

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS

FACULTAD DE CIENCIAS MATEMÁTICAS

UNIDAD DE POSGRADO

**Sistema de razonamiento basado en casos, para la
mejora de atención de salud en un centro rural**

TESIS

Para optar el Grado Académico de Magíster en Computación e
Informática

AUTOR

Miguel Angel Aguilar Luna Victoria

ASESOR

Augusto Cortéz Vásquez

Lima – Perú

2016

FICHA CATALOGRÁFICA

AGUILAR LUNA VICTORIA, Miguel Angel

SISTEMA DE RAZONAMIENTO BASADO EN CASOS, PARA LA MEJORA DE ATENCIÓN DE SALUD EN UN CENTRO RURAL

PROGRAMA: C.0.3.TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

LÍNEA DE INVESTIGACIÓN:

C.0.3.17 Reconocimiento de Patrones

C.0.3.19 Informática y Sociedad

C.0.3.24 Almacenamiento y Recuperación de Información

Tesis: Facultad De Ciencias Matemáticas, Universidad Nacional Mayor de San Marcos
Formato 297 x 210 mm. Páginas x 140

(Lima, Perú 2016)

SISTEMA DE RAZONAMIENTO BASADO EN CASOS, PARA LA MEJORA DE ATENCION DE SALUD EN UN CENTRO RURAL

Autor: Miguel Angel Aguilar Luna Victoria

Tesis presentada a consideración del jurado examinador nombrado por la Unidad de Posgrado de la Facultad de Ciencias Matemáticas de la Universidad Nacional Mayor de San Marcos como parte de los requisitos para obtener el grado académico de **Magister en computación e informática.**

.....
Dra. Luzmila Pró Concepción
Presidenta

.....
Mg. Javier Cabrera Díaz
Miembro

.....
Mg. Luz Cotrina Del Pino Rodríguez
Miembro

.....
Mg. Esther Berger Vidal
Miembro

.....
Mg. Augusto Parcemón Cortez Vásquez
Miembro Asesor

DEDICATORIA:

Con especial aprecio y cariño,

para mi hermano

PEPO.

AGRADECIMIENTOS

A nuestro Creador, por brindarme en cada momento de mi vida, el aliento para avanzar, conocimiento para discernir y salud para disfrutar de su maravillosa creación.

A mi asesor, Mg. Augusto Cortéz Vásquez, por sus apreciados conocimientos en este tema y quien en todo momento me alentó a concluir este trabajo.

A mis colegas de la Facultad de Ciencias de la Universidad José Faustino Sánchez Carrión, quienes de una u otra manera me apoyaron en la culminación.

A mi familia, por el apoyo constante.

A Carmen, por su comprensión.

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS

FACULTAD DE CIENCIAS MATEMÁTICAS UNIDAD DE POSTGRADO MAESTRÍA EN COMPUTACIÓN E INFORMÁTICA

SISTEMA DE RAZONAMIENTO BASADO EN CASOS, PARA LA MEJORA DE ATENCION DE SALUD EN UN CENTRO RURAL

Autor : AGUILAR LUNA VICTORIA, Miguel Angel
Asesor : CORTEZ VÁSQUEZ, Augusto
Grado : Tesis para optar el Grado de Magister en Computación
Fecha : Agosto 2016

RESUMEN

La práctica médica en las localidades rurales de nuestro país es un problema álgido, pues debido al limitado uso de los recursos, se hace difícil y mucho más por la reducida disponibilidad de especialistas. Asimismo, en estos lugares remotos, no se cuenta con un sistema que sirva como instrumento de apoyo para realizar sus funciones. Dicho lo anterior, la presente tesis responde al objetivo de diseñar un modelo de sistema utilizando el razonamiento basado en casos (RBC) como apoyo al médico, para diagnosticar enfermedades más comunes en pobladores de un centro rural, con la finalidad de paliar en parte las necesidades básicas de salud en aquellos lugares. El modelo del sistema propuesto, presentará la estructura teniendo en consideración el ciclo del RBC, considerando al caso en estudio en sus cuatro fases: recuperación, adaptación, revisión, evaluación y aprendizaje, para finalmente, integrar el caso aprendido para incrementar la base de conocimiento. Con este sistema, se tendría un apoyo al médico como especialista en salud, no solo para el diagnóstico y solución de emergencias, sino inclusive, para realizar una acción preventiva a las enfermedades en favor de los pobladores alejados. Se trabajó con una muestra de 40 casos, construyéndose la estructura de casos sobre los síntomas más comunes de las dos enfermedades: malaria y leishmaniasis. Se utilizó el software JCOLIBRI, llamando previamente al archivo creado en Protegé, donde se elaboró la ontología dominio, es decir los síntomas y la relación con los casos.

Palabras clave: Sistema de Razonamiento Basado en Casos (SRBC), Ciclo, Similitud de casos, Base de casos, Conector, Diagnóstico basado en casos, Atención de salud, malaria, leishmaniasis.

NATIONAL UNIVERSITY OF SAN MARCOS
FACULTY OF MATHEMATICS
GRADE UNIT POST
MASTER OF COMPUTING AND INFORMATICS

**REASONING SYSTEM BASED ON CASES, FOR THE IMPROVEMENT
OF HEALTH CARE IN A RURAL CENTRE**

Author : AGUILAR LUNA VICTORIA, Miguel Angel
Advisory : CORTEZ VÁSQUEZ, Augusto
Academic degree : Thesis for degree of MA in computing
Date : August 2016

ABSTRACT

Medical practice in rural localities of our country is a critical problem, because due to the limited use of resources, it becomes much more difficult and reduced availability of specialists. Also, in these remote areas, it does not have a system that serves as a support to perform their functions. That said, this thesis with the objective of designing a model system using the case-based (RBC) to support the physician to diagnose common diseases in residents of a rural center, in order to alleviate some of the reasoning basic health needs in those places. The model of the proposed system will present the structure taking into account the cycle of RBC, considering the case study in four phases: recovery, adaptation, review, evaluation and learning, to eventually integrate the case learned to increase the knowledge base . With this system, a support specialist health doctor would have, not only for the diagnosis and resolution of emergencies, but even to make a preventive disease in remote villagers for action. We worked with a sample of 40 cases, the case structure constructed on the most common symptoms of the two diseases: malaria and leishmaniasis. The software was used JCOLIBRI previously calling file created in Protege, where domain ontology was developed, ie the symptoms and the relationship with the cases.

Keywords: System Based Reasoning (SRBC), Cycle, Similarity, Base case, Connector, case-based diagnosis, health care, malaria, leishmaniasis.

CONTENIDO

FICHA CATALOGRÁFICA.....	I
AGRADECIMIENTOS	IV
RESUMEN	V
ABSTRACT.....	VI
LISTA DE TABLAS	XI
LISTA DE FIGURAS.....	XII
CAPÍTULO I.....	1
INTRODUCCIÓN	1
1.1. Antecedentes	1
1.2. Problemática	3
1.2.1 Problema general.....	3
1.2.2 Problemas específicos	4
1.3. Justificación	4
1.4. Objetivos	6
1.4.1. Objetivo general	6
1.4.2. Objetivos específicos	7
CAPÍTULO II	8
MARCO TEÓRICO	8
2.1. Marco Filosófico de la Investigación	8
2.2. Antecedentes de la Investigación	10
2.2.1. Del problema.....	10
2.2.2. De la técnica.....	11
2.2.2.1. Antecedentes Internacionales.....	12
2.2.2.2. Antecedentes Nacionales	16
2.3. Bases teóricas.....	18
2.3.1. Ingeniería del Conocimiento	18
2.3.2. Sistemas Basados en Conocimiento.....	18
2.3.3. Definición de SBC	20
2.3.4. Adquisición del conocimiento.....	21
2.3.5. Ventajas y desventajas de un sistema experto.....	23
2.3.6. Tipos de SBC	25

2.3.7.	Estructura de un Sistema Basado en Conocimiento.....	26
2.3.8.	Técnicas para la adquisición del conocimiento.....	27
2.3.9.	Análisis de Tareas y Protocolos	29
2.3.10.	Metodología CommonKads	31
2.3.11.	Estructura del modelo CommonKads	32
2.3.12.	Descripción de los Modelos en CommonKads	33
2.3.13.	Conceptos Utilizados en la Metodología CommonKads	36
2.3.14.	Impacto y Análisis de Mejora: Tarea y Modelado de Agente.....	37
2.4.	Representación del conocimiento	42
2.4.1.	Ontología.....	42
2.4.2.	Aplicación de las ontologías	43
2.4.3.	Tipología de las ontologías	45
2.4.4.	Los conceptos.....	47
2.4.5.	Las relaciones.....	49
2.4.6.	Las funciones	50
2.4.7.	Los atributos.....	51
2.4.8.	Los axiomas	51
2.4.9.	Las instancias	51
2.4.10.	Las reglas	51
2.5.	Razonamiento Basado en Casos RBC	51
2.5.1.	Breve Historia del RBC	53
2.5.2.	Funcionamiento de un RBC	55
2.5.3.	Definición de Caso.....	57
2.5.4.	Representación e Indexación de Casos	58
2.5.5.	Representación del Conocimiento en RBC.....	59
2.5.6.	Recuperación de los casos.....	60
2.5.7.	Mecanismos de similitud en RBC.....	61
2.5.8.	Adaptación y evaluación de los casos	63
2.5.9.	Aprendizaje y mantenimiento de la base.....	64
2.5.10.	Ventajas y Desventajas de usar un Sistema RBC	64
	CAPÍTULO III.....	66
	METODOLOGÍA	66
3.1.	Diseño Metodológico	66
3.1.1.	Tipo	66
3.1.2.	Enfoque	66

3.2.	Taxonomía.....	66
3.3.	Población y muestra.....	67
3.3.1.	Población.....	67
3.3.2.	Tamaño de muestra	67
3.3.3.	Muestra.....	68
3.3.4.	Marco muestral	68
3.4.	Entornos de desarrollo a utilizar	68
3.4.1.	Protégé	69
3.4.2.	jColibri : Entorno utilizado en esta tesis	70
3.5.	Confiabilidad del modelo RBC.....	78
	CAPÍTULO IV	81
	ESTADO DEL ARTE.....	81
4.1.	Sistemas utilizados en los diagnósticos médicos	82
4.2.	Antecedentes del sistema RBC en medicina	87
4.3.	El ciclo RBC del Software jCOLIBRI utilizado	89
4.4.	El contexto para la decisión clínica	91
4.5.	¿Dónde y por qué usar un RBC?.....	92
4.6.	Sistemas RBC más Conocidos.....	95
	CAPÍTULO V.....	97
	RESULTADOS Y DISCUSIÓN	97
5.1.	Estructura del Modelo	97
5.2.	Tareas en el Sistema Propuesto.....	99
5.3.	Arquitectura del Sistema en el Modelo	99
5.3.1.	Conocimiento del Dominio	100
5.3.2.	Aplicación CBR	100
5.3.3.	Representación de los Casos	101
5.3.4.	Recuperación del Caso	101
5.3.5.	Adaptación de Casos	102
5.3.6.	El Mediador o Mediator	103
5.4.	Implementación	104
5.4.1.	Ontología Dominio	104
5.4.2.	Implementación del Modelo CBR Propuesto en jCOLIBRI.....	106
5.4.3.	Definición de estructura de casos.....	107
5.4.4.	Construyendo el Caso-Base	109
5.4.5.	Llamada al Caso Base y Configuración de los Conectores.....	109

5.4.6.	Administración de las Medidas de Similitud	111
5.4.7.	Configuración del Comportamiento del Proceso CBR	112
5.4.8.	Aplicación del sistema propuesto con la muestra de datos	114
5.5.	Evaluación de los resultados	119
5.6.	Confiabilidad del modelo propuesto	121
5.7.	Conclusiones	124
5.7.1.	Conclusión General	124
5.7.2.	Conclusiones Específicas	126
5.8.	Recomendaciones	127
CAPÍTULO VI.....		128
IMPACTO DEL ESTUDIO		128
6.1.	Propuesta para la solución del problema.....	128
6.2.	Beneficios que aporta la propuesta	128
6.3.	Sugerencias	130
6.4.	Costos de implementación del sistema propuesto.....	130
BIBLIOGRAFÍA.....		131

LISTA DE TABLAS

Tabla 2. 1. Tipos de sistemas basados en conocimiento	26
Tabla 2. 2. Análisis de tareas. Hoja de trabajo TM-1	40
Tabla 2. 3. Análisis del ítem de conocimiento. Hoja de trabajo TM-2	41
Tabla 2. 4. Ítem de conocimiento. Hoja de trabajo Agente AM-1	42
Tabla 3. 1. Concordancia observa y esperada entre el modelo RBC y el diagnóstico médico	79
Tabla 3. 2. Valoración del índice Kappa.....	80
Tabla 4. 1. Ventajas y desventajas de los métodos KBS e ICM	84
Tabla 5. 1. Parámetros de evaluación en el diagnóstico de las enfermedades	104
Tabla 5. 2. Síntomas de la malaria y leishmaniasis.....	114
Tabla 5. 3 Evaluación del sistema.....	120
Tabla 5. 4 Diagnóstico del experto y el modelo RBC propuesto.....	121
Tabla 5. 5 Porcentaje de acierto entre el experto y el modelo CBR	121
Tabla 5. 6 Concordancia observa y esperada entre el modelo RBC y el diagnóstico médico.....	122
Tabla 5. 7 Valor kappa de Cohen y significancia de la prueba.....	123

LISTA DE FIGURAS

Figura 2. 1. Sistema basado en conocimiento	27
Figura 2. 2. Esquema de la metodología CommonKads	33
Figura 2. 3. Etapas CommonKads: dominio-modelo-software de aplicación.....	35
Figura 2. 4. Descripción general de los diferentes roles de tareas en el diseño de un SBC...	38
Figura 2. 5. Descripción general del modelo de tareas en CommonKads	39
Figura 2. 6. Funcionamiento de un RBC	56
Figura 2. 7. Ciclo de un sistema RBC	57
Figura 2. 8. Contenedor de conocimiento según Richter	60
Figura 2. 9. Taxonomía de los mecanismos de similitud.....	61
Figura 3. 1. jColibri: Estructura del framework.....	72
Figura 3. 2. jColibri1.1: Arquitectura del framework	73
Figura 3. 3. jColibri1.1: Arquitectura persistente	74
Figura 3. 4. Esquema de la arquitectura de ColibriStudio	76
Figura 3. 5. Esquema de la arquitectura de ColibriStudio	77
Figura 4. 1. Espacio de soluciones y espacio de descripciones	86
Figura 4. 2. Proceso de solución de un problema en el sistema CBR.....	90
Figura 5. 1. Modelo para el sistema propuesto	98
Figura 5. 2. Arquitectura del sistema	100
Figura 5. 3. Creación de clases y relaciones en Software Protégé	105
Figura 5. 4. Caso2001 y sus síntomas, elaborado en PROTÉGÉ	106
Figura 5. 5. Definición de los parámetros para el diagnóstico en jCOLIBRI	108
Figura 5. 6. Asignación de la estructura del caso a la ontología en jCOLIBRI	110
Figura 5. 7. Administración de los conectores en jCOLIBRI	110
Figura 5. 8. Sistema CBR propuesto: Pre-Ciclo. Ciclo y Post-Ciclo.....	113
Figura 5. 9. Pantalla de consulta a manejar por el especialista	115
Figura 5. 10. Pantalla de salida, mostrando las similitudes con el caso en consulta.....	117

Figura 5. 11. Caso con mayor similitud al de la consulta	117
Figura 5. 12. Parámetros ingresados	118
Figura 5. 14. Pantalla de salida, mostrando los casos similares al caso en consulta.....	119

CAPÍTULO I

INTRODUCCIÓN

1.1. Antecedentes

Los sistemas informáticos en el área de la medicina tienen variadas aplicaciones importantes desde hace varias décadas¹, lo que ha permitido al sector de la salud no solo poseer métodos novedosos, sencillos y eficaces de gestión administrativa en consultas, hospitales y centros de investigación biomédica, sino también disponer de complejas aplicaciones de sistemas expertos que reducen la posibilidad de error en el diagnóstico de las enfermedades y la gestión de las Historias Clínicas Electrónicas (HCE)².

Sin embargo, en nuestro país, en los lugares alejados de las capitales de provincia (sierra y selva), existe carencia de hospitales y especialistas; en las llamadas postas médicas o centros de salud comunales o rurales, se aprecia esto con mayor incidencia, donde a los médicos les toca diagnosticar enfermedades y en otros casos ejecutar prácticas que en ocasiones no son de su especialidad.

Es así que a menudo se presentan casos al médico y éste quiere conocer cómo respondieron otros pacientes al tratamiento; sin embargo, no cuenta con un repositorio de casos ni de sus tratamientos en el que se registre la experiencia de otros médicos acerca de tal o cual descripción de enfermedad, su tratamiento, los efectos, la duración de la enfermedad y otras tantas características, especialmente de casos encontrados en

1 Hersh W. Medical informatics: improving healthcare through information. JAMA: the journal of the American Medical Association. 2002; 288 (16): 1955-8.

2 Van Bommel JH. Medical Informatics. Art or Science? Methods of information in medicine. 1996; 35 (3): 157-172.

las lejanías de nuestro país para que le permita tomar una decisión acertada. Esto significa que el especialista al diagnosticar la enfermedad y al analizar los casos similares se informa mejor sobre los tratamientos dados a otros, de la misma manera al contrario al indicarse los síntomas puede ayudar al médico a un diagnóstico más certero. Obviamente no todos los pacientes tendrán los mismos síntomas y es allí donde el sistema experto puede ser en el caso de la medicina, de gran utilidad.

Por esto, es necesario almacenar de alguna manera estos casos, y precisamente el Sistema de Razonamiento Basado en Casos (SRBC) o Case Based Reasoning System (CBRS), el cual es un tipo de sistema experto, es un excelente método a utilizar para el razonamiento en este tipo de datos. Cada tratamiento de un paciente sería un caso y tendría características tales como medicamento o medicamentos que el paciente está usando, cualquier efecto secundario experimentado y así sucesivamente. Estas características pueden ser ponderadas y, a continuación ser utilizadas para razonar lo que una buena solución podría ser la de un nuevo caso.

Estudios empíricos demuestran los beneficios de la inclusión de apoyo a las decisiones en escenarios médicos, lo que conduce a los ambientes de trabajo más seguros y la prevención de errores [Durieux et al., 2000]. Ejemplos de aplicaciones satisfactorias de sistemas de soporte de decisión clínica en flujos de trabajo clínicos comprenden los sistemas de registro de pacientes [Patel et al., 2000], los sistemas de conocimientos de gestión para la ingeniería biomédica [Rinkus et al., 2004] y los sistemas de formación basados en ordenador en la patología [Crowley et al., 2003].

Dicho esto, la presente tesis pretende crear un modelo donde se establece aspectos generales y fundamentales para la elaboración de bases de casos y la utilización del razonamiento, tomando diferentes elementos para su elaboración así como también para su utilización, con la finalidad de lograr un producto útil como apoyo al diagnóstico de la dolencia o enfermedad. El SRBC, sería un método excelente para manejar el razonamiento o manera de pensar del expertise o médico especialista, donde cada diagnóstico y tratamiento de un paciente formaría parte del sistema.

1.2. Problemática

En casi todos los aspectos, la práctica médica es común e indispensable tanto en el entorno urbano como rural; sin embargo, en el rural, el cuidado está comprometido sin los suficientes recursos disponibles que si existen los hospitales urbanos; aun en grandes hospitales rurales o regionales alejados, los médicos requieren un amplio rango de habilidad o destreza que otros colegas poseen; y, debido al limitado recurso médico, hay también una reducida disponibilidad de personal especialista. A diferencia de los médicos de hospitales urbanos, los que trabajan en los rurales o centros de salud remotos, usualmente tienen a su cargo una gran cantidad de pacientes con diferentes afecciones.

En el más remoto contexto rural, todo el cuidado debe ser asumido en un pequeño hospital o posta médica, por médicos, practicantes y enfermeras en un número reducido, donde inclusive no existen especialidades. Este personal de salud no solamente es responsable de emergencias médicas sino también de otros campos de la medicina incluyendo práctica de comunidad general y la administración misma del centro de salud.

Dicho lo anterior, el planteamiento de nuestro problema responde a la necesidad urgente de paliar las necesidades básicas de salud en los centros rurales alejados de nuestro país mediante el sistema RBC, ya que con éste se tendría un apoyo al médico como especialista en salud en cuanto al diagnóstico en el que se centra esta tesis, lo cual serviría no sólo para la solución de emergencias, sino inclusive, para realizar una acción preventiva de las enfermedades a favor de los pobladores alejados a muchos kilómetros de distancia de un verdadero centro poblado que tenga los recursos médicos.

Se ha fijado, por esto el siguiente enunciado para nuestro problema general

1.2.1 Problema general

¿Se puede crear un modelo de Sistema de Razonamiento Basado en Casos (SRBC) como apoyo al médico en cuanto al diagnóstico, para la mejora de la atención de salud de un poblador en un centro rural?

1.2.2 Problemas específicos

Al no contar con un repositorio de casos en el que se almacene todo lo concerniente al diagnóstico, tratamiento y demás consideraciones para la evaluación de un paciente y asimismo no contar con la aplicación informática que permita tomar decisiones en función de los casos en estudio, se ha planteado los siguientes problemas específicos.

- 1 No se cuenta con una base de casos que incluya los síntomas y diagnósticos médicos más comunes de enfermedades, partiendo de los requerimientos o necesidades de atención de un poblador rural.
- 2 No se cuenta con un aplicativo informático que permita gestionar los casos almacenados en la base de casos.
- 3 No existe un aprovechamiento de la experiencia del médico que permita reutilizar el conocimiento de casos ya tratados.

1.3. Justificación

1.3.1. Justificación teórica

El razonamiento basado en casos, es una técnica de inteligencia artificial utilizada para resolver problemas usando experiencias similares para su solución, estas experiencias son almacenadas en casos, por lo que el sistema revisa las similitudes en la base de casos con el caso específico, recuperando luego la mejor similitud. el caso es recuperado por simple comparación en la consulta. Una solución mas avanzada consiste en usar razonamiento intensivo de conocimiento, agregando un modelo de dominio con conocimiento general explícito, entonces el mejor caso o casos coincidentes son reusados para formar parte de la solución del problema; esta solución es luego revisada para ajustarla al problema actual y finalmente el sistema puede retener el nuevo caso solución para su uso posterior, almacenándola en la base de casos y formando un ciclo RBC de conocimiento [Aamodt, A., 2004].

El RBC, cubre un amplio rango de métodos para generar, organizar, recuperar y utilizar el conocimiento retenido de casos pasados. Algunos de sus métodos pueden

ser puramente automáticos o pueden interactuar con el usuario para apoyo y orientación. Algunos otros métodos asumen gran cantidad de casos distribuidos en su base de casos, mientras otros tienen un limitado conjunto de casos típicos. El dominio de RBC es versátil y ha encontrado muchas aplicaciones en varios campos tales como la medicina, ingeniería, leyes, finanzas y manufactura; por decir unos pocos. Este sistema también ha sido satisfactoriamente aplicado para resolver problemas de clasificación de patrones encontrados frecuentemente en los campos antes mencionados [A. Aamodt, E. Plaza, 1994].

En lo que respecta al campo de la medicina, como se verá luego en los antecedentes de la técnica, se puede afirmar que esta metodología se ajusta muy bien al desempeño de los profesionales de salud, cuando se presenta un caso de enfermedad, incorporándolo dentro de su escenario clínico. Sobre todo, el RBC es atractivo en el entorno médico, porque un caso base ya existente tiene almacenado en sí mismo los síntomas, diagnóstico, tratamiento y resultado para cada paciente [Periklis, A. et al., 2011]

1.3.2. Justificación práctica

En áreas rurales o comunidades alejadas de capitales de provincia e inclusive de distrito, donde existen las postas médicas o instalaciones hospitalarias pequeñas con un casi nulo acceso a las facilidades que tiene un hospital, pero sí con el personal de salud, se hace necesario dentro de éstas, un apoyo para este personal, pues no todos son especialistas para diagnosticar tal o cual dolencia cuando se suscita una enfermedad o emergencia médica. En ese sentido, el sistema RBC, sería una gran herramienta de apoyo para los profesionales de salud, teniendo en consideración la lejanía así como la no especialización en muchos casos de este personal; pues con este sistema, localizando la enfermedad, inmediatamente se procede al diagnóstico y por consiguiente el tratamiento adecuado. Así pues el trabajo de tesis que presento contribuiría a atenuar en parte el precario escenario del sistema de salud en aquellas zonas, y por lo tanto un apoyo en los ámbitos siguientes:

- En el ámbito social, porque son los pobladores de zonas rurales y comunidades alejadas sea de sierra o selva, los que recibirán este beneficio sostenido en el tiempo, mejorando la calidad de salud y por lo tanto su calidad de vida;

asimismo, si se estableciera como medida común la instalación de un sistema de este tipo a nivel de todas las zonas alejadas, el poblador de esas zonas sentiría la confianza necesaria en los profesionales de salud, sabiendo que le van a dar la atención médica correspondiente y acorde a las exigencias que merece el tratamiento de la dolencia o enfermedad.

- En el ámbito de salud, porque en estas zonas, mayormente la presencia médica se ve limitada por la falta de equipos, como rayos X, de análisis de sangre, etc.; siendo este sistema propuesto, una ayuda al médico al comparar los diferentes síntomas del poblador con la base de casos para luego dar el diagnóstico apropiado y por lo tanto, el respectivo tratamiento o si fuera el caso, preparar al paciente hasta trasladarlo a un centro hospitalario. Es decir, con este sistema de razonamiento basado en casos, se tendría un apoyo no sólo para la solución de emergencias, sino inclusive, para realizar una acción preventiva de salud a favor de los pobladores alejados a muchos kilómetros de distancia de un verdadero centro poblado que tenga los recursos médicos suficientes.
- En el ámbito económico, porque aunque no suple a un equipo médico, ni a un hospital, retribuiría en gran medida al ahorro de realizar gastos por enfermedades que no han sido tratadas a tiempo y que fácilmente se pueden prevenir, al inspeccionar los síntomas y compararlos con la base de casos, que el médico de esa zona puede diagnosticar, con un futuro ahorro para el Estado, porque muchas veces dichas enfermedades comunes o tropicales de sierra alta se transforman en epidemias que luego es oneroso combatirlas.

1.4. Objetivos

1.4.1. Objetivo general

Crear un modelo de sistema de razonamiento basado en casos como apoyo al médico en el diagnóstico, para la mejora de la atención de salud de un poblador en un centro rural.

1.4.2. Objetivos específicos

- a) Crear el modelo para el almacenamiento en una base de casos de los síntomas más comunes de enfermedades, partiendo de los requerimientos o necesidades de atención de un poblador rural.
- b) Construir un modelo que permita gestionar la base de casos.
- c) Aprovechar la experiencia del médico almacenada en la base de casos.

Estructura de la tesis

La estructura de la tesis está organizada en 6 capítulos que a continuación se explican brevemente:

En el presente capítulo se expone los aspectos metodológicos

En el capítulo dos, se describe el marco teórico, donde se presenta el marco filosófico de la investigación, los antecedentes, tanto del problema como de la técnica, así como los conceptos abordados para la presente tesis.

En el capítulo tres se muestra el enfoque metodológico utilizado, describiéndose sobre todo los entornos de desarrollo que fueron utilizados para la construcción de nuestro modelo.

En el capítulo cuatro se presenta lo referente al estado del arte, especialmente para describir en que estado se encuentran los sistemas basados en conocimiento en lo referente al razonamiento basado en casos aplicados a las ciencias médicas.

En el capítulo cinco se informa sobre los resultados y discusión, en lo que se refiere a a la estructura del sistema desarrollado y su aplicación a dos enfermedades específicas, la malaria y la leishmaniasis.

Finalmente, en el capítulo seis, se expone el impacto que el estudio podría tener, así como los beneficios de su implementación en nuestro país.

CAPÍTULO II

MARCO TEÓRICO

2.1. Marco Filosófico de la Investigación

Los filósofos existían antes que los computadores y llevaban tiempo intentando solucionar algunas cuestiones relacionadas con la Inteligencia artificial (IA). Algunos filósofos han intentado demostrar que la IA es imposible; que las máquinas no tendrán posibilidad de actuar inteligentemente. Algunos han utilizado argumentos que tratan de dar el alto a la investigación en IA. El que la IA sea posible o no dependerá de cómo se la defina. En esencia la IA consiste en la búsqueda del mejor programa agente en una arquitectura dada. Con esta formulación, la IA es posible por definición: para cualquier arquitectura digital de k bits de almacenamiento existirán 2^k programas agente y todo lo que habrá que hacer para encontrar el mejor, es enumerarlos y probar cada uno de ellos, para cualquier arquitectura digital de k bits de almacenamiento. Es evidente que las computadoras pueden hacer muchas cosas tan bien o mejor que el ser humano, incluso cosas que la personas creen que requieran mucha intuición y entendimiento humano. También es cierto, desde luego que existen todavía muchas tareas en donde las computadoras no sobresalen. Muchos filósofos han afirmado que una máquina que pasa el Test de Turing no significa que realmente esté pensando, sino que sería solo una simulación de la acción de pensar [Russell, S., 1999].

El tema base de esta tesis, tiene sus raíces en la IA, que es el área de investigación de la cual nacen diferentes desarrollos entre los que se encuentran los agentes inteligentes, es una de las áreas de desarrollo para esta propuesta. Una estrategia de acercamiento de las tecnologías a los usuarios, se encuentra en los desarrollos que se han realizado de agentes inteligentes, del área de la inteligencia artificial. Son

conocidas las metáforas referenciadas por Negroponte³ respecto a la capacidad de relación entre las personas y los computadores, en la capacidad de estos de manejar la información y fíltrala según los intereses y necesidades de cada usuario. Existen diferentes definiciones o significados de agentes inteligentes, dependiendo principalmente del dominio del conocimiento en el cual se refiere. Nos referiremos a agentes inteligentes como fragmentos de software con características humanas que facilitan el aprendizaje. Las características pueden expresarse desplegando texto, gráfico, iconos, voz, animación, multimedia o realidad virtual.

En cuanto al marco filosófico de la técnica de razonamiento basado en casos, tema que nos atañe, es necesario situar el contexto en el que se encuentra para poder realizar una primera aproximación al tema y poder comprender el porqué de su utilización. Así pues, el CBR es un tipo de sistema experto y éste desde el punto de vista de la inteligencia artificial, es un sistema que intenta imitar el comportamiento de un ser humano experto en alguna temática, es decir imitan las actividades de un ser humano para intentar resolver los problemas de distinta índole. [Hammond, 1986]

Este tipo de sistemas basa su funcionamiento en experiencias anteriormente vividas, ya sea por el propio sistema o bien por la persona experta, y a partir de este conocimiento de vivencias realizar una asociación con estas experiencias para extraer una solución de esto. Si nos ponemos a pensar en los niños pequeños, el tipo de razonamiento que utilizan es el basado en las experiencias que le proporciona su trato con el medio. Un ejemplo relacionado con esto sería cuando un niño está en la cocina y toca una olla que está en el fuego y se quema, este habrá adquirido una nueva experiencia. Si otro día ve una olla en el fuego ya no la tocará porque tendrá esa experiencia. Aun así podría tener la experiencia de ver a su madre coger esa olla con los paños de cocina. De todo ese conjunto de experiencias, el niño, la siguiente vez que quisiese tocar una olla sabría que si lo quiere hacer tendría que hacerlo con paños para no quemarse. Este es un claro ejemplo del funcionamiento del razonamiento basado en casos. Podemos ver como el niño pequeño a través de vivencias anteriores, tanto si son errores como aciertos, adquiere experiencias que utilizará cuando tenga

³ Nicholas Negroponte (1 de Diciembre 1943), arquitecto estadounidense de origen griego, mas conocido como fundador y director del MIT Media Lab, en el cual fué profesor desde 1962.

que enfrentarse a una nueva situación y que constituirán su nueva base de razonamiento. Esta es una pequeña introducción del funcionamiento.

2.2. Antecedentes de la Investigación

2.2.1. Del problema

Se observa que la atención de salud en los entornos rurales del Perú, la proporción de población que busca atención por problemas de salud crónico y no crónico va en aumento, así al 2014, aumentó en 3,1 y 5,5 puntos porcentuales respectivamente y del total de la población que presentó algún problema de salud, sólo el 53,2% realizó consulta para aliviar el mal que le aquejaba. Al indagar por el lugar o establecimiento de consulta, el 29,2% manifestó que acudió a consultar a un establecimiento del Ministerio de Salud, el 10,3% respondió haber consultado en farmacia o botica, el 3,2% señaló consultar en establecimientos de salud del Sector Privado como Clínicas y Consultorios Particulares, el 1,2% lo hizo en un establecimiento de Seguro Social de Salud, el 0,2% consultó en su domicilio y el 0,8% en Otros. [INEI-CVP, 2014]

A nivel de la comunidad campesina Cruz de Mayo, que es de donde se extrajo la información para esta tesis, existen puestos de salud en Antash, Llacshu, Huauya y Pampacocha, los mismos que no cuentan con el personal suficiente, ni con el equipamiento adecuado para asistir a la población, ni menos un sistema informático de apoyo al diagnóstico médico. En esta comunidad por no poseer sistema de alcantarillado, así como por el consumo de agua con altos niveles de turbidez, se han incrementado las enfermedades gastrointestinales y parasitarias. [CEAS, 2013]. Precisamente en este entorno rural, se encontraron pacientes diagnosticados con malaria y leishmaniasis, las cuales ya se deberían de haber erradicado; sin embargo, según informe del Ministerio de Salud, hasta el 2013 se reportaron 35275 casos de malaria plasmodium vivax y 5472 casos de malaria plasmodium falciparum a nivel nacional. [MINSA, 2014]

La malaria o paludismo es una enfermedad parasitaria que involucra fiebres altas, escalofríos, síntomas seudogripales y anemia, causada por un parásito que se transmite de un humano a otro por la picadura de mosquitos *anofeles* infectados.

Después de la infección, los parásitos (llamados esporozoítos) migran a través del torrente sanguíneo hasta el hígado, donde maduran y producen otra forma, los merozoítos. Los parásitos ingresan en el torrente sanguíneo e infectan los glóbulos rojos. [MedlinePlus, 2015]

La leishmaniasis es una enfermedad infecciosa transmitida por la picadura del flebótomo o mosquito simúlido hembra, afecta la piel y las membranas mucosas. Las llagas en la piel por lo regular comienzan en el sitio de la picadura del flebótomo. En unas pocas personas, las llagas se pueden desarrollar en las membranas mucosas. En ocasiones se manifiesta 2 a 8 meses después de que la persona es picada por el flebótomo. La mayoría de las personas no recuerdan haber tenido una llaga en la piel. Esta forma puede llevar a complicaciones mortales. Los parásitos dañan al sistema inmunitario disminuyendo el número de células que combaten la enfermedad. [MedlinePlus, 2015]

En lo que respecta a un sistema informático de apoyo que pudiera tener el médico, no hay ningún antecedente en nuestro país; sin embargo, en otros, el RBC se ha aplicado con éxito, principalmente para tareas de diagnóstico y terapéuticas. Métodos relacionados a esta técnica han sido utilizados por ejemplo, en capacitación orientada a casos o métodos de recuperación de imágenes [MacRad, 1995], así como para dar soluciones de diagnóstico como apoyo a los profesionales de salud [El-Fakdi, 2013].

2.2.2. De la técnica

En el ámbito de la medicina, en países desarrollados, el RBC se ha aplicado principalmente para tareas de diagnóstico y terapéuticas. Métodos relacionados a esta técnica han sido utilizados por ejemplo, en capacitación orientada a casos o métodos de recuperación de imágenes [MacRad, 1995]. Una de las más tempranas aplicaciones para sistemas expertos que uso técnicas de RBC fue la de CASEY, mediante el cual se diagnosticó fallas del corazón humano. El sistema usó tres pasos: una búsqueda o investigación de casos similares; la determinación de los diferentes procesos concernientes al caso y sus evidencias entre los casos similares y el actual; y, la transferencia del diagnóstico al caso similar o actual –o si la diferencia entre ambos

casos era muy importante- intentar una explicación y modificar el diagnóstico [Koton,1988].

Por otro lado el sistema FLORENCE, trata con el planeamiento del cuidado de la salud en sentido amplio para enfermera(o)s, los cuales tienen un campo menor de especialización. Este sistema cumple tres tareas de planeamiento básico: *diagnóstico*, *tratamiento* y *prescripción*. *El diagnóstico* no es usado en el sentido médico de identificación de la enfermedad, sino para buscar la respuesta a la pregunta: ¿cuál es el estado de salud actual del paciente?, entonces las reglas concernientes a su estado son aplicadas. El estado de salud es determinado como el resultado de los pesos de los indicadores para la enfermedad. *Tratamiento*, busca responder a la pregunta ¿cómo se puede cambiar el estado de salud del paciente?, aquí se usa las aproximaciones de casos parecidos; el paciente actual se compara con pacientes similares previos, cuyos estados de salud progresivos son conocidos. Pacientes similares son analizados, en primer lugar referente a su estado de salud y luego, a sus indicadores de salud personal. Debido a que el desarrollo de salud de un paciente no solo depende de esa situación (salud actual, enfermedades comunes y presentes), sino adicionalmente de mas de un tratamiento, muchos otras proyecciones de tratamientos son generados. *Prescripción*, busca responder a la pregunta: ¿cómo mejorará la salud del paciente?; la respuesta se da al utilizar conocimiento general acerca del efecto del tratamiento en particular para ese caso y también considerando el resultado de un tratamiento particular en pacientes similares. Este sistema es pues una combinación de casos basados en reglas y aproximaciones [Bradburn, C., 1993].

Puntualmente, algunos antecedentes a nivel internacional y nacional que se pueden indicar son los siguientes:

2.2.2.1. Antecedentes Internacionales

En el extranjero, los métodos de Razonamiento Basado en Casos han sido afrontados desde diferentes aspectos y la medicina ha encontrado aplicaciones interesantes, como lo han sido los trabajos de Schmidt y de Evans desde 1995; según los especialistas en medicina el SRBC es un método reconocido para construcción de sistemas médicos.

Antecedente 1

Existe un trabajo “Avances y Tendencias de SRBC en Medicina” donde se identifican las fortalezas y debilidades de este sistema en la medicina, el cual es utilizado como base para publicaciones recientes y proyectos de investigación en esta área [8]. Según este trabajo, algunas de las fortalezas de este sistema son: que el RBC se ajusta a las exigencias específicas de una cierta clínica o cirujano; además de obtener un conocimiento incremental, los registros de pacientes se almacenan en soportes legibles, lo que simplifica la integración con sistemas RBC que pueden luego ser utilizados.

Como desventajas identificadas se tiene: aunque la fiabilidad de un sistema RBC aumenta, ésta no se puede garantizar en un 100%; la adición de nuevos casos no necesariamente hace que el sistema converja hacia una mayor fiabilidad; el sistema funciona siempre y cuando se encuentre en la base de casos una experiencia previa.

Conclusiones:

- Podría decirse que los sistemas híbridos se utilizan en salud, partiendo del sistema RBC como dominio.
- La mayoría de los sistemas construidos para desarrollar soluciones médicas parte de la combinación de RBC y otras técnicas de inteligencia artificial.

Antecedente 2

En Cuba, por ejemplo, se han logrado resultados de relativa importancia en la elaboración de softwares para el Razonamiento Basado en Casos, donde estas aplicaciones han estado orientadas al diagnóstico médico. Estos resultados han sido no solamente de interés práctico, sino también en el plano teórico, al que ha contribuido el gran volumen de información que se ha registrado en Bases de Casos.

Así, se tiene un sistema RBC en Ciencias Médicas Sobre Plataforma Web [Delgado M., 2009], cuyo valor práctico consiste en su carácter eminentemente docente y clínico, ya que facilita tanto a estudiantes de medicina y de enfermería así como residentes y médicos especialistas a que puedan interactuar con Bases de Casos

que permitan ayudarlos en el proceso del diagnóstico, así como exponer toda la información relacionada con dicho diagnóstico. Esto permite mejorar el grado de información e instrucción de profesionales de la salud de diversas áreas.

El sistema anterior, permite mostrar cualquier información sobre un caso a partir de la selección de cualquier diagnóstico; en determinados rasgos de un caso se puede mostrar una misma información (valor del rasgo) por varios canales de salida: como texto, imagen, animación o sonido, para un mismo valor del rasgo, lo que permite que los usuarios puedan interiorizar con rapidez todo lo relacionado con cualquier caso.

Conclusiones:

- Se realizó un diseño que permite la confección de cualquier Base de Casos y también, la interfaz que tendrá la aplicación en formato Web utilizando el Razonamiento Basado en Casos.
- Se desarrollaron métodos que permiten mostrar cualquier información sobre un caso a partir de la selección de cualquier diagnóstico.
- Se crearon normas que permitan el diseño de preguntas para evaluar y comprobar el conocimiento adquirido por el usuario, en el proceso de aprendizaje de cualquier Base de Casos.

Antecedente 3

En el Medical College of Winsconsin (USA), se está probando un sistema basado en RBC llamado PROTOISIS [10], que sirve como entrenamiento clínico preliminar para el apoyo en la toma de decisiones médicas, el cual fue probado y aplicado con éxito en la selección de los procedimientos de diagnóstico por imagen teniendo un uso potencial para el apoyo radiológico y tomografía computarizada, ayudando a los físicos para seleccionar el apropiado procedimiento, que luego será aplicado en la medicina.

Conclusiones:

- Una vez completado el PROTOISIS, será integrado en el departamento clínico como un sistema de información para proporcionar conocimiento a los médicos

en todo momento en las áreas de clínica, salas de hospitalización, unidades de cuidados intensivos y servicios de urgencias.

- El sistema tendrá un excelente potencial para mejorar significativamente la calidad y costo-efectividad de la atención médica, y ofrecerá una oportunidad de estudiar el papel del razonamiento basado en casos en el día a día de la toma de decisiones médicas.

Antecedente 4

En España, el Grupo de Aplicaciones de Inteligencia Artificial [GAIA] [GAIA, 2011] de la Universidad Complutense, ha desarrollado la plataforma JColibrí, (Software libre). Esta plataforma permite crear aplicaciones de RBC para la mayoría de las áreas del conocimiento, de forma rápida y sencilla, siendo utilizada en la actualidad como herramienta educativa, en más de una docena de universidades de todo el mundo. Como exponente de la repercusión, lograda por esta iniciativa, JColibrí ha conseguido la cifra de 5000 descargas, distribuidas en más de 70 países. Cuenta además, con el apoyo de varios grupos internacionales de investigación que contribuyen a su desarrollo.

Antecedente 5

En España, otro trabajo importante encontrado es “Sistema Recomendador Orientado a la Educación Terapéutica del Paciente Diabético” [Carrillo A. et. al. 2011], desarrollado en la Universidad Complutense de Madrid, para utilizar datos y mediciones analíticas del historial del paciente, en base a los cuales recomienda una serie de consejos y contenidos formativos, cuyo objetivo es ayudar al paciente a mantener una buena salud y aumentar su calidad de vida. Este sistema utilizó la herramienta JColibrí nombrada anteriormente.

Conclusión:

- Es un sistema recomendador para mejorar la educación terapéutica del paciente diabético, especialmente el de diagnóstico reciente.

2.2.2.2. Antecedentes Nacionales

Antecedente 1:

Indagando sobre información acerca de este sistema orientado a salud o medicina en el Perú, queda claro que no hay trabajos sobre SRBC orientado al diagnóstico y tratamiento de enfermedades; lo que si se tiene como antecedente, una ponencia en el I Congreso Nacional de Científicos Peruanos en la Universidad Cayetano Heredia (2003), donde se disertó el tema “Reducción de Bases de Casos Clusterizadas” cuyos expositores fueron profesionales de la Universidad de San Agustín Arequipa [Parisaca V. Abigail, et al., 2003]. El trabajo propone una técnica de reducción de bases de casos para sistemas de RBC, “clusterizando” primeramente la base de casos mediante una red neuronal artificial de tipo Kohonen para luego reducirla basándose en las propiedades de la base de casos

Conclusión:

- La selección de casos se realiza mediante clusterización garantizando que cada caso almacenado en la base de casos, sea el idóneo.

Por otro lado, lo que sí existe en nuestro país, es una cantidad significativa de investigaciones sobre este tema orientado a otras áreas, así tenemos por ejemplo:

Antecedente 2:

SRBC Aplicados a Sistemas de Líneas de Productos de Software [Cortez V. Augusto et al., 2010] (en el área de informática). Este trabajo trata de aprender de ejemplos, casos o datos conocidos de tal forma que se tome decisión sobre nuevos casos. Desde el enfoque de minería de datos, en tareas de clasificación, podemos asignar una clase a un nuevo caso observando las clases similares. Igualmente, en tareas de agrupamiento, asignaremos un nuevo ejemplo al grupo donde estén los individuos más similares.

Conclusión:

- Las ventajas de la reutilización de software son la reducción de costos, rapidez en el desarrollo de software y reducción de riesgos.

- No es de extrañar que las líneas de productos software surgen a partir de aplicaciones existentes. Es decir, una organización desarrolla una aplicación y, cuando se requiere una nueva aplicación, se utiliza la primera como base para la nueva aplicación.

Antecedente 3:

Sistema RBC en la Contraloría General de la República [Contraloría General de la República, 2006] (en el área de gestión), sistema basado en la experiencia del diseño e implementación de un sistema de gestión del conocimiento en la Contraloría General de la República.

Conclusión:

- Detalla todos los procesos a lo largo de todas sus etapas, resaltando los logros, dificultades, perspectivas, y resultados de cada uno de ellos.

Antecedente 4:

Modelo Híbrido para Análisis de Riesgo Crediticio en Pymes-Caja Municipal de Arequipa (en el área de finanzas). Este estudio propone un modelo híbrido basado en redes neuronales y razonamiento basado en casos como una herramienta que permite la evaluación de solicitudes de crédito MES para apoyar a las entidades bancarias en su proceso de análisis. [Cruz Q. Laura, 2009]

Conclusión:

- El sistema de RBC, con la utilización de una red neuronal, hacen que sea un sistema completo que se puede utilizar en la evaluación de solicitudes de créditos para las PYMES.

Así pues, aunque esta técnica es muy usada por ciencias de la salud en otros países, como base de conocimiento o referencial para la toma de decisiones sobre todo para el diagnóstico; en el nuestro, todavía no se considera, a pesar que este conocimiento tiene más de una década de iniciado.

2.3. Bases teóricas

2.3.1. Ingeniería del Conocimiento

La ingeniería de conocimiento, es la disciplina de la inteligencia artificial que proporciona los métodos y técnicas para construir los sistemas computacionales denominados sistemas basados en conocimiento, en adelante SBC o sistemas expertos, que permiten la generación de nuevo conocimiento a partir del conocimiento objetivo [Popper, 1979]⁴ o explícito [Nonaka y Takeuchi, 1999]⁵ almacenado en las bases de datos.

Para esto, se apoya en metodologías instruccionales y en las ciencias de la computación y de las tecnologías de la información, intentando representar el conocimiento y razonamiento humanos en un determinado dominio, dentro de un sistema artificial.

El trabajo de los ingenieros del conocimiento consiste en extraer el conocimiento de los expertos humanos en una determinada área, y en codificar dicho conocimiento de manera que pueda ser procesado por un sistema. El problema es que el ingeniero del conocimiento no es un experto en el campo que intenta modelar, mientras que el experto en el tema no tiene experiencia modelando su conocimiento (basado en la heurística) de forma que pueda ser representado de forma genérica en un sistema.

La ingeniería del conocimiento engloba a los científicos, tecnología y metodología necesarios para procesar el conocimiento. Su objetivo es extraer, articular e informatizar el conocimiento de un experto.

2.3.2. Sistemas Basados en Conocimiento

Se empezó a identificar a inicio de la década de los setentas, que los métodos generales y las técnicas de búsqueda desarrolladas, empleadas para solucionar los problemas orientados a aplicaciones particulares eran insuficientes. Se determinó que era necesario tener un conocimiento específico sobre el problema, es decir que esté

4 Karl Popper (Viena 28.07.1902 – Londres 17.09.1994), filósofo y teórico de la ciencia nacido en Austria.

5 Nonaka y Takeuchi afirman que la teoría de creación de conocimiento organizacional, se debe a la naturaleza del conocimiento: la ontológica y la epistemológica.

limitado a los dominios de interés de la aplicación sobre lo específico, en lugar de aplicar el conocimiento general a muchos dominios. Esto condujo al desarrollo de los sistemas basados en conocimiento. [Lezcano, M., 1998].

Generalmente para dar la definición de los SBC, muchos autores lo consideran como sistemas basados en el conocimiento o sistema experto basado en el conocimiento.

Esta terminología utilizada se presta a veces a confusión, algunos autores limitan el uso de emplear el término de sistema experto o es empleado como sinónimo de los sistemas basados en conocimiento.

Los SBC basan su potencia en el conocimiento; el conocimiento en cualquier especialidad es usualmente de dos tipos: público y privado. El conocimiento público incluye las definiciones, hechos y teorías publicadas; pero la experticidad usualmente incluye más que esta clase de conocimiento. Los expertos humanos generalmente poseen conocimiento privado. El conocimiento representado en los SBC, es el de los expertos en el dominio; poner en claro y reproducir tal conocimiento es la tarea central en la construcción de sistemas basados en conocimiento. [Alonso J., et. al., 2012]

También es necesario expresar que el usuario (experto), define el conocimiento y el sistema lo integra directamente en la aplicación existente. Por esto es que coexisten dos paradigmas de los sistemas inteligentes:

- a) Los sistemas basados en conocimiento con orientación estructural cognoscitiva, cuyas características son:
 - El poder resolver determinados problemas radica en un “saber”, o conocimiento específico, sobre el tipo de problemas, mas que en capacidades intelectivas generales.
 - La separación entre el conocimiento codificado en la máquina (base de conocimientos) y los mecanismos deductivos (motor de inferencias)
- b) Los sistemas expertos con orientación funcional o conductista, cuyas características son:
 - Se analiza la actividad de un experto humano cuando resuelve problemas en un área muy concreta y se intenta emularlo o ayudarlo.

- Capacidad para adquirir incrementalmente experiencia
- Y, capacidad para conversar con los usuarios y explicarles sus líneas de razonamiento

El término sistema experto, fue usado por primera vez por el Doctor Edward E. Feigenbaum, en la Universidad de Stanford, quien estipuló que el poder de resolución de un problema en la computadora viene del conocimiento de un dominio específico, no sólo de las técnicas de programación y el formalismo que contiene. [Feigenbaum E., 1977].

En síntesis, un sistema experto, desde el punto de la inteligencia artificial, es un sistema que intenta imitar el comportamiento de un ser humano experto en alguna temática (base de conocimiento); es decir, imitan las actividades de un ser humano para intentar resolver los problemas de distinta índole.

2.3.3. Definición de SBC

Un Sistema basado en conocimiento (SBC) es una técnica del área de la Inteligencia Artificial que se basa en el conocimiento del experto

Según Feigenbaum,

“Los sistemas basados en conocimiento son aquellos cuya capacidad para resolver problemas reside en el conocimiento que poseen”.
[Feigenbaum E., 1977].

Asimismo, Guida y Tasso, señala:

“Un sistema basado en conocimiento es un sistema software capaz de soportar la representación explícita del conocimiento de un dominio específico y de explotarlo a través de los mecanismos apropiados de razonamiento para proporcionar un comportamiento de alto nivel en la resolución de problemas”. [Guida y Tasso, 1994]

Según F. Alonso y N. Juristo,

“Los SBC, tratan con problemas poco estructurados en los que podemos encontrar requisitos subjetivos, entradas inconsistentes,

incompletas o con incertidumbre y que no pueden ser resueltos aplicando los algoritmos clásicos o la investigación operativa". [F. Alonso y N. Juristo, 1996]

En esta última definición hace referencia a problemas poco estructurados, por lo que es necesario aclarar que existen tres tipos de problemas: *Estructurados*, aquellos en los que se conoce todas y cada una de las componentes de la solución; *Semiestructurados*, aquellos en los que no se conocen algunas de las componentes de la solución y los *No estructurados*, en las que los componentes de la solución son vagas y desconocidas; es decir, que la meta no está definida y esta forma parte del problema. Los problemas que tratan los SBC son los semiestructurados y no estructurados [Cabrera L., 2011].

Teniendo en consideración las definiciones anteriormente dadas, se considera entonces que los SBC son sistemas que usan conocimiento sobre un dominio específico para llegar a solucionar problemas de ese dominio. La solución que se da a un problema sería esencialmente la misma que la obtenida por una persona experimentada en el dominio del problema cuando se enfrenta al mismo problema.

2.3.4. Adquisición del conocimiento

Uno de los procesos más costosos que hay que desarrollar en el proceso para construir procesos computacionales denominados Sistemas Basados en Conocimiento (SBC), es la adquisición del conocimiento y dado que esta tarea es difícil, se han identificado varias etapas en las que se ha de dividir su desarrollo y así permitir abordar esta labor de una manera más sistemática.

Existen diferentes versiones sobre la división de esta tarea; la más aceptada en la literatura es la siguiente [Peña, A. 2006]

- a) Identificación del problema
- b) Conceptualización
- c) Formalización
- d) Implementación
- e) Prueba

a) *Identificación del problema.*

En esta fase, se ha de determinar en primer lugar si el problema se puede o se debe abordar mediante las técnicas de los SBC. También ha de ser necesario tener acceso a las fuentes de conocimiento suficientes para completar la tarea. Por último, el problema a tratar ha de tener un tamaño adecuado para que no constituya una tarea inabordable por su complejidad. El siguiente paso consiste en buscar las fuentes de conocimiento que serán necesarias para el desarrollo del sistema, las más comunes son:

- Expertos humanos en la definición del problema
- Libros y manuales que expliciten el problema y técnicas de solución
- Ejemplos de casos resueltos.

b) *Conceptualización*

Aquí, es necesario detallar los elementos básicos del problema y descubrir las relaciones entre ellos. En particular, es necesario observar cómo el experto resuelve problemas típicos y abstrae de ellos principios generales que pueden ser aplicados en diferentes contextos.

Hay también que obtener una descomposición del problema en subproblemas, realizando un análisis por refinamientos sucesivos hasta que el Ingeniero del conocimiento pueda hacerse una idea de la relación jerárquica de las diferentes fases de resolución hasta los operadores de razonamiento más elementales. Con esta descomposición jerárquica y el flujo del razonamiento, el IC puede caracterizar los bloques de razonamiento superiores y los principales conceptos que definen el problema. Hará falta distinguir entre evidencias, hipótesis y acciones necesarias en cada uno de los bloques y determinar la dificultad de cada una de las subtareas de resolución. De esta manera se conseguirá captar la estructura del dominio y las diferentes relaciones entre sus elementos.

c) Formalización

En este punto se ha de poder comprender la naturaleza del espacio de búsqueda y el tipo de búsqueda que habrá que hacer. Para ello, se puede comparar ésta con diferentes mecanismos prototípicos de resolución de problemas como la clasificación, abstracción de datos, razonamiento temporal, estructuras causales, etc.; también tendrá que analizarse la certidumbre y completitud de la información disponible, dependencias temporales, o la fiabilidad y consistencia de la información. Se deberá descubrir qué partes del conocimiento constituyen hechos seguros y cuáles no. Para éstos últimos deberá adaptarse alguna metodología de tratamiento de la incertidumbre, de manera que ésta pueda ser modelizada dentro del sistema.

d) Implementación

En esta fase se han de tomar decisiones sobre la especificación del control de la resolución y del flujo de la información. Se deberán tomar decisiones sobre el modo concreto de representar el conocimiento para que se adapte a las estrategias de resolución que se necesiten y las relaciones entre los diferentes conjuntos de conocimiento; asimismo, se definirán las reglas, e inevitablemente se descubrirán problemas que obligarán a revisar fases anteriores.

e) Prueba

Se ha de elegir un conjunto de casos resueltos representativos y se ha de comprobar el funcionamiento del sistema con éstos. En esta fase se descubrirán errores que permitirán corregir análisis anteriores; por lo general aparecerán problemas por falta de reglas, incompletitud, falta de corrección, y posibles fallas en el análisis de las reglas pre- establecidas.

2.3.5. Ventajas y desventajas de un sistema experto

Las ventajas y desventajas que se presentan al comparar con otras soluciones son:

a) Ventajas:

- Incrementa la difusión de la experticia escasa.
- El RBC no necesita un modelo explícito del dominio, pues utiliza las experiencias pasadas (casos) en su lugar.
- Los grandes volúmenes de información pueden ser gestionados aplicando técnicas de gestión de bases de datos.
- El proceso de implementación se simplifica ya que la manipulación de los casos es mucho más sencilla.
- Los sistemas RBC pueden adquirir nuevo conocimiento fácilmente a través de nuevos casos, lo que simplifica en gran medida su mantenimiento.
- Es de fácil modificación debido a que el conocimiento es explícito y accesible.
- Es consistente y diverso en las respuestas ya que los expertos humanos pueden diferir en sus explicaciones, incluso un mismo experto puede responder de forma diferente en formas diferentes.
- Posee gran accesibilidad debido a que los SBC trabajan las 24 horas del día.
- Posee la capacidad para adquirir nuevo conocimiento además de perfeccionar el que posee.
- Explica las soluciones dadas.
- Permite evaluar el efecto de nuevas estrategias añadiendo o modificando conocimiento.
- Construye un entrenador en el dominio de aplicación.

De las ventajas antes mencionadas la más importante en un SBC es la gran accesibilidad que posee, pues trabaja todo el tiempo y a toda hora a diferencia a un experto humano. Otra de las ventajas importantes es la de dar explicación a las soluciones al justificar las conclusiones y explicando porque se hace cada pregunta.

b) Desventajas

- Las respuestas no siempre son correctas.
- Conocimiento limitado al dominio de experticia.
- Ausencia de sentido común.
- No reconocen el límite de su conocimiento.

2.3.6. Tipos de SBC

Como se ha mencionado, existen muchas definiciones sobre lo que es un Sistema Basado en Conocimiento, que a su vez han ido evolucionando según se iba investigando sus posibilidades. Actualmente, un SBC se puede definir como una rama de la inteligencia artificial, que utiliza conocimiento sobre un dominio de aplicación para obtener una solución de un problema en este dominio. El dominio suele ser muy específico y se suele utilizar a un experto del dominio como fuente de conocimiento (también se pueden usar otras fuentes, como fuentes textuales, técnicas de análisis de datos, etc.).

En un SBC se separa el conocimiento y el motor de inferencia. Esta separación permite que se apliquen distintas técnicas de razonamiento a una misma base de conocimiento fácilmente, pudiendo comparar los resultados para un problema concreto. Para obtener el conocimiento se debe extraer de un experto del dominio y representarlo en una base de conocimiento (tareas básicas de la Ingeniería del Conocimiento). Existen distintas técnicas de adquisición de conocimiento y de representación, según sea el tipo de conocimiento (declarativo, procedimental o metaconocimiento). Igualmente hay muchos paradigmas no excluyentes para la resolución de problemas: paradigma lógico, de búsqueda heurística, basado en conocimiento, conexionista, basado en experiencia. Dentro del paradigma basado en experiencia se encuentran el razonamiento basado en reglas (Rule Based Reasoning), razonamiento basado en casos y las ontologías. Nosotros nos centraremos en este paradigma y más concretamente en el RBC.

En la tabla 2.1 se presenta en resumen los diferentes sistemas relacionados al SBC.

Tabla 2. 1. Tipos de sistemas basados en conocimiento

Nombre	Formas de representación del conocimiento	Explicación	Método de solución del problema	Fuentes de conocimiento
Sistemas basados en reglas	Reglas de producción	Reglas activas	Encadenamiento hacia atrás y hacia adelante o combinadas – (backward or forward chaining)	Expertos, publicaciones, ejemplos
Sistemas basados en casos	Casos	Casos semejantes	Razonamiento basado en casos (búsqueda por semejanza y adaptación de las soluciones)	Ejemplos
Sistemas basados en probabilidades	Probabilidades y frecuencias	Valores de probabilidades condicionales	Teorema de Bayes y otras técnicas de inferencia estadística	Ejemplos
Redes expertas	Pesos y alguna otra FRC	Según el modelo simbólico	Cálculo de niveles de activación de las neuronas	Ejemplos

Fuente: Institute of Computer Science. University of Hildesheim

2.3.7. Estructura de un Sistema Basado en Conocimiento

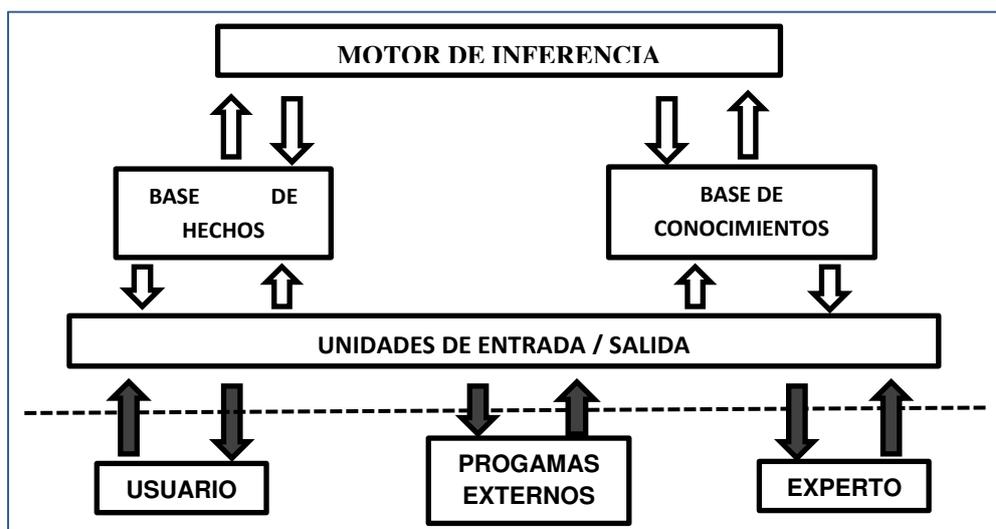
Los sistemas basados en conocimiento, suelen tener una estructura común, a saber:

- a) **Base de conocimiento:** contiene los conocimientos relativos a la tarea. Usa formalismos de representación para codificar los conocimientos en la base. En esta etapa se refleja la estructura cognitiva y los procesos humanos, es la llamada memoria de largo plazo, en las que se guarda los hechos (base de hechos) y los conocimientos (base de conocimientos) acerca del dominio en el que tiene experiencia.
- b) **Motor de inferencias:** medio por el cual controlan y aplican los conocimientos. Permiten que el sistema razone a partir de los datos, noticias o conocimientos de entrada para producir los resultados de salida. Gobiernan

el orden en el que el sistema realiza los pasos de razonamiento, acepta entradas y produce salidas. Esta etapa se puede resumir como el motor que realiza la función de razonamiento para resolver problemas.

- c) **Interfaz de E/S:** no solo para que el usuario proporcione hechos y datos y el sistema responda, sino que además desde aquí se puede interactuar con la base de conocimientos.

Figura 2. 1. Sistema basado en conocimiento



Fuente: WINOGRAD, T.A., FLORES, F. Understanding Computers and Cognition. Norwood, N.J. 1986.

2.3.8. Técnicas para la adquisición del conocimiento

- **Entrevistas**

Es uno de los métodos más familiares de adquirir conocimiento sobre la terminología y los principales componentes del dominio, sirviendo para conseguir algunos conceptos básicos y establecer información se pueden estructurar de varias maneras, la más sencilla es solicitar al experto la preparación de una hora de exposición acerca de los temas principales concernientes al dominio, luego esto proporciona información sobre aspectos relevantes con mayor profundidad.

Esta técnica tiene algunas limitaciones y aparecen al refinar las versiones preliminares del sistema experto, por ejemplo una limitación es intentar

representar en forma de reglas un conocimiento que no es tratable con dicha técnica, pudiendo tener implicaciones futuras; por lo tanto, en ocasiones el experto no puede ser directamente comunicable en una entrevista.

En el marco de las entrevistas suelen articularse los protocolos de adquisición. Entre los más usados están:

- **Protocolos de pensar en voz alta**

El pensamiento en voz alta durante la realización de una tarea por sí sola no es tan plausible, sin embargo conjuntamente con la observación del experto en el desarrollo de la misma da información invaluable para incrementar el entendimiento del ingeniero del conocimiento de los requerimientos de la tarea.

- **Protocolos de discusión**

El desarrollo de un problema a través de la discusión de dos expertos, puede ser grabado para el seguimiento del proceso. La existencia de dos perspectivas brinda más información al ingeniero del conocimiento para analizarla y clasificarla y tomar una decisión.

- **Verbalización retrospectiva**

Consiste en que el experto realice la tarea e inmediatamente después reportar el conocimiento empleado en la misma. Los problemas encontrados cuando se trata de verbalizar durante la tarea son evitados. Acortando el tiempo entre el comportamiento actual y recordando mejora la habilidad del experto, siempre y cuando se acorte el tiempo entre el desarrollo de la tarea y la subsecuente entrevista.

Este tipo de entrevista tiene sus limitaciones, una de ellas es el olvido entre el periodo de la realización y el recuerdo. Este déficit de recuerdo se puede remediar, subdividiendo las tareas en pequeñas unidades para que la cantidad de conocimiento a ser recordada sea mínima. Otra limitación es que la tarea puede ser variada de acuerdo al contexto en el que se realiza, por ejemplo una pericia es aplicada solo bajo circunstancias especiales, en este caso el ingeniero

del conocimiento deberá juntar todo el conocimiento relevante y no sólo el conocimiento aplicado a una tarea específica; de igual manera el experto debe ser instruido para verbalizar cualquier alternativa de decisión que salga fuera de lo común, así como aquellas que son aplicadas para la tarea en particular. Estos procesos permitirán al ingeniero del conocimiento aislar aquellos componentes de los comportamientos del experto que son suficientemente ambiguos para requerir más investigación durante la entrevista de adquisición de conocimiento.

- **Verbalización retrospectiva con una ayuda memoria**

La ayuda memoria permite recrear los procesos cognoscitivos empleados. Sirve de igual manera la videograbación del experto en pleno desarrollo de la tarea, para luego hacer las preguntas necesarias sobre determinados pasos específicos en su desarrollo.

2.3.9. Análisis de Tareas y Protocolos

2.3.9.1. Análisis de Tareas

Esencialmente es descomponer tareas en subtareas o pasos y analizar cada una de ellas para capturar la información y destreza requerida para ejecutarlas. La técnica consiste en llegar a la clasificación de los factores implicados en la solución del problema y la identificación de las tareas atómicas necesarias. Las categorías a aplicar a una tarea individual pueden incluir: tiempo necesario, frecuencia de ejecución, procedimientos usados, acciones empleadas, objetos utilizados, ratio de error, posición en la jerarquía de la tarea, etc. Esto implica que también es necesario identificar las acciones y conceptos de forma taxonómica. Por ejemplo, si se está estudiando el juego de póker, se puede comenzar con la estructura siguiente:

Conceptos: Cartas, Mesa, Mano, Jugador, Dinero, Jugada,...

Acciones: Barajar, Repartir, Descantarse, Ver, Apostar,...

Una forma de Análisis de Tareas supone que los demás conceptos son derivables a partir de un emparejamiento de las acciones con los objetos. Por ejemplo, Ver jugada, Repartir cartas. Una vez identificados los conceptos, es necesario identificar los planes

(ganar el juego, hace dinero) y las estrategias (faroles aleatorios, jugar a amarrar), y usar este análisis para identificar los conocimientos requeridos y usados equiparando pares concepto-acción con las descripciones de tareas que ocurren en la secuencias de la tarea.

2.3.9.2. Análisis de Protocolos (AP)

Es una técnica similar a la observación de tareas habituales. En este caso, no hay un intervalo entre el acto de pensar o actuar y el acto de reportarlo. En ambas técnicas existe alguien (usuario) que está desarrollando su comportamiento para efectuar una tarea normal con un problema específico. En el AP, además de registrar las sesiones, se capturan los protocolos (descripciones) en voz alta realizados por los usuarios (clientes) al enfrentar una situación particular del dominio para descubrir el proceso subyacente.

Este método es particularmente útil cuando los casos se han seleccionado previamente y, habitualmente, revela una gran cantidad de detalles interesantes. Sin embargo, las preguntas pueden interferir con la tarea y el interrogatorio se efectúa normalmente bajo presión, con frecuencia, el experto adoptará un enfoque atípicamente sistemático. Si el sistema debe realizar una emulación cognoscitiva del usuario, esto último podría ser un problema. Esta técnica puede utilizarse complementariamente con otras técnicas como Análisis de tareas.

Generalmente, va más allá de lo que un experto explícitamente contaría en una situación de solución de un problema para permitir información acerca de los conocimientos que está utilizando, pero que no pueden verbalizarse conscientemente, Es particularmente útil para extraer información sobre heurísticas que los usuarios utilizan en la solución de problemas, pero que no pueden explicar.

Resumidamente, la técnica consiste en grabar, en un protocolo, el comportamiento de los usuarios mientras trabaja en la solución del problema, y ese protocolo se transcribe y analiza para, finalmente, convertirlo en un conjunto de razonamientos que llevan a la solución del problema. Reconstruyendo esta solución pueden modelarse los conocimientos de los usuarios.

El protocolo puede ser de distintos tipos entre los que caben destacar: la copia literal, el parafraseado o copia del contenido semántico, la inferencia o generación de una verbalización a partir de la información utilizable, y el recuerdo o recuperación de la información previamente considerada.

El análisis de protocolos es adecuado para el tipo de tareas en que el pensar en voz alta puede ser una forma aceptable y útil de educación de conocimientos. Concretamente, aquellas tareas para las cuales la verbalización es una parte natural del pensamiento, son justamente las que se puede tomar el pensamiento en voz alta como datos. Es decir, si se produce la información verbal mientras alguien efectúa inferencias, o al identificar características sobresalientes de los objetos en una situación, entonces la información obtenida a partir de protocolos contiene informaciones aceptables. Sin embargo, hay otros tipos de tareas, por ejemplo, aquellas que usan en el proceso un lenguaje especial para las partes de la pieza que están describiendo o la sección que actualmente están instanciando, para las cuales el proceso de pensar en voz alta y explicarlo puede ser distorsionante o erróneo. Y, por descontado, hay tareas para las cuales no existe una verbalización natural, las tareas perceptivo-motoras son ejemplo de ello.

2.3.10. Metodología CommonKads (Kads: knowledge acquisition and documentation structuring)

Esta metodología está diseñada y concebida para el desarrollo de sistemas expertos basados en la ingeniería del conocimiento. Es un método que implica tecnologías organizativas, de seguimiento, documentación, control de proceso, y diseño para la construcción de Sistemas Expertos (SE), Basados en Conocimiento (SBC), similar a la metodología de diseños de sistemas tradicional denominada Ingeniería de Software (IS).

Surgió en el ámbito de la comunidad de investigadores de diversos países de la comunidad Europea alrededor de 1983 [Schreiber, A., 2000]. En este método existen tres conceptos fundamentales: modelado, reutilización y gestión del riesgo. Los mismos no son específicos de la Ingeniería del Conocimiento (IC), también se incluyen en el ámbito de la Ingeniería del Software en general.

La tecnología y métodos de CommonKADS cubren todos los espacios del desarrollo de software a través de un conjunto de tipos de modelos interrelacionados que permiten plasmar los rasgos más destacados del sistema y su medio. En el desarrollo de SBC utilizando CommonKADS se debe completar un conjunto de planillas estandarizadas, donde se conceptualizan los modelos. En función de ellas, CommonKADS establece estados de los modelos, que definen puntos destacados o críticos en el desarrollo. Estos estados establecen un camino dentro de la gestión del proyecto, donde el desarrollo se produce en forma cíclica (reiterada) dirigida por los riesgos.

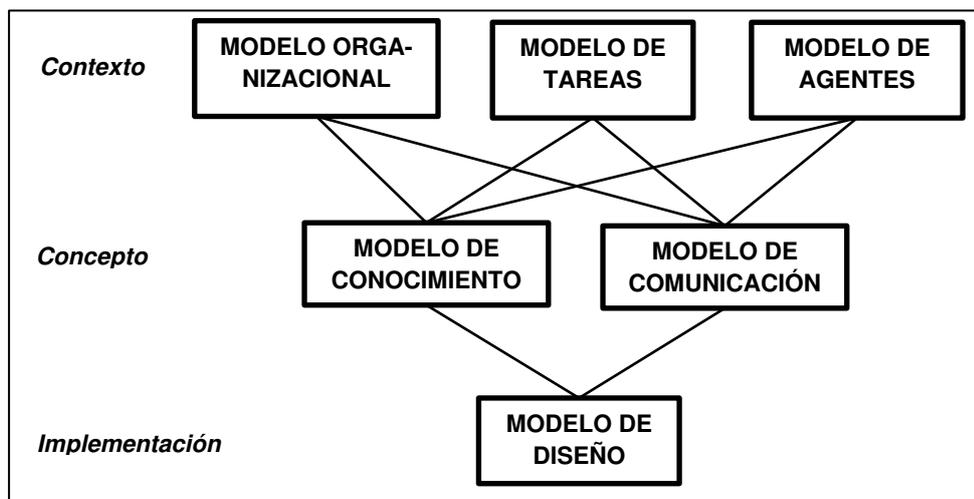
2.3.11. Estructura del modelo CommonKads

Los Modelos que rigen la estructura de la Metodología de CommonKADS son el modelo organizacional (OM), modelo de tarea (TM), modelo de agente (AM), modelo de comunicación (CM), modelo de conocimiento (EM) y responden a tres preguntas básicas:

1. ¿Por qué?, ¿Por qué el sistema de conocimiento es una potencial solución?, ¿Cuáles serán los beneficios, costo e impacto organizacional? La comprensión del contexto en la organización y el medio ambiente es el tema importante en este caso.
2. ¿Cuál?, ¿Cuál es la naturaleza y estructura de los conocimientos involucrados? La descripción conceptual del conocimiento aplicado es la tarea principal a responder con esta pregunta.
3. ¿Cómo?, ¿Cómo puede el conocimiento ser implementado en un sistema informático? ¿Cómo se observa la arquitectura de software y los mecanismos computacionales? Los aspectos técnicos del sistema computacional a construir es lo principal al responder esta pregunta.

Las respuestas a estas preguntas sirven para desarrollar (pieza por pieza) el modelo de conocimiento. CommonKADS tiene un conjunto predefinido de modelos, cada uno de ellos centrado en un aspecto limitado, pero juntos proporcionan una visión integral.

Figura 2. 2. Esquema de la metodología CommonKads



Fuente: CommonKads Methodology. Royal Institute of Technology. Stockholm, Sweden. 2010

2.3.12. Descripción de los Modelos en CommonKads

- **Modelo Organizacional (OM).** Este modelo analiza las principales características de una organización con el fin de averiguar los problemas y las oportunidades para el conocimiento del sistema. Además, incluye el estudio de viabilidad y los impactos de la intención del sistema de conocimiento en la organización.
- **Modelo de Tarea (TM).** Las tareas son los subprocessos de un proceso de negocio. Es necesario para el SBC, conocer el modelo de tareas, el análisis de la distribución de las tareas, su entrada y salida, condiciones previas y criterios de actuaciones y los recursos y competencias necesarios.
- **Modelo de Agente (AM).** Los agentes son los ejecutores de una tarea. Un agente puede ser un ser humano, un sistema de información, o cualquier otra entidad que sea capaz de realizar una tarea. Aquí se describe las características de los agentes, en particular, sus competencias, autoridad para actuar, y las limitaciones en este sentido. Por otra parte, se enumeran los enlaces de comunicación entre agentes en la realización de una tarea.

- **Modelo de Conocimiento (EM).** El objetivo del modelo de conocimiento es determinar los tipos y estructura del conocimiento utilizado en la ejecución de una tarea. También identifica todas las funciones de componentes del sistema de conocimientos que contribuyen en la solución de problemas. Esto hace que el modelo de conocimiento sea una herramienta importante para comunicarse con expertos y usuarios sobre las características de un sistema de conocimiento, tanto durante el desarrollo y el sistema de resolución de problemas.
- **Modelo de Comunicaciones (CM).** Como son varios los agentes que pueden participar en una tarea, es importante tener el protocolo de comunicación para presentar todas las transacciones entre los agentes. Esto se hace por el modelo de comunicación, de una manera conceptual y su implementación es independiente, al igual que con el modelo de conocimiento. La solución del problema se encuentra dividida entre dos conocimientos: Conocimiento de la metodología (aplicativo CommonKads) y el conocimiento del experto para la solución del problema.

Por otro lado en el conocimiento de la herramienta subyacen tres conocimientos diferentes:

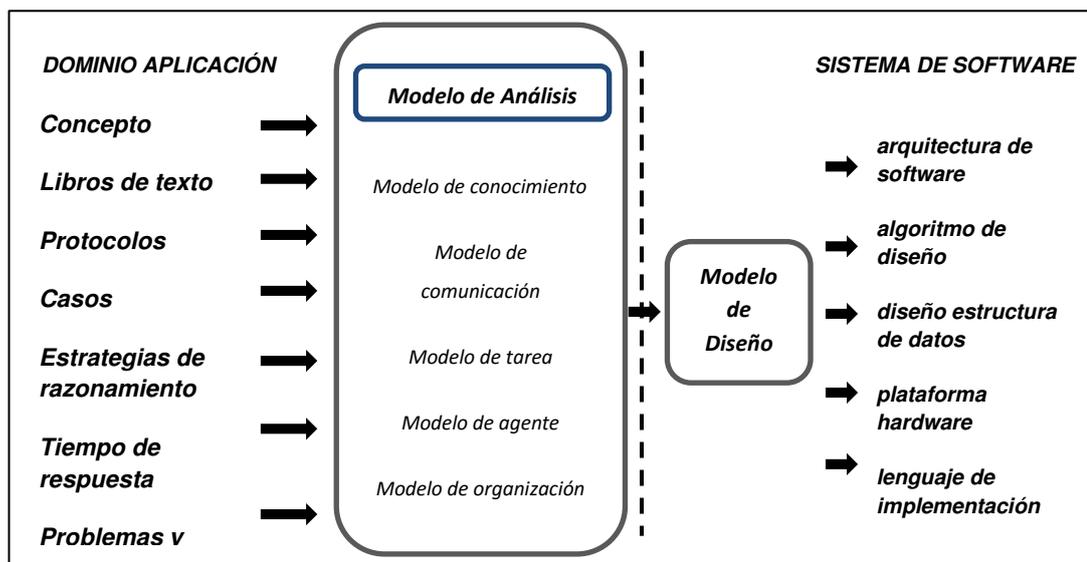
- a) Conocimiento sobre el dominio* (conceptos, relaciones y tipos de reglas).
 - b) Conocimiento de inferencia*, cómo esas estructuras que componen el dominio, pueden realizar un razonamiento. El proceso de las inferencias se expresa a través de una especificación declarativa de entrada y salida
 - c) Conocimiento de tareas*, el cual describe las metas y los procedimientos o métodos que se utilizarán para conseguirlas.
- **Modelo de Diseño (DM).** El modelo de diseño es el instrumento dentro de la tecnología de CommonKADS que se relaciona con el software y la organización que éste implica. Las planillas y documentación que en esta etapa se consideran están dirigidas a conceptualizar sobre el modelado de un sistema que contempla las siguientes características: a) se construyen las especificaciones del futuro sistema basadas en las especificaciones que resulten del modelo de conocimiento y comunicación, b) Se prepara una arquitectura

de sistema, c) se divide en cuatro pasos el diseño, donde cada uno de estos pasos genera una hoja de trabajo individual.

El modelo de diseño presenta la especificación del sistema de conocimientos en términos de la arquitectura, implementación de la plataforma, módulos de software, constructos representacionales y mecanismos computacionales que son necesarias para implementar las funciones en los modelos de conocimiento y comunicación.

Juntas, la organización, las tareas, y los modelos de agente, analizan el entorno de la organización y los correspondientes factores críticos de éxito para un sistema de conocimiento. Los modelos de conocimiento y de comunicación crean la descripción conceptual de las funciones de problema-solución y los datos que han de ser manipulados y entregados por un sistema de conocimiento. El modelo de diseño la convierte en una especificación técnica que es la base para la implementación del sistema de software. Cabe señalar, sin embargo, que no es necesario construir todos los modelos. Esto depende de los objetivos del proyecto, así como de la experiencia adquirida en la gestión del proyecto. Así, una elección sensata debe ser hecha por el administrador de proyectos.

Figura 2. 3. Etapas CommonKads: dominio-modelo-software de aplicación



Fuente: Dpto. de Ingeniería de Sistemas. Universidad Tecnológica Nacional. Córdoba

En consecuencia, un proyecto CommonKADS, produce tres tipos de productos o entregables, mostrados en la figura 2.3, los cuales son:

- Modelo de documentos CommonKads
- Información del administrador de proyectos
- Software del sistema de conocimiento

2.3.13. Conceptos Utilizados en la Metodología CommonKads

- ***Dominio (Domain):*** Dominio es algún *área de interés*. Ejemplo de dominio es el área de sistema de mantenimiento. El dominio puede ser jerárquicamente estructurado, por ejemplo el proceso de mantenimiento puede ser dividido en varios sub-dominios, tal como el análisis funcional, análisis de fallas, monitoreo basado en ciertas condiciones.
- ***Tarea (Task):*** Una tarea es una *pieza fundamental de trabajo a ser utilizada por un agente*, estas tareas deben ser escogidas de manera tal que el conocimiento juega un papel importante, por ejemplo, análisis de fallas funcionales, monitoreo basado en condición y pronóstico.
- ***Agente (Agent):*** Un agente es cualquier *humano o sistema de software que ejecuta una tarea* en un cierto dominio. Por ejemplo, un personal de mantenimiento puede llevar a cabo la tarea de diagnosticar los ruidos producidos por mal funcionamiento del sistema. Un sistema de conocimiento podría ser capaz de ejecutar la tarea de seguimiento del punzón de una máquina de corte.
- ***Aplicación (Application):*** Una aplicación es el *contexto que ofrece la combinación de un dominio y un tarea llevada a cabo por uno o más agentes*.
- ***Aplicación de dominio/tarea (Application domain/task):*** Estos dos términos se usan para referirse al dominio y/o tarea desarrollado en cierta aplicación.
- ***Sistema basado en conocimiento (Knowledge-based system):*** Este término (KBS) se ha utilizado por mucho tiempo y se deriva de la arquitectura de primera generación en la que los dos componentes principales son un motor de

razonamiento y una base de conocimientos. En los últimos años el término ha sido sustituido por el término "sistema de conocimiento" más neutral. Vale la pena señalar que no hay un límite fijo entre los sistemas de conocimiento y sistemas de software "normales". Cada sistema contiene el conocimiento en cierta medida; esto es cada vez más cierto en las aplicaciones de software modernos.

La principal distinción es que en un sistema de conocimiento se asume que hay alguna representación explícita del conocimiento incluido en el sistema. Esto plantea la necesidad de prestar especial atención a las técnicas de modelado.

- **Sistema experto (*Expert system*):** Se puede definir un sistema experto como un sistema de conocimiento capaz de ejecutar una tarea que en el caso de llevarse a cabo por un humano requiere experiencia. En la práctica el término a menudo se usa como un sinónimo de sistema basado en conocimiento.

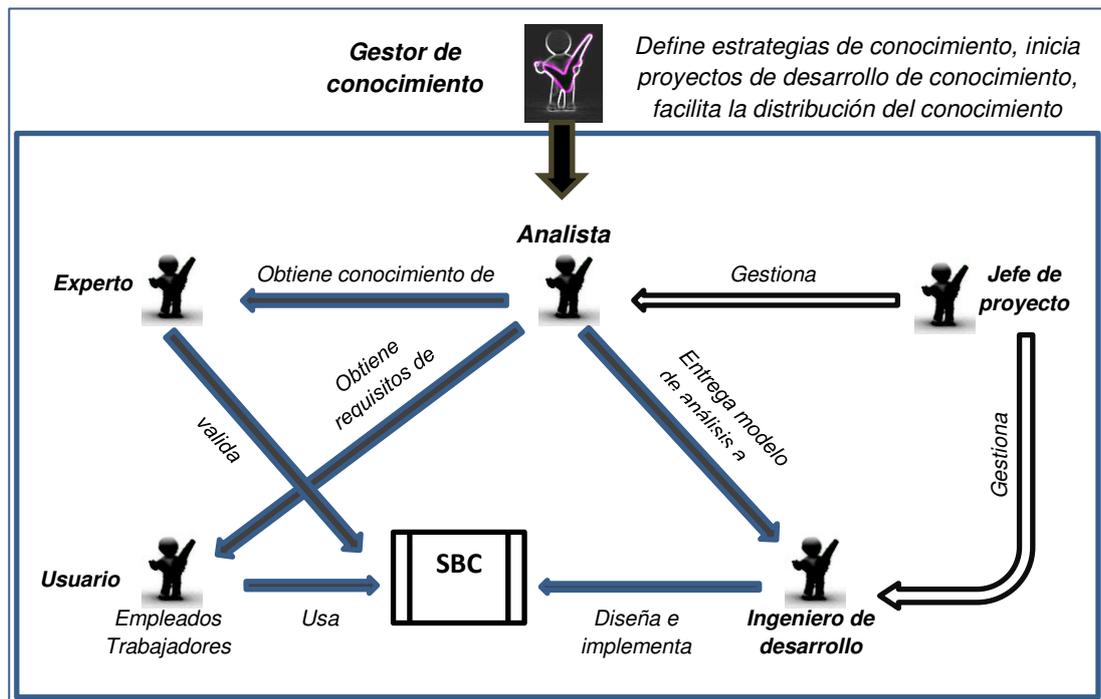
2.3.14. Impacto y Análisis de Mejora: Tarea y Modelado de Agente

El modelo de tareas se ocupa de la distribución global de tareas, su entrada y salidas, requisitos, criterios de rendimiento, los recursos necesarios y las competencias. La Figura 2.4 muestra los roles en el sistema base de conocimientos de mantenimiento.

Los diferentes roles en el diseño y mantenimiento de un SBC, son como sigue:

- **Conocimiento de usuario:** Es el que utiliza directa o indirectamente el conocimiento del sistema, pudiendo ser los empleados, trabajadores de producción, etc.
- **Jefe de proyecto:** Es el responsable por el desarrollo y mantenimiento del sistema.
- **Gestor de conocimiento:** Determina las estrategias de conocimiento en el ámbito del sistema, inicia las actividades de distribución y desarrollo de conocimientos.

Figura 2. 4. Descripción general de los diferentes roles de tareas en el diseño de un SBC



Fuente: CommonKads Methodology. Royal Institute of Technology. Stockholm, Sweden. 2010

En caso de que el resultado del estudio de factibilidad sea positivo, es el momento de dar el siguiente paso y centrarse en las características de las tareas pertinentes, los agentes que los llevan a cabo, y los conocimientos utilizados por los agentes en la ejecución de tareas. Por su descripción, CommonKADS ofrece los modelos de trabajo y de agente, ilustrados en la figura 2.5.

La noción de tarea también se ha convertido en un crítico de la teoría y la metodología de los sistemas de conocimiento y de intercambio de conocimientos y la reutilización. Por lo tanto, es necesaria una relación entre la noción de tarea en el sentido humano y organizacional de la palabra, y el de los sistemas de información. El modelo de tarea CommonKADS sirve como este pasador de articulación entre el aspecto de la organización y el aspecto del sistema de conocimiento de una tarea. Una tarea es una sub-parte de un proceso de negocio que considera lo siguiente:

Figura 2. 5. Descripción general del modelo de tareas en CommonKads



Fuente: CommonKads Methodology. Royal Institute of Technology. Stockholm, Sweden. 2010

- Representa una actividad orientada al objetivo de añadir valor a la organización.
- Maneja los insumos (inputs) y entrega resultados deseados de manera estructurada y controlada.
- Consume recursos.
- Requiere (y ofrece) el conocimiento y otras competencias respecto a la actividad.
- Se lleva a cabo de acuerdo con criterios de calidad y las prestaciones declaradas.
- Se lleva a cabo por los agentes responsables quienes rinden cuentas.

La tabla 2.2 presenta diferentes partes del modelo de tarea, las cuales deben ser determinadas y explicadas.

Tabla 2. 2. Análisis de tareas. Hoja de trabajo TM-1

Modelo de Tarea	Análisis de la Tarea. TM-1 (*)
Tarea	Identificador de área y nombre de la tarea
Organización	Indica la tarea como parte del proceso de negocio y en que parte de la organización se lleva a cabo (estructura, personas).
Metas y valores	Describe la meta de la tarea y el valor que su ejecución agrega al proceso en ejecución.
Dependencia y flujo	Input: tareas de entrada para su realización. Output: los resultados son utilizados (o parte de ellos) por que tareas. Puede utilizar un diagrama de flujo de datos o un diagrama de actividades para su descripción
Objetos manipulados	Input: objetos de entrada, incluida información y conocimiento, elementos que son de entrada a la tarea. Output: objetos, información y conocimiento que son entregados por la tarea, como salida. Objetos internos: Objetos importantes (si los hubiera) incluyendo información e ítes de conocimiento que son usados internamente dentro de la tarea, pero que no son ni de entrada ni de salida. Es posible que se desee incluir un diagrama de clases, para describir la información de objetos manipulados por la tarea.
Tiempo y control	Describe frecuencia y duración de la tarea Describe la relación de control con otras tareas, para esto se debe elaborar un diagrama de estado o un diagrama de actividades Describir el control de la siguiente manera: <ul style="list-style-type: none"> • Requisitos antes de ejecutar la tarea. • Post-condición que se debe obtener como resultado de la ejecución de la tarea.
Agentes	Los miembros del personal de los sistemas de información (cf. OM-2/3, personas) y / o (cf. OM-2/3, Recursos) que son los responsables de llevar a cabo la tarea
Conocimiento y competencia	Competencias necesarias para la ejecución de la tarea con éxito. Para el conocimiento de los elementos involucrados, hay una hoja de cálculo independiente TM-2. Haga una lista de otras habilidades y competencias pertinentes aquí. Indique cuáles elementos de la tarea son intensivas en el conocimiento. Tener en cuenta que las tareas también se pueden entregar como competencias a la organización, lo cual puede ser útil de indicar aquí.
Recursos	Describir y preferentemente cuantificar los recursos consumidos por la tarea (tiempo de personal, sistema y equipos, materiales, presupuestos). La descripción es básicamente un refinamiento de la descripción de los recursos en OM-2
Calidad y desempeño	Listar las medidas de calidad y desempeño que son usadas por la organización para determinar el éxito de la ejecución de la tarea.

Fuente: CommonKads Methodology. Royal Institute of Technology. Stockholm, Sweden. 2010

(*) TM: modelo de tareas – OM: Modelo de organización – AM: Modelo de agente

El tema de conocimiento y competencia, es un elemento significativo en el modelo de tarea, por esta razón se modela otra vez por medio de una hoja de trabajo separada TM-2, como se muestra en la tabla 2.3. También es útil la información desde el punto de vista de los agentes individuales. Esto se hace en el modelo de agente CommonKads que se ilustra en la tabla 2.4 como AM-1.

Tabla 2. 3. Análisis del ítem de conocimiento. Hoja de trabajo TM-2

Modelo de Tarea		Análisis del ítem de conocimiento. TM-2
Nombre :		Conocimiento del ítem
Ejecutado por :		Agente
Usado en :		Identificador y nombre de la tarea
Dominio :		Amplio dominio del conocimiento de la tarea inmerso en (especialidad, disciplina, rama de la ciencia o la ingeniería, la comunidad profesional)
Naturaleza del conocimiento		¿Cuello de botella / a mejorarse?
Formal, riguroso		
Empírico, cuantitativo		
Heurística, reglas de oro		
Altamente especializado, dominio específico		
Basado en experiencia		
Basado en acciones		
Incompleto		
Incierto, puede ser incorrecto		
Cambiante rápidamente		
Difícil de verificar		
Tácito, difícil de transferir		
Forma de conocimiento		
Mente		
Papel		
Electrónico		
Máscara de acciones		
Disponibilidad de conocimiento		
Limitaciones en tiempo		
Limitaciones en espacio		
Limitaciones de acceso		
Limitaciones en calidad		
Limitaciones en forma		

Fuente: CommonKads Methodology. Royal Institute. Technology. Stockholm-Sweden 2010

(*) TM: modelo de tareas – OM: Modelo de organización – AM: Modelo de agente

Tabla 2. 4. Ítem de conocimiento. Hoja de trabajo Agente AM-1

Modelo de Agente	Hoja de Agente. AM-1
Tarea	Nombre del agente
Organización	Indica la posición del agente en la organización, es heredado de la hoja de descripción OM, incluyendo el tipo (humano, sistema de información), función, posición en la estructura de la organización.
Participa en	Lista de tareas (ref. TM-1)
Comunicación con	Listado de los nombres de los agentes
Conocimiento	Listado de conocimientos perteneciente al agente.
Otras competencias	Listado de otras competencias presentes o requeridas del agente
Responsabilidad y restricciones	Listado de responsabilidades que el agente tiene en la ejecución de la tarea y de restricciones al respecto. Las restricciones pueden referirse a las de autoridad, pero también puede referirse a las normas legales o profesionales o de otro tipo.

Fuente: CommonKads Methodology. Royal Institute. Technology. Stockholm, Sweden. 2010
 (*) TM: modelo de tareas – OM: Modelo de organización – AM: Modelo de agente

2.4. Representación del conocimiento

La representación del conocimiento, es un campo de investigación que tiene como objetivo representar formalmente y organizar el conocimiento para utilizarlo y compartirlo con un conjunto de métodos y herramientas. En una representación formal, el conocimiento está representado por objetos lógicos vinculados por propiedades, axiomas y reglas, a la vez que automatiza diversos tratamientos en este conocimiento, siendo las ontologías las herramientas formales de representación de dominios de conocimiento complejos. Se han desarrollado sistemas de ayuda médica de ontología basados en el diagnóstico, por ejemplo, para el diagnóstico computarizado en emergencias dentales realizados por Charles Le Moing [Le Moing, 2009] y el sistema Oddin dirigido por García-Crespo [García-Crespo et al., 2010].

2.4.1. Ontología

Según Christopher Welty el término "ontología" fue tomado de una rama de la filosofía llamada Ontología, que es el estudio del ser en cuanto ser, o el estudio de lo

que existe es decir, todo el conocimiento que tenemos sobre el mundo [Welty & Guarino, 2001].

En el campo de la inteligencia artificial, el término ontología tiene una definición diferente, a continuación se va a mencionar las más relevantes:

- Robert Neches [Neches et al., 1991]: "Una ontología define los términos y las relaciones básicas con el vocabulario de un dominio, así como las reglas para combinar términos y relaciones para definir extensiones del vocabulario".
- Según Borst Willem [Willem Borst, 1997]: "Una ontología es una especificación formal de una conceptualización compartida".
- Rudi Studer [Studer, Benjamins, y Fensel, 1998] es una combinación de las definiciones de Thomas Gruber [Gruber 1993] y Borst Willem [Willem Borst, 1997]: "Una ontología es una especificación explícita y formal de una conceptualización compartida". Conceptualización se refiere a un modelo abstracto de un fenómeno en el mundo. Esta es una idea que se tiene de algo. Explícita significa que el tipo de conceptos y limitaciones de su uso se definen explícitamente. Establece que la ontología formal debe ser traducida a un lenguaje interpretable de máquina. Compartido significa que una ontología captura conocimiento consensual, es decir, no reservado para unos pocos individuos, pero compartida por un grupo o una comunidad de actores.

Una ontología es una lista finita de términos y relaciones entre estos términos. Estos términos corresponden a conceptos (clases de objetos) del dominio. Una ontología es también una forma de representar los objetos de manipulación informática. En esta tesis nos vamos a referir a la definición de "ontología" de Rudi Studer [Studer et al., 1998).

2.4.2. Aplicación de las ontologías

Las ontologías juegan un papel importante en las organizaciones y los sistemas para: (a) facilitar la comunicación entre los agentes a través del intercambio de conocimientos [Dolia, 2010], (b) proporcionar una base de conocimientos para los sistemas de ayuda a decisión [Bodenreider 2008], (c), proporcionar un repositorio

único para la integración de datos entre sistemas. También se pueden utilizar en el campo de la recuperación de información y procesamiento de lenguaje natural automático [Cardoso, 2007; Haav y Lubi, 2001]. De manera mas detallada podemos decir que el uso de ontologías ofrece muchas ventajas como:

- Proporcionar un marco para la representación y el intercambio de conocimientos a través de un vocabulario común. Proporcionan un formato de intercambio de conocimientos y un vocabulario específico que facilitan la comunicación y la interoperabilidad [Bodenreider, 2008].
- Facilitar la reutilización del conocimiento [Musen, 1992]. Es posible heredar las propiedades de los conceptos contenidos en dichas ontologías de alto nivel para el establecimiento de una ontología de nivel inferior. Por ejemplo, una ontología de dominio y ontología de la tarea o la aplicación, etc.
- Facilitar la integración de los datos [Bodenreider, 2008]. Se pueden utilizar como un repositorio de pivote durante las transferencias de datos entre sistemas que permiten la correspondencia entre las condiciones del sistema de origen y los términos de referencia (los de la ontología) antes de que se registre en el sistema destino.
- Proporcionar una base de conocimientos para el sistema de gestión de reclutamiento de pacientes elegibles para ensayos clínicos [Cuggia, et.al., 2011; Stenzhorn et al., 2010]. En efecto, en este tipo de sistema, las ontologías son las bases de conocimiento que contienen los criterios de elegibilidad y los grupos correspondientes a estos criterios. Luego se hace una unión entre los datos de los pacientes y de los criterios de elegibilidad presentes en la base de conocimientos. El principio es el mismo para los sistemas de apoyo para el diagnóstico médico [Bertaud-Gounot, 2011; Bodenreider 2008] porque, como en el caso de reclutamiento de pacientes para ensayos clínicos, las ontologías biomédicas se utilizan como bases de conocimiento. Las relaciones entre la enfermedad y los signos están representados y el sistema es en sí un mapeo de los datos o signos del paciente y el contenido de la base de conocimientos.
- Facilitar la automatización de la adquisición de conocimientos.

- Facilitar el mantenimiento del conocimiento declarativo, separándolo del de procedimiento y de control.

2.4.3. Tipología de las ontologías

Independientemente de lenguajes de representación, hay diferentes niveles de ontologías modeladas de acuerdo a la zona y, posiblemente, a las tareas para las que han sido diseñados [Burgun y Bodenreider, 2001].

- Las ontologías superiores (ontologías de nivel superior: ULO) categorías del grupo de alto nivel, es decir, los conceptos generales que son aplicables en todas las áreas, tales como "entidades", "eventos" , "estados", "proceso", "acciones", "tiempo", "espacio", "relaciones", "propiedades". Las ontologías superiores se supone que son universales, es decir, no deben referirse a áreas específicas. Deben ser polivalentes, es decir que no deberían haber sido diseñados para una tarea en particular.
- Ontologías generales, las cuales representan el conocimiento general, sin tener en cuenta las áreas o tareas específicas, con un nivel medio de precisión. También llamado meta-ontologías o núcleo-ontologías, que transmiten conocimientos genéricos, menos abstractos que las realizadas por las ontologías superiores, pero sin embargo, bastante general para su reutilización a través de diferentes dominios. Pueden enviar conocimiento factual (ontología de dominio genérico) o conocimientos para resolver problemas genéricos (conocimiento procedimental), propiedad o reutilizar en diferentes dominios (Ontología de dominio genérico). Dos ejemplos de este tipo de ontología son: (a) la ontología mereológica [Borst Willem, 1997) las relaciones con "parte-de", y (b) la ontología topológica que contiene las relaciones "Conectado a".
- Ontologías de dominio, son específicas de un dominio (por ejemplo, médico, electrónico, mecánico, etc.), pero independiente de una tarea en particular. Se supone que deben reflejar la realidad subyacente y la teoría del campo.
- Las ontologías de aplicación, las cuales se refieren a un ámbito limitado y para tareas específicas. Por lo general parten de una aplicación y contienen un número relativamente pequeño de conceptos, debe ser definido con cierto

detalle con las relaciones y reglas de inferencia para razonar sobre conceptos de la tarea. Esta ontología es la más específica. Los conceptos en la aplicación de esta ontología, a menudo corresponden a las funciones desempeñadas por las entidades del dominio en el desempeño de una actividad determinada [Maedche y Staab, 2001].

- Ontologías de referencia, mientras que las ontologías de aplicación están destinadas a un uso específico, las ontologías de referencia se desarrollan de forma independiente de cualquier aplicación para reflejar la realidad. Por definición, las ontologías de alto nivel son las ontologías de referencia, ya que están destinadas a formar la estructura de alto nivel de muchas otras ontologías (dominio o aplicación). Algunas ontologías de dominio pueden ser ontologías de referencia [Burgun, 2006]. Por ejemplo, "el Modelo Fundacional de Anatomía" (FMA) es una ontología de referencia de toda la anatomía humana. Se propuso como referencia para describir la fisiología y la patología [Rosse y Mejino, 2003]. Estas ontologías deben permitir diferentes tipos de aplicaciones, tales como la gestión de los datos heterogéneos a través de una normalización semántica de conceptos, razonando acerca de las entidades complejas con la realidad o el razonamiento sobre las entidades individuales de datos relacionados.

Por otro lado, las ontologías también se pueden clasificar de acuerdo al grado de integridad [Bachimont, 2000; Ikeda y Mizoguchi, 1998], es decir, el nivel de cobertura de la semántica del dominio estudiado. Bruno Bachimont, ofrece la clasificación en tres niveles [Bachimont, 2000]:

- Nivel ontología 1 - Semántica: Todos los conceptos (que se caracterizan por un término / etiqueta) deben cumplir con cuatro principios diferenciales: 1) El padre de la Comunidad; 2) Diferencia (especificación) del padre; 3) Comunidad con los conceptos hermanos (al mismo nivel); 4) Diferencia entre hermanos conceptos (de lo contrario no habría ninguna necesidad de definirlo). Estos principios corresponden a la participación semántica [Bachimont, 2000], que asegura que cada concepto tiene un significado contextual asociado y no único. Dos conceptos semánticos son los mismos si la interpretación de la

expresión / etiqueta, se traduce en una forma equivalente a través de cuatro principios diferenciales.

- Nivel ontología 2 - Repositorio: Además de las características que se especifican en el nivel anterior, repositorios de conceptos (o formal) se caracterizan por un término / idioma cuya semántica se define por un objeto de extensión. El compromiso ontológico, especifica los objetos de dominio que se pueden asociar con el concepto, de acuerdo a su significado formal. Dos conceptos formales serán iguales si tienen la misma extensión (por ejemplo, los "conceptos" de estrella: estrella de la mañana y la tarde están asociados a la instancia Venus⁶).
- Nivel ontología 3 - Operacional: Además de las características enumeradas en el nivel anterior, los conceptos de nivel operativo y computacional se caracterizan por las operaciones es posible aplicarlas para generar inferencias (compromiso computacional). Dos conceptos operacionales son iguales si tienen el mismo potencial para la inferencia.

El conocimiento producido por una ontología son transportados mediante los siguientes elementos [Gómez-Pérez, 1999]: 1) Conceptos; 2) las relaciones; 3) Funciones; 4) los atributos; 5) Los axiomas; 6) Las instancias.

2.4.4. Los conceptos

El conocimiento se utiliza en una ontología para relacionar los objetos a los que nos referimos a través de conceptos (o clases de la ontología) [Uschold y King, 1995]. El concepto representa un conjunto de objetos, seres y sus propiedades comunes. Los conceptos son las abstracciones relevantes de un segmento de la realidad (el dominio del problema, seleccionados sobre la base de los objetivos y el uso previsto de la ontología).

El concepto, también conocido como "intención" del concepto (concepto semántico) es el conjunto de características comunes a un cierto ente o cosa. Se expresa

⁶ De ello se interpreta que sigue o precede al sol en su curso en el cielo, por eso a veces es visible en la tarde, a veces a altas horas de la noche.

en términos de propiedades y atributos, reglas y restricciones. Por ejemplo, el concepto de "auto", la intensión incluye las características de "vehículo de carretera, por lo general de 4 ruedas y 4 asientos para un coche para 4 personas".

El conjunto de objetos, también llamado "extensión" del concepto es la colección de objetos a los que se aplica el concepto. Por ejemplo, el concepto de "auto", la extensión incluye "Mazda MX5 con el número de registro 2657ZB35", "coche verde estacionado en la esquina de la calle frente a mi oficina". Un concepto de este modo se proporciona con una semántica referencial (la impuesta por su extensión) y un diferencial semántico (la impuesta por su intensión). Los conceptos utilizados en un campo de conocimiento están organizados en una red de conceptos. En una ontología, todos los conceptos se estructuran jerárquicamente y están vinculados por las propiedades conceptuales. La propiedad se utiliza para estructurar la jerarquía de conceptos es subsunción.

Los conceptos se unen de la siguiente manera: Concepto C1 resume un concepto si cualquier propiedad semántica C1 es una propiedad semántica de C2, es decir, si C2 es más específico que C1. La extensión de un concepto es necesariamente menor que la de un concepto que resume. A continuación se presenta una lista de las principales propiedades que se pueden asociar con un concepto. La propiedad está en un concepto:

- **Genericidad:** un concepto es genérico si tiene, dentro de la jerarquía de los mismos una base general. Por ejemplo, el concepto de "enfermedad" es genérico; teniendo una connotación mayor o mas amplia que engloba a sus sub-conceptos.

Las propiedades de dos conceptos relacionados son las siguientes:

- **Equivalencia:** dos conceptos son equivalentes si tienen la misma extensión. Por ejemplo, los conceptos que protagonizan la estrella de la mañana y la tarde, están asociados con la extensión "Venus" (visto en nivel ontología 2).
- **Subsunción:** un concepto C1 subsume un concepto C2, si cualquier propiedad semántica C1 es una propiedad semántica de C2; es decir, si C2 es más

específico que C1. Por ejemplo, el concepto de "enfermedad" subsume el concepto de "enfermedad infecciosa".

- Incompatibilidad: dos conceptos son incompatibles si sus extensiones son disjuntas. Por ejemplo, los hombres y las mujeres.

2.4.5. Las relaciones

Si algunos vínculos conceptuales entre conceptos pueden expresarse mediante propiedades realizadas por conceptos (relación taxonómica), otros deben ser representados mediante relaciones autónomas. Una "relación" es una noción de asociación o vínculo que se establece entre los conceptos, por lo general expresada por un término o un símbolo literal en general u otro (o más generalmente por una señal). Una relación puede vincular las instancias de conceptos, conceptos con casos con valores (Cadena numérica de carácter, etc.) o conceptos genéricos.

Las relaciones traducen asociaciones (pertinentes) entre los conceptos presentes en el segmento analizado la realidad. Estas relaciones incluyen las siguientes asociaciones: (a) Subclase-de (generalización-especialización); (B) Parte-de (agregación o composición); (C) Asociado a; (D) Instancia-de, etc. Estas relaciones nos permiten ver la estructura y la interrelación de conceptos, el uno del otro. Las propiedades intrínsecas en una relación son:

- Propiedades algebraicas
 - Simetría: Relación definida en un conjunto E de tal manera que, para cada par de elementos (x, y) de $E \times E$, donde x está relacionada con y, a continuación, y está en relación con x. Por ejemplo, Paul "ama" Jeanne, Jeanne "ama" Paul.
 - Reflexividad: Relación por la cual cada elemento *a* de un conjunto está asociado consigo mismo.
 - Transitividad: Relación por la cual consecutivamente se conectan una serie de objetos a una relación entre el primero y el último. Por ejemplo, Pablo

"es el ancestro" de Jacques, Jacques "es el ancestro" de Marcos, Pablo "es el ancestro" de Marcos.

- Funcional: Para cualquier elemento x que pertenece a E , x está en relación con 0 ó 1 elemento y de F . Por ejemplo, Pablo "es padre" de María.

Las propiedades que vinculan dos relaciones son:

- Incompatibilidad: dos relaciones son incompatibles si no se pueden relacionar las mismas instancias de conceptos. Por ejemplo las relaciones, el "síntoma de" y "anatomía de" son incompatibles.
- Inversa: dos relaciones binarias son inversas entre sí, cuando existe una relación de dos instancias $I1$ e $I2$, y otra relación que une $I2$ e $I1$. Por ejemplo, la relación es "padre de" y "hijo de" son el inverso de la otra.
- Exclusividad: Dos relaciones son exclusivas entre sí, cuando una relación que une dos conceptos, automáticamente excluye otras posibles alternativas. La exclusividad es causa de incompatibilidad. Por ejemplo, la pertenencia y no pertenencia son exclusivas.

La propiedad que une las relaciones y los conceptos son:

- El enlace relacional (propiedad propuesta por G. Kassel): existe un vínculo relacional entre una relación R y dos conceptos $C1$ y $C2$ si para cualquier par de instancias de conceptos $C1$ y $C2$, existe una relación de tipo R entre las dos instancias de $C1$ y $C2$. Un enlace relacional puede estar limitada además por una propiedad de cardinalidad o asumir directamente una instancia de concepto [Kassel, 2012]. Por ejemplo, existe un vínculo de relación entre los conceptos de "texto" y "autor", por un lado y la relación "estaba escrito" en el otro. también conocido como dominio y co-dominio.

2.4.6. Las funciones

Las funciones son casos especiales de las relaciones, en la que un elemento de la relación, el n -ésimo (salida) se define de acuerdo a los $n-1$ elementos anteriores (entradas). Formalmente una función F se define por:

$$F: C_1 * C_2 * C_3 * \dots * C_{n-1} \rightarrow C_n$$

2.4.7. Los atributos

Los atributos describen las propiedades de instancias de una clase. Esos siempre se expresan en términos de tipo de datos o dominio.

2.4.8. Los axiomas

Los axiomas son afirmaciones, aceptadas como verdaderas, sobre las abstracciones del dominio como traducido por la ontología. Los elementos (conceptos, relaciones, valores, etc.) de la ontología deben revisar las condiciones con el fin de garantizar la coherencia y permitir inferir nuevos conocimientos.

2.4.9. Las instancias

Las instancias constituyen la definición extensional de la ontología, transmitiendo el conocimiento (sea estático, o de hecho) sobre el dominio.

2.4.10. Las reglas

Las reglas permiten enriquecer el conocimiento de la ontología. Estas reglas capturan las dependencias entre las propiedades definidas en la ontología.

Finalmente se puede decir que las ontologías son una herramienta adecuada para la representación, compartición y reutilización del conocimiento de un dominio. Sin embargo, para implementar ontologías de modo que sean legibles por los sistemas informáticos, se deben utilizar lenguajes de representación de conocimiento. Uno de los idiomas más conocidos y ampliamente utilizado por los actores de la web semántica es el "Lenguaje de Ontologías Web" (OWL) [Cardoso, 2007; McGuinness y Harmelen, 2004] que implementa las lógicas descriptivas (LD) [Baader, et.al., 2003].

2.5. Razonamiento Basado en Casos RBC

El RBC no es más que otro paradigma de resolución de problemas, pero son precisamente las diferencias con el resto de los métodos de la inteligencia artificial las

que lo hacen tan especial. En lugar de confiar únicamente en el conocimiento general del dominio del problema, o realizar asociaciones a lo largo de relaciones entre descripciones del problema y conclusiones, este paradigma es capaz de utilizar conocimiento específico de experiencias previas, es decir, situaciones de un problema concreto (casos). Un problema nuevo (al decir nuevo nos referimos a nunca antes tratado) es resuelto cuando se encuentra un caso pasado similar y se reutiliza en la situación del problema nuevo.

Una segunda diferencia, no por ello menos importante, es un acercamiento al aprendizaje incremental, sostenido, ya que se guarda una experiencia nueva cada vez que se resuelve un problema, pasando a estar disponible para futuros problemas desde ese mismo momento.

El Razonamiento Basado en Casos sugiere un modelo de razonamiento que incorpora los aspectos ya mencionados de resolución de problemas, entendimiento y aprendizaje e integra todo ello en meros procesos de memoria. En resumen, estas son las premisas subyacentes al modelo:

- La referencia a casos pasados es interesante y de gran utilidad para tratar situaciones que vuelven a darse. Por ello, recordar un caso para usarlo en un problema futuro (e integrar ambos) es, necesariamente, un proceso de aprendizaje.
- Debido a que las descripciones de los problemas son, a menudo, incompletas, es necesario una etapa de entendimiento o interpretación., ya que no puede llevarse a cabo un razonamiento, una resolución adecuada de una nueva situación, si ésta no se entiende con cierta completitud. Se puede considerar que esta etapa es a la vez un prerrequisito y una parte del ciclo de razonamiento, pues el entendimiento de las situaciones mejora conforme progresa el razonador.
- La práctica demuestra que no suele existir un caso pasado exactamente igual que un caso nuevo. Por ello, es muy usual el tener que adaptar la solución pasada para que se ajuste a la nueva situación.

- El aprendizaje es una consecuencia natural del razonamiento. Si se halla un nuevo procedimiento en el curso de la resolución de un problema complejo y su ejecución resulta positiva, entonces se aprende el nuevo procedimiento para resolver esta nueva clase de situaciones.
- La revisión de la solución propuesta y el análisis de la revisión son dos partes necesarias para completar el ciclo de razonamiento/aprendizaje. Este análisis de la revisión (habitualmente llevada a cabo por un agente externo, léase humano) puede conllevar una reparación de posibles fallos.

Estas premisas sugieren que la calidad de un razonador basado en casos depende de:

- 1- La experiencia que tiene.
- 2- La habilidad para entender situaciones nuevas en términos de experiencias pasadas.
- 3- Su capacidad de adaptación.
- 4- Su capacidad de evaluación y reparación.
- 5- Su habilidad para integrar nuevas experiencias en su memoria adecuadamente.

2.5.1. Breve Historia del RBC

Existen varias referencias en lo que respecta a los inicios del RBC; sin embargo todas convergen en el trabajo realizado por el grupo de investigación de Roger Universidad de Yale [Schank, R. y Abelson, R. P., 1977] a principios de los ochenta, quienes construyeron el primer modelo cognitivo y la primera aplicación sustentada en RBC. Tomaron como punto de partida la idea de que el conocimiento humano sobre distintas situaciones se guarda en la mente en forma de recuerdos y son éstos los que utiliza el ser humano para llegar o inferir conclusiones. Luego, la investigación de Robert Schank, [Schank, R., 1983] concluyó en el primer modelo de memoria dinámica.

Partiendo de estos conocimientos, posteriormente fue Janet Kolodner la que desarrolló el primer sistema RBC [Kolodner, J., 1983]. Cyrus Vance, ex-secretario de los Estados Unidos, utilizó este sistema al cual pusieron su nombre, CYRUS, el cual

contenía en cada caso información sobre sus viajes y reuniones; luego apareció el sistema CASEY [Koton, P. 1989] con este mismo modelo de memoria, el cual generaba explicaciones sobre los síntomas de pacientes cardíacos; y otros sistemas en la década de los ochenta como JULIA y PERSUADER, orientados al diseño de menús y generación de soluciones para negociaciones laborales aceptables para ambas partes, respectivamente. Más tarde, Bruce Porte [Porte, B. y Bareiss, E., 1986] de la Universidad de Texas, creó el sistema PROTOS, como una alternativa al modelo creado por Schank el cual utiliza clasificaciones heurísticas y métodos de aprendizaje de máquina, aplicados al aprendizaje de trastornos al oído.

Hasta en el entorno legal, se ha utilizado el RBC, ya que la práctica de la ley está basada en la consideración de precedentes y en la noción de caso. Así, en la Universidad de Massachusetts se desarrolló HYPO, sistema que genera argumentos legales, citando casos pasados a favor y en contra como justificación de sus argumentos. Posteriormente, este sistema fue combinado con razonamiento basado en reglas, produciendo CABARET.

También en Europa, en el Reino Unido, se generaron aplicaciones de RBC a la ingeniería civil. En la Universidad de Salford lo aplicó a diagnósticos fallidos, para rehabilitación y reparación de edificios así como a la construcción [Yans, S. y Robertson, D., 1994]. En España, cabe destacar los siguientes grupos, GAIA-Group for Artificial Intelligence Applications (Universidad Complutense, Madrid), BISITE-Bioinformática. Sistemas Inteligentes, Tecnología Educativa (Universidad de Salamanca). eXiT-Ingeniería de Control y Sistemas Inteligentes. (Universidad de Girona). Instituto de Investigación en Inteligencia Artificial-IIIA (Consejo Superior de Investigaciones Científicas, CSIC), entre otros.

Se puede decir, pues, que el RBC es una técnica estudiada ampliamente a nivel internacional y con vastas aplicaciones en diferentes dominios que van desde el control de tráfico, finanzas, clasificación de clientes, diagnóstico de enfermedades, filtro de spam, agricultura, etc. Puede decirse que el éxito del RBC se debe principalmente al incremento del número de aplicaciones comerciales de RBC con éxito y además a las múltiples investigaciones que sobre este tema se han hecho y siguen haciendo.

Un registro es una información estática, sin ningún objetivo en particular, tarea o acción asociado a este; en cambio, un *caso* comprende una información activa, en el sentido que a él hay asociada una experiencia en tiempo real. Un *caso*, se define como “Una pieza de conocimiento contextualizada, representado una experiencia que muestra una lección fundamental para obtener un objetivo razonable”.

2.5.2. Funcionamiento de un RBC

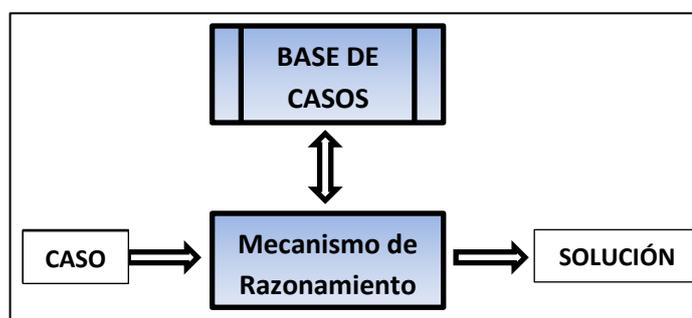
El RBC es un sistema que resuelve problemas haciendo uso de la experiencia; así, cuando el sistema se enfrenta a un problema, recuerda soluciones que funcionaron bien en situaciones similares, utilizándolas como punto de partida en la resolución de dicho problema. Un típico ejemplo de como trabaja el RBC es el diagnóstico médico, el galeno tiene una serie de experiencias almacenadas, que se corresponden con las enfermedades de los pacientes tratados en el desempeño de su función, de manera tal que cuando un nuevo paciente es tratado, lo compara con los síntomas similares de otros casos tratados anteriormente, pudiendo entonces dar el mismo tratamiento o mejorarlo, agregando nuevo conocimiento a su experiencia.

Un sistema de RBC, resuelve problemas adaptando soluciones de problemas antiguos, esto implica razonamiento a través de la experiencia, reteniendo problemas en la memoria con sus respectivas soluciones, para luego resolver los nuevos problemas haciendo referencia a ese conocimiento almacenado. Generalmente cuando un sistema RBC encuentra un nuevo problema, éste buscará en su memoria problemas pasados en lo que se llama *base de casos*, e intentará encontrar tal problema al que llamamos *caso*, cuyas características sean iguales o similares al del problema tratado. Si el sistema no puede encontrar un caso idéntico, buscará luego uno o varios que sean los más parecidos posible al caso actual. En situaciones donde el caso recuperado es idéntico al problema planteado, y asumiendo que la solución con la que se almacenó fue un éxito, ésta es propuesta como solución para resolverlo. Si por el contrario, lo recuperado no es un caso idéntico, entonces se adapta la solución. Para ello se identifican las diferencias entre ambos casos y se modificará la solución del caso recuperado tomando en cuenta tales diferencias.

La estructura de un sistema de RBC a un nivel alto de abstracción se muestra en la figura 2.6, incorporando los siguientes factores externos:

- Los detalles de entrada del problema o *caso*.
- La salida del sistema o *solución*.
- La memoria de los casos pasados o de *base de casos*, la cual es utilizada por el *mecanismo de razonamiento*, para obtener a solución correcta.

Figura 2. 6. Funcionamiento de un RBC



En cuanto a la estructura interna del mecanismo de razonamiento, al que comúnmente se le denomina razonador, está dividida en dos partes principales que son: *el recuperador de casos* y *el razonador de casos*, la tarea del primero es encontrar el caso más apropiado en la base de casos, mientras que la del segundo, es usar el caso recuperado para encontrar la solución al problema. Cuando el caso en estudio coincide con el caso recuperado, no es necesario ningún razonamiento, debido a que el caso recuperado contiene directamente la solución del caso actual, y desde luego carece de importancia en el proceso interno de funcionamiento del RBC. Sería un caso extremo de entrada en el que el RBC tendría poco que aportar.

El proceso de razonamiento dentro del sistema de RBC posee un ciclo que consta generalmente de cuatro etapas, más conocido con el nombre de *ciclo del RBC*: *Recuperar (retrieve)*, *Reutilizar (reuse)*, *Revisar (revise)* y *Retener (retain)*, que se observa en la figura 2.7

2.5.4. Representación e Indexación de Casos

La información que se tenga de un caso, el cual es una porción de conocimiento, dependerá del contexto en el que sucedió, por lo tanto en RBC será necesario guardar dicho contexto en el que se produjo la experiencia, ya que será la base para determinar cuándo es aplicable ese conocimiento, además es muy importante para el proceso, la forma en que se represente un caso así como su indexación en la base de casos, teniendo en consideración lo siguiente:

- La estructura y la representación de los casos
- El modelo de memoria usado para organizar la base de casos
- La selección de los índices usados para identificar cada caso, si los casos fueron indexados.
- **Representación de los casos**, es la forma en que es guardada la información que se le proporciona al sistema, tras algún caso de generalización entre los casos que ya había en memoria y que se le conoce como *proptotipo*. Un prototipo, requiere menor cantidad de memoria, al almacenarse solamente un caso cuando no se asemeje a otro existente. En general, se dice que un caso queda representado, cuando de él se conocen las características siguientes:
 - La descripción del problema, es decir, los objetivos y/o restricciones a satisfacer.
 - Las soluciones del problema.
 - Los datos iniciales.
 - Un esbozo que indique como se ha llevado cabo la resolución de caso, si fuera necesario.

Los métodos de representación de casos se clasifican en tres categorías principales: atributo de valor, estructuras y textuales. Ésta es una de las características importantes del sistema de RBC, porque permite almacenar las experiencias anteriores de forma estructurada o no.

Asimismo, los casos pueden ser representados en diferentes tipos de conocimientos y dependiendo del sistema RBC, podrían representar a las personas, objetos, situaciones, diagnósticos, diseños, planos o resoluciones entre muchas otras representaciones [Shiu, 2004].

- **Indexación de los casos**, consiste en asignar uno o varios índices a cada caso para facilidad de la recuperación del mismo, etapa importante cuando la librería de casos es grande, si no lo es, se podría hacer simplemente secuencial.

El establecimiento de índices no es una tarea sencilla, sin embargo, es posible utilizar técnicas automáticas de indexación. Los procesos automáticos de indexación han demostrado ser altamente predictivos, siempre y cuando exista un conjunto de características muy bien definidas o cuando se trabaje en dominios reducidos [Juárez, 2005].

2.5.5. Representación del Conocimiento en RBC

El modelo de contenedor de Richter [Richter, 2005] se muestra en la figura 2.8 y es conocido como el método de estructura del sistema RBC.

Hay en los cuatro contenedores de conocimiento:

- La **Base de Casos** conteniendo los casos
- La **Medida de Similaridad** contiene las diferentes medidas de similaridad
- La **Adaptación del Conocimiento** (también conocido como la *Solution Transformation*), contiene la información necesaria para adaptar una solución de un caso antiguo de la base de casos, para adaptarse mejor al caso problema.
- El **Vocabulario** contiene toda la información acerca de los atributos y que estructura de datos se usó para representarla.

Figura 2. 8. Contenedor de conocimiento según Richter



Fuente: [Richter, M.M., 2005]

La información que se tenga de un caso, el cual es una porción de conocimiento, dependerá del contexto en el que sucedió, por lo tanto en RBC será necesario guardar dicho contexto en el que se produjo la experiencia, ya que será la base para determinar cuándo es aplicable ese conocimiento, además es muy importante para el proceso, la forma en que se represente un caso así como su indexación en la base de casos, teniendo en consideración lo siguiente:

2.5.6. Recuperación de los casos

En esta etapa, se busca dentro de la base de casos, aquellos semejantes al caso actual, comparando sus características con los del problema considerado como un caso entero, si no existiera un caso que coincidiera completamente, es suficiente con solo comparar una porción del caso. La recuperación de un caso es altamente dependiente, tanto del modelo de memoria como de la forma en que los casos están indexados, pudiendo utilizarse varios métodos desde los ya conocidos, como el del vecino más cercano hasta otros más complejos. Luego de la recuperación del caso, se realiza un análisis para determinar si éste es suficientemente parecido al caso planteado.

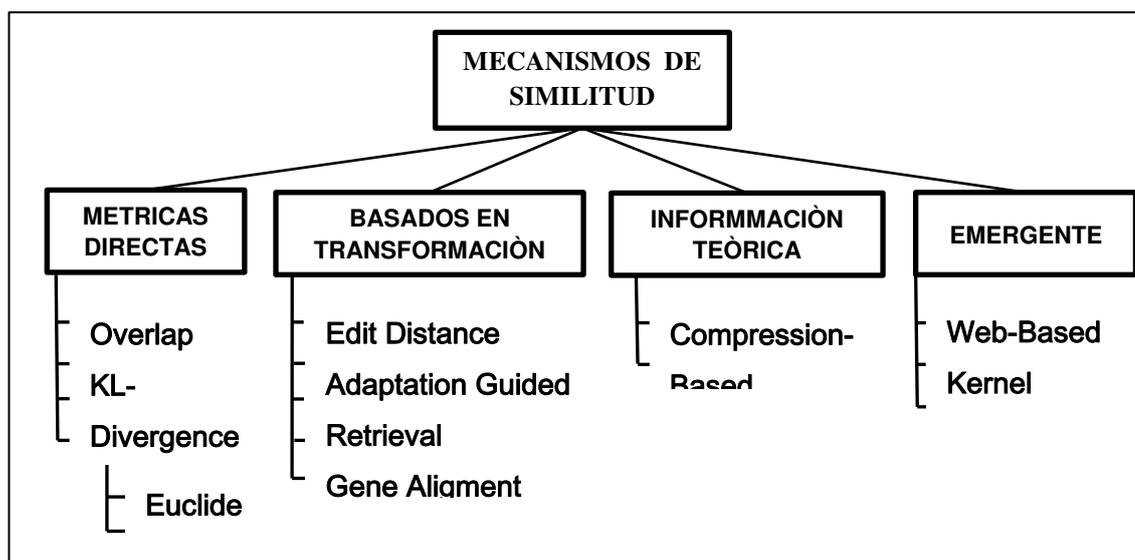
La similitud puede ser interpretada de diferentes maneras: el grado de relación de las características, la situación contextual o la dificultad que supondría su fusión

(adaptación). La medida de similitud es el grado de cercanía existente entre los atributos que componen el caso y nos permitirá cuantificar sus correspondencias parciales. De esta manera, las medidas de similitud son las herramientas de las que dispone un Sistema RBC para cuantificar cuan similares son los casos. Así, una medida de similitud contiene conocimiento, más o menos sofisticado, sobre la forma de cuantificar dicha similitud, tradicionalmente entre pares de casos.

2.5.7. Mecanismos de similitud en RBC

Cuando se observa dos casos diferentes; es decir, el caso consulta y el caso de la base de casos se tiene que formalizar el modo de representar cuan similares son el uno del otro. La similitud entre los dos casos es a menudo calculada por la similitud de sus características individuales con los pesos de cada una de ellas. Una taxonomía ha sugerido dividir los mecanismos de similitud en cuatro categorías diferentes [Cunningham, 2009]. Estas categorías y subcategorías se ilustran en la figura 2.9

Figura 2. 9. Taxonomía de los mecanismos de similitud



Fuente: Cunningham, P. Knowledge and Data Engineering, (Volume: 21 , Issue: 11)

Los mecanismos de similitud son altamente dependientes de la representación de los valores de las características. Un caso puede variar de un vector de características de simples estructuras de casos hasta otro muy complejas. Cunningham categoriza representaciones de casos en tres tipos; 1: representación de un valor característico, 2: representación estructural y 3: secuencias y cadenas.

1. Construcción de valores numéricos simples, en un Rango normalizado de cero a uno. Estos valores pueden ser representaciones internas o externas.
2. Los valores característicos de los casos pueden tener una estructura interna más sofisticada que un vector característica, tal como una estructura jerárquica, una red semántica o alguna estructura no atómica.
3. Un agrupamiento de palabras (texto libre) o una secuencia de palabras/cadenas.

Mecanismos directos. Esta es una estrategia dominante de acuerdo a Cunningham. En la mayoría de los casos las características deben ser representadas por vectores y la similitud es calculada directamente en base a ellos. La técnica del vecino más cercano (k-nearest) es usada casi siempre para encontrar los k casos más similares, entonces a clase se determina de alguna forma (por ejemplo, la mayoría de los votos o una distancia ponderada de los votos, de los electores respecto a un candidato). Este mecanismo tiene una ventaja computacional, se ata a una representación característica de los casos.

Mecanismos Basados en la Transformación. Ofrece varias formas de determinar la similitud de los casos. Una de ellas es en base a la similitud de comprensión, trabajando directamente con los casos en bruto. Si dos documentos/casos son muy similares, la compresión de éstos no será mucho mayor que el tamaño comprimido de un solo documento/caso. La información basada en la similitud, tiene alguna de las mismas características que la similitud basada en la transformación.

Mecanismos de Medida Emergente. Incluyen diferentes formas de utilización de la máquina de aprendizaje para encontrar similitud, utilizando el poder de procesamiento significativo para producir una caracterización de los datos. Bosques aleatorios y núcleos de racimo son del tipo interno, y descubren nuevas relaciones de análisis dentro del conjunto de datos, donde el primero de ellos por lo general crea una gran cantidad de árboles sin podar decisiones que trabajan juntas, luego una técnica los ensambla en conjunto. Los núcleos de racimo son algoritmos de aprendizaje semi-supervisado. Núcleos basadas en la Web son externos al traer nuevos conocimientos desde fuera del conjunto de datos mediante la búsqueda en la web.

2.5.8. Adaptación y evaluación de los casos

En esta etapa, se procesa la solución recuperada para transformarla en una apropiada, con el objetivo de resolver el problema, esto es lo que le añade inteligencia, sino sería solamente una simple comparación de emparejamientos. Hay dos formas generales de hacer adaptación:

- Mediante la sustitución de aquellos valores que aparecen en el caso recuperado de la memoria, por aquellos otros valores del caso actual, de forma que la nueva solución hace uso de la situación actual que se requiere resolver.
- Mediante la aplicación al caso que hay que resolver, del mismo conjunto de procedimientos, reglas o inferencias utilizados como solución del caso recuperado.

Si hubiere un conjunto de casos candidatos recuperados, se debe escoger uno de ellos mediante algún criterio o de otro modo, usar las soluciones conjuntas en la búsqueda de una apropiada para el caso en estudio. Posteriormente a la adaptación, se evalúa el caso ya sea por un experto o de manera semiautomática.

Una vez elegida una posible y antes de completar la etapa de adaptación, se debe comprobar si esta solución adaptada tiene en cuenta las diferencias entre el caso recuperado y el problema actual, si no fuera el caso, en esta etapa se debe considerar la decisión a tomar si la solución propuesta no resuelve el problema con éxito.

Para ilustrar lo anterior, consideremos una base de datos para una librería, conteniendo cientos de registros (libros, journals, artículos, etc.); para cambiar uno de estos registros dentro de un caso, se requiere tener un experto el cual utiliza su experiencia para el almacenamiento, tal como un estudiante accede a una librería en busca de por ejemplo, una novela o un libro de ciencia ficción; en ese contexto, el estudiante debe tener una estrategia de búsqueda (ej. año de publicación, autor, etc.) para obtener un resultado o retroalimentación y registrar luego dentro de sí mismo la satisfacción o no.

2.5.9. Aprendizaje y mantenimiento de la base

- *Aprendizaje*

Durante el aprendizaje, el sistema necesita de algún juicio capaz para valorar el rendimiento de la prueba, luego el sistema se renueva con la información obtenida acerca de la solución, de este modo el sistema va aprendiendo. Esta información ya almacenada aumenta la probabilidad de encontrar un caso igual o parecido haciéndose el sistema más robusto o completo. Así, el aprendizaje puede ocurrir no sólo cuando el caso ha sido solucionado con éxito, sino también al darle la posibilidad, si el sistema así lo permite, de almacenar las causas de error.

- *Mantenimiento de la base de casos*

En esta etapa se añade, revisa o elimina casos, aunque también incluye cambios en el conocimiento. En el sistema RBC hay una relación directa entre el número de casos almacenados en la librería de casos almacenados y la eficiencia en su recuperación, dependiendo en mucho del sistema que debe eliminar los casos redundantes para obtener una mayor eficiencia. Existen diversos métodos para un correcto mantenimiento de la base de casos [Leake, D., 2000] y [Smyth, B. 1998].

2.5.10. Ventajas y Desventajas de usar un Sistema RBC

El RBC presenta múltiples ventajas, como son:

1. Propone soluciones rápidamente sin tener que crearlas desde su inicio.
2. Evita cometer errores anteriores.
3. El provecho está en la adquisición de conocimiento.
4. Propone soluciones en dominios complicados.
5. A través del conocimiento incierto llega a un razonamiento útil.
6. Su razonamiento infiere de forma parecida a cualquier expertise.

7. Aprende y previene soluciones erróneas.
8. Responde a situaciones poco comunes.
9. El sistema permite que se especialice en el dominio en el cual trabaja.
10. Parte de soluciones globales, pudiendo descomponer el problema en sub-problemas para luego unir las soluciones parciales.
11. Los casos son de utilidad para interpretar conceptos no definidos claramente.
12. Permite difundir conocimiento experto.

Algunos autores argumentan que el RBC acepta evidencia anecdótica, por lo que la generalización no podría ser la correcta. Aquí algunas desventajas:

1. El sistema puede utilizar casos, sin comprobar si las soluciones son válidas.
2. El razonador puede ser influenciado por los casos previos.
3. Los datos de entrada pueden ser escasos para el tratamiento de un problema.
4. El sistema puede ser inexperto.

CAPÍTULO III

METODOLOGÍA

3.1. Diseño Metodológico

La presente tesis está enmarcada dentro del desarrollo tecnológico, por lo tanto está orientada a la obtención de un producto tangible, que para nuestro caso es un modelo de razonamiento basado en casos.

3.1.1. Tipo

El tipo de investigación será retrospectivo y transversal.

3.1.2. Enfoque

El enfoque metodológico que fue usado es el descriptivo-analítico, no experimental, además de cuantitativo al cuantificar las puntuaciones de acierto (porcentaje) que el modelo proporciona al evaluar la certeza del diagnóstico a un paciente y compararlo con el grupo de caso base.

3.2. Taxonomía

Considerando la Association for Computing Machinery (ACM), esta tesis se enmarca dentro de la clasificación siguiente:

PROGRAMA: C.03.	LÍNEA DE INVESTIGACIÓN
Tecnologías de la información y comunicación	C.0.3.17 Reconocimiento de patrones C.0.3.19 Informática y sociedad

3.3. Población y muestra

3.3.1. Población

Para nuestro estudio se escogió como población objetivo a los habitantes del ámbito rural de la Comunidad Campesina Cruz de Mayo, formada por 13 sectores y que según el último censo de población tiene 3700 habitantes dentro de una extensión de tierras comunes que superan las 4000 hectáreas. Se consultó con los médicos, cuál o cuáles eran las enfermedades localizadas en esta región, llegando a la conclusión que la malaria y leishmaniasis, aunque no están en un porcentaje extremo, pues sólo afecta al 1% y 0,1% de la población de Huaraz respectivamente; sin embargo, por el mismo hecho que su sintomatología es muy penosa y puede conllevar a la muerte, desearían exterminarlas totalmente. Es así que se tomó como característica de estudio a estas dos enfermedades, para desarrollar nuestro modelo de razonamiento basado en casos.

3.3.2. Tamaño de muestra

Considerando el párrafo anterior, el tamaño de la muestra estará dada en función de la siguiente fórmula:

$$n_0 = \frac{\frac{t^2 P \cdot Q}{d^2}}{1 + \frac{1}{N} \left(\frac{t^2 P \cdot Q}{d^2} - 1 \right)}$$

donde:

t : Es la abscisa de la curva normal que corta un área de α en las colas de la distribución.

P : Proporción estimada de los pobladores de la Comunidad que padecen las enfermedades de malaria y leishmaniasis.

Q : Proporción estimada de los pobladores de la Comunidad que no tienen calidad que no padecen dichas enfermedades

d : Error de estimación

Para nuestro estudio se tiene los siguientes datos:

$t = 1,96$ Para un nivel de significancia del 5%, en la distribución normal

$P = 0,01$ Sólo el 1% padece malarico o leishmaniasis

$Q = 0,99$ El 99% están sanos o padecen otra enfermedad

$d = 0,03$ Margen de error muestral

$N = 3700$ Población total de la comunidad, estimada al año 2013

$$n = \frac{\frac{t^2 P \cdot Q}{d^2}}{1 + \frac{1}{N} \left(\frac{t^2 P \cdot Q}{d^2} - 1 \right)} = \frac{\frac{1,96^2 (0,01)(0,99)}{0,03^2}}{1 + \frac{1}{3700} \left(\frac{1,96^2 (0,01)(0,99)}{0,03^2} - 1 \right)} = 41,79 \cong 42$$

3.3.3. Muestra

La muestra recolectada, fue a través de un muestreo no probabilístico por conveniencia, en la que se tomó como referencia un tipo de enfermedad (malaria y posteriormente leishmaniasis), escogiéndose a los sujetos en estudio que padecen la misma. Se tomaron la totalidad de casos: 40 casos de malaria, 5 de leishmaniasis; y, por motivos de ilustrar esta investigación, cargamos además 5 pacientes indicados como sanos en su historial médico. Por lo tanto excedimos en 8 pacientes a la muestra indicada.

3.3.4. Marco muestral

Para la toma de información se tomó como marco muestral los padrones de historias clínicas de la Comunidad Campesina Cruz de Mayo, correspondientes a los puestos de salud de los centros poblados de Antash, Llacsu, Huauya y Pampacocha.

3.4. Entornos de desarrollo a utilizar

Los siguientes entornos de desarrollo y softwares han sido utilizados durante el desarrollo de la presente tesis.

- **Protégé⁷**

Es un software open source, editor de ontologías y base de conocimientos. El entorno está escrito en Java y es compatible con varios plug-ins.

- **jColibri⁸**

Es una plataforma para desarrollar software orientado al razonamiento basado en casos, escrito completamente en Java.

- **MyCBR⁹**

Es una herramienta de código abierto, construido en base al software Protégé y permite la creación fácil y rápida de prototipos para un sistema RBC.

3.4.1. Protégé

Es un editor de ontologías de código abierto y un sistema de adquisición de conocimiento, desarrollado por la Universidad de Stanford, en colaboración con la Universidad de Manchester. Este software proporciona una interface gráfica de usuario para definir ontologías e incluye clasificadores deductivos para validar los modelos consistentes e inferir información nueva basada en el análisis de una ontología. La aplicación está escrita en Java y usa fuertemente swing para crear su completa interfaz, actualmente tiene más de 240000 usuarios inscritos en su página web.

La primera versión de Protégé fue creada en 1988 y fue como resultado de Mycin y ONCOCIN [Grosso, W.E. et al., 1999], actualmente está en la versión 5. Fue originalmente diseñado para facilitar el proceso de adquisición de conocimiento al permitir al usuario escribir dominio de conocimiento de una manera simple, mas o menos sin la ayuda de un experto del dominio o ingeniero de conocimiento.

La base de conocimientos en Protégé consiste de la ontología y un conjunto de instancias de las clases en la ontología. La base del modelo de conocimiento se trata a

7 <http://protege.stanford.edu/>

8 <http://gaia.fdi.ucm.es/projects/jcolibri/>

9 <http://mycbr-project.net/>

través de una jerarquía de clases. Una clase debe tener al menos una súper clase y una o muchas subclases, siendo la meta clase llamada *thing* y que es el top de la jerarquía, así muchas clases o subclases dependen de él. Las clases pueden tener instancias y a la vez datos individuales y clases pueden a su vez ser instancias.

3.4.2. jColibri : Entorno utilizado en esta tesis

Es un entorno de desarrollo orientado a la creación de sistemas de razonamiento basado en casos. es un producto del Grupo de Aplicaciones de Inteligencia Artificial, conocido por sus siglas GAIA, grupo integrado por profesores y estudiantes interesados en el campo de la Inteligencia Artificial de la Universidad Complutense de Madrid. A continuación su historia de este entorno como motivación al estudio.

COLIBRI es el acrónimo de Cases and Ontology Libraries Integration for Building Reasoning Infrastructures, es el producto de la tesis doctoral de Belén Díaz Agudo [GAIA, 2011] con la guía de Pedro A. González Calero, en el que se proponen una arquitectura independiente de dominio con el fin de diseñar sistemas CBR intensivos en conocimiento. Aquí algunas de las versiones que ha desarrollado GAIA.

- **COLIBRI** [GAIA. Díaz-Agudo et al., 2002a]

Se basa en la adquisición de conocimiento desde una librería de ontologías independientes y CBROnto [Díaz-Agudo et al., 2002b]. CBROnto es una ontología con terminología CBR que guía hacia la representación de los casos y brinda los métodos de solución a un problema al resolver tareas CBR. COLIBRI y CBROnto fueron desarrollados en lenguajes/entornos LISP and LOOM, los cuales son quizá no muy utilizados por los que no están familiarizados en descripción lógica (DL). Con la finalidad de ampliar la comunidad se creó jColibri. COLIBRI es el predecesor de jColibri 1.x y jColibri 2.x

Su principal objetivo es proporcionar la infraestructura necesaria para desarrollar nuevos sistemas de RBC y sus componentes de software asociados. COLIBRI está diseñado para ofrecer un entorno de colaboración donde los usuarios podían compartir sus esfuerzos en la implementación de aplicaciones

de RBC. Además, es una plataforma abierta donde los usuarios pueden contribuir con diferentes diseños o componentes que ser reutilizados por otros usuarios.

COLIBRI apoya el desarrollo de una amplia colección de sistemas CBR: sistemas estándar de RBC, textual CBR, conocimiento intensivo, intensiva data, sistemas recomendadores y aplicaciones distribuidas en CBR. También incluye herramientas de evaluación, mantenimiento y visualización de caso-base. Muchos de los componentes disponibles se han desarrollado por grupos de investigación de terceros y ha contribuido a que la plataforma sea compartida con la comunidad [GAIA, 2011].

La idea que subyace en el proceso de desarrollo propuesto en COLIBRI es la reutilización (algo inherente en Razonamiento Basado en Casos) de ambos sistemas diseños y sus componentes. El proceso de desarrollo define las actividades, roles, recursos y herramientas que reducen el tiempo de desarrollo en comparación con el esfuerzo requerido para implementar sistemas de RBC a partir de cero. Además, promueve la evaluación de los resultados reportados por otros investigadores simplificando la generación de sistemas de RBC existentes.

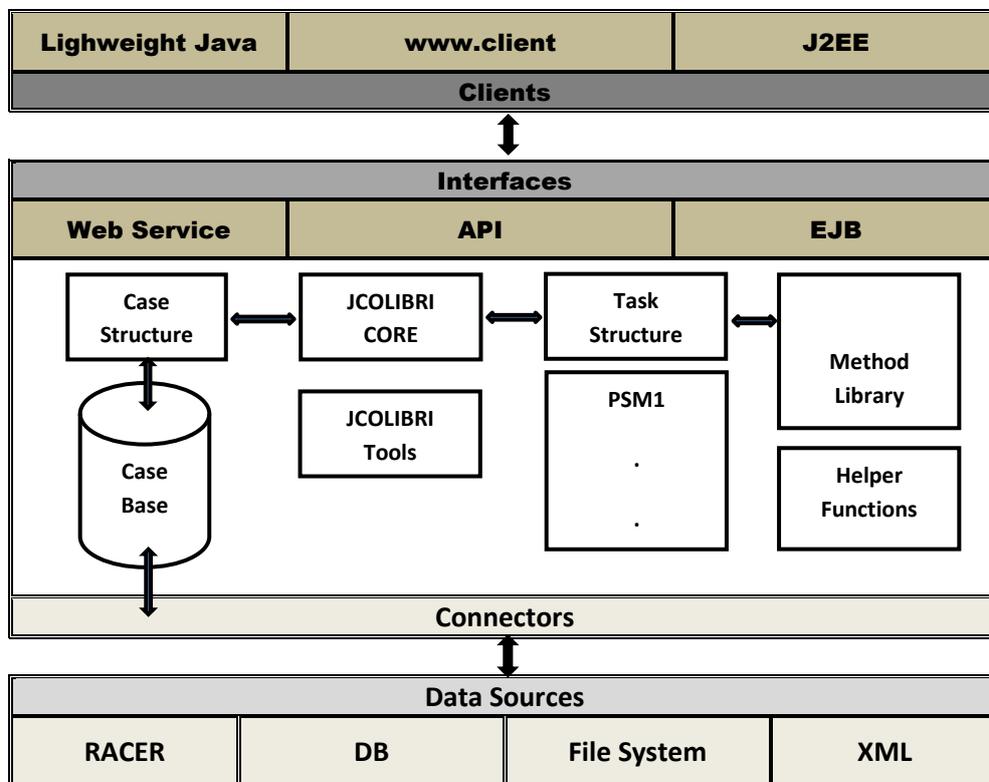
La plataforma ofrece un catálogo de sistemas CBR ya implementados representados como flujos de trabajo, y permite a los usuarios seleccionar el sistema más adecuado para adaptarlo a las necesidades concretas de la aplicación de destino. Estos flujos de trabajo son las llamadas plantillas y comprenden diseños de sistemas CBR que especifican el comportamiento pero no definen explícitamente los detalles funcionales de los medios de tarea. Este proceso de desarrollo incluye varias actividades relacionadas con la generación, publicación, intercambio, recuperación y reutilización de plantillas y componentes. Estas actividades se llevan a cabo por diferentes actores y con el apoyo de las herramientas integradas en *COLIBRI Studio*.

- **jColibri**

Su entorno se ilustra en la figura 3.1, como se mencionó anteriormente, es un completo desarrollo de la arquitectura COLIBRI. Fue originalmente creado

por Juan José Bello, mientras Antonio A. Sánchez-Ruiz Granados gratamente contribuyó al proyecto [GAIA, 2011].

Figura 3. 1. jColibri: Estructura del framework



Fuente: Adaptado de [Díaz-Agudo et al., 2002a]

El diseño del framework consiste de un número de archivos XML y clases de Java, organizado ALREDEDOR DE CUATRO ELEMENTOS BÁSICOS:

Tareas y Métodos los cuales son soportados por el entorno. Ellos son descritos por los archivos XML.

Casos Base los cuales son compatibles junto con conectores para apoyar relaciones entre bases de datos y sistemas de archivos.

Casos son proporcionados en el framework en forma de clases abstractas e interfaces para apoyar un caso real en un mundo real.

Métodos de resolución de problemas son representados por código actual en el framework

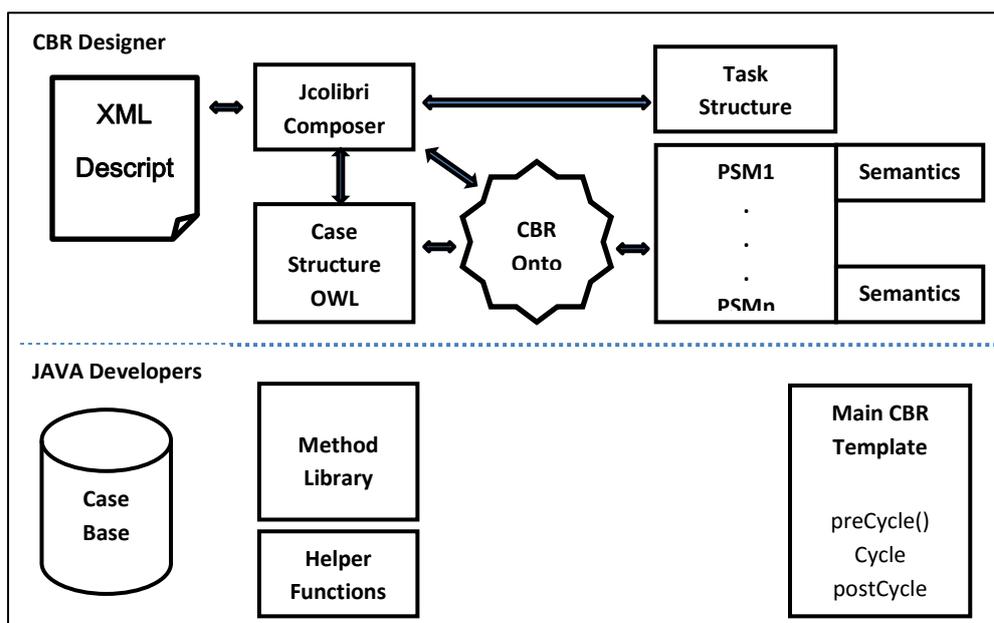
Uno de los principales problemas con el framework es cómo usarlo. jColibri soporta una configuración semiautomática, herramientas que ayuda al usuario a crear un sistema CBR a través una interfaz gráfica dinámica. Esta característica recién se implementó el 10 de junio de 2011, en jColibri 2.x.

- **jColibri 1.1**

jColibri 1.1 es un lanzamiento importante, donde la mayoría de las características de la estructura 1.0 jColibri han sido reimplantadas. Juan A. Recio-García es el desarrollador principal del proyecto jColibri 2 quien también completó la primera versión del framework. El desarrollo de la segunda versión del framework también fue la base de su tesis doctoral [Recio-García, 2008].

La principal razón para la reimplantación de jColibri, fue hacer el entorno mas amigable con la finalidad de extenderlo con código propio y hacerlo más robusto. El framework es también diseñado en lo que respecta a la separación de las técnicas de resolución de problemas y el conocimiento del dominio. Herramientas de autoría están presentes en jColibri 1.x no están incluidas, pero se encuentran bajo desarrollo.

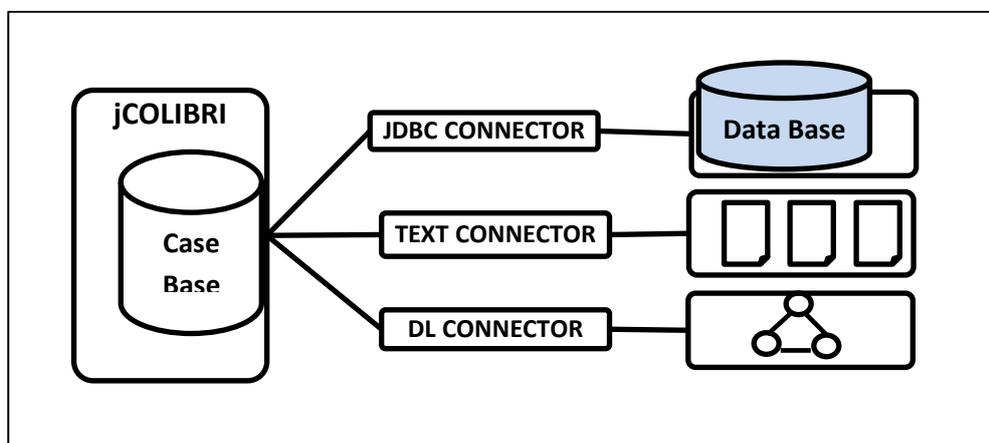
Figura 3. 2. jColibri1.1: Arquitectura del framework



Fuente: Adaptado de [Recio -García, 2008]

La arquitectura de jColibri tiene dos capas de persistencia, ilustradas en la Figura 3.3, compuesta de conectores en memoria con la organización de los casos. Los conectores son interfaces que saben como comunicarse y cargar los casos desde un medio físico. Hay actualmente tres tipos de conectores en jColibri; DatabaseConnector, PlainTextConnector y OntologyConnector. El conector más adecuado según los especialistas sería el DatabaseConnector, ya que es más fácil de mantener y depurar una base de datos si surgen problemas. El DatabaseConnector también es compatible con Hibernate internamente.

Figura 3. 3. jColibri1.1: Arquitectura persistente



Fuente: Adaptado de [Recio -García, 2008]

Cada conector se configura a través de archivos de configuración XML, los cuales afirman que clase en el proyecto es la descripción para el caso, y cual es la solución. Básicamente es un archivo que dice cómo se asignan los objetos de clase a las tablas en un determinado esquema de base de datos.

La segunda capa de la arquitectura se relaciona con la forma en que se organizan los casos una vez se cargan en la memoria. Hay tres organizaciones de base en memoria de los casos admitidos en jColibri; LinealCaseBase, CachedLinealCaseBase y IDIndexedLinealCaseBase. También hay una interfaz que permite a los desarrolladores crear sus propias representaciones si la quisieran.

Mecanismos para la recuperación, reutilización, revisar y retener los casos se incluyen con código bien documentado. Con el fin de aprender a usar el framework que utilizamos el tutorial jColibri [Recio-García et al., 2008] y poner en práctica el recomendador de viajes sistema. El sistema recomendador de viajes es un ejemplo de prueba creado por los autores de jColibri, y utiliza muchos de los componentes necesarios para construir un sistema CBR de trabajo.

Se incluyen dentro del sistema los mecanismos para la recuperación, reutilización, revisión y retención de los casos y el código está bien documentado, inclusive este software trae consigo un ejemplo llamado recomendador de viajes creado por los autores de jColibri. Este ejemplo utiliza muchos de los componentes necesarios para construir un sistema de trabajo CBR.

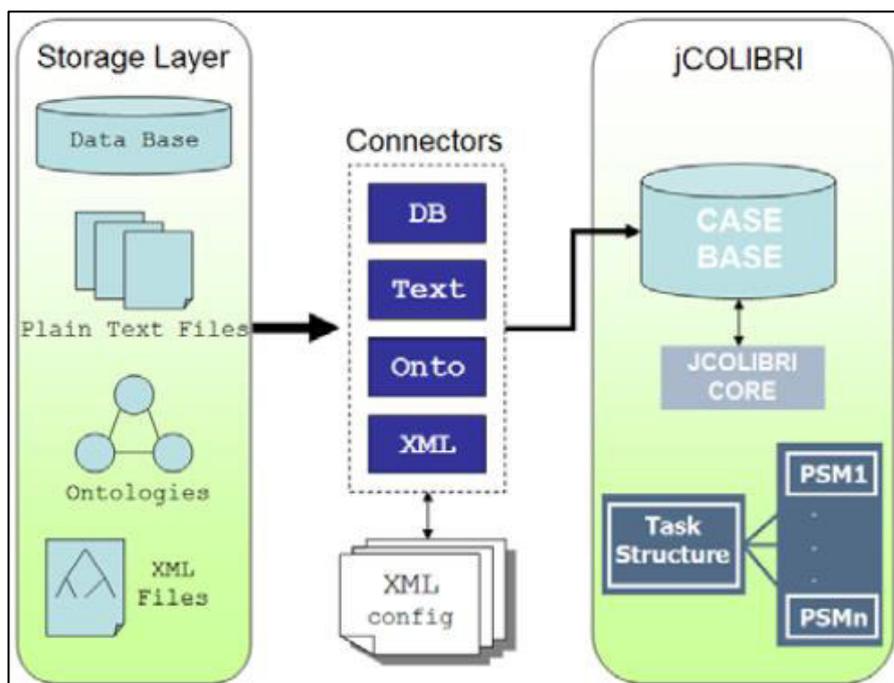
- **Colibri Studio**

Al utilizar colibrí Studio es necesario, entender algunos conceptos básicos respecto a la estructura de la aplicación a desarrollar, además de reconocer que Colibri Studio se basa en los componentes provistos por jColibri. Estos componentes están organizados como se muestra en la figura 3.5, basados en los siguientes principios de diseño:

- **Organización dentro de persistencia, núcleo y aspectos de presentación.**

Casos o experiencias pueden ser almacenados utilizando diferentes medios, siendo el medio mas común las bases de datos o el texto plano. Así jColibri, define una familia de componentes llamados *conectores (connectors)*, para invocar o llamar a los caso desde diferentes medios hacia la organización de la memoria: el caso base (*CBRCaseBase*). Hay muchas implementaciones del caso base en memoria, usando diferentes estructuras de datos: listas lineales, árboles, correlaciones, etc. La división en tres capas, permite una eficiente administración de los casos, un tema que se hace mas relevante cuando el tamaño de la base de casos crece.

Figura 3. 4. Esquema de la arquitectura de ColibriStudio



Fuente: Adaptado de [GAIA, 2011]

- **Organización de las aplicaciones en preciclo, ciclo y postciclo.**

Esta organización está definida por la interfaz StandardCBRAApplication mostrada en la figura 3.5. En primer lugar, el pre-ciclo carga los recursos necesarios y pre procesa el conocimiento utilizado por la aplicación. Este paso se ejecuta solamente una vez. Entonces el ciclo está listo para ser ejecutado cada vez que se recibe una consulta. Se activan los procesos CBR procesos. Finalmente, el post-cycle libera los recursos cargados para cerrar el sistema.

- **Estructura de caso.**

La estructura de casos está compuesta de cuatro componentes: *description*, *solution*, *result* y *justification solution*. Los componentes de caso están representados usando *javabeans*. Un Java Bean es una clase que tiene un método `get()` y un método `set()` para cada atributo público. El uso de Java Beans, permite a los desarrolladores diseñar sus casos como clases normales de Java, eligiendo el diseño más natural. De esta manera, cada atributo del Java Bean (enteros, cadenas, booleanos) se convierte en un simple atributo del caso. Por otra parte, se puede seguir un patrón *Composite* e incluir atributos compuestos que son otras clases de Java Beans que encapsulan algunos otros atributos. Estas clases se referencian como componentes de casos, ya que deben implementar la interfaz `CaseComponent` mostrada en la Figura 3.5. Esta interfaz sólo impone la definición de un atributo como el identificador del componente. Es utilizada por los conectores y métodos como una especie de clave primaria para identificar los componentes de casos.

- **Métodos que implementan y encapsulan la funcionalidad del sistema CBR.**

Los métodos implementan diferentes procesos de la aplicación (recuperación, selección, adaptación, etc.).

Si se quiere programar un sistema CBR utilizando directamente jCOLIBRI, se debe tener un conocimiento más profundo de los elementos que se muestran en la Figura 3.5; sin embargo, COLIBRI Studio proporciona un conjunto de herramientas que ocultan la complejidad del código Java y son capaces de generar el código fuente de su sistema de forma automática.

3.5. Confiabilidad del modelo RBC

La importancia de la confiabilidad en el área de la salud reside en que existen diversas maneras de valorar los fenómenos de la naturaleza y por lo tanto aparecen distintas aproximaciones o métodos diagnósticos usados para medir los mismos fenómenos o enfermedades. Por lo tanto, la confiabilidad adquiere importancia cuando se desea conocer si con un método o instrumento nuevo, diferente al habitual, se

obtienen resultados equivalentes de tal manera que eventualmente uno y otro puedan ser remplazados o intercambiados ya sea porque uno de ellos es más sencillo, menos costoso y por lo tanto más costo-efectivo, o porque uno de ellos resulta más seguro para el paciente, entre otras múltiples razones. En términos generales, la confiabilidad es el grado de concordancia en que dos o más observadores, métodos, técnicas, etc. Están de acuerdo sobre el mismo fenómeno observado.

Por esto, para establecer si el modelo de RBC propuesto es confiable, se utilizó el índice estadístico Kappa de Cohen (k) [Cohen, 1960]. El valor de k simplemente es la razón entre el exceso de concordancia observado mas allá del atribuible al azar y el máximo exceso posible.

En el evento en que el fenómeno observado se expresa o determina de manera binaria o dicotómica, por ejemplo, la presencia o ausencia de un signo clínico o imagenológico, la concordancia se realiza mediante la estimación del índice de Kappa (k), denotado por a siguiente ecuación

$$k = \frac{P_0 - P_e}{1 - P_e}$$

Los valores P_0 y P_e son hallados partiendo de una tabla de contingencia de 2x2 en la que se tiene clasificada la concordancia observada y la concordancia por azar de dos métodos, mostrados en la tabla siguiente.

Tabla 3. 1 Concordancia observa y esperada para el acuerdo entre el modelo RBC y el diagnóstico médico

MÉTODO 1	MÉTODO 2				Total Frec.
	Clasificación 1		Clasificación 2		
	Concord. observada	Concord. esperada	Concord. observada	Concord. esperada	
Clasificación 1	a	A	b	B	a + b
Clasificación 2	c	C	d	D	c + d
TOTAL	a + c	A + B	b + d	B + D	n

Proporción de concordancia observada global (P_0)

$$P_0 = \frac{(a + d)}{n}$$

Proporción de concordancia observada global (P_0)

$$P_e = \frac{(A + D)}{n}$$

Donde P_0 es la proporción de concordancia observada, P_e es la proporción de concordancia esperada por azar y $1 - P_e$, representa el acuerdo o concordancia máxima posible no debida al azar. Entonces, el numerador del coeficiente Kappa expresa la proporción del acuerdo observado menos el esperado, en tanto que el denominador es la diferencia entre un total acuerdo y la proporción esperada por azar. En conclusión, el índice k corrige el acuerdo sólo por azar, en tanto es la proporción del acuerdo observado que excede la proporción por azar. Si este valor es igual a 1, estaríamos frente a una situación en que la concordancia es perfecta (100% de acuerdo o total acuerdo) y por tanto, la proporción por azar es cero; cuando el valor es 0, hay total desacuerdo y entonces la proporción esperada por azar se hace igual a la proporción observada.

La evaluación del grado de acuerdo se da mediante el valor de k y clasificado según la escala mostrada en la tabla 3.2.

Tabla 3. 2. Valoración del índice Kappa

Valor de k	Fuerza de concordancia
< 0,20	Pobre
0,21 - 0,40	Débil
0,41 - 0,60	Moderada
0,61 - 0,80	Buena
0,81 - 1,00	Muy buena

Fuente: Adaptado de Landis JR, Koch GG. Biometrics 1977; 33: 159-174

CAPÍTULO IV

ESTADO DEL ARTE

Las denominaciones de “*Sistemas basados en conocimientos*” y “*sistemas expertos*” aparecen en la literatura como intercambiable, y han sido objeto de numerosas y detalladas investigaciones por investigadores en Inteligencia artificial, sin embargo no son exactamente lo mismo. La primera hace mas énfasis a aspectos mas estructurales, es decir se refiere a la forma como se construye el sistema, mientras que la segunda se refiere mas bien a las características funcionales, es decir que clases de tareas ha de realizar y con que aportaciones específicas. Aunque existen preferencias por usar la primera denominación, por otro lado los sistemas expertos pueden concebirse como un subconjunto de los SBC.

Los Sistemas Basados en Conocimiento (KBS: knowledge Based System) y los Sistemas Expertos representan una de las áreas más exitosas de la Inteligencia Artificial. Desde los primeros sistemas de los años 70 hasta hoy, la repercusión en la investigación y la industria de los KBS ha sido de gran importancia. El objetivo principal de este tipo de sistemas es ayudar a los seres humanos en la resolución de problemas y toma de decisiones.

La principal diferencia entre un KBS y un programa convencional radica en su estructura y organización. En los sistemas software convencionales, el conocimiento sobre el dominio se entremezcla con el flujo de control de la aplicación. Sin embargo, los Sistemas Basados en Conocimiento dividen claramente estos dos roles [Newel, 1982]. Por un lado contienen una base de conocimiento (o conocimiento del dominio), y por otro, el motor de inferencia necesario para manejar dicho conocimiento. Esta separación explícita entre conocimiento y control facilita la inclusión de nuevo conocimiento y la extensión de las capacidades del sistema. Aquí existe una analogía con el proceder de los humanos, ya que el proceso de razonamiento suele permanecer

invariable mientras que se añade continuamente nueva información en forma de experiencia.

A parte del problema principal en la adquisición del conocimiento existen otro tipo de inconvenientes [Watson y Marir, 1994]. La implementación de estos sistemas es un proceso muy complejo que debe llevarse a cabo por personas muy especializadas dentro grandes y costos proyectos. Además, una vez implementados su eficiencia suele ser baja debido a la dificultad en la gestión de grandes volúmenes de información. Por último, y no menos importante, el mantenimiento y extensión de este tipo de software suele ser muy laborioso.

4.1. Sistemas utilizados en los diagnósticos médicos

Mayoritariamente los sistemas de diagnósticos de enfermedades están contruidos sobre dos técnicas principales¹⁰: los Sistemas Basados en Conocimiento (KBS) y Modelos Computacionales Inteligentes (ICM). Hay tres variantes del KBS que son: Razonamiento Basado en Reglas (RBR), Razonamiento Basado en Casos (RBC) y Razonamiento Basado en Modelos (MBR), mientras que las técnicas ICM incluye Redes Neuronales Artificiales (ANN), Algoritmos Genéticos (GA) y Lógica Difusa (FL).

En RBR, las reglas son construidas desde la información coleccionada, para represrntar la base de conocimiento. Las reglas son particularmente patrones que representan el conocimiento [Liao, 2004]. Una máquina de inferencia desarrolla las inferencias encadenando las reglas recursivamente. RBR es popular en tanto las reglas puedan ser construidas, depuradas y mantenidas fácilmente [Lyn et. al., 2003]. Ejemplos de sistemas de diagnóstico basados en RBR son MYCIN para diagnosticar infecciones en la sangre y sistema nervioso [Pandey and Mishra, 2009] y ESSED para diagnosticar enfermedades a la vista como catarata, glaucoma, conjuntivitis, síndrome de ojo seco y keratitis en Malaysia [Ibrahim et al., 2001].

RBC es un tipo de KBS que recupera casos similares, basándose en casos previos. En el dominio médico, las soluciones, historial clínico, experiencias pasadas,

¹⁰ Siglas en idioma inglés. KBS: Knowledge-Based Systems; ICM: Intelligent Computing Models; RBR: Rule-Based Reasoning; CBR: Case-Based Reasoning; MBR: Model-Based Reasoning; ANN: Artificial Neural Network; GA: Genetic Algorithm; FL: Fuzzy Logic

pericia humana y conocimiento son almacenados en bases de datos que son usadas para resolver nuevos casos que surgen [Liao,2004]. Uno de los primeros sistemas RBC es CASEY, para diagnosticar enfermedades del corazón. El sistema busca casos similares en la base de conocimientos, luego observa la evidencia de la diferencia entre ambos casos y finalmente transfiere el diagnóstico al caso actual [Schmidt and Gierl 2001].

Otros notables sistemas RBC, son *Intelligent Patient Knowledge Management System*, una aplicación móvil para acceder a la información del paciente durante la consulta [Wilson et al., 2006] y un *intelligent system to diagnose liver diseases* en Taiwán [Lin, 2009]. En RBC, el conocimiento existente sobre los casos y tratamientos se distribuyen entre médicos y practicantes con la finalidad de conocer diagnósticos similares y la conservación de los casos ayuda a incrementar la precisión de los mismos; sin embargo, esto también puede ser una desventaja, porque debido a la redundancia e incremento de la base de casos puede colapsar el sistema [Schmidt and Gierl 2001].

Las últimas variantes del KBS es MBR, una técnica que usa modelos o representación del conocimiento para propósitos de observación, predicción y evaluación [Davis y Hamscher, 1998]. Un ejemplo de sistema MBR es YAQ, un sistema de diagnóstico para el síndrome de angustia respiratoria [Pandey y Mishra, 2009]. Este sistema fue desarrollado usando lenguaje descriptivo, el cual diagnostica fácilmente la condición y estado clínico [Uckun et al., 1993]. MBR es una buena técnica para el modelamiento de casos o diagnósticos inesperados, en tanto el modelo mismo y los desarrolladores necesiten desarrollar muchos modelos que puedan necesitar para referenciarlos con otros [Lee, 2000].

En tanto para los métodos ICM, las redes neuronales artificiales (ANN), es una técnica exitosa usada en el descubrimiento, clasificación, y agrupamiento de patrones, debido a que es considerada altamente precisa en la predicción de datos [Papik et al., 1998]. Un ejemplo de esta técnica en el dominio médico, es Hacettepe System, el cual consiste de dos modelos de ANN, desarrollados para reducir la morbilidad y mortalidad perinatal. Un modelo se usa para diagnosticar el desorden genético y el otro es usado para diagnosticar la salud fetal del embarazo en conjunto [Beksac et al., 1995]. ANN puede ser usada para representar la interacción de parámetros complejos,

así como múltiples variables [Mobley et al., 2000; Mobley et al., 2005]; sin embargo, la predicción puede ser difícil si los datos no están considerados meticulosamente y en algunos casos están ocultos. [Long, 2001; Pandey and Mishra, 2009]. Otro sistema que usa redes neuronales es el sistema de diagnóstico para enfermedades de la piel, el cual se usa para evaluar y reconocer la salud de la piel [Abadi et al., 2010].

FL usa variables lingüísticas para definir la base de conocimiento del sistema como una colección de reglas difusas del tipo IF THEN [Vitez et al., 1996]. DoctorMoon es un popular sistema basado en FL que es capaz de diagnosticar tuberculosis pulmonar, absceso pulmonar, cáncer pulmonar, neumonía y bronquitis. Los sistemas FL son simples y fáciles de diseñar, sin embargo, el sistema no es fácil de mantener, cuando un gran número de reglas o condiciones son cambiadas, tanto que afecta al rendimiento del sistema.

Tabla 4. 1. Ventajas y desventajas de los métodos KBS e ICM

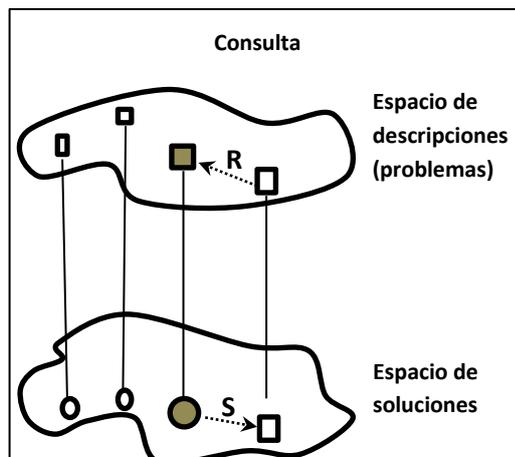
VENTAJAS KBS			VENTAJAS ICM		
RBR	RBC	MBR	NN	FL	GA
1. Modularidad	1. Fácil adquisición de conocimiento	1. Fácil interpretación y explicación	1. Múltiples variables	1. Diseño fácil y simple	1. Diseño fácil y simple
2. Uniformidad	2. Usa experiencia pasada	2. Casos inesperados	2. Compleja interacción de parámetros	2. Problemas complejos fáciles de manejar	2. Problemas complejos fáciles de manejar
3. Naturalidad	3. Aplicabilidad	3. Completo	3. Tratamiento fácil de los datos	3. Administración manejable	3. Administración manejable
4. Representación Compacta	4. Se actualiza		4. No necesita conocimiento matemático	4. No genera muchas reglas	
5. Provee explicación	5. Fácil de manejar los datos o valores perdidos				
DESVENTAJAS KBS			DESVENTAJAS ICM		
RBR	RBC	MBR	NN	FL	GA
1. Informal	1. Alto costo de búsqueda	1. Modelamiento difícil	1. Difícil de entender	1. Complicada interpretación de las características	4. No es sencillo el diseño
2. Cuello de botella en la adquisición de conocimiento	2. Problemas en índices de los casos	2. Carece de constructores del modelo	2. Teoría limitada	2. No es fácil el mantenimiento	5. Es costoso
3. Problemas en la inferencia de conocimiento	3. Dificultad en expresar el conocimiento general	3. Necesita la integración con otros modelos			6. Demanda mucho cálculo
	4. Problemas en eficiencia de la inferencia				
	5. Recuperación y selección son dependientes				

Finalmente GA, es un método de búsqueda basado en los principios de la selección natural y genética. Es típicamente un procedimiento de iteración que genera y muta nuevos puntos de muestra en el espacio de búsqueda usando dos operadores genéticos denominados crossover y mutation. GA fue satisfactoriamente usado en [Podgorelec et al., 1999] para diagnosticar disturbios en la válvula mitral y [Vinterbo y Ohno-Machado, 2000] para diagnosticar múltiples desordenes entre los pacientes. GA resuelve problemas de optimización y es capaz de trabajar con muchos diferentes problemas debido a su flexibilidad pero sus algoritmos de cálculo no son sencillos [Moorkamp, 2005].

Un resumen de las ventajas y desventajas de cada una de estas técnicas se describe en la tabla 3.1.

Uno de los principios más importantes del RBC [Kolodner, 1996] es: problemas similares tienen soluciones similares y los problemas tienden a recurrir. Por lo tanto, si un sistema es capaz de obtener los problemas más similares a la consulta planteada por el usuario podrá encontrar su solución ya que ésta deberá ser similar a la de los problemas recuperados al comparar las descripciones. De este modo aparecen dos de los componentes más importantes en un sistema RBC aparte de los casos: el conocimiento de similitud. Este conocimiento será distinto para el espacio de descripciones y el espacio de soluciones, pero en cualquier caso será mucho menor que el necesario para resolver el problema desde cero como se hace en los Sistemas Basados en Conocimiento clásicos. Esta suposición básica sobre el funcionamiento de los sistemas RBC se muestra gráficamente en la Figura 3.1. En dicha figura, la distancia R representa la similitud entre descripciones, mientras que la distancia A representa la similitud entre soluciones.

Figura 4. 1. Espacio de soluciones y espacio de descripciones (problemas)



Todos los casos estructurales son descritos como el par problema-solución. El problema $p_i = (a_i, v_i)$ se organiza como una estructura de atributos y valores, descrito por el vector de atributos $a_i = (a_{i1}, a_{i2}, \dots, a_{ir})$ y por el vector de valores $v_i = (v_{i1}, v_{i2}, \dots, v_{ir})$. La solución s_i se representa como vectores, definidos por tareas específicas. En las tareas de control, la decisión incluye dos vectores $s_i = (s_{pi}, s_{ri})$, donde el primer vector $s_{pi} = (sp_{i1}, sp_{i2}, \dots, sp_{iq})$ consiste de un conjunto de controladores de primer nivel jerárquico, y el segundo $p_{ri} = (pr_{i1}, pr_{i2}, \dots, pr_{im})$, valores de los parámetros observados correspondientes al conjunto. [S. Pal, S. Shiu, 2004]

A continuación se presenta una revisión-descripción acerca de los fundamentos, modelos, métodos y aplicativos que a la actualidad conciernen con nuestro estudio.

Los fundamentos del RBC se basan en la intersección de una serie de diversas disciplinas que realizaron distintas aportaciones desde el punto de vista filosófico, teórico o matemático.

El campo de las ciencias cognitivas aportó los conceptos básicos de experiencia, memoria y analogía. El trabajo de Schank y Abelson (1977) suele citarse como el origen del RBC y se basa en estos conceptos y teorías asociadas [Tulving, 1972; Smith et al., 1978]. Schank representó las experiencias como guiones prototípicos: ir a un restaurante, al médico, etc. Su trabajo continuó explorando el papel de la memoria de situaciones anteriores (casos) en la resolución de problemas y aprendizaje. A partir del

modelo de memoria dinámica desarrollado por Schank, Janet Kolodner desarrolló CYRUS; la primera aplicación RBC. Este modelo basado en casos serviría como base a los siguientes sistemas RBC: MEDIATOR [Simpson, 1985], CHEF [Hammond, 1989], PERSUADER [Sycara, 1987], CASEY [Koton, 1988].

Un conjunto de sistemas RBC muy importante es el RBC textual, el cual engloba los relacionados con textos. Este tipo de aplicaciones RBC presenta grandes posibilidades en dominios como el médico o legal donde almacenan su información mediante texto. Además, en la gran mayoría de escenarios donde es posible aplicar RBC para mejorar los procesos de gestión de la información ésta aparece representada mediante texto. Los textos pueden ser utilizados directamente como casos si se consigue extraer la información contenida en el lenguaje natural y estructurarla adecuadamente. El conjunto de métodos desarrollados para posibilitar la utilización de texto como origen de casos se engloban en el denominado RBC Textual (TRBC). La investigación en este campo se centra en resolver cuatro problemas fundamentales [Weber et al., 2006]: cómo calcular la similitud entre textos; cómo obtener representaciones estructuradas de los textos; cómo adaptar los casos textuales; y cómo generar automáticamente representaciones para TRBC.

4.2. Antecedentes del sistema RBC en medicina

El diagnóstico se ha considerado a menudo un problema de reconocimiento de patrones: encontrar a qué clase pertenece un patrón desconocido dentro de un conjunto de clases o hipótesis especificadas de antemano. La aplicación de estos métodos requiere un conjunto de características o medidas que pueden ser usadas para clasificar el patrón desconocido.

Específicamente para diagnóstico médico, las características son un conjunto de síntomas, o resultados de pruebas de laboratorio descubiertos por el médico, y las clases específicas son las categorías de enfermedades. El problema consiste en decidir a qué categoría de enfermedad pertenece el patrón desconocido con unas características de un paciente concreto. Ejemplos de categorías de enfermedades podrían ser los diferentes tipos de enfermedades del corazón, mientras las características pueden incluir pruebas como la presión de la sangre o síntomas como dolor de cabeza [Weis, S., 1974].

En la década de los 70, Weiss y Kulikowski [1974-1978] desarrollaron en la Universidad de Rutgers un sistema experto para ayuda en el diagnóstico y terapia del glaucoma y otras enfermedades oculares llamado CASNET (Causal ASociational NETwork). Los procedimientos de razonamiento interpretan los hallazgos de un paciente particular en términos de un modelo de una red causal asociativa (CASNET) o red semántica que caracteriza los mecanismos fisiopatológicos y el curso clínico de enfermedades tratadas y no tratadas. Las aportaciones de este programa son: la generación de interpretaciones complejas a partir de un modelo cualitativo de un proceso de enfermedad, el razonamiento sobre el manejo del seguimiento detallado de un paciente, la incorporación de opiniones alternativas de expertos sobre los temas bajo debate, y su prueba y actualización mediante una red colaborativa informática de investigadores en glaucoma.

Bichindaritz y Sullivan [Bichindaritz I. y Sullivan K., 2002] desarrollaron un sistema llamado de atención Care-Partner. Este sistema de tutores de los estudiantes de medicina a través de Internet, brindándoles practicar los casos para ayudar a probar sus habilidades. Care-Partner también le da al estudiante el resultado de ya sea falla, no cumplimiento /cumplimiento de todas las normas.

Los especialistas en medicina, argumentan que RBC interactivo es una buena solución para una gran cantidad de problemas de diagnóstico secuencial. También la implementación de una interfaz llamada estrategia de RBC con la demostración de su uso en el dominio de diagnósticos fallidos a través del computador [McSherry D., 2001].

Nilson y Sollenborn, indican que existe una buena visión general sobre el campo de la RBC en el ámbito médico. Se dividen en 4 grupos de sistemas mirando su aplicación médica, sistemas de diagnóstico, sistemas de clasificación, sistemas de tutoría y los sistemas de planificación. También discuten las tendencias de construcción de RBC específicas para los sistemas de RBC en el campo médico [Nilson M., Sollenborn, M., 2004].

Otro exitoso campo de aplicación del RBC es el diagnóstico [Göker et al., 2006]. Este tipo de sistemas deben identificar el origen de un funcionamiento anormal y proponer una solución. Las aplicaciones de este tipo de RBC varían desde el soporte a

usuarios y solución de problemas técnicos: SMART [Acorn y Walden, 1992], HOMER [Göker et al., 1998] o CASSIOPÉE [Heider, 1996]; hasta aplicaciones de diagnóstico médico: CASEY [Koton, 1988] o BOLERO [López y Plaza, 1997].

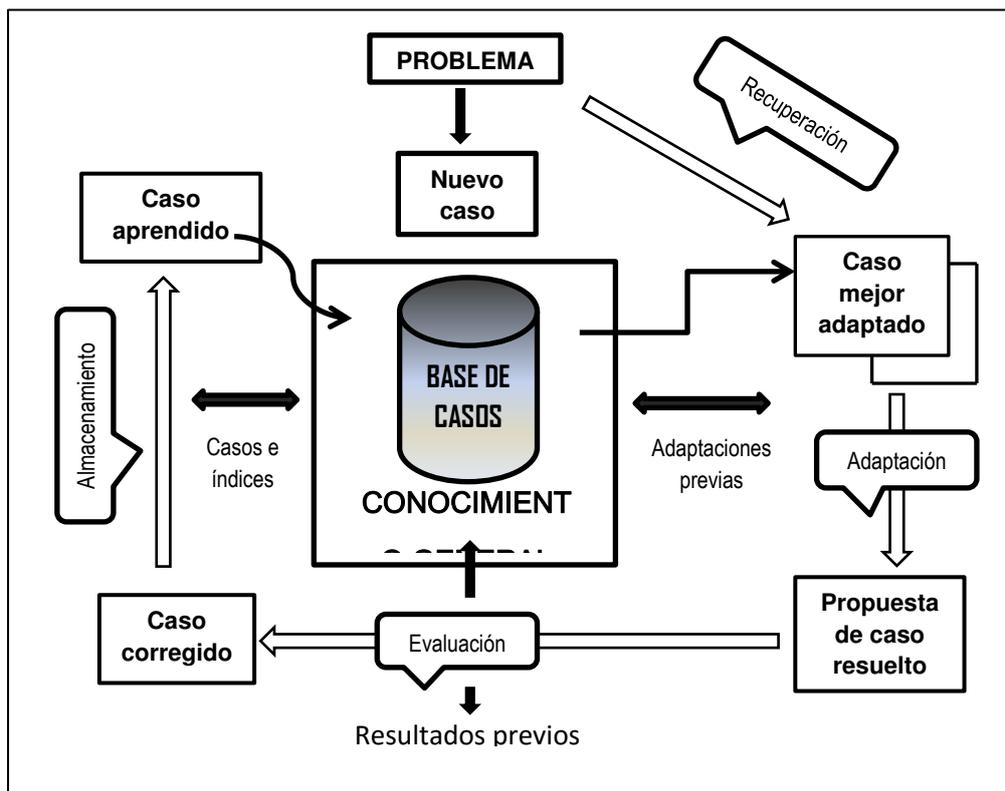
4.3. El ciclo RBC del Software jCOLIBRI utilizado

Los sistemas RBC se basan en la premisa de que problemas parecidos pueden resolverse a través de soluciones similares y que los problemas tienden a repetirse. Esto les permite nutrirse de la experiencia pasada para poder dar solución a los problemas que se les presentan. Cuando un sistema RBC se enfrenta a la resolución de un nuevo problema, buscará en su base de casos los problemas similares al problema actual. Una vez encontrados, adaptará las soluciones de los casos previos al problema actual para dar una respuesta válida. El razonamiento RBC surgió a raíz de los trabajos de Roger Schank, en memoria dinámica y el papel central que los episodios previos (casos) tenían en la resolución de problemas y el aprendizaje [Schank R., 1983].

El proceso de solución de un problema en un sistema RBC consta de cuatro fases, en lo que conocemos como ciclo RBC (Figura 4.2). Las fases de este ciclo son:

- **Recuperación:** Durante esta fase se busca en la base de casos del sistema el caso más similar al problema actual.
- **Adaptación:** Se adapta el caso recuperado al problema actual.
- **Revisión:** Se comprueba que la solución del caso adaptado resuelva el problema inicial.
- **Aprendizaje:** La solución adaptada junto con la descripción del problema se incluyen a la base de casos como un nuevo caso.

Figura 4. 2. Proceso de solución de un problema en el sistema CBR



Fuente: Jcolibri: Una plataforma para sistemas CBR. Universidad Complutense. Madrid

El aprendizaje de los sistemas RBC se produce durante la fase de aprendizaje. Al introducir los nuevos casos en la base de casos se consigue que el sistema aprenda de la experiencia pasada. De esta forma, ante la llegada de un problema, el sistema RBC sabrá resolverlo si en su base de casos encuentra casos similares al problema al que se está enfrentando. En caso contrario, no podrá resolver el problema o la solución propuesta será errónea, no siendo validada durante la fase de revisión y por lo tanto no entrando a formar parte de su base de casos.

Dado que los sistemas RBC necesitan disponer de una base de casos para trabajar, no son apropiados para resolver aquellos problemas de los que no se tiene casos previos, es decir, problemas a los que va a tener que enfrentarse el sistema RBC cubriendo todo el rango posible junto con sus soluciones. Son apropiados por tanto, para resolver problemas en los que si existe este conjunto de problemas junto con sus soluciones, como por ejemplo, problemas de diagnóstico o de clasificación.

Para determinar qué casos de los contenidos en su base de casos se parecen más al problema que debe resolver, el sistema RBC utiliza las medidas de similitud. Una medida de similitud es una función que dados dos casos es capaz de determinar cuanto de parecidos son. Las medidas de similitud resultan fundamentales para el correcto funcionamiento de los sistemas RBC, puesto que van a determinar los casos que son devueltos durante la fase de recuperación.

En muchas ocasiones, cuando el tamaño de la base de casos crece mucho es necesario aplicar técnicas que permitan aumentar el rendimiento del sistema. Estas técnicas son conocidas como mecanismos de indexación. Gracias a ellas, las bases de casos son indexadas y ordenadas de tal forma que las nuevas búsquedas se realizan de una forma más rápida, mejorando el rendimiento del sistema. Los atributos de indexación en torno a los cuales se realice el proceso de indexación dependerán del algoritmo utilizado, variando desde métodos de selección manual de atributos hasta métodos automáticos.

4.4. El contexto para la decisión clínica

El problema de la decisión se puede definir como la elección entre muchas opciones para obtener el objetivo lo más eficientemente posible. Muchas de las veces la habilidad del médico va más allá de la capacidad de la computadora por lo tanto la herramienta de ayuda no tiene ningún valor. El mejor ejemplo es el diagnóstico; sin embargo, la actividad médica se podría definir como una cadena de decisiones (más exactamente una red de trabajo), que involucre los siguientes pasos: el diagnóstico, el pronóstico, la terapia y el tratamiento a seguir. Algunos de estos pasos se hacen más y más complejos y una herramienta de ayuda sería muy útil, especialmente durante el pronóstico y etapa de la terapia.

Los especialistas en inteligencia artificial, estuvieron muy interesados en la naturaleza cognitiva de la actividad del diagnóstico médico; por esto, muchas aplicaciones fueron construidas en este dominio. Según las informaciones, muchas de ellas no se usan, debido a que no brindan alguna mejora a la decisión del especialista. Paradójicamente, muchos profesionales médicos están interesados en los sistemas de información y algunos de ellos han desarrollado pequeños o algunas veces mas sofisticados sistemas de soporte de decisión; sin embargo, estos sistemas son usados

muy particularmente en un hospital o dentro de su departamento de investigación por pequeños grupos de especialistas quienes tienen que resolver problemas más precisos o de especialización.

En Francia, por ejemplo, el usuario final del sistema y el diseñador es a menudo la misma persona, quien prefiere desde su hogar dar soluciones. En algunos departamentos la actividad médica es tan especializada que no hay lugar para esta clase de DSS.

Dicho lo anterior, en lo que sigue de este capítulo, se va a describir la estructura del modelo, las tareas del sistema en el modelo propuesto y la arquitectura del sistema y finalmente la implementación del mismo, utilizando como se dijo anteriormente el software Protegé y jCOLIBRI.

4.5. ¿Dónde y por qué usar un RBC?

Aunque RBC es muy útil para varios tipos de problemas y dominios, hay situaciones donde no es apropiada la utilización de esta metodología. Existen un número de características en los problemas candidatos y sus dominios en los que se determina la aplicabilidad del RBC [A. Aamodt, E. Plaza, 1994] y que se enumeran a continuación:

- El dominio no tiene un modelo subyacente.
- Hay excepciones y casos nuevos.
- Los casos se repiten frecuentemente.
- Hay un significativo beneficio en la adaptación de soluciones pasadas.
- Se obtienen relevantes casos previos.

En general hay méritos para el uso del RBC:

- *Reduce la tarea de adquisición de conocimiento.* Al eliminar la necesidad de un modelo o un conjunto de reglas, como si lo hace un sistema basado en modelo/reglas, la tarea de adquisición de conocimiento del RBC, consiste

principalmente de una colección de casos/experiencia y su representación y almacenamiento.

- *Evita repetir los errores sucedidos en el pasado.* Este sistema puede registrar fallas así como éxitos, y la información que causó estas fallas/éxitos en el pasado, puede servir para predecir potenciales fallas en el futuro.
- *Proporciona flexibilidad en el modelamiento del conocimiento.* Contrario a los sistemas basados en modelos, quienes no se adaptan cuando hay datos faltantes o el modelo se desvía un poco del problema, los sistemas basados en casos usan la experiencia pasada como dominio del conocimiento y a menudo proveen soluciones razonables debido a su adaptación apropiada a este tipo de problemas.
- *Razón en dominios que no han sido completamente interpretados, definidos o modelados.* En situaciones donde el conocimiento existente es insuficiente para construir un modelo causal de un dominio o no se pueda derivar un conjunto de heurísticas para ello, el razonamiento basado en casos puede todavía ser desarrollado usando un pequeño conjunto de casos del dominio. La teoría subyacente del conocimiento del dominio no tiene que ser cuantificada o entendida completamente para que un razonador basado en casos funcione.
- *Sugiere soluciones.* Cuando la información es almacenada con respecto al nivel de éxitos de soluciones pasadas, el razonador basado en casos puede predecir los éxitos de la solución sugerida para el problema actual en consulta. Esto se hace al consultar la solución almacenada, el nivel de éxito de tal solución y la diferencia entre el previo y actual contexto de aplicación de estas soluciones.
- *Aprende en el tiempo.* Tanto como se use el sistema RBC, encontrará mas situaciones problemáticas y creará mas soluciones. Si las soluciones son subsecuentemente probadas en el mundo real, y se determina el nivel de éxitos para esas soluciones, entonces estos casos pueden ser adicionados a la base de casos y usados para ayudar a resolver futuros problemas. Como os casos son agregados, un sistema RBC deberá poder razonar en una amplia variedad de situaciones y con un alto grado de refinamiento y éxito.

- *La razón en un dominio con poco conocimiento.* El RBC puede empezar con pocos casos e ir incrementando su conocimiento al agregar nuevos casos. La adición de nuevos casos causarán que el sistema se expanda en direcciones que son determinadas por los casos encontrados en su esfuerzo por encontrar solución al problema.
- *La razón con data o conceptos incompletos.* Los casos que son recuperados pueden no ser idénticos al caso consulta. Sin embargo, cuando ellos están dentro de la medida de similaridad al caso en consulta, cualquier incompitud o imprecisión puede ser tratada por el razonador basado en casos. Mientras estos factores pueden causar una leve degradación en la performance, debido a la disparidad entre el caso actual y el recuperado, el razonamiento puede todavía seguir trabajando.
- *Evita repetir los pasos para llegar a una solución.* En los dominios en los que el problema requiere significativos procesos para llegar a la solución, la alternativa al modificar una solución previa, puede reducir significativamente el requerimiento del proceso. Además reusar una solución previa permite que el razonador la tome consigo y pueda ser reusada para resolver otros problemas.
- *Provee un medio de explicación.* Los sistemas RBC pueden suplir casos previos y sus soluciones exitosas a convencer al usuario o justificar las razones con respecto a porque la solución propuesta para el actual problema debe ser considerada. En muchos dominios, hay ocasiones en que un usuario desea asegurarse acerca de la calidad de la solución propuesta por el sistema. Al explicar como un caso previo fue exitoso en una situación, se usa la similaridad entre los casos y el razonamiento incurrido en esa adaptación, un sistema RBC puede explicarle al usuario la solución.
- *Puede ser usado en diferentes formas.* El número de formas o maneras que un sistema RBC puede ser implementado es casi ilimitado, se puede usar ara múltiples propósitos: creación de un plan, elaborar un diagnóstico, argüir puntos de vista, etc. Por lo tanto el trato con un sistema RBC, es asimismo de muchas formas, y la adaptación de métodos también es variada.

- *Puede ser aplicado a un amplio rango de dominios.* Esto se debe a su aparentemente ilimitado número de maneras de representar, indexar, recuperar y adaptar los casos.
- *Refleja el razonamiento humano.* Como se dijo anteriormente, nosotros los humanos, usamos una forma de razonamiento basado en casos, no es difícil convencer a los implementadores, usuarios y administradores de la validez de este paradigma. Como humanos podemos entender el sistema RBC, su razonamiento y explicaciones y a su vez estar convencidos de la validez de sus soluciones recibidas del sistema.

4.6. Sistemas RBC más Conocidos

Han sido muchos los sistemas de razonamiento basado en casos implementados en los últimos años, a continuación se da una lista de algunos de ellos, construidos tanto para el estudio como para el trabajo, libres como comerciales, hechos para el mundo real.

- **Cyrus**

Su autor fue Kolodner, CYRUS [Kolodner, 1993] fue el primer sistema RBC implementado y fue basado en la teoría de la memoria dinámica de Schank. Era un sistema que contó con viajes y reuniones de un secretario de Estado norteamericano.

- **Protos**

Sus autores fueron Bruce W. Porter y Ellis R. Bareiss. Es un sistema de resolución y aprendizaje para las tareas de clasificación heurística [Bareiss, 1989, Porter et al., 1990].

- **Creek**

Creek [Aamodt, 2004] es un enfoque RBC intensivo en conocimiento, con estrecha relación entre el aprendizaje y los casos de dominio general.

- **DrillEdge**

Es un sistema RBC construido por un grupo de profesores y estudiantes de la Universidad Noruega de Ciencia y Tecnología, basado en CREEK, utilizado en el ámbito de la extracción de petróleo [Sφrmo, 2009].

- **Casey**

Diagnostica los pacientes aplicando heurísticas de adaptación y emparejamiento basado en el modelo independientes del dominio y son tan precisas como el modelo del dominio en el que se aplican. La diagnosis la realiza en dos pasos: primero busca en la memoria casos y utiliza reglas de evidencia basadas en el modelo para determinar cuales de los casos que ajustan parcialmente son suficientemente similares al nuevo problema para proporcionar una diagnosis precisa. Después aplica reglas de reparación basadas en el modelo (estrategias de adaptación) para adaptar el diagnóstico antiguo a la nueva situación [Koton, P. 1989].

CAPÍTULO V

RESULTADOS Y DISCUSIÓN

En este capítulo se presenta la estructura del sistema desarrollado mediante el modelo propuesto, así como su aplicación para una determinada enfermedad como es la malaria, posteriormente se da el procedimiento a realizar utilizando los softwares Protégé y jCOLIBRI, el primero para construir una ontología breve y el segundo para elaborar la estructura del modelo de razonamiento..

Existen diferentes modelos de sistemas de soporte para la toma de decisión (DSS: Decision Support System) usados en medicina como ayuda a los médicos en el diagnóstico de enfermedades, pronóstico de evaluación y prescripción de la terapia o tratamiento, siendo modelos DSS que dependen de teorías matemáticas o computacionales. El modelo propuesto en esta tesis trata de integrar las diferentes clases funcionales integradas para la toma de decisión y usarlas para encontrar un diagnóstico.

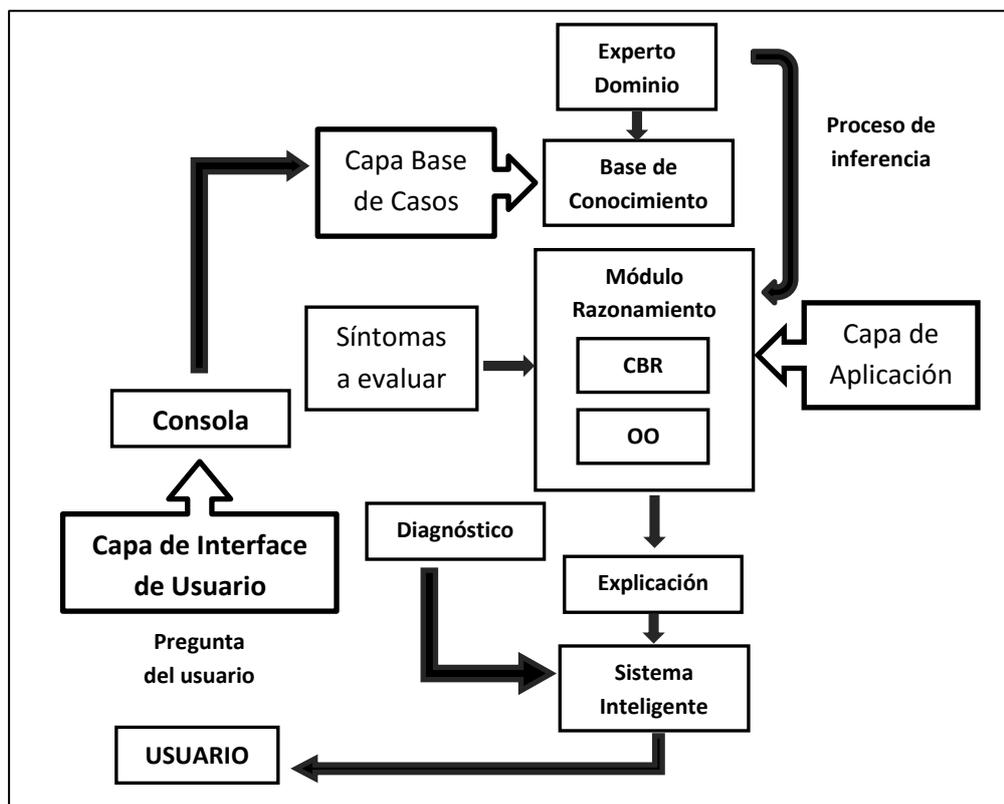
En el caso de nuestro modelo, éste brinda a través del entorno de jCOLIBRI, un apoyo al especialista en salud, al brindar el diagnóstico mas parecido al caso en consulta, para que luego este especialista tome la decisión en cuanto al tratamiento a ejercer.

5.1. Estructura del Modelo

En esta sección se muestra el modelo general para el sistema propuesto, siendo los componentes centrales la base de conocimientos (capa de la base de datos) y el módulo de razonamiento (capa de aplicación), es decir la máquina de inferencia. La base de conocimiento está complementado con el conocimiento explícito del dominio

del experto y estructurado dentro de la base de casos. La máquina de inferencia accederá a la base de conocimientos para luego pasar a evaluar todos los casos de los pacientes de acuerdo al síntoma o síntomas, generándose luego el proceso de razonamiento para dar una conclusión (explicación) final. Los signos o síntomas son ingresados por el usuario vía consola (capa de interface del usuario). La figura 4.1 muestra el sistema en sus dos principales funciones; esto es, el diagnóstico y la propuesta de una explicación como por ejemplo, por qué se hizo cierto diagnóstico; conocido esto el usuario del sistema ya puede dar el tratamiento sugerido o prescripción de los medicamentos. Los usuarios pueden ser médicos o personal profesional en salud.

Figura 5. 1. Modelo para el sistema propuesto



Fuente: Journal of Social Sciences 8 (1): 66-73, 2012. ISSN 1549-3652 © 2012 Science Publications

Como se dijo anteriormente la base de conocimiento contiene la información utilizada para la toma de decisiones. Esta información presente es adquirida por los expertos en una determinada área de la medicina.

5.2. Tareas en el Sistema Propuesto

Acorde a la metodología RBC, nuestro sistema se resume en los siguientes pasos:

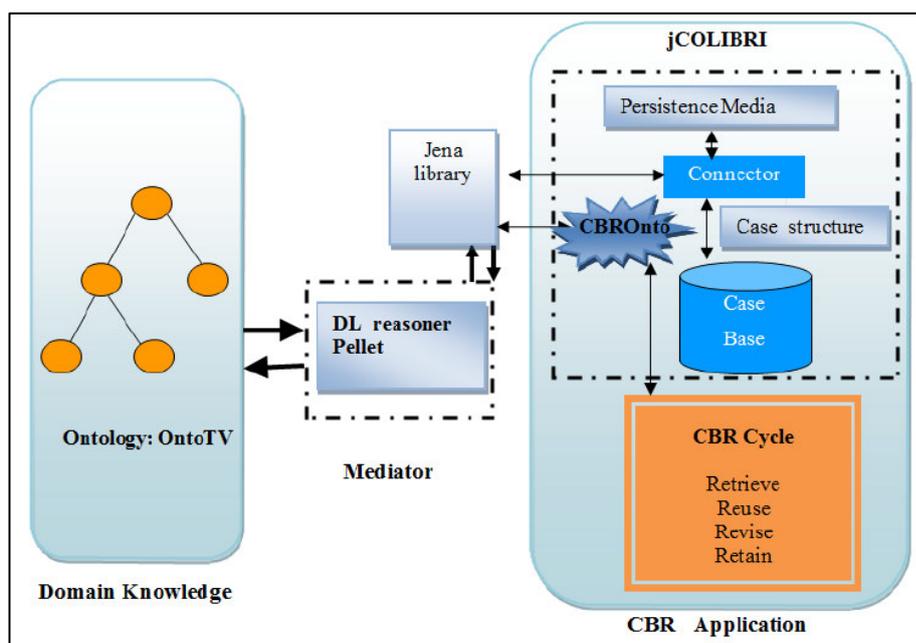
1. El sistema buscará en la memoria de casos un caso existente que se ajuste al problema ingresado.
2. El sistema encontrará un caso que coincide exactamente con el problema de entrada y va directamente a una solución (el sistema se hará más potente mientras se añade nuevos casos).
3. Si no encontrara un caso exactamente igual al planteado, se recupera un caso que es similar a nuestra situación de entrada, pero no del todo adecuada para proporcionar una solución completa.
4. El sistema debe encontrar y modificar pequeñas porciones del caso de que no cumplan con la especificación de entrada recuperada. Este proceso se llama "caso-adaptación".
5. El resultado de proceso de adaptación a un caso es (a) completar la solución, y (b) generar un nuevo caso que se agrega automáticamente a la memoria del sistema para uso futuro.

5.3. Arquitectura del Sistema en el Modelo

La arquitectura del sistema propuesto tiene el enfoque del uso de ontologías para construir modelos de conocimiento del dominio general. Aunque en un sistema CBR la principal fuente de conocimiento es el conjunto de experiencias anteriores, nuestro enfoque de la CBR es hacia aplicaciones integradas que combinan el conocimiento específico con los modelos de conocimiento del dominio general. Cuanto más conocimiento esté imbuido el sistema, más eficaz se espera que sea. Los procesos CBR semánticos pueden aprovechar este conocimiento del dominio y obtener resultados más precisos. En este tipo de sistemas, las ontologías juegan un papel importante [García, J. et al. 2006], tales como un vocabulario para describir los casos, como estructura de conocimiento donde son localizados estos, y como fuente de conocimiento permitiendo el razonamiento semántico en los métodos de cálculo de similitud, adaptación y aprendizaje.

Sobre la base de lo dicho anteriormente, la arquitectura propuesta está compuesta de tres componentes funcionales: ontología del dominio, aplicación CBR y un mediador descriptivo lógico, como se muestra en la figura 5.2.

Figura 5. 2. Arquitectura del sistema



Fuente: International Journal of Hybrid Information Technology Vol. 5, No. 3, July, 2012

5.3.1. Conocimiento del Dominio

La ontología del dominio se implementa para construir el modelo del conocimiento general, el cual incluye vocabularios, conceptos y relaciones para representar todo el conocimiento concerniente al diagnóstico y su posterior tratamiento.

5.3.2. Aplicación CBR

Este componente se utiliza para resolver un problema de diagnóstico y está orientado a la estructura de los casos y a las cuatro tareas básicas del proceso de CBR (Recuperar el o los casos más similares, Reutilizar su conocimiento para resolver el problema, Revisar la solución propuesta y Retener la experiencia). Las tareas antes mencionadas se asignan respectivamente a los siguientes procesos:

- a. La representación del caso
- b. La localización del caso y recuperación
- c. La adaptación del caso, y,
- d. Mantenimiento de la base de casos

Para aprovechar el conocimiento del dominio, el CBR necesita de algunos procesos que deben expresarse de una manera similar, por lo tanto, proponemos el uso de una ontología para CBR denominada CBRonto [Agudo, B.D., et al. 2000] que proporciona el vocabulario para describir los elementos que intervienen en los procesos de RBC.

5.3.3. Representación de los Casos

Dentro de la base de casos, el caso se describiría como algo principal dentro del vocabulario propuesto por el dominio del modelo. La característica de la representación de un caso, tiene que ver con el tipo y estructura del dominio del conocimiento dentro de los casos.

Para representar la descripción de un caso, específicamente la ontología de una CBR tiene tres clases:

- CBR_CASE, resume los conceptos sobre la representación de varios tipos de casos, los cuales pueden existir en el sistema.
- CBR_DESCRIPTION, resume los conceptos representativos de las partes de un caso (Descripción, solución).
- CBR_INDEX, contiene los índices para la estructura y contenido del caso base.

5.3.4. Recuperación del Caso

Cuando se trabaja con ontologías, el concepto de jerarquía influye en la valoración de la similitud. Intuitivamente, es obvio que la jerarquía de clases contiene los objetos similares y por lo tanto sus conocimientos son también similares. Hay dos enfoques para el cálculo de la similitud [B. D-Agudo et.al, 2001]:

- *Recuperación basada en la clasificación*, construye un concepto o una descripción individual, usando las restricciones especificadas en la consulta. Este concepto/descripción, es entonces clasificado al mismo tiempo que se recupera las instancias que tienen características comunes.
- *Recuperación basada en el cálculo*, usa funciones de similitud numéricas para evaluar y ordenar los casos con respecto a la consulta. El uso de las representaciones estructuradas de los casos requiere de varios enfoques para evaluar las similitudes que permiten comparar dos objetos estructurados de manera diferente; en particular, los objetos pertenecientes a diferentes clases.

Las medidas de similitud para la representación de la estructura de casos, tiene por finalidad determinar la similitud entre dos objetos; es decir, un objeto representando el caso (o una parte de él), a esto se le denomina similitud global. La similitud de los objetos está determinada recursivamente de abajo hacia arriba, es decir, para cada atributo simple, una medida de similitud local, determina la similitud entre los valores de dos atributos, y para cada slot relacional de los objetos, una medida de similitud recursiva compara dos sub-objetos relacionados. Entonces los valores de similitud de las medidas de similitud local y las medidas de similitud de objetos respectivamente, son agregados para los objetos semejantes, entre los objetos que están siendo comparados.

5.3.5. Adaptación de Casos

La adaptación de casos juega un rol importante en la habilidad de los sistemas CBR para resolver nuevos problemas. Para nuestro caso, se trabaja con la representación explícita de los términos dentro del dominio. De esa manera cierto conocimiento de adaptación es explícitamente representado en la taxonomía del dominio, eso indica por ejemplo, que los individuos que están más juntos dentro de la taxonomía, son eventualmente intercambiables.

Se propone en esta tesis usar una ontología basada en el modelo y adaptación del esquema por eliminación y sustitución. Las dependencias dentro de un caso están explícitamente representadas para guiar la adaptación. Si un elemento es eliminado o removido, los elementos dependientes a éste también son eliminados. Si un elemento

e_1 es sustituido por otro elemento e_2 , entonces los elementos dependientes son sustituidos. La búsqueda de sustitutos es guiada por la ontología. La adaptación se requiere cuando el caso recuperado no cumple totalmente con todos los requisitos dados en la consulta, o cuando resuelve sólo parte de ésta, aquí es necesario adaptar los operadores de borrado o sustitución. Se propone, por tanto, una adaptación de los mecanismos como un proceso para propagar los cambios de la descripción para solución de los ítems, como sigue:

- Se obtiene una lista L de ítems en la solución que necesita ser adaptada. Se encuentra estos ítems siguiendo una relación de la ruta + el concepto.
- Cada ítem en L es sustituido por un nuevo ítem apropiado. La búsqueda del sustituto adecuado se realiza mediante una clase especial de búsqueda que toma ventaja del conocimiento de descripción lógico (DL), basado en la organización y aplicación de funciones de similitud para encontrar un sustituto que sea similar al requerido.
- Substituir los ítems que dependen de los otros ítems en la solución que ya ha sido adaptada.

Las búsquedas especializadas como se describen [J. Kolodner, 1993], son un camino para encontrar candidatos a substituir en un caso solución, donde las instrucciones son dadas para encontrar el necesario ítem.

5.3.6. El Mediador o Mediator

El descriptor lógico (*DL: Description Logic*) denominado el razonador Pellet [E. Sirin, et al., 2007], se usa como mediador mostrado en la figura anterior, y es el responsable por un lado, de custodiar la consistencia del conocimiento base y por otro, de inferir nuevo conocimiento, que tal vez no se puede explicar acertadamente pero que puede ser deducido.

5.4. Implementación

Se utilizó el software editor *Protégé* [Protégé, 2009], para la generación manual y modelamiento de la ontología dominio y jCOLIBRI [GAIA, 2011] para la aplicación de la construcción del sistema CBR.

5.4.1. Ontología Dominio

En esta *tesis*, se ha trabajado con dos enfermedades típicas del entorno rural de la Comunidad Campesina Cruz de Mayo, como son la malaria y la leishmaniasis. Es importante considerar el concepto de territorio social, en la medida en que en su contenido se articula con los determinantes sociales de la salud, dado que estas enfermedades se presentan en territorios sociales específicos y concretos con dinámicas históricas y sociales únicas, en un paisaje definido en continua transformación: el territorio rural de las zonas tropicales y la constante construcción o deconstrucción de su ruralidad [Santos, M., 2000]. A continuación, en la tabla 5.1. se presentan los síntomas y sus niveles de afección dentro de cada síntoma.

Tabla 5. 1. Parámetros de evaluación en el diagnóstico de las enfermedades

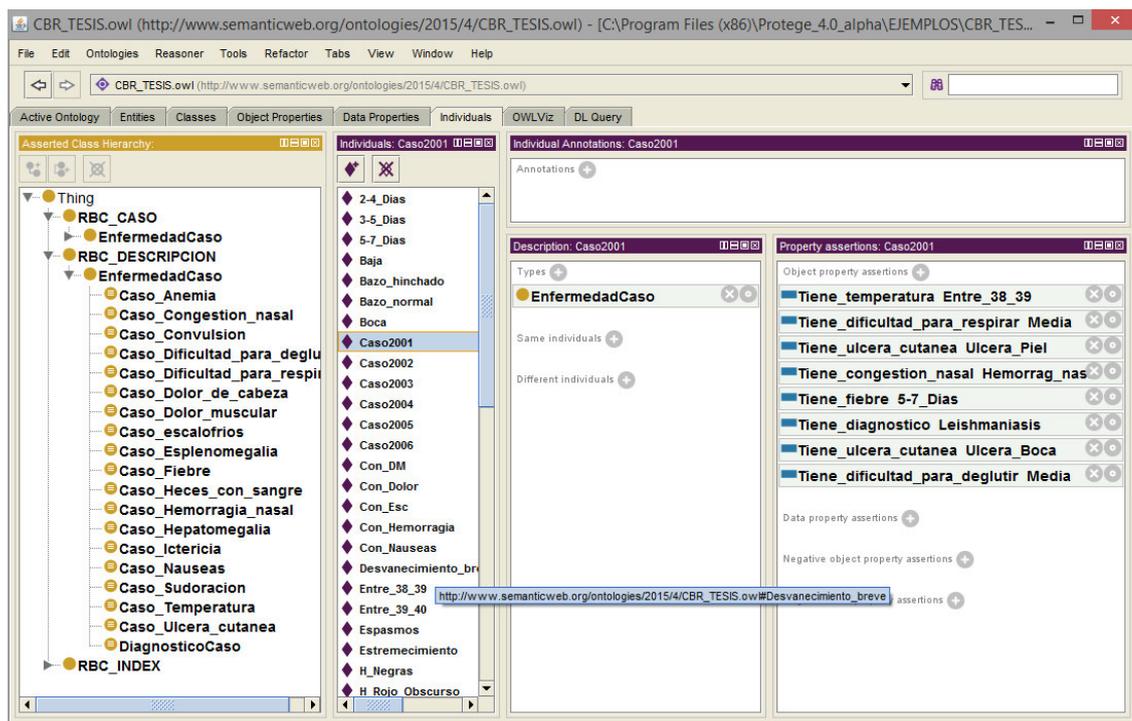
Parámetros de Evaluación	Niveles	Parámetros de Evaluación	Niveles
Anemia	Sin anemia Baja Media Severa	Fiebre	[2-4) días [4-5) días [5-7) días > 7 días
Congestión nasal	Sin congestión Hemorragia nasal Rinorrea	Heces con sangre	Heces negras Heces rojo oscuro
Convulsión	Sin convulsión Desvanecimiento breve Estremecimiento Espasmos.	Hemorragia nasal	Sin hemorragia Con hemorragia
Dificultad para deglutir	Sin dificultad Baja Media Severa	Hepatomegalia	Hígado normal Hígado hinchado
Dificultad para respirar	Sin dificultad Baja Media Severa	Ictericia	Boca Heces Ojos y piel Orina
Dolor de cabeza	Sin dolor Con dolor	Náuseas	Sin náuseas Con náuseas

Parámetros de Evaluación	Niveles	Parámetros de Evaluación	Niveles
Dolor muscular	Sin dolor Con dolor	Sudoración	Sin sudor Sudor frío
Escalofríos	Sin escalofríos Con escalofríos	Temperatura	Normal ≤ 38 [38-39) [39-40) ≥ 40
Esplenomegalia	Bazo normal Bazo hinchado	Úlcera cutánea	Úlcera en la boca Úlcera en las encías Úlcera en la piel Úlcera en el tabique

Fuente: <http://www.icd10data.com/ICD10CM/Codes/A00-B99/B50-B64/B54-/B54>

La figura 5.3 ilustra la ontología obtenida para nuestro estudio, mediante el software Protégé, conteniendo la descripción de clases, propiedades e instancias.

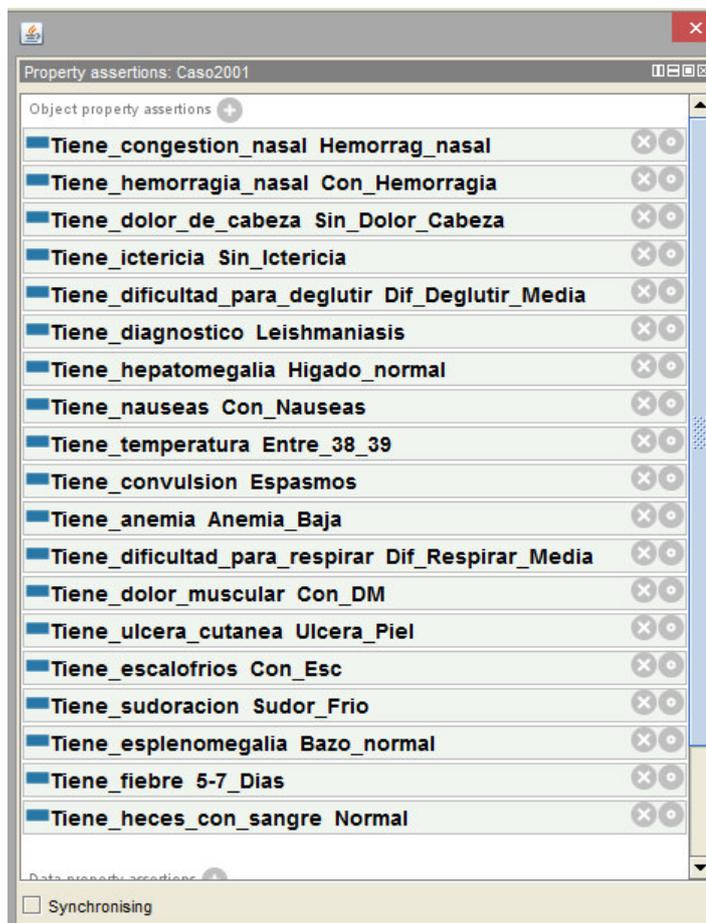
Figura 5. 3. Creación de clases y relaciones en Software Protégé



En Protégé, sólo se ha construido las clases y subclases, así como las propiedades de los objetos y las características individuales (niveles de los síntomas) para cada subclase. Asimismo, cada caso, se ha relacionado con su síntoma correspondiente, como se observa en la Fig. 5.4, para uno de los casos. Esta construcción, luego es

recuperada desde el software jCOLIBRI, para la aevaluación del sistema de razonamiento basado en casos.

Figura 5. 4. Caso2001 y sus síntomas, elaborado en PROTÉGÉ



5.4.2. Implementación del Modelo CBR Propuesto en jCOLIBRI

Para la implementación del sistema de razonamiento basado en casos propuesto, se ha utilizado el software jCOLIBRI, el cual es un entorno orientado a objetos, construido en Java, desarrollado por el grupo GAIA (Group of Artificial Intelligence Applicatios) (véase 3.9). Los programadores expertos pueden usar Java para acomodar el entorno a sus preferencias o estudio en particular, aunque su modo más fácil de uso es su herramienta de configuración gráfica, la cual se ha usado en esta tesis.

En jCOLIBRI, la idea subyacente del CBR, se ha aplicado consistentemente a través de los dominios de aplicación; sin embargo, el desarrollo de un sistema CBR es

una tarea difícil en la cual se debe tomar algunas decisiones complejas. Se ha decidido entre una gama de diferentes métodos para organizar, recuperar y reutilizar el conocimiento retenido de casos anteriores y que se muestra en 5.5.2.1.

Por otro lado, la ontología en jCOLIBRI no se representa como una nueva fuente. Todos los conceptos de CBR son mapeados dentro de clases e interfaces de su entorno de trabajo. Las clases que representan los conceptos de ontología, sirven como plantillas, donde los nuevos tipos de CBR pueden ser agregados. Asimismo, las clases también proporcionan las tareas e interfaces de los métodos. El diseño del entorno de este software, comprende una jerarquía de clases Java, más un número de archivos XML, los cuales están organizados a través de los siguientes elementos:

- Tareas y métodos: Las tareas soportadas por el entorno y los métodos que las resuelven, están almacenados en un conjunto de archivos XML.
- Caso Base: Existen diferentes conectores definidos para soportar varios tipos de determinación de casos, desde los archivos del sistema hasta la base de datos.
- Casos: Un número de interfaces y clases están incluidas en el framework para proporcionar una representación abstracta de casos que soportan cualquier tipo de la actual estructura de casos.
- Métodos de solución del problema: El código puede soportar los métodos incluidos en el entorno de trabajo.

jColibri, incluye una interfaz de usuario (GUI) completamente gráfica, que guía al usuario en el diseño del sistema CBR.

Los pasos para el diseño de nuestro sistema CBR en jCOLIBRI, son los siguientes:

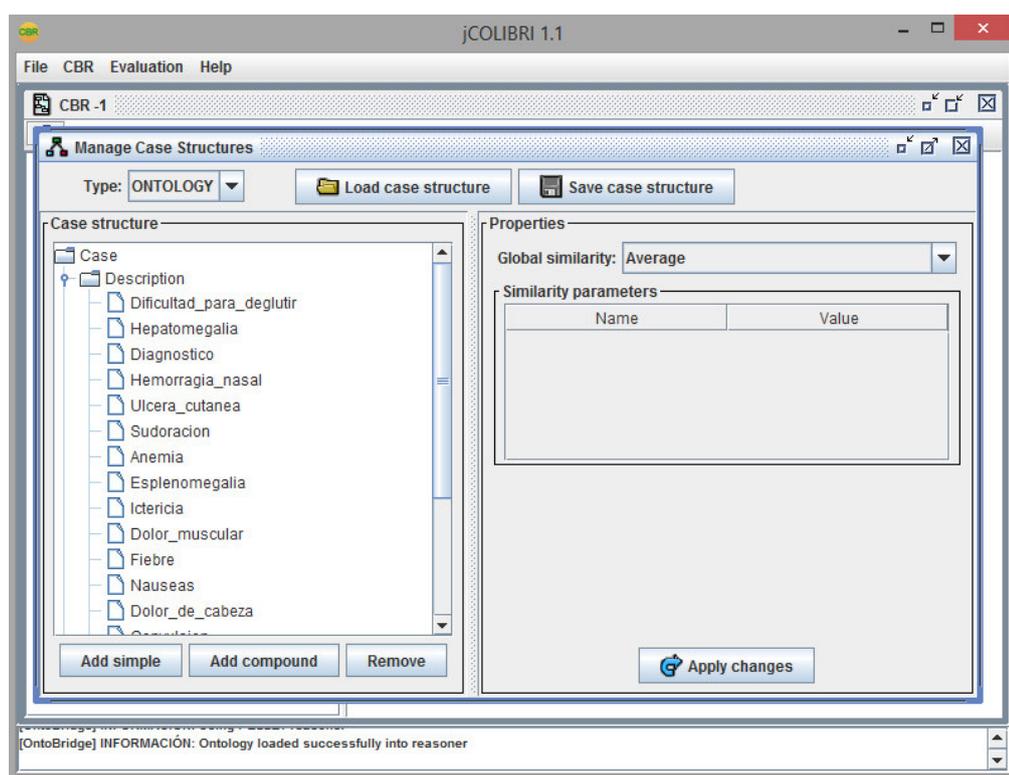
5.4.3. Definición de estructura de casos

Siguiendo la estructura que nos da el entorno de jCOLIBRI, se ha tenido por conveniente establecer tres componentes para el sistema propuesto: descripción (describe el problema), solución (representa un posible enfoque de solución) y el

resultado (revela que la solución propuesta resuelve el problema). Descripción y solución son colecciones de de atributos simples y compuestos, los cuales permiten construir una estructura jerárquica del caso. Al usar jCOLIBRI GUI, se ha creado la estructura del caso, definiendo atributos simples y compuestos que describen los casos conjuntamente con sus tipos, ponderaciones, medidas de similitud, escogidas de una librería existente de funciones de similitud y parámetros u otros que pueden ser fácilmente incluidos. Esto genera un archivo XML con la estructura de la información.

Cuando el usuario ha definido la estructura del caso, luego ha de configurar un conector que use esa información para mapear los casos y elegir los mas parecidos. Este mapeo, también se graba en un archivo XML. La figura 5.5 muestra la definición de los parámetros del caso para el diagnóstico.

Figura 5. 5. Definición de los parámetros para el diagnóstico en jCOLIBRI



En otras palabras, antes de la creación de una aplicación CBR, se necesita configurar la estructura de los casos, partiendo por la descripción que para nuestra tesis viene a ser los síntomas al que le denominamos con la clase EnfermedadCaso en Protégé, con el botón Add Simple del aplicativo jCOLIBRI se agregan los síntomas (atributos), como se observa en la figura anterior y si se desea borrarlos simplemente

con el botón Remove, se retiran. Cuando se marca un caso como síntoma con el mouse, luego en la parte derecha aparecen las propiedades a diseñar para ese caso.

5.4.4. Construyendo el Caso-Base

El concepto de caso se construye en torno a los conectores, representados por los objetos que conocen como acceder y recuperar los casos desde los medios de almacenamiento y devuelve aquellos casos que tienen características comunes. Por lo tanto, los conectores proporcionan un mecanismo de abstracción que permite al usuario recuperar los casos desde diferentes fuentes de almacenamiento [Recio García, 2005]. Los conectores definidos pueden trabajar con archivos de texto plano, archivos XML, bases de datos relacionales u ontologías.

Sin embargo, el uso de ontologías como medio de persistencia, significa que los atributos (slots) de la estructura de los casos, están definidos por los conceptos y propiedades de la ontología y los atributos completos (slots-filler) de las instancias de esos conceptos. De esta manera, no tiene sentido hacer una representación de la estructura del caso y luego asignarla (usando el conector) con la misma estructura de casos contenidos en la ontología.

Para resolver este problema nuestra estructura ontológica pura del caso representa directamente los conceptos y propiedades de la ontología utilizando las clases Java explotadas para el razonamiento. Con este enfoque el conector para los descriptores lógicos (DLs), no necesitan ningún archivo de configuración y pueden recuperar los casos desde la ontología, usando solamente la información de la estructura del caso. La figura 5.6, muestra como la estructura de caso es asignada con la ontología.

5.4.5. Llamada al Caso Base y Configuración de los Conectores

En esta etapa es donde se configura el conector que va a llamar a la base de casos y esto se consigue seleccionando la opción Manage Cnnectors, para configurar el tipo de conector que se desea. Es necesario en esta etapa, dar el detalle de la base de casos y la ruta para llamar al archivo que contiene la estructura.

Figura 5. 6. Asignación de la estructura del caso a la ontología en jCOLIBRI

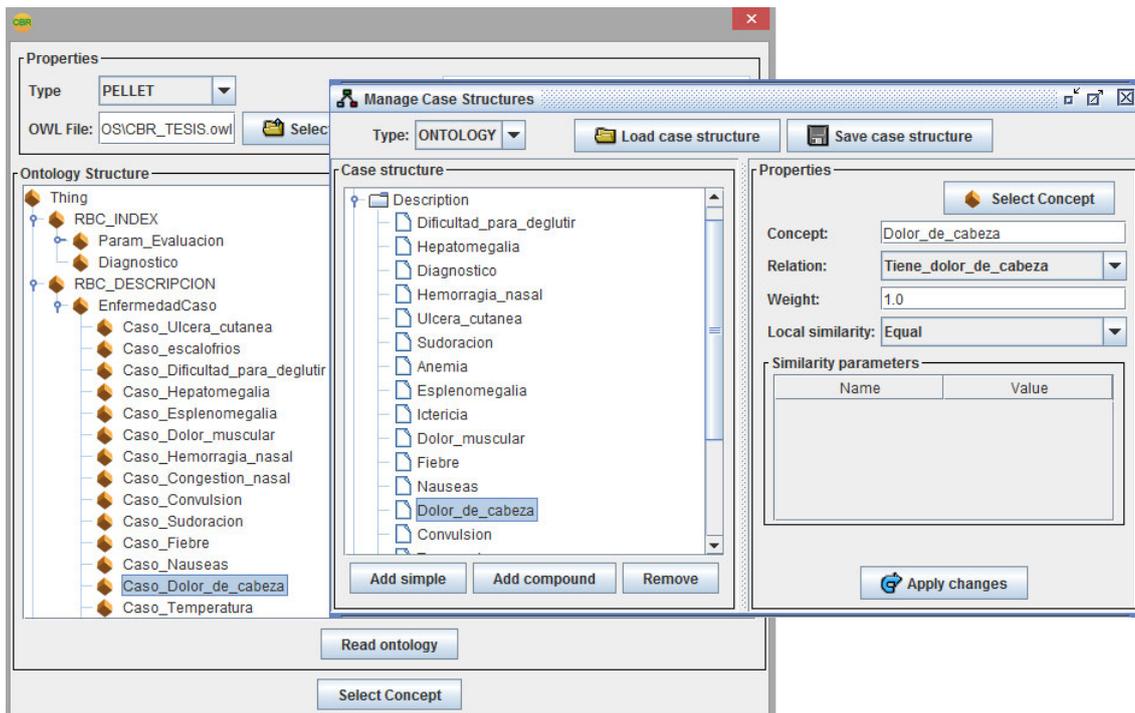
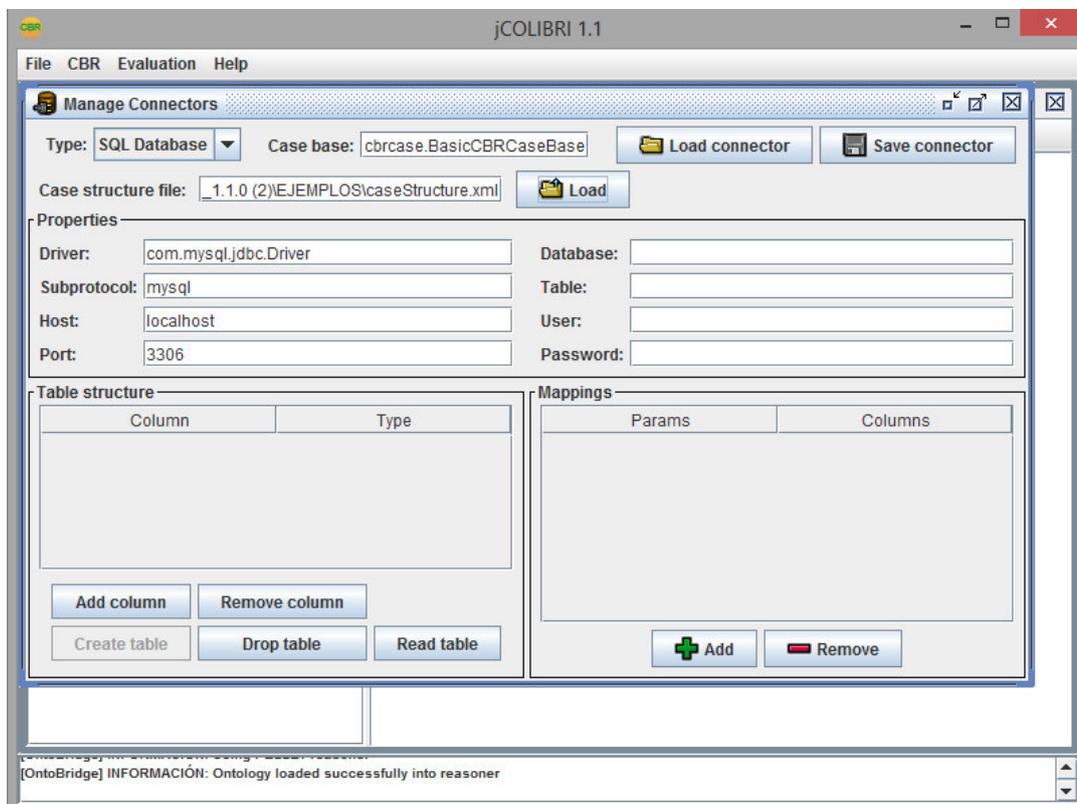


Figura 5. 7. Administración de los conectores en jCOLIBRI



Para nuestro caso como por ser una muestra de 50 datos, no ha habido la necesidad de configurar en esta etapa el llamado a la base de casos, por lo que no se ha establecido dentro de lo solicitado en “Manage Connectors” la estructura de la tabla, así como el mapeo de los parámetros, simplemente se ha realizado un llamado al archivo con la estructura del caso.

5.4.6. Administración de las Medidas de Similitud

Ya se ha cubierto algunos conceptos sobre similitud en el capítulo anterior. Note que para nuestro caso se ha establecido la descripción y solución del caso, siendo necesario encontrar el camino para conocer cuan similares son dos casos y sus atributos.

Las funciones de similitud más usadas son la de equivalencia (igual) y la de intervalo.

- a) **Igual**, En este modo se retorna 1 ó 0 dependiendo si el objeto es idéntico al objeto 2. El método de cálculo de la clase involucrada, está implementado en la clase Object de jCOLIBRI, de la forma: `object1.equals(object.2)`
- b) **Intervalo**, Para este caso la medida de similitud retorna la similitud de dos números dentro de un intervalo. La medida de similitud es aplicable a `java.lang.Double`. La clase toma un parámetro de tipo `double`. La similitud es medida al tomar 1 menos el valor absoluto de la diferencia entre los valores x e y , para luego ser dividido por el intervalo, de la siguiente manera:

$$Sim(x, y) = 1 - \frac{|x - y|}{intervalo}$$

En este caso las funciones de similitud numérica se usan para evaluar y ordenar los casos con respecto a la consulta. Cuando dos casos son comparados, las funciones de similitud local, se usan para comparar valores de los atributos simples. Las funciones de similitud global están vinculadas a los atributos compuestos y se utilizan para encontrar los atributos similares escogidos en un valor de similitud único.

Al menos, los valores de similitud de dos casos, se calculan como la similitud de la descripción de sus conceptos. Las medidas de similitud disponibles son listadas en un archivo de configuración, y puede ser administrada a través de la interface gráfica del software jCOLIBRI. Estas funciones calculan la similitud entre la consulta y el caso, y se usan para escoger el caso mas similar en la consulta. Hay dos tipos de función de similitud: las funciones locales que calculan la similitud entre atributos simples y las globales que se centran en una especie de promedio sobre las similitudes locales.

La similitud entre la consulta (q) y los casos(c), $Sim(q, c)$ se define como sigue:

$$Sim(q, c) = \frac{\sum_{s \in CS} (Sim(q.s, c.s) \cdot w_s)}{|CS|}$$

donde:

w_s es la ponderación asociada para cada atributo s .

CS son todos los atributos simples en q y c .

$|CS|$ su cardinalidad.

$q.s$ (o $c.s$) representan los atributos simples de q (o de c)

$Sim(q.s, c.s)$ es la similitud entre estos dos atributos.

Así $Sim(q.s, c.s)$, está definido como sigue:

$$Sim(q.s, c.s) = \begin{cases} 1, & \text{si } v_{q.s} = v_{c.s} \\ 0, & \text{de lo contrario} \end{cases}$$

Donde $v_{q.s}$ (o $v_{c.s}$) es el valor de este atributo en q (o en c).

5.4.7. Configuración del Comportamiento del Proceso CBR

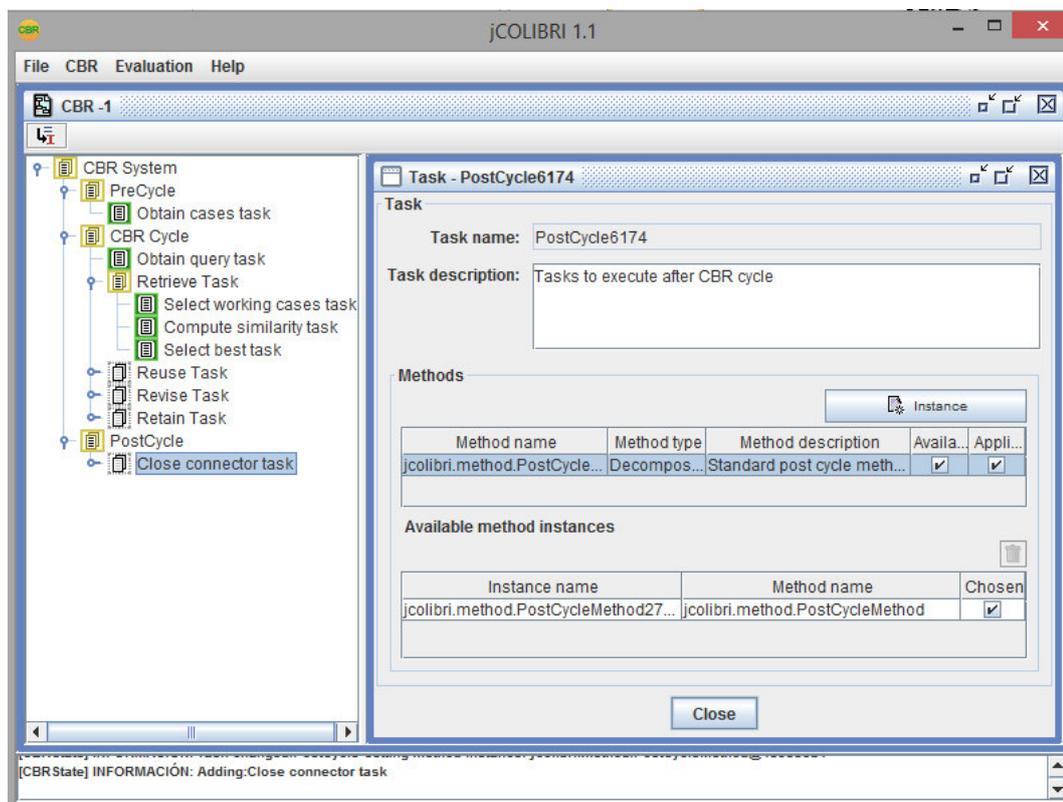
El software jCOLIBRI formaliza el conocimiento dentro del razonamiento basado en casos utilizando CBR_{Onto}, una descripción del nivel de conocimiento de las tareas del CBR y una biblioteca reusable de métodos de solución de problemas

(PSMs) [García J. A. et al, 2005]. La configuración de las tareas se realizan mediante un enfoque interactivo de una biblioteca de métodos reutilizables, se escoge uno que sea el adecuado para resolver la tarea seleccionada, la cual presenta restricciones durante el proceso de configuración, para que solo los métodos aplicables al contexto solicitado se ofrezcan a los usuarios.

Nuestra aplicación CBR está compuesta por un Pre-Ciclo, el cual tiene la tarea de llamar los casos; el Ciclo, el cual tiene cuatro tareas fundamentales que son consulta, recuperación, reuso, revisión y captación; y, un Post-Ciclo que almacena los casos. Cada tarea puede ser desarrollada por varios métodos y que pueden ser escogidos por el usuario. Normalmente hay dos tipos de PSMs (Métodos de resolver el problema):

- Descomposición, divide la tarea principal en sub-tareas.
- Resolución, resuelve la tarea directamente.

Figura 5. 8. Sistema CBR propuesto: Pre-Ciclo. Ciclo y Post-Ciclo



Al hacer clic en cada uno de los ciclos de desarrollo, en la parte derecha aparece la descripción de la tarea, el nombre y descripción del método a utilizar y que puede ser escogido por el usuario.

En la presente tesis, nos centramos sólo en las tareas de representación, de recuperación y adaptación. La aplicación CBR finaliza cuando todas las tareas han sido configuradas y luego los usuarios pueden probar el sistema desde dentro de la interfaz gráfica. La primera tarea del sistema CBR, obtiene la consulta que contiene la descripción del problema, la cual va a ser usada para recuperar los casos más similares, aplicando la tarea de recuperación.

5.4.8. Aplicación del sistema propuesto con la muestra de datos

Como se dijo en 3.3.3, se tomó una muestra de datos correspondientes a dos enfermedades que preocupan a los especialistas de salud que trabajan en la Comunidad Andina de Cruz de Mayo: la malaria y la leishmaniasis, a pesar que esta última ya se está erradicando; sin embargo, por su sintomatología es difícil de tratar. Nuestra muestra consta de 40 casos de malaria, 5 de leishmaniasis (los únicos casos de leishmaniasis); y, por motivos de ilustrar esta investigación, cargamos además 5 pacientes indicados como sanos en su historial médico, pero que concurrieron al Centro de Salud por resfríos comunes.

Los síntomas resaltantes indicados por los especialistas en salud, para las dos enfermedades en estudio se dan en las tablas siguientes:

Tabla 5. 2. Síntomas de la malaria y leishmaniasis

Síntomas de la Malaria	
<ul style="list-style-type: none"> • Anemia • Coma • Convulsión • Dolor de cabeza • Dolor muscular • Escalofríos • Esplenomegalia 	<ul style="list-style-type: none"> • Fiebre intermitente • Heces con sangre • Hepatomegalia • Ictericia • Nauseas • Sudor frío • Vómitos

Síntomas de la Leishmaniasis	
<ul style="list-style-type: none"> • Congestión nasal • Dificultad para respirar • Dificultad para deglutir • Espasmos 	<ul style="list-style-type: none"> • Fiebre intermitente • Hemorragia nasal • Úlceras en la piel • Sudor frío • Úlceras en boca, lengua, tabique nasal

Fuente: <http://www.icd10data.com/ICD10CM/Codes/A00-B99/B50-B64/B54-/B54>

Los pasos a desarrollar para la aplicación del sistema a nuestros datos son:

1. Creamos en el software Protégé, las clases referentes a los síntomas de las dos enfermedades en conjunto y las relaciones entre los individuos y sus síntomas, como se observa en la fig. 5.3.
2. Se definió la estructura de los casos en el software jCOLIBRI, llamando previamente desde éste, el archivo creado en Protégé con las clases respectivas a los síntomas. (Fig. 5.5)
3. Se estableció en el software jCOLIBRI, las propiedades para cada síntoma, como el concepto, la relación, el peso que se le puede dar a cada síntoma dentro de la enfermedad y la similitud local, que para nuestro caso es la de equivalencia (igual), como se mostró en 5.5.2.4. (Fig. 5.6)
4. Se configuró las tareas para el Pre-Ciclo, Ciclo y Post-Ciclo, estableciendo el nombre del método a utilizar para cada una de las tareas y sub-tareas (Fig. 5.8).
5. Finalmente, se muestra la pantalla de consulta para ingresar los síntomas del paciente. (Fig. 5.9)

En este caso podemos observar que la pantalla nos muestra las posibles características del paciente en cuanto a sus múltiples dolencias, para que el usuario, en este caso el especialista en salud que puede ser un médico o enfermero(a) escoja el tipo de dolencia y el nivel de la misma, que el paciente indica; por ejemplo, dentro de la dolencia Heces_con_sangre, hay dos alternativas H_Negras y H_Rojo_obsuro, como se observa en la figura anterior. Todas estas dolencias las consideramos como parámetros (síntomas) del caso en consulta y que será luego evaluado por el modelo.

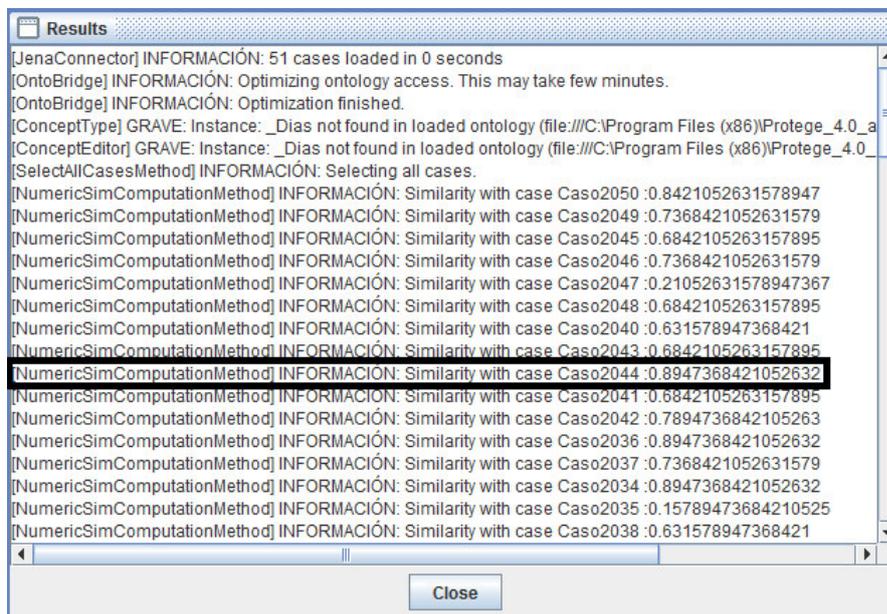
Figura 5. 9. Pantalla de consulta a manejar por el especialista

Requested parameter	Selected value	Weight
Ulcera_cutanea	<empty>	0.0
Sudoracion	Sudor_Frio	1.0
Anemia	Media	1.0
Esplenomegalia	Bazo_hinchado	1.0
Ictericia	Ojos_Piel	1.0
Dolor_muscular	Con_DM	1.0
Fiebre	_Dias	1.0
Dolor_de_cabeza	Con_Dolor	1.0
Convulsion	Estremecimiento	1.0
Temperatura	Entre_38_39	1.0
Escalofrios	Con_Esc	1.0
Congestion_nasal	<empty>	0.0
Heces_con_sangre	H_Negras	1.0
Dificultad_para_respirar	<empty>	0.0

Al lado derecho de cada parámetro, se tiene la ponderación de cada uno de ellos y que puede ser establecido por el especialista en salud, simplemente desplazando la barra deslizante, teniendo a 0 y 1 como los valores de menor y mayor peso.

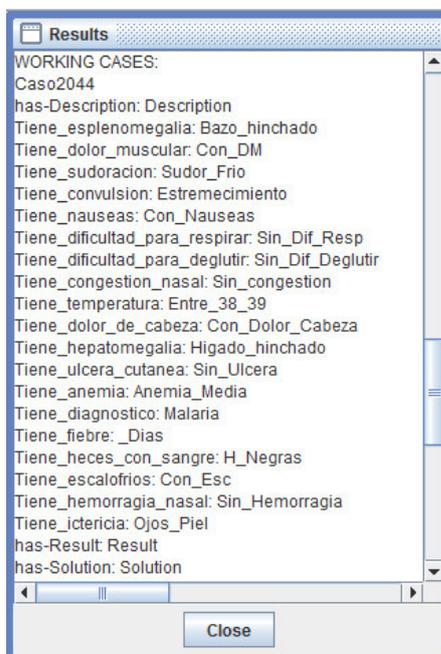
Luego de alimentar la pantalla de consulta, el software está listo para procesar la información ingresada. La fig. 5.10, muestra los resultados de la tarea de recuperación, se puede observar que el caso más similar al consultado es el caso2044 con una similitud del 89,47%

Figura 5. 10. Pantalla de salida, mostrando las similitudes con el caso en consulta



Otra de las salidas que da el software jCOLIBRI es precisamente el caso más parecido al consultado, con sus respectivos síntomas.

Figura 5. 11. Caso con mayor similitud al de la consulta



Revisando la base de casos, se puede observar que el registro correspondiente al caso2044, corresponde a un caso de malaria.

Como un segundo ejemplo, a fin de poner a prueba nuestro sistema, se ha creído por conveniente ingresar en el caso consulta los parámetros correspondientes a los síntomas de la leishmaniasis, con la intención que el sistema logre encontrar a un caso de esta enfermedad, a pesar de haber solo 5 casos de ésta. En las fig. 5.12 y Fig. 5.13, se muestra los parámetros (síntomas) en consulta, debe notarse que en este caso el diagnóstico no se ha ingresado.

Figura 5. 12. Parámetros ingresados

Requested parameters	Value	Weight
Dificultad_para_deglutir	Dif_Deglutir_Media	1.0
Hepatomegalia	Higado_normal	1.0
Diagnostico	<empty>	0.0
Hemorragia_nasal	Con_Hemorragia	1.0
Ulcera_cutanea	Ulcera_Piel	1.0
Sudoracion	Sudor_Frio	1.0
Anemia	Anemia_Baja	1.0
Esplenomegalia	Bazo_normal	1.0
Ictericia	Sin_Ictericia	1.0
Dolor_muscular	Con_DM	1.0

Figura 5. 13. Parámetros ingresados

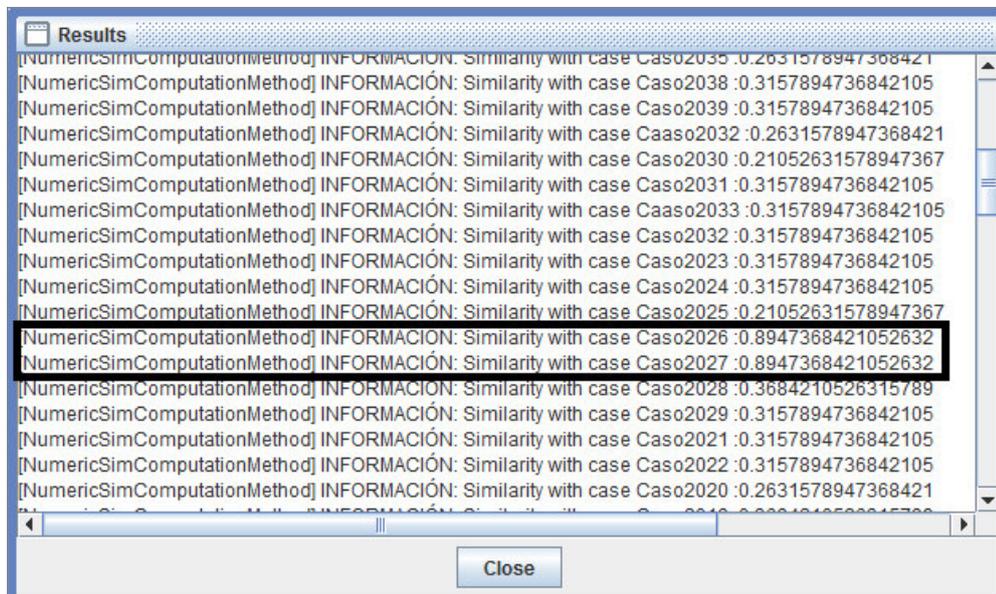
Requested parameters	Value	Weight
Dolor_muscular	Con_DM	1.0
Fiebre	Dias	1.0
Nauseas	Con_Nauseas	1.0
Dolor_de_cabeza	Sin_Dolor_Cabeza	1.0
Convulsion	Espasmos	1.0
Temperatura	Entre_38_39	1.0
Escalofrios	Con_Esc	1.0
Congestion_nasal	Hemorrag_nasal	1.0
Heces_con_sangre	Congestion_nasal	1.0
Dificultad_para_respirar	Dif_Respirar_Media	1.0

Como resultado, se observa en esta ocasión (Fig. 5.14), hay dos casos similares al caso consulta, son el Caso2026 y Caso2027, ambos con una similitud del 89,47%. Efectivamente, si nos remitimos a la base de casos, ambos casos son de pacientes con leishmaniasis.

Resumiendo, todo lo dicho anteriormente en este capítulo, se puede apreciar que jCOLIBRI, nos permite definir la estructura de los casos, la configuración de los conectores para cargar y almacenar la base de casos y las medidas de similitud para comparar atributos de los casos. Asimismo, la parte más importante es la configuración del sistema presentado en esta tesis, la cual es el comportamiento mismo del proceso CBR, y que se ha obtenido utilizando la ontología CBR (CBROnto) de este software,

el cual nos muestra la descripción del nivel de las tareas y una biblioteca de métodos reutilizables (PSM's).

Figura 5. 14. Pantalla de salida, mostrando los casos similares al caso en consulta



5.5. Evaluación de los resultados

En los dos ejemplos propuestos se puede observar que hay una similitud en el diagnóstico de cerca del 90% con el caso en consulta, dentro de una muestra escogida de 50 casos entre los que se encuentran casos de malaria y leishmaniasis.

Como hemos podido observar, lo ideal de este sistema, es que no se crea ningún aplicativo en particular, sino que el sistema propuesto parte del manejo de los softwares libres, y es dentro de ellos en donde se configura las necesidades y requerimientos del sistema mismo.

Finalmente, el resultado es un sistema en el que mediante la creación de una ontología en el software Protégé y el enlazamiento de ésta a través de jCOLIBRI, se puede establecer muy bien un sistema de razonamiento basado en casos, para el diagnóstico de enfermedades como apoyo al médico o profesional de salud y que muy bien se puede portar, simplemente llevando el sistema a accesos remotos, pues ambos softwares son libres.

Luego de elaborado el sistema, se ha creído por conveniente, evaluar el manejo de este sistema, para lo cual se ha manejado la hoja de trabajo TM-2 del CommonKads y, aunque es conocido que ésta debe de establecerse al inicio como base para la construcción del sistema de conocimiento a través de diferentes tareas; en esta tesis la utilizamos no como hoja de tarea sino con algunas variaciones en los ítems para ver las deficiencias del sistema al ser manejado por un especialista en salud.

Tabla 5. 3 Evaluación del sistema

Modelo de Tarea		Análisis del ítem de conocimiento. TM-2	
Nombre :		Conocimiento del sistema	
Ejecutado por :		Agente especialista en salud	
Usado en :		Diagnóstico de malaria y leishmaniasis	
Dominio :		Médico general	
Naturaleza del conocimiento		¿Cuello de botella / a mejorarse?	
Formal, riguroso	Si	Debe capacitarse al especialista en el manejo del sistema	
Cuantitativo	Si		
Altamente especializado, dominio específico	Si	Dominio sólo de dos enfermedades	
Basado en experiencia	Si		
Basado en acciones	No		
Incompleto	Si	Puede haber otros síntomas	
Incierto, puede ser incorrecto	Si	Cerca de un 10% no concuerda con el caso en consulta	
Cambiante rápidamente	No		
Difícil de verificar	No		
Tácito, difícil de transferir	No		
Forma de conocimiento			
Mente	Si		
Electrónico	Si		
Máscara de acciones	Si	Difícil el manejo de las ventanas	
Disponibilidad de conocimiento			
Limitaciones en tiempo	Si	Por la función del médico	
Limitaciones en espacio	No		
Limitaciones de acceso	No		
Limitaciones en calidad	No		
Limitaciones en forma	No		

5.6. Confiabilidad del modelo propuesto

Para medir la confiabilidad de los resultados obtenidos con nuestra aplicación RBC, se ha creído por conveniente en primer lugar, realizar un análisis descriptivo de los 50 casos en estudio (45 enfermos y 5 sanos) comparando el diagnóstico dado por el médico, escrito en los padrones de historias clínicas de la Comunidad Cruz de Mayo y la simulación dada por el modelo nuestro ejecutado para los 45 casos de enfermedad, encontrándose los siguientes resultados mostrados en la tabla 5.4

Tabla 5. 4 Diagnóstico del experto y el modelo RBC propuesto

Caso Evaluado	Experto (médico)	Modelo RBC	Caso Evaluado	Experto (médico)	Modelo RBC
Caso_2001	Leishmaniasis	Leishmaniasis	Caso_2026	Leishmaniasis	Leishmaniasis
Caso_2002	Sano	Malaria	Caso_2027	Leishmaniasis	Leishmaniasis
Caso_2003	Malaria	Malaria	Caso_2028	Malaria	Malaria
Caso_2004	Leishmaniasis	Leishmaniasis	Caso_2029	Malaria	Malaria
Caso_2005	Leishmaniasis	Leishmaniasis	Caso_2030	Malaria	Malaria
Caso_2006	Malaria	Malaria	Caso_2031	Malaria	Malaria
Caso_2007	Malaria	Malaria	Caso_2032	Malaria	Malaria
Caso_2008	Malaria	Malaria	Caso_2033	Malaria	Malaria
Caso_2009	Malaria	Malaria	Caso_2034	Malaria	Malaria
Caso_2010	Malaria	Malaria	Caso_2035	Sano	Sano
Caso_2011	Malaria	Malaria	Caso_2036	Malaria	Malaria
Caso_2012	Malaria	Malaria	Caso_2037	Malaria	Malaria
Caso_2013	Malaria	Malaria	Caso_2038	Malaria	Malaria
Caso_2014	Malaria	Malaria	Caso_2039	Sano	Sano
Caso_2015	Malaria	Malaria	Caso_2040	Malaria	Malaria
Caso_2016	Malaria	Leishmaniasis	Caso_2041	Malaria	Malaria
Caso_2017	Malaria	Malaria	Caso_2042	Malaria	Malaria
Caso_2018	Malaria	Malaria	Caso_2043	Malaria	Malaria
Caso_2019	Malaria	Malaria	Caso_2044	Malaria	Malaria
Caso_2020	Malaria	Malaria	Caso_2045	Malaria	Malaria
Caso_2021	Malaria	Malaria	Caso_2046	Malaria	Malaria
Caso_2022	Malaria	Malaria	Caso_2047	Sano	Sano
Caso_2023	Malaria	Malaria	Caso_2048	Malaria	Malaria
Caso_2024	Malaria	Malaria	Caso_2049	Malaria	Malaria
Caso_2025	Sano	Sano	Caso_2050	Malaria	Malaria

Tabla 5. 5 Porcentaje de acierto entre el experto y el modelo CBR

Diagnóstico	% de acierto
Leishmaniasis	(5/5) 100,0
Malaria	(39/40) 98,0
Sano	(4/5) 80,0
Total	50 100,0

Se puede observar que el porcentaje de aciertos del modelo comparado con el diagnóstico médico, es bastante alto; para la leishmaniasis el acierto es en un 100%, para la malaria en un 98% y el diagnóstico de sano está en un 80%.

Asimismo, se recurrió al análisis estadístico, con la finalidad de dar un sustento más científico y valedero al estudio y así comprobar el grado de concordancia entre el modelo RBC propuesto y el diagnóstico médico (obtenido de los archivos clínicos) para los 45 pacientes de la muestra. Para esto se utilizó el índice estadístico Kappa (k), cuyo objetivo es proporcionar una medida del grado de acuerdo existente entre dos instrumentos, observadores o jueces al evaluar una serie de sujetos u objetos (Cohen, 1960); es decir, evalúa la concordancia o reproductibilidad de instrumentos de medida cuyo resultado es categórico (2 o más categorías), representando la proporción de acuerdos observados más allá del azar.

Para el desarrollo de esta prueba se utilizó el software estadístico IBM SPSS Statistics 22 en su versión de prueba para calcular el estadístico kappa mostrado en 3.5, partiendo con el planteamiento de las siguientes hipótesis:

H_0 : No existe concordancia entre el modelo RBC y diagnóstico médico

H_1 : Existe concordancia entre el modelo RBC y diagnóstico médico

En la tabla 5.6 se muestran los valores observados y esperados para el diagnóstico del modelo RBC y el mostrado en las historias clínicas

Tabla 5.6 Concordancia observada y esperada para el acuerdo entre el modelo RBC y el diagnóstico médico

DIAGNÓSTICO Modelo RBC	DIAGNÓSTICO MÉDICO				Total Frec.
	Malaria		Leishmaniasis		
	Conc. observada	Conc. esperada	Conc. observada	Conc. esperada	
Malaria	39	34,7	0	4,3	39
Leishmaniasis	1	5,3	5	0,7	6
TOTAL	40	40,0	5	5,0	45

$$k = \frac{P_0 - P_e}{1 - P_e} = \frac{0,9778 - 0,7867}{1 - 0,7867} = 0,897$$

Tabla 5. 7 Valor del estadístico de la prueba kappa de Cohen y significancia de la prueba

	Valor	Significancia
Medida de acuerdo kappa	0,897	1,4785x10-9

La tabla 5.7 recoge el valor del estadístico kappa y su nivel crítico (significación aproximada), el cual permite decidir sobre la hipótesis de acuerdo nulo: puesto que el nivel crítico es muy pequeño (tiende a cero), rechazamos la hipótesis de acuerdo nulo y concluimos que existe un acuerdo significativamente más alto que el esperado por el azar.

Además dentro de la valoración del índice kappa, la fuerza de concordancia se encuentra en el nivel más alto (0,81 - 1,00), es decir, el acuerdo entre el modelo RBC propuesto en esta tesis y el diagnóstico médico mostrado en las historias clínicas es “muy bueno”.

5.7. Conclusiones

5.7.1. Conclusión General

El punto de partida de esta tesis ha sido la motivación de construir un modelo RBC que sirva tanto como ayuda para el médico como para el poblador alejado de poblaciones con los recursos de asistencia hospitalaria; en ese sentido, la tesis presenta un modelo de razonamiento basado en casos, orientado al diagnóstico médico como apoyo al especialista en salud, con la finalidad de mejorar la atención médica del poblador en un centro rural, empleándose para tal fin el entorno del software jCOLIBRI.

Esta investigación ha estado limitada sólo a dos enfermedades: malaria y leishmaniasis; asimismo, se han presentado algunas limitaciones para la toma de datos, especialmente en lo que concierne a la toma de información de los diferentes centros poblados de la Comunidad Cruz de Mayo; sin embargo, pienso que es un aporte innovador en el ámbito del razonamiento basado en casos, especialmente dirigida al diagnóstico médico que luego se puede generalizar para otros diagnósticos

La implementación fue exitosa, y se trató de ejecutarla varias veces con las dos enfermedades en estudio (malaria y leishmaniasis), suficientes para demostrar que se ejecuta de acuerdo a los requerimientos o parámetros de ingreso (síntomas), brindando luego la pantalla de salida de una manera simple y sencilla, mostrando cuáles son los casos mas idénticos al ingresado; con lo cual podemos decir que efectivamente se ha cumplido con el objetivo general de este estudio, es decir el de crear un modelo RBC de este tipo y su posterior implementación.

De igual manera, se ha comprobado específicamente con la prueba estadística kappa de Cohen (k) la concordancia de diagnóstico para las enfermedades en mención, entre el modelo RBC propuesto y el diagnóstico médico, encontrado en las fichas clínicas, resultando un índice de 0,89 muy cercano a 1, indicando que el acercamiento entre ambos diagnósticos es muy bueno dentro de la escala de valoración de este indicador.

En cuanto a los cinco pacientes diagnosticados como sanos en los padrones de historias clínicas, el modelo clasificó a uno como enfermo de malaria, lo cual puede ser considerado como un error del modelo, pues al estar de por medio la vida, no se puede realizar un diagnóstico equivocado. Este diagnóstico erróneo lo da el modelo debido a que específicamente el sistema fue alimentado con los síntomas de la malaria y leishmaniasis y dentro de la base de casos solo hay cinco pacientes sanos, generándose una confusión al recuperar los casos entre los síntomas de un simple resfrío y considerarlos como síntomas de la malaria. Por esto, se recomienda al momento de crear el sistema en sí, se prevea también un buen número de casos de pacientes sanos o con simple resfrío. Aunque esto es un error, nos permite también valorar al modelo en un sentido positivo, pues indica que recupera efectivamente a los casos que mas concuerdan con los síntomas ingresados.

Es necesario indicar también como conclusión, que los softwares libres Protegé y jCOLIBRI, fueron de gran ayuda para ejemplificar brevemente el modelo RBC propuesto; el primero sirve para crear la base de casos e inclusive formar las ontologías y el segundo sirve como marco de referencia para la creación del prototipo de RBC, aprovechando la creación de base de casos integrándola en su entorno para finalmente poner en práctica el modelo a través de un sistema que aunque es un poco difícil de manejar, opino que brinda los recursos necesarios para la implementación del sistema CBR.

Se concluye asimismo, que este modelo de sistema de razonamiento basado en casos, tiene como ventaja evaluar las soluciones cuando no existe ningún método algorítmico para hacerlo, pues resulta mas sencillo adquirir nuevos casos que descubrir reglas y generalizaciones. En ese sentido, nuestro aporte servirá como apoyo no sólo para el diagnóstico médico, (sin necesidad de establecer algoritmos) sino también para la solución de emergencias e inclusive para realizar una acción preventiva de salud a favor de los pobladores alejados a muchos kilómetros de distancia de un verdadero centro poblado que tenga los recursos médicos suficientes.

5.7.2. Conclusiones Específicas

A parte de que el modelo CBR propuesto, reduce la tarea de adquisición de conocimiento, eliminando la necesidad de extraer conocimiento de un conjunto de reglas, en cambio, sólo se almacenan casos, evitado así un gran desgaste en la confección de algoritmos, lo cual ya es un gran logro; se presenta a continuación las conclusiones para cada uno de los objetivos específicos.

- **Para el objetivo específico 1.**

Se creó el modelo para el almacenamiento de una base de casos de los síntomas de dos enfermedades: la malaria y leishmaniasis (5.1 y 5.4.7); mediante la configuración del comportamiento CBR, el sistema evalúa al nuevo caso mediante una consulta de sus síntomas o parámetros, arrojando inmediatamente cuál es el caso más similar al caso consultado.

- **Para el objetivo específico 2.**

Se construyó la estructura de casos sobre los síntomas más comunes de las dos enfermedades: malaria y leishmaniasis (5.4.3), mediante el software jCOLIBRI, llamando previamente al archivo creado en Protégé, donde se elaboró la ontología dominio, es decir los síntomas y la relación con los casos.

- **Para el objetivo específico 3.**

Se definió el criterio de similitud de dos casos y sus atributos (5.4.6), mediante la función de similitud de equivalencia (igual), pudiendo dentro del sistema creado en jCOLIBRI establecer la función más adecuada al tipo de enfermedad y de acuerdo al conocimiento médico del especialista.

5.8. Recomendaciones

- Considerando que el trabajo expuesto en esta tesis considera sólo al diagnóstico de dos enfermedades, se recomienda desarrollar el estudio abarcando mas enfermedades, englobadas en dominios específicos, para lo cual existe la clasificación estadística de enfermedades de la Organización Mundial de la Salud (ICD-10-World Health Organization). Entre ellas están las enfermedades infecciosas y parasitarias muy comunes en nuestro país.
- El sistema propuesto, parte del manejo de softwares libres como son PROTÉGÉ y jCOLIBRI, y para realizar el manejo de ellos por parte de un médico que tiene un conocimiento ínfimo de manejo de aplicativos informáticos, se le hace un poco complicado, por lo que se propone una capacitación sostenida, pues luego de ella, se verá que el manejo no es difícil.
- Para estudios posteriores se recomienda construir el mismo sistema pero ya no con el manejo por separado de ambos softwares sino construyendo un aplicativo que combine ambos, bajo un mismo entorno, el cual sería mucho más amigable al usuario que lo maneja.
- Aunque al tener el diagnóstico, el médico ya puede dar el tratamiento adecuado, se sugiere ampliar el estudio presentando además del diagnóstico, el tratamiento; incluyendo por ejemplo las dosis adecuadas de tal o cual medicamento.
- Sería interesante implementar el mismo modelo con el software MyCBR, herramienta de código abierto, construido también en base al software Protegé y permite la creación fácil y rápida de prototipos para un sistema RBC, ya que éste permite elegir visualmente de un grupo de medidas de similitud global o local, cual es la mas adecuada para la comparación entre el caso y el valor de la consulta.

CAPÍTULO VI

IMPACTO DEL ESTUDIO

6.1. Propuesta para la solución del problema

Esta tesis, surge como respuesta ante la problemática de salud sobre todo en los entornos alejados de capitales de provincia y distrito de nuestro país, donde el acceso de los pobladores a un sistema de salud, se hace frecuentemente a través de postas médicas y donde el médico o personal de salud, no es un especialista en tal o cual enfermedad, por lo que necesita de un apoyo para diagnosticar mejor la dolencia o enfermedad del paciente que recurre a ellos.

En ese sentido, el sistema propuesto puede muy bien servir de apoyo en el diagnóstico, obviamente no supe al médico; pero si, al contar con una base de casos con síntomas y sus respectivos diagnósticos, el especialista podrá hacer la consulta respectiva, simplemente ingresando los síntomas y sus respectivas ponderaciones de los mismos del poblador enfermo (nuevo caso); y luego, encontrar los casos más parecidos al consultado. Obviamente en nuestro estudio sólo se ha contado con dos enfermedades, pero puede hacerse más amplia la cobertura hacia otras, sobre todo del entorno rural.

6.2. Beneficios que aporta la propuesta

Así pues el trabajo de tesis presentado contribuiría a atenuar en parte el precario escenario del sistema de salud en aquellas zonas, y por lo tanto brindaría un apoyo en los ámbitos siguientes:

- En el ámbito social, porque son los pobladores de zonas rurales y comunidades alejadas sea de sierra o selva, los que recibirán este beneficio sostenido en el tiempo, mejorando la calidad de salud y por lo tanto su calidad de vida; asimismo, si se estableciera como medida común la instalación de un sistema de este tipo a nivel de todas las zonas alejadas, el médico o especialista en salud, que en muchos casos pertenece al servicio rural y urbano marginal de salud (SERUMS) destacado a esas zonas, sentiría la confianza necesaria en dar el tratamiento adecuado, partiendo de un diagnóstico certero, al comparar la consulta con los casos base.
- En el ámbito de salud, porque en estas zonas, mayormente la presencia médica se ve limitada por la falta de equipos, como rayos X, de análisis de sangre, etc.; siendo este sistema propuesto, una ayuda al médico al comparar los diferentes síntomas del poblador con la base de casos para luego dar el diagnóstico apropiado y por lo tanto, el respectivo tratamiento o si fuera el caso, preparar al paciente hasta trasladarlo a un centro hospitalario. Es decir, con este sistema de razonamiento basado en casos, se tendría un apoyo no sólo para la solución de emergencias, sino inclusive, para realizar una acción preventiva de salud a favor de los pobladores alejados a muchos kilómetros de distancia de un verdadero centro poblado que tenga los recursos médicos suficientes.
- En el ámbito económico, porque aunque no suple a un equipo médico, ni a un hospital, retribuiría en gran medida al ahorro de realizar gastos por enfermedades que no han sido tratadas a tiempo y que fácilmente se pueden prevenir, al inspeccionar los síntomas y compararlos con la base de casos, que el médico de esa zona puede diagnosticar, evitando la congestión de la enfermedad y por lo tanto ahorrándole al poblador tiempo y dinero para el tratamiento de su enfermedad.

6.3. Sugerencias

Asimismo, con la finalidad de contribuir mucho mejor al desarrollo y mejorar este sistema, partiendo del razonamiento basado en casos, se sugiere para estudios posteriores lo siguiente:

- Para ser mas exhaustivo en la creación de sistemas RBC, se sugiere realizar estudios, considerando el análisis cluster o de conglomerados para agrupar las características mas comunes a las enfermedades, aunque esto involucra algunos requisitos estadísticos de análisis multivariado, que en muchas ocasiones no es fácil cumplir.
- Actualmente se habla de minería del conocimiento, por lo tanto es factible que se integre estas dos técnicas como son la minería de datos y el razonamiento basado en casos, considerando que la minería de datos son a la vez técnicas de análisis multivariado integrados a los sistemas informáticos.

6.4. Costos de implementación del sistema propuesto

Los costos son realmente ínfimos y que lo podríamos clasificar en los siguientes ítems:

- Adquisición de un ordenador portátil (laptop, Tablet o notebook, etc.)
- Adquisición de los softwares libres PROTÉGÉ y jCOLIBRI.
- Construcción del sistema.
- Digitación para la entrada de datos sobre síntomas y diagnósticos.
- Capacitación de personal de salud en el manejo del sistema.

BIBLIOGRAFÍA

- [A. Aamodt, E. Plaza ,1994]. A. Aamodt, E. Plaza (1994), Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AI Communications. IOS Press, Vol. 7: 1, pp. 39-59.
- [Aamodt, A., 2004]. Agnar Aamodt. Knowledge-intensive case-based reasoning in creek. *Advances in Case-Based Reasoning*, 7th European Conference, ECCBR 2004, pages 1-15, 2004.
- [Abbadi et al., 2010]. Abbadi, N.K.A., N.S. Dahir, M.A.A. Dhalimi and H. Restom, 2010. Psoriasis detection using skin color and texture features. *J. Comput. Sci.*, 6: 648-652, DOI: 10.3844/jcssp.2010.648.652
- [Acorn y Walden, 1992]. T. L. Acorn y S. H. Walden, 1992. Smart: Support: Management automated reasoning technology for compaq customer service. En A. C. Scott y P. Klahr, editores, IAAI, páginas 3–18. AAAI. ISBN 0-262-69155-8.
- [Agnar Aamodt & Enric Plaza, 1994]. Case-based reasoning: Foundational issues,
- [Agudo, B.D., et al. 2000]. B. D-Agudo and P. G-Calero, “An architecture for knowledge intensive CBR systems”, *Advances in Case-Based Reasoning*”. (EWCBR’00). Springer-Verlag, Berlin Heidelberg New York, (2000).
- [Alonso J., et. al., 2012]. Alonso Jimenez, José A. y Gutiérrez Naranjo, Miguel A. Introducción a los Sistemas Basados en Conocimiento. Recuperado de <http://www.cs.us.es/cursos/iic-2012/temas/tema-02-iic03.pdf>
- [B. D-Agudo et.al, 2001]. B. D-Agudo and P. A. G-Calero, “Knowledge intensive CBR through ontologies”, In *Procs of the UK CBR Workshop*. (2001).
- [Baader, et.al., 2003]. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., & Patel-Schneider, P. (2003). *The Description Logic Handbook: Theory, Implementation and Applications*. (F. Baader, D. Calvanese, D. McGuinness, D. Nardi, & P. Patel-Schneider, Eds.) (1st ed., pp. 1–574). Cambridge: Cambridge University Press.
- [Bachimont, 2000]. Bachimont, B. (2000). *Engagement sémantique et engagement ontologique : conception et réalisation d ’ ontologies en Ingénierie des*. Paris.
- [Bareiss 1998] Bareiss, R., “PROTOS; a unified approach to concept representation, classification and learning”, University of Texas, Tesis doctoral.

- [Beksac et al., 1995]. Beksac, M.S., B. Durak, O. Okzan, A.N. Cakar and S. Balci et al., 1995. An artificial intelligent diagnostic system with neural networks to determine genetical disorders and fetal health by using maternal serum markers. *Eur. J. Obstetrics Gynecol. Reprod. Biol.*, 59: 131-136. PMID
- [Bertaud-Gounot, 2011]. Bertaud-Gounot, V., Donfack Guefack, V., Lasbleiz, J., Bourdé, A., & Duvaufferrier, R. (2011). Creating an ontology driven rules base for an expert system for medical diagnosis. *Studies in Health Technology and Informatics*, 169(1), 714–718.
- [Bichindaritz I. y Sullivan K., 2002]. Bichindaritz, I., and Sullivan, K. 2002. Generating practice cases for medical training from a knowledge-based decision-support system. In *Workshop Proceedings*, 3–14. ECCBR'02.
- [Bodenreider, 2008]. Bodenreider, O. (2008). Biomedical ontologies in action: role in knowledge management, data integration and decision support. *Yearbook of medical informatics*, 1(1), 67–79.
- [Borst Willem, 1997]. Borst Willem, N. (1997). Construction of engineering ontologies for Knowledge Sharing and Reuse. The Dutch Graduate School for Information and Knowledge Systems. Recuperado de <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.79.917>
- [Bradburn, C., 1993]. C. Bradburn and J. Zeleznikow, The application of case-based reasoning to the tasks of health care planning. In: S. Wess et al.: *Proc of European Workshop on CBR*, Springer, Berlin, 1993, pp. 365-378
- [Burgun, 2006]. Burgun, A. (2006). Desiderata for domain reference ontologies in biomedicine. *Journal of biomedical informatics*, 39(3), 307–13.
- [Burgun, A., & Bodenreider, O. 2001]. Mapping the UMLS Semantic Network into general ontologies. In *Proceedings / AMIA. Annual Symposium. AMIA Symposium* (pp. 81–5).
- [Cabrera L., 2011]. Cabrera L. Sistemas basados en el conocimiento. Recuperado de <http://www.cs.us.es/blogs/iic2012/files/2012/02/IIC-Teoria2.pdf>
- [Cardoso. 2007]. Cardoso, J. (2007). The Semantic Web Vision : Where are We ? *IEEE Intelligents systems*, 22(5), 84–88.
- [Carrillo A. et. al. 2011]. Alberto Felipe Carrillo de Comas, Jose María Llovet Rodríguez e Israel Rodríguez Paulete. Sistema recomendador orientado a la educación terapéutica del paciente diabético. Recuperado de http://eprints.ucm.es/13097/1/Sistema_recomendador_orientado_a_la_educación_terapeutica_del_paciente_diabético.pdf
- [CEAS, 2013]. Comisión episcopal de acción social (2013). *La Comunidad Campesina Cruz de Mayo: Una Comunidad que defiende y promueve la vida.*

- [Cohen, 1960]. Cohen J. A coefficient of agreement for nominal scales. *Educ Psychol Meas* 1960; 20: 37-46.
- [Contraloría General de la República, 2006]. CGR en Coordinación con el Ministerio Federal de Cooperación Económica y Desarrollo-Alemania. Una Experiencia de Gestión del Conocimiento.
- [Cortez V. Augusto et al., 2010]. Cortez V. Augusto, Navarro D. Carlos, Pariona Q. Jaime. SRBC Aplicados a Sistemas de Líneas de Productos de Software. *Revista de Investigación de Sistemas e Informática*. UNMSM (2010).
- [Cruz Q. Laura, 2009]. Cruz Q. Laura. Propuesta de Modelo Híbrido para Análisis de Riesgo Crediticio en Pymes: Caso Crédito MES Caja Municipal de Arequipa. Universidad Nacional San Agustín de Arequipa.
- [Crowley et al., 2003]. Crowley, R., Naus, G., & Friedman, C. (2003). Development of visual diagnostic expertise in pathology: An information processing study. *Journal of American Medical Informatics Association*, 10(1), 39-51.
- [Cuggia, Besana, y Glasspool, 2011]. Cuggia, M., Besana, P., & Glasspool, D. (2011). Comparing semi-automatic systems for recruitment of patients to clinical trials. *International journal of medical informatics*, 80(6), 371–88. doi:10.1016/j.ijmedinf.2011.02.003
- [Cunningham, 2009] Cunningham, P. A taxonomy of similarity mechanisms for case-based reasoning. *Knowledge and Data Engineering, IEEE Transactions on*, 21(11):1532–1543, 2009.
- [Davis and Hamscher, 1998]. Davis, R. and W.C. Hamscher, 1998. *Model-Based Reasoning: Troubleshooting*. 1st Edn., Defense Technical Information Center, San Francisco, CA, USA., pp: 56.
- [Díaz & Gonzáles, 2002a]. Díaz-Agudo, B., & González-Calero, P. A., 2001. “Knowledge Intensive CBR through Ontologies”. *Proc. of the 6th UK Workshop on Case-Based Reasoning (UKCBR’01)*, Lees, B., (Ed.), University of Paisley, pp. 43-52.
- [Díaz & Gonzáles, 2002b]. Díaz-Agudo, B., Gervás, P. & González-Calero, P., 2002. “Poetry Generation in COLIBRI”. *Procs 6th European Conference on Case Based Reasoning, Aberdeen, Scotland, 4-7 September 2002 (ECCBR’02)*. Springer. To appear.
- [Delgado, M., 2009]. Delgado, Martha Dunia. *Definición del Modelo de Negocio y del Dominio utilizando Razonamiento Basado en Casos*. Centro de Estudios en Ingeniería de Sistemas, Cuba. (2009)
- [Dolia, 2010]. Dolia, P. (2010). Integrating Ontologies into Multi-Agent Systems Engineering (MaSE) for University Teaching Environment. *Journal of Emerging Technologies in Web Intelligence*, 2(1), 42–47.

- [Durieux et al., 2000]. Durieux, P., Nizard, R., Ravaud, P., Mounier, N., & Lepage, E. (2000). A clinical decision support system for prevention of venous thromboembolism: effect on physician behavior. *Journal of the American Medical Association*, 283(21), 2816-21.
- [E. Sirin, B., et al. 2007]. E. Sirin, B. C. Parsia, B. C. Grau, A. Kalyanpur and Y. Katz, “Pellet: A practical OWL-DL reasoner”, *Journal of Web Semantics*, vol. 5, no. 2, (2007).
- [El-Fakdi, 2013]. Andres El-Fakdi. eXiTCDSS: A framework for a workflow-based CBR for interventional Clinical Decision Support Systems and its application to TAVI. HAL- Centre pour la Communication Scientifique Directe - UMS3668, 2013
- [Feigenbaum E., 1977]. Edward Feigenbaum. “The Art of Artificial Intelligence”. *International Joint Conference on Artificial Intelligence*, pages 1014-1029, Cambridge, MA.
- [F. Alonso y N. Juristo, 1996]. F. Alonso, N. Juristo, J. L. Maté, and J. Pazos. *Software engineering and knowledge engineering: Towards a common life cycle*. *Journal of Systems and Software*, Volume 33, Issue 1, April 1996, p. 65–79.
- [GAIA, 2011]. Juan Recio-García, Belén Díaz-Agudo, and Pedro A. González Calero. jcolibri 2 tutorial. Technical report, University Complutense of Madrid, September 16, 2008. <http://gaia.fdi.ucm.es/projects/jcolibri/jcolibri2/docs.html> ISBN 978-84-691-6204-0.
- [Gómez-Pérez, 1999]. Gomez-Perez, A. (1999). Ontological Engineering: A state of the art. *Expert Update*, 1(1), 33–43.
- [García J. A. et al, 2005] J. A. R-García, A. Siánchez, B. D-Agudo and P. A. G-Calero, “jCOLIBRI 1.0 in a nutshell. A software tool for designing CBR systems”, in M Petridis, editor, *Proceedings of the 10th UK Workshop on Case Based Reasoning*, CMS Press, University of Greenwich, (2005), pp. 20–28.
- [García, J. et al., 2006]. J. R-Garcia, B. D-Agudo, P. G-Calero and A. Sanchez, “Ontology based CBR with jCOLIBRI”, In *Procs. of the 26th SGAI Int. Conf. (AI-2006)*, (2006), pp. 149–162, UK, Springer.
- [García-Crespo et al., 2010]. García-crespo, Á., Rodríguez-González, A., Mencke, M., Gómez-berbís, J. M., & Colomo-palacios, R. (2010). Expert Systems with Applications ODDIN : Ontology-driven differential diagnosis based on logical inference and probabilistic refinements. *Expert Systems With Applications*, 37(3), 2621–2628.
- [Guida y Tasso, 1994]. G. Guida and C. Tasso. *Design and Development of KnowledgeBased Systems. From Life Cycle to Methodology*. John Wiley and Sons Ltd., Baffins Lane, Chichester, England, 1994

- [Göker et al., 2006]. M. H. Göker, R. J. Howlett, y J. E. Price, 2006. *Case-based reasoning for diagnosis applications*. The Knowledge Engineering Review, 20(03):277–281.
- [Göker et al., 1998]. M. H. Göker, T. Roth-Berghofer, R. Bergmann, T. Pantleon, R. Traphöner, S. Wess, y W. Wilke, 1998. *The development of homer: A case-based cad/cam help-desk support tool*. En Smyth y Cunningham (1998), páginas 346–357.
- [Grosso W. E., et. Al., 1999]. W. E. Grosso, H. Eriksson, R. W. Fergerson, J. H. Gennari, S. W. Tu, & M. A. Musen. Knowledge Modeling at the Millennium (The Design and Evolution of Protege-2000). In Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management (KAW99), Banff, Alberta, Canada, October 16-21, 1999
- [Gruber, T. R., 1993]. A Translation Approach to Portable Ontology Specifications. Knowledge Creation Diffusion Utilization, 5(2), 199–220.
- [Haav y Lubi, 2001]. Haav, H., & Lubi, T. (2001). A Survey of Concept-based Information Retrieval Tools on the Web. In H. Haav & T. Lubi (Eds.), 5th East-European Conference, ADBIS 2001 (pp. 29–41). Vilnius: Vilnius “Technika.”
- [Hammond, 1989] Hammond, K., “CHEF: a model of case-based planning”, proceedings of the Fourth National Conference on Artificial Intelligence, (1986).
- [Heider, 1996]. R. Heider, 1996. *Troubleshooting cfm 56-3 engines for the boeing 737 - using cbr and data-mining*. En Smith y Faltings (1996), páginas 512–518.
- [Ibrahim et al., 2001]. Ibrahim, F., J.B. Ali, A.F. Jaais and M.N. Taib, 2001. Expert system for early diagnosis of eye diseases infecting the Malaysian population. Proceedings of the IEEE Region 10th International Conference Electrical Electronic Technology, Aug. 19-22, IEEE Explore Press, Kuala Lumpur, pp: 430-432. DOI: 10.1109/TENCON.2001.949629
- [Ikeda y Mizoguchi, 1998], Mizoguchi, R., & Ikeda, M. (1998). Towards Ontology Engineering. In Technical Report (pp. 1–10).
- [INEI-CVP, 2014]. INEI (2014). Condiciones de vida en el Perú. INEI. Recuperado el 06.03.2015 de http://www.inei.gov.pe/srienaho/Consulta_por_Documentos.asp
- [Juárez+ 2005] Juárez J. y Palma J., “Inteligencia Artificial”: Razonamiento Basado en Casos, Universidad de Murcia, Vol. 74, (2005), Murcia-España.
- [Kassel, 2012]. Kassel, G. OntoSpec : une méthode de spécification semi-informelle d’ontologies. In *Actes des journées francophones d’ Ingénierie des onnaissances (IC’2002)*, (pp. 75–87). Paris: IC’2002.
- [Kolodner, J., 1983] Maintaining organization in a dynamic long-term memory. Cognitive Science: A Multidisciplinary Journal, 7(4):243 280, 1983.

- [Kolodner, J., 1993] J. Kolodner, "Case-Based Reasoning", Morgan Kaufmann, San Mateo, (1993).
- [Koton, P. 1989] Using experience in learning and problema solving. PhD Thesis, Massachusetts Institute of Technology , Laboratory of Computer Science, 1989.
- [Koton,1988]. P. Koton, Reasoning about evidence in causal explanations. In: J. Kolodner (ed.): First Workshop on CBR. ISBN: 0-934613-93-1, Morgan Kaufmann, San Mateo, 1988, pp. 260-270
- [L. Moing, 2009]. Le Moing, C. Mémoire de thèse d'exercice : *Système informatisé d'aide au diagnostic en urgences odontologiques*. Université de Rennes 1. Retrieved from www.sudoc.fr/135996198
- [Landis, 1977]. Landis JR, Koch GG. *The measurement of observer agreement for categorical data*. Biometrics 1977; 33: 159-174
- [Leake, D.B., y Wilson, D., 1998] Remembering why to remember: Performance-guided case-base maintenance. In Lecture Notes in Computer Science, pages 83-99, 2000.
- [Lee, 2000]. Lee, M.H., 2000. Model-based reasoning: A principled approach for software engineering. *Software-Concepts Tools*, 19: 179-189.
- [Lezcano, M., 1998]. Algunas Experiencias en la Utilización de Sistemas de EAC para la Enseñanza de la Inteligencia Artificial. Universidad SantaClara-Cuba. Recuperado de <http://www.emis.de/journals/DM/v62/art8.pdf>
- [Liao,2004] Liao, S.H., 2004. Expert system methodologies and applications-a decade review from 1995-2004. *Expert Syst. Appli.*, 28: 93-103.
- [Lin, R., 2009]. R. H. Lin, *An intelligent model for liver disease diagnosis*, Artificial Intelligence in Medicine, vol. 47, pp. 53-62, Sept. 2009.
- [Long, W. J. (2001)]. *Medical informatics: Reasoning methods. Artificial Intelligence in Medicine*, 23(1), 71-87. doi: 10.1016/S0933-3657(01)00076-8
- [López y Plaza, 1997]. B. López y E. Plaza, 1997. *Case-based learning of plans and goal states in medical diagnosis*. Artificial Intelligence in Medicine, 9(1):29-60.
- [Lyn et. al., 2003] Lin, Y.T., S.S. Tseng and C.F. Tsai, 2003. Design and implementation of new object-oriented rule base management system. *Expert Syst. Appli.*, 25: 369-385. DOI: 10.1016/S0957-4174(03)00064-2
- [MacRad, 1995]. R. and K. Macura, MacRad: Radiology image resource with a case-based retrieval system. In: M. Veloso and A. Aamodt (eds.): Proc of 1st Int Conference on CBR, Springer, Berlin, 1995, pp. 43-54

- [McGuinness, et al 2004] McGuinness, D. L., & Harmelen, F. van. (2004). OWL Web Ontology Language Overview. *W3C Working Group Note*. Retrieved from <http://www.w3.org/TR/owl-features/>
- [Maedche y Staab, 2001]. Maedche, A., & Staab, S. (2001). Ontology Learning for the semantic web. *IEEE Intelligent Systems*, 16(2), 72–79.
- [Simpson, 1985]. R. Simpson, 1985. *A Computer Model of Case-based Reasoning in Problem-solving: An Investigation in the Domain of Dispute Mediation*. Tesis Doctoral, School of Information and Computer Science, Georgia Institute of Technology. Georgia Institute of Technology, Technical Report GIT-ICS-85/18.
- [MedlinePlus, 2015]. Biblioteca Nacional de los EE.UU. (2015). Malaria y Leishmaniasis.
- [MINSA, 2014]. Recuperado de <http://www.minsa.gob.pe/portada/Especiales/2014/dms/situacion.html>
- [Mobley et al., 2000]. Mobley, B.A., E. Schechter, W.E. Moore, P.A. McKee and J.E. Eichner, 2000. Predictions of coronary artery stenosis by artificial neural network. *Artif. Int. Med.*, 18: 187-203.
- [Mobley et al., 2005]. Mobley, B.A., E. Schechter, W.E. Moore, P.A. McKee and J.E. Eichner, 2005. Neural network predictions of significant coronary artery stenosis in men. *Artif. Intell. Med.*, 34: 151-161. DOI: 10.1016/j.artmed.2004.08.003
- [Moorkamp, 2005]. Moorkamp, M., 2005. *Genetic Algorithms: A step by step tutorial*. Dublin Institute for Advanced Studies, Barcelona.
- [Musen, 1992]. Musen, M. A. (1992). Dimensions of knowledge sharing and reuse. *Computers and Biomedical Research*, 25(5), 435–467.
- [Neches et al., 1991]: Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., & Swartout, W. R. (1991). Enabling Technology for Knowledge Sharing. *Artificial Intelligence Magazine*, 12(3), 36–56.
- [Newel, 1982]. A. Newel, 1982. *The knowledge level*. *Artificial Intelligence*, 18:87–127.
- [Nilson M., Sollenborn, M., 2004]. Advancements and trends in medical case-based reasoning: An overview of systems and system development. *Proceedings of the 17th International FLAIRS Conference, Special Track on Case-Based Reasoning*, pages 178-183, 2004.
- [Nonaka y Takeuchi, 1999]. Ikujiro Nonaka, Hirotaka Takeuchi. *La organización creadora de conocimiento*. Oxford University Press. México, 1999
- [Pandey y Mishra, 2009]. Pandey, B. and R.B. Mishra, 2009. Knowledge and intelligent computing system in medicine. *Comput. Biol. Med.*, 39: 215 -230. PMID: 19201398

- [Papik et al., 1998]. Papik, K., B. Molnar, R. Schaefer, Z. Dombovari and Z. Tulassay et al., 1998. Application of neural networks in medicine-a review. *Diagnostics Med. Technol.*, 4: 538-546.
- [Patel et al., 2000]. Patel, V., Kushniruk, A., Yang, S., & Yale, J. (2000). Impact of a computer based patient record system on data collection, knowledge organisation, and reasoning. *Journal of American Medical Informatics Association*, 7(6), 569-85.
- [Parisaca V. Abigail, et al., 2003]. Parisaca V. Abigail, Mamani S., Liliana, Mercedes M. Miriam, Tapia T. Silvia Reducción de Bases de Casos Clusterizadas. U.N. San Agustín. (2003).
- [Peña, A. 2006]. Peña Ayala, Alejandro. *Sistemas Basados en Conocimiento: Una Base para su Concepción y Desarrollo*. Instituto Politécnico Nacional. ISBN: 970-94797-4-I. México (2006).
- [Periklis, A. et al., 2011]. Andritsos, P, A., Jurisica, I., Glasgow, J. (2011). Case-Based for Biomedical Informatic and Medicine. *Machine Learning Methods for the Analysis, Modeling and Knowledge Discovery from Bioinformatics Data*. 207-208. Recuperado de <http://www.cs.toronto.edu/~periklis/pubs/cbr14.pdf>
- [Podgorelec et al., 1999]. Podgorelec, V., P. Kokol and J. Završnik, 1999. Medical diagnosis prediction using genetic programming. *Proceedings, of the International Conference 12th IEEE Symposium on Computer- Based Medical Systems*, Jun. 18-20, IEEE Explore Press, Stamford, CT , USA., pp: 202-207. DOI: 10.1109/CBMS.1999.781271
- [Popper, K., 2009]. Popper, Karl. *Objective knowledge: an evolutionary approach*. Oxford: Oxford University Press.
- [Porte, B. y Bareiss, E., 1986]. *Protos: An experiment in knowledge acquisition for heuristic classification tasks*. Technical report. University of Texas at Austin. 1986.
- [Protégé, 2009] Protege, Ontology Editor and Knowledge Acquisition System, <http://protege.stanford.edu/>, (2009).
- [Recio-García, 2008]. Juan Recio-García. *jCOLIBRI: A multi-level platform for building and generating CBR systems*. PhD thesis, 2008.
- [Richter, 2005]. Richter, M.M.: The knowledge contained in similarity measures. Invited Talk at the First International Conference on Case-Based Reasoning, ICCBR'95, Sesimbra, Portugal (2005)
- [Rinkus et al., 2004]. Rinkus, S., Walji, M., Johnson-Throop, K., Malin, J., Turley, J., & Smith, J. (2004). Human-centered design of a distributed knowledge management system. *Journal of Biomedical Informatics*.

- [Rosse, C., & Mejino, J. L. V., 2003]. A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *Journal of biomedical informatics*, 36(6), 478–500.
- [Russell, Stuart., 1999]. Stuart Russell. (1999). *Inteligencia Artificial*. Edit.Pearson. España
- [Sφrmo, 2009]. Real-time drilling performance improvement. *Scandinavian Oil & Gas*, 2009.
- [Santos, M., 2000]. Santos M. *La naturaleza del espacio. Técnica y tiempo. Razón y emoción*. Barcelona: Editorial Ariel Geografía, S.A.; 2000.
- [Schank, R. y Abelson, R. P., 1977]. *Scripts, Plans, Goals and understanding: An inquiry into human knowledge structures*. Lawrence Erlbaum Associates, 1977
- [Schank, R., 1983]. *Dynamic Memory*. Cambridge Univ. Press, 1983.
- [Schmidt and Gierl 2001]. Schmidt, R. and L. Gierl, 2001. Case-bases reasoning for medical knowledge-based systems. *Int. J. Med. Inform.*, 64: 355-367.
- [Schreiber, A., 2000]. *Knowledge Engineering and Management. The CommonKADS*
- [Shiu, 2004] Shiu, S., *Foundations of soft case-based reasoning*. Willey Series on Intelligent Systems, Hoboken. New Jersey (2004)
- [Smyth, B. 1998]. Case-base maintenance. . In *Lecture Notes in Computer Science*, pages 507 516, 19980.
- [Stenzhorn et al., 2010]. Stenzhorn, H., Gabriele, W., Mathias, B., Fatima, S., Vangelis, K., Manolis, T., Norbert, G. (2010). The ObTiMA system - ontology-based managing of clinical trials. *Studies in Health Technology and Informatics*, 160(2), 1090 – 1094.
- [Studer, Benjamins, y Fensel, 1998]. Studer, R., Benjamins, R., & Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1-2), 161–197.
- [Sycara, 1987]. K. Sycara, 1987. *Resolving Adversarial Conflicts: An Approach Integrating Casebased and Analytic Methods*. Tesis Doctoral, School of Information and Computer Science, Georgia Institute of Technology. Georgia Institute of Technology, Technical Report GIT-ICS-87/26.
- [Tulving, 1972; Smith et al., 1978]. E. Tulving, 1972. *Episodic and semantic memory*. *Organisation of Memory*, páginas 381–403. From: Tulving and Donaldson, *Organization of Memory*, chap 10.

- [Uckun et al., 1993]. Uckun, S., B.M. Dawant and D.P. Lindstrom, 1993. Model-based diagnosis in intensive care monitoring: The YAQ approach. *Artif. Int. Med.*, 5: 31-48. DOI: 10.1016/0933-3657(93)90004-M
- [Uschold, M. y King, M., 1995]. Uschold, M., & King, M. Towards a methodology for building ontologies. In M. Uschold & M. King (Eds.), *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing*, IJ AI' 95 (pp. 1–15). Edinburgh, UK: Uschold, Mike King, Martin.
- [Vinterbo y Ohno-Machado, 2000] Vinterbo, S. and L. Ohno-Machado, 2000. A genetic algorithm approach to multi-disorder diagnosis. *Artif. Intell. Med.*, 18: 117-132.
- [Vitez et al., 1996]. Vitez, T.S., R. Wada and A. Macario, 1996. Fuzzy logic: Theory and medical applications. *J. Cardiothoracic Vasc. Anesth.*, 10: 800-806. PMID: 8910164
- [Watson y Marir, 1994]. Watson y F. Marir, 1994. . The Knowledge Engineering Review, 9(4):355–381.
- [W.E. Grosso, et al., 1999]. W. E. Grosso, H. Eriksson, R. W. Ferguson, J. H. Gennari, S. W. Tu, & M. A. Musen. Knowledge Modeling at the Millennium (The Design and Evolution of Protege-2000). In Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management (KAW99), Banff, Alberta, Canada, October 16-21, 1999.
- [Weber et al., 2006]. B. Weber, W. Wild, y R. Breu, 2006. Cbrflow: Enabling adaptive workflow management through conversational case-based reasoning. En Funk y González- Calero (2004), páginas 434–448.
- [Weis, S., 1974]. Weiss, S. M., “A system for model-based computer-aided diagnosis and therapy”, Rutgers University. The State University of New Jersey, Ph. D.
- [Weis, S., Kulikowski, C; 1974]. Weiss, S. M., Kulikowski, C. A., Amarel, S., Safir, A., "A Model-Based Method for Computer-Aided Medical Decision-Making", *Artificial Intelligence* 11, 1978, pp. 145-172.
- [Weis, S., Kulikowski, C; 1978]. Weiss, S., Kulikowski, C. A., Safir, A., "Glaucoma Consultation by Computer", *Comput Biol Med*, 1978, Vol. 8, pp. 25-40.
- [Welty & Guarino, 2001]. Welty, C., & Guarino, N. (2001). Supporting ontological analysis of taxonomic relationships. *Data & Knowledge Engineering*, 39(1), 51–74.
- [Wilson et al., 2006]. Wilson, D., D. O’Sullivan, E. McLoughlin and M. Bertolotto, 2006. Case-based decision support for intelligent patient knowledge management. *Proceeding of the 3rd International IEEE Conference Intelligent Systems*, Sep. 4-6, IEEE Explore Express, London, pp: 130-135. DOI: 10.1109/IS.2006.348406