuentes de datos a ser consultadas, teniendo la tarea principal de unificar sólo datos, suministrando una interfaz uniforme para la consulta de múltiples fuentes de datos. Por ejemplo, los llamados sistemas de esquema global [9], en donde se especifican las relaciones entre el esquema global y las fuentes a ser consultadas, con la finalidad de interpretar las consultas definidas en el primero.

Cada uno de los trabajos antes mencionados contiene tanto estructuras como técnicas propias, que han sido definidas para lograr dichos resultados.

La aproximación planteada en este trabajo corresponde al desarrollo de un sistema para la adquisición y procesamiento de estructuras para la representación del conocimiento. Éstas formarán parte de una BC con la finalidad de obtener información de una BD, y de esta manera proporcionar soluciones integrales en la toma de decisiones, fundamentada tanto en los datos como en la experiencia.

El trabajo expuesto no constituye un desarrollo aislado, ya que complementa las estructuras para la representación del conocimiento que contiene el medio ambiente HArises, el cual es un lenguaje para la programación del conocimiento que permite la construcción de sistemas inteligentes basados en conocimiento.

Este medio ambiente proporciona una variedad de estructuras que permiten representar el conocimiento acerca de un dominio en particular. Sin embargo, no contaba con representaciones del conocimiento para realizar consultas a BDs, para obtener información que en determinado momento la BC necesite para otorgar soluciones a los problemas planteados.

Para lograr lo anterior han definido dos estructuras: una denominada "variable extracción de datos" y otra llamada "variable cuantificador". La primera se ha definido para obtener valores de una BD y relacionarla con diferentes estructuras dentro de la BC y la segunda, para realizar consultas que sean capaces de cuantificar la cantidad de registros de la BD que cumplan con determinadas condiciones para posteriormente utilizar el resultado obtenido dentro de la BC.

La implementación de dichas estructuras fue posible debido a que se cuenta con el código fuente y la estructura del medio ambiente HArises, con lo cual se pueden realizar modificaciones como las planteadas en este trabajo.

Para el logro de los resultados obtenidos se plantearon los siguientes objetivos:

Objetivo General

Implementar las estructuras para la representación del conocimiento variable extracción de datos y variable cuantificador dentro del medio ambiente HArises, que permitan la integración de Bases de Conocimiento y Bases de Datos, para la solución de problemas específicos.

Objetivos Específicos

1. Diseño e implementación de la estructura de representación del conocimiento Variable Extracción de Datos, para la transferencia de información de una base de datos a una base de conocimiento.
2. Diseño e implementación de las estructuras de dos cuantificadores parametrizados, existencial y universal, que permitan interrogar una base de datos desde una base de conocimiento, con manipulación de incertidumbre.
3. Desarrollo de una interfaz para la comunicación de una BC con diversos manejadores de bases de datos.
4. Desarrollo del sistema de adquisición para la captura y mantenimiento de las variables extracción de datos y cuantificador.
5. Implementar el procesamiento de las estructuras variable extracción de datos y variable cuantificador dentro del sistema consultante del medio ambiente HArises.

La implementación computacional de una estructura para la representación del conocimiento, implica el desarrollo de un sistema que se encargue de la adquisición del conocimiento que se almacenará en la estructura como parte de una BC; así como de una máquina de inferencia, capaz de procesar dicha estructura de manera que su ejecución cumpla con los objetivos para los cuales fue diseñada.

Es por ello que las variables extracción de datos y cuantificador necesitan tanto un sistema para la adquisición de conocimiento, como una máquina de inferencia. Estos componentes son necesarios para el procesamiento de los parámetros otorgados por el ingeniero del conocimiento (IC) durante el desarrollo de un sistema inteligente en el medio ambiente HAries.

En este documento, se presentan el diseño y construcción de los sistemas para la adquisición de la variable extracción de datos y la variable cuantificador, en los cuales se definen los requerimientos para consultar e interrogar a una BD, identificados previamente por el IC y que son necesarios para representar el conocimiento que se ha obtenido.

Por otra parte, se implementó una máquina de inferencia que se encarga de procesar los parámetros proporcionados por el IC, en cuanto a la definición realizada en el sistema para la adquisición del conocimiento de ambas variables y que será utilizada durante una ejecución de consulta de la BC por parte del usuario.

Para trabajar de una forma integrada con datos y conocimiento, se pensó en la posibilidad de que los datos podían provenir de BD desarrolladas con distintos manejadores y que también podrían estar almacenados en unidades remotas, para lo cual fue necesario el desarrollo de un módulo para realizar la conexión con las distintas fuentes de datos.

Este módulo puede ser activado por la máquina de inferencia, para realizar el procesamiento de la estructura correspondiente utilizando la conexión con el manejador de base de datos en cuestión y obtener la información de la BD requerida por cada una de las estructuras.

La implementación se realizó basándose en el paradigma de programación orientada a objetos, ya que beneficia el desarrollo del software. Los objetos bien diseñados, son la base para sistemas que se construyen a partir de módulos reutilizables, dando lugar a una mayor productividad. La reutilización de las clases que han sido probadas en circunstancias reales, en proyectos anteriores, dan lugar a sistemas de mayor calidad y con menos errores, la herencia hace posible definir sistemas más fiables, más fáciles de ampliar y menos costosos de mantener.

Para facilitar la construcción de las estructuras tratadas en este trabajo y la implementación de los nuevos módulos al medio ambiente HAries, se utilizó el lenguaje Visual C++ debido a que todas las características antes mencionadas cubren ampliamente la necesidad de proporcionar gran robustez a la aplicación desarrollada. Además, HAries utiliza el lenguaje Visual C++ para desarrollar las estructuras que lo componen.

Los beneficios aportados por el presente trabajo son:

- Se posibilita la construcción de sistemas donde se requiera que los resultados, otorgados para la solución de un de un problema, se encuentren influenciados tanto por los conocimientos del especialista en el dominio del problema, como por los datos provenientes de la observación de ese fenómeno.

- Se amplían las posibilidades de aplicación, puesto que el sistema es capaz de obtener información de la base de datos proporcionada por el usuario, debido al módulo de comunicación para el manejo de diversos manejadores de bases de datos.
- Se hace posible la consulta de BD para obtener valores que en determinado momento se necesitan dentro de los SBC.
- Se introduce una nueva variante para el desarrollo de sistemas inteligentes donde se involucre la toma de decisiones basada en datos y conocimiento, que consiste en la utilización de cuantificadores parametrizados y manejo de incertidumbre, implementándose dos nuevas estructuras de representación del conocimiento dentro del medio ambiente HAries.

La definición de las estructuras encargadas de explorar BD, se realiza de una forma muy sencilla, puesto que se utiliza un medio ambiente visual para ayudar al IC a identificar las componentes necesarias para realizar su definición.

Por último, el mantenimiento a la BC, en cuanto a la ejecución de consultas de la BD, resulta sencillo puesto que consiste en el cambio de valores a los parámetros de la estructura de las variables correspondientes.

El presente documento se encuentra organizado de la siguiente manera. En el Capítulo I se presenta una revisión bibliográfica acerca de los trabajos relacionados con las representaciones del conocimiento y las aproximaciones existentes en cuanto a la manipulación de BD se refiere.

El Capítulo II presenta el marco teórico de trabajo en el que se detallan las estructuras de representación del conocimiento que han sido implementadas, tanto para su adquisición como para su procesamiento, dentro el medio ambiente HAries.

Los Capítulos III y IV están dedicados a la descripción del diseño y funcionamiento de las variables extracción de datos y cuantificador. Aquí, también se explica el funcionamiento de los módulos encargados de adquirir y procesar dichas estructuras para la representación del conocimiento.

En el Capítulo V se explica el ambiente a través del cual el sistema consultante de HAries, podrá realizar la comunicación soportada por ODBC, con diversos manejadores de bases de datos a partir de la definición de las estructuras variable extracción de datos y variable cuantificador.

Es importante señalar que el único requerimiento para el funcionamiento del sistema, es contar con sistema operativo Windows 95 o superior.

El Capítulo VI se enfoca a la presentación de distintos casos de estudio de las variables extracción de datos y cuantificador, a partir de aplicaciones desarrolladas con el medio ambiente HAries, las cuales están encaminadas al apoyo para la toma de decisiones. Con el análisis de estos ejemplos reales se hace evidente el amplio rango de aplicación que tiene el sistema desarrollado.

Por último, se presentan las conclusiones y trabajos futuros derivados del presente trabajo.

Capítulo I

Revisión Bibliográfica.

1.1. Introducción.

En este capítulo se plantea la necesidad de la utilización tanto de datos como de conocimiento en el desarrollo de sistemas. Se presenta un panorama general de la Inteligencia Artificial (IA), así como también la revisión de algunas técnicas derivadas de esta disciplina poniendo un mayor énfasis en los Sistemas Basados en Conocimiento (SBC), las Formas para la Representación del Conocimiento (FRC) y las aplicaciones encaminadas a la solución de problemas donde intervienen tanto datos como conocimiento.

Considerando que la creación de una Base de Datos (BD) conlleva un proceso que involucra planeación, análisis, diseño e implementación, así como su mantenimiento y protección, implica un trabajo que regularmente no es proporcional con la cantidad y calidad de beneficios que reporta en términos del conocimiento que de estos datos se destila u obtiene.

Muchas organizaciones invierten infinidad de recursos en la construcción y mantenimiento de BD que contienen grandes cantidades de datos, surgiendo otra desventaja en cuanto al aprovechamiento de la información se refiere, puesto que frecuentemente todos estos datos no pueden ser analizados con las herramientas computacionales tradicionales con las que se cuenta.

Son bien conocidos los trabajos que se realizan, por ejemplo en disciplinas como minería de datos, aprendizaje automático, análisis exploratorio de datos y estadística por citar algunas, para lograr estos objetivos, los cuales van dirigidos hacia la obtención de información como apoyo para la toma de decisiones.

Sin embargo, en muchos casos estas disciplinas proporcionan resultados insuficientes para la solución del problema planteado, sencillamente porque se aplican directamente sobre las fuentes de información y no se toma en cuenta el posible conocimiento existente sobre el problema en particular, que ayuda de manera primordial en la búsqueda de soluciones más completas.

Constituye por tanto uno de los objetivos del presente trabajo brindar una aproximación a la interrogante de cómo combinar ambas fuentes de información (datos y conocimiento), para abordar en forma integrada la solución de problemas específicos.

1.2. Técnicas de IA.

La IA se preocupa por la conducta inteligente en la computadora. La conducta inteligente, a su vez, involucra percepción, razonamiento, aprendizaje, comunicación y desempeño en ambientes complejos. La IA tiene como uno de sus objetivos a largo plazo, el desarrollo de máquinas que puedan realizar tareas tan bien como el ser humano, o posiblemente mejor [46].

El uso de técnicas de IA, permite el desarrollo de sistemas computacionales que pueden percibir el medio ambiente a través de ojos y oídos artificiales, comunicarse en lenguaje natural, razonar, resolver problemas y formular planes, actuar en su medio ambiente para lograr sus metas, y aprender a partir de sus propias experiencias [58].

Una técnica de IA es un método que explota conocimientos que deben ser representados de tal forma que constituyan generalizaciones de los problemas, sean entendidos por las personas que lo proporcionan, sea fácil de corregir, permita su modificación para corregir errores y pueda ser utilizado en una gran cantidad de situaciones, aunque no sea exacto o completo [52].

Sin embargo, la utilización de datos como parte estas técnicas también juegan un papel fundamental, de aquí que existan investigaciones dirigidas al uso combinado de ambas fuentes de información (Datos y Conocimiento), para la solución de problemas.

Entre las principales técnicas enfocadas en este sentido se encuentran: Redes Neuronales Artificiales, Algoritmos Genéticos y Sistemas Basados en Conocimiento. Aunque en cada caso los objetivos que se buscan alcanzar son distintos.

Puesto que el presente trabajo también se encuentra enfocado a la integración de Datos y Conocimiento. A continuación describiremos algunas técnicas de IA con las que se encuentra relacionada esta tesis.

1.2.1. Redes Neuronales Artificiales.

La teoría y modelado de Redes Neuronales Artificiales (RNAs) está inspirada en la estructura y funcionamiento de los sistemas nerviosos, donde la neurona es el elemento fundamental [33].

Las RNAs han llegado a ser populares por sus diversas aplicaciones desde las predicciones financieras hasta la visión de máquina [4]. Están compuestas por unidades computacionales paralelas e interconectadas. Cada una de dicha unidades efectúa un número de operaciones simples y transmite sus resultados a las unidades vecinas. Aprenden a reconocer patrones por medio de un entrenamiento basado en varios ejemplos diferentes y son eficientes para reconocer patrones en un conjunto de datos.

Debido a su constitución, las RNAs simulan un gran número de características semejantes a las del cerebro humano. Por ejemplo, son capaces de aprender de la experiencia, de generalizar de casos anteriores a casos nuevos, de abstraer características esenciales a partir de entradas que representan información irrelevante. Esto hace que

ofrezcan numerosas ventajas y que este tipo de tecnología se esté aplicando en múltiples áreas [33]. Estas ventajas incluyen:

- Aprendizaje Adaptativo. Capacidad de aprender a realizar tareas basadas en un entrenamiento o una experiencia inicial.
- Autoorganización. Una red Neuronal puede crear su propia organización o representación de la información que recibe mediante una etapa de aprendizaje.
- Tolerancia a fallos. La destrucción parcial de una red conduce a una degradación de su estructura, algunas capacidades de la red se pueden retener, incluso sufriendo un gran daño.
- Operación en tiempo real. Los computadores neuronales pueden ser realizados en paralelo, y se diseñan y fabrican máquinas con hardware especial para obtener esta capacidad.
- Fácil inserción dentro de la tecnología existente. Se pueden obtener chips especializados para redes neuronales que mejoran su capacidad en ciertas tareas.

También son efectivas en el aprendizaje de patrones a partir de datos no lineales, incompletos, con ruidos y hasta compuestos por ejemplos contradictorios. La habilidad de manipular estos datos imprecisos, hace que las redes sean muy eficaces en el procesamiento de información.

1.2.2. Algoritmos Genéticos.

Los Algoritmos Genéticos son algoritmos matemáticos de optimización de propósito general inspirados en mecanismos naturales de selección y genética, proporcionando excelentes soluciones en problemas complejos con gran número de parámetros [42].

Los algoritmos genéticos conforman un paradigma de búsqueda probabilística, inspirada en la teoría de la evolución de Darwin. Una población de individuos, cada uno representando una posible solución a un problema, es creada inicialmente al azar.

Entonces se combinan las parejas de individuos (soluciones) para producir otro individuo (prole) para la próxima generación. Un proceso de mutación, también generada al azar, modifica la estructura genética de algunos miembros de cada nueva generación. En cada ciclo, las bondades de los individuos son evaluadas con respecto al objetivo buscado y el sistema se vuelve a ejecutar para docenas o cientos de generaciones.

Dado que la probabilidad que un individuo se reproduzca es proporcional a la posible solución que representa, la calidad de las soluciones mejora automáticamente en las generaciones sucesivas. El proceso se termina cuando una solución es aceptable o se determina el óptimo para la función.

El objetivo principal de un algoritmo genético es el de producir la evolución de una población de soluciones para un determinado problema, intentando encontrar nuevas

generaciones de soluciones que sean mejores que la anterior. Estos algoritmos operan en un ciclo simple: creación de la población inicial, selección y reproducción, éste último implicando una recombinación y mutación del material genético de las soluciones.

La reproducción, en estos algoritmos, se describe a través de tres operadores genéticos: reproducción, cruzamiento y mutación.

- Reproducción. Las cadenas con valores más altos tiene probabilidades más altas de contribuir con uno o más sucesores en la siguiente generación. Este operador es una versión artificial de la selección natural de Darwin. En las poblaciones naturales la aptitud esta determinada por la habilidad del organismo de sobrevivir y de llegar a la madurez y su subsecuente reproducción. En el sistema artificial, la función objetivo es el árbitro final en la vida o muerte de una cadena dada.
- Cruzamiento. Una vez que se ha realizado la reproducción, se procede a la reproducción sexual o cruzamiento de los individuos seleccionados. La población nueva intercambia material cromosómico y sus descendientes forman la siguiente generación.
- Mutación. Cuando un sistema se ha detenido en una estructura genética que se sospecha no es la óptima, o cuando el sistema se ha viciado con diversas cadenas muy parecidas, es necesario infiltrar mutaciones que reanimen el sistema. La mutación no permite la estabilización de poblaciones en soluciones locales.

Dentro de los algoritmos genéticos se han propuesto mecanismos para mejorar su eficiencia, puesto que se ignoraba la experiencia adquirida por el individuo durante su vida, pues se partía del hecho que la experiencia de un individuo afecta directamente a su descendencia, por lo que el conocimiento era hereditario y las generaciones siguientes no necesitarían adquirirlo.

1.2.3. Minería de Datos.

La Minería de Datos es definida como el proceso para descubrir patrones en los datos. El proceso puede ser automático o, usualmente, semiautomático. Los patrones descubiertos pueden ser significantes para obtener alguna ventaja, normalmente una ventaja económica [61].

La Minería de Datos comprende el proceso de extraer información nueva de grandes BD, validarla y hacerla comprensible para ser usada, es decir, permite conocer el comportamiento analizado y tomar decisiones.

La información obtenida puede ser usada para una predicción, obtener un modelo de clasificación e identificar relaciones entre distintos archivos.

La Minería de Datos consiste en la aplicación de un gran número de operaciones, cada una apoyada en una variedad de técnicas dependiendo del tipo de datos a ser examinados o el problema específico que se presente, así estos sistemas pueden integrar técnicas como reconocimiento de patrones, análisis de imágenes, procesamiento de señales,

técnicas de graficación, tecnología web, economía, negocios, bioinformática o psicología por mencionar algunas.

Dentro de las principales funcionalidades podemos incluir la descripción de clases, asociación, clasificación, predicción, agrupamiento (clustering), análisis de tendencias, análisis de desviación y análisis similares. Estas funcionalidades pueden ser utilizadas para especificar el tipo de patrones a ser encontrados dentro de las tareas de la minería de datos. En general, las tareas pueden ser clasificadas en dos categorías: descriptiva y predictiva. Las tareas descriptivas caracterizan las propiedades generales de los datos en la BD. Las tareas predictivas ejecutan inferencia sobre los datos actuales con la finalidad de realizar predicciones [37].

Existen una gran cantidad de aplicaciones tales como el análisis de mercado, análisis financiero, detección de fraudes, etc.

El proceso de Minería de Datos, puede dividirse en tres etapas:

1. Las distintas fuentes de datos deben integrarse de forma que estén representados todos los componentes de la organización. Esta visión de conjunto enriquece notablemente los resultados, en lugar de analizar información de un departamento o un sector específico. Frecuentemente es necesario aumentar las bases con datos externos.
2. Ejecutar el proceso de análisis de las bases de información.
3. Transformar las conclusiones obtenidas en reglas comprensibles que posibiliten la toma de decisiones.

1.2.4. Sistemas Basados en Conocimiento.

Los Sistemas Basados en Conocimiento constituyen una de las áreas más activas hoy en día dentro del campo de la IA, debido a que son utilizados en empresas, industrias, hospitales, oficinas y muchos otros sitios.

Es bien conocido que la cantidad de sistemas de este tipo es muy grande en la actualidad y que muchos de ellos se encuentran implantados en la práctica, rindiendo importantes beneficios en las esferas o dominios hacia los cuales se enfoca su acción.

La cuestión actual no radica ya en demostrar la idoneidad del enfoque y la nueva filosofía de abordar la solución a problemas que vienen tras los SBC, sino en el desarrollo de herramientas que posibiliten y de hecho faciliten su construcción y su manejo. Así como también, la interacción con las aplicaciones computacionales existentes, brindando con esto cada día más y mejores alternativas de acción para el logro de diversos objetivos.

Se puede describir un SBC como un programa de computadora que exhibe, dentro de un dominio específico, un grado de especialización en la solución de problemas, comparable con un experto humano.

Un sistema de este tipo se distingue de los programas de aplicaciones convencionales ya que simula el razonamiento humano dentro del dominio de un problema específico, en lugar de simular el dominio mismo, realiza razonamiento sobre representaciones del conocimiento humano, además de hacer cálculos numéricos o recuperación de los datos, resuelve además problemas por métodos heurísticos o de aproximación, distinto a las soluciones algorítmicas que no garantizan éxito [35].

Un SBC consta de dos componentes principales: los hechos y relaciones existentes en un dominio en particular y los mecanismos encargados de procesar el conocimiento proveniente del dominio en cuestión.

El conocimiento contenido en un SBC consiste de un conocimiento a priori: los hechos y las relaciones entre estos que son conocidas acerca de un dominio en específico antes de una sesión de consulta y de un conocimiento inferido: los hechos y las reglas concernientes a un caso específico que es deducido durante y en la conclusión de una consulta [34]. El conocimiento contenido en un sistema de este tipo, define los hechos y relaciones provenientes de un dominio de aplicación enfocados hacia la realización de una tarea dentro de un área específica. El conocimiento normalmente se expresa en algún lenguaje de propósito especial y se mantiene separado del código que ejecuta el razonamiento [35].

El conocimiento es representado teniendo en cuenta un enfoque general sobre cómo se resuelven los problemas, y se implementa con el proceso de razonamiento que genera soluciones a partir de la consulta de casos particulares sobre dicho conocimiento.

1.3. Representación del Conocimiento.

Existen muchas fuentes de información o conocimiento involucradas dentro de la IA. El campo de la representación del conocimiento (RC) comprende los mecanismos para la representación y manipulación de la información. Por ejemplo, en algunos casos la información puede ser representada por objetos (o variables) y por reglas lógicas (que representan las relaciones entre los objetos). De ésta manera la representación es apropiada para su manipulación utilizando un análisis lógico. Éste es el mecanismo de representación utilizado por ejemplo en los sistemas expertos basados en reglas [10].

Dado que el conocimiento es importante y primordial para el comportamiento inteligente, su representación constituye una de las máximas prioridades de la investigación en IA. El conocimiento puede ser representado como imágenes mentales en nuestros pensamientos, como palabras habladas o escritas en algún lenguaje, en forma gráfica o en imágenes, como cadenas de caracteres o colecciones de señales eléctricas o magnéticas dentro de una computadora y de otras innumerables formas.

La RC es el área de la IA que trata con los problemas de representar, mantener y manipular conocimiento acerca de un dominio de aplicación.

Los principales esfuerzos en RC están dirigidos hacia las áreas de aplicación (conocimientos médicos, conocimiento acerca del tiempo, conocimiento sobre sistemas

físicos), el desarrollo de lenguajes de representación apropiados, los mecanismos de razonamiento y el procesamiento de certidumbre [40].

Una RC ha sido definida como un conjunto de convenciones sintácticas y semánticas que hacen posible la descripción de las cosas [53]. En IA, las cosas normalmente significan el estado de algún dominio de un problema. Una descripción utiliza las convenciones de una representación para describir alguna cosa en particular [57]. La sintaxis de una representación especifica un conjunto de reglas que combinan símbolos para formar expresiones en el idioma de la representación. La semántica de una representación define como deben ser interpretadas las expresiones construidas, esto es, que significado puede ser derivado de la forma [35].

Un sistema para la representación del conocimiento en un dominio particular, deberá poseer las cuatro propiedades generales siguientes:

- Suficiencia Representacional: La capacidad de representar todos los tipos de conocimiento que son necesarios en cierta aplicación del dominio.
- Suficiencia Inferencial: la capacidad de manipular las estructuras representacionales en forma tal, que se deriven nuevas estructuras correspondientes al conocimiento nuevo inferido a partir del conocimiento viejo.
- Eficiencia Inferencial: la capacidad de incorporar dentro de la estructura del conocimiento, información adicional que pueda ser utilizada para enfocar la atención del mecanismo de inferencia en las direcciones más prometedoras.
- Eficiencia Adquisicional: La capacidad de adquirir nueva información fácilmente. El caso más simple implica la inserción directa, por una persona, del nuevo conocimiento en la base. Idealmente, el programa por sí mismo podría controlar la adquisición de conocimiento.

Desafortunadamente, no ha sido encontrado un sistema simple que perfeccione todas las capacidades para todos los tipos de conocimiento. Como resultado, existen múltiples técnicas para la representación del conocimiento. Muchos programas cuentan con más de una técnica.

En una representación híbrida del conocimiento, los objetos del dominio son representados utilizando combinaciones de diferentes formalismos de manera tal que se compensan las debilidades tanto de unos como de otros y se tomen las ventajas de sus virtudes para complementarse. La ventaja de una representación híbrida sobre una representación uniforme del conocimiento, es que permite una representación más natural de los diversos tipos de piezas del conocimiento característicos de una aplicación de dominio dada [58].

Las representaciones del conocimiento constan de dos tipos de entidades: los hechos o verdades sobre algún mundo pertinente, siendo éstas las cosas que nosotros queremos representar y las representaciones de hechos en algún formalismo elegido, éstas son las cosas que nosotros realmente podremos manipular. Las Reglas de Producción, las Redes Semánticas, los Frames, los Scripts, CYC, etc., son estructuras declarativas clásicas en las cuales se puede representar el conocimiento [52].

Una FRC es utilizada para describir las características esenciales de un dominio, la manera de razonar para encontrar soluciones a través de estas descripciones y los algoritmos y estructuras de datos utilizados para implementarlos.

Algunas de las técnicas más importantes para la representación del conocimiento serán descritas a continuación.

1.3.1. Reglas de producción.

Newell (1973) propuso un sistema de producción como modelo psicológico del comportamiento humano [55]. En este modelo se considera que el conocimiento humano se representa en forma de producciones o reglas de producción. Se asemeja al proceso de memoria humano: memoria a corto plazo (deducciones intermedias) y memoria a largo plazo (producciones).

Normalmente las reglas de producción se ven como un formalismo en el cual representar el conocimiento y es el formalismo más usado en los sistemas expertos.

Los expertos tienden a expresar sus técnicas de solución de problemas en forma de reglas "situación - acción".

Las reglas de producción constituyen un formalismo que se usó desde antes en teoría de autómatas, gramáticas formales y en el diseño de lenguajes de programación.

Originalmente las producciones eran reglas gramaticales para manipular cadenas de símbolos. Las reglas de producción usadas en los sistemas expertos difieren un poco de las producciones, pero los principios son los mismos.

Las Reglas de producción manipulan estructuras de símbolos, como listas o vectores (más que *strings*). Se tiene:

- Un conjunto N de nombres de objetos en el dominio
- Un conjunto P de propiedades que representan atributos de los objetos
- Un conjunto V de valores que los atributos pueden tener

Generalmente se usa una tripleta: (objeto, atributo, valor).

Propiedades de las reglas:

- Modularidad: cada regla define un pequeño y relativamente independiente pedazo de conocimiento.
- Incrementalidad: nuevas reglas pueden ser añadidas a la base de conocimiento de forma independiente a las demás.
- Modificabilidad: como consecuencia de la modularidad, las reglas viejas pueden ser modificadas.
- Transparencia: habilidad de explicar sus decisiones y soluciones.

1.3.2. Redes Semánticas.

Las Redes Semánticas constituyen un modelo que trata de simular la memoria humana para capturar la semántica de las palabras. Puede ser considerado un tipo de red en la cual los nodos representan objetos, conceptos o situaciones y los arcos representan relaciones entre ellos [57].

Realmente es una estructura de datos sofisticada y mucho depende del programa que la mantiene y la usa. Se llama red semántica porque se usaron originalmente para representar el sentido en expresiones de lenguaje natural.

Cada palabra o nodo conceptual se consideraba la cabeza de un "plano" que tiene su definición (ejemplo, si banco tiene 3 significados, entonces existen 3 planos para él). Las ligas en el plano representan su definición.

Pueden existir apuntadores a: superclases (es-un), modificaciones, disyunciones, conjunciones y sujeto/objeto. Apuntadores fuera del plano hacen referencia a otros objetos (y planos) en donde se definen. Para obtener la relación entre dos palabras se buscan las intersecciones en las redes. Esta activación de todo lo que rodea a una palabra se espera que represente la definición completa de un concepto.

Se distinguen dos tipos de ligas principales:

1. Subclase (es-un): las clases de "arriba" están definidas en términos de conceptos generales que se asumen que se cumplen en todas sus subclases.
2. Modificadores: propiedades particulares de conceptos específicos.

En una Red Semántica se puede representar la herencia de propiedades. Esto originó la definición de distancia semántica entre conceptos, en base al número de ligas a recorrer. Estas redes permiten tener valores por default y cierta expectación acerca de los posibles valores de un atributo. La idea es tratar de tener un conjunto adecuado (pequeño) de nodos y de un conjunto de arcos de salida genéricos.

La mayoría de las redes semánticas se pueden ver como un conjunto de pares atributo valor. Utilizan sólo predicados atómicos (no variables), binarios y unarios, sin símbolos funcionales y la suposición del mundo cerrado. Los predicados están organizados en taxonomías y existen tipos que restringen los valores.

Una característica interesante es que se puede usar una representación gráfica, que posibilita realizar inferencias usando técnicas de búsqueda en grafos.

Por otro lado, la representación gráfica, sugiere diferentes tipos de inferencia basadas en la estructura, por ejemplo, similitud entre conceptos, y también muy importante, la capacidad de heredar información.

1.3.3. Frames.

Los Frames son estructuras de datos utilizadas para representar situaciones prototípicas. Proporcionan una estructura conveniente para representar objetos que son comunes a una situación dada, como los estereotipos. En particular, los frames son útiles para simular conocimiento de sentido común [30].

Una de las ideas intuitivas detrás de los frames, es que la memoria humana se basa en estereotipos (propiedades típicas de los objetos).

Los sistemas de frames razonan acerca de clases de objetos usando representaciones prototípicas, pero que pueden modificarse para capturar las complejidades del mundo real, como lo pueden ser las expectativas.

Se parte de la idea de tener una sola estructura de datos para codificar el conocimiento relevante acerca de una clase de objetos, en lugar de tener el conocimiento distribuido en forma de reglas o fórmulas lógicas.

Permite construir conocimiento declarativo y procedural en un registro con slots y fillers o facets. Los slots son atributos y los fillers o facets son los valores.

Los frames pueden ser construidos utilizando cierta jerarquía en donde los frames de "abajo" pueden heredar los valores de los slots de los frames de "arriba". Normalmente la herencia se hace por medio de los arcos: es-un (al final *instance-of*).

Se pueden hacer deducciones a través de la jerarquía (se distinguen entre los frames clases o genéricos y los frames instancias).

Los frames son una extensión de las redes semánticas. La diferencia radica en la estructura de los frames, en particular sobre sus atributos, los cuales pueden tener: valores, defaults, restricciones y/o procedimientos asociados.

1.3.4. Scripts.

Un script es una forma de representar una situación prototípica (parecido a los frames), pero en lugar de tener una descripción de un objeto, el script describe una secuencia de eventos [52].

Para describir una secuencia de eventos, el script usa un conjunto de slots que tienen información acerca de gente, objetos y acciones involucrados en los eventos

Algunos de los elementos típicos incluyen:

- Condiciones (*entry conditions*): describen que se debe de satisfacer para poder aplicar el script.
- Objetos (*props*): tiene los objetos que se usan en la secuencia de eventos.

- Roles: la gente involucrada.
- Variantes (*track*): variantes que pueden ocurrir en un script particular.
- Escenas: describe la secuencia de eventos.
- Resultados: las condiciones que existen después de aplicar el script.

Los scripts pueden predecir que se espera que pase en una situación. Aunque no se mencione lo que pase, el script puede "llenar los huecos" (se utilizaron para entender historias), ya que estos se almacenan y se obtienen de memoria los que más se parezcan a la situación.

1.3.5. CYC.

CYC es una BC y máquina de inferencia multi-contextual desarrollada por Cycorp [39]. La tecnología CYC incluye los componentes esenciales siguientes:

- BC.
- Lenguaje de representación.
- Máquina de inferencia.
- Herramientas de interfaz.
- Módulos de aplicación.

La BC de CYC contiene representaciones de una gran cantidad de conocimiento humano fundamental: hechos, reglas y heurísticas para el razonamiento acerca de los objetos y eventos de la vida cotidiana.

El lenguaje de representación de CYC es esencialmente una argumentación del cálculo de predicados de primer orden, con extensiones para manejar uniformidad, razonamiento estándar, representación estandarizada y algunas características de segundo orden, por ejemplo cuantificación de predicados disponibles en algunas circunstancias o contextos.

La máquina de inferencia de CYC desarrolla deducciones lógicas generales, incluyendo las reglas de inferencia "modus ponens", "modus tolens" y cuantificaciones existenciales y universales.

1.4. Adquisición y Procesamiento del Conocimiento.

Además de la estructura propia para la representación del conocimiento es necesario un sistema de adquisición para la misma y una máquina de inferencia que sea capaz de procesarla.

La adquisición del conocimiento es el traslado y transformación de potencial de solución de problemas especializados de alguna fuente de conocimiento a un programa. Este

traslado es normalmente cumplido por una serie de entrevistas largas e intensivas entre un IC de conocimiento que normalmente es un especialista en computación y un experto del dominio que puede articular su especialización hasta cierto punto [35].

La adquisición del conocimiento es un proceso por medio del cual, se hace posible que los sistemas contengan el conocimiento necesario para desarrollar sus tareas.

Por su parte, la máquina de inferencia contiene, como la base de conocimiento, un conjunto de reglas. Sin embargo, las reglas en ésta son usadas específicamente para controlar el proceso de inferencia, el cual determina una conclusión sobre la base de conocimiento, dado un cierto conjunto de hechos [34].

La máquina de inferencia realiza el procesamiento de las estructuras de la representación del conocimiento, analizando un caso específico a la vez, realizando la comparación con el conocimiento que se ha definido en ella y proporcionando los resultados para los cuales fue diseñada.

1.5. Integración de Datos y Conocimiento.

Múltiples son los trabajos desarrollados y las investigaciones realizadas alrededor de las BDs puesto que representan una fuente muy confiable de información, en nuestro caso se describirán aquellos que están dirigidos a la utilización de ambas fuentes de información (datos y conocimientos).

Matt Ginsberg (1993), en su libro "Essentials of Artificial Intelligence", presenta un trabajo encaminado a obtener información de la BD como se realizaría con cualquier Manejador de BD. Esto es, se trata de obtener los mismos resultados que se obtendrían en una consulta tradicional a la BD, utilizando representaciones del conocimiento. Plantea la construcción de BD que incluyan cuantificadores lógicos clásicos, la extensión de Modus Ponens y las reglas de resolución para manipular cuantificadores, así como también la inclusión de cuantificadores en sentencias en Forma Normal. Su trabajo propone la creación de BD deductivas, empleando representaciones del conocimiento en BD para extraer información específica.

Tradicionalmente las BD tienen entidades pasivas. La diferencia principal entre las BD activas y las tradicionales, es la pasividad en éstas últimas, todos los procedimientos lógicos, incluyendo la recuperación y modificación de los datos son coordinados fuera del dominio del manejador de datos; si el resultado de una operación de actualización de datos en particular (un evento) sugiere la ejecución de otra sucesión de acciones, la ejecución de éstas otras acciones deberá ser inicializada por lógica desde la aplicación de algún agente externo. En contraste, un medioambiente de BD activas, soporta la iniciación y el control de éstas otras acciones dentro del ambiente de las BD según las reglas prefijadas, sin la necesidad de obtener una guía extensa de las aplicaciones o de otra fuente externa [56].

Un SBC contiene tanto conocimiento como información y datos; una metodología unificada representa los datos, la información y el conocimiento de una forma homogénea, así como también las relaciones entre ellas. Dicha metodología construye

un mecanismo de mantenimiento dentro del diseño [28]. Más que una forma de representación del conocimiento, se propone una metodología para la normalización de BCs, la cual tendría su equivalente en las formas normales para las BD, es decir, propone una nueva estructura para la implementación de BCs la cual contiene datos, información y conocimiento.

El desarrollo de Bases de Datos Inteligentes (BDI) está motivado por la necesidad de presentar grandes cantidades de información en un formato fácil de consultar y comprender. Las BDI pueden ser definidas como BD que manejan información de una manera natural, haciendo que la información sea fácil de almacenar, acceder y usar.

La meta en el desarrollo de cualquier aplicación de BDI es primero comprender los procesos que generan la información y entonces usar ésta información para controlar o explotar dichos procesos. Una BDI organiza y transmite información, pero es la aplicación la que interpreta la información dentro del contexto de una tarea significativa. Proporciona una arquitectura de tres niveles: interfaz de usuarios, herramientas de alto nivel y máquina de BD [47].

Los sistemas desarrollados bajo la disciplina de la Minería de Datos están encaminados al apoyo para la Toma de Decisiones, esta disciplina cuenta con innumerables sistemas computacionales uno de ellos es el DBMiner, el cual ha sido desarrollado para la minería interactiva de múltiples niveles de conocimiento en grandes BD relacionales y Datawarehouses. El sistema implementa un amplio espectro de funciones para la minería de datos, como son: caracterización, comparación, asociación, clasificación, predicción y clustering.

La función principal de DBMiner es la generalización de los datos. Son consideradas dos estructuras de datos: la relación generalizada y el cubo de datos multidimensional. Las asociaciones derivadas son especificadas en forma de metareglas, las cuales restringen la búsqueda a formas específicas de reglas que representan diferentes propiedades en una BD con un porcentaje en relación a la consulta realizada [36].

La integración de datos bajo un Esquema Global de Datos proporciona el acceso a un conjunto de datos heterogéneos y autónomos en un sistema [9]. Para esta tarea se contempla la combinación de datos residentes en diferentes fuentes, con la finalidad de proveer al usuario una vista unificada de los datos. Se utilizan técnicas para las consultas con tratamiento de respuestas a partir de información incompleta, en caso de requerirse por parte de estos sistemas de integración de información.

Los trabajos antes presentados exponen varias alternativas de solución. Se propone en algunos trabajos, la creación de BD que contengan representaciones del conocimiento e incluso utilicen las técnicas de inteligencia artificial para el logro de resultados. En otros, se propone la inclusión de los datos en las BCs, como por ejemplo CYC. En nuestro trabajo se propone la unificación de BD con BCs, por medio de estructuras para la representación del conocimiento, para la integración de datos y conocimiento.

Capítulo II

Estructuras de Representación del Conocimiento para la Manipulación de Bases de Datos desde Bases de Conocimiento.

2.1. Introducción.

Como se presentó en el capítulo anterior, la integración de datos y conocimiento es una forma utilizada para buscar soluciones a problemas de la vida real que requieren tanto de la experiencia expresada en el conocimiento que se tiene acerca del problema, como de los casos que se han estudiado dentro de dicho problema. En el capítulo anterior se mencionaron algunas de las soluciones dadas al proceso de unificación de estas dos fuentes de información.

En este trabajo, el esquema que se usa para la integración de conocimiento y datos, se basa en el uso de SBCs dentro de los cuales se expone una teoría fundamentada en FRCs que permiten obtener información de diverso tipo de una BD, con el objetivo de que ésta sea utilizada como fuente en una BC para la toma de decisiones.

Para ello se utilizan dos estructuras para la interrogación de una BD de dos formas diferentes: una para la extracción de valores y otra para el análisis del cumplimiento de condiciones dentro de la BD que serán definidas por parámetros denominados cuantificadores, logrando con esto la integración de BCs y BDs. El resultado de la ! interrogación en ambos esquemas será utilizado en la consulta que se está ejecutando a la BC.

Estas estructuras para la representación del conocimiento fueron adicionadas al lenguaje para la programación del conocimiento denominado HAries, con el propósito de incorporar a los SBCs, desarrollados sobre este lenguaje, datos reales del problema que se consulta.

2.2. Lenguaje para la Programación del Conocimiento. Medio Ambiente HAries.

El medio ambiente HAries puede ser utilizado para desarrollar sistemas inteligentes híbridos en cualquier dominio del conocimiento. Esto se debe a que está compuesto por diversas estructuras que permiten simular diversas formas de representar el conocimiento humano, así como de un conjunto de máquinas de inferencia diseñadas para procesar dichas estructuras. Todo ello independiente del dominio en que se pueda desarrollar una aplicación.

Este medio ambiente basa su funcionamiento en la definición de un conjunto de estructuras o formas para la representación del conocimiento, que permiten al IC codificar y procesar los conceptos y relaciones obtenidos acerca de un dominio específico.

La teoría en que se encuentra sustentado, así como su implementación en la computadora, permiten asegurar que constituye un "Lenguaje de Programación del Conocimiento", debido a que cuenta con las instrucciones básicas de cualquier lenguaje, como son: tipos de datos, variables, operadores, sentencias de ciclo, condicionales, etc., pero éstas a su vez, se encuentran basadas en estructuras para la representación del conocimiento, lo que permite incrementar sus posibilidades mediante uso de técnicas propias de los sistemas de IA.

Este concepto no es nuevo, realmente apareció alrededor del año 1985 con el sistema OPS5, donde se introdujo el concepto de "Programación Basada en Reglas" [8], como una generalización de un conjunto de sistemas de propósito general anteriores conocidos como "shells".

Posteriormente estas ideas fueron evolucionando y se comenzaron a desarrollar sistemas de programación en los cuales se mezclaban conceptos de programación tradicional con diferentes Formas para la Representación del Conocimiento, surgiendo el concepto de ambientes para la "Programación del Conocimiento" para identificar este tipo de desarrollo, que no solamente basa su funcionamiento en reglas [21], [22].

HARIES está estructurado, fundamentalmente, por un Sistema de Adquisición de Conocimiento llamado **HARIESA**, un Sistema para el Procesamiento del Conocimiento previamente adquirido denominado **HARIESC** y un conjunto de utilitarios.

En el sistema para la adquisición del conocimiento HARIESA se construye la BC. Ésta contendrá las estructuras que representarán el conocimiento existente sobre un dominio de aplicación específico. Este sistema proporciona las interfaces necesarias para que el IC seleccione las FRCs para contener el conocimiento que se desea representar.

La figura 2.1 muestra la pantalla principal del sistema HARIESA, donde el IC elige las diversas estructuras para poder representar los distintos tipos de conocimiento. Está diseñada para poder realizar múltiples operaciones con cada una de las estructuras que puede contener una BC.

Estas operaciones se refieren a la creación de una nueva BC, abrir una existente, leer desde archivo texto o binario y grupos de operaciones referentes a la edición, compilación y visualización para las distintas FRCs. Antes de leer o cargar la BC, es necesario seleccionar cuales serán las estructuras que se cargarán en memoria durante el proceso de lectura.

Al ejecutar la operación de crear nueva BC, el sistema realizará la creación de la estructura básica, constituida por un conjunto de proposiciones que definirá el IC y que se creará implícitamente, y a la que posteriormente, se le anexarán otras estructuras para completar el conocimiento que se quiere representar.

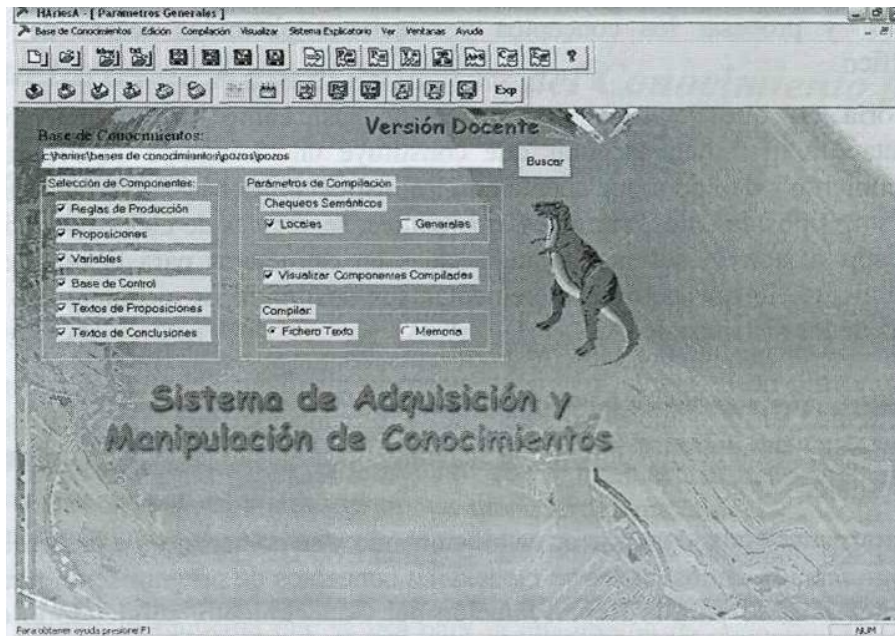


Figura 2.1 Pantalla principal del sistema HArIESA.

La opción abrir una BC existente, se ejecuta para identificar el nombre de los archivos que contienen las estructuras componentes de una BC, a la que se pretende dar mantenimiento.

La opción cargar BC, se refiere a la lectura de las estructuras de una BC ya existente de acuerdo a las estructuras elegidas en el control selección de componentes que aparece en el ambiente principal del sistema de adquisición. La lectura se puede realizar ya sea desde archivo texto o desde archivo binario, esto con la finalidad de facilitarle el trabajo al IC, en cuanto al respaldo de su BC.

El grupo de operaciones referentes a la edición son utilizadas para mostrar las estructuras componentes de una BC y es en ella en donde se proporcionan los mecanismos necesarios para su mantenimiento.

En cuanto al grupo de operaciones dedicadas a la compilación, se ejecutarán cuando se desee preparar a la BC para que pueda ser interpretada por el sistema HArIESC.

Las operaciones para la visualización proporcionan un listado de cada uno de los tipos de estructuras que conforman una BC.

Para desarrollar una aplicación, el programador, conocido como IC es el encargado de entrevistar al experto en el dominio, con el propósito de que éste identifique las estructuras necesarias para representar el conocimiento obtenido y posteriormente realizar la construcción de la BC a través del sistema HArIESA.

La adquisición del conocimiento es un proceso de extracción, estructuración y organización del conocimiento que va a ser capturado y transformado en formas entendibles por la computadora.

La figura 2.2 muestra una perspectiva general del proceso que se lleva a cabo para realizar la adquisición del conocimiento. En ella se puede observar como el IC interactúa con el experto en el dominio para comprender cómo él resuelve las diversas situaciones que se presentan en el problema y qué conocimiento utiliza para ello.



Figura 2.2 Adquisición del conocimiento.

Una vez que el IC haya entrevistado al experto en el dominio que se pretende trabajar, se deben de identificar las estructuras más adecuadas para representar el conocimiento obtenido, para posteriormente comenzar con el desarrollo de la BC en el sistema HArisesA.

Sin embargo, el conocimiento representado a través de diversas estructuras en una BC de manera aislada no tiene ningún significado si éstas no se procesan para lograr que dicho conocimiento se manifieste de la forma para lo cual fueron diseñadas.

Por tanto, se hace necesario la implementación de los mecanismos para el procesamiento del conocimiento y en el lenguaje HArises, HArisesC es el encargado de proporcionar los mecanismos necesarios para procesar las estructuras contenidas en una BC desarrollada en el sistema HArisesA.

La pantalla principal del sistema HArisesC, mostrada en la figura 2.3, es utilizada para proporcionar una interacción visual entre el sistema y el usuario. En ella se especifica el nombre de la BC que se desea ejecutar y una vez realizada la carga de los archivos correspondientes, en este caso los archivos ejecutables, se procede con la consulta del caso que se quiera evaluar.

HArisesC es utilizado para la puesta a punto del sistema y para la consulta de un caso particular que se desee evaluar a partir del conocimiento contenido en la BC, otorgando al IC todas las facilidades requeridas para llevar a cabo ambas tareas. Por ejemplo, se pueden mostrar e inicializar los valores de los parámetros y los valores de cada estructura contenida en la BC, para verificar que el sistema otorgue los resultados que se requieren para un caso en particular.

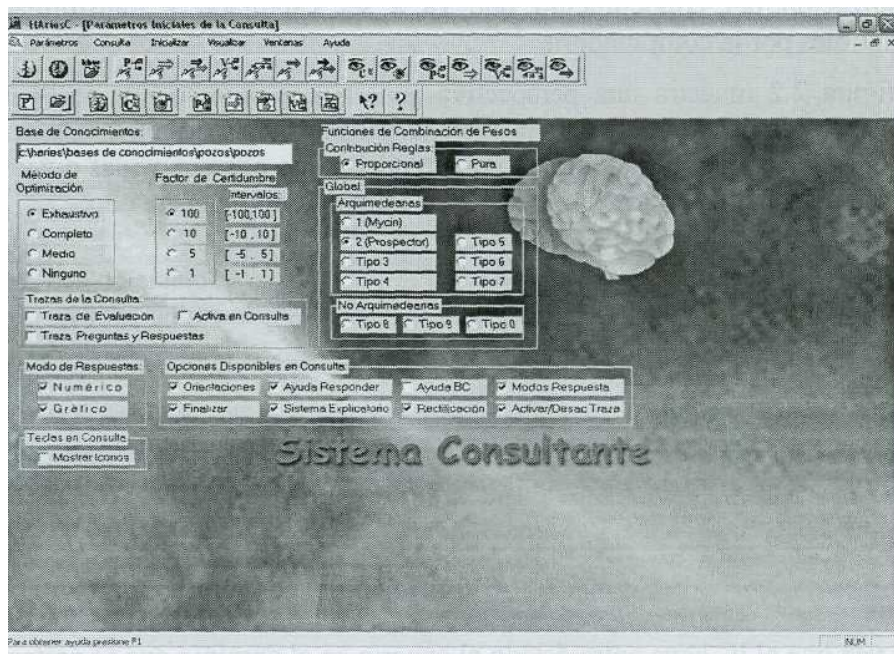


Figura 2.3 Pantalla principal del sistema HArIESC.

Como se muestra en la figura 2.4, el IC ejecuta la BC en el sistema HArIESC, verificando los resultados que proporciona, para después, en caso de ser necesario, modificar las estructuras de la BC en el sistema HArIESA y corregir los posibles errores que se tengan dentro de ella, realizando este procedimiento hasta lograr la puesta a punto del sistema.

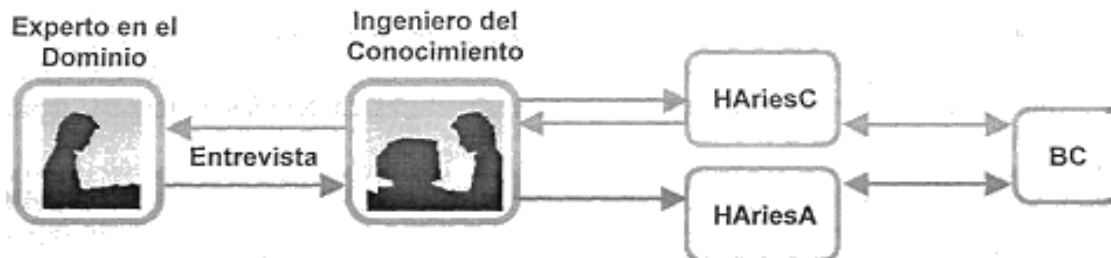


Figura 2.4 Puesta a punto del conocimiento.

Una vez que se ha logrado la puesta a punto del conocimiento, el usuario será el encargado de realizar la consulta a la BC para evaluar un caso en particular, como se muestra en la figura 2.5.

Al ejecutar el procesamiento de la BC, dentro del medio ambiente HArIESC, se realiza la evaluación de las estructuras que contienen el conocimiento. La evaluación de las estructuras consiste en el procesamiento, por parte de la máquina de inferencia correspondiente, por medio del cual se otorgarán resultados obtenidos a partir de la evaluación de un caso en particular. Es decir, el usuario ingresa al sistema los valores del

caso que se desea evaluar y en base a ellos el sistema proporcionará las soluciones basadas en el conocimiento previamente adquirido, finalizando así una consulta a determinada BC.



Figura 2.5 Consulta del conocimiento.

2.3. Manipulación de BDs desde HAries.

El medio ambiente HAries proporciona una variedad de estructuras para representar el conocimiento de diversas formas o tipos. Sin embargo, éste no contaba con estructuras capaces de consultar fuentes de datos, de donde se puede obtener una gran cantidad de información.

Para resolver esta limitante, se planteó la implementación de nuevas estructuras de representación del conocimiento en dicho medio ambiente, las cuales son necesarias para obtener información que requiere una BC a partir de la consulta a una BD.

A partir de la definición de estas estructuras se podrá interrogar la BD que se desee examinar, obteniéndose información que será manipulada por otras estructuras contenidas en la BC. Ya sea con la asignación de valores obtenidos de la BD ó en su caso, con la asignación del resultado obtenido de la cuantificación de los registros que cumplen con un conjunto características. Con esto se logra contar con herramientas que permiten construir sistemas para la ayuda en la toma de decisiones basada tanto en las experiencias como en los datos.

La perspectiva general para la consulta de un sistema inteligente en HAries con estas nuevas estructuras se amplía. La figura 2.6 muestra el proceso de consulta de un caso particular a un sistema inteligente con las nuevas posibilidades de consulta a BDs.

2.4. Estructuras para la Representación del Conocimiento de HAries que se Utilizan en la Propuesta.

En el proceso que se lleva a cabo para la construcción de una aplicación dentro del medio ambiente HAries, se construye una BC apoyándose de las estructuras para la representación del conocimiento que proporciona dicho medio ambiente.

Las estructuras que se implementan en este trabajo hacen uso de algunas de las estructuras para la representación del conocimiento del medio ambiente HAríes. Ello, con la finalidad de apoyar el proceso de integración de la información proveniente de una BD en la BC, las cuales se explicarán a continuación.

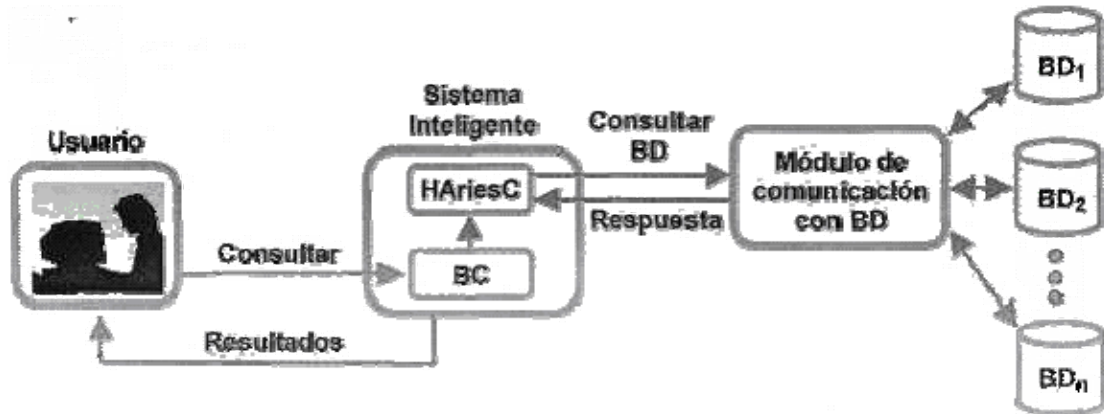


Figura 2.6 Perspectiva general para la consulta de un sistema inteligente en HAríes.

2.4.1. Estructura Proposición.

Las proposiciones representan los conceptos básicos de una BC, a partir de los cuales se construyen las relaciones provenientes de un dominio en específico.

Aunque este tipo de concepto, en realidad representa una estructura con varias componentes, cada una de las cuales contiene información específica, solamente necesitaremos un análisis general de su significado. Las definiciones formales pueden ser consultadas en [27].

Una proposición siempre se encuentra asociada a un valor de verdad que expresa si se cumple o no la situación planteada.

Este valor de seguridad será considerado como un valor numérico del intervalo $[-1,1]$, donde 1 indica verdad absoluta (verdadero), -1 falsedad absoluta (falso) y 0 desconocimiento total (no se sabe), mientras que el resto de los valores representan gradaciones asociadas con la creencia acerca del concepto que representa. Así por ejemplo, 0.9 puede indicar que existe mucha seguridad para establecer el cumplimiento de la proposición, -0.3 por el contrario que la situación no se cumple pero esto se puede afirmar con poca seguridad o certidumbre.

Desde el punto de vista práctico, el usuario puede introducir un factor de certidumbre (FC), para cambiar la forma de presentación y obtención de grados de certidumbre que se manipulan en el sistema, el FC puede encontrarse dentro de distintos intervalos, los cuales pueden ser:

- 100: representa el intervalo $[-100,100]$.

- 10: representa el intervalo [-10, 10].
- 5: representa el intervalo [-5, 5].
- 1: representa el intervalo [-1, 1].

Es importante mencionar que la introducción del FC es sólo de carácter práctico (presentación y obtención de grados de certidumbre), ya que el medio ambiente HAries realiza la manipulación interna de la certidumbre utilizando el intervalo [-1, 1].

2.4.2. Estructura Variable.

Existen diversos tipos de variables en el lenguaje HAries [27], cada una con características que la diferencian de las otras, de aquí que se necesite un estudio particular de cada caso. Es necesario señalar que no se realizará un análisis exhaustivo de todos los tipos disponibles.

Para representar los aspectos necesarios que se relacionan con el presente trabajo, se discutirán los aspectos generales vinculados con este concepto y se hará hincapié en los nuevos tipos de valores que se implementan.

La utilidad y objetivos de este tipo de conceptos dependen de cada tipo particular, pero en general, se pueden resumir como sigue:

- Constituyen una vía utilizada para definir conceptos.
- Permiten la definición de los tipos de datos admitidos por el lenguaje, que son: enteros, reales de punto flotante, cualitativos, certidumbre y tiras de caracteres.
- Se utilizan para definir arreglos de valores.
- Posibilitan la definición de preguntas.
- Se utilizan para establecer relaciones entre los tipos de valores admitidos y las proposiciones asociadas.
- Posibilitan la introducción de expresiones aritmética.
- Permiten definir expresiones condicionales.
- Constituyen una vía que se utiliza para incorporar comunicación con Bases de Datos y con Hipermedias internas que maneja HAries.

La definición general de una variable puede ser expresada como se explica a continuación.

Se llama "**Estructura Variable**", V_j , a una estructura que tiene las siguientes componentes: $V_j = \langle T_j, V_{r_j}, Pr_j, Atr_j \rangle$ donde:

- T_j Representa un conjunto de textos,
- V_{r_j} Valores de la variable,
- Pr_j Proposiciones asociadas y
- Atr_j Atributos

Los textos se utilizan para identificar el concepto, efectuar preguntas o brindarle al usuario explicaciones.

Las variables almacenan sus posibles valores en la componente **V_{rj}** de la estructura. Pueden ser únicos, pasando por un intervalo, o arreglo de intervalos, hasta una tira de caracteres, lo cual va a depender del tipo de variable, en algunos casos, o de la situación particular a resolver en otros. En este último caso, la definición de los tipos de valores es dinámica, puesto que se define en tiempo de ejecución. La cantidad y tipos de valores que admite una variable permiten establecer una clasificación de éstas que se utiliza para ingresarlas durante el desarrollo de una aplicación.

Regularmente, las variables definen un esquema de cálculo que posibilita obtener los valores de seguridad de las proposiciones asociadas, de ahí la relación que se establece en la componente **P_{rj}**, entre ambos conceptos.

Las proposiciones asociadas son también una vía que se utiliza para activar el funcionamiento de las variables, es decir, si existe una relación entre una proposición y una variable, cuando se necesita el valor de la primera, esto provocará la activación de la segunda, cuyo proceso de evaluación brindará entre sus resultados los valores de certidumbre asociados a las proposiciones.

Finalmente, los atributos, constituyen en sí otra estructura cuyas componentes dependen directamente del tipo de variable. En general se pueden citar los siguientes: esquemas de cálculo, los tipos de respuestas y rangos de valores admitidos, los elementos multimedia que pueden tener asociados (imágenes, sonidos, videos, etc.), las condicionales, expresiones aritméticas, conexiones con bases de datos, etc.

El proceso de análisis de una variable, que se denomina "Proceso Evaluativo", se identifica como los pasos que se necesitan ejecutar para obtener los valores de la variable, la ejecución de las acciones asociadas y los pasos para obtener los valores de certidumbre de las proposiciones asociadas.

Estos pasos no son fijos ni son ejecutados siempre de la misma forma, por ejemplos, en algunas variables los valores son asignados directamente y por tanto no se necesita un mecanismo para su obtención y en otros casos pueden no existir acciones asociadas.

2.5. Estructuras para la Manipulación de BD.

Para construir una aplicación que pueda procesar información adquirida de una BD se han desarrollado nuevas FRCs dentro del medio ambiente HAries, que posibilitan la operación con BD.

Estas estructuras, son del tipo variable pues a ellas se les asigna un valor, producto de la interrogación a una BD. De acuerdo a su naturaleza se clasifican como **Variable Base de Datos** y según las operaciones que realizan se clasifican en **Extracción de Datos** y **Cuantificador**, como se puede ver en la figura 2.7. La primera, como su nombre lo indica, fue diseñada para extraer valores de la BD y la segunda para cuantificar la cantidad de registros que cumplen una condición dada.

Dichas estructuras se definen para establecer la información general que se necesita extraer de la BD y definir las relaciones entre estos datos y las componentes de la BC, que finalmente es la que contiene el conocimiento para utilizar la información obtenida como apoyo para resolver el o los problemas planteados.

En los siguientes apartados se detallará la definición de las variables extracción de datos y cuantificador.



Figura 2.7 Formas para la representación del conocimiento en HARIES.

2.5.1. Estructura Variable Extracción de Datos.

La estructura variable extracción de datos se define con la finalidad de brindar vías para la comunicación entre un BC y una o varias BDs, y permitir la extracción de valores de las BDs para utilizarlos en la BC.

Las estructuras de la BC que pueden ser relacionadas con los datos de una BD son proposiciones, variables numéricas y variables cualitativas.

Se llama Variable Extracción de Datos (VED) a una estructura que define la relación entre la BC y una BD con el objetivo de extraer datos de esta última hacia la primera y tiene las siguientes componentes:

$V_{EDj} = \langle T_{EDj}, Vr_{EDj}, Pr_{EDj}, Atr_{EDj} = \langle CBD_{EDj}, PL_{EDj}, R_{EDj} \rangle \rangle$ donde $T_{EDj}, Vr_{EDj}, Pr_{EDj}$ se interpretan como en la estructura general de las variables, Atr_{EDj} identifica los atributos específicos como sigue:

- CBD_{EDj} define las características de la conexión con una BD,
- PL_{EDj} representa un conjunto de parámetros de localización y
- R_{EDj} constituye un conjunto de relaciones entre la BD y la BC.

Es necesario aclarar que en el caso de la **VED** el conjunto **Pr_{EDi}** de proposiciones asociadas está compuesto por un solo elemento, que se denomina "Proposición Central".

Ésta se utiliza como una posible vía para la activación del proceso evaluativo de la variable y para almacenar la validez del proceso ejecutado de la siguiente forma: si la comunicación con la BD resulta satisfactoria y además los datos son extraídos correctamente, la proposición quedará con un valor de certidumbre positivo asignado, pero si alguno de estos dos aspectos falla la certidumbre asignada será 0, para indicar que el proceso falló.

Lo anterior permite, que el proceso activador de la variable cuente con información para tomar decisiones en caso de aparecer algún error y evitar así posibles interrupciones imprevistas en la ejecución del sistema.

La conexión con la BD (**CBDEDJ**), establece los parámetros que determinan como se hará la conexión con la BD utilizando el "Nombre" que identifica a la BD, el nombre del "Usuario" y la "Clave" de acceso.

Se necesita mencionar que la comunicación implementada en esta variable, utiliza ODBC como vía para establecer comunicación con una BD, por tanto, se necesita que en la computadora donde se ejecute la aplicación se encuentren instalados los "drivers" correspondientes.

Por otra parte, se puede configurar la conexión previamente utilizando las opciones de \ administración de orígenes de datos del Windows, si se trata de una BD fija o esto puede ser hecho en tiempo de ejecución de la aplicación si no se conocen los parámetros de conexión previamente (sobretudo el nombre que identifica a la BD).

Los parámetros de localización (**PLEDJ**) tienen que ver con la "Tabla", el "Campo" guía y "Registro" de donde se extraerán los datos. Esta información puede ser conocida a priori, pero también puede ser parcial o completamente desconocida, en cuyos casos durante la ejecución el usuario deberá completar la información.

Finalmente, en el conjunto **REDJ** de la variable, se encuentran las relaciones entre la BD y la BC. Cada relación se compone por un elemento de la BC, que puede ser una variable numérica, cualitativa o una proposición y un campo de la tabla indicada en **PLEDJ**.

De inicio, es responsabilidad del desarrollador que las relaciones sean establecidas correctamente, en el siguiente sentido: si se considera una variable numérica de la BC, por ejemplo, la relación deberá ser formulada con un campo donde se almacenen números y de la misma forma con el resto.

Sin embargo, el sistema en tiempo de ejecución realiza ciertos chequeos de concordancia entre los tipos de datos, que ayudan al buen funcionamiento del proceso y permiten informar al usuario sobre posibles incongruencias.

En el caso de las proposiciones regularmente se establece relación con campos de tipo "Boolean" (verdadero o falso) aunque pudiera existir un campo que defina valores de certidumbre, como admiten las proposiciones en **HAries**.

Para lograr una mejor comprensión de la definición anterior, se presentan un ejemplo de su uso.

2.5.1.1. Ejemplo.

Se tiene la información de pacientes en una BD y se desean obtener algunos datos, a partir de la consulta a la tabla mostrada en la figura 2.8, para manipularse en la BC con la finalidad de evaluar la atención que se le ha brindado a la paciente.

Con este propósito, se crea una variable extracción de datos con la definición de los parámetros correspondientes. En este caso no se requiere de una proposición asociada, es decir $Prj = 0$, puesto que solo se requieren obtener los valores de la BD que necesita la BC para realizar su procesamiento.



Paciente	Positivo	Edad	Etapa
07VEOL19580302	0	42	A
09CAPA19740318	1	26	C
09MIOP19700201	1	30	A
09PESC19600206	0	20	A
09RICR19720318	0	28	A
09RORA19611002	0	39	A
09ZUCA19620710	1	38	B
12CASA19611030	1	39	C

Figura 2.8 Tabla a consultar.

Para realizar la conexión con la tabla (CBDEDJ), se especifican los siguientes parámetros:

- Nombre de la base de datos: "epicacu".
- Usuario: "FOAM".
- Clave: "Cacu2000".

La BC requiere valores que contiene el registro de una paciente dada para evaluar su caso en particular. Estos especifican el resultado de su examen citológico, la edad y la etapa de atención en la que se encuentra, la localización de ellos se especifica en **PLEDJ**, esto es: Tabla = "Pacientes", Campo Identificador = "Paciente" e Identificador de Registro = "Variable", puesto que se requiere obtener los datos de distintas pacientes y es con éste último atributo con el que se controlará la consulta de un caso en específico.

El resultado del examen citológico esta almacenado en la BD como "0" para indicar un resultado negativo y "1" para indicar que el resultado es positivo. Esta información se tratará en la BC a través de una proposición.

El valor correspondiente a la edad de la paciente se manipulará en la BC como una estructura del tipo variable numérica. La etapa de atención en la que se encuentra la

paciente esta dividida en A, B y C, las cuales indican la evolución de la atención que ha recibido determinada paciente, para manejar estos valores se tendrá que crear una variable cualitativa con tres categorías una para cada etapa.

El conjunto de relaciones BC-BD (**REDJ**), con el formato *<Estructura de la BC>* *<Número>* - *<Columna de la BC>*, se detallan a continuación:

- *<Proposición>* *<1>* - *<Positivo>*, suponiendo que la Proposición 1 es la encargada de recibir un valor de certidumbre correspondiente al resultado almacenado en la BD que indica si la paciente tiene o no un resultado positivo.
- *<Variable Numérica>* *<1>* - *<Edad>*, suponiendo que Variable Numérica 1 es la estructura de la BC encargada de contener la edad de la paciente.
- *<Variable Cualitativa>* *<2>* - *<Etapa>*, suponiendo que en la Variable Cualitativa 2 se pueden manipular las categorías de A, B ó C, que son las etapas de atención en las que se pueden encontrar la pacientes registradas en la BD.

Con la definición de ésta variable se proporciona a una BC los datos de una paciente determinada, a través de la consulta a una BD, para evaluar la atención que ha recibido, a partir del conocimiento contenido en una BC desarrollada en HArIES.

2.5.2. Estructura Variable Cuantificador.

La definición de la variable cuantificador, tiene el propósito de proporcionar una FRC que sea capaz de interrogar una BD, para obtener la cuantificación de los registros que cumplen con una serie de características y, en base a un conjunto de parámetros, verificar el cumplimiento de las propiedades que se requieren que cumpla la BD explorada. Como resultado de este procesamiento, se asignará un valor de certidumbre a una estructura del tipo proposición de la BC. Dicha cuantificación está sustentada en la definición de cuantificadores.

Estos cuantificadores están inspirados en los que se utilizan en el cálculo de predicados. Sin embargo, en ellos interviene un conjunto de parámetros que acompañan la definición.

2.5.2.1. Cuantificadores en el Cálculo de Predicados.

En el cálculo de predicados, la semántica de los cuantificadores indican la frecuencia con la cual es verdadera una fórmula bien formada (FBF) en un dominio dado. Específicamente, se utiliza el cuantificador universal para indicar que una FBF siempre es verdadera, mientras que el cuantificador existencial indica que una FBF es verdadera en al menos un elemento del dominio. Estos se definen de la siguiente manera:

Sea A una FBF, y sea x una variable. Si se desea indicar que A es verdadera para todos los posibles valores de x, se escribe $\forall xA$, y si se desea indicar que A es verdadero para al menos un valor de x, se escribe $\exists xA$.

Aquí, \forall se denomina cuantificador universal y \exists cuantificador existencial, A se denomina ámbito (alcance) del cuantificador y se dice que la variable x está ligada por el cuantificador. El símbolo \forall se lee "para todo" y \exists "existe".

Ejemplos:

1. La frase: "todos los hombres son mortales", se puede expresar en el cálculo de predicados como sigue. Si se define el predicado M de forma tal que signifique "ser mortal" entonces $M(x)$ significa que x es mortal. Por otra parte, la palabra "todos" de la frase indica que esto es cierto para todos los x por tanto se tendría: $\forall x M(x)$.
2. Sea P la propiedad "son maestros". Entonces $\exists x P(x)$ se puede traducir como: "algunas personas son maestros" o "existen personas que son maestros".

En términos generales, una expresión cuantificada divide el universo de valores de la variable en dos conjuntos, uno formado por aquellos elementos que satisfacen la sentencia y otro por aquellos que no la satisfacen.

Desde el punto de vista práctico, estas definiciones tienen un uso limitado, puesto que los resultados que brindan son extremos, es decir, sólo indican si se cumple o no la situación planteada.

Si se piensa por ejemplo, en una BD de pacientes con una determinada enfermedad y se aplica cualquiera de estos cuantificadores a los datos, sólo podrán obtenerse propiedades que cumplan todos los individuos o evaluar si existe una propiedad determinada, lo que pudiera ser interesante, pero no brinda mucha información.

A continuación se presenta la definición de la generalización de los cuantificadores que permiten ampliar estas posibilidades, orientando su aplicación hacia la consulta de BD.

2.5.2.2. Cuantificadores Parametrizados.

En el modelo que se utiliza para el desarrollo del presente trabajo, se plantea una generalización de la definición de los cuantificadores \forall y \exists para explorar el cumplimiento de una expresión en una BD, brindando como resultado valores de verdad que no necesariamente tienen que ser verdadero o falso y que son denominados Cuantificadores Parametrizados (**CP**).

Se puede plantear que un **CP** constituye una FRC en la que se pueden expresar consultas a una BD, donde el tipo y características de las mismas se definen mediante los parámetros asociados, y las respuestas indican el valor de seguridad o certidumbre que se tiene sobre el cumplimiento de la condición de consulta efectuada.

El proceso que se lleva a cabo por medio de este modelo se ilustra en la figura 2.9. En ella se puede observar que los **CP** se definen como parte de una BC y que el proceso general, consiste en ejecutar preguntas a una BD para actuar en correspondencia con las respuestas recibidas producto de la consulta.

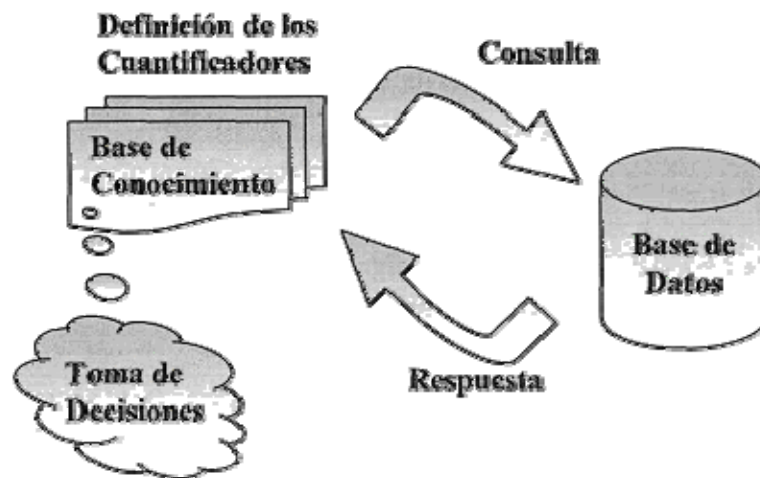


Figura 2.9 Funcionamiento de los cuantificadores parametrizados.

2.5.2.2.1. Cuantificador Existencial Parametrizado.

Con el Cuantificador Existencial Parametrizado (CEP) se pueden representar los cuantificadores universal y existencial clásicos, simplemente asignándole parámetros apropiados, como se presentará posteriormente. Esto permite diversificar el cumplimiento de una amplia gama de posibilidades en casos reales, pero también, introduce la posibilidad de regresar valores que no necesariamente tienen que ser verdadero o falso. Se llama Cuantificador Existencial Parametrizado a una expresión de la forma:

$\exists^{CK} x \varphi(x)$, donde:

$\varphi(x)$ representa una fórmula o sentencia, cuyas componentes son campos de una BD y cuyos operadores admitidos son: "&" conjunción, "V" disyunción y "|" disyunción excluyente. Los parámetros **C** y **K** establecen las condiciones que deben cumplir los registros de la BD para que la expresión se cumpla.

Los parámetros **C** y **K** del CEP permiten establecer la cantidad de registros que deben cumplir $\varphi(x)$, para que se cumpla la condición general del cuantificador.

$\exists^{CK} x \varphi(x)$ se lee: existe una cantidad de objetos o registros de la BD, determinados por **C** y **K**, que cumplen la relación definida por $\varphi(x)$. A continuación se analizan detalladamente cada una de las componentes de la definición del CEP.

El parámetro **K** define un umbral que indica la cantidad de registros que deben cumplir la relación definida en $\varphi(x)$ para el cumplimiento del cuantificador. Este puede ser expresado como un número o un porcentaje de la siguiente forma:

$K \in \{n, p\}$ donde **n** es un número entero positivo o cero y **p** un valor porcentual.

El parámetro **C** expresa el tipo de comparación que se establece en relación con el umbral establecido en **K**. Este se define como un elemento del conjunto $\{<, >, =, <,>, \neq\}$.

A continuación se muestran algunos ejemplos de utilización de los parámetros **C** y **K** del **CEP**:

1. Considerando **C** = "=" y **K** = 100%, se obtiene una expresión que es equivalente al cuantificador universal clásico, puesto que $\exists^{=100\%}\varphi(\mathbf{x})$ se cumplirá si y solo si todos los registros de la BD satisfacen $\varphi(\mathbf{x})$.
2. Para **C** = ">" y **K** = 0 se expresa, una situación equivalente al cuantificador existencial clásico, debido a que $\exists^{>0}\mathbf{x} \varphi(\mathbf{x})$ se cumple si y solo si al menos al menos un registro cumple con $\varphi(\mathbf{x})$.

Nota: Como se puede observar, los cuantificadores existencial y universal clásicos pueden considerarse como casos particulares del **CEP**.

3. Si se requiere representar lo siguiente: "Como mínimo 60 objetos satisfacen una sentencia $\varphi(\mathbf{x})$ "; se debe tomar **C** como ">", **K** = 60 y formar la expresión:

$$\exists^{\geq 60}(\mathbf{x}\varphi).$$

4. "La situación $\varphi(\mathbf{x})$ es muy frecuente", se podría expresar como: $\exists^{>70}\mathbf{x} \varphi(\mathbf{x})$, siempre y cuando se defina el concepto de frecuente como que existe un porcentaje mayor al 70% de los objetos.

$\varphi(\mathbf{x})$ representa una sentencia que expresa determinados requisitos que deben cumplirse entre las variables (campos o atributos) de un registro determinado de la BD.

Una fórmula $\varphi(\mathbf{x})$ es una expresión de la forma:

$$\varphi(\mathbf{x}) = \varphi_1 (\& \vee |) \varphi_2 (\& \vee |) \varphi_3 (\& \vee |) \dots (\& \vee |) \varphi_r \text{ donde:}$$

$\varphi_i = C_{1i} \mathcal{R} C_{2i}$ ($i = 1, \dots, r$ número de componentes de la relación), $C_{1i} \neq C_{2i}$, $C_{1i}, C_{2i} \in \{V_j, A_j\}$, V_j es un campo j de la BD y A_j una constante relacionada con el valor del campo. Ambos, V_j y A_j , no pueden ser constantes.

$\mathcal{R} \in \{<, >, =, <,>, \neq\}$. Operadores relacionales.

$\&, \vee, |$ representan los conectivos conjunción, disyunción inclusiva y disyunción excluyente, que definen si se deben cumplir respectivamente todas, al menos una o exactamente una, entre las componentes de la fórmula $\varphi(\mathbf{x})$.

A continuación se presentan algunos ejemplos de formulas $\varphi(\mathbf{x})$:

1. $\varphi(\mathbf{x}) = \text{Temperatura} < 25^\circ\text{C}$. En este caso la fórmula esta constituida por una sola componente ($r = 1$) donde; $C_{11} = \text{"Temperatura"}$ representa un campo de la BD, $\mathcal{R} = "<"$ la relación "menor que" y $C_{21} = 25^\circ\text{C}$ es un valor constante numérico que es válido para el campo temperatura.

2. $\varphi(\mathbf{x}) = (\text{crisis1} \geq \text{crisis2})$. En este caso se plantea una sentencia constituida también por una sola componente, pero entre dos campos de la BD, crisis1 es la cantidad de crisis epilépticas registradas para un paciente en el 1er. año de enfermedad y crisis2 la cantidad de crisis epilépticas para el mismo paciente en el 2do. año de enfermedad.
3. $\varphi(\mathbf{x}) = (\text{Color_de_pelo} = \text{negro}) \vee (\text{Color_de_Pelo} = \text{amarillo})$. Aquí se observa una expresión formada por dos componentes ($r = 2$) donde se necesita el cumplimiento de al menos una, para el cumplimiento de la expresión.
4. $\varphi(\mathbf{x}) = (\text{Temperatura} > 38^\circ\text{C}) \& (\text{Edad} < 5 \text{ años}) \& (\text{Hemoglobina} < 10)$. Este caso define una expresión conjuntiva, lo cual obliga al cumplimiento de las tres componentes.
5. $\varphi(\mathbf{x}) = (\text{Extracción} > 2000 \text{ pesos}) \& (\text{Saldo} < 10000 \text{ pesos})$. Aquí se define una expresión conjuntiva con dos componentes, utilizando dos campos de la BD que se comparan con constantes.

2.5.2.2.2. Cuantificador Universal Parametrizado.

Aunque el CEP contiene como casos particulares los cuantificadores existencial y universal y posibilita representar otros tipos de situaciones, todavía se presentan casos que éste no puede abarcar como, por ejemplo, cuando se necesita restringir el espacio de búsqueda dentro de la BD a un subconjunto de registros que cumplan una determinada propiedad.

Para posibilitar la representación de situaciones como ésta, se propone un nuevo cuantificador denominado Cuantificador Universal Parametrizado (CUP).

El CUP posee algunas características semejantes al CEP, éstas son el manejo de inseguridad en el cumplimiento de la expresión y la posibilidad del uso de cantidades específicas de registros para su análisis, pero en este caso se incluye entre sus parámetros otra fórmula que determinará el subconjunto de elementos a considerar dentro del universo especificado. Su definición es la siguiente.

Se llama Cuantificador Universal Parametrizado a una expresión de la forma:

$\forall^{\text{CK}} \mathbf{x} \varphi(\mathbf{x}) : \beta(\mathbf{x})$, donde:

$\varphi(\mathbf{x})$ y $\beta(\mathbf{x})$ representan fórmulas entre variables o campos de una BD, semejantes a la explicada para el CEP. Los parámetros **C** y **K** establecen las cantidades de registros de la BD que deben cumplir $\beta(\mathbf{x})$ dada la restricción $\varphi(\mathbf{x})$, para que se cumpla la expresión.

$\forall^{\text{CK}} \mathbf{x} \varphi(\mathbf{x}) : \beta(\mathbf{x})$ se lee: De los objetos que cumplen $\varphi(\mathbf{x})$ una cantidad definida por **C** y **K** también cumplen $\beta(\mathbf{x})$.

Como se comentaba, las sentencias $\varphi(\mathbf{x})$ y $\beta(\mathbf{x})$, que componen el CUP constituyen expresiones similares a la presentada en el epígrafe anterior, al igual que los parámetros

C y **K** cuyo significado es idéntico al explicado para el **CEP**. Veamos algunos ejemplos para comprender la definición.

Ejemplos:

1. Para representar la siguiente situación: "de los objetos que cumplen una relación $\varphi(\mathbf{x})$, la mayoría tiene la propiedad $\beta(\mathbf{x})$ " se puede utilizar el **CUP** como sigue, $\forall^{>50\%} \mathbf{x} \varphi(\mathbf{x}) : \beta(\mathbf{x})$.
2. La expresión $\forall^{<5} \mathbf{x} \varphi(\mathbf{x}) : \beta(\mathbf{x})$ puede ser interpretada como: "entre los objetos que cumplen $\varphi(\mathbf{x})$, menos de 5 cumplen la propiedad $\beta(\mathbf{x})$ ".
3. "La mayoría de los objetos que cumplen $\varphi(\mathbf{x})$ no cumplen $\beta(\mathbf{x})$ " se puede expresar como: $\forall^{<50\%} \mathbf{x} \varphi(\mathbf{x}) : \beta(\mathbf{x})$.
4. Para representar el siguiente caso: "La mayoría de los clientes con salario por encima o igual a 10,000.00 pesos, ejecutan extracciones menor o igual a 5,000.00 pesos" podríamos construir un **CUP** de la siguiente forma: $\forall^{>50\%} \mathbf{x} \varphi(\mathbf{x}) : \beta(\mathbf{x})$, donde **C** = ">", **K**=50%, $\varphi(\mathbf{x})$ = Salario_Cliente >10,000.00 y $\beta(\mathbf{x})$ = Extracción \leq 5,000.00 donde "Salario_Cliente" y "Extracción" representan campos de la BD y la mayoría es considerada como más del 50% de los casos.

En resumen, se puede observar que la diferencia fundamental entre los dos cuantificadores radica en que el **CEP** trabaja toda la BD mientras que el **CUP** define una restricción que delimita el análisis sobre un subconjunto de registros.

2.5.2.3. Relación con la Teoría de Tipos.

Los tipos son colecciones de elementos, probablemente vacíos. Un elemento es un dato. Cuando un tipo es definido se especifica la estructura de los elementos. La definición de un tipo nos indica como construir elementos y como diferenciarlos. La forma básica para hacerlo es proveer un patrón de construcción [14].

En el presente trabajo, el **CEP** y el **CUP** pueden ser considerados como tipos, ya que en ambos definen un conjunto de características, mediante sus parámetros asociados. En cada uno de los casos se pueden expresar consultas, en las cuales las colecciones de elementos estarán dadas por los registros de una base de datos que cumplan con el patrón definido en los cuantificadores.

En el Apéndice A se amplía el tema de teoría de tipos.

2.5.2.4. Tratamiento de Incertidumbre en los Cuantificadores.

De las definiciones de los cuantificadores parametrizados, se puede apreciar que la respuesta obtenida de la evaluación de las expresiones representadas es un valor de verdad que puede ser solamente verdadero o falso. Sin embargo, es claro que en los problemas reales muchas veces no es posible hablar del cumplimiento absoluto de una

situación y, por lo tanto, esto representa una limitante para la representación de situaciones reales utilizando los cuantificadores de la forma que han sido presentados hasta aquí.

Para vencer esta limitante, se introduce el cálculo de incertidumbre como parte del proceso de evaluación de los cuantificadores, de forma tal que puedan expresar situaciones como las siguientes:

- Existe bastante seguridad sobre el cumplimiento de una expresión dada.
- Se tiene poca seguridad sobre el cumplimiento de una situación.
- No se sabe si se cumple, o no, una relación determinada.

Para permitir la representación de situaciones con estas características, se considera que el valor de verdad asociado con un cuantificador puede ser un valor de certidumbre ó peso, representado como un valor numérico del intervalo $[-1,1]$, como se explicó en el epígrafe 2.4.1.

Por tanto, un cuantificador existencial o universal parametrizado será considerado como inexacto, si éste puede retornar un valor de certidumbre perteneciente al intervalo $(-1,1)$ como resultado de su proceso evaluativo, además de los valores extremos verdadero (1) y falso (-1).

Esta definición conlleva a establecer un esquema de cálculo diferente que el establecido en las definiciones originales de **CEP** y **CUP**, el cual estará condicionado por el tipo de comparación definida a través del parámetro **C**, como se presenta en los siguientes casos.

Caso 1: Si $C \in \{\geq, >\}$ entonces **K** se descompone en otros tres parámetros **K₁**, **K₂**, **K₃**, de forma tal que, **K₁** representa el corte entre certidumbre positiva y negativa, **K₂** el valor por encima del cual la expresión se cumple con certidumbre absolutamente segura (verdadero) y **K₃** el valor por debajo del cual se cumple con certidumbre absolutamente falsa (ver figura 2.10, Caso 1).

Caso 2: Si $C \in \{\leq, <\}$ **K** también se descompone en los tres parámetros **K₁**, **K₂**, **K₃**, donde **K₁** mantiene la misma definición anterior, pero en este caso, **K₂** será el valor por debajo del cual la expresión se cumple con completa seguridad y **K₃** el valor por encima del cual la expresión no se cumple (ver figura 2.10, Caso 2).

Caso 3: Si $C \in \{=\}$ entonces **K** se descompone en tres parámetros **K₁**, **K₂** y **F**, donde **[K₁, K₂]** representa el intervalo de valores donde la certidumbre es positiva y **F** $\in [0,1]$ define la fracción del intervalo **[K₁, K₂]** con peso positivo extremo, es decir, 1 (ver figura 2.10, Caso 3). Por ejemplo si $F = 0.5$, entonces, la mitad del intervalo tendrá un peso positivo extremo (1).

Caso 4: Si $C \in \{\neq\}$, **K** se descompone de la misma manera que en el caso 3 pero el intervalo **[K₁, K₂]** representa los valores donde la certidumbre es negativa y **F** por tanto,

la fracción de este intervalo con peso negativo extremo, es decir, -1 (ver figura 2.10, Caso 4).

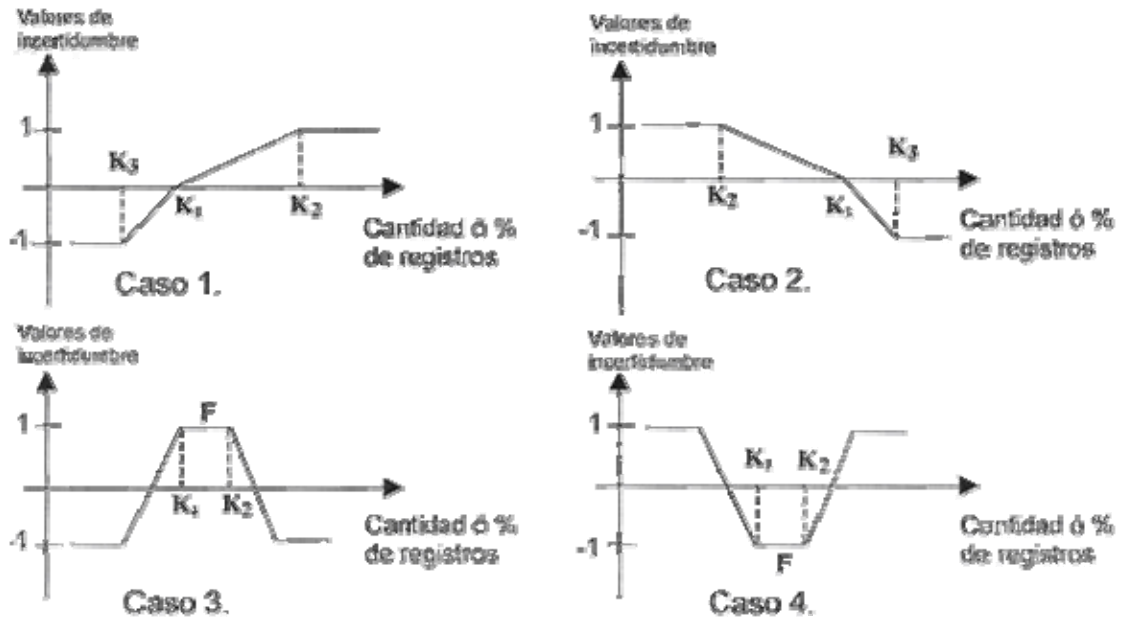


Figura 2.10 Esquema para el cálculo del valor de certidumbre de la proposición asociada a la variable.

Como se puede observar, en la figura 2.10 el eje de las abscisas representa la cantidad o porcentaje de registros que cumplen con la relación $\varphi(x)$ para el CEP o $\varphi(x):\beta(x)$ para el CUP, mientras que el eje de las ordenadas representa los valores de certidumbre asociados a las cantidades de registros que cumplen la relación.

Ejemplo: La expresión $\exists^{>70\%, 80\%, 65\%} x \varphi(x)$, indica que la mayoría de los objetos cumplen la relación $\varphi(x)$, valor de certidumbre, de la siguiente manera:

1. "Positivo", si el 70% o más de los casos satisfacen $\varphi(x)$.
2. "Absolutamente segura" si, como mínimo, el 80% de los objetos cumplen $\varphi(x)$.
3. "Negativo", si el 70% o menos de los casos satisfacen $\varphi(x)$.
4. "No se cumple" con "Seguridad Absoluta", si $\varphi(x)$ es verdadera para el 65% o menos de los casos.

2.5.2.5. Variable Cuantificador.

La implementación computacional de los cuantificadores parametrizados se realizó a través de una variable que fue denominada "variable cuantificador".

Se llama Variable Cuantificador (**VC**) a una estructura que permite representar una situación mediante los Cuantificadores Parametrizados (Universal o Existencial) y se expresa de la siguiente manera:

$V_{Cj} = \langle T_{Cj}, Vr_{Cj}, Pr_{Cj}, Atr_{Cj} = \langle CBD_{Cj}, PL_{Cj}, Q_{Cj} \rangle \rangle$ donde

T_{Cj} y Vr_{Cj} se interpretan como en la estructura general de las variables,

Pr_{Cj} constituye una proposición, cuyo valor representa el resultado de la evaluación del cuantificador que se define,

Atr_{Cj} identifica los atributos específicos como sigue:

CBD_{Cj} define las características de la conexión con una BD,

PL_{Cj} representa los parámetros de localización,

Q_{Cj} identifica el tipo de cuantificador y sus parámetros correspondientes.

La conexión con la BD se representa de la misma manera que en **VED**, pero los parámetros de localización (PL_{Cj}) se diferencian en que aquí no existe un registro guía, puesto que el análisis se ejecuta sobre todos o un subconjunto determinado de la BD.

La componente Q_{Cj} , por su parte, se define mediante los siguientes parámetros:

- Tipo de cuantificador, **CEP** o **CUP**.
- Tipo de análisis, que puede ser, "Exacto" o "Inexacto".
- Si el análisis es Exacto se introducen los parámetros **C** y **K**.
- Si el análisis es Inexacto se introducen **C**, **K₁**, **K₂**, y **K₃** o **F** en correspondencia con el caso.
- Fórmula $\varphi(\mathbf{x})$.
- Si el tipo de cuantificador definido es **CUP** se necesita además introducir la fórmula $\beta(\mathbf{x})$.

El funcionamiento general de esta variable, se puede resumir en los siguientes pasos:

1. Establecer la conexión con la BD.
2. Obtener la cantidad de registros que cumple la condición $\varphi(\mathbf{x})$.
3. Si se evalúa el **CUP** obtener, entre los registros que cumplieron $\varphi(\mathbf{x})$, la cantidad que cumple $\beta(\mathbf{x})$.
4. Evaluar el cuantificador, en correspondencia con los parámetros **C** y **K** si el análisis es Exacto o en correspondencia con **C**, **K₁**, **K₂**, y **K₃** o **F** si el análisis es Inexacto.

5. Asignar el valor de certidumbre resultado de las operaciones, en la proposición asociada incluida en el conjunto **Pr_{cj}**.

2.6. Ventajas de la Incorporación de las Estructuras para la Manipulación de BDs.

La incorporación de las variables extracción de datos y variable cuantificador dentro del medio ambiente HArises, constituyen un apoyo en la construcción de BCs donde la solución a los problemas necesite el empleo tanto de conocimiento como de datos.

El manejo de la información proveniente de una BD se realiza de una forma muy sencilla, puesto que la estructura de la variable extracción de datos puede ser contenida en una BC como una FRC, sin necesidad de realizar modificaciones a las estructuras originales de una BC desarrollada en HArisesA.

La variable cuantificador para la consulta de datos se implementa de una manera muy práctica, puesto que su definición esta basada en las definiciones del **CEP** y **CUP**, haciendo más fácil la comprensión de su naturaleza.

Ambas variables pueden ser utilizadas para manipular distintos tipos de datos contenidos en diversas fuentes, logrando con esto que el conocimiento representado en una BC sea más amplio.

El empleo de los datos obtenidos de la BD por ambas variables, para la solución de problemas utilizando conocimiento, representan un nuevo tipo de conocimiento que manejan las BC desarrolladas en HArises.

El sistema que se ha implementado tiene la capacidad de conectarse a diversos manipuladores de bases de datos como son: Oracle, Informix, SyBase, Fox Pro, Access, SQL Server, y cualquier otro que tenga disponible comunicación vía ODBC.

La BD no tiene que estar ubicada en la propia computadora donde se encuentre instalado el sistema, es decir, el acceso puede ser remoto.

Es importante señalar que los algoritmos para la consulta a una BD han demostrado ser bastante eficientes. Se realizó una prueba con una BD desarrollada en Informix de más de 300,000.00 registros de autos robados (ver epígrafe 6.2.3) y el sistema logró su análisis en un tiempo que osciló entre 7 y 8 minutos.

Capítulo III

Diseño y Funcionamiento de las Variables Extracción de Datos y Cuantificador dentro del Sistema HAriesA.

3.1. Introducción.

Como ya hemos mencionado, los módulos encargados de adquirir las estructuras de las variables extracción de datos y cuantificador han sido adicionados al sistema HAries con la finalidad de proporcionar herramientas para la construcción de sistemas inteligentes híbridos, de acuerdo con las necesidades de información que se tengan en una BC.

En este capítulo se presenta el diseño y funcionamiento de dichos módulos. Para realizarlo se utilizó como herramienta el lenguaje unificado de modelado (UML), ya que proporciona una notación estándar que permite expresar el diseño orientado a objetos de un forma sencilla, facilitando la explicación de las estructuras para la representación del conocimiento variable extracción de datos y variable cuantificador.

En los epígrafes siguientes se explicarán las tareas que llevan a cabo los módulos encargados de adquirir las estructuras de estas variables, en cuanto a la construcción del sistema inteligente, dentro del sistema de adquisición de conocimiento HAriesA.

3.2. Sistema para la Adquisición de Conocimiento HAriesA.

El medio ambiente HAries proporciona las estructuras para la representación del conocimiento que permiten crear BCs que modelen dominios específicos. El sistema para la adquisición del conocimiento del medio ambiente HAries (HAriesA) suministra las interfaces necesarias para adquirir dichas estructuras.

Entre las distintas FRCs disponibles en el lenguaje, se encuentran las estructuras del tipo variable, concepto explicado en el epígrafe 2.2.2., en las que todas las operaciones para la manipulación de las estructuras son válidas, y son representadas en el diagrama de caso de uso correspondiente al proceso de adquisición mostrado en la figura 3.1. Donde se pueden observar también las opciones editar, compilar, guardar y visualizar, correspondientes a las variables que tiene a su disposición el IC.

Leer, es la opción que se ejecuta al momento de realizar la carga de los archivos contenidos en una BC. Esta consiste en interpretar las estructuras almacenadas previamente, ya sea en un archivo binario o en un archivo texto, con la finalidad de manipular las estructuras en la edición correspondiente. La lectura de la variable también podrá ser realizada desde el diálogo, en este caso se utiliza una operación para actualizar diálogo con la finalidad de presentar al IC los parámetros contenidos en la estructura de

la variable. Actualizar variable es la operación que efectúa el reemplazo de los parámetros de la estructura correspondiente desde la información visual que ha proporcionado el IC.

La operación de edición puede ser de tres tipos: crear una nueva variable, modificar o eliminar alguna existente, logrando así el mantenimiento de las estructuras del tipo variable contenidas en la BC de una forma muy sencilla, puesto que se cuenta con ambientes visuales que facilitan estas operaciones de manipulación correspondientes.

La opción compilar, se refiere al proceso de traducción de las estructuras del tipo variable a un archivo binario, el cual está codificado de forma tal que el sistema HArIESC pueda interpretarlo durante la consulta a la BC para evaluar un caso específico.

Guardar, es la opción con la que se salvan las estructuras del tipo variable en un archivo con formato texto o formato binario. Ambos formatos tienen una estructura diferente para cada tipo de variable, de acuerdo a los componentes que contiene cada una.

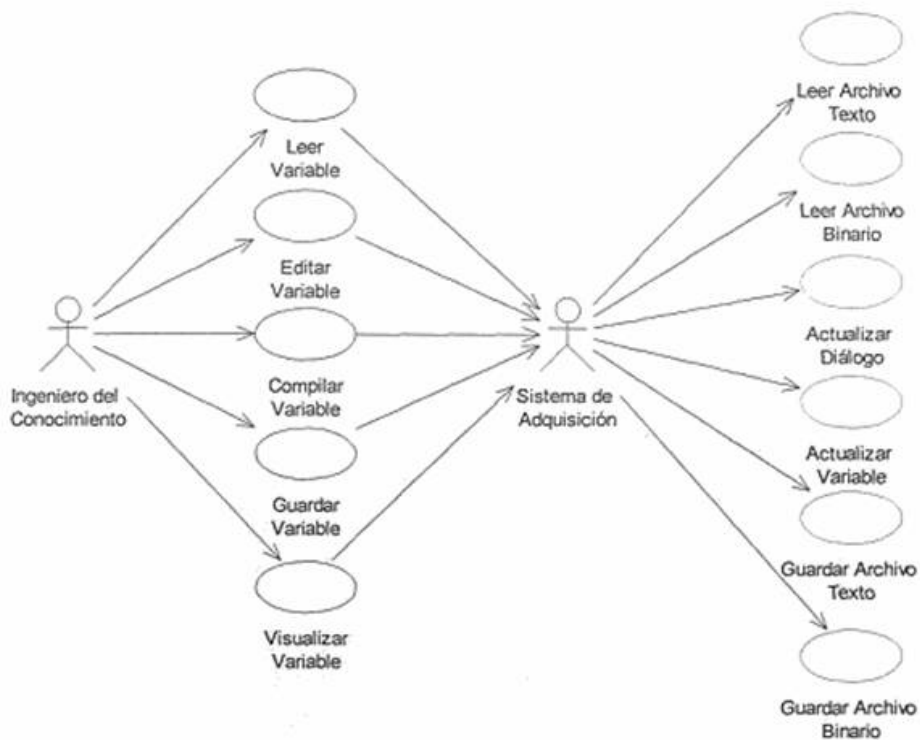


Figura 3.1 Caso de Uso: Manipulación de las estructuras tipo variable en el sistema HArIESA.

Por último, la opción visualizar, ofrece un listado de las componentes de todas las variables que contiene una BC. Con la finalidad de que el IC pueda verificar los elementos que contienen las variables definidas en la BC de una forma más sencilla y

práctica de cómo se realizaría con la edición de cada una por separado. Independientemente a la propia operación visualizar, cuando se ejecuta la carga de la BC estas pueden ser visualizadas.

Para implementar computacionalmente estas operaciones se diseñó la jerarquía de clases que se muestra en la figura 3.2, en la que pueden observarse las variables extracción de datos y cuantificador como parte de la jerarquía de clases para las variables del sistema HArisesA. Se observa que estas variables incorporadas heredan las propiedades de las clases Var, ConcepVar y ConexionBD.

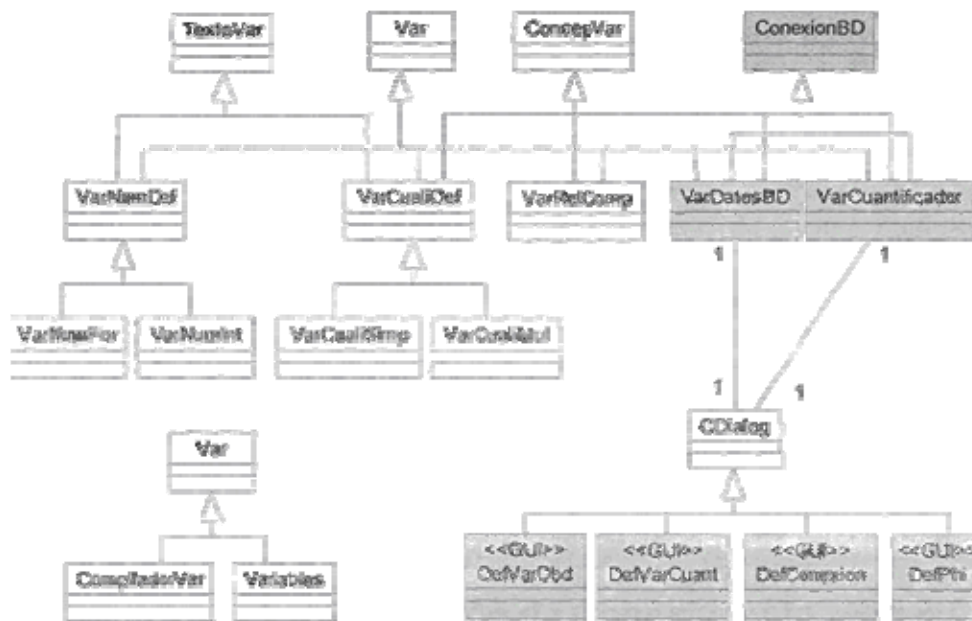


Figura 3.2 Jerarquía de clases de las estructuras del tipo variable en HArisesA.

La clase Var, es la clase base de esta jerarquía definida para implementar las propiedades de herencia y polimorfismo, con lo cual se logra un diseño más eficiente para la ampliación y mantenimiento del sistema de adquisición HArisesA.

Var, es una clase abstracta que contiene la definición de las operaciones que realizan las variables del sistema, utilizadas para crear una nueva variable, editar una existente, leer y guardar la variable en archivo texto, leer y guardar en archivo binario, borrar variable, listar variable, obtener la proposición asociada a la variable, obtener el concepto de la variable, actualizar los parámetros de la variable desde el diálogo, actualizar el diálogo a partir de los parámetros contenidos en la variable y calcular el tamaño en bytes de la variable. Es necesario señalar que todas las clases que heredan de Var tienen sus propias versiones de estos métodos para implementar sus operaciones específicas.

La clase ConcepVar, contiene las propiedades relacionadas con el texto que define el concepto de las variables, que se refiere a una identificación de texto que se proporciona

a la variable para identificarla. Esta clase es utilizada para realizar todas las operaciones relacionadas con el concepto de las variables, como: leer el concepto desde archivo binario, leer el concepto desde archivo texto, guardar el concepto en archivo texto, guardar el concepto en archivo binario, calcular el espacio en bytes del concepto, editar y listar el concepto.

Con el uso del polimorfismo se logran implementar todas las operaciones de manipulación de las variables, definidas en las clases Var y ConcepVar, cada una con los requerimientos específicos de acuerdo al tipo de variable que se este manipulando.

En la clase ConexionBD, se almacenan los requerimientos necesarios para realizar la conexión con la BD. Esta contendrá la especificación de los parámetros, que necesitan ya sea la variable extracción de datos ó la variable cuantificador, con los cuales se realizará el enlace a las BD que serán consultadas, como se definirá a detalle en el capítulo V.

Las interfaces de usuario DefVarDbd, DefVarCuant, DefConexion y Defphi heredan las propiedades de la clase CDialog, que es utilizada para desplegar ventanas de diálogo. Dichas interfaces proporcionan un ambiente visual para la adquisición de los parámetros expresados en las variables extracción de datos y cuantificador del sistema HAriesA.

Para detallar los aspectos específicos en el diseño y funcionamiento de las variables extracción de datos y cuantificador en el sistema para la adquisición del conocimiento HAriesA, se realizará la explicación de los módulos utilizados para su construcción y los sistemas de adquisición con que cuentan ambas variables.

3.3. Diseño del Módulo para la Adquisición de la Variable Extracción de Datos.

La variable extracción de datos, es la estructura de representación del conocimiento proporcionada por el medio ambiente HAries que formará parte de una BC, cuando se requiera obtener información almacenada en una BD, que es necesaria para el proceso que ejecuta el sistema.

Para crear o darle mantenimiento a una variable de este tipo, se debe crear una nueva o leerla de la BC y luego utilizar las operaciones mencionadas en el epígrafe anterior, con la particularidad de que en este tipo de variable los parámetros a definir o manipular se muestran en el caso de uso de la figura 3.3.

El diagrama de caso de uso muestra la interacción que se tiene con el sistema de adquisición de conocimiento durante la edición de los parámetros que componen a la variable extracción de datos.

Los parámetros que el IC define dentro del sistema de adquisición de la variable extracción de datos se refieren a la proposición asociada, el concepto, la definición de la conexión con la BD y la creación de las relaciones existentes entre la BC y la BD.

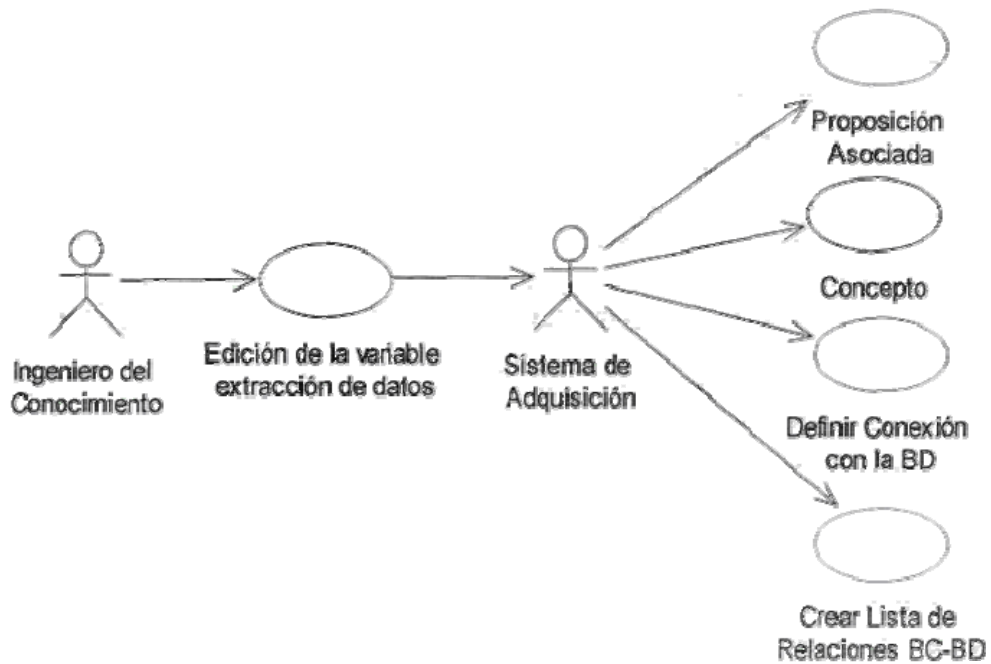


Figura 3.3 Caso de Uso: Edición de la variable extracción de datos.

En caso de existir una proposición asociada, ésta obtendrá el valor de 0 ó 1 en correspondencia a la evaluación o no de la variable durante la consulta de un caso específico en el sistema consultante.

El concepto será utilizado para identificar el objetivo que tiene la definición de un variable extracción de datos, dentro de la BC para el caso particular que se este trabajando.

En cuanto a la conexión con la BD, el IC deberá proporcionar el nombre, el usuario y la clave (password) correspondientes para realizar la comunicación con la BD, además, deberá indicar el nombre de la tabla donde se encuentran los datos requeridos, el campo identificador con el que se guiará la consulta y el registro del cual se desea obtener la información. No obstante, puede ocurrir que algunos de estos parámetros no sean fijos; y por tanto, no sean suministrados durante el proceso de adquisición.

Para crear la lista de relaciones entre la BC y la BD, que se refiere a la correspondencia; que debe existir entre el campo del cual se va a tomar la información de la BD, y la estructura de la BC. Aquí se describe el tipo de estructura dentro de la BC que se desea relacionar, la cual puede ser una proposición, una variable numérica o una variable cualitativa, y el campo correspondiente a la BD donde se encuentran los datos, definiendo con esto la información que se obtendrá de la BD y que se necesita manipular dentro de la BC.

Para contener la estructura de la variable extracción de datos, se definió la clase VarDatosBD como parte de la jerarquía de clases del sistema HArisesA, la cual implementa las operaciones requeridas en cuanto a la adquisición de la estructura se refiere. En la figura 3.4 se presenta el diagrama de clases correspondiente.

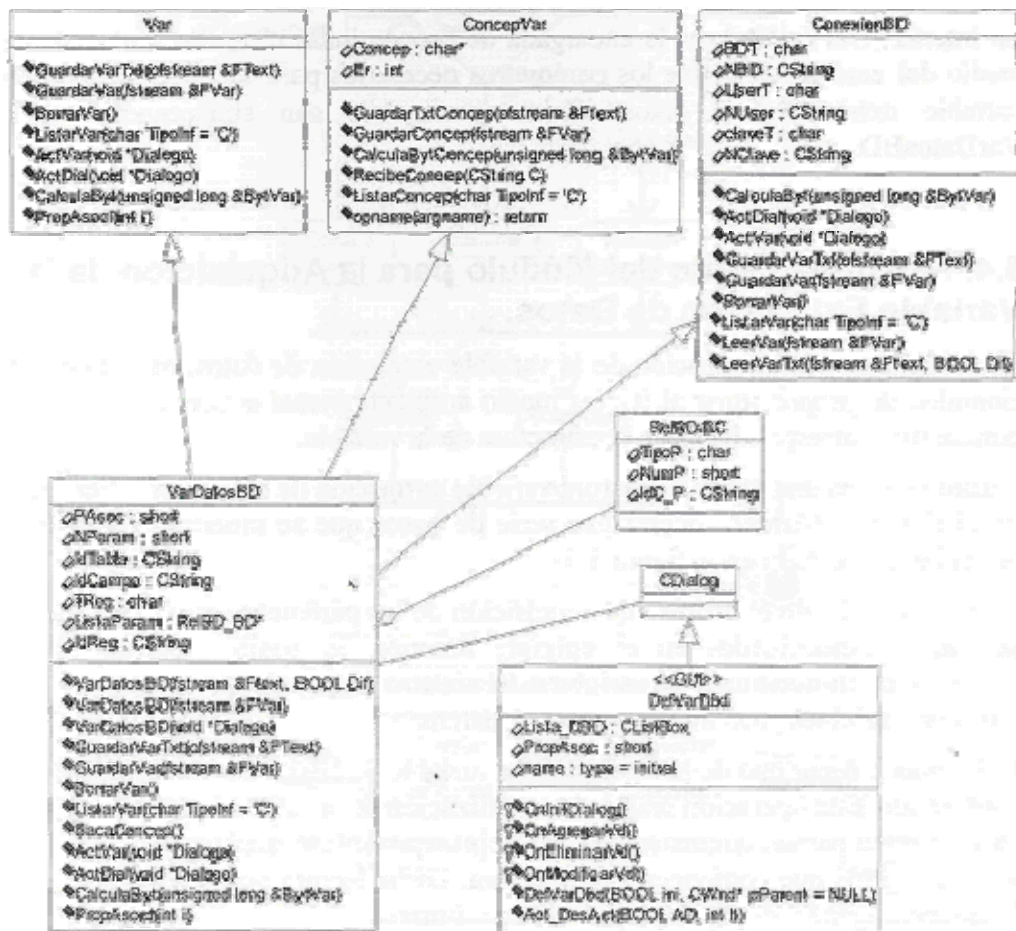


Figura 3.4 Diagrama de clases de la variable extracción de datos.

La clase VarDatosBD cuenta con los métodos para manipular las operaciones de las variables, explicadas en los epígrafes anteriores, estos métodos están principalmente orientados a la implementación de las funciones propias de la variable extracción de datos mediante el uso del polimorfismo.

Además, esta clase facilita el mantenimiento de la variable extracción de datos, ya que proporciona los métodos para modificar y eliminar la estructura, involucrando los métodos para la actualización de los parámetros a partir de la información visual y actualización de la información visual a partir de los parámetros contenidos en dicha estructura, para manipular la información visual contenida en el sistema de adquisición.

La estructura RelBD-BC, proporciona la lista de relaciones, donde se definen los parámetros a partir de los cuales se identifican los elementos de la BD, que contienen los datos a extraer desde la BC. Estos se refieren al tipo de estructura en la cual se asignará la información proveniente de la BD, ya sea proposición, variable numérica ó variable

cualitativa, número de la estructura en la BC y el identificador del campo de donde será extraída la información.

La interfaz DefVarDbd, es la encargada de proporcionar el medio ambiente visual por medio del cual el IC, define los parámetros necesarios para construir la estructura de la variable extracción de datos, dichos parámetros son almacenados en la clase VarDatosBD.

3.4. Funcionamiento del Módulo para la Adquisición de la Variable Extracción de Datos.

El módulo para la adquisición de la variable extracción de datos, ha sido creado con la finalidad de proporcionar al IC, un medio ambiente visual a través del cual, defina los parámetros correspondientes a la estructura de la variable.

Cuando se crea una nueva estructura variable extracción de datos o se edita una existente en el sistema HAriesA, ocurre una serie de pasos que se muestran en el diagrama de actividad presentado en la figura 3.5.

Una vez terminado el proceso de adquisición de los parámetros de la variable extracción de datos, especificados en el epígrafe anterior, es posible realizar una serie de operaciones en la edición de variables del sistema HAriesA, las cuales son válidas para todas las variables proporcionadas por el sistema.

El IC puede hacer uso de la edición de la variable, la cual muestra los parámetros que la conforman. Esta operación realiza la actualización de la información que será presentada en el sistema para la adquisición de conocimiento de la variable, a partir de la lectura de los parámetros que componen su estructura. Dicha lectura puede realizarse, como ya se había mencionado, desde un archivo texto o binario.

La operación eliminar, se deberá utilizar cuando se desee suprimir a la variable de la BC, esta operación borra de forma definitiva la estructura que conforman a la variable seleccionada.

Activar-Desactivar, es la operación utilizada para cambiar el estado de la variable, que deberá ser activada o desactivada para realizar las operaciones para las cuales ha sido definida, al momento de ser ejecutada en el sistema consultante HAriesC.

La compilación de las variables se efectúa con el propósito de que el sistema HAries, cree un archivo binario con la información de la estructura de las variables que será interpretada por el sistema HAriesC al momento de realizar la consulta de un caso en particular.

La operación guardar, salva la estructura en el archivo texto correspondiente, el cual será identificado con el nombre que el IC haya proporcionado para la BC con la extensión ".vbl", con la finalidad de tener un resguardo de las estructuras que se han creado y que puede ser usado también para modificar la información en ella contenida, utilizando cualquier editor. Este es considerado como el código fuente de la BC.

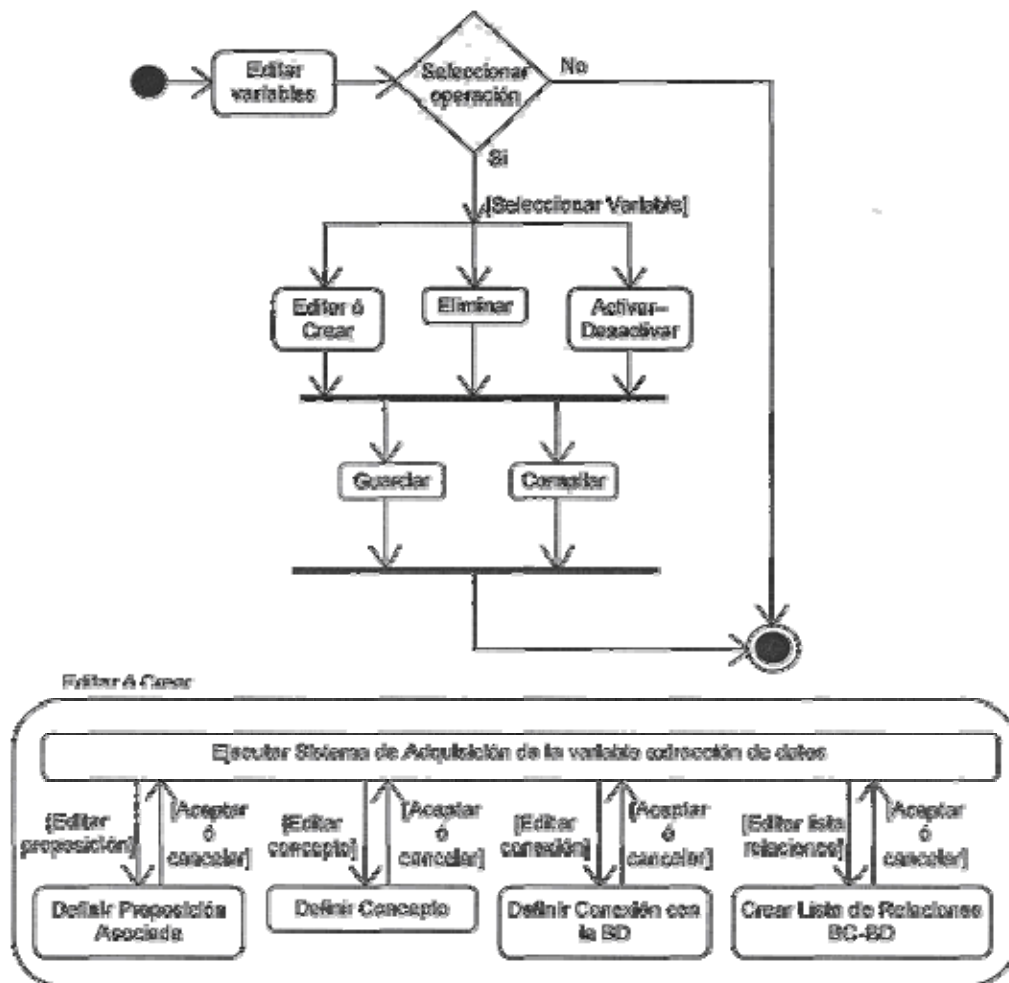


Figura 3.5 Diagrama de Actividad: Proceso de creación o edición de la variable extracción de datos.

El sistema para la adquisición de la variable extracción de datos, cuya pantalla de interacción con el usuario se muestra en la figura 3.6, permite relacionar las estructuras del ambiente HArises con una BD de una forma muy sencilla puesto que solo se deben proporcionar los parámetros que se muestran y que fueron descritos en el epígrafe anterior.

En caso de que la estructura de la BC a relacionar sea una proposición, se obtendrá de la BD un valor correspondiente al peso que se le asignará a la proposición seleccionada. Si el elemento de la BC que contendrá la información proveniente de la BD es una variable numérica, se recibirá de la consulta el valor numérico correspondiente. Para la variable cualitativa se extraerá de la DB un valor que exprese una categoría o cualidad.

También se deberá tomar en cuenta que en caso de que la variable extracción de datos, tenga proposición central asociada, ésta tomará como peso 1 al final de la asignación de

las relaciones correspondientes, independientemente de que se encuentren otras estructuras del tipo proposición entre las relaciones especificadas

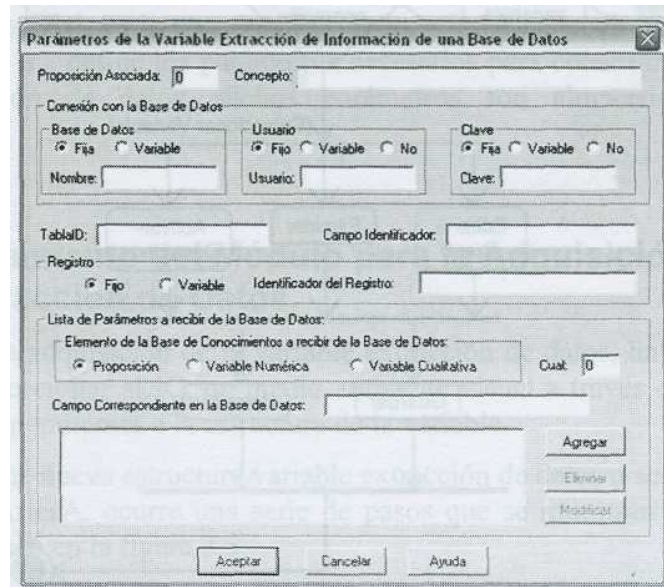


Figura 3.6 Interfaz de Adquisición de la variable extracción de datos.

3.5. Diseño del Módulo para la Adquisición de la Variable Cuantificador.

La variable cuantificador es la FRC del lenguaje HArises, encargada de consultar una BD determinada, con la finalidad de evaluar el cumplimiento o no de un cierto planteamiento general sobre los datos. Como resultado de esta evaluación, se obtendrá la asignación de un peso a la proposición central asociada, que indica el cumplimiento de las propiedades que poseen los datos.

La consulta a la BD a través de la variable cuantificador, consiste en verificar la cantidad de registros que cumplen las condiciones establecidas en la expresión definida en los cuantificadores.

El proceso para la creación de una nueva variable cuantificador, comienza en la pantalla de edición de variables del sistema de adquisición HArisesA, donde el IC activa la adquisición de la variable, proporcionando los parámetros para construir la estructura que contendrá el conocimiento referente a la información requerida de una BD, y a partir de la cual se realizará la cuantificación de los registros que cumplen con un conjunto de cualidades o características necesarias para realizar la evaluación de un caso específico.

Dentro de los parámetros a definir se encuentran aquellos que se refieren a la proposición asociada, el concepto, el tipo de cuantificador y análisis a utilizar, la conexión con la BD, los parámetros de la relación requerida con el resultado de la

cuantificación y las condiciones de consulta que deben cumplir los datos a ser cuantificados.

El caso de uso en el que se ve involucrado el IC en cuanto a la edición de la variable cuantificador se refiere, se muestra en la figura 3.7.

La proposición asociada se define para identificar a la proposición contenida que obtendrá un valor de certidumbre producto de la evaluación de la relación definida en el cuantificador.

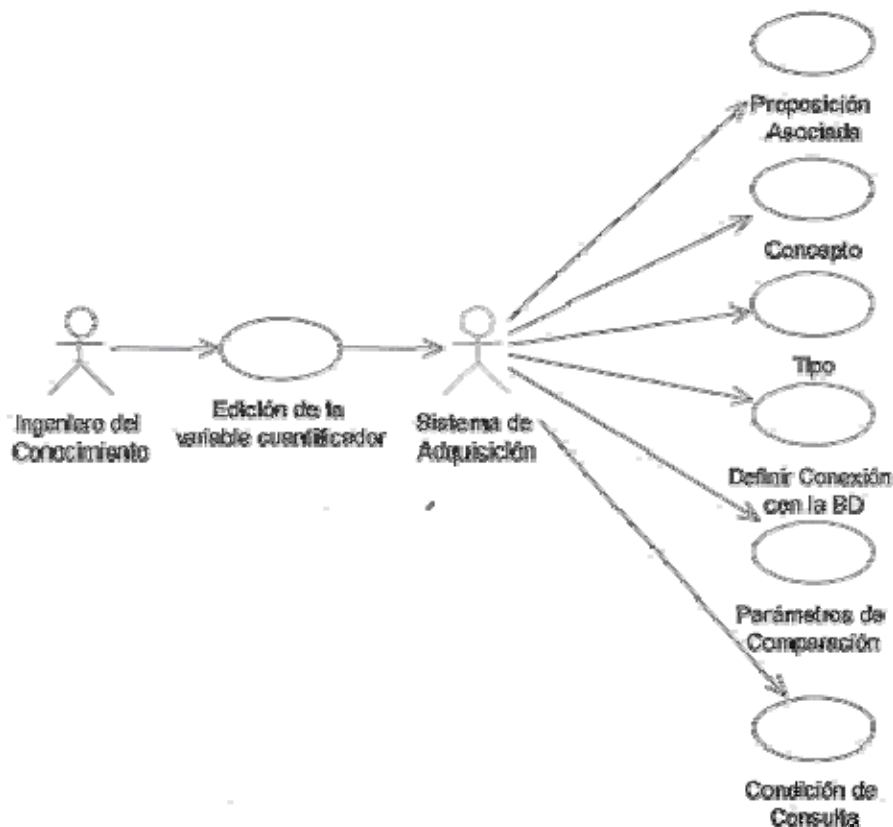


Figura 3.7 Caso de Uso: Edición de la variable cuantificador.

El concepto es un texto utilizado para identificar la función que tendrá la variable dentro de la BC.

El tipo de cuantificador puede ser universal o existencial generalizado. Estos definen si el proceso se ejecuta sobre todos los registros de la BD ó sobre aquellos que cumplan una determinada condición, como fue explicado en el capítulo II.

En el tipo de análisis define si el resultado será un valor exacto (si ó no) ó se necesita un análisis de certidumbre, en cuyo caso se necesita además otros parámetros para definir la función de cálculo.

Al definir la conexión con la BD, el IC deberá proporcionar el nombre, usuario y password de la BD que se desea consultar, el nombre o nombres de las tablas de donde se obtendrá la información y el campo con el cual se guiará el proceso de búsqueda.

La definición de los cuantificadores, se refiere al cambio de la expresión que definen las componentes $\varphi(x)$ y $\beta(x)$ de los cuantificadores, donde se construye la lista de relaciones que identifican las propiedades, que se requiere que cumplan los registros de la BD a explorar, con la finalidad de realizar posteriormente la cuantificación de aquellos que cumplen con las características establecidas, el resultado de la cuantificación se utilizará en correspondencia con los parámetros especificados en la variable cuantificador, para asignar el valor de certidumbre correspondiente a la proposición central asociada de dicha variable.

La clase VarCuantificador, fue anexada a la jerarquía de clases del sistema HAriesA para contener la estructura de la variable cuantificador y realizar las operaciones requeridas en cuanto a la adquisición de la estructura se refiere. En la figura 3.8 se presenta el diagrama de clases correspondiente.

Esta clase contiene los métodos que ejecutan las operaciones básicas que realizan las variables del medio ambiente HAries, durante el proceso de adquisición, para implementar las relacionadas con la variable cuantificador.

En ella se proporcionan las estructuras necesarias, para contener la lista de parámetros que componen la expresión de los cuantificadores $\varphi(x)$ y $\beta(x)$.

También contiene un método para presentar en la pantalla de adquisición de la variable cuantificador, la gráfica correspondiente con el esquema de cálculo a realizar por la variable cuantificador cuando se define un tipo de análisis inexacto, así como los métodos necesarios para la validación de los datos proporcionados por el IC, para evitar una construcción errónea de la variable.

La estructura EsquemaProp es utilizada para la definición del esquema con el cual se desea que la variable realice la evaluación, como se explico en el capítulo II.

La estructura Relaciones, contiene la lista de relaciones que componen la expresión de los cuantificadores $\varphi(x)$ y $\beta(x)$.

La estructura Miembros, contiene a los miembros componentes de cada relación, que a su vez forman parte de la expresión del cuantificador ya sea $\varphi(x)$ o $\beta(x)$. Estos componentes especifican las tablas donde se localizan los campos a partir de los cuales se obtendrá la información.

La interfaz DefVarCuant, proporciona un ambiente visual por medio del cual el IC suministra los parámetros que son almacenados en los atributos de la clase VarCuantificador y que son necesarios para construir la expresión con la que se realizará la cuantificación de los registros contenidos en la BD. En esta interfaz se proporcionan

las opciones necesarias para definir las expresiones de los cuantificadores $\varphi(x)$ y $\beta(x)$, posteriormente son almacenadas en la estructura Relaciones. También se proporciona la interfaz DefConexion, donde se definen los requerimientos necesarios para realizar la conexión con la BD.

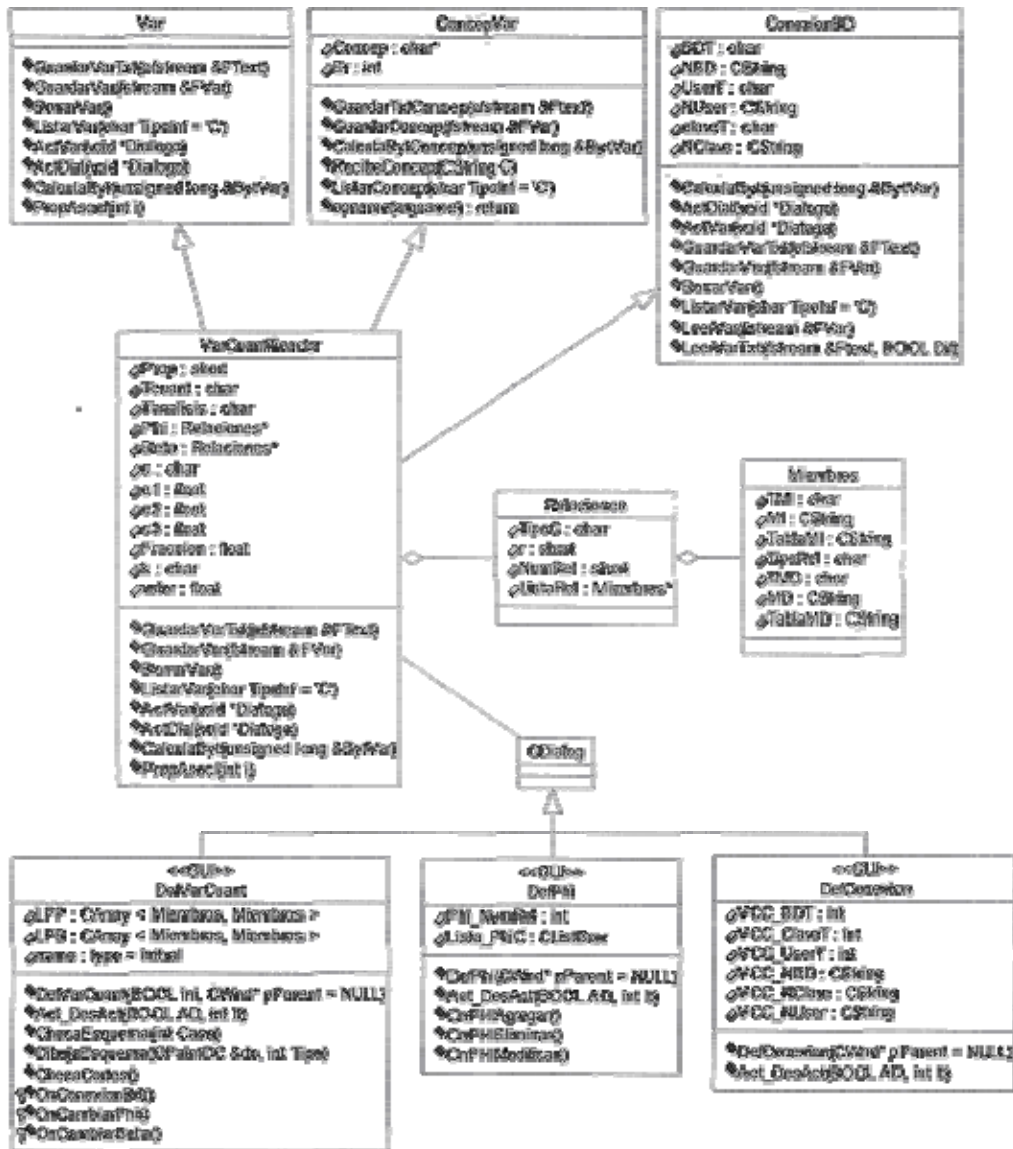


Figura 3.8 Diagrama de clases de la variable cuantificador.

3.6. Funcionamiento del Módulo para la Adquisición de la Variable Cuantificador.

El módulo para la adquisición de la variable cuantificador, se ha diseñado con la finalidad de proporcionar un ambiente visual, para la adquisición de los parámetros que conforman a la estructura para la representación del conocimiento denominada variable cuantificador.

El diagrama de actividad mostrado en la figura 3.9, es utilizado para describir el proceso de adquisición de la variable cuantificador. Dicho proceso comienza con la ejecución del sistema de adquisición HArisesA, donde se encuentra la ventana para la edición de las variables, que permite elegir las acciones editar ó crear una nueva variable del tipo cuantificador.

Al elegir la opción de crear nueva variable cuantificador, se activa el sistema de adquisición correspondiente, donde se definen los parámetros propios de la estructura, explicados en el epígrafe anterior, haciendo uso de los diálogos para la definición de la conexión con la BD, $\varphi(x)$ y $\beta(x)$.

En el diálogo de definición de la conexión con la BD, se introducen los datos de la fuente de la cual se obtendrá la información. Estos datos se refieren al nombre, usuario y clave de acceso a la BD en cuestión.

En los diálogos para la definición de $\varphi(x)$ y $\beta(x)$, se proporcionan los parámetros para construir la expresión del cuantificador, para realizar la consulta a la BD requerida y obtener la cantidad de registros de la BD que cumplen la condición establecida.

Los demás parámetros que componen la estructura de la variable extracción de datos, se deberán especificar en la pantalla principal del sistema de adquisición de dicha variable.

Una vez creada la variable cuantificador, se pueden realizar varias operaciones sobre esta, como editar, eliminar, activar-desactivar, compilar y guardar, el funcionamiento de estas operaciones es similar al utilizado por las demás estructuras del tipo variable del sistema HArisesA. Con estas opciones se complementan las operaciones necesarias para facilitar el mantenimiento adecuado de esta representación del conocimiento, en lo que se refiere a su adquisición.

El sistema de adquisición de la variable cuantificador, permite generar la expresión requerida para cuantificar, en relación ya sea de un número exacto o un porcentaje determinado de registros contenidos en la BD, dependiendo del problema en particular que se este tratando. Su construcción es muy sencilla, ya que el sistema proporciona todas las herramientas para que el IC defina los parámetros necesarios para arribar al resultado final esperado. La interfaz utilizada para la interacción del IC con el sistema para la adquisición de la variable cuantificador se muestra en la figura 3.10.

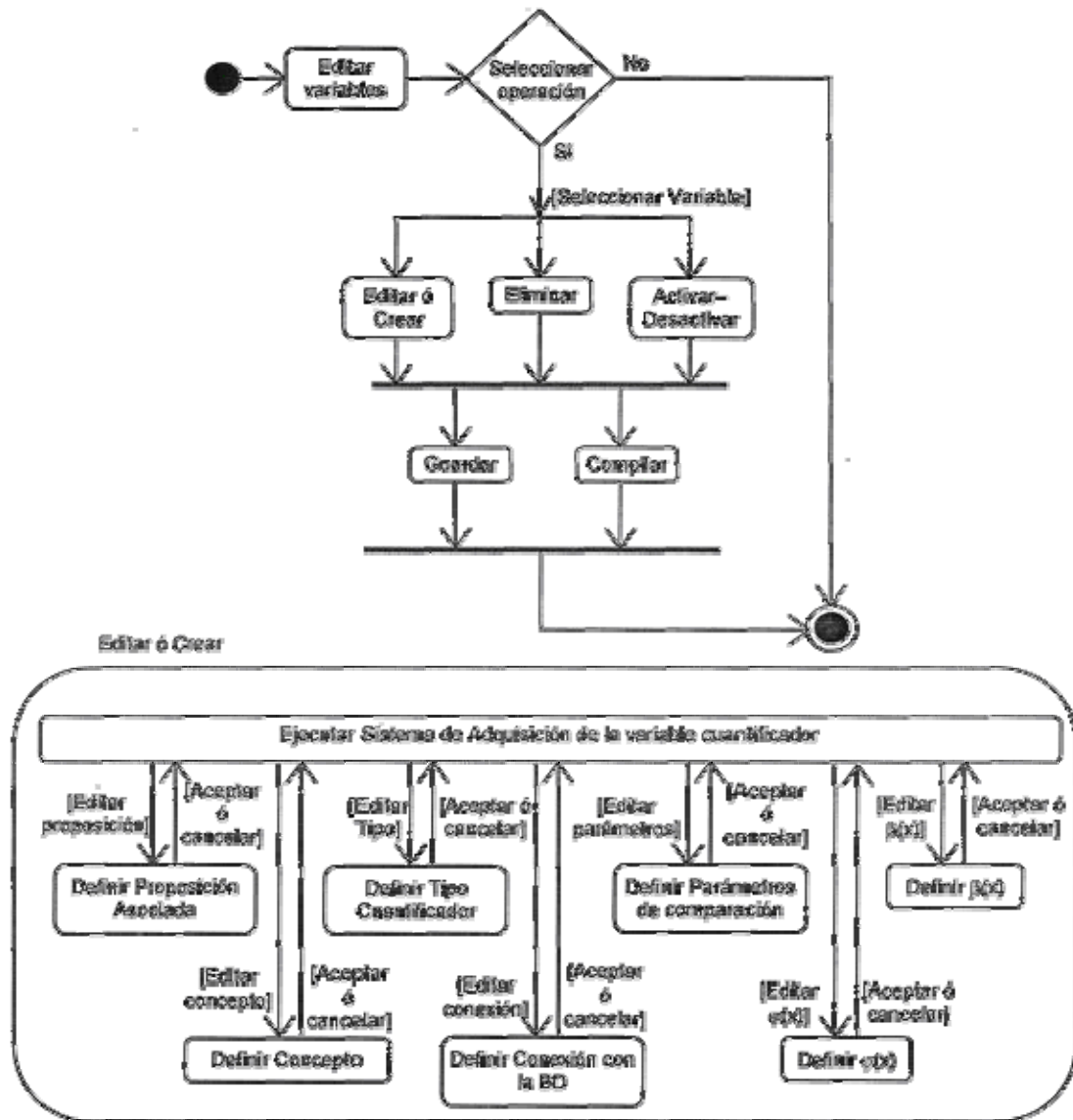


Figura 3.9 Diagrama de Actividad: Proceso de creación o edición de la variable cuantificador.

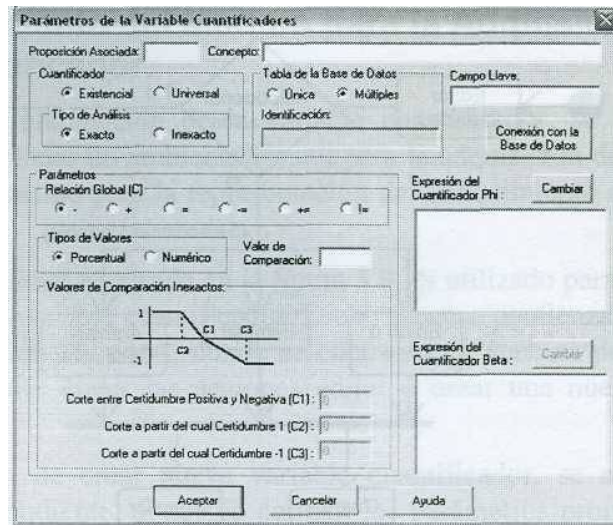


Figura 3.10 Interfaz para la adquisición de la variable cuantificador.

Capítulo IV

Diseño y Funcionamiento de las Variables Extracción de Datos y Cuantificador dentro del Sistema HAriesC.

4.1. Introducción

En el capítulo anterior se detalló el diseño y funcionamiento de las estructuras variable extracción de datos y cuantificador, en cuanto a su adquisición dentro del sistema HAriesA. En este capítulo realizaremos la explicación del diseño y funcionamiento del procesamiento de estas variables dentro del Sistema Consultante HAriesC.

Los módulos encargados de procesar las estructuras de las variables extracción de datos y cuantificador han sido adicionados al sistema HAriesC, con la finalidad de proporcionar los mecanismos necesarios para consultar, de acuerdo con las necesidades de información que se tengan en una BC.

Las tareas que llevan a cabo los módulos encargados de procesar las estructuras de las variables extracción de datos y cuantificador, dentro del sistema para la consulta de un caso con HAriesC, se explicarán en los epígrafes siguientes. Se utiliza UML, igual que en el capítulo IV, como herramienta para presentar el diseño y funcionamiento de los módulos mencionados.

4.2. Sistema Consultante HAriesC

El sistema consultante del medio ambiente HAries contiene los mecanismos y estrategias de procesamiento, que permiten utilizar una BC dada para solucionar un caso en particular.

En el sistema consultante HAriesC se encuentran las máquinas de inferencia de las variables extracción de datos y cuantificador, encargadas de procesar estas estructuras para la representación del conocimiento y otorgar los resultados para los cuales fueron construidas.

La clase para procesar la variable extracción de datos (VarDatosBD) y la clase para procesar la variable cuantificador (VarCuantificador) heredan las propiedades de las clases Var, ConcepVar y ConexionBD, de igual forma que en su diseño para la adquisición. Todas ellas forman parte de la jerarquía de clases de la estructura variable en HAriesC, la cual se muestra en la figura 4.1. Sin embargo, los miembros funciones de las clases son diferentes.

Var es la clase base que implementa el polimorfismo a través de la herencia de las variables en HAriesC, sus operaciones son utilizadas para activar el proceso de

evaluación, obtener las proposiciones asociadas y listar los parámetros que contienen las estructuras del tipo variable.

La clase ConcepVar proporciona las operaciones para obtener y listar el concepto asociado a las variables, a partir de la lectura del archivo binario correspondiente.

ConexionBD es la clase utilizada para leer, desde el archivo binario correspondiente, los parámetros para la conexión con la BD, proporciona los métodos para realizar la búsqueda de la BD, realizar consultas para obtener la información y activa el diálogo para la identificación de los parámetros que establecen la conexión con la BD en caso de ser necesario.

Las operaciones básicas que realizarán las variables extracción de datos y cuantificador, en cuanto a su procesamiento en el sistema consultante HAriesC, serán lectura y visualización desde archivo binario.

El sistema HAriesC, como ya se había mencionado, ha sido desarrollado con la finalidad de proporcionar las máquinas de inferencia encargadas de procesar las estructuras para la representación del conocimiento contenidas en una BC desarrollada en HAriesA.

El proceso que lleva a cabo la máquina de inferencia encargada del procesamiento de las variables extracción de datos y cuantificador se explica a continuación, partiendo del diseño y funcionamiento de sus módulos respectivos.

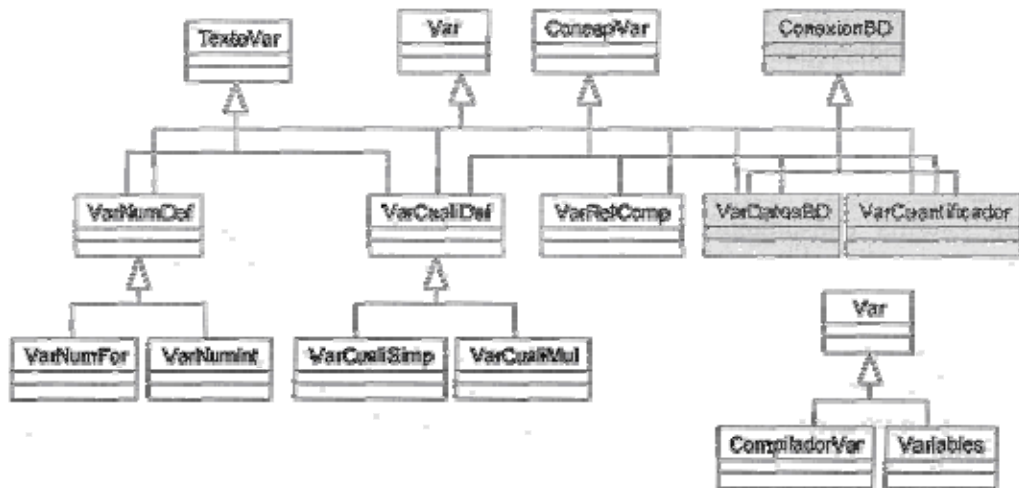


Figura 4.1 Jerarquía de clases de las variables en HAriesC

4.3. Diseño del Módulo para el Procesamiento de la Variable Extracción de Datos.

La variable extracción de datos dentro de HArIESC esta diseñada para procesar la estructura para la representación del conocimiento definida en el lenguaje, con el objetivo de extraer valores de una BD.

La interacción que tiene el usuario con el sistema consultante HArIESC, durante la ejecución de una consulta, es sólo para ejecutar el procesamiento de la BC y otorgar los datos que el sistema le solicite, en caso de ser necesario.

Como se puede observar en el diagrama de caso de uso, mostrado en la figura 4.2, la BC, que entre las acciones que realiza se encuentra la de obtener valores de una BD, ejecuta el procesamiento de la estructura variable extracción de datos para obtener la información requerida.

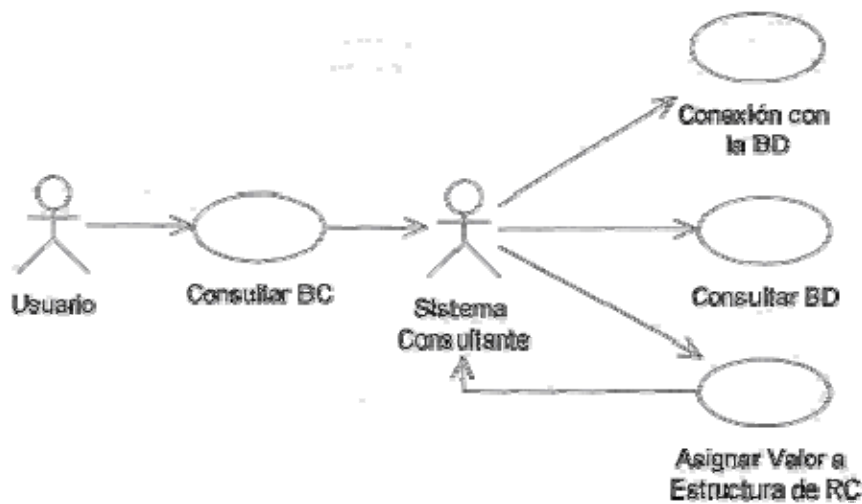


Figura 4.2 Caso de Uso: Procesamiento de la variable extracción de datos.

El procesamiento de la variable extracción de datos agrega la funcionalidad requerida por una BC desarrollada en HArIESA, para realizar la tarea encaminada a obtener información de un BD. Este consiste en realizar la conexión con la BD correspondiente para buscar los datos requeridos durante la consulta y posteriormente asignar a las estructuras correspondientes los valores que ha obtenido.

La máquina de inferencia desarrollada es la encargada de realizar el procesamiento de la estructura, la cual es activada en el momento que algunas estructuras de representación del conocimiento requiera de valores que deben ser obtenidos de una BD.

De igual forma que en el sistema de adquisición, en este sistema se tiene una clase encargada de ejecutar los diversos pasos para poder realizar el proceso para el cual fue

diseñada la variable extracción de datos, el diagrama de clases correspondiente es mostrado en la figura 4.3.

Var es la clase base definida para implementar las herencias y polimorfismos necesarios para el procesamiento de las variables proporcionadas por el sistema HArries, en este caso se implementan los mecanismos para procesar la estructura de la variable extracción de datos.

ConcepVar es la clase encargada de manipular las operaciones relacionadas con el texto que define el concepto de las variables, las cuales son leer y listar concepto de la variable extracción de datos.

La clase ConexionBD es la encargada de efectuar la comunicación con la BD desde la BC como se explico en el capítulo III.

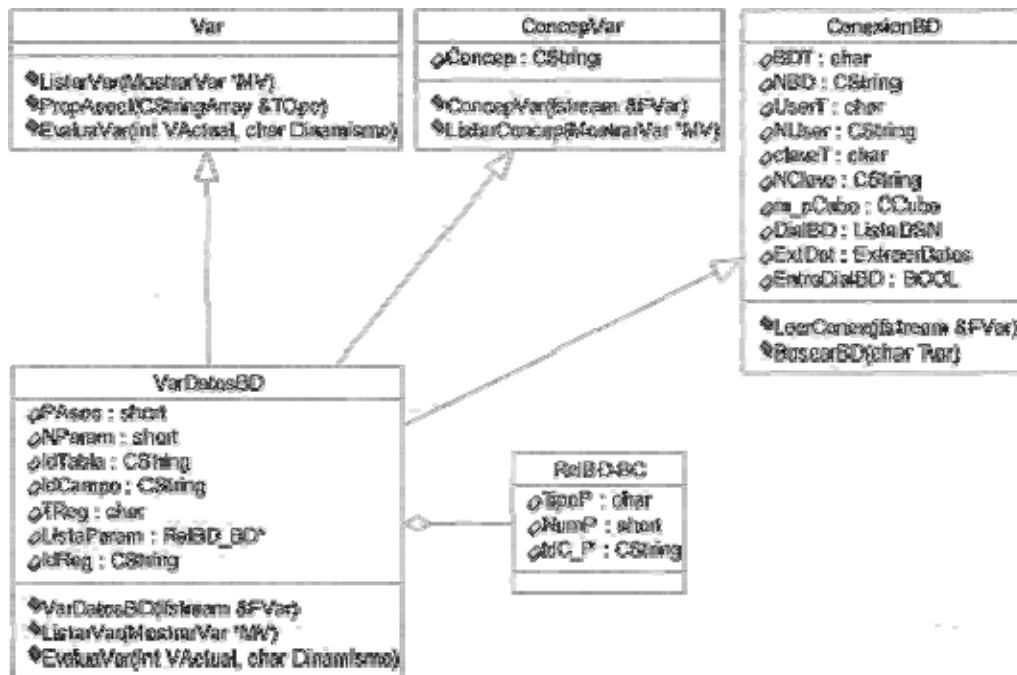


Figura 4.3 Diagrama de Clases de la variable extracción de datos (VarDatosBD).

La clase VarDatosBD es la encargada de proporcionar las estructuras necesarias para procesar la variable extracción de datos.

La estructura RelBD-BC proporciona los atributos necesarios para contener la lista de relaciones BC-BD que manipulará la clase VarDatosBD al momento de ejecutarse el procesamiento de la variable cuantificador.

4.4. Funcionamiento del Módulo para el Procesamiento de la Variable Extracción de Datos.

El módulo para el procesamiento de la variable extracción de datos, realiza la lectura de los archivos correspondientes a su estructura y contiene la máquina de inferencia para su procesamiento.

La ejecución de la variable extracción de datos es llevada a cabo por el sistema consultante HArIESC, en donde se cargan los archivos compilados y se ejecuta la BC. En el momento en que la máquina de control de inferencia del sistema HArIESC requiere del valor de la proposición central asociada a la variable extracción de datos, se activa la evaluación correspondiente, que a su vez realiza las consultas a una BD, finalizando la ejecución con la asignación de información obtenida de la BD a las estructuras de la BC y la correspondiente asignación de peso a la proposición central asociada con un valor de 1. Las tareas antes descritas se ilustran en la figura 4.4. Posteriormente el sistema proporcionará la visualización de los resultados y finalizará la consulta.

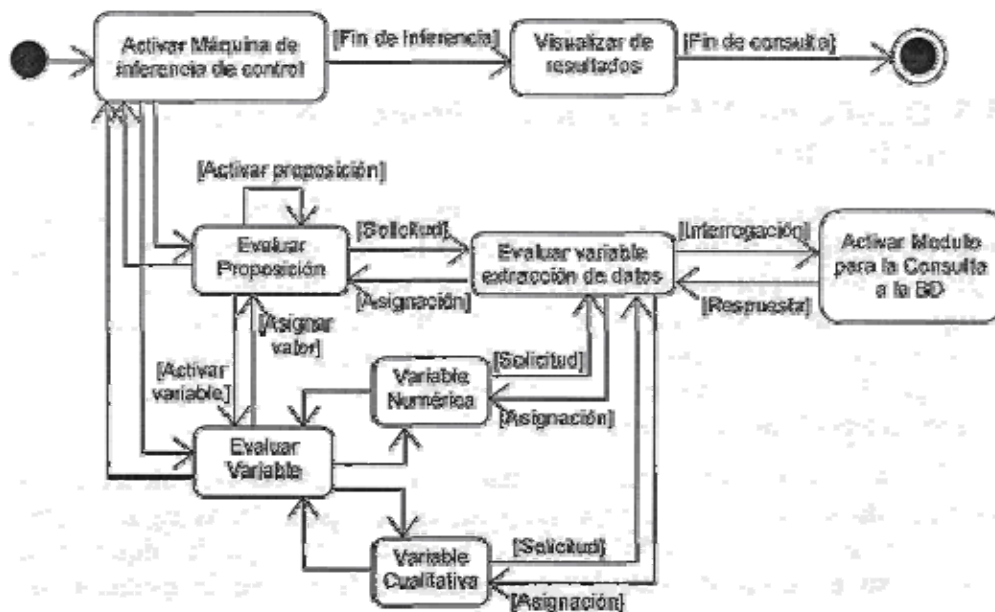


Figura 4.4 Diagrama de Actividad: Procesamiento de la variable extracción de datos.

El procesamiento de la variable extracción de datos también puede ser activado por las estructuras proposición, variable numérica y variable cualitativa, independientemente de que la variable tenga o no proposición asociada.

Cuando la máquina de control de inferencia requiera del valor de certidumbre de una proposición, el cual se encuentra almacenado en una BD y que se ha especificado en la

variable extracción de datos, solicita la evaluación de la variable para realizar la interrogación a la BD y obtener la respuesta que será el valor de certidumbre requerido por la estructura proposición de la BC que se está evaluando.

En el caso de que la máquina de control de inferencia requiera de la evaluación de una variable numérica o cualitativa cuyo valor se desea obtener de la BD, se realiza la ejecución de la variable extracción de datos para obtener los valores requeridos de la BD.

El sistema consultante contiene la máquina de inferencia que realiza el proceso de consulta definido en la variable extracción de datos, proporcionando además una interfaz, la cual se muestra en la figura 5.2 del capítulo V, para definir las relaciones que tiene la BC con la BD en caso de ser necesario. Esta interfaz se activa cuando los parámetros para realizar la conexión con la BD proporcionados por el IC en la adquisición de la variable no concuerdan con los que se encuentran en la BD al momento de realizar la consulta.

La máquina de inferencia utiliza el módulo de comunicación, encargado de realizar el enlace con la BD, para obtener la información que será relacionada con las estructuras de la BC.

4.5. Diseño del Módulo para el Procesamiento de la Variable Cuantificador.

Dentro del sistema consultante de HArIES, el procesamiento de la variable cuantificador tiene como objetivo realizar una consulta a la BD con la finalidad de cuantificar los registros que cumplen con determinadas condiciones. El resultado de la cuantificación es comparado con un conjunto de parámetros que definen la relación que se busca en los datos. Como consecuencia de todo este procesamiento, la variable cuantificador realiza la asignación de un valor de certidumbre a su proposición central asociada en correspondencia con la evaluación efectuada.

El proceso que realiza la variable cuantificador consiste en obtener la cantidad de registros de la BD que cumplen las condiciones establecidas en la expresión de la variable y que han sido definidas por el IC durante la etapa de adquisición. Con este valor y un esquema de cálculo previamente definido, se le asignará un valor de certidumbre a la proposición asociada.

El caso de uso en el que se ve involucrado el usuario al momento de realizarse el procesamiento de la variable cuantificador se muestra en la figura 4.5. Como se puede observar sólo realiza la activación de la consulta con la BC, durante la cual otorgará los datos necesarios para evaluar su caso en particular, si es que el sistema lo requiere.

La máquina para el control de inferencia del sistema consultante HArIESC será la encargada de activar el proceso evaluativo de la variable cuantificador en el momento que se requiera del valor de certidumbre de su proposición asociada.

Primero se efectúa la conexión con la BD a la que se desea explorar, llevando a cabo la cuantificación de los registros que cumplen la condición establecida en los parámetros de la estructura de dicha variable.

El resultado producto de la cuantificación de los registros de la BD que cumple con un conjunto de características se compara con la relación que se desea evaluar en los datos. Que está definido por un esquema de cálculo previamente establecido y a través del cual, se asigna el peso correspondiente a la proposición asociada, en correspondencia con el tipo de análisis a efectuar por la variable cuantificador.

La ejecución de la variable cuantificador concluye cuando su máquina de inferencia realiza la asignación del valor de certidumbre a su proposición asociada, retornado al control de inferencia del sistema HAriesC.

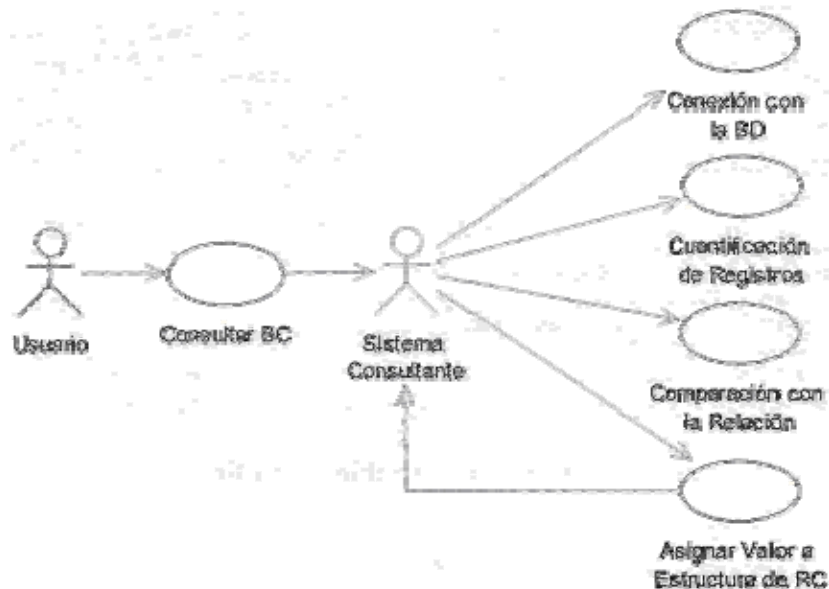


Figura 4.5 Caso de Uso: Procesamiento de la variable cuantificador.

Este procesamiento se implementó computacionalmente con el diseño de clases según la jerarquía que se muestra en la figura 4.6, con la finalidad de proporcionar los mecanismos necesarios para el procesamiento de la variable cuantificador que forma parte de una BC definida en el sistema de adquisición de conocimiento HAriesA.

En esta clase se realiza la definición de los métodos para el procesamiento de la estructura variable cuantificador, implementado así el polimorfismo definido en las clases Var y ConcepVar de las cuales se heredan las propiedades de las variables en HAries.



Figura 4.6 Diagrama de clases de la variable cuantificador (VarCuantificador).

4.6. Funcionamiento del Módulo para el Procesamiento de la Variable Cuantificador.

Cuando la máquina para el control de la inferencia del sistema HArtesC requiere la evaluación de la proposición central asociada a la variable cuantificador, se activará el proceso para evaluar la cantidad de registros que cumplen con la expresión construida en el sistema de adquisición.

La variable cuantificador constituye la estructura que concentra la información y controla la ejecución y comunicación entre la BC y la BD. El diagrama de actividad de la figura 4.7 describe el comportamiento del sistema consultante HArtesC durante su procesamiento.

El proceso de evaluación de la variable cuantificador se activa cuando el mecanismo de inferencia del sistema HArtesC requiere del valor de certidumbre de la proposición

central asociada a dicha variable y será precisamente en ese momento cuando inicie la evaluación de la misma, que comienza con la conexión a la base de datos a la cual se consultará, realizando la cuantificación de los registros que deben cumplir la condición establecida y que posteriormente serán comparados con la relación que se requiere que cumplan el total de los datos cuantificados.

El módulo de comunicación con la BD es utilizado para identificar los elementos involucrados en la cuantificación, en caso de no encontrarlos se activa la interfaz para la conexión con la BD mostrada en la figura 5.2, para realizar esta tarea.

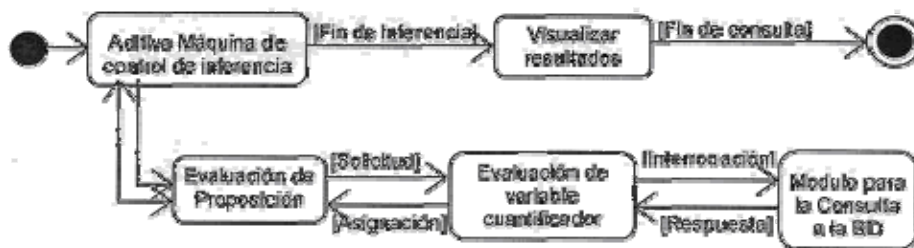


Figura 4.7 Diagrama de Actividad: Procesamiento de la variable cuantificador.

Una vez identificados los elementos involucrados en el procesamiento de la variable cuantificador, se evalúa la condición definida y se realiza la cuantificación de los registros que cumplen la condición especificada en la expresión de la variable cuantificador. Una vez concluido este proceso, se ejecuta la evaluación de la proposición utilizando el esquema de certidumbre correspondiente. Ésta queda evaluada con un valor de certidumbre que indica el grado de cumplimiento de la condición según los datos almacenados en la BD. El valor obtenido de este paso es retornado a la máquina de inferencia como resultado final de todo el proceso.

Se obtendrá como resultado la asignación de un grado de certidumbre a la proposición central, la cual podrá ser utilizada dentro de la BC, logrando con esto ayudar a la toma de decisiones basada tanto en las experiencias como en los datos.

Capítulo V

Ambiente de Comunicación con ODBC.

5.1. Introducción.

En este capítulo presentaremos el ambiente que se ha desarrollado con el fin de facilitar la comunicación con diversos Sistemas Manejadores de Bases de Datos (DBMS, por sus siglas en inglés), así como también la detección de los errores que puedan existir durante la comunicación entre la BC y la BD.

En el presente capítulo se realiza la explicación de las fuentes de datos que pueden ser consultadas, así como también el diseño y funcionamiento del módulo desarrollado para tal efecto.

5.2. Ambiente de comunicación.

Para realizar consultas a datos fue necesario desarrollar un medio ambiente para la comunicación entre una BC y los datos contenidos en diversas fuentes. Este ambiente esta soportado por ODBC (Open DataBase Connectivity - conectividad abierta de bases de datos), el cual es un método para el acceso estándar a BD.

La finalidad de ODBC es la de proporcionar el acceso a datos desde cualquier aplicación, independientemente del manipulador que se utilice, haciendo que la aplicación sea capaz de enviar comandos ODBC que el DBMS pueda responder.

En nuestro trabajo, la aplicación encargada de la comunicación con BD utiliza interfaces, las cuales son de dos tipos. El primero es empleado para la adquisición de los parámetros necesarios con los que identificarán las fuentes de datos a ser consultadas y en el segundo se identifican los parámetros de la BD a consultar durante la ejecución del sistema.

En los siguientes epígrafes realizaremos la explicación de las interfaces utilizadas para la adquisición y ejecución de los parámetros necesarios para la consulta a una fuente de datos durante el procesamiento de las variables extracción de datos y cuantificador.

5.2.1. Interfaz para la Adquisición.

Las variables extracción de datos y cuantificador tienen un sistema propio para la adquisición de los parámetros que componen su estructura, proporcionados por el sistema HArisesA. Dentro de estos parámetros se encuentran aquellos que se refieren a la fuente de datos a ser consultada.

Para la variable extracción de datos se deben de especificar los parámetros referentes al nombre, usuario y password de la BD, así como también la tabla, el campo y registro por medio de los cuales se guiará la búsqueda de los datos a obtener, como se explicó en el capítulo II, a través del sistema para la adquisición de la variable presentado en el capítulo III.

El atributo "Variable" para el parámetro "Base de Datos" se refiere a que los valores necesarios para la variable serán obtenidos de diversas fuentes de datos, o que el nombre de la BD varia dependiendo del usuario que utilizará el sistema.

En cuanto al parámetro "Usuario" se definirá como "Fijo" cuando se desee especificar el usuario que tiene acceso a la BD. El atributo "Variable" indica que existen distintos tipos de usuario y el atributo "No" se utilizará cuando no se requiera de un usuario para acceder a la BD a consultar.

El parámetro "Clave" es utilizado para especificar el password correspondiente al usuario que puede acceder a la BD cuando el atributo es "Fija". En el caso "Variable" se especifica que la clave de acceso se proporcionará en tiempo de ejecución. El atributo "No" indica que no se requiere de este parámetro para realizar la conexión.

Se debe de especificar el nombre de la tabla de la cual se obtendrá la información, esto en cuanto al parámetro "TablaID".

"Campo Identificador", es el campo correspondiente a la BD de dónde será extraída la información.

En el parámetro "Registro" se especifica un registro correspondiente a la BD para guiar la búsqueda. Registro puede ser fijo o variable, en caso de ser fijo se deberá de proporcionar el atributo de la tabla de donde se obtendrán los valores, en caso de ser variable el sistema consultante lo solicitará en tiempo de ejecución.

En cuanto al sistema para la adquisición de conocimiento de la variable cuantificador, los parámetros referentes a la conexión con la fuente de datos a explorar, BD, Usuario y Clave, se definen de la misma manera que para la variable extracción de datos y se especifican en la ventana de diálogo implementada para tal efecto, la cual es mostrada en la figura 5.1, proporcionada en dicho sistema de adquisición de conocimiento.

La tabla de la BD puede ser "Única" ó "Múltiples", para el primer caso se indica cual es la tabla de donde se obtendrán los valores requeridos por la variable. En el segundo caso se deben especificar los nombres de las tablas a ser consultadas, así como, el campo con el que se guará la búsqueda de la información requeridos por las expresiones de los cuantificadores $\varphi(x)$ y $\beta(x)$.

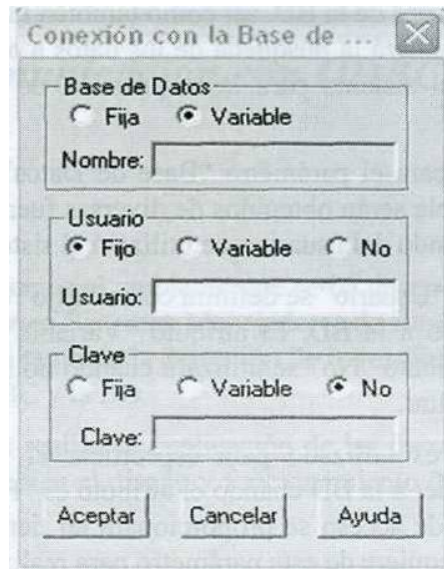


Figura 5.1 Interfaz para la conexión con la BD para la variable cuantificador.

5.2.2. Interfaz para la Ejecución.

En lo que se refiere a la ejecución el medio ambiente para la comunicación, las variables extracción de datos y cuantificador contienen los mecanismos necesarios para activar la consulta a una fuente de datos determinada, cada una con sus propias características, en el momento que se requiere realizar una consulta específica.

Dichos mecanismos son proporcionados por el sistema consultante HAriesC, la máquina para el control de inferencia de este sistema realiza la evaluación de las variables extracción de datos y cuantificador identificando los elementos de la fuente a examinar para obtener la información requerida.

Al activarse cualquiera de las variables, el primer paso a realizar es la comprobación de la conexión, según los parámetros definidos para ello en las variables.

Si los parámetros relacionados con la conexión que contienen las estructuras de la BC coinciden con los que posee la BD, el sistema realizará la consulta sin necesidad de que el usuario realice ninguna operación en cuanto a la especificación de las componentes de la BD relacionadas con el caso que se desea evaluar.

En caso de que los parámetros otorgados en el sistema para la adquisición del conocimiento de las variables extracción de datos y cuantificador no concuerden con los elementos de la fuente de datos con la que se ha realizado la conexión ó en caso que la definición de los parámetros se especificó con algún atributo variable, se activará la interfaz para la especificación de las componentes de la BD que se tomaran en

consideración para la realizar la consulta correspondiente. Esta interfaz es mostrada en la figura 5.2.

En ella se definirán los parámetros requeridos para realizar la consulta específica del caso que se esta evaluando, logrando con esto que la conexión con una BD sea exitosa puesto que se podrán señalar las componentes de la BD de la cuales se obtendrá la información.

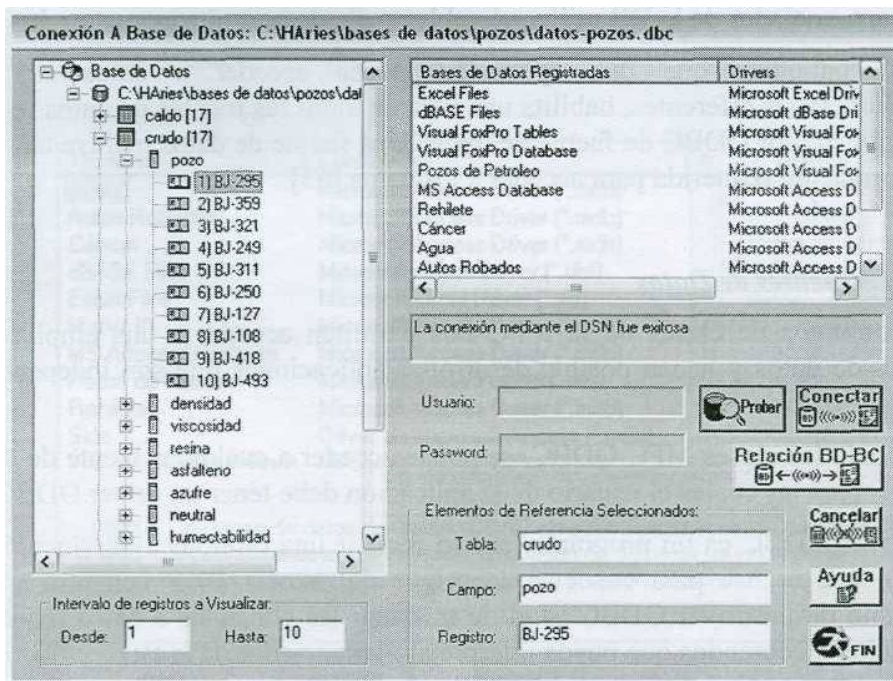


Figura 5.2 Interfaz para la conexión con la BD durante la ejecución o procesamiento de las variables.

5.3. Implementación del Ambiente de Comunicación.

Para el desarrollo del ambiente encargado de realizar la comunicación con la Fuente de datos se ha utilizado la biblioteca de clases Microsoft Foundation Class ODBC (MFC ODBC) ya que proporciona los elementos necesarios para obtener la información requerida proveniente de una BD, a partir de la definición de las estructuras contenidas en el medio ambiente HARies.

A continuación presentaremos la explicación de cómo se realiza la comunicación con BD a través de ODBC, las fuentes de datos que pueden ser consultadas, los tipos de datos que se pueden manipular y el diseño y funcionamiento del módulo de comunicación dentro de los sistemas HARiesA y HARiesC.

5.3.1. Comunicación Vía ODBC.

El estándar ODBC de Microsoft es una interfaz que permite el acceso a diferentes bases de datos desde una aplicación C/C++, Para acceder a una base de datos simplemente necesita el administrador ODBC y el controlador ODBC de la base de datos a la que se quiere acceder.

El funcionamiento de ODBC esta dado por los drivers que emplea para lograr la comunicación con distintos manejadores de BD. El driver en turno interactúa con el sistema manejador de la DB utilizando el lenguaje de consulta estructurado (SQL).

ODBC garantiza que una aplicación pueda acceder a datos en formatos y configuraciones diferentes, habilita una lista de todas las fuentes de datos registradas con el administrador ODBC de fuente de datos. Una fuente de datos incluye tanto los datos y la información requerida para acceder a los datos [63].

5.3.1.1. Fuentes de Datos

Los conjuntos de clases de la MFC nos permiten acceder a una amplia variedad de fuentes de datos y hacen posible desarrollar aplicaciones que son independientes de la fuente de datos.

Utilizando las clases MFC ODBC, se puede acceder a cualquier fuente de datos, local o remota, para las cuales el usuario de la aplicación debe tener un driver ODBC.

Un driver ODBC es un programa, por lo regular una biblioteca de vínculos dinámicos (DLL), que se usa para conectar un origen de datos ODBC con otra aplicación. El propósito de un driver ODBC es el de trasladar las consultas a datos requeridas por la aplicación a comandos que pueda interpretar el manejador de la BD.

El módulo se encuentra diseñado para permitir la comunicación con cualquier manejador de BD que implemente comunicación vía ODBC, pero se ha probado con el siguiente software:

- SQLServer
- Microsoft Access
- Microsoft FoxPro
- dBASE
- Microsoft Oracle ODBC
- Informix
- Sybase
- InterBase

Las BD que serán analizadas por el sistema consultante HAriesC, podrán ser registradas previamente. Para registrar o dar de baja una fuente de datos, se ejecuta el administrador correspondiente que suministra Windows, el cual es mostrado en la figura 5.3, dando doble clic en el icono orígenes de datos ODBC en el panel de control.

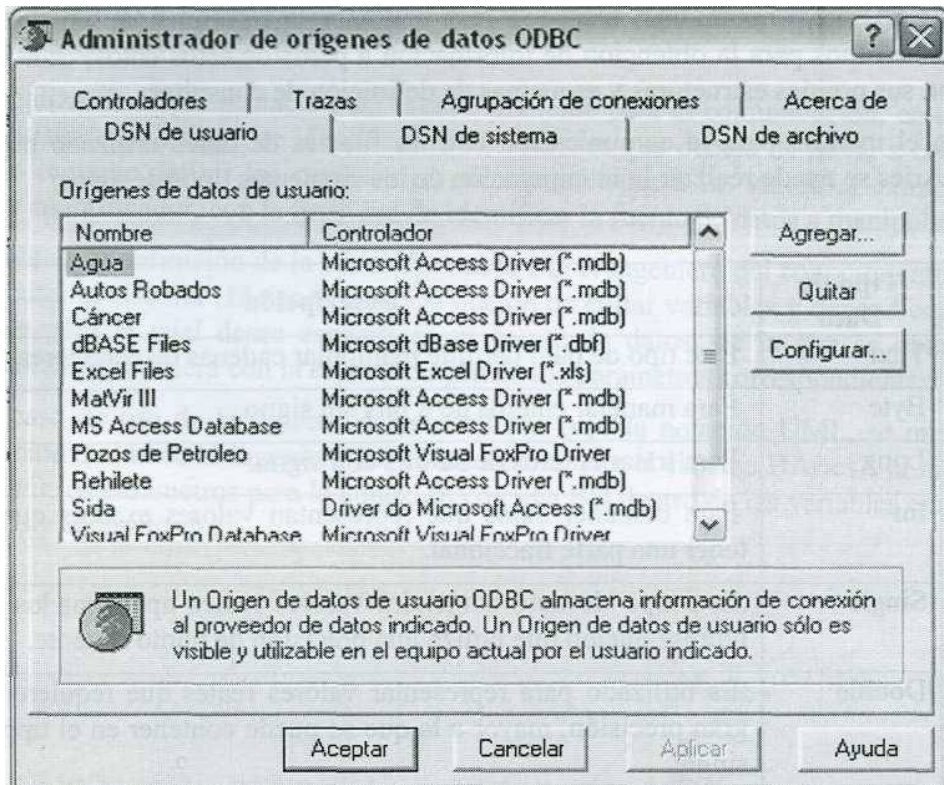


Figura 5.3 Administrador de orígenes de datos ODBC.

Visual C++ normalmente prepara al administrador durante la instalación, pero si se eligió una instalación personalizada, el administrador podría no existir en su sistema. Si el icono orígenes de datos ODBC no aparece en el panel de control, se debe ejecutar el programa de instalación correspondiente para instalar los archivos ODBC necesarios. Es necesario aclarar que no se necesita instalar Visual C++ para el funcionamiento del sistema.

5.3.1.2. Tipos de Datos.

Los tipos de datos son utilizados para representar la información obtenida de la observación de una tarea o fenómeno en particular de un dominio determinado, específicamente para representar los atributos de los objetos o entes detallados dentro de un esquema de BD.

Los BDMS proporcionan formatos para contener la información, así como también, mecanismos para la obtención de información a partir de estos datos, cada uno de ellos con sus propias estructuras y esquemas de definición de consultas.

En el módulo para la comunicación con las fuentes de datos utilizado por el sistema HAries se puede realizar la manipulación de los siguientes tipos de datos:

Tipo de Dato	Descripción
Text	Este tipo de dato permite manipular cadenas de caracteres.
Byte	Para manejar enteros de 8 bits sin signo.
Long	Para tratar enteros de 32 bits con signo.
Int	Para contener datos que representan valores exactos que pueden tener una parte fraccional.
Single	Este tipo de dato contendrá valores reales aproximados con una representación que utiliza una precisión de punto flotante.
Double	Es utilizado para representar valores reales que requieren de una gran precisión, mayor a la que se puede contener en el tipo de dato single.
Bit	Es utilizado para los atributos de la BD que aceptan solamente valores de 0 ó 1. Que es utilizado usualmente para identificar si un objeto tiene o no determinada propiedad, también identificado como tipo de dato booleano por contener únicamente el significado de verdadero ó falso.
Binary	Este tipo de dato soporta arrays dinámicos de bytes.
LongBinary	Tipo de dato para trabajar con objetos binarios, obtenidos de la BD, más grandes que los que se pueden manejar con el tipo de datos Binary.
Date	Para manejar información correspondiente a fechas, con el formato año, mes y día.
TimeStamp	Es utilizado para contener información acerca del tiempo manejando el formato de precisión que contiene año, mes, día, hora, minuto y segundo.

Todos estos tipos de datos permiten manejar la información contenida en las fuentes de datos especificadas en el epígrafe anterior, logrando así que el sistema HAries manipule información proveniente de datos de diverso tipo.

5.3.2. Diseño y Funcionamiento del Módulo de Comunicación en el Sistema de Adquisición.

Para realizar la implementación del medio ambiente para la comunicación, en cuanto a su adquisición se refiere, se identificaron las acciones que deberá de llevar a cabo el IC cuando requiere definir los parámetros, correspondientes a las variables extracción de datos y cuantificador, con la finalidad de identificar la fuente de datos a manipular.

En cuanto a la definición de la conexión con la BD el ingeniero del conocimiento deberá de ejecutar el sistema HAriesA, elegir la opción de editar variables y dependiendo de la variable para la cual desee especificar su fuente de datos, de la que se obtendrá la información, procederá con la especificación de los parámetros correspondientes.

En el caso de uso de la figura 5.4, desarrollado bajo la notación UML, se muestra la interacción que tiene el ingeniero del conocimiento con el sistema HAriesA al momento de definir los parámetros para la conexión con una BD dentro de las variables extracción de datos y cuantificador.

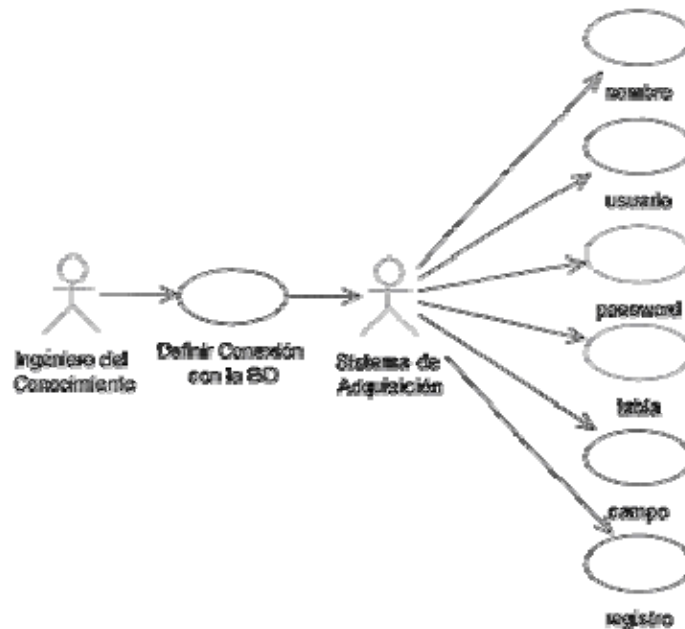


Figura 5.4. Caso de Uso: Definición de la conexión con la BD.

Una vez definida la conexión con la BD, las variables extracción de datos y cuantificador podrán ejecutar una serie de operaciones para almacenar y manejar la información proporcionada por el ingeniero del conocimiento. Estas se ejecutan desde los respectivos sistemas de adquisición de las variables, y que son necesarias para especificar los parámetros que componen las estructuras que se han definido durante el proceso de adquisición, así como también para realizar el mantenimiento correspondiente.

Para la implementación de las operaciones necesarias durante el proceso de adquisición de los parámetros correspondientes a la conexión con la BD, se definió la clase ConexionBD que forma parte de la jerarquía de clases del sistema HArIESA, esto con la finalidad de heredar sus propiedades desde las variables extracción de datos y cuantificador, como se puede observar el diagrama de clases mostrado en la figura 5.5.

La clase ConexionBD proporciona los atributos y métodos necesarios para contener y procesar los parámetros referentes a la conexión con la BD en cuanto a su adquisición se refiere.

En cuanto a los atributos, esta clase puede contener el nombre, usuario y password correspondientes a una determinada BD.

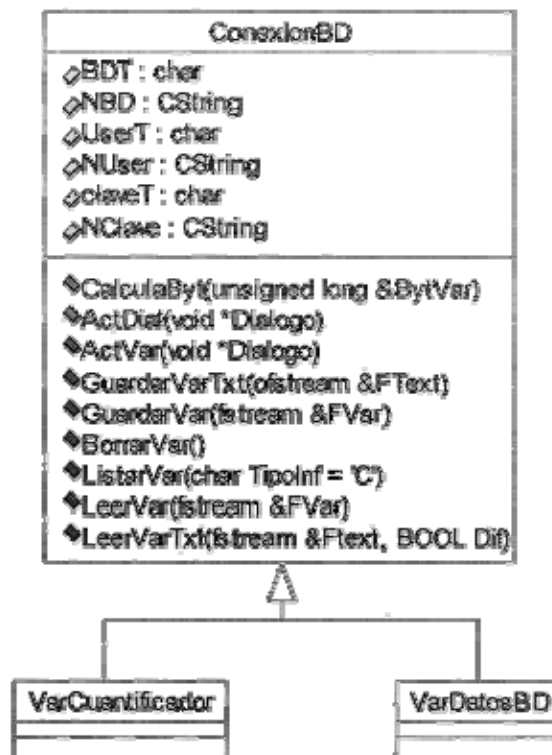


Figura 5.5. Diagrama de clases: Clase ConexionBD dentro de HArIESA.

Los métodos que son implementados en esta clase se utilizan para realizar las operaciones de guardar y leer desde archivo texto, guardar y leer desde archivo binario, actualizar información a partir del diálogo, actualizar el diálogo a partir de la información, calcular el tamaño en bytes de los parámetros y eliminar, correspondientes a la identificación de la BD a consultar.

5.3.3 Diseño y funcionamiento del Módulo de Comunicación en el Sistema Consultante.

Cuando el usuario necesite realizar la consulta de un caso en específico, para el cual se ha creado la BC, sólo debe realizar la ejecución de ésta en el sistema consultante HAriesC. A partir de este momento el usuario interactuará con el sistema para proporcionar las especificaciones necesarias para evaluar la consulta.

El sistema HAriesC comienza la evaluación de un caso específico activando la máquina de inferencia de control. Traslado el control a las máquinas de inferencia de las distintas estructuras del medio ambiente HAries, de acuerdo a las necesidades de evaluación de las mismas.

En el momento que la máquina de inferencia requiere de la consulta a una base de datos se activa el módulo para el acceso a la fuente específica que se desea examinar, otorgando este los resultados necesarios para continuar con el proceso de evaluación de consulta definido.

Al activarse el procesamiento de las variables extracción de datos y cuantificador, el primer paso que ejecutan es el de realizar la conexión con la BD de donde obtendrán la información que necesitan para culminar su ejecución.

Este módulo para la comunicación con diversos DBMS, el cual es utilizado por la máquina de inferencia de dichas variables, proporciona los resultados requeridos por el sistema HAriesC durante la consulta de un caso en particular. Éste es el encargado de interactuar con el manejador de la BD en turno, logrando la comunicación necesaria para obtener la información requerida.

La interacción en la que se encuentra involucrado el módulo de comunicación durante la ejecución del sistema consultante HAriesC se muestra en la figura 5.6.

Al activarse la máquina de inferencia se ejecuta el módulo para la comunicación con la fuente de datos. Realizando éste como primer tarea la identificación de los parámetros correspondientes al DBMS que consultará.

En caso de que la conexión sea errónea se realiza la solicitud al usuario, de los parámetros necesarios para especificar la consulta que se desea realizar, activando para ello la interfaz para la conexión y una vez identificados los parámetros en dicha interfaz se procede con la consulta a la BD.

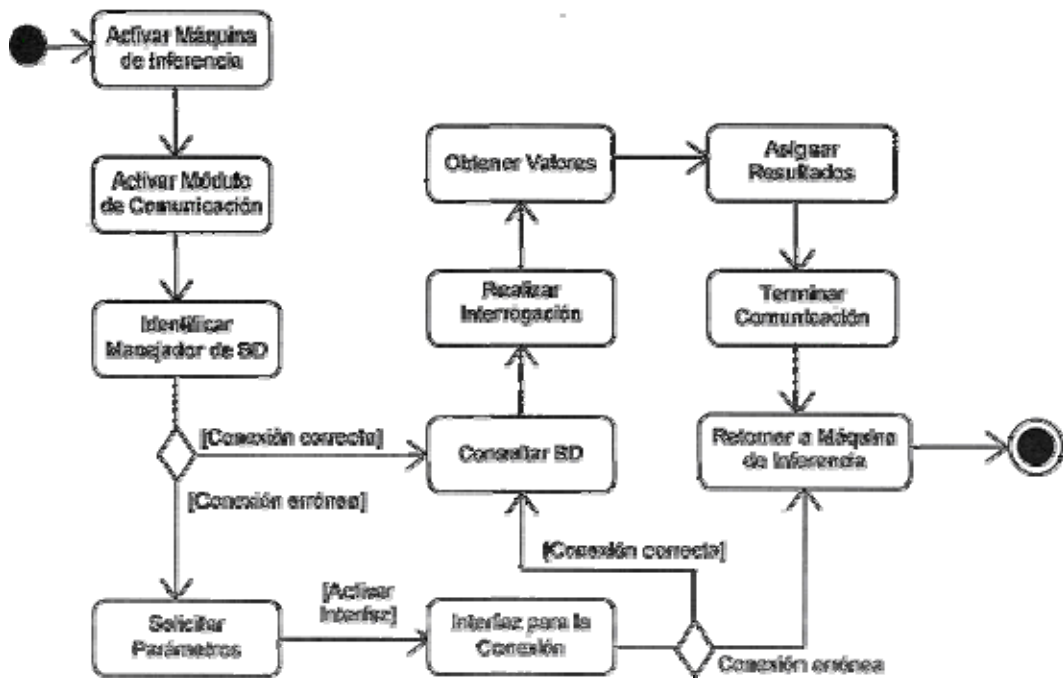


Figura 5.6. Interacción durante la consulta del Módulo de Comunicación.

Las variables realizan la interrogación a la BD en cuestión, utilizando también el módulo de comunicación y obtienen los valores necesarios para la asignación de los resultados correspondientes a las estructuras diseñadas para tal efecto, con esto termina la comunicación y se retorna el control a la máquina de inferencia correspondiente.

Al finalizar la consulta con la BD, cada una de las estructuras de las variables realiza el proceso de asignación de la información que han obtenido hacia las estructuras de la BC que se han especificado.

Para realizar el procesamiento de las estructura de las variables extracción de datos y cuantificador, se utiliza la clase ConexiónBD, definida en la jerarquía de clases del sistema HARiesC y que es mostrada en la figura 5.7, lo que cambia con respecto al sistema de adquisición son los métodos necesarios para implementar las operaciones relacionadas con el módulo de comunicación, qué será explicado en el siguiente epígrafe.

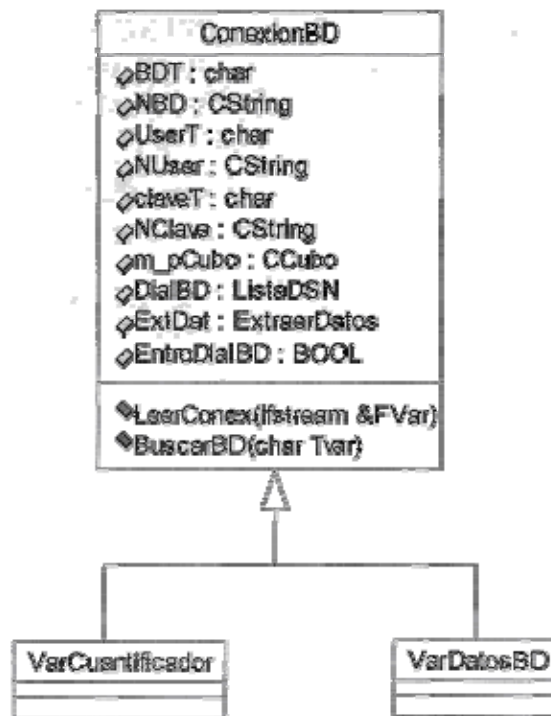


Figura 5.7. Diagrama de clases : Clase ConexionBD dentro de HArriesC.

5.3.3.1. Diseño del Módulo para la Comunicación con ODBC.

El módulo de comunicación con las BD está contenido en la DLL "AccesoBD.dll" la cual contiene las clases CCubo, CCuadro, CColumns, GenericSet y un conjunto de clases para manejar los tipos de datos, como se muestra en el diagrama de clases de la figura 5.8.

CCubo: Es la clase utilizada para realizar la extracción de la información en un cubo de datos, el cual contendrá un conjunto de cuadros con la información obtenida de las tablas de la BD.

CCuadro: Esta clase proporciona los métodos necesarios para realizar actualizaciones de registros, agregar registros, obtener el tipo de dato, modificar un dato, borrar un dato y obtener información de una tabla específica.

CColumns: Esta clase identifica las columnas de las tablas definidas en el catálogo que son accesibles para un usuario dado.

CGenericSet: Es utilizada para identificar el tipo de datos genérico que será consultado en la BD.

CtipoDato: Es la clase utilizada para implementar polimorfismo sobre los distintos tipos de datos.

Las clases CRFXText, CRFXLong, CRFXInt, CRFXDouble, CRFXSingle, CRFXBit, CRFXByte, CRFXBinary, CRFXLongBinary, CRFXDate y CRFXTimeStamp son las encargadas de proporcionar la manipulación de los distintos tipos de datos, que han sido especificadas en el epígrafe 5.3.1.2.

Todas estas clases proporcionan la funcionalidad requerida por las estructuras de la BC en cuanto a la consulta de una BD se refiere.

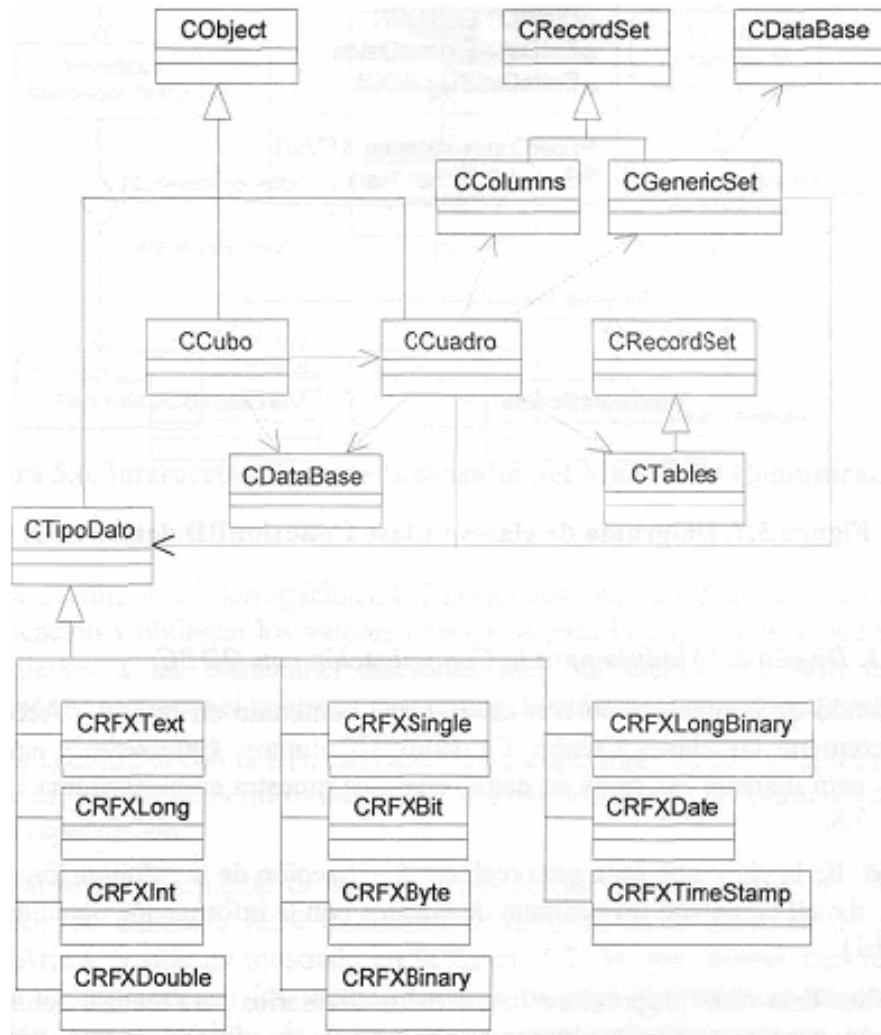


Figura 5.8. Diagrama de clases de la DLL para la conexión con la BD.

5.3.4. Ventajas del Uso del Ambiente de Comunicación.

Mediante el uso del medio ambiente para la comunicación se provee la comunicación con diversos DBMS.

Se hace posible la consulta a BD para obtener valores que necesita una BC determinada.

Los tipos de datos pueden ser tratados con un formato Standard independientemente de la fuente de datos de donde sean obtenidos.

Permite supervisar los errores que ocurran durante la comunicación.

Se puede definir la comunicación en tiempo de ejecución. Posibilita además, el acceso remoto a fuentes de datos.

Se posibilita el desarrollo de sistemas en los cuales la BD es variable e independiente de la aplicación.

Capítulo VI

Ejemplos de Aplicación de las Variables Extracción de Datos y Cuantificador.

6.1. Introducción.

En este capítulo se presentan diversos ejemplos de aplicaciones reales, en las que se utilizan las estructuras de representación de conocimiento implementadas en este trabajo, para darle solución a problemas de naturaleza muy diversa.

Los ejemplos que se presentan muestran la utilización de las representaciones del conocimiento desarrolladas para manipular BDs, desde diferentes BCs.

En cada aplicación se proporcionan recomendaciones como apoyo para la toma de decisiones en la BC, como resultado de las consultas realizadas a las BDs.

En cada ejemplo mostrado se explica la problemática a resolver, se presenta la preparación de construcción de las estructuras que garantizan la funcionalidad requerida y que serán incorporadas a la BC, así como ejemplos de los resultados que se obtienen con el sistema.

6.2. Aplicaciones Construidas.

Utilizando la teoría y los módulos construidos para el sistema HAries, se han desarrollado varias aplicaciones, que permiten validar el funcionamiento de dichos módulos.

Como se mencionaba, las aplicaciones desarrolladas se enmarcan en diversos dominios del conocimiento pero pueden dividirse en cuatro grupos fundamentales: aplicaciones en medicina, educación, los servicios y la industria.

Es importante mencionar que la mayoría de estas aplicaciones, fueron desarrolladas por alumnos de la Maestría en Ciencias Computacionales, impartida en el Centro de Investigación en Tecnologías de Información y Sistemas (CITIS), de la Universidad Autónoma del Estado de Hidalgo (UAEH). Las cuales se pueden consultar en el CD anexo a este documento.

6.2.1. Aplicaciones en Medicina.

Dado que las aplicaciones que se desarrollan para resolver problemas en el sector de salud están ligadas generalmente al uso de bases de datos y de la experiencia, estas son

situaciones típicas para el uso combinado de ambas técnicas. En este dominio de se han desarrollado dos sistemas:

Prototipo de un sistema para el pronóstico de pacientes enfermas de cáncer cérvico uterino.- Donde se pretende detectar zonas o regiones geográficas del estado de Hidalgo en las que se presente un incremento significativo en los niveles de la enfermedad para alertar a las autoridades de salud sobre la necesidad de incrementar jornadas de prevención y detección. Esta aplicación fue desarrollada por el autor de la presente tesis, y utilizada en la Unidad de Patología Cervical del Hospital General de Pachuca, para evaluar la jornada de prevención y detección de cáncer 2002.

Prototipo de un sistema para la evaluación del riesgo de SIDA.- En este caso se analizan los factores de riesgo fundamentales tomando en cuenta factores geográficos, para determinar los riesgos individuales que tiene una persona en contraer SIDA dado estos factores. Esta aplicación fue desarrollada por la alumna Dalia Erika Islas Pérez, como proyecto final de la materia Métodos Híbridos de Computación Inteligente.

6.2.2. Aplicaciones en Educación.

Se utiliza la teoría desarrollada como apoyo en la construcción de **sistemas educativos para matemática, física y química** [1], posibilitando el diagnóstico y evaluación de cada alumno de forma individual, o un grupo de alumnos en general.

En estos sistemas se logra detectar los temas con mayores dificultades o incluso, determinar las partes de los sistemas donde existen fallas en los métodos de enseñanza utilizados, lo cual facilita la toma de decisiones para mejorar el proceso de enseñanza aprendizaje.

En estos casos, el propio sistema va creando la BD a partir de la interacción que realiza con usuarios que lo consultan, para posteriormente utilizar la información contenida en ella para realizar diagnósticos individuales o colectivos.

6.2.3. Aplicaciones en los Servicios.

Algunas de las aplicaciones desarrolladas en este sentido son las siguientes:

Prototipo de un sistema para el control de la calidad en los servicios ofrecidos en el museo del niño el Rehilete.- Se analiza una BD con opiniones de los visitantes para encontrar las áreas de mayor interés o aquellas donde se necesita mejorar la calidad para hacerlas más atractivas. Esta aplicación fue desarrollada por la alumna Genoveva Ruiz Casañas, como proyecto final de la materia Métodos Híbridos de Computación Inteligente.

Prototipo de un sistema para el control del consumo de agua.- En este caso se monitorea el consumo de agua por regiones, para detectar, de acuerdo a las características de los usuarios, alguna fuga accidental o intencional en los consumos, y

actuar en consecuencia con ello. Esta aplicación fue desarrollada por la alumna Irasema de J. Juárez Guadalupe, como proyecto final de la materia Métodos Híbridos de Computación Inteligente.

Prototipo de un sistema para el control de autos robados.- Esta aplicación permite controlar y brindar avisos a los usuarios sobre diversos aspectos de una BD de autos robados, tanto por regiones geográficas como por las características de los autos (modelo, color, año, etc.). Este sistema ha permitido descubrir diversas formas de operación o incluso tendencias en los delitos, muchos de los cuales eran desconocidos anteriormente. Esta aplicación fue desarrollada por el alumno Jaime Asiaín López, como proyecto final de la materia Métodos Híbridos de Computación Inteligente.

6.2.4. Aplicaciones en la Industria.

Existen otras aplicaciones relacionadas con la industria, donde se han utilizado los módulos desarrollados para permitir la extracción de información. Como ejemplos de ello se encuentran:

Sistema para pronósticos en pozos de petróleo.- En este sistema se evalúa un pozo de petróleo para realizar la recomendación sobre la aplicación de tenso-activos para aumentar la producción [25]. El sistema es capaz de hacer el pronóstico deseado, además de sugerir la tecnología más apropiada en cada caso. Esta aplicación fue desarrollada por el Dr. Argelio V. de la Cruz Rivera.

Sistema experto para diagnóstico de transformadores.- Es un sistema para diagnóstico de grandes transformadores, basado en análisis de gases disueltos en el aceite, obtenidos mediante métodos cromatográficos [18]. Es capaz de diagnosticar fallos incipientes en los transformadores como pueden ser: sobrecalentamientos, contactos defectuosos, pérdidas de aislamiento, etc. Esta aplicación fue desarrollada por el Dr. Argelio V. de la Cruz Rivera.

Para estos sistemas, se brinda la posibilidad de extraer los casos a evaluar a partir de una BD suministrada por el usuario.

En estos casos, los sistemas presentan un buen nivel de ejecución donde la incorporación de los módulos desarrollados ha posibilitado la manipulación de BD existentes.

Como desde el punto de vista de nuestro trabajo, la utilización de la BC y las BDs son semejantes para ambos sistemas, tomaremos el primero para mostrar el uso de los módulos desarrollados.

6.3. Prototipo de un Sistema para el Pronóstico de Pacientes Enfermas de Cáncer Cérvico Uterino.

Los servicios de salud del estado de Hidalgo realizan jornadas de detección y prevención de Cáncer Cérvico Uterino (CaCu), las cuales consisten en practicar un estudio citológico a las pacientes que deseen prevenir dicha enfermedad. Las jornadas son realizadas en determinada región, municipio o municipios. Los resultados obtenidos en estos estudios son almacenados en una BD, que permite llevar el control sobre la evolución de la enfermedad.

Sin embargo, entre los problemas fundamentales a resolver se encontraba la toma de decisión adecuada para determinar los periodos de tiempo en que se deben realizar las jornadas de detección y prevención.

Para solucionar este problema, se desarrolló el prototipo de un sistema capaz de brindar sugerencias para tomar la decisión de la fecha en la que se debe realizar una jornada más en la región, municipio o municipios evaluados, a partir de un monitoreo a la BD que contiene los registros de las pacientes atendidas durante una determinada jornada de prevención y detección.

Dicha evaluación se realiza en base al número de pacientes que fueron atendidas y conforme al número de resultados positivos a CaCu o alguna enfermedad que predisponga cáncer. Esto es, si se tienen más o igual a un 9% de resultados positivos se deberá de realizar una jornada de prevención y detección más en la región o municipio a los seis meses de haber realizado la anterior. Por el contrario, si el porcentaje es menor al 9% se deberá de realizar una jornada de prevención y detección a los 18 meses de haberse realizado la anterior.

Dadas las posibilidades que tiene el lenguaje HArries para el desarrollo de aplicaciones que requieren de toma de decisiones y considerando además su nueva capacidad de interrogar BDs, se creó una BC para dar respuesta a esta situación.

Como primer aspecto a considerar es el hecho de poder cuantificar la cantidad de mujeres con resultados positivos, lo cual obliga a revisar los registros de la BD para conocer quienes cumplen con la condición.

Luego, además de definir otras estructuras de representación del conocimiento, la necesidad de consultar la BD exigía del empleo de una variable cuantificador.

En la BD se tiene una tabla en donde se registran los datos de las pacientes atendidas en el Estado de Hidalgo. A continuación se realizará la explicación de cómo se efectúa la evaluación para la Jurisdicción Tula.

Como se requiere conocer si un 9% o más de las pacientes de la Jurisdicción Tula tienen un resultado positivo a cáncer o una enfermedad que predispone cáncer se necesita de un cuantificador de tipo universal con manejo de valores porcentuales, esto debido a que se necesita establecer una relación entre las pacientes atendidas y el resultado citológico que tienen.

Se identifica el campo de la BD de donde será obtenida la información, en este caso la Jurisdicción Tula se ha representado en la BD con el número 3 y el resultado citológico de las pacientes en el campo R14, ambos en la tabla Pacientes.

El resultado citológico ha sido representado en la BD de la siguiente manera:

1. Negativo a Cáncer (Normal)
2. Negativo con proceso inflamatorio
3. Displasia Leve (NIC 1)
4. Displasia Moderada (NIC 2)
5. Displasia Grave (NIC 3)
6. Cáncer In situ
7. Cáncer invasor
8. Adenocarcinoma
9. Maligno no especificado

Para obtener el resultado requerido de la BD, se necesita una expresión que permita conocer cuantas pacientes atendidas en la Jurisdicción Tula tienen en su resultado citológico igual a uno de los identificados en la BD con los números 5, 6, 7, 8 y 9. Para ello se requiere de un cuantificador de tipo universal con una expresión $\varphi(x)$ para identificar a las pacientes que fueron atendidas en la Jurisdicción Tula y una expresión $\beta(x)$ para cuantificar a las pacientes atendidas en ésta Jurisdicción que tienen un resultado citológico igual al requerido.

Para manejar la información que se obtenga de la BD, en la BC se deberá tener una proposición asociada a la variable cuantificador, la proposición contendrá el valor de certidumbre que se tiene en relación al porcentaje de las pacientes que presentan diagnóstico positivo a cáncer o una enfermedad que predispone cáncer.

Los parámetros requeridos para la conexión a la BD de donde será obtenida la información se refieren al nombre, usuario y clave. Para el caso planteado la BD de donde se obtendrá la información tiene el nombre de "epicacu", el usuario autorizado para acceder "FOAM" y la clave de acceso correspondiente al usuario es "Cacu2000".

A partir de todo el planteamiento anterior, se necesita definir la variable cuantificador con la siguiente estructura:

$$V_{CI} = \langle T_{CI}, Vr_{CI} Pr_{CI} Atr_{CI} = \langle CBD_{CI}, PL_{CI}, Q_{CI} \rangle \rangle \text{ donde:}$$

T_{CI} = "Pacientes con lesión grave", texto necesario para identificar la variable.

Vr_{CI} = Datos obtenidos de la BD.

Pr_{CI} = "1", es el número de la proporción contenida en la BC que obtendrá el valor de certidumbre producto de la evaluación de la variable cuantificador.

Atr_{CI} identifica los atributos específicos como sigue:

CBD_{CI} = <Nombre, Usuario, Clave> donde:

Nombre: "epicacu", es el nombre de la BD en la cual se realizarán las operaciones de cuantificación.

Usuario: "FOAM", es el nombre de usuario autorizado para acceder a la BD.

Clave: "Cacu2000", es la clave de acceso correspondiente al usuario.

PL_{CI}= <Tabla de la Base de Datos = Múltiples, Campo Llave = Paciente>.

Q_{CI} = <CUP, Exacto, > 9%, <p(x):P(x)> donde:

$\varphi(x)$ = (Jurisdicción = 3)

$\beta(x)$ = (R14 = 5) v (R14 = 6) v (R14 = 7) V (R14 = 8) v (R14 = 9) Con

los siguientes parámetros:

1. Relación global: ">", puesto que se desea obtener como resultado si el número de registros de la BD que cumplen la condición son mayor o igual a una cantidad determinada.
2. Tipos de valores: "porcentual", ya que la relación se evaluará en términos del porcentaje de los registros que cumplen la relación $\varphi(x)$ en comparación con el total de los registros cuantificados.
3. Valor de comparación: "9", representa el umbral de comparación con los registros de la BD que cumplen la relación definida.
4. Tabla de la Base de Datos: "Múltiples", se elige esta opción debido a que los registros a cuantificar son contenidos por dos tablas de la BD.
5. Campo llave: "Paciente", correspondiente al campo con el que se guiará la búsqueda de los registros a cuantificar.
6. $\varphi(x)$: Se construye para considerar solamente a las pacientes que fueron atendidas en determinada jurisdicción, municipio o municipios, en este caso se logra reducir el dominio de la cuantificación que realizará nuestra variable a las pacientes a quienes se les practicó el estudio citológico en la jurisdicción Tula, la expresión es la siguiente:

(Jurisdicción = 3)

y tiene los siguientes parámetros:

- La relación $\varphi(x)$: "Conjunción", en este caso el uso de una relación conjunción o disyunción inclusiva es indistinto ya que se obtiene el mismo resultado debido a que la relación está compuesta de un solo elemento.
- El Miembro Izquierdo es "variable", con una identificación del campo igual a "Jurisdicción" que es el nombre del atributo de la tabla que contiene la

Jurisdicción en la que fue atendida la paciente y tabla de la variable igual a "pacientes", que es la tabla que contiene al campo "Jurisdicción".

- La relación particular es "=", ya que se desea saber si la paciente corresponde a la Jurisdicción evaluada.
- El miembro derecho es "constante" con el valor igual a 3, que es el número asignado en la BD a la Jurisdicción Tula.

7. $\beta(x)$: Su construcción se realizó con la finalidad de identificar a las pacientes dentro de la BD que tienen un resultado positivo a cáncer o alguna enfermedad que predisponga cáncer, la expresión final construida es:

$$(R14 = 5) \vee (R14 = 6) \vee (R14 = 7) \vee (R14 = 8) \vee (R14 = 9)$$

y tiene los siguientes parámetros:

- Relación $\beta(x)$: "Disyunción Inclusiva", ya que se desea evaluar si la paciente tiene alguno de los resultados citológicos registrados en la BD con los números 5(Displasia Grave -NIC 3-), 6(Cáncer In situ), 7(Cáncer invasor), 8(Adenocarcinoma) ó 9(Maligno no especificado).
- Miembro Izquierdo: "variable", con una identificación del campo igual a "R14" que es el nombre del atributo de la tabla que contiene las respuestas del estudio citológico y tabla de la variable igual a "RespuestasHorizontal", que es la tabla que contiene al campo "R14".
- La relación es "=", ya que se desea saber si la paciente tiene un resultado igual al deseado.
- Miembro Derecho es "constante", se compone de los posibles valores a considerar: 6, 7, 8 y 9, que como se había mencionado anteriormente, son los números asignados a las pacientes en la BD dependiendo del tipo de cáncer o enfermedad detectada en el estudio citológico.

Esta variable cuantificador se define en el sistema de adquisición de la variable, como se muestra en la figura 6.1, permitiendo generar la expresión requerida para obtener ya sea el número exacto de pacientes o un porcentaje determinado, en este caso se deberá declarar un cuantificador universal con un tipo de análisis exacto ya que se requiere saber si un 9% o más del total de las pacientes de una determinada jurisdicción, municipio o municipios tienen un resultado positivo.

La proposición central asociada a este cuantificador se puede expresar de la siguiente manera: "Un 9% o más de las pacientes atendidas en la Jurisdicción Tula presentan diagnóstico positivo a cáncer ó una enfermedad que predispone cáncer".

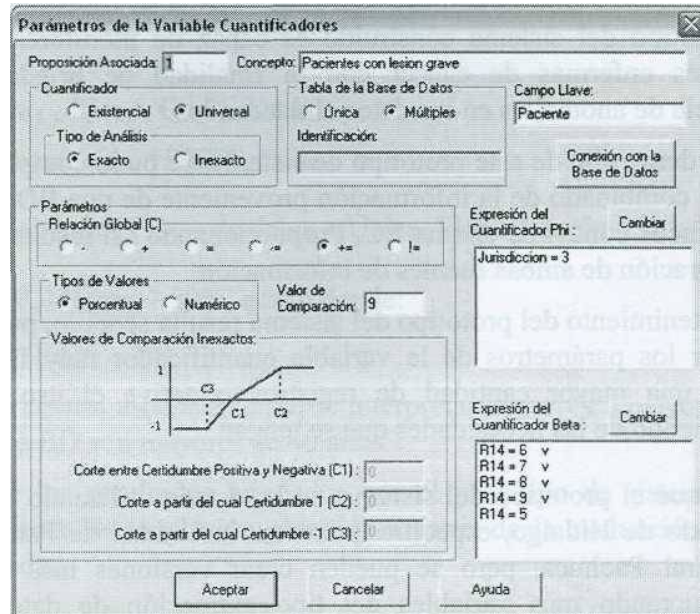


Figura 6.1 Adquisición de la variable cuantificador requerida para evaluar a las pacientes de la Jurisdicción Tula con un resultado positivo.

El trabajo del cuantificador consiste en obtener el valor de certidumbre asociado con la proposición central, cuyo valor es utilizado en la BC para la toma de decisiones. Un ejemplo de las conclusiones brindadas, por el prototipo del sistema que se desarrolló, se muestra en la figura 6.2.

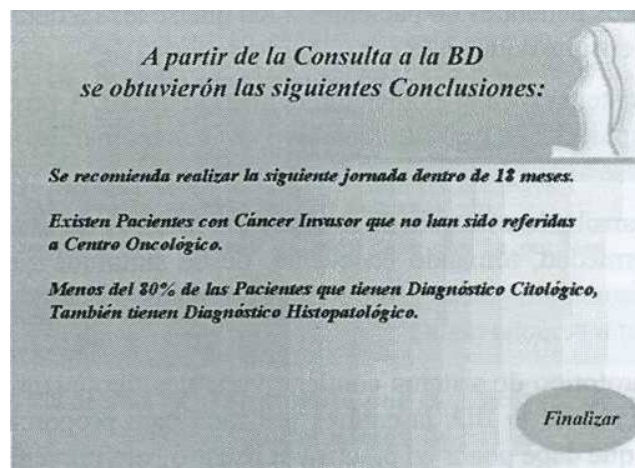


Figura 6.2 Un resultado obtenido por el Prototipo de Sistema para el Pronóstico de Pacientes Enfermas de Cáncer Cérvico Uterino.

En resumen, se pueden plantear las siguientes conclusiones para este prototipo de sistema.

1. El prototipo del sistema construido es capaz de monitorear cualquier BD de pacientes enfermas de cáncer con la finalidad de brindar avisos sobre la presencia de anomalías en los datos almacenados.
2. Con el desarrollo de este prototipo de sistema, se pudo comprobar la factibilidad del uso combinado de la información proveniente de una BD y la experiencia de los médicos contenida en una BC, Proporcionando así resultados que involucran la integración de ambas fuentes de información
3. El mantenimiento del prototipo del sistema resulta sencillo, puesto que se pueden cambiar los parámetros de la variable cuantificador muy fácilmente para que evalúe una mayor cantidad de registros o active el uso de otros campos, dependiendo de las necesidades que se tengan.

Es de señalar que el prototipo del sistema, se ha estado utilizando en los Servicios de Salud del Estado de Hidalgo, específicamente en la Unidad de Patología Cervical del Hospital General Pachuca, pero se pueden crear versiones más completas de éste sistema, involucrando más variables del tipo extracción de datos y cuantificador, otorgando resultados a medida de las necesidades que se presenten en un futuro.

6.4. Prototipo de un Sistema para la Evaluación del Riesgo de SIDA.

Los servicios de salud del estado de Hidalgo cuentan con una BD en la que se han registrado los datos generales de pacientes a los que se les ha detectado SIDA (Síndrome de Inmunodeficiencia Adquirida).

El prototipo de sistema para la evaluación del riesgo de SIDA proporciona a sus usuarios un informe acerca del riesgo de contagio y la recomendación de un método de prevención en base al conjunto de sus características.

El prototipo desarrollado es capaz de analizar los factores de riesgo fundamentales para contraer la enfermedad, tomando en cuenta, desde síntomas que presenta el paciente durante la enfermedad, hasta factores geográficos, para determinar los riesgos individuales de una persona dada.

La BC de este prototipo de sistema contiene variables tipo cuantificador utilizadas para obtener información de la BD, la cual es utilizada para proporcionar recomendaciones sobre el método que debe poner en práctica el usuario para prevenir un posible contagio.

Un ejemplo de variable cuantificador definida en el sistema es aquella utilizada para saber si en la BD se ha registrado un número mayor del 50% de pacientes con 40 o más años de edad, la cual contiene la siguiente estructura:

$$V_{C1} = \langle TC1, Vr_{C1}, Pr_{C1}, Atr_{C1} = \langle CBD_{C1}, PL_{C1} Q_{C1} \rangle \rangle \text{ donde}$$

T_{C1} = "Pacientes con 40 o más años de edad".

Vr_{C1} = Datos obtenidos de la BD.

Pr_{C1} = "6".

Atr_{C1} = $\langle CBD_{C1}, PL_{C1}, Q_{C1} \rangle$ donde:

CBD_{C1} = $\langle base_sida, "DEIP", "sida" \rangle$

PL_{C1} = \langle Tabla de la Base de datos = Única, Identificación = sida \rangle .

Q_{C1} = \langle CEP, Exacto, $>50\%$, $\varphi(x)$ \rangle donde:

$$\varphi(x) = (Edad > 40)$$

La proposición central asociada se puede interpretar como: "La mayoría de los pacientes registrados en la BD son mayores de 40 años".

Un ejemplo de los resultados brindados por el prototipo de sistema, mostrado en la figura 6.3, presenta la conclusión obtenida a partir de la evaluación de la proposición antes mencionada.

Este prototipo sirve como apoyo para el tratamiento de la enfermedad, se puede orientar a la contribución de una prevención eficaz con la finalidad de aumentar la aceptación de la prueba de detección y la importancia del apoyo psicológico de personas voluntarias y confidenciales.

El prototipo de sistema puede complementarse para que sea capaz de mantener en estrecho contacto a las personas que viven con SIDA y a los grupos vulnerables, con los sistemas de atención de salud y con ello facilitarles el acceso a la información, al apoyo psicológico y los medios de prevención.

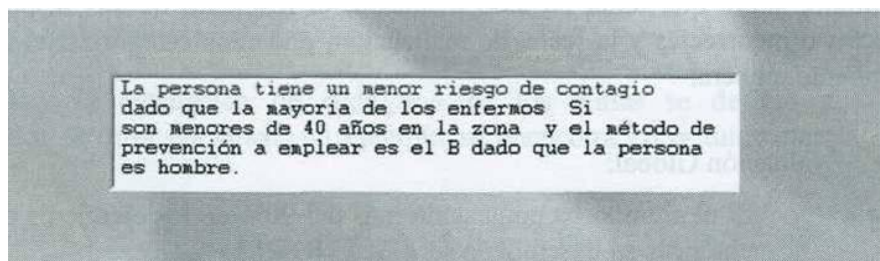


Figura 6.3 Un resultado proporcionado por el Prototipo de Sistema para la Evaluación del Riesgo de SIDA.

6.5. Sistemas Educativos.

Entre los aspectos fundamentales de un sistema educativo, se encuentra su capacidad para dar seguimiento a la evolución del aprendizaje de uno o varios alumnos.

Actualmente en nuestro centro se encuentran en desarrollo diversos sistemas educativos, los cuales han comenzado a incorporar el uso de las variables del tipo BD para lograr este objetivo.

Un ejemplo es el sistema MatVir3, para la enseñanza de la trigonometría, del cual tomaremos un ejemplo

Durante el desarrollo de sistema se identificó la necesidad de informar al alumno, el cual es usuario del sistema, sobre el desempeño que ha tenido durante una ejercitación en particular, esto con la finalidad de brindar recomendaciones que puedan ser útiles al alumno para mejorar su aprendizaje, ejercitando las áreas en la que no ha mostrado un buen desempeño.

Dentro de este sistema se planteó la evaluación del desempeño del alumno basándose en una clasificación dentro de alguna de las siguientes categorías:

- Bueno
- Regular
- Malo

Esta clasificación se utiliza para evaluar el desempeño del alumno tanto desde un enfoque global como de algún tema en específico, ya que en ocasiones es necesario para el alumno saber que tan bueno ha sido su desempeño desde varias perspectivas de evaluación.

Partiendo entonces de la consulta a una BD que contiene los ejercicios que ha realizado el alumno, sobre que tema ha sido evaluado, el resultado de sus respuestas ya sean correctas o incorrectas y la fecha de realización; podemos categorizar el desempeño de la siguiente manera:

- Evaluación Global:
 - o Si el alumno ha contestado más del 90% de los ejercicios correctamente, entonces su desempeño será BUENO
 - o Si el alumno ha contestado entre el 50% y 90% de los ejercicios correctamente, entonces su desempeño será REGULAR
 - o Si el alumno ha contestado menos del 50% de los ejercicios correctamente, entonces su desempeño será MALO

➤ Evaluación por temas:

- o Se clasifica de la misma manera que la evaluación global, pero en este caso, solo se tomarán en cuenta los ejercicios correctamente contestados del tema sobre el que se desea evaluar su desempeño.

La BC del sistema tutorial MatVir3 implementa estas opciones, utilizando variables del tipo cuantificador, para consultar la BD que contiene los registros del desempeño de los alumnos que han utilizado el sistema tutorial.

Se destaca que para la evaluación global sólo son necesarias variables cuantificador de tipo existencial, y para la evaluación por temas la variable cuantificador de tipo universal.

Así por ejemplo, para evaluar una ejercitación de forma global se define una variable cuantificador de tipo existencial.

La expresión de la variable cuantificador construida es la siguiente:

$$V_{C1} = \langle T_{C1}, Vr_{C1}, Pr_{C1}, Atr_{C1} = \langle CBD_{C1}, PL_{C1}, Q_{C1} \rangle \rangle \text{ donde}$$

T_{C1} = "Análisis del porcentaje de ejercicios correctamente contestados".

Vr_{C1} = Datos obtenidos de la BD.

Pr_{C1} = "3".

$Atr_{C1} = \langle CBD_{C1}, PL_{C1}, Q_{C1} \rangle$ donde:

$CBD_{C1} = \langle \text{alumnos, "KJMH", "matem-3"} \rangle$.

$PL_{C1} = \langle \text{Tabla de la Base de datos = Única, Identificación = ejercitación} \rangle$.

$Q_{C1} = \langle \text{CEP, Exacto, } >90\%, \varphi(x) \rangle$ donde: $\varphi(x)$
= (respuesta = correcta)

Para realizar la evaluación de cada uno de los temas se definió una variable cuantificador de tipo universal, un ejemplo se encuentra constituido por la siguiente expresión:

$$V_{C2} = \langle T_{C2}, Vr_{C2}, Pr_{C2}, Atr_{C2} = \langle CBD_{C2}, PL_{C2}, Q_{C2} \rangle \rangle \text{ donde}$$

T_{C2} = "Análisis del porcentaje de ejercicios del tema 1 (90%)".

Vr_{C2} = Datos obtenidos de la BD.

Pr_{C2} = "11".

Atr_{C2} identifica los atributos específicos como sigue:

$CBD_{C2} = \langle \text{alumnos, "KJMH", "matem-3"} \rangle$.

$PL_{C2} = \langle \text{Tabla de la Base de datos} = \text{Única, Identificación} = \text{ejercitacion} \rangle$.

$QC_2 = \langle \text{CUP, Exacto, } > 90\%, \varphi(x):\beta(x) \rangle$ donde:

$\varphi(x) = (\text{id_tema} = 1)$

$\beta(x) = (\text{respuesta} = \text{correcta})$

De esta forma el sistema tutorial MatVir3 es capaz de brindar la evaluación de los alumnos que lo consultan, proporcionando un servicio más completo y especializado. Un ejemplo de evaluación se muestra en la figura 6.4.

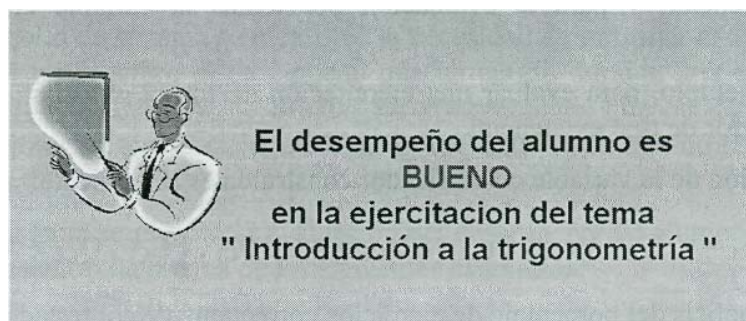


Figura 6.4 Un resultado de la evaluación en el tema "Introducción a la Trigonometría".

Los alumnos tienen la posibilidad de evaluar los conocimientos que poseen acerca de un tema en particular o de forma general.

El sistema es capaz de proporcionar una evaluación basada no solo en una consulta si no en un conjunto de consultas que se hayan realizado con la finalidad de verificar la evolución y nivel de conocimiento que ha logrado el alumno.

6.6. Prototipo de un Sistema para el Control de la Calidad en los Servicios Ofrecidos en el Museo "El Rehilete".

Con el objeto de mejorar la calidad de los servicios ofrecidos en el museo "El Rehilete" de la ciudad de Pachuca, Hidalgo, se aplican encuestas para que el visitante de a conocer su opinión, las cuales son capturadas en una BD.

Tomando como base estos datos, se desarrolló un prototipo de sistema para el monitoreo de la calidad en los servicios y la detección de las áreas con mayores problemas.

Existen diversas formas de enfocar el problema y por tanto la búsqueda. Si pensamos por ejemplo en la relación entre visitantes que entran al museo y los que visitan un área determinada, se puede detectar aquellos lugares de menor interés general o también al contrario cuales son las más atractivas en correspondencia con las edades de los visitantes.

Un ejemplo sencillo entre los utilizados en el prototipo de sistema, es la evaluación o apreciación global de los niños sobre el servicio general que brinda el museo, para ello se define una variable cuantificador que tiene la estructura siguiente:

$$V_{C1} = \langle T_{C1}, Vr_{C1}, Pr_{C1}, Atr_{C1} = \langle CBD_{C1}, PLC1, QC1 \rangle \rangle \text{ donde}$$

T_{C1} = "Al niño le parece bueno". Vr_{C1}

= Datos obtenidos de la BD. Pr_{C1} =

"8".

$Atr_{C1} = \langle CBD_{C1}, PLC1, QC1 \rangle$ donde:

$CBD_{C1} = \langle \text{rehilete, "ruizg", "niño-rehilete"} \rangle$.

$PLC1 = \langle \text{Tabla de la Base de datos = Única, Identificación = visitantes} \rangle$.

$QC1 = \langle \text{CEP, Exacto, } >90\%, \varphi(x) \rangle$ donde:

$\varphi(x) = (\text{tipo_visitante} = \text{niño}) \ \& \ (\text{servicio_adecuado} = \text{bueno})$

La proposición central asociada a esta variable se puede interpretar como: "La mayoría de los niños que visitan el museo consideran que el servicio es bueno".

Ha quedado evidenciado, que el prototipo de sistema para el control de la calidad en los servicios ofrecidos en el museo proporciona resultados fáciles de interpretar, apoyando así a la toma de decisiones, uno de los resultados que se pueden obtener con el prototipo de sistema es mostrado en la figura 6.5. El FC utilizado para la presentación de éste resultado es igual a 100.

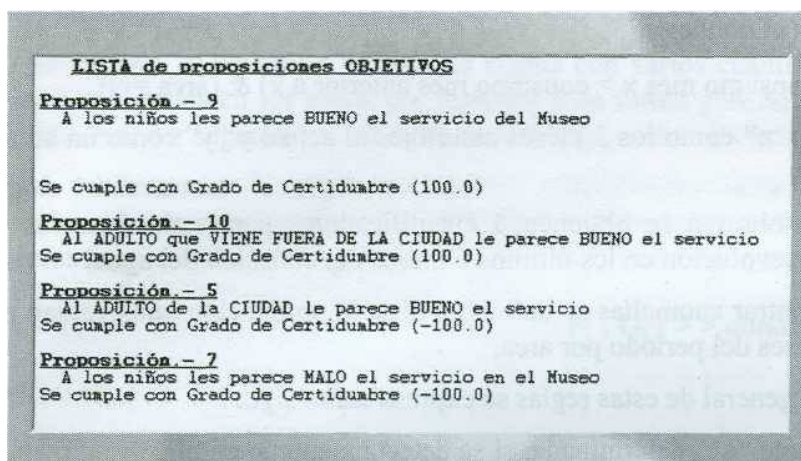


Figura 6.5 Un resultado de la evaluación de los servicios que ofrece el museo.

Además el prototipo de sistema es capaz de suministrar la posibilidad de monitorear constantemente la calidad de los servicios que éste ofrece en base a la opinión de los visitantes al Museo.

Con su utilización se pueden detectar incluso aquellas áreas de mayor o menor aceptación para actuar en consecuencia con ello.

6.7. Prototipo de un Sistema para el Control del Consumo de Agua.

Se tiene una BD con información general de los usuarios consumidores del servicio de agua potable, así como el consumo de cada usuario en al menos los 6 últimos períodos para esta tarea.

El objetivo que se busca es determinar el incremento en el consumo de los usuarios y verificar si este es adecuado, según las características de los mismos.

Para esto, necesitamos extraer información de la BD, esto lo hacemos a través de las variables tipo cuantificador definidas en el HArises.

En el HArises se realizó la definición de las variables cuantificador, utilizadas por este prototipo de sistema, así como también, las proposiciones asociadas correspondientes que son las estructuras de la BC que contendrán el valor de certidumbre producto de la evaluación de un caso en particular.

Para determinar si se tiene incremento del consumo, se utilizan variables tipo cuantificador existencial, en las que se evalúa si más del 50% de los usuarios cumplen con las condiciones de incremento de un mes con respecto al anterior.

La expresión de la variable cuantificador construida es la siguiente:

$\exists^{>50\%} x \varphi(x)$ donde

$\varphi(x) = (\text{consumo mes } x > \text{consumo mes anterior a } x) \ \& \ (\text{área} = y)$

Se define "x" como los 5 meses anteriores al actual y "y" como un área determinada de la región.

Como se observa se obtienen 5 cuantificadores para cada área, los cuales permiten obtener la evolución en los últimos 5 meses del consumo del agua.

Para encontrar anomalías se definen reglas de producción que evalúan un aumento en 1 o más meses del periodo por área.

La forma general de estas reglas se expresa como sigue:

Para un área "al" determinada, si se detecta aumento en el consumo en más del 50% de los clientes del área "al" durante un mes entonces brindar un aviso preventivo.

Pero si por el contrario se detecta este mismo comportamiento en dos meses o más, entonces brindar una alerta de anomalía en el consumo.

Como se puede observar este monitoreo, junto con otros factores que se incorporan, permite pronosticar diversas causas del incremento, como son: roturas, alteración de los medidores, robos, etc. Un ejemplo de los resultados que otorga el prototipo de sistema se muestra en la figura 6.6. El FC utilizado para la presentación de éste resultado es igual a 100.

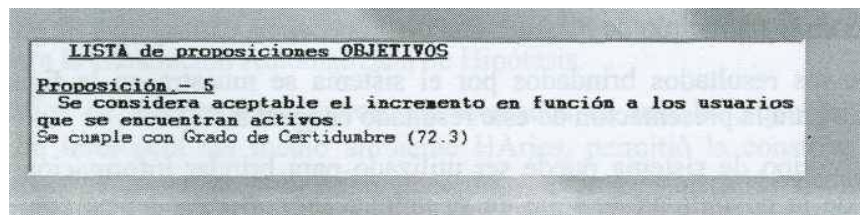


Figura 6.6 Un resultado otorgado por el Prototipo de Sistema para el Control del Consumo de Agua.

6.8. Prototipo de Sistema para el Control de Autos Robados.

El prototipo de sistema para el control de autos robados tiene como objetivo fundamental: obtener información de una BD para identificar automóviles robados bajo las condiciones de Municipio, Nacionalidad, Color, Año Modelo, además de porcentajes de robo.

Con este prototipo de sistema se puede proporcionar una evaluación de la BD con la finalidad de presentar las características de los autos que tienen un mayor índice de robo en determinado municipio.

Para efectuar su análisis, el prototipo de sistema cuenta con varios cuantificadores que permiten obtener tendencias en los robos por regiones o de forma general, para a partir de estos resultados brindar sugerencias sobre las acciones que deben tomarse.

Como ejemplo analizaremos la siguiente variable cuantificador utilizada por este prototipo de sistema y que es requerida para cuantificar el robo de autos de procedencia nacional, de color negro y que están registrados en el municipio de Pachuca:

$$V_{C1} = \langle T_{C1}, Vr_{C1}, Pr_{C1}, Atr_{C1} = \langle CBD_{C1}, PL_{C1}, Q_{C1} \rangle \rangle \text{ donde}$$

T_{C1} = "Cuantificador para Pachuca".

Vr_{C1} = Datos obtenidos de la BD.

Pr_{C1} = "8".

Atr_{C1} identifica los atributos específicos como sigue:

$$CBD_{C1} = \langle \text{robos}, \text{"asiainj"}, \text{"AutosRobo"} \rangle.$$

$PL_{C1} = \langle \text{Tabla de la Base de datos} = \text{Única},$

Identificación = ms_parque>. $Q_{C1} = \langle \text{CUP, Exacto},$

$>70\%, \varphi(x):\beta(x)\rangle$ donde:

$\varphi(x) = (\text{id_municipio} = 48) \ \& \ (\text{procedencia} = \text{N}) \ \& \ (\text{color} = \text{NEGRO})$

$\beta(x) = (\text{robo} = 1)$

La proposición central asociada se puede interpretar como: "la mayoría de los autos robados en el municipio de Pachuca son de procedencia extranjera y de color negro".

Uno de los resultados brindados por el sistema se muestra en la figura 6.7. El FC utilizado para la presentación de éste resultado es igual a 100.

Este prototipo de sistema puede ser utilizado para brindar información a los usuarios acerca de la posibilidad de robo de su unidad, en correspondencia con la información registrada en la BD, así como recomendaciones para la prevención de robos.

El prototipo de sistema también puede ser utilizado para brindar apoyo a la toma de decisiones para aseguradoras de automóviles, para que dependiendo de un análisis al registro de autos robados se brinden opciones de aseguramiento a los clientes.

Es importante mencionar que esta BD se encuentra diseñada con Informix y cuenta con más de 300,000 registros.

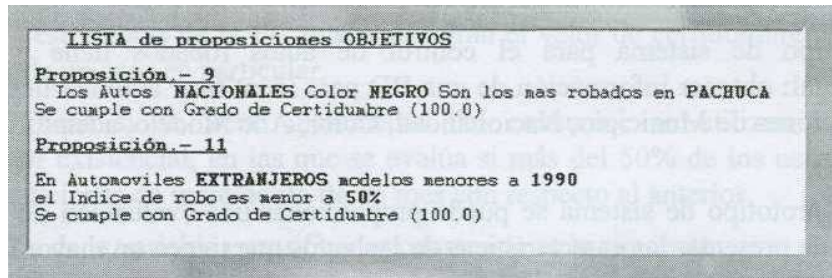


Figura 6.7 Un resultado brindado por el Sistema para el Control de Autos Robados.

En esta línea se pueden mencionar otras posibles aplicaciones, como por ejemplo, un sistema que sea capaz de proporcionar al usuario la mejor opción para el aseguramiento de su automóvil a partir de la consulta de diversos proveedores de seguros.

Algunos de los beneficios que podría obtener el cliente serían: el servicio de búsqueda de mejor producto de acuerdo con la relación calidad/precio establecida por el cliente sin tener que invertir tiempo en solicitar multitud de presupuestos a las diferentes compañías. Comparación de múltiples productos de seguros de forma totalmente imparcial según los criterios seleccionados por el usuario. Total libertad, control y privacidad, sin tener la presión de contratar el servicio que el sistema le recomiende adquirir. Asesoramiento en todos los aspectos relacionados con el seguro, sus derechos y sus obligaciones. Servicio de respuesta a las consultas más frecuentes. Fácil acceso a información de valor.

6.9. Sistema para Pronósticos en Pozos de Petróleo.

El sistema inteligente para pronósticos en yacimientos de petróleo fue construido utilizando los resultados del procesamiento de los datos obtenidos de un experimento, consistente en la inyección de tenso-activos en varios pozos de un yacimiento de petróleo [25].

La información obtenida del experimento fue procesada utilizando diversas técnicas exploratorias de datos, como son: Reconocimiento de Patrones, Selección de Variables y Métodos para la Generación Automatizada de Hipótesis.

Los resultados de este proceso fueron representados en una BC, que junto con las máquinas de inferencia del medio ambiente HArries, permitió la construcción de un sistema capaz de hacer el pronóstico deseado y además sugerir la tecnología más apropiada en cada caso. Este sistema además de pronosticar el efecto, también brinda sugerencias generales sobre la tecnología a utilizar y los parámetros del tenso-activo a producir. El sistema se aplicó a diversos pozos y el resultado en todos los casos fue exitoso.

Este sistema además de pronosticar el efecto, también brinda sugerencias generales sobre la tecnología a utilizar y los parámetros del tenso-activo a producir. En la figura 6.8 se muestra un ejemplo de la salida brindada por el sistema para uno de los pozos con efecto positivo.

La construcción de la BC se basó en las estructuras para la representación del conocimiento disponibles en el sistema HArries [24], [26], [27]. Las estructuras fundamentales fueron las proposiciones, para representar los conceptos necesarios, las reglas de producción para representar relaciones de inferencia entre las proposiciones y reglas de conclusión para definir los resultados que debe brindar el sistema.

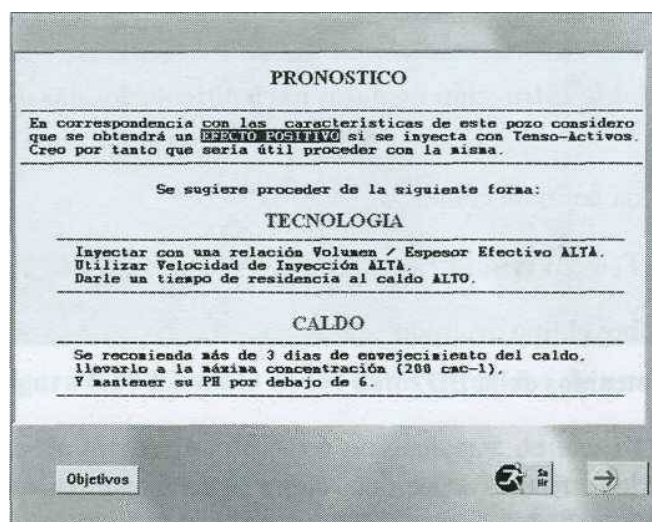


Figura 6.8 Ejemplo de una pantalla de conclusiones brindada por el Sistema para Pronósticos en Pozos de Petróleo.

El problema en nuestro caso consistió en incorporarle al sistema desarrollado, la posibilidad de extraer los datos directamente de una BD proporcionada por el usuario del sistema.

Para lograr esto se diseñaron dos variables del tipo extracción de datos que permiten establecer la relación entre las variables de la BC y los campos de la BD.

La primera variable se define para obtener los datos relacionados con el tipo de crudo, como se muestra en la figura 6.9.

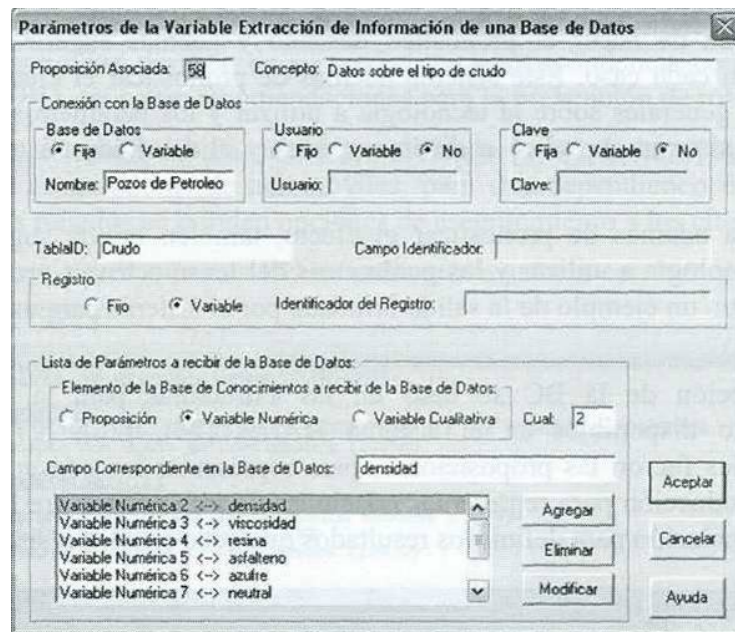


Figura 6.9 Variable extracción de datos para obtener los datos relacionados con el tipo de crudo.

Esta variable queda definida como:

$$V_{ED1} = \langle T_{ED1}, Vr_{ED1}, Pr_{ED1}, Atr_{ED1} = \langle CBD_{ED1}, PL_{ED1}, R_{ED1} \rangle \rangle \text{ donde:}$$

T_{ED1} - "Datos sobre el tipo de crudo".

Vr_{ED1} = Datos obtenidos de la BD.

Pr_{ED1} = 58.

Atr_{ED1} identifica los atributos específicos como sigue:

CBD_{ED1} = Abierta. La conexión se dejó abierta ya que su definición es establecida en tiempo de ejecución en interacción con el usuario.

PL_{EDj} = <TablaID = Crudo>.

R_{EDJ} = Lista de parámetros a recibir de la BD, especificada de la siguiente manera:

- Variable Numérica 2: Densidad.
- Variable Numérica 3: Viscosidad.
- Variable Numérica 4: Contenido de Resina.
- Variable Numérica 5: Contenido de Asfalteno.
- Variable Numérica 6: Contenido de Azufre.
- Variable Numérica 7: Número de Neutralización.
- Variable Numérica 8: Humectabilidad.
- Variable Numérica 9: Compatibilidad.
- Variable Numérica 10: Producción original.

La segunda variable se define para extraer los datos relacionados con la geología y se encuentra organizada de forma similar a la primera.

Se debe tener en cuenta que los datos obtenidos de la BD y que serán asignados a las estructuras de la BC corresponden al pozo que se este evaluando en el caso particular que se haya elegido. Un ejemplo de los datos extraídos de una BD para la evaluación de un pozo, se muestra en la figura 6.10.

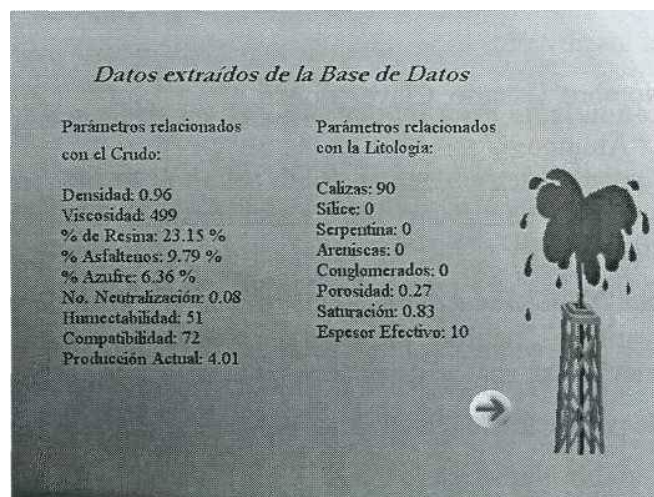


Figura 6.10 Ejemplo de los datos extraídos de una BD.

Todo este proceso de asignación de datos provenientes de una BD a estructuras de la BC, posibilitan una evaluación más rápida de los casos en los que se puede realizar la consulta a la BC, puesto que el usuario ya no tiene que proporcionarlos de una forma manual, aunque han quedado abiertas ambas variantes a elección del usuario.

6.10. Ejemplo para la consulta a una BD de Alumnos.

Se tiene una DB con información referente a alumnos y se requiere construir un sistema que apoye la toma de decisiones, a partir de la consulta a los datos contenidos en ésta.

Se necesita que el sistema proporcione las calificaciones de un alumno determinado y el promedio que ha obtenido. También se requiere, cuantificar los registros para verificar si más del 60% de ellos han aprobado un determinado número de materias. Además, se requiere conocer si los grupos han aprobado sus materias en un porcentaje mayor al 90%.

Para éste ejemplo se trabajará con tres grupos (123, 124 e 125), y tres materias Informática, Redes y Programación (INF-21, INF-22 e INF-23).

En cuanto a la consulta que se necesita para controlar el promedio de calificaciones, se definió una variable Extracción de datos para obtener la calificación que obtuvo un alumno determinado en las tres materias antes citadas.

Esta variable queda definida como:

$V_{ED1} = \langle T_{ED1}, Vr_{ED1}, Pr_{ED1}, Atr_{ED1} = \langle CBD_{ED1}, PL_{ED1}, R_{ED1} \rangle \rangle$ donde:

T_{ED1} = "Extracción de Calificaciones".

Vr_{ED1} - Datos obtenidos de la BD.

PL_{ED1} = 7, proposición utilizada para activar la consulta a la BD.

Atr_{EDJ} identifica los atributos específicos como sigue:

CBD_{ED1} = <Nombre, Usuario, Clave> donde:

Nombre: "Alumnos".

Usuario: "Marco".

Clave: "mark77"

$PLEDJ$ = <TablaID = Calificaciones, Registro: Variable>. El parámetro "Registro" con el atributo "variable" permite al usuario seleccionar en tiempo de ejecución al Alumno (Matricula) del cual se desea obtener los datos.

R_{EDJ} = Lista de parámetros a recibir de la BD, especificada de la siguiente manera:

Variable Numérica 8: INF-21.

Variable Numérica 9: INF-22.

Variable Numérica 10: INF-23.

La tabla de la BD a consultar por esta variable se muestra en la figura 6.11, en la que se encuentran los datos de los alumnos correspondientes a las calificaciones obtenidas en las materias que cursan.

	Matricula	INF-21	INF-22	INF-23
▶ +	2021066	6	5	7
+	2021068	8	3	6
+	2021099	5	4	8
+	2021128	6	5	7
+	2021159	8	9	9
+	2021162	9	7	9
+	2021165	9	5	7
+	2021182	6	5	6
+	2021184	6	8	9
+	2021187	9	3	7

Registro: 1

Figura 6.11 Tabla *Calificaciones* de la BD *Alumnos*.

Un ejemplo de los resultados obtenidos por esta variable, se muestran en la figura 6.12.

Materia	Calificación
Informática:	8
Redes:	5
Programación:	8
Promedio:	7

Figura 6.12 Ejemplo de los datos obtenidos para un alumno determinado.

Para conocer si más del 60% de los alumnos que cursan la materia de informática se requirió definir una variable cuantificador como se describe a continuación:

$V_{C1} = \langle T_{C1}, Vr_{C1}, Pr_{C1}, Atr_{C1} = \langle CBD_{C1}, PL_{C1}, Q_{C1} \rangle \rangle$ donde:

T_{C1} = "Más del 60 % de los alumnos en los grupos 123, 124 e 125 aprobaron Informática", texto necesario para identificar la variable.

Vr_{C1} = Datos obtenidos de la BD.

Pr_{C1} = "1", es el número de la proporción contenida en la BC que obtendrá el valor de certidumbre producto de la evaluación de la variable cuantificador.

Atr_{C1} identifica los atributos específicos como sigue:

$CBD_{C1} = \langle \text{Nombre, Usuario, Clave} \rangle$ donde:

Nombre: "Alumnos", es el nombre de la BD en la cual se realizarán las operaciones de cuantificación.

Usuario: "Marco", es el nombre de usuario autorizado para acceder a la BD.

Clave: "mark77", es la clave de acceso correspondiente al usuario.

$PL_{C1} = \langle \text{Tabla de la Base de Datos} = \text{Múltiples}, \text{Campo Llave} = \text{Matricula} \rangle$.

$Q_{C1} = \langle \text{CUP}, \text{Exacto}, > 60\%, \varphi(x):\beta(x) \rangle$ donde:

$\varphi(x) = (\text{clvGrupo} = \text{I23}) \vee (\text{clvGrupo} = \text{I24}) \vee (\text{clvGrupo} = \text{I25})$

$\beta(x) = (\text{INF-21} \geq 7)$

Con los siguientes parámetros:

1. Relación global: ">", puesto que se desea obtener como resultado si el número de registros de la BD que cumplen la condición son mayor a una cantidad determinada.
2. Tipos de valores: "porcentual", ya que la relación se evaluará en términos del porcentaje de los registros que cumplen la relación $\varphi(x)$, en comparación con el total de los registros cuantificados.
3. Valor de comparación: "60", representa el umbral de comparación con los registros de la BD que cumplen la relación definida.
4. Tabla de la Base de Datos: "Múltiples", se elige esta opción debido a que los registros a cuantificar son contenidos por dos tablas de la BD.
5. Campo llave: "Matricula", correspondiente al campo con el que se guiará la búsqueda de los registros a cuantificar y que se encuentra en las tablas requeridas para la consulta.
6. $\varphi(x)$: Se construye para considerar solamente a los alumnos de los grupos I23, I24 e I25, que son los grupos a cuantificar y tiene los siguientes parámetros:
 - La relación $\varphi(x)$: "Disyunción Inclusiva", ya que se desea evaluar a los alumnos de los tres grupos.
 - El Miembro Izquierdo es "variable", con una identificación del campo igual a "clvGrupo" que es el nombre del atributo de la tabla que contiene la clave del grupo al que pertenece el alumno y la tabla de la variable igual a "Alumnos", que es la tabla que contiene al campo "clvGrupo", como se puede ver en la figura 6.13.
 - La relación particular es "=", ya que se desea saber si el alumno corresponde al grupo evaluado.
 - El miembro derecho es "constante" con el valor igual a I23, I24 e I25, respectivamente y en correspondencia a los números asignados en la BD a los grupos que se desean evaluar.

	Matricula	Nombre	Edad	CivGrupo
+	2021245	Aguilar Heras Elizabeth Delfina	20	I23
+	2061319	Aguilar Islas Nancy	20	I24
+	2061320	Alarcón Flores América	19	I24
+	2061321	Alarcón Villalpa José Luis	19	I24
+	2061322	Angelas Hernández José Anahi Ma	19	I24
+	2061323	Atilano Licono Hugo David	19	I24
+	2061324	Avila Martínez Juana	19	I24
+	2021228	Barranco Quintero Edgar	19	I23
+	2021248	Barrera Gayosso Gelacia	19	I23
+	2061349	Bustamante Domínguez Juana	19	I25
+	2061325	Cabrera Avila Loany	19	I24
+	2061350	Cabrera Escamilla Efrén	19	I25

Figura 6.13 Tabla *Alumnos* de la BD *Alumnos*.

7. $\beta(x)$: Su construcción se realizó con la finalidad de identificar a los alumnos dentro de la BD que tienen una calificación mayor o igual a 7 en las materias de Informática, Redes y Programación, y tiene los siguientes parámetros:

- Relación $\beta(x)$: "Conjunción", en este caso el uso de una relación conjunción o disyunción inclusiva es indistinto ya que se obtiene el mismo resultado debido a que la relación está compuesta de un solo elemento.
- Miembro Izquierdo: "variable", con una identificación del campo igual a "INF-21" que es el nombre del atributo de la tabla que contiene las calificaciones que obtuvieron los alumnos en la materia de Informática y tabla de la variable igual a "Calificaciones", que es la tabla que contiene al campo "INF-21". como se puede ver en la figura 6.11.
- La relación es " \geq ", ya que se desea saber si los alumnos tienen una calificación mayor o igual al deseado.
- Miembro Derecho es "constante", igual a 7, que es la calificación mínima aprobatoria.

Un ejemplo de los resultados obtenidos por esta variable, se muestran en la figura 6.14.

A partir de la Consulta a la Base de Datos se han obtenido los siguientes Resultados:

Menos del 60% de los Alumnos de los Grupos I23, I24 e I25 aprobaron la Materia De Informática

Menos del 60% de los Alumnos de los Grupos I23, I24 e I25 aprobaron la Materia de Redes

Más del 60% de los Alumnos de los Grupos I23, I24 e I25 aprobaron la Materia de Programación

Figura 6.14 Ejemplo de los resultados que proporciona el sistema.

El sistema también es capaz de realizar la cuantificación de los alumnos que aprobaron las materias de Redes y de Programación, para ello se definieron dos variables similares a la que acabamos de especificar.

Para conocer si los grupos han aprobado sus materias en un porcentaje mayor al 90%, se definieron tres variables del tipo cuantificador, una para cada grupo que se requiere evaluar, a continuación presentaremos la definición de la variable utilizada para la cuantificación del grupo I23:

$V_{C1} = \langle T_{C1}, Vr_{C1}, Pr_{C1}, Atr_{C1} = \langle CBD_{C1}, PL_{C1}, Q_{C1} \rangle \rangle$ donde:

T_{C4} = "Más del 90% de los alumnos del I23 aprobaron las 3 materias", texto necesario para identificar la variable.

Vr_{C4} = Datos obtenidos de la BD.

Pr_{C4} = "4", es el número de la proporción contenida en la BC que obtendrá el valor de certidumbre producto de la evaluación de la variable cuantificador.

Atr_{C4} identifica los atributos específicos como sigue:

$CBD_{C4} = \langle \text{Nombre, Usuario, Clave} \rangle$ donde:

Nombre: "Alumnos", es el nombre de la BD en la cual se realizarán las operaciones de cuantificación.

Usuario: "Marco", es el nombre de usuario autorizado para acceder a la BD.

Clave: "mark77", es la clave de acceso correspondiente al usuario.

$PL_{C4} = \langle \text{Tabla de la Base de Datos = Múltiples, Campo Llave = Matricula} \rangle$.

$Q_{C4} = \langle \text{CUP, Exacto, } > 90\%, \varphi(x):\beta(x) \rangle$ donde:

$\varphi(x) = (\text{clvGrupo} = \text{I23})$

$\beta(x) = (\text{INF-21} \geq 7) \ \& \ (\text{INF-22} \geq 7) \ \& \ (\text{INF-23} \geq 7)$

Con los siguientes parámetros:

1. Relación global: ">", puesto que se desea obtener como resultado si el número de registros de la BD que cumplen la condición son mayor a una cantidad determinada.
2. Tipos de valores: "porcentual", ya que la relación se evaluará en términos del porcentaje de los registros que cumplen la relación $\varphi(x)$, en comparación con el total de los registros cuantificados.
3. Valor de comparación: "90", representa el umbral de comparación con los registros de la BD que cumplen la relación definida.
4. Tabla de la Base de Datos: "Múltiples", se elige esta opción debido a que los registros a cuantificar son contenidos por dos tablas de la BD.

5. Campo llave: "Matricula", correspondiente al campo con el que se guiará la búsqueda de los registros a cuantificar y que se encuentra en las tablas requeridas para la consulta.
6. $\varphi(x)$: Se construye para considerar solamente a los alumnos del grupo I23, que es el grupo a cuantificar y tiene los siguientes parámetros:
 - La relación $\varphi(x)$: "Conjunción", en este caso el uso de una relación conjunción o disyunción inclusiva es indistinto ya que se obtiene el mismo resultado debido a que la relación está compuesta de un solo elemento.
 - El Miembro Izquierdo es "variable", con una identificación del campo igual a "clvGrupo" que es el nombre del atributo de la tabla que contiene la clave del grupo al que pertenece el alumno y la tabla de la variable igual a "Alumnos", que es la tabla que contiene al campo "clvGrupo", como se puede ver en la figura 6.13.
 - La relación particular es "=", ya que se desea saber si el alumno corresponde al grupo evaluado.
 - El miembro derecho es "constante" con el valor igual a I23, ya que es el grupo a evaluar.
7. $\beta(x)$: Su construcción se realizó con la finalidad de identificar a los alumnos del grupo I 23, que tienen una calificación mayor o igual a 7 en las materias de Informática, Redes y Programación, y tiene los siguientes parámetros:
 - Relación $\beta(x)$: "Conjunción", ya que se desea evaluar a los alumnos del grupo I23 que aprobaron las tres materias.
 - Miembro Izquierdo: "variable", con una identificación del campo igual a "INF-21", "INF-22" e "INF-23" que es el nombre del atributo de la tabla que contiene las calificaciones que obtuvieron los alumnos en las tres materias que se requieren evaluar y tabla de la variable igual a "Calificaciones", que es la tabla que contiene a los tres campos a consultar. como se puede ver en la figura 6.11.
 - La relación es ">=", ya que se desea saber si los alumnos tienen una calificación mayor o igual al deseado.
 - Miembro Derecho es "constante", igual a 7, que es la calificación mínima aprobatoria.

Un ejemplo de los resultados obtenidos por esta variable, se muestran en la figura 6.15, como se puede observar en ésta figura, el sistema muestra la evaluación realizada para los distintos grupos, y como se menciona anteriormente se requirió de una variable para cada una de las conclusiones mostradas en la pantalla.

**A partir de la Consulta a la Base de Datos
se han obtenido los siguientes Resultados:**

Menos del 90% de los alumnos del grupo I23,
aprobaron las tres Materias

Menos del 90% de los alumnos del grupo I24,
aprobaron las tres Materias

Menos del 90% de los alumnos del grupo I25,
aprobaron las tres Materias

Figura 6.15 Ejemplo de los resultados que proporciona el sistema.

Las Variables que se definieron en el sistema son mostradas en la figura 6.16, en ella se muestra el sistema para la adquisición de las variables en HArIES. Las variables Extracción de Datos y Cuantificador son utilizadas para obtener la información de la BD que utiliza la Base de Conocimiento para mostrar los resultados que se requieren para ayudar a la toma de decisiones, en este caso se obtiene información que ayuda a conocer algunas relaciones que se pueden obtener durante la consulta a la BD en cuestión.

Variables (Concepto)	Tipo General	Tipo Particular	Dinamismo	Estado
1) Más del 60 % de los alumnos en los grupos I23, I24 e I25 aprobaron Informática	Base de Datos	Cuantificador	Dinámica	Desactivada
2) Más del 60 % de los alumnos en los grupos I23, I24 e I25 aprobaron Redes	Base de Datos	Cuantificador	Dinámica	Desactivada
3) Más del 60 % de los alumnos en los grupos I23, I24 e I25 aprobaron Programación	Base de Datos	Cuantificador	Dinámica	Desactivada
4) -(0,0)Pantalla para mostrar conclusiones obtenidas	Incertidumbre	Imagen Sensible	Dinámica	No Asociada
5) Más del 90% de los alumnos del I23 aprobaron las 3 materias	Base de Datos	Cuantificador	Dinámica	Desactivada
6) Más del 90% de los alumnos del I24 aprobaron las 3 materias	Base de Datos	Cuantificador	Dinámica	Desactivada
7) Más del 90% de los alumnos del I25 aprobaron las 3 materias	Base de Datos	Cuantificador	Dinámica	Desactivada
8) Calificación en la Materia de Informática	Númerica	Interna	Dinámica	No Asociada
9) Calificación en la Materia de Redes	Númerica	Interna	Dinámica	No Asociada
10) Calificación en la Materia de Programación	Númerica	Interna	Dinámica	No Asociada
11) Promedio	Númerica	Fórmula	Dinámica	No Asociada
12) Extracción de Calificaciones	Base de Datos	Extracción de Da...	Dinámica	Activada
13) 14(80,80)Se presentan los Datos del Alumno	Incertidumbre	Imagen Sensible	Dinámica	No Asociada
14) -(0,0)Pantalla para finalizar	Incertidumbre	Imagen Sensible	Dinámica	No Asociada

Figura 6.16 Variables utilizadas para la consulta a la BD de Alumnos.

Conclusiones

Fueron construidos y probados los módulos de adquisición de conocimiento y la correspondiente máquina de inferencia para las estructuras de representación del conocimiento definidas, con lo cual quedaron cumplidos los objetivos planteados para el desarrollo del trabajo.

Se logró el desarrollo de un módulo para la comunicación entre Bases de Conocimiento y Bases de Datos, que posibilita la construcción de sistemas Híbridos, este módulo puede ser utilizado en otras aplicaciones o como base para el desarrollo de otros sistemas.

Con el software construido es posible desarrollar aplicaciones específicas, dados los ambientes visuales de comunicación con los usuarios, que se implementaron.

Quedaron probadas las posibilidades del sistema desarrollado para trabajar con una o varias BDs, las cuales pueden estar desarrolladas en distintos DBMSs y a su vez consultadas por medio de una conexión ya sea local o remota.

Como parte del trabajo se logró la construcción de diversas aplicaciones, que permiten demostrar la utilidad práctica de la teoría y software desarrollado para abordar la solución de problemas específicos.

Trabajos Futuros

Con el desarrollo del sistema y sobre todo de las aplicaciones presentadas, se detectaron un conjunto de necesidades que han permitido definir los trabajos futuros que deben dar continuidad a los resultados que se presentan. Entre estos se pueden mencionar los siguientes:

1. La posibilidad de implementar, dentro del medio ambiente HArises, una nueva estructura que sea capaz de realizar actualizaciones a una Base de Datos determinada, en base a los resultados obtenidos durante la consulta de una Base de Conocimiento.
2. Realizar cálculos sobre los datos contenidos en una BD, a partir de una estructura que sea definida en una BC.
3. La obtención de tendencias registradas en una BD en particular, establece otra tarea de investigación para su futura implementación.
4. Continuar el trabajo de implementación de las aplicaciones presentadas en el capítulo VI, con el objetivo de proporcionar mejores resultados de los que se obtienen actualmente con los sistemas desarrollados.

Bibliografía.

- [1] Alonso M., De la Cruz A.: "Desarrollo de Sistemas Educativos Utilizando Técnicas de Computación Inteligente".
Ier Coloquio de Investigación en Ciencias de la Computación. Memoria Técnica. Pp. 73-79,2001.
- [2] Alonso M., De la Cruz A.: "Técnicas de Representación de Conocimientos para la Construcción de un Sistema Educativo Inteligente".
Simposium Internacional de Computación. (CIC'98) Pp. 184-196, 1998.
- [3] Alonso M.: "Máquina de Inferencias para la Representación de Meta-Conocimientos".
Tesis para optar por el Grado Científico de Maestro en Informática Aplicada. ISPJAE 1996.
- [4] Anthony M., Bartlett P. L.: "Neural Network Learning: Theoretical Foundations"
Cambridge University Press, United Kingdom 1999.
- [5] Atzeni P., Ceri S., Paraboschi S., Torlone R.: "Database Systems: Concepts, languages & architectures".
McGraw-Hill International, United Kingdom 1999.
- [6] Banzhaf W., Nordin P., Keller R., Francoe F.: "Genetic Programming: An Introduction".
Morgan Kaufmann Publishers, Inc., U.S.A. 1998.
- [7] Booch G.: "Object-Oriented Analysis And Design". Second Edition.
Addison Wesley, U.S.A. 1994.
- [8] Brownston, L., Farrell, R., Kant, E., and Martin, N.: "Programming Expert Systems in OPS5", Reading, MA, Addison-Wesley, 1985.
- [9] Cali A., Calvanese D., De Giacomo G., Lenzerini M.: "Data integration under integrity constraints".
In Proc. of the 14th Conf. on Advanced Information Systems Engineering (CAiSE 2002), volume 2348 of Lecture Notes in Computer Science, pages 262-279. Springer, 2002.

- [10] Castillo E., Gutierrez J. M., Hadi A. S.: "Expert Systems and Probabilistic Networks Models".
Springer-Verlag New York, Inc., U.S.A. 1997.
- [11] Ceballos F.: "Programación Orientada a Objetos con C++", Segunda Edición. RA-MA, Madrid, España 1997.
- [12] Ceballos F.: "Visual C++ 6: Aplicaciones para Win32", Segunda Edición.
Alfaomega, México 2000.
- [13] Cherkassky V., Mulier F.: "Learning From Data: Concepts, Theory, and Methods".
|John Wiley & Sons, Inc., U.S.A. 1998.
- [14] Constable Robert L.: "Computational Type Theory"
Febrero 11, 2003.
- [15] Cox E.: "The Fuzzy Systems Handbook", Second Edition.
Academic Press, U.S.A. 1999.
- [16] Date C: "Introducción a los sistemas de bases de datos", quinta edición.
Addison Wesley, U.S.A. 1993.
- [17] Dawid H.: "Adaptive Learning by Genetic Algorithms".
Springer-Verlag Derlin Heidelberg, Germany 1999.
- [18] De la Cruz A., Alonso M., Ramírez O.: "Sistema Experto para Diagnóstico de Transformadores".
Revista Tecnolab. Número 82. Volumen XIV. Pp. 29-32, 1998.
- [19] De la Cruz A., Alonso M.: "Desarrollo de Sistemas Computacionales Inteligentes Híbridos".
Ier Coloquio de Investigación en Ciencias de la Computación.
Memoria Técnica. Pp. 66-72, 2001.
- [20] De la Cruz A., Valdés J., Jócik E., Balsa J., Rodríguez A.: "Manual de Usuario del Sistema ARIES".
Editorial Academia, La Habana, Cuba 1992.
- [21] De la Cruz A., Valdes J., Pérez A., Jócik E., Balsa J., Rodríguez A.: "General description of trie ARIES environment (version 4.00) for the construction of knowledge-based expert systems".
Segundo Congreso Internacional de Informática (Informática'90). Reporte de participación, La Habana Cuba, Pp. 241-258, 1992.

- [22] De la Cruz A., Valdés J., Pérez A., Jócik E., Balsa J., Rodríguez A.: "The ARIES Environment for the Development of Knowledge Based Expert Systems". Industrial Applications of Artificial Intelligence. Elsevier Science Publishers. B.V. North-Holland, Pp. 203-209, 1991.
- [23] De la Cruz A.: "Una máquina de inferencias de propósito general para la elaboración de conclusiones en sistemas expertos". Reporte participación Informática'88. Pp. 220-239, 1988.
- [24] De la Cruz, A., Alonso, M: "Teoría para la Construcción de Sistemas Multimedia Inteligentes". Memorias del Simposium Internacional de Computación CIC'98, México DF, Pp. 184-196, 1998.
- [25] De la Cruz, A., Alonso, M.: "Utilización de Técnicas de Minería de Datos e Inteligencia Artificial para pronósticos en un yacimiento de petróleo". Avances en Inteligencia Artificial. MIC AI/TAINA 2002. Pp. 151-160, 2002.
- [26] De la Cruz, A., Valdés J., Jocik E., Balsa J., Rodríguez A.: "Fundamentos y Práctica de la Construcción de Sistemas Expertos". Editorial Academia, La Habana, Cuba 1993.
- [27] De la Cruz, A.: "Construcción del concepto de Sistemas Expertos: Ejemplos de Aplicaciones". Memorias Foro Computación de la teoría a la práctica, México DF. Pp. 262-278, 1999.
- [28] Debenham J.: "Knowledge Engineering: Unifying Knowledge Base and Datábase Design". Springer-Verlag, Berlín Heidelberg, Germany 1998.
- [29] Fowler M., Scott K.: "UML Distilled: Applying Standard Object Modeling Language". Addison Wesley, U.S.A. 1997.
- [30] Giarratano J., Riley G.: "Sistemas Expertos: Principios y programación", Tercera Edición. Internacional Thomson Editores, S.A. de C.V., México 2001.
- [31] Gingsberg M.: "Essentials of Artificial Intelligence". Morgan Kaufmann Publishers, Inc., U.S.A. 1993.
- [32] Graham L: "Object Oriented Methods, Principies & Practice". Third Edition. Addison Wesley, U.S.A. 2001.

- [33] Hilera J., Martínez V.: "Redes Neuronales Artificiales: Fundamentos, modelos y aplicaciones".
Alfaomega Grupo Editor, S.A. de C.V., México 2000.
- [34] Ignizio J.: "Introduction to Expert Systems: The Development and Implementation of Rule-Based Expert Systems".
McGraw-Hill, Inc., U.S.A. 1991.
- [35] Jackson P.: "Introduction to Experts Systems", Third Edition.
Addison Wesley, England 1999.
- [36] Jiawei H., Kamber M.: "Data Mining: Concepts and Techniques".
Academic Press, U.S.A. 2001.
- [37] Jiawei H.: "DBMiner: A system for Data Mining in Relational Databases and Data Warehouses".
Proc. CASCON'97: Meeting of Minds, Toronto, Canada, November 1997.
- [38] Johnson J.: "Bases de datos: Modelos, lenguajes, diseño". Oxford
University Press, México, 2000.
- [39] Knapik M., Johnson J.: "Developing Intelligent Agents for Distributed Systems".
McGraw-Hill, Inc., U.S.A. 1998.
- [40] Lakemeyer G., Nebel B.: "Foundations of Knowledge Representation and Reasoning".
Springer-Verlag, Berlín Heidelberg, Germany 1994.
- [41] Luger G., Stubblefield W.: "Artificial Intelligence: Structures and Strategies for Complex Problem Solving". Third edition.
Addison Wesley, U.S.A. 1998.
- [42] Michalewicz Z.: "Genetic Algorithms + Data Structures = Evolution Programs".
Springer-Verlag Derlin Heidelberg, U.S.A. 1999.
- [43] Michalski R., Bratko I., Kubat M.: "Machine Learning and Data Mining: Methods and Applications".
John Wiley & Sons, Inc., U.S.A. 1998.
- [44] Mitchel T.: "Machine Learning".
McGraw-Hill, Inc., U.S.A. 1997.
- [45] Morik K., Wrobel S., Kietz J., Emde W.: "Knowledge Acquisition and Machine Learning: Theory, methods and applications".
Academic Press Limited, U.S.A. 1997.

- [46] Nilson N. J.: "Artificial Intelligence: A New Synthesis". Morgan Kaufmann Publishers, Inc., U.S.A. 1998.
- [47] Parsaye K., Chignell M.: "Intelligent Database Tools and Applications". Jhon Wiley & Sons, Inc., U.S.A. 1993.
- [48] Pfeifer R., Scheier C: "Understanding Intelligence". Massachussets Institute of Technology Press, U.S.A. 1999.
- [49] Pierce Benjamin C: "Types and Programming Languages" MIT Press, U.S.A., 2002.
- [50] Quatrani T.: "Visual Modeling with Rational Rose and UML". Addison Wesley, U.S.A. 1998.
- [51] Reghis M., Roventa E.: "Classical and Fuzzy Concepts in Mathematical Logic and Applications". CRC Press, U.S.A. 1998.
- [52] Rich E., Knigth K.: "Artificial Intelligence", Second Edition. McGraw-Hill, Inc., U.S.A. 1991.
- [53] Rolston W. D.: "Inteligencia Artificial y Sistemas Expertos". Me Graw Hill Interamericana de México, Estado de México, México 1992.
- [54] Russell S., Norving P.: "Inteligencia Artificial: Un enfoque moderno". Prentice Hall Hispanoamericana, S.A., México 1996.
- [55] Shapiro S.: "Encyclopedia of Artificial Intelligence", Second Edition. John Wiley & Sons, Inc., U.S.A. 1992.
- [56] Simon A. R.: "Strategic Database Tecnology: Management for the year 2000". Morgan Kaufmann Publishers, Inc., U.S.A. 1995.
- [57] Stefik M.: "Introduction to Knowledge Systems". Morgan Kaufmann Publishers, Inc., U.S.A. 1995.
- [58] Tecuci G.: "Building Intelligent Agents". Academic Press, U.S.A. 1998.
- [59] Valdés J.J., De la Cruz A.V., Jocik E., Balsa J., Rodríguez A.: "Ingeniería del Conocimiento en el Medio Ambiente ARIES". Editorial Academia, La Habana (1993).

- [60] Westphal C., Blaxton T.: "Data Mining Solutions". John Wiley & Sons, Inc., U.S.A. 1998.
- [61] Witten I., Frank E.: "Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations". Academic Press, U.S.A. 2000.
- [62] Wooldridge M. J., Veloso M.: "Artificial Intelligence Today". Springer-Verlag, Berlín Heidelberg, Germany 1999.
- [63] Zaratian B.: "Microsoft Visual C++ 6: Programmer's Guide". Microsoft Press, U.S.A. 1998.

Apéndice A. Teoría de Tipos.

Los conceptos básicos de teoría de tipos son fundamentales para la Informática, Lógica, y Matemática. Juega un papel muy importante en la ciencia de la computación y la ciencia de la información. En un nivel más amplio, la teoría de tipos es una rama de la Matemática y la Lógica que tiene relación con la clasificación de entidades en conjuntos llamados tipos [14].

Para el presente trabajo, es importante mencionar que la teoría de tipos se encuentra estrechamente relacionada con la Lógica, ya que las reglas para identificar las evidencias en los tipos son precisamente las reglas utilizadas en los operadores lógicos.

Un sistema de tipos es un método sintáctico, utilizado para validar la ausencia de ciertas conductas dentro de un programa, clasificando frases según los tipos de valores que ellos calculan a partir de una secuencia de instrucciones [49].

En otras palabras, un sistema de tipos divide valores del programa en conjuntos bien definidos llamados tipos (esta acción se conoce como una "asignación de tipos"), y define ciertas conductas que adoptará el programa, en base a los tipos que se asignan.

Existe una relación notable entre la lógica y teoría de tipos. De cierta forma, en la vida cotidiana usamos el idioma lógico. Por ejemplo, para que la fórmula $\exists x: A. B(x)$, la cual puede traducirse como "existe un x de tipo A , tal que $B(x)$ ", sea verdadera, se deberá tener un objeto " a " del tipo " A ", llamado *testigo* y una *evidencia* $B(a)$ que también sea verdadera.

Cuando se define el tipo de evidencia relacionada a una proposición, también se especifica el contenido computacional. Por ejemplo, si sabemos que P implica Q ; dadas las proposiciones P y Q , se tiene un procedimiento para encontrar la evidencia de P a partir de una evidencia para Q .

El significado computacional de $\exists x: A. B(x)$ es que se puede exhibir un método efectivo para tomar elementos de A , por ejemplo a , que tendrá que evidenciarse para $B(a)$. Si cualquier evidencia en $B(x)$ es evidencia para cualquier elemento de A , entonces es una evidencia computacional para $\forall x: A. B(x)$. En este caso el tipo de la evidencia se comporta como $\forall x: A \rightarrow B(x)$.