



cenidet

**Centro Nacional de Investigación y Desarrollo Tecnológico
Departamento de Ciencias Computacionales**

TESIS DOCTORAL

**Ontología Compartida de Apoyo en la Comunicación entre
Agentes de Negociación que Usan Lenguajes Heterogéneos**

presentada por

Maricela Claudia Bravo Contreras

M. en C. en Ciencias de la Computación por el CENIDET

como requisito para la obtención del grado de:
Doctor en Ciencias en Ciencias de la Computación

Director de tesis:

Dr. Joaquín Pérez Ortega

Co-Director de tesis:

Dr. Rodolfo A. Pazos Rangel

Tabla de contenido

	Pág.
1. Introducción	1
1.1 Motivación	2
1.2 Descripción del problema de investigación	3
1.3 Objetivo de la tesis	4
1.4 Hipótesis	4
1.5 Contexto de la investigación	5
1.6 Alcances de la investigación	6
1.7 Justificación	7
1.8 Aportación	8
1.9 Organización del documento	8
2. Marco teórico	9
2.1 Negociación electrónica	10
2.1.1 Revisión de los trabajos relacionados	11
2.2 La comunicación en los sistemas de negociación	18
2.2.1 Lenguaje de comunicación entre agentes	18
2.2.2 Semántica en el lenguaje de comunicación	20
2.2.3 Revisión de los trabajos relacionados	22
3. Metodología de solución	27
3.1 Desarrollo de la ontología	28
3.1.1 Definición de ontología	28
3.1.2 Metodologías para el desarrollo de ontologías	30
3.2 Metodología para la integración de múltiples agentes de negociación heterogéneos	38
4. Arquitectura del sistema de negociación	41
4.1 Sistema de negociación electrónica	42
4.2 Sistema de traducción entre agentes	45
4.3 Plataforma de servicios Web	48

	Pág.
5. Evaluación de la metodología de solución	51
5.1 Escenario para la experimentación	52
5.2 Selección de los casos para la experimentación	52
5.3 Experimento 1: heterogeneidad alta	57
5.4 Experimento 2: heterogeneidad media	62
5.5 Experimento 3: heterogeneidad baja	68
5.6 Discusión de los resultados	73
6. Conclusiones y trabajos futuros	75
6.1 Conclusiones	75
6.2 Trabajos futuros	78
Referencias	80
Anexo A. Lenguajes de los agentes	86
Anexo B. Resultados de la investigación	93

Lista de tablas

	Pág.
Tabla 1. Significado de los elementos del mensaje	6
Tabla 2. Comparación entre los sistemas de negociación	17
Tabla 3. Términos de la ontología	33
Tabla 4. Resumen de las primitivas de negociación	53
Tabla 5. Conjunto de posibles combinaciones entre agentes heterogéneos	55
Tabla 6. Resultados del experimento 1	60
Tabla 7. Resultados del experimento 2	66
Tabla 8. Resultados del experimento 3	71
Tabla 9. Características de las pruebas experimentales	74

Lista de figuras

	Pág.
Fig. 1. Ejemplo de las relaciones entre primitivas	4
Fig. 2. Categorías de investigación de la negociación	5
Fig. 3. Diagrama de la taxonomía de conceptos	35
Fig. 4. Implementación de la ontología con OWL	37
Fig. 5. Arquitectura general del sistema	42
Fig. 6. Interacción entre los agentes	43
Fig. 7. Tratamiento de mensajes de los agentes	45
Fig. 8. Traducción entre lenguajes diferentes	46
Fig. 9. Comparación entre arquitecturas de traducción	47
Fig. 10. Funcionalidad del sistema de traducción	48
Fig. 11. Tecnologías de servicios Web	50
Fig. 12. Escenario de pruebas	52
Fig. 13. Matriz de casos de negociación	54
Fig. 14. Diagrama de estados finitos, agentes A y B	59
Fig. 15. Gráfica de resultados del experimento 1	61
Fig. 16. Diagrama de estados finitos, agentes G y E	65
Fig. 17. Gráfica de resultados del experimento 2	67
Fig. 18. Diagrama de estados finitos, agente B	70
Fig. 19. Gráfica de resultados del experimento 3	72

Dedicatorias

Dedico esta tesis a José, el hombre con el que comparto mi vida y que siempre me ha apoyado en todo lo que he emprendido.

A mi hijo José Enrique, porque siempre me ha dado la fuerza y la motivación para luchar y salir adelante.

A mi futuro bebé por la ilusión y motivación que me proporciona.

A mis padres Carlos y Maricela por las enseñanzas que me dieron y los valores de lucha y persistencia que me han ayudado tanto en los momentos más difíciles y más importantes.

A mis hermanos Carlos y Gabriela por su solidaridad y todos los buenos momentos que hemos pasado juntos.

A mis sobrinas Brenda y Jennifer por el cariño y afecto que me inspiran.

A la familia Contreras, especialmente a todas mis tías que siempre me han acompañado con sus oraciones.

Agradecimientos

A Dios por haberme permitido llegar a este momento tan importante en mi vida.

Al Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET), por la oportunidad que me brindaron para desarrollar mis estudios de postgrado.

A todas las instituciones que me brindaron su apoyo para la realización de este trabajo doctoral.

A todo el personal administrativo del CENIDET por su valiosa ayuda en todos los trámites que realicé para el buen término de este trabajo.

Al los directores de esta tesis Dr. Joaquín Pérez y Dr. Rodolfo A. Pazos, por el tiempo que dedicaron para la realización de este trabajo.

Especialmente agradezco al Dr. Víctor Sosa por su amistad, apoyo y confianza en mi trabajo.

Agradezco la amistad de la Dra. Azucena Montes, quien en los momentos difíciles me motivó y me dio palabras de ánimo.

Al Dr. Máximo López por su apoyo y la confianza que ha depositado en mí.

A la Dra. Laura Cruz y Dr. Juan Frausto por el tiempo dedicado en la revisión y las observaciones que mejoraron este documento.

Capítulo 1

Introducción

En los últimos años muchas compañías se han visto en la necesidad de reorganizar sus negocios y actualizar sus sistemas utilizando mejores tecnologías de información, con el objetivo de ser más competitivas. El uso de estas tecnologías ha acelerado el proceso de desarrollo de sistemas de comercio electrónico, tales como tiendas virtuales y mercados electrónicos. El objetivo de estos sistemas es reemplazar las transacciones de negocios físicas con las transacciones de negocios electrónicas, utilizando la Internet como medio de comunicación.

Uno de los procesos más relevantes que tienen lugar en los mercados electrónicos es la negociación. La negociación consiste en el intercambio de mensajes, y la búsqueda de nuevas alternativas, para lograr un acuerdo mutuamente beneficioso. En los mercados electrónicos este proceso se realiza mediante programas llamados agentes. A pesar de los avances en los sistemas de negociación basados en agentes, no se ha logrado alcanzar el nivel de automatización deseado. Esto es debido, entre otros factores, a que los sistemas de negociación desarrollados hasta la fecha carecen de mecanismos de comunicación flexibles que les permitan participar en plataformas abiertas como la Internet.

1.1 Motivación

La negociación juega un papel fundamental en las actividades de comercio electrónico, permitiendo que los participantes interactúen y tomen decisiones para beneficio mutuo. Los sistemas de negociación tradicionales se han implementado en plataformas de agentes múltiples [1]. Estos sistemas se han desarrollado en ambientes controlados donde el número de agentes involucrados es limitado, la plataforma de desarrollo es homogénea y el lenguaje de comunicación es acordado previamente.

Recientemente, se ha incrementado el interés en implementar y ejecutar negociaciones de comercio electrónico en plataformas abiertas como la Internet [2, 3], y en construir comunidades de agentes a gran escala, basadas en las tecnologías emergentes de los servicios Web. El reto de integrar y desplegar múltiples agentes de negociación en entornos dinámicos y abiertos, es lograr comunicaciones efectivas entre los agentes y resolver el problema inherente de heterogeneidad en el lenguaje.

Tradicionalmente los agentes de negociación se comunican mediante el intercambio de mensajes, estos mensajes se derivan del uso de algún Lenguaje de Comunicación entre Agentes (ACL, por sus siglas en inglés), ya sea un estándar o un ACL particular. El objetivo de utilizar un ACL estándar es lograr la comunicación efectiva entre agentes, sin problemas de entendimiento. Sin embargo, este enfoque de solución no es flexible, ya que restringe la integración de agentes de negociación que manejan lenguajes de comunicación con diferencias en las primitivas.

1.2 Descripción del problema de investigación

El problema de comunicación entre agentes de negociación con lenguajes heterogéneos es un problema abierto y complejo, del cual en este trabajo se aborda una parte. De manera específica se aborda el problema de encontrar las relaciones de equivalencia entre las primitivas de comunicación de agentes de negociación con lenguajes heterogéneos.

Este problema se describe de la siguiente manera:

Dados dos o más lenguajes de comunicación heterogéneos con sus respectivas primitivas de negociación, determinar las relaciones de equivalencia entre dichas primitivas, de tal forma que se posibilite la traducción de las primitivas durante la ejecución de procesos de negociación electrónica en ambientes abiertos basados en Internet.

En la Figura 1 se muestran un ejemplo de las posibles relaciones entre las primitivas de comunicación de los agentes X y Y.

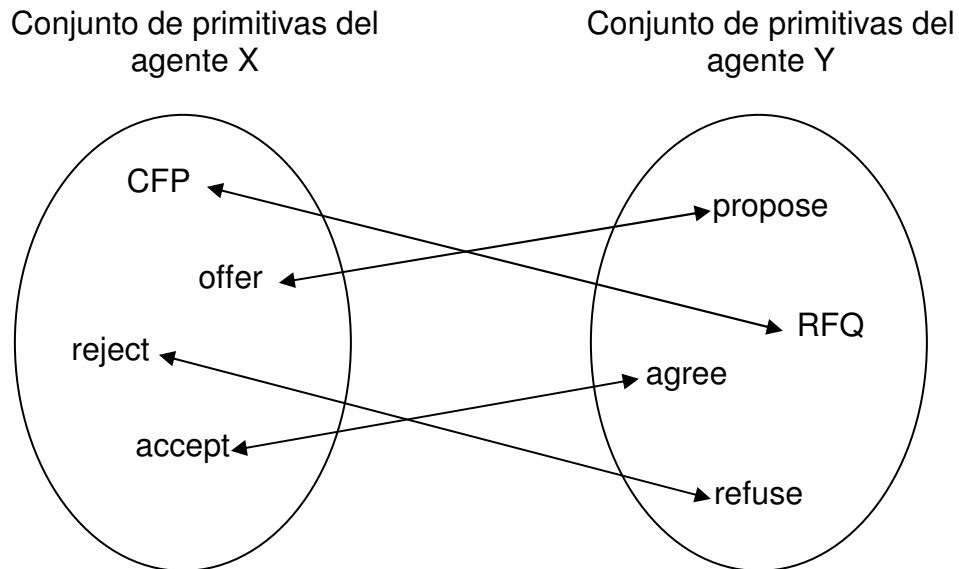


Figura 1. Ejemplo de relaciones de equivalencia entre primitivas

1.3 Objetivo de la tesis

El objetivo de esta tesis es contribuir en la comunicación efectiva entre los agentes de negociación que manejan lenguajes de comunicación heterogéneos.

El objetivo sustantivo de este trabajo es encontrar una solución para definir las relaciones de equivalencia entre las primitivas de un conjunto de agentes de negociación.

1.4 Hipótesis

Es posible que un conjunto de agentes de negociación electrónica que manejan diferentes implementaciones de lenguajes de comunicación compatibles con la especificación FIPA ACL, se comuniquen sin que se reprogramen sus lenguajes locales, mediante el uso de una ontología compartida.

1.5 Contexto de la investigación

Jürgen Müller en [4] presenta una serie de categorías para organizar los distintos enfoques de investigación sobre la negociación. En la Figura 2 se presentan la categoría del lenguaje en la negociación, la categoría de la toma de decisiones y la categoría del proceso de la negociación. Tomando como referencia la clasificación propuesta por Müller, nuestro trabajo de investigación se encuentra ubicado dentro de la categoría del lenguaje. Una descripción más detallada de esta categoría se encuentra en la Sección 2. En particular, centramos nuestro enfoque de estudio en las primitivas del lenguaje de negociación.

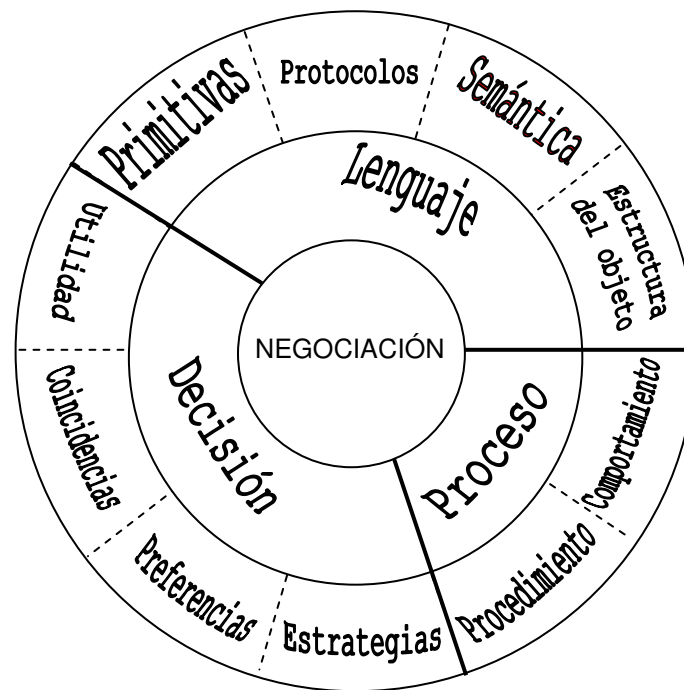


Figura 2. Categorías de investigación de la negociación

La comunicación en los sistemas de agentes múltiples se realiza a través del intercambio de mensajes. Con base en el trabajo de

Müller, se define la sintaxis de un lenguaje para construir un mensaje de negociación con los siguientes elementos.

<message> ⇒ <<sender><receiver><primitive><parameters>>

<sender> ⇒ <agent-name>

<receiver> ⇒ <agent-name>

<primitive> ⇒ <request>|<propose>|<modify>|<accept>|<reject>

<parameters> ⇒ <numbered-list-of-items>

El significado de cada uno de estos elementos se describe en la tabla 1.

Tabla 1. Significado de los elementos del mensaje

<message>	Es el mensaje que se intercambia entre los participantes de la negociación.
<sender>	Identifica al agente que remite el mensaje.
<receiver>	Identifica al agente destinatario del mensaje.
<primitive>	Representa al conjunto de primitivas de negociación que pueden utilizar los agentes para comunicar la intención o deseo de alguna acción.
<parameters>	Representa al conjunto de parámetros que acompañan a cada primitiva.

1.6 Alcances de la investigación

El problema de heterogeneidad en el lenguaje de comunicación que se aborda en esta tesis, se centra específicamente en el uso de diferentes primitivas por parte de los participantes de la negociación. No se consideran los aspectos de heterogeneidad de los otros elementos del lenguaje de comunicación. No se considera la heterogeneidad en el protocolo de comunicación.

1.7 Justificación

Aunque se han desarrollado varios sistemas de negociación electrónica basados en Internet, aún quedan por resolver aspectos muy importantes que dificultan la automatización del proceso de negociación. La investigación y desarrollo que se realice con este fin debe considerar los factores que se sabe que afectan de manera importante el proceso de automatización: plataformas de desarrollo basadas en estándares de interoperabilidad, lenguajes enriquecidos semánticamente y su integración en plataformas abiertas basadas en Internet.

En la literatura especializada se mencionan trabajos que incorporan algunos de los factores anteriores aplicados a distintas áreas, como lo son los servicios Web, las ontologías y la Web semántica. Exploran el uso y desarrollo de tecnologías que faciliten que los servicios Web descubran a otros servicios o agentes e interactúen con ellos de forma autónoma. Desarrollan tecnología y estándares orientados a favorecer la interoperabilidad. Diseñan y desarrollan ontologías de múltiples usos basadas en lenguajes de marcas para describir las propiedades y capacidades de sus servicios sin ambigüedad y de forma que puedan ser interpretados por las máquinas.

Dado que los resultados obtenidos son alentadores, consideramos que es factible utilizar lenguajes enriquecidos semánticamente, y plataformas de desarrollo basadas en estándares para facilitar la automatización de la negociación electrónica y su integración en mercados abiertos. No tenemos conocimiento de un estudio que considere todos los aspectos aplicados al proceso de negociación electrónica.

1.8 Aportación

La principal contribución de este trabajo de investigación consiste en el desarrollo de una ontología, dentro de la cual se publican y se describen las primitivas de negociación utilizadas por los agentes para comunicarse. La incorporación de esta ontología dentro del sistema de negociación electrónica resuelve el problema de comunicación entre múltiples agentes heterogéneos, y permite que los agentes continúen negociando al reducir los casos de terminación del proceso por falta de entendimiento. El conjunto de pruebas que se realizaron muestran evidencia experimental que permite observar los beneficios del enfoque de solución.

1.9 Organización del documento

El resto del documento está organizado de la siguiente manera: en la sección dos se presentan los conceptos relacionados con el tema, y la revisión del estado del arte, en la sección tres se plantea la metodología de solución al problema, en la sección cuatro se describe la arquitectura del sistema de negociación, en la sección cinco se presenta la evaluación de la metodología, y en la sección seis se presentan las conclusiones de esta tesis doctoral y los trabajos a futuro.

Capítulo 2

Marco teórico

En esta sección se presenta el estudio de los fundamentos teóricos relacionados con este tema de esta investigación doctoral. Primero se presentan varias definiciones de negociación electrónica y una revisión de los sistemas de negociación que han sido desarrollados por otros autores. Luego se profundiza en el tema de la comunicación en los sistemas de negociación, que es el tema central de esta tesis, y una revisión de los trabajos relacionados.

2.1 Negociación electrónica

La negociación es uno de los temas más frecuentemente abordados en el campo de la Inteligencia Artificial Distribuida (DAI, por sus siglas en inglés). La DAI es un campo de la inteligencia artificial, la cual estudia sociedades de agentes que interactúan para resolver un problema común. Estas sociedades de agentes se denominan sistemas de agentes múltiples (MAS, por sus siglas en inglés), los cuales trabajan en un ambiente de red para resolver problemas que se encuentran más allá de sus posibilidades individuales.

Algunas definiciones de negociación que se han presentado en la literatura son las siguientes.

Davis y Smith en [5, 6] presentaron el concepto de negociación como “una discusión en la cual las partes interesadas intercambian información y llegan a un acuerdo”. Ellos establecen que la negociación tiene tres elementos principales: a) existe un intercambio de información en dos sentidos, b) cada parte en la negociación evalúa la información desde su propia perspectiva, y c) el acuerdo final es logrado mediante selección mutua.

Desarrollaron el protocolo de contrato de red (CNP, por sus siglas en inglés) para el intercambio de información, el cual funciona esencialmente de la siguiente forma:

1. Un agente presenta una oferta,
2. Otros agentes responden mediante propuestas,
3. La oferta es otorgada al mejor agente postor, y las otras propuestas son rechazadas.

La definición de Pruitts [7] está basada en consideraciones socio psicológicas. Pruitts define la negociación como “el proceso mediante el cual se toma una decisión conjunta entre dos o más partes. Las partes primero verbalizan demandas contradictorias y entonces se mueven hacia un acuerdo mediante un proceso de concesión o búsqueda de nuevas alternativas”.

Bichler y Kersten [2] introdujeron el término e-negociación para describir la negociación que es realizada por medios electrónicos. Un sistema de negociación electrónica es un tipo de software que emplea tecnologías de Internet, es desplegado en la Web, y tiene una o más de las siguientes capacidades:

1. Da soporte para la toma de decisiones y concesiones.
2. Sugiere ofertas y acuerdos.
3. Estructura y organiza el proceso de negociación.
4. Proporciona información y experiencia.
5. Facilita y organiza la comunicación.
6. Ayuda a la preparación de acuerdos.
7. Proporciona acceso al conocimiento de la negociación, etc.

En resumen, podemos definir la negociación electrónica como el proceso mediante el cual dos o más agentes negociadores, intercambian mensajes, si es necesario realizan concesiones y buscan nuevas alternativas, sujetándose a un protocolo o conjunto de reglas para alcanzar un contrato o acuerdo.

2.1.1 Revisión de los trabajos relacionados

La investigación y desarrollo de sistemas de negociación computarizados inició en los años 80s. Estos primeros estudios y propuestas tenían como objetivo el permitir que los negociadores lograran un acuerdo mutuamente benéfico motivando la colaboración creativa. Estos sistemas aportaron la teoría de utilidad basada en la obtención de la preferencia del agente y las herramientas de apoyo para la toma de decisiones, los cuales integraban los valores máximos sugeridos en los casos previos de negociación. Ejemplos de estos sistemas son BATNA (Best Alternative to a Negotiated Agreement, por sus siglas en inglés) y ZOPA (Zone of Possible Agreements) [8].

Kersten y Noronha [9] desarrollaron **INSPIRE**, un sistema de soporte para la negociación entre negocios. El enfoque de su trabajo se centró en el comportamiento de los negociadores en un ambiente basado en Internet. Observaron más de 2000 casos de negociación y se basaron en los experimentos y comentarios recibidos de los usuarios. Encontraron que una gran parte de los usuarios estuvieron dispuestos a utilizar el sistema de soporte para la negociación en la práctica.

Los sistemas de negociación basados en agentes consideran negociaciones totalmente automatizadas. Este enfoque abarca los trabajos sobre los agentes de software inteligentes, los cuales conocen las preferencias, se comunican y toman decisiones en representación del dueño, de acuerdo con las estrategias programadas. Por ejemplo, Su [3] presentó un servidor de negociación basado en Internet, que se comunica tanto con seres humanos como con otros sistemas de negociación. El sistema

automáticamente toma decisiones basado en las restricciones especificadas y envía mensajes; sin embargo, la decisión final la toma el usuario.

BargainFinder es un agente de compras que ha sido utilizado en la intermediación mercantil. Este agente fue desarrollado por Andersen Consulting en 1995, con el objetivo de ser enviado a la Web para realizar búsquedas de los precios más bajos de determinado producto, especificado por su dueño. **PersonaLogic** y **Firefly** también son sistemas basados en agentes de software, capaces de realizar búsquedas de productos [10]. Sin embargo, estos últimos no son autónomos y no pueden realizar tareas independientemente.

Los mercados electrónicos utilizan arquitecturas complejas basadas en múltiples agentes inteligentes, que en representación de sus dueños, compran y venden siguiendo algún protocolo. Ejemplos de estas arquitecturas son **Kasbah** [1] y **MAGMA** [11].

Kasbah es un mercado electrónico poblado por agentes de software que compran y venden, los cuales se enlazan en una negociación. Los vendedores y los compradores les proporcionan a sus agentes los datos sobre sus preferencias de compra y venta, y después son cargados en el sistema Kasbah, donde buscan a otros agentes para realizar negociaciones.

Las subastas son uno de los mecanismos más predominantes para realizar comercio electrónico mediado por agentes [12]. **Fishmarket** [13] y **AuctionBot** [14] son ejemplos de sistemas de subasta basados en agentes. El proyecto Fishmarket emula en un entorno de comercio electrónico, una estructura de comercio que se basa en el mercado de pescado en España. El proyecto Fishmarket está

limitado al entorno que emula y no ofrece una solución de negocios completa para un mercado electrónico.

Existen sistemas comerciales para la administración de negocios, los cuales incorporan funciones básicas de soporte para la negociación. Moai [15] es considerado como un sistema de negociación electrónica que incluye una subasta y varios componentes de negociación bilateral. Perfect Commerce Inc. [16] ofrece un sistema que proporciona preferencia de orden, y métodos de optimización. Ozro Inc. [17] implementa soluciones para la negociación y manejo de contratos dentro del contexto de las relaciones con proveedores.

SmartSettle es un sistema comercial de soporte para la negociación basado en Web, el cual es una versión extendida de [18]. Utiliza técnicas analíticas de decisión para facilitar y dar soporte a la negociación. El soporte analítico del sistema SmartSettle tiene sus raíces en el análisis de la negociación y la decisión, y su objetivo es dirigir a las partes hacia un acuerdo.

Los sistemas de soporte para la negociación también han sido estudiados desde el punto de vista del modelado del proceso de la negociación. La teoría de acción del habla, la cual modela el proceso de negociación como un diálogo entre los participantes, ha sido la metodología más popular. Kumar y Feldman [19] sugirieron la utilización de máquinas de estado finito para describir formalmente varios mecanismos de subastas. Benyoucef y Keller [20] compararon metodologías para especificar el proceso de negociación: lenguaje natural, lenguaje de coordinación de agentes, máquinas de estado finito, diagramas de estado. Sugirieron que los diagramas de estado son la mejor alternativa para la descripción

formal de las reglas de negociación. En el laboratorio de investigación en Zurich de IBM, también se están implementando herramientas para el diseño y modelado de sistemas de negociación basados en máquinas de estado finito.

En [21 y 22] los autores presentaron sistemas de negociación combinados. Establecieron que en una negociación combinada, el usuario está interesado en varios productos o servicios, y por consiguiente, se enrola en muchas negociaciones al mismo tiempo. Desde el punto de vista de la administración del flujo del trabajo, desarrollaron un sistema de soporte para la negociación combinada que ayuda al usuario a realizar todas las negociaciones dependientes al mismo tiempo. Mostraron la funcionalidad del sistema y desarrollaron un prototipo llamado CONSENSUS. En la arquitectura de CONSENSUS, los usuarios diseñan el proceso de negociación combinada, y los agentes de software asignados a cada individuo en la negociación ejecutan los procesos diseñados. El prototipo fue implementado utilizando MQSeries WFMS de IBM [23] y BEA WebLogic [24].

El **Matchmaker** [43] fue desarrollado en la Carnegie Mellon con el objetivo de buscar servicios Web de acuerdo a las capacidades que éstos proporcionan. El matchmaker basa su búsqueda en seleccionar al proveedor del servicio Web, cuyas capacidades descritas coincidan más cercanamente con las capacidades del solicitante del servicio Web. Utiliza un enfoque de representación de capacidades basado en el uso de ontologías, las cuales proporcionan una representación explícita de las tareas realizadas por los servicios Web. El matchmaker no es un sistema de negociación electrónica, sino un buscador de servicios Web. Sin

embargo, un aspecto importante de este proyecto es el uso de ontologías para la representación de las capacidades.

A pesar de la variedad de estudios sobre la automatización de la negociación, éstos se han enfocado fuertemente en el aspecto de la toma de decisiones. Poco trabajo se ha hecho sobre los aspectos de comunicación en sistemas de negociación heterogéneos. En la sección 2.2.3 se presentan los trabajos relacionados con el tema.

De la revisión de los sistemas de negociación, se concluye que aún no se ha logrado el nivel de automatización deseable en este proceso. Esto es debido a que los sistemas de negociación desarrollados hasta la fecha carecen de la infraestructura computacional necesaria para que participen en plataformas abiertas como la Internet y se integren de forma automática y dinámica. En la tabla 2 se muestra un resumen comparativo de los sistemas de negociación. Los aspectos que se consideraron fueron los siguientes:

- a) Interoperabilidad. Este aspecto considera si un sistema de negociación fue construido de tal manera que permita la interoperabilidad con otros sistemas, independientemente de la plataforma de desarrollo con el que fue construido.
- b) Adaptabilidad/heterogeneidad. Se refiere a la característica de permitir que un nuevo agente de negociación que maneja un lenguaje de comunicación diferente, se pueda integrar en el sistema de negociación.
- c) Negociación bilateral y multilateral. Identifica si en el sistema se pueden realizar negociaciones de uno a uno (bilateral); o uno a muchos, muchos a uno, o muchos a muchos (multilateral).

d) Basado en agentes. Se refiere a que si el sistema de negociación se desarrolló basándose en modelos de agentes.

Tabla 2. Comparación entre los sistemas de negociación

	Universidad	Interoperabilidad	Adaptabilidad	Negociación Bilateral	Negociación Multilateral	Basado en Agentes
Inspire	Carleton University	x	x	✓	x	x
SmatSettle	Cornell University	x	x	✓	x	x
Bargain Finder	Andersen Consulting	x	x	x	x	✓
FireFly		x	x	x	x	✓
Kasbah	MIT	x	x	✓	✓	✓
MAGMA	Minnesota University	x	x	✓	✓	✓
AuctionBot	Michigan University	x	x	✓	✓	✓
FishMarket	U.A.B. Barcelona	x	x	✓	✓	x
Matchmaker	Carnegie Mellon University	✓	✓	x	x	x
Tesis	CENIDET	✓	✓	✓	✓	✓

Dado que los sistemas de negociación desarrollados hasta ahora no han sido construidos considerando la interoperabilidad, carecen de soluciones que favorezcan la comunicación durante el intercambio de mensajes entre agentes de negociación heterogéneos. Esto trae como consecuencia que no se ha logrado su total integración en mercados dinámicos y abiertos basados en Internet. Por lo tanto, se puede considerar que la automatización del proceso de

comunicación entre agentes de negociación heterogéneos es un problema abierto.

2.2 La comunicación en los sistemas de negociación

Tradicionalmente, los sistemas de comunicación que utilizan los agentes de negociación son desarrollados adhoc para un determinado conjunto de agentes y para un lenguaje específico. Una de las limitaciones de este tipo de implementaciones es que los mecanismos de comunicación no se encuentran documentados, de tal manera que resulta casi imposible para un desarrollador que no participó en el diseño, implementar una comunicación externa con este tipo de agentes.

2.2.1 Lenguaje de comunicación entre agentes (ACL)

El ACL es un lenguaje de alto nivel cuyas primitivas y estructuras están expresamente diseñadas para dar soporte a los diferentes tipos de colaboración, **negociación** y transferencia de información requerida en la interacción entre múltiples agentes. El ACL se ubica en una capa lógica sobre los protocolos de transporte tales como TCP/IP, HTTP e IIOP. Tales protocolos trabajan con aspectos de comunicación en el nivel de transporte de mensajes y datos, mientras que ACL trata la comunicación en el nivel intencional y social. El ACL es una estructura compleja compuesta de diferentes sub-lenguajes que especifican el contenido del mensaje, los parámetros de interpretación tales como el remitente,

la actitud proposicional bajo la cual el receptor debe interpretar el contenido del mensaje y muchos otros componentes.

Entre los ACLs más utilizados se encuentran KQML [25] y FIPA ACL [26], los cuales se describen a continuación:

KQML (por sus siglas en inglés, Knowledge Query and Manipulation Language) es un lenguaje desarrollado dentro del proyecto de conocimiento compartido ARPA. Contiene dos componentes principales: (1) un lenguaje de representación para el contenido de los mensajes, llamado formato de intercambio de conocimiento (KIF), el cual es una extensión de la lógica de primer orden; y (2) un lenguaje de comunicación KQML, el cual consiste de un conjunto de primitivas de comunicación con el objetivo de soportar la interacción en sistemas de agentes múltiples. KQML incluye muchas performativas de actos del habla, todas asertivas o directivas, que los agentes utilizan para asegurar hechos, solicitar información o suscribirse a servicios.

FIPA-ACL es la especificación de un lenguaje desarrollado por iniciativa de la fundación de agentes físicos inteligentes FIPA (por sus siglas en inglés, Foundation for Intelligent Physical Agents). Este lenguaje también se basa en la teoría de actos del habla, y sus mensajes también son considerados como actos comunicativos cuyo objetivo es realizar alguna acción. Un mensaje de FIPA contiene un conjunto de uno o más parámetros. El único parámetro que es obligatorio dentro de un mensaje es el *performative*; sin embargo, se espera que la mayoría de los mensajes FIPA también especifiquen los parámetros *sender*, *receiver* y *content*. FIPA especifica que si un agente no puede reconocer o no puede

procesar uno o más de los parámetros, puede responder con un mensaje *not-understood*.

El uso exitoso de ACL para comunicar a dos programas de agentes, depende del apropiado funcionamiento de toda una infraestructura de comunicaciones. Esta infraestructura involucra el orden de los mensajes, la entrega, formato y direccionamiento, servicios de directorios, traducción estilo pasarela, calidad del servicio, y otros aspectos de redes y comunicaciones. En la práctica, los sistemas de agentes implementados han utilizado tanto estrategias centralizadas que manejan todos los aspectos de mensajes entre agentes, así como sistemas descentralizados que dejan esta funcionalidad a los manejadores de comunicación de los mismos agentes. Los sistemas de agentes ofrecen muy poca estandarización en esta área.

La actual teoría semántica de FIPA-ACL depende de una teoría de sociedad de agentes que provee un conjunto de primitivas estilo BDI (por sus siglas en inglés, Beliefs Desires and Intentions). La semántica de FIPA-ACL está basada en conceptos mentales tales como creencia e intención, y trata el intercambio de mensajes entre agentes como un tipo de acción.

2.2.2 Semántica en el lenguaje de comunicación

Existen varios enfoques sobre el estudio de la semántica en el área de la lingüística y la filosofía. En ciencias computacionales frecuentemente se hacen referencias a estos enfoques para formalizar sistemas de comunicación; por ejemplo, en sistemas de agentes se utiliza el término de actos del habla para referirse al uso de un lenguaje por sus emisores y receptores. Sin embargo, la comunicación (en agentes de software se empaca en el contenido

de un mensaje) que se realiza entre dos sistemas de agentes puede representarse por una combinación de semántica explícita e implícita.

El estudio formal del lenguaje tiene tres aspectos:

- a. Sintaxis, trata con la forma cómo los símbolos son estructurados.
- b. Semántica, se refiere al significado explícito de los símbolos.
- c. Pragmática, es la forma cómo son interpretados y usados.

El significado es la combinación de semántica y pragmática.

De acuerdo con Uschold [27], la discusión sobre la semántica no se debe centrar en el hecho de que un sistema tiene semántica, sino que si la semántica está implícita o explícitamente codificada. Por ejemplo si dos sistemas, A y B, son capaces de comunicarse entre ellos, y actúan de acuerdo a cada mensaje recibido, se puede decir que tienen semántica compartida. Si esta semántica es *explícita*, entonces debe existir una especificación formal del significado de cada mensaje. Sin embargo, si tal definición explícita no existe, es decir, si los sistemas se comunican mediante el envío de códigos conocidos solamente por los diseñadores de los sistemas, entonces la semántica está *implícitamente* codificada en el software de cada sistema.

Otro aspecto relevante es si la semántica explícita del significado de los mensajes se encuentra escrita en algún lenguaje formal legible por las máquinas, el cual a su vez podría tener semántica explícita (un segundo nivel de formalidad) o implícita (a través de algún intérprete).

2.2.3 Revisión de los trabajos relacionados

Como se señaló en la sección 1.5 del Contexto de la investigación, este trabajo se centra en el tema del lenguaje de comunicación entre agentes negociadores. En esta sección se presentan los trabajos relacionados, señalando particularmente las consideraciones sobre el lenguaje de comunicación entre agentes manejado por otros autores.

En el lenguaje de comunicación de agentes (ACL) de FIPA, se proponen varias primitivas para la negociación entre agentes. Sin embargo, la implementación de los sistemas de negociación basados en las especificaciones FIPA, difieren entre unos y otros. Estas diferencias van desde el lenguaje de programación utilizado, hasta pequeñas diferencias sintácticas de las primitivas de comunicación. De hecho, la especificación FIPA de actos de comunicación, permite que se definan y se implementen nuevas primitivas de comunicación particulares, lo que ocasiona mayores diferencias entre los distintos sistemas implementados.

Müller [4] publicó un artículo sobre principios de negociación. En este trabajo se listan los nombres de varias primitivas de negociación, las cuales se dividen en tres grupos: iniciadores, reactivos y completadores. Sin embargo, la sintaxis y la semántica de estas primitivas no se describen.

En el trabajo de Rueda [28] se da especial importancia al lenguaje, estableciendo que el éxito de una aplicación desarrollada desde el modelo del agente, depende considerablemente de la habilidad del agente para comunicarse. Desde esta perspectiva, el rol del

lenguaje de comunicación es fundamental, porque permite que los agentes interactúen y compartan conocimientos. A pesar de que la comunicación tiene lugar técnicamente mediante mensajes sobre una red que utiliza un protocolo de bajo nivel, a nivel conceptual los agentes no intercambian mensajes, sino que mantienen conversaciones basadas en sus propósitos.

En el trabajo de Stanislav [29] se plantea que el proceso de negociación es el intercambio interactivo de propuestas hasta que se logra un acuerdo. Discuten los problemas para la automatización del proceso de negociación, entre los cuales plantean el de la falta de entendimiento común. Proponen una solución mediante la definición de la estructura del mensaje basada en el trabajo de Jennings y Bartolini, quienes proponen la instanciación de una plantilla de mensaje, la cual es enviada a la contraparte.

En el trabajo de Haifei [30] se explica que existen dos aspectos importantes en el proceso de negociación: la comunicación entre los participantes y la toma de decisiones. La comunicación trata los aspectos de especificación de los requerimientos y las restricciones de los productos o servicios, y cómo se establece el paso de mensajes entre los participantes. En dicho trabajo los autores definen la sintaxis y la semántica de un conjunto de primitivas de negociación. Introducen un lenguaje de especificación de restricciones, para aumentar el poder de expresividad de los objetos de negociación.

Chang y Woo [31] propusieron un protocolo de negociación llamado SANP (por sus siglas en inglés, Speech-Act-based Negotiation Protocol). Este protocolo está basado en la clasificación de actos

del habla de Ballmer y Brennenstuhl y en la literatura de análisis de la negociación.

En su trabajo sobre controversia, Crawshay-Williams [32] enfatizó la necesidad de clarificación de conceptos previo a la discusión. Asevera que si los participantes en una discusión toman acuerdos sobre los criterios bajo los cuales un enunciado será probado, entonces el acuerdo puede alcanzarse más rápido.

Naes [33] establece que los criterios para la verificación o falsificación de un enunciado son esenciales. Incluye el acuerdo sobre los términos como una etapa en la discusión, no como un requisito para iniciarla. Las tres etapas para la discusión que él sugiere son: interpretación, clarificación y argumentación. Para cualquier enunciado T existe un conjunto de posibles interpretaciones de T , y los participantes deben estar de acuerdo sobre cuál interpretación desean discutir. Naes asegura que la precisión de los enunciados ayuda a eliminar la *falta de entendimiento*, donde U es más preciso que T si cualesquiera interpretaciones de U son también interpretaciones de T , pero hay interpretaciones de T que no son interpretaciones de U . Esto es útil solamente si el desacuerdo ha ocurrido a través de diferentes interpretaciones.

Carbogim [34] presenta un estudio sobre los elementos que son manejados por sistemas de argumentación automatizada y sugiere direcciones para investigación a futuro. En este estudio la argumentación sobre el significado de los términos usados no es considerada explícitamente en el texto.

Alison [35] establece que la ambigüedad juega un rol importante dentro de una discusión. Es decir, cuestionar el significado de los términos en una discusión es una estrategia válida. Los participantes podrían no haberse dado cuenta inicialmente que tienen diferentes interpretaciones de una palabra, de hecho es posible que ni siquiera cuenten con una clara interpretación.

Bartolini, Preist y Jennings [36] estudian los diferentes protocolos que se utilizan para negociaciones automatizadas: uno a uno, uno a muchos, y muchos a muchos. Argumentan que la mayoría de los sistemas de agentes múltiples son diseñados con un solo protocolo de negociación, explícitamente codificado en todos los agentes. Lo que ocasiona un entorno inflexible, solamente capaz de aceptar agentes diseñados para él.

Ramos, Frausto y Camargo [44] presentaron una metodología basada en el uso de redes de Petri coloreadas, para modelar la interacción en sistemas de múltiples agentes.

En los trabajos revisados se abordan varios aspectos relevantes de la comunicación en sistemas de negociación, como son: el significado de los términos, el uso de lenguajes formales para la especificación semántica, la interpretación de los mensajes transmitidos, y sobre todo el problema por falta de entendimiento común. Del análisis de estos trabajos relacionados podemos concluir que existe un claro consenso acerca de la importancia de la comunicación en los sistemas de negociación. También podemos observar que en general los autores proponen como medida de solución al problema de heterogeneidad en el lenguaje de comunicación entre sistemas de negociación, el uso de una misma

sintaxis y especificación del lenguaje para todos los agentes que deseen participar, lo que ocasiona, por supuesto, el rediseño de los mecanismos de comunicación y su exacta implementación. Esta solución tiene un gran inconveniente, no es una solución flexible: ni está orientada hacia la automatización del proceso de integración e interacción dinámica entre agentes heterogéneos.

En contraste, en este trabajo de investigación se presenta una solución que favorece la interoperabilidad de manera flexible, basada en el diseño de una ontología para compartir las definiciones de las primitivas de negociación usadas en el lenguaje de comunicación entre agentes.

Capítulo 3

Metodología de solución

Para la solución de los problemas de comunicación en sistemas de negociación, varios autores proponen la adopción y el uso de una misma sintaxis y el previo acuerdo sobre las definiciones del lenguaje, para todos los agentes que deseen participar. Sin embargo, esta solución tiene un gran inconveniente, no es una solución flexible, ni favorece la automatización del proceso de interacción dinámica entre agentes heterogéneos.

En este capítulo se presenta una metodología de solución que permite que los agentes se comuniquen efectivamente en un ambiente abierto, dinámico basado en Internet, sin imponer restricciones sobre el lenguaje de comunicación y con el objetivo de reducir los casos de falta de entendimiento.

3.1 Desarrollo de la ontología

En los últimos años se ha incrementado el interés en la investigación y desarrollo de ontologías como medio para compartir y unificar definiciones provenientes de diversas fuentes de información y/o conocimientos en aplicaciones existentes. Este término fue originalmente utilizado en filosofía, donde se refería a la explicación semántica de la existencia. Más recientemente, el término ha sido utilizado en varias áreas de la inteligencia artificial y más ampliamente en ciencias de la computación.

El desarrollo de la ontología compartida que se presenta en esta sección, representa la principal aportación de este trabajo de investigación. Para mostrar este elemento fundamental de la metodología de solución, primero se presenta la definición de ontología, así como sus características y sus diferencias específicas con otras formas usadas para compartir vocabularios.

3.1.1 Definición de ontología

Una de las definiciones más utilizadas de ontología es la que presentó Gruber [37], quien define una ontología como: “Una especificación explícita de una conceptualización”. La definición de

Gruber se basa en la idea de que la formalización declarativa del dominio del conocimiento comienza desde la conceptualización del dominio, es decir, la identificación de los objetos que hipotéticamente existen en el mundo y las relaciones entre ellos.

Otra definición de ontología es la de Guarino [38], quien establece que “Una ontología es un conjunto de axiomas lógicos diseñados para representar el significado pretendido de un vocabulario”.

El uso de ontologías para abordar problemas de interoperabilidad fue presentado por Uschold [27]. Plantea que entre las aplicaciones que se pueden beneficiar del uso de ontologías, se encuentran las arquitecturas basadas en agentes heterogéneos, los cuales requieren establecer enlaces de comunicación para intercambiar datos.

Existen otras formas de clasificar y organizar la información con un enfoque semántico; entre éstas se encuentran los esquemas de bases de datos, las taxonomías, los thesaurus, etc.

a). **Esquemas de bases de datos relacionales.** De acuerdo con Fensel [39], una ontología tiene una función similar a la de los esquemas de bases de datos relacionales; sin embargo, posee diferencias importantes:

- Un lenguaje para definir ontologías tiene una sintaxis y semántica más enriquecida que los enfoques comunes de las bases de datos relacionales.
- La información que se describe en una ontología consiste de texto en lenguaje natural semiestructurado, y no información tabular.

- Una ontología debe ser una terminología compartida y consensual, porque es usada para compartir e intercambiar información.
- Una ontología representa el conocimiento de un dominio y no la estructura de un contenedor de datos.

b). **Taxonomías**. Una taxonomía es un sistema de clasificación de entidades de información en forma de una jerarquía, de acuerdo a las supuestas relaciones de las entidades en el mundo real. Las taxonomías establecen una semántica simple. Las ontologías utilizan las taxonomías como su estructura básica.

c). **Thesaurus**. Es un vocabulario controlado diseñado para dar soporte en la recuperación de la información. Un thesaurus se refiere a las relaciones entre términos (palabras o frases) estructurados en una taxonomía.

La razón por la cual se seleccionó una ontología como medio de solución al problema de heterogeneidad es por las características que posee. Una ontología está integrada por **clases** que representan los elementos principales del dominio de interés, **instancias** que representan los objetos particulares, **relaciones** entre los objetos y propiedades de los objetos. Además los lenguajes ontológicos permiten la definición de reglas de inferencia lógica. El principio de inferencia deriva nuevo conocimiento a partir del conocimiento que ya existe. Sin embargo, cabe señalar que en este trabajo no se definieron ni se usaron reglas de inferencia.

3.1.2 Metodologías para el desarrollo de ontologías

Para el desarrollo de la ontología se analizaron varias metodologías y procedimientos propuestos. Uschold y Gruninger [27] presentaron una de las primeras metodologías, la cual define una serie de pasos para el desarrollo y evaluación de ontologías. Otra metodología es METHONTOLOGY [42], desarrollada en la Universidad Politécnica de Madrid. Esta metodología permite la construcción de ontologías en el nivel de conocimiento, y tiene sus orígenes en las principales actividades identificadas por el proceso de desarrollo de software del IEEE y en otras metodologías de ingeniería del conocimiento.

Estas metodologías comienzan con la identificación del propósito de la ontología y la necesidad de adquisición del conocimiento del dominio. Sin embargo, habiendo adquirido suficiente conocimiento, la metodología de Uschold propone la codificación en un lenguaje formal; mientras que METHONTOLOGY propone expresar las ideas en un conjunto intermedio de representaciones y después usa traductores para generar la ontología. Para el desarrollo de la ontología se utilizó el enfoque de la metodología de Uschold, ya que el enfoque de METHONTOLOGY está más orientado a la interacción directa con las personas que poseen el conocimiento del dominio, y además requiere de un proceso de traducción adicional.

En lo referente a la necesidad de la evaluación de la ontología, la metodología de Uschold incluye esta actividad, pero no especifica cómo debe realizarse. Gruninger propone identificar un conjunto de preguntas de competencia. Las preguntas de competencia son la base de una caracterización rigurosa del conocimiento que una ontología debe cubrir, especifican el problema y lo que constituye

una buena solución. Una vez que la ontología ha sido expresada formalmente, se compara contra este conjunto de preguntas de competencia. METHONTOLOGY propone que la evaluación ocurra durante el desarrollo de la ontología. La mayor parte de la evaluación sucede durante la conceptualización.

Muchos sistemas de representación utilizan un enfoque de modelado basado en marcos, un enfoque basado en lógica, o ambos para formalizar la ontología. Modelan el mundo utilizando conceptos, ejemplares, relaciones, funciones y axiomas. Una ontología formalizada utilizando cualquiera de estos enfoques puede implementarse de manera directa.

De las metodologías revisadas se identificaron los pasos que fueron la guía metodológica para el desarrollo de la ontología. Estos pasos se describen a continuación.

1. Describir el propósito de la ontología.
2. Identificar los usuarios de la ontología.
3. Identificar los conceptos y relaciones clave.
4. Construir el diagrama de la taxonomía de conceptos.
5. Codificar la ontología utilizando algún lenguaje formal.
6. Integrar las ontologías existentes.
7. Evaluar la ontología.

Aplicación de la metodología para el desarrollo de la ontología.

1. Propósito de la ontología. El objetivo de implementar la ontología de negociación es servir como base para resolver el problema de

falta de entendimiento que puede ocurrir entre agentes durante el intercambio de mensajes en la negociación; por lo tanto la ontología servirá como un vocabulario compartido. Esta ontología forma parte de un sistema de negociación electrónica, y es consultada principalmente por un módulo de traducción.

2. Usuarios de la ontología. Los usuarios de la ontología son desarrolladores, administradores, aplicaciones cliente, y específicamente el módulo de traducción del sistema de negociación electrónica.
3. Identificación de conceptos y relaciones. Tomando como referencia el estudio presentado por Jürguen Müller sobre principios de negociación, los principales elementos que intervienen en la negociación electrónica son: el lenguaje en la negociación, la toma de decisiones y el proceso de la negociación. Se tomó como base esta clasificación para derivar las principales clases de la ontología, las subclases, las relaciones, los atributos y los ejemplares.

El conjunto de términos que se presenta en la tabla 3 muestra los principales términos de la ontología de negociación. Estos términos y descripciones se presentan en inglés, ya que es el lenguaje que se está utilizando para el manejo de los mensajes de negociación entre agentes.

Tabla 3. Términos de la ontología

Término	Sinónimo	Descripción	Tipo
Language	Communication	Is concerned with the communication primitives for negotiation, their semantics and their usage.	Class

Decision	Strategy	Algorithms to compare the negotiation topics and correlation functions. Negotiation strategies fall in this class.	Class
Protocols	Process	General models of the negotiation process and the global behavior of the negotiation participants.	Class
Participants	Agents	Identifies the negotiating agents which participate in the negotiation process.	Class
Primitives		The basic form of communication between agents.	Subclass
Parameters		The context transmitted together with a negotiation primitive.	Subclass
Initiator	Starter	Negotiation primitives that initiate a negotiation.	Subclass
Reactor		Negotiation primitives that react on a given statement.	Subclass
Completer	Ender	Negotiation primitives that complete a negotiation.	Subclass

Deal		Negotiation primitives that complete a negotiation with an agreement.	Subclase
No-deal		Negotiation primitives that complete a negotiation with no deal or failure.	Subclass

4. Construir el diagrama de la taxonomía de conceptos. A partir del conjunto de términos de la sección anterior se construyó el diagrama de la taxonomía de conceptos. En la Figura 3 se muestra el diagrama general de la taxonomía de conceptos de la ontología.

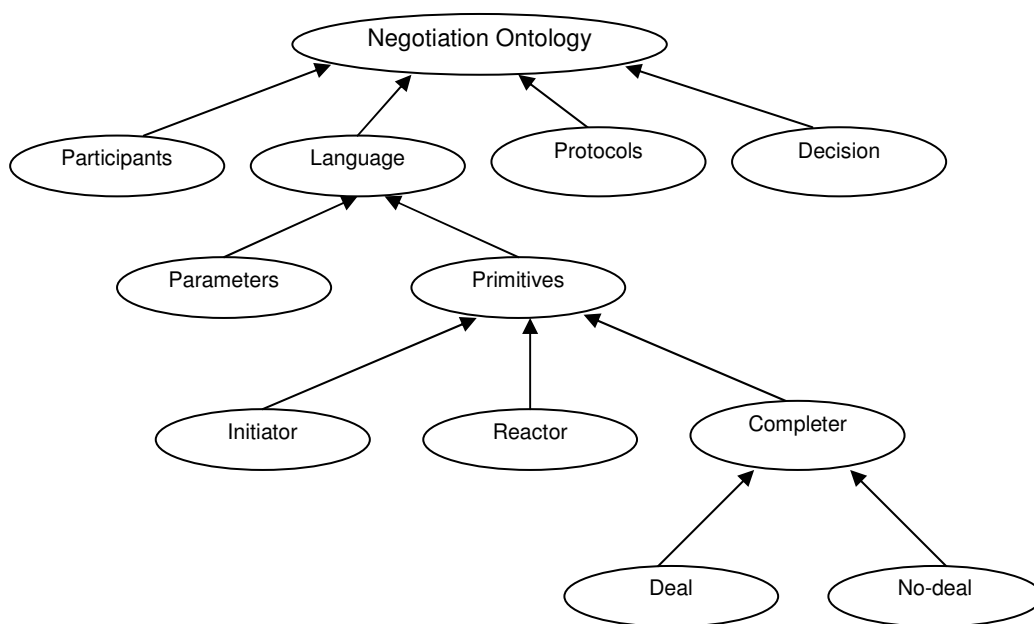


Figura 3. Diagrama de la taxonomía de conceptos

Una de las funciones elementales de la ontología es el establecimiento de las relaciones de equivalencia entre las primitivas. Estas relaciones de equivalencia son muy importantes para resolver el problema de comunicación, ya que es a través de

éstas que se encuentran las primitivas equivalentes durante el proceso de traducción.

A continuación se definen las relaciones necesarias para la traducción y las reglas para usarlas.

- a) Dos primitivas de negociación son iguales si su uso es idéntico, pero su sintaxis es diferente.
 - b) Dos primitivas de negociación son similares si no se puede asegurar que su uso es idéntico, pero se puede asumir que poseen intenciones similares.
 - c) Si dos primitivas de negociación tienen sintaxis idéntica, y su uso es el mismo, entonces no se establece relación de equivalencia entre ambas, ya que no ocasionarán problemas de comunicación.
 - d) Si dos primitivas de negociación tienen sintaxis y uso totalmente diferente, y no existe ninguna clave que permita establecer una relación, entonces se quedan sin relacionar.
5. Codificar la ontología. Para la codificación de la ontología se utilizó el lenguaje ontológico OWL, el estándar más reciente del consorcio de la World Wide Web (W3C). La ontología se implementó en Protégé [40, 41], una plataforma abierta para el modelado de ontologías y adquisición de conocimiento. Protégé ofrece un Plugin de OWL, el cual puede ser utilizado para editar ontologías OWL, para acceder a los razonadores de lógica descriptiva, y adquirir ejemplares de marcas semánticas. En la Figura 4 se muestran las clases, subclases y las relaciones que se definen para las instancias de la ontología, codificados con el lenguaje OWL.

```

<owl:Class rdf:ID="Participants">
  <rdfs:subClassOf rdf:resource="#negotiation"/>
</owl:Class>
<owl:Class rdf:ID="Language">
  <rdfs:subClassOf rdf:resource="#negotiation"/>
</owl:Class>
<owl:Class rdf:ID="Protocol">
  <rdfs:subClassOf rdf:resource="#negotiation"/>
</owl:Class>
<owl:Class rdf:ID="Decision">
  <rdfs:subClassOf rdf:resource="#negotiation"/>
</owl:Class>

<owl:Class rdf:ID="Primitives">
  <rdfs:subClassOf rdf:resource="#Language"/>
</owl:Class>
<owl:Class rdf:ID="Parameters">
  <rdfs:subClassOf rdf:resource="#Language"/>
</owl:Class>

<owl:Class rdf:ID="Initiator">
  <rdfs:subClassOf rdf:resource="#Primitives"/>
</owl:Class>
<owl:Class rdf:ID="Reactor">
  <rdfs:subClassOf rdf:resource="#Primitives"/>
</owl:Class>
<owl:Class rdf:ID="Completer">
  <rdfs:subClassOf rdf:resource="#Primitives"/>
</owl:Class>

<owl:ObjectProperty rdf:ID="isSuccessorOf">
  <owl:inverseOf rdf:resource="#hasSuccessor"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasSuccessor">
  <owl:inverseOf rdf:resource="#isSuccessorOf"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="isSynonymOf">
  <owl:inverseOf rdf:resource="#hasSynonym"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasSynonym">
  <owl:inverseOf rdf:resource="#isSynonymOf"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasSimilar">
  <owl:inverseOf rdf:resource="#isSimilarOf"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="isSimilarOf">
  <owl:inverseOf rdf:resource="#hasSimilar"/>
</owl:ObjectProperty>

```

Figura 4. Implementación de la ontología con OWL

6. Evaluación de la ontología. De acuerdo con Fernández [42], la evaluación se refiere al juicio técnico de la ontología, su entorno de software asociado y la documentación con respecto a un marco de referencia. El marco de referencia pueden ser los requerimientos, las especificaciones, o preguntas de competencia del mundo real. En este caso en particular, la ontología desarrollada fue evaluada dentro de la aplicación para corroborar si satisfacía el objetivo inicial.

Uno de los aspectos relevantes de la ontología, es que se puede considerar como una ontología flexible, ya que entre más agentes se incorporen más definiciones explícitas de las primitivas de negociación existen, más relaciones se establecen entre las primitivas y cada vez será más rápida la comunicación en comunidades de agentes múltiples.

3.2 Metodología para la integración de múltiples agentes de negociación, con lenguajes de comunicación heterogéneos

En esta sección se presenta la metodología propuesta para la integración de diferentes agentes de negociación que manejan lenguajes heterogéneos. El objetivo de esta metodología es asistir a los desarrolladores o dueños de agentes de negociación, en la tarea de publicar y describir en la ontología compartida, el conjunto de primitivas de negociación codificadas en su agente.

1. **Se caracterizan los agentes de negociación.** En este paso, para cada par de agentes se describen los aspectos relevantes del agente, como son: identificador, nombre de la compañía, dirección electrónica, valores para la negociación, rol que juega

(comprador o vendedor), y el conjunto de primitivas de negociación y su descripción utilizados para comunicarse.

2. **Se clasifican las primitivas de negociación de acuerdo a las clases de la ontología.** Este paso consiste en identificar la clase a la que pertenece cada primitiva de negociación a partir de la descripción proporcionada por el desarrollador del agente. La clasificación de las primitivas depende del momento en el que son utilizadas dentro del protocolo de negociación. De acuerdo con esta clasificación las primitivas pueden ser: de **inicio**, si inician el proceso de negociación; de **reacción**, si responden a un enunciado dado; y de **finalización**, si terminan el proceso de negociación.
3. **Se alinean las primitivas de negociación en un diagrama de estados finitos.** El tercer paso de la metodología consiste en la alineación de las primitivas utilizando un diagrama de estados finitos del proceso de negociación, para comparar la definición de la primitiva y su uso dentro del protocolo de negociación. Existen tres estados generales dentro del ciclo de la negociación, éstos son: inicio de la negociación, toma de decisión, y finalización de la negociación. Los estados del ciclo se describen a continuación:
 - a. Un agente se encuentra en un estado de inicio cuando emite un primer mensaje de solicitud de propuesta,
 - b. Un agente se encuentra en un estado de toma de decisión cuando recibe un mensaje con una oferta o contraoferta, y
 - c. Un agente se encuentra en un estado de finalización cuando recibe uno de tres mensajes: aceptación, rechazo o falla de la oferta negociada.

4. **Se establecen las relaciones de sinónimos y similares para las primitivas que pueden ocasionar problemas de entendimiento.** En el diagrama de estados finitos se identifican las relaciones entre las primitivas de la negociación. El establecimiento de la relaciones de equivalencia entre las primitivas de negociación es un proceso que requiere de un análisis comparativo de la descripción textual de las primitivas y su uso dentro del ciclo de la negociación, es por ello, que para este análisis se realizan los pasos 2 y 3.
5. **Se publican las primitivas en la ontología con sus respectivas propiedades.** Este paso consiste en la actualización de la ontología con las nuevas definiciones de primitivas de negociación.

Capítulo 4

Arquitectura del sistema de negociación

En esta sección se presenta el diseño detallado de los módulos del sistema de negociación electrónica, el cual sirve como base para la experimentación y validación de la aportación de este trabajo de investigación.

4.1 Sistema de negociación electrónica

Para la ejecución de procesos de negociación, se implementó la arquitectura que se muestra en la Figura 5. En esta sección se describen la funcionalidad y las tecnologías de implementación de cada componente.

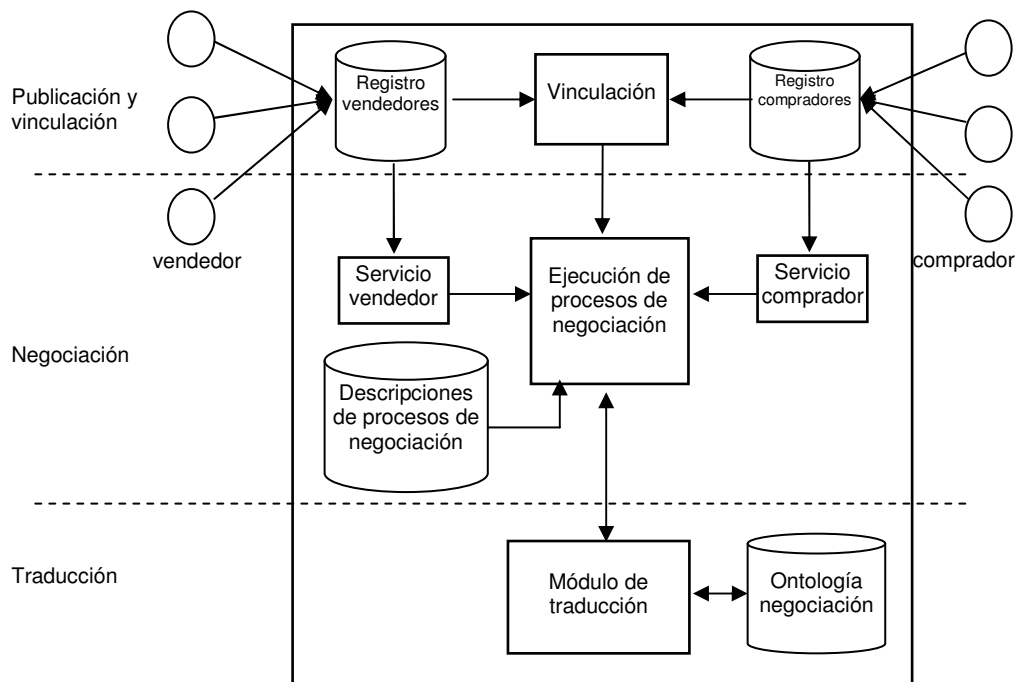


Figura 5. Arquitectura general del sistema

- a. *Módulo de vinculación.* Es un módulo en Java, el cual se encuentra continuamente rastreando los registros de compradores y los registros de vendedores, en busca de coincidencias. Cuando este módulo encuentra un potencial vínculo de negocio entre la descripción de lo que el comprador busca con lo que el vendedor ofrece, envía un mensaje a los participantes para que inicien un proceso de negociación.

Arquitectura del sistema de negociación

b. *Motor de procesos de negociación.* Es un motor basado en BPEL4WS (Business Process Execution Language for Web Services) que controla la ejecución de procesos de negociación entre múltiples agentes de acuerdo con los protocolos preestablecidos. BPEL4WS proporciona un lenguaje para la especificación formal de procesos de negocios y protocolos de interacción entre negocios. La interacción con cada agente ocurre a través de interfaces de servicios Web, y la estructura de la relación en el nivel de interfaz se encapsula en lo que se llama una liga de socio. En la Figura 6 se muestra la interacción entre los agentes a través del proceso de negociación.

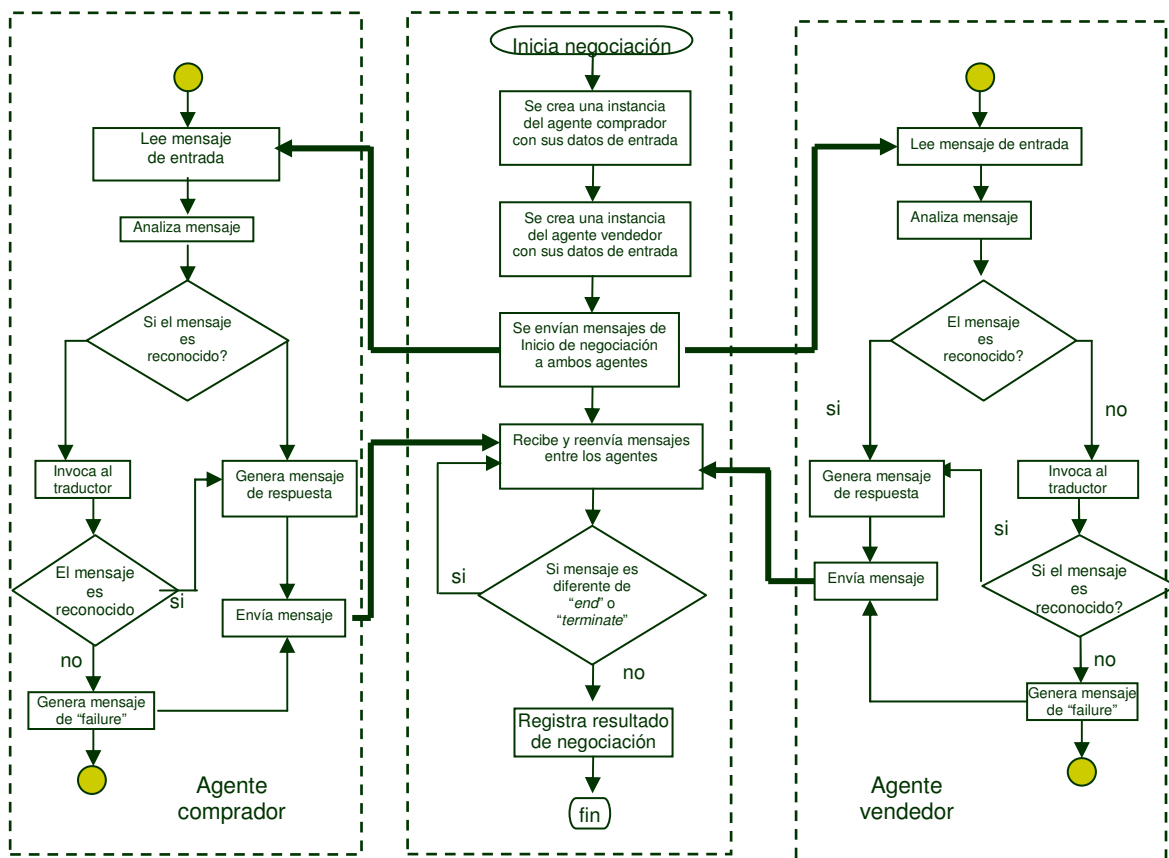


Figura 6. Interacción entre los agentes

c. *Agentes comprador y vendedor.* Son entidades de software utilizadas por sus respectivos dueños para programar sus preferencias y estrategias de negociación. Por ejemplo, un agente vendedor se programa para maximizar sus ganancias, estableciendo el precio mínimo aceptable y el precio deseable de venta. Por el contrario, un agente comprador está buscando minimizar su pago. Durante el diseño de los agentes de negociación, se identificaron tres elementos importantes: estrategias de negociación, el conjunto de mensajes para la comunicación y el protocolo para la ejecución de procesos de negociación. En la Figura 7 se muestra el diagrama del tratamiento de los mensajes de los agentes. Los requerimientos para estos elementos fueron especificados de la siguiente forma:

1. Las estrategias de negociación deben ser privadas para cada agente, debido a que son agentes que compiten y no muestran claramente sus intenciones.
2. Los mensajes deben ser generados de forma privada e independiente.
3. El protocolo de negociación debe ser público o compartido por todos los participantes, para que todos los participantes en la negociación sigan el mismo conjunto de reglas de interacción. El protocolo de negociación establece las reglas que los agentes deben seguir para interactuar.

d. *El traductor.* Es un módulo invocado cuando un agente que recibe un mensaje desconoce la primitiva recibida. En la siguiente sección se describe este módulo.

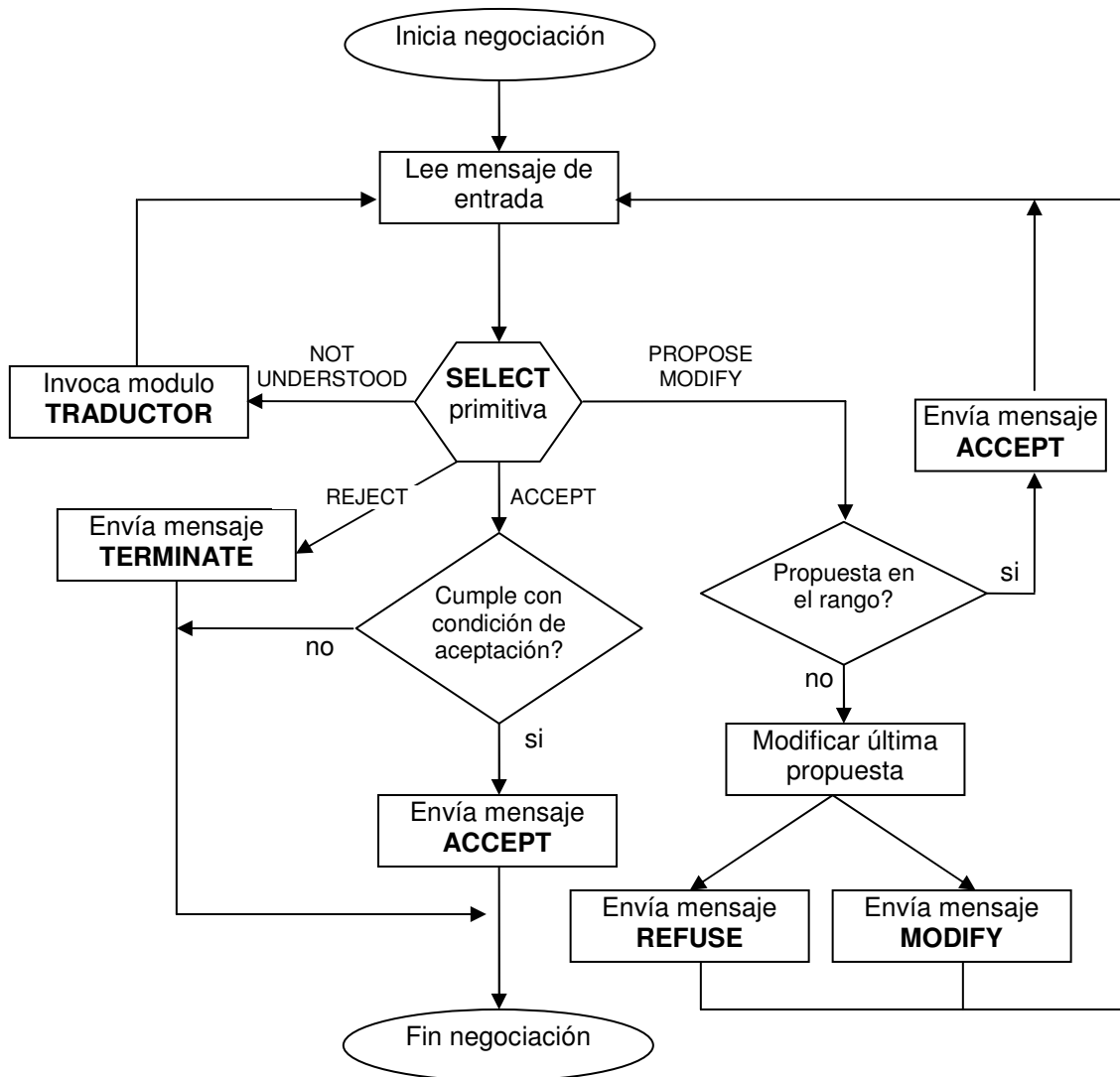


Figura 7. Tratamiento de mensajes de los agentes

4.2 Sistema de traducción entre agentes

Uno de los usos importantes de las ontologías en la comunicación, es asistir en la traducción entre lenguajes y representaciones diferentes. Se han presentado varias propuestas de traducción entre lenguajes. Un enfoque es diseñar un traductor

por cada par de agentes que manejan un lenguaje diferente, sin embargo, esto requeriría $(n^2-n)/2$ traductores para n lenguajes diferentes, (ver inciso (a) de la Figura 8).

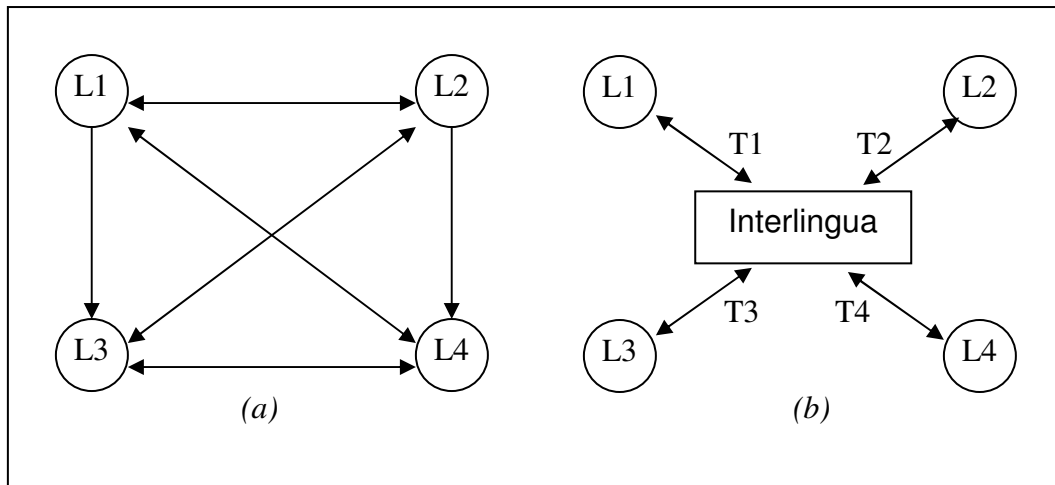


Figura 8. Traducción entre lenguajes diferentes

Otro enfoque es el uso de una ontología como una *interlingua* (lengua intermedia), para dar soporte a la traducción, con este enfoque se reduciría el número de traductores a n para n lenguajes diferentes (ver inciso (b) de la Figura 8). Un ejemplo de este enfoque fue presentado por Uschold [27], quien plantea el uso de una ontología para dar soporte en la interoperabilidad entre diferentes sistemas de software (ver inciso (a) de la Figura 9).

El diseño del sistema de traducción que se seleccionó para este proyecto, se basa en el enfoque de la *interlingua*, con la diferencia de que la traducción se ejecuta solamente cuando es necesaria (ver inciso (b) de la Figura 9). Este diseño considera que el problema de comunicación entre agentes depende del nivel de heterogeneidad en el lenguaje de comunicación.

Arquitectura del sistema de negociación

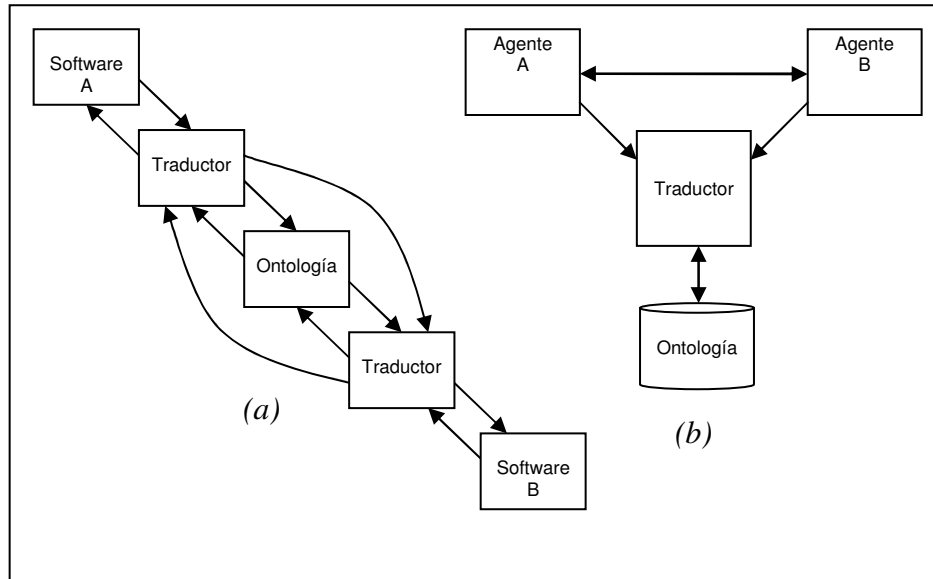


Figura 9. Comparación entre arquitecturas de traducción

El modulo traductor fue implementado utilizando Jena, un marco de trabajo para construir aplicaciones de la Web semántica. Proporciona un entorno de programación para OWL, incluyendo un motor de inferencia basado en reglas.

La funcionalidad del traductor se muestra en la Figura 10. El traductor recibe como parámetros de entrada el agente remitente, el destinatario y la primitiva. Establece una conexión con la ontología y reduce el rango de búsqueda seleccionando la clase de la primitiva. Realiza una búsqueda de las primitivas dentro de esta clase, si encuentra una primitiva del agente destinatario con relación de equivalencia con la primitiva del remitente la devuelve, si no devuelve un mensaje de primitiva no relacionada.

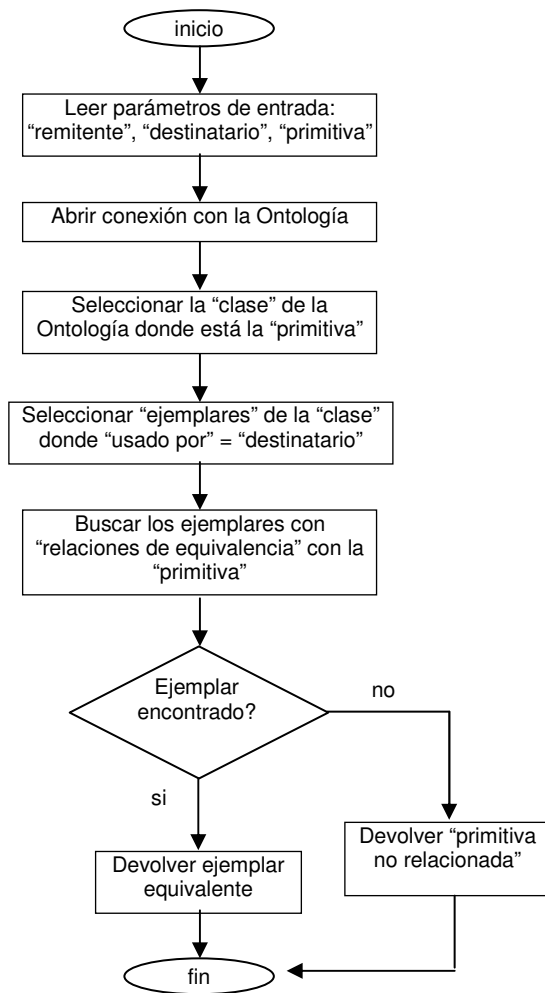


Figura 10. Funcionalidad del sistema de traducción

4.3 Plataforma de servicios Web

Uno de los requisitos que se plantearon al inicio de este trabajo, fue el de implementar el sistema de negociación en un ambiente distribuido basado en la arquitectura de los servicios Web. Sin embargo, se descubrió que los protocolos y estándares básicos utilizados para implementar los servicios Web, tienen limitaciones. Estas limitaciones se deben a que la comunicación entre un cliente de software y un servicio Web, solamente se realiza de forma

sincrónica mediante una petición y una respuesta; cuando lo que se pretende implementar son procesos que implican ejecuciones de larga duración e intercambio de mensajes ya sea de forma sincronizada o asíncrona. Para resolver este problema se estudiaron varias propuestas que han surgido para dar solución a problemas similares, entre ellas se encuentran: OWL-S y BPEL4WS. A continuación se detalla cada uno de ellos.

OWL-S

Es una ontología para la definición de conceptos de servicios. Con OWL-S, un servicio se presenta mediante el modelo del proceso, el perfil y el grounding del servicio. El modelo del proceso describe cómo un servicio realiza sus tareas. Incluye la información sobre las entradas, las salidas, las precondiciones y los resultados. El perfil del servicio proporciona la descripción general de un servicio Web, con el objetivo de ser publicado y compartido para facilitar el descubrimiento del servicio. El grounding especifica cómo se invoca un servicio, detallando cómo los procesos atómicos en el modelo del proceso del servicio se enlazan mediante un protocolo de mensajes concreto.

BPEL4WS

Es un estándar para la ejecución de procesos basado en los servicios Web. Representa la convergencia de las ideas de las especificaciones XLANG y WSFL. Este estándar se propuso como una solución para la ejecución de procesos de larga duración y con modelos de comunicación asíncrona. La negociación se puede considerar un proceso de negocio en el sentido de que representa un conjunto de actividades que son realizadas siguiendo un protocolo para lograr un objetivo. Más específicamente, la

Arquitectura del sistema de negociación

negociación es un proceso de larga duración y cuyo modelo de comunicación puede ser asíncrono. BPEL4WS proporciona un lenguaje formal para la especificación de procesos de negocios y protocolos de interacción de negocios. Extiende el modelo de interacción básico de los servicios Web y da soporte para las transacciones de negocios. En la Figura 11 se muestra la pila de protocolos y estándares de las tecnologías de servicios Web.

Se seleccionó BPEL4WS junto con la tecnología de servicios Web (SOAP, WSDL) para la implementación de los servicios de negociación. Esto es debido a que BPEL4WS cuenta con más soporte de entornos de desarrollo e implementaciones de programas servidores de BPEL.

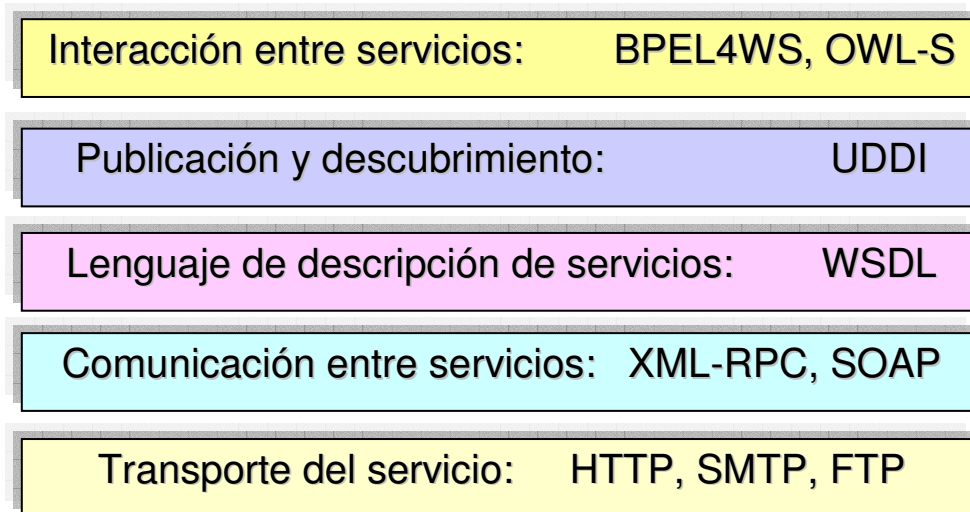


Figura 11. Tecnologías de servicios Web

Capítulo 5

Evaluación de la metodología de solución

En este capítulo se describe el escenario de pruebas para la experimentación, la forma cómo se seleccionaron los casos representativos y la metodología para la experimentación. En particular se describen los casos experimentales que permiten mostrar la aportación.

5.1 Escenario para la experimentación

Las pruebas experimentales tienen como objetivo observar el comportamiento del sistema en diferentes casos. El resultado de estas pruebas se organiza en un conjunto de estadísticas mediante las cuales se realiza un análisis de los resultados obtenidos. La plataforma de experimentación que se montó se muestra en la Figura 12. Este escenario de pruebas consiste de un servidor de mercado, un registro público de servicios y un proveedor de servicios.



Figura 12. Escenario de pruebas

5.2 Selección de los casos para la experimentación

Para seleccionar los casos de experimentación, se consideraron un conjunto de 8 agentes de negociación descritos por diferentes autores (ver tabla 4), que reportaron el uso de distintas primitivas de negociación para la comunicación entre agentes, donde cada agente puede comportarse de dos formas (como

comprador o como vendedor). Estas primitivas se utilizaron para poblar la ontología un conjunto de individuos.

Tabla 4. Resumen de las primitivas de negociación

Autores	Primitivas de Negociación			
Jin Baek Kim, Arie Segev	Initial_offer RFQ Accept Reject Offer Counter-offer			
Stanley Y. W. Su, Chunbo Huang, Joachim Hammer, Patrick C. K. Hung	CFP Propose Accept Terminate Reject Acknowledge Modify Withdraw			
Anthony Chavez, Pattie Maes	accept-offer?(agent, from-agent, offer) what-is-price?(agent, from-agent) what-is-item?(agent, from-agent) add-sell-agent add-buy-agent add-potential-customers(sell-agent, potential-customers) add-potential-sellers(buy-agent, potential-sellers) agent-terminated(marketplace, agent) deal-made(marketplace, sell-agent, buy-agent, item, price)			
Sonia V. Rueda, Alejandro J. García, Guillermo R. Simari	Request_Add(s, h, p) Authorize_Add(s, h, p) Require(s, h, p) Demand(s, h, p) Accept(s, h, p)		Reject(s, h, p) Unable(s, h, p) Require-for(s, h, p, q) Insist_for(s, h, p, q) Demand_for(s, h, p, q)	
Haifei Li, Chunbo Huang and Stanley Y.W Su	Call for proposal Propose proposal Reject proposal Withdraw proposal		Accept proposal Modify proposal Acknowledge message Terminate negotiation	
Dignum, Jan Dietz, Egon Verharen and Hans Weigand	request-quotation give-quotation order delivered paid			
FIPA Communicative Acts	Accept Proposal Agree Cancel Call for Proposal Confirm Disconfirm	Failure Inform Inform If Inform Ref Not Understood Propagate	Propose Proxy Query If Query Ref Refuse Reject Proposal	Request Request When Request Whenever Subscribe
Jürgen Müller	Initiators: Propose, Arrange, Request, Inform, Query, Command, Inspect	Reactors: Answer, Refine, Modify, Change, Bid, Send, Reply, Refuse, Explain	Completers: Confirm, Promise, Commit, Accept, Reject, Grant, Agree.	

Sea $A = \{a, b, c, d, e, f, g, \text{ y } h\}$, el conjunto de agentes heterogéneos. El total de posibles enlaces o casos de negociación entre ellos está dado por $n \times n$, dónde n es el número de agentes heterogéneos. En la Figura 13 se muestra la matriz de los enlaces de negociación que pueden existir entre los 8 agentes heterogéneos.

$N =$

a,a	a,b	a,c	a,d	a,e	a,f	a,g	a,h
b,a	b,b	b,c	b,d	b,e	b,f	b,g	b,h
c,a	c,b	c,c	c,d	c,e	c,f	c,g	c,h
d,a	d,b	d,c	d,d	d,e	d,f	d,g	d,h
e,a	e,b	e,c	e,d	e,e	e,f	e,g	e,h
f,a	f,b	f,c	f,d	f,e	f,f	f,g	f,h
g,a	g,b	g,c	g,d	g,e	g,f	g,g	g,h
h,a	h,b	h,c	h,d	h,e	h,f	h,g	h,h

Figura 13. Matriz de casos de negociación

Para este conjunto de 8 agentes el total de casos de negociación es de 64, de los cuales identificamos aquellos casos en los que la negociación sería entre agentes iguales, es decir, todos aquellos enlaces del tipo $\{(a, a), \dots, (h, h)\}$; los cuales no poseen ningún grado de heterogeneidad. Por lo tanto el número de casos de negociación heterogéneos está dado por:

$$n^2 - n$$

dónde n es el número de agentes heterogéneos.

Sin embargo, esta expresión se puede reducir aún más, porque cómo se puede observar en la matriz de la Figura 13, se necesitan exactamente las mismas relaciones de equivalencia por pares de agentes. Por ejemplo, el enlace de negociación entre el agente e , f ; requiere las mismas relaciones de equivalencia entre primitivas de negociación que el enlace f , e . Por lo tanto el número de casos de negociación heterogéneos está dado por:

$$\frac{n^2 - n}{2} \quad (4.1)$$

dónde n es el número de agentes heterogéneos.

Para el conjunto de 8 agentes, aplicamos la expresión 4.1, y el total de casos de negociación que pueden poseer algún grado de heterogeneidad es de 28.

Para seleccionar los casos más representativos de estos 28, se construyó la tabla 5, en la cual se colocan por pares los conjuntos de primitivas de negociación de cada uno de los agentes, se compara cada par de conjuntos de primitivas y se calcula el número de primitivas iguales, la tabla completa se muestra en el anexo A.

Tabla 5. Conjunto de posibles combinaciones entre agentes heterogéneos

Caso	Agentes	Primitivas iguales	Nivel de heterogeneidad
1	b, a	2	Alta
2	c, a	1	Alta
3	d, a	2	Alta
4	e, a	2	Alta
5	f, a	0	Alta
6	g, a	2	Alta
7	h, a	2	Alta
8	c, b	1	Alta
9	d, b	2	Alta

Evaluación de la metodología

10	e, b	7	Baja
11	f, b	0	Alta
12	g, b	3	Media
13	h, b	2	Alta
14	d, c	1	Alta
15	e, c	0	Alta
16	f, c	0	Alta
17	g, c	1	Alta
18	h, c	1	Alta
19	e, d	2	Alta
20	f, d	1	Alta
21	g, d	3	Media
22	h, d	2	Alta
23	f, e	0	Alta
24	g, e	4	Media
25	h, e	4	Media
26	g, f	1	Alta
27	h, f	0	Alta
28	h, g	9	Baja

La clasificación de los casos se realizó de la siguiente manera, si el número de primitivas iguales es mayor, entonces el nivel de heterogeneidad es baja, por el contrario, si el número de primitivas iguales es menor o cero, entonces el nivel de heterogeneidad es alta.

- a. Los casos con nivel de heterogeneidad baja son aquellos que manejan la mayoría de sus primitivas iguales, pero que pueden tener alguna diferencia en tres momentos dentro del protocolo, es decir, que tienen una primitiva diferente al inicio, en el ciclo intermedio o al terminar la negociación
- b. Los casos con nivel de heterogeneidad alta son todos aquellos que tienen 0, 1 ó 2 primitivas iguales y la mayoría son diferentes.
- c. Todos los demás casos son intermedios, ya que tienen desde 3 hasta 4 primitivas iguales.

Por lo tanto se seleccionó un caso de cada clase para evaluar la aportación de esta tesis. De cada caso se ejecutaron 30 pruebas. El teorema del límite central [45] establece que la distribución normal estándar ofrece una buena aproximación a una distribución con una desviación estándar desconocida para muestras de tamaño 30 o mayor.

5.3 Experimento 1: heterogeneidad alta

Objetivo

Aplicar la metodología de solución propuesta para permitir que los agentes A y B, con nivel de heterogeneidad alta interactúen, reduciendo los problemas de comunicación por falta de entendimiento.

Procedimiento

1. Se definen los agentes de negociación.

Agente A	Valores
Identificación	ID: 00015 Company: "Firefox" Rol: "Buyer"
Estrategia	Quantity: <valor numérico> maxPayment: <valor numérico>
Lenguaje	{(CFP, "Initiate a negotiation process by calling for proposals"), (Propose, "Issue a proposal or a counterproposal"), (Accept, "Accept the terms specified in a proposal without further modifications"), (Terminate, "Unilaterally terminate the current negotiation process"), (Reject, "Reject the current proposal with or without an

attached explanation”),
 (Acknowledge, “Acknowledge the receipt of a message”),
 (Modify, “Modify the proposal that was sent last”),
 (Withdraw, “Withdraw the last proposal”}}

Agente B	Valores
Identificación	ID: 00012 Company: “Microsoft” Rol: “Seller”
Estrategia	initialPrice: <valor numérico> lastPrice: <valor numérico> deliveryPrice: <valor numérico>
Lenguaje	{ (Initial_offer, “Send initial offer”), (RFQ, “Send request for quote”), (Accept, “Accept offer”), (Reject, “Reject offer”), (Offer, “Send offer”), (Counter-offer, “Send counter offer”) }

2. Se clasifican las primitivas de acuerdo a las clases de la ontología.

Agente	Starter	Reactor	Completer
A (Buyer)	CFP	Propose Modify Withdraw Acknowledge	Accept Reject Terminate NotUnderstood
B (Seller)	RFQ	Initial_Offer Offer Counter-offer	Accept Reject NotUnderstood

3. Se alinean las primitivas de negociación en el diagrama de estados finitos.

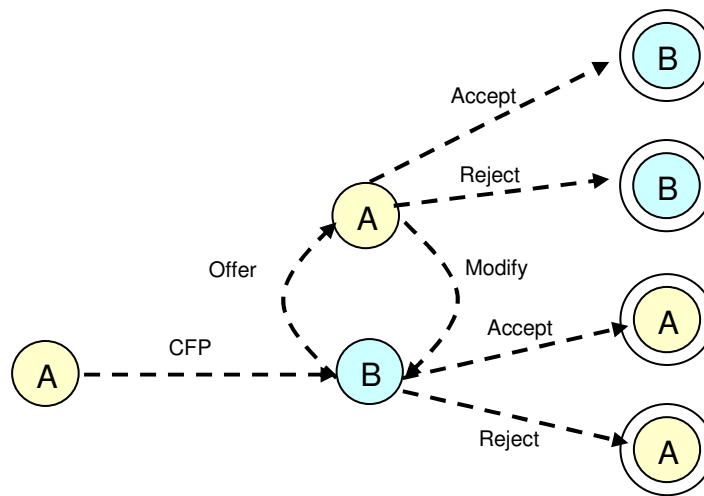


Figura 14. Diagrama de estados finitos, agentes A y B

4. Se establecen las relaciones de sinónimos y similares para las primitivas que pueden ocasionar problemas de entendimiento.

A		B
CFP	isSynonymOf	RFQ
Propose	isSynonymOf	Offer
Propose	isSynonymOf	Initial_Offer
Modify	isSynonymOf	Counter_Offer
Withdraw	isSynonymOf	Counter_Offer
Terminate	isSimilarOf	Reject

En este caso, la primitiva *Acknowledge* queda sin relación en la ontología, sin embargo, queda la posibilidad de que el agente que no la usa, la adquiera para favorecer la comunicación.

5. Se publican las primitivas en la ontología con sus respectivas propiedades.

6. Se ejecuta el proceso de negociación. Se realizaron 30 pruebas con estos dos agentes, los valores de los datos de precio del comprador, precio del vendedor y cantidad, se generaron al azar.

Resultado

En la tabla 6 se muestran los resultados del experimento. Esta tabla muestra los datos relevantes de las 30 negociaciones realizadas con los agentes A y B.

Tabla 6. Resultados del experimento 1

Prueba	Precio	Precio	Resultado	Resultado
	Comprador	Vendedor	Sin traducción	Con traducción
1	1963	974	notUnderstood	Accept
2	145	1778	notUnderstood	Reject
3	206	855	notUnderstood	Accept
4	621	908	notUnderstood	Accept
5	64	530	notUnderstood	Accept
6	866	705	notUnderstood	Accept
7	260	1407	notUnderstood	Reject
8	165	1853	notUnderstood	Reject
9	1927	1680	notUnderstood	Accept
10	1745	1717	notUnderstood	notUnderstood
11	555	841	notUnderstood	Accept
12	1959	1699	notUnderstood	Accept
13	1241	1180	notUnderstood	Accept
14	331	1186	notUnderstood	Accept
15	730	1636	notUnderstood	Accept
16	1616	1491	notUnderstood	Accept
17	461	1890	notUnderstood	Reject
18	1116	301	notUnderstood	Accept
19	1257	798	notUnderstood	Accept
20	1461	1066	notUnderstood	Accept
21	460	676	notUnderstood	Reject
22	1024	1977	notUnderstood	Accept
23	1242	1276	notUnderstood	Accept
24	981	133	notUnderstood	notUnderstood
25	333	1036	notUnderstood	Reject
26	63	1311	notUnderstood	Reject
27	218	1449	notUnderstood	Reject
28	870	1897	notUnderstood	Accept
29	650	1828	notUnderstood	Reject
30	713	215	notUnderstood	notUnderstood

En la Figura 15 se muestra la gráfica comparativa de los resultados de este caso. El primer conjunto de barras muestra los resultados de la negociación entre los agentes A y B sin utilizar el traductor. Y el segundo muestra los resultados para estos agentes utilizando el traductor.

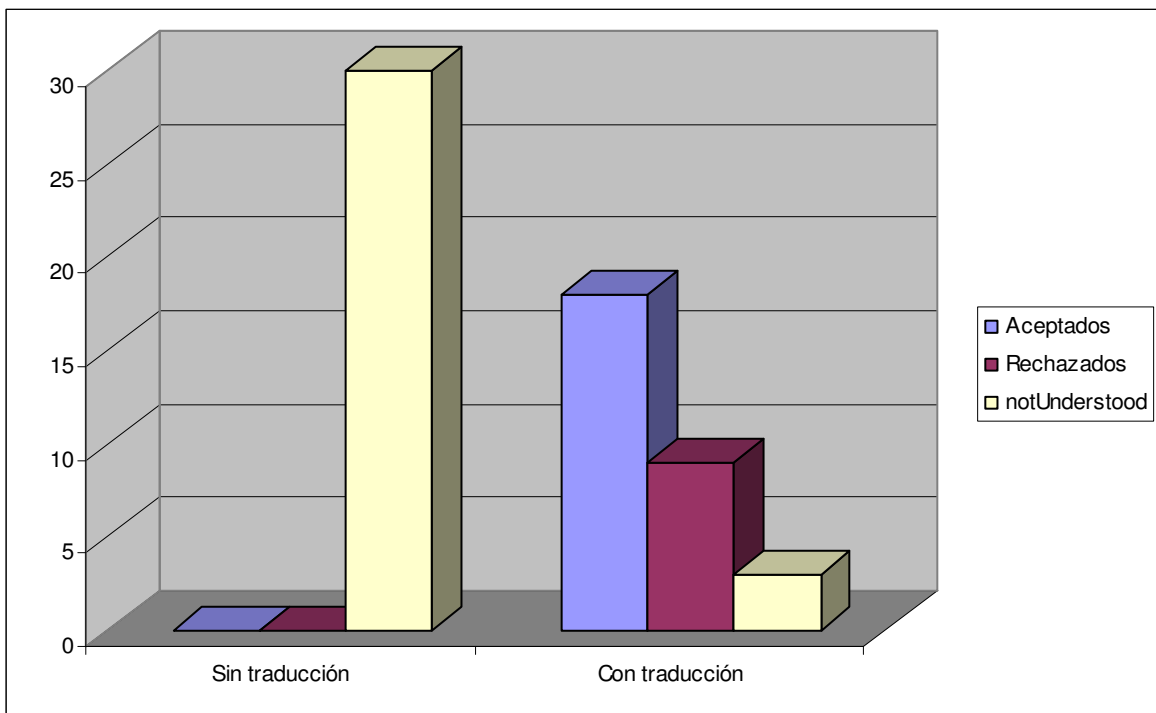


Figura 15. Gráfica de resultados del experimento 1

Análisis del resultado

Como se observa en la tabla 6, hubo algunas negociaciones que terminaron por falta de entendimiento. Esto es debido a la ocurrencia del mensaje *Acknowledge*, el cual quedó sin relación en la ontología. Se sugiere que el agente B adquiriera el uso de esta primitiva para reducir completamente el problema por falta de entendimiento.

Sin embargo, el experimento arroja resultados satisfactorios con respecto al objetivo de la tesis, ya que sin el uso de la ontología y el traductor, estas negociaciones no se pueden realizar, todas tienen como resultado el mensaje *notUnderstood*, ya que desde la primera iteración ocurren problemas de falta de entendimiento.

Los resultados de la gráfica de la Figura 15 muestran que para un caso de heterogeneidad alta el número de fallas de la negociación por problemas de heterogeneidad en el lenguaje se puede reducir hasta un 90%, utilizando el traductor.

5.4 Experimento 2: heterogeneidad media

Objetivo

Aplicar la metodología de solución propuesta a un caso de heterogeneidad media, utilizando los agentes G y E, reduciendo los problemas de comunicación por falta de entendimiento.

Procedimiento

1. Se definen los agentes de negociación.

Agente G	Valores
Identificación	ID: 00016 Company: "Chinese" Rol: "Buyer"
Estrategia	Quantity: <valor numérico> maxPayment: <valor numérico>
Lenguaje	{ (Accept Proposal, "The action of accepting a previously submitted proposal to perform an action"), (Agree, "The action of agreeing to perform some action, possibly

in the future”),

(Cancel, “The action of one agent informing another agent that the first agent no longer has the intention that the second agent perform some action”),

(Call for Proposal, “The action of calling for proposals to perform a given action”),

(Confirm, “The sender informs the receiver that a given proposition is true, where the receiver is known to be uncertain about the proposition”),

(Disconfirm, “The sender informs the receiver that a given proposition is false, where the receiver is known to believe that the proposition is true”),

(Failure, “The action of telling another agent that an action was attempted but the attempt failed”),

(Inform, “The sender informs the receiver that a given proposition is true”

(Inform If, “A macro action for the agent of the action to inform the recipient whether or not a proposition is true”),

(Not Understood, “The sender of the not-understood communicative act received a communicative act that it did not understand”),

(Propagate, “The sender wants the receiver to identify the agents denoted by the given descriptor and send the received propagate message to them”),

(Propose, “The action of submitting a proposal to perform a certain action, given certain preconditions”),

(Proxy, “The sender wants the receiver to select target agents denoted by a given description and send an embedded message to them”),

(Query If, “The action of asking another agent whether or not a given proposition is true”),

(Query Ref, “The action of asking another agent for the object referred to by a referential expression”),

(Refuse, “The action of refusing to perform a given action, and explain the reason for the refusal”),

(Reject Proposal, “The action of rejecting a proposal to perform some action during a negotiation”),

(Request, “The sender requests the receiver to perform some action”

(Request When, “The sender wants the receiver to perform some

```

action when some given proposition becomes true”),
(Request Whenever, “The sender wants the receiver to perform
some action as soon as some proposition becomes true and
thereafter each time the proposition becomes true again”),
(Subscribe, “The act of requesting a persistent intention to notify
the sender of the value of a reference, and to notify again
whenever the object identified by the reference changes”)
}
    
```

Agente E	Valores
Identificación	ID: 00017 Company: “Casio” Rol: “Seller”
Estrategia	initialPrice: <valor numérico> lastPrice: <valor numérico> deliveryPrice: <valor numérico>
Lenguaje	{ (Call for proposal, “Initiate a call-for-proposal”), (Propose proposal, “Send a proposal or a counterproposal”), (Reject proposal, “Reject the received proposal with or without an attached explanation”), (Withdraw proposal, “Withdraw the previous proposal that was sent”), (Accept proposal, “Accept the terms and conditions specified in a proposal without further modifications”), (Change proposal, “Change the proposal that was sent”), (Inform proposal, “Inform the receipt of a proposal”), (Terminate negotiation, “Unilaterally terminate the negotiation process”) }

2. Se clasifican las primitivas de acuerdo a las clases de la ontología.

Agente	Starter	Reactor	Completer
G (Buyer)	Call for Proposal Request Request When	Confirm Disconfirm Failure	Accept Proposal Agree Cancel

	Request Whenever	Inform Inform if Propagate Propose Proxy Query if Query ref Subscribe	Refuse Reject Proposal Not Understood
E (Seller)	Call for proposal	Propose proposal Withdraw proposal Change proposal Inform proposal	Accept proposal Reject proposal Terminate negotiation Not Understood

3. Se alinean las primitivas de negociación en el diagrama de estados finitos.

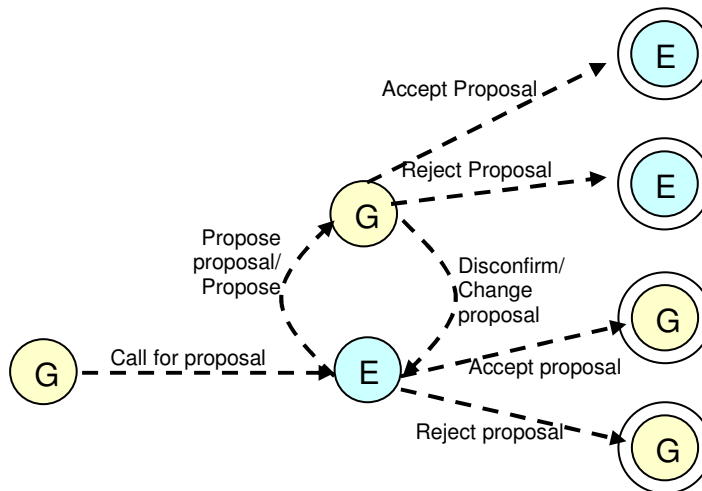


Figura 16. Diagrama de estados finitos, agentes G y E

4. Se establecen las relaciones de sinónimos y similares para las primitivas que pueden ocasionar problemas de entendimiento.

G		E
Request	isSynonymOf	Call for proposal
Propose	isSynonymOf	Propose proposal
Inform	isSimilarTo	Inform proposal
Disconfirm	isSimilarTo	Change proposal

Confirm	isSimilarTo	Acknowledge message
Agree	isSimilarTo	Accept proposal
Cancel	isSimilarOf	Withdraw proposal
Refuse	isSimilarTo	Reject proposal

5. Se publican las primitivas en la ontología con sus respectivas propiedades.

6. Se ejecuta el proceso de negociación. Se realizaron 30 pruebas con estos dos agentes, los valores de los datos de precio del comprador, precio del vendedor y cantidad, se generaron aleatoriamente.

Resultado

En la tabla 7 se muestran los resultados del experimento. Esta tabla muestra los datos relevantes de las negociaciones realizadas con los agentes G y E.

Tabla 7. Resultados del experimento 2

Prueba	Precio Comprador	Precio Vendedor	Resultado Sin traducción	Resultado Con traducción
1	606	1289	notUnderstood	Disconfirm
2	1431	679	Accept proposal	Accept proposal
3	1722	122	Accept proposal	Accept proposal
4	1944	674	Accept proposal	Accept proposal
5	589	383	Accept proposal	Accept proposal
6	327	1871	notUnderstood	Disconfirm
7	1246	1526	notUnderstood	Accept proposal
8	1823	1548	Accept proposal	Accept proposal
9	1494	458	Accept proposal	Accept proposal
10	184	814	Disconfirm	Disconfirm
11	1281	1243	Accept proposal	Accept proposal
12	1912	1970	Disconfirm	Accept proposal
13	1717	1254	Accept proposal	Accept proposal
14	359	1780	notUnderstood	Disconfirm
15	805	1715	notUnderstood	Accept proposal
16	994	669	Accept proposal	Accept proposal
17	1775	417	Accept proposal	Accept proposal

Evaluación de la metodología

18	1984	431	Accept proposal	Accept proposal
19	999	1961	notUnderstood	Disconfirm
20	1811	983	Accept proposal	Accept proposal
21	1233	805	Accept proposal	Accept proposal
22	1762	1455	Accept proposal	Accept proposal
23	652	1264	notUnderstood	Disconfirm
24	1088	420	Accept proposal	Accept proposal
25	1027	1582	notUnderstood	Disconfirm
26	1025	1555	notUnderstood	Accept proposal
27	821	1419	notUnderstood	Accept proposal
28	1208	1172	Accept proposal	Accept proposal
29	795	1889	notUnderstood	Disconfirm
30	1909	464	Accept proposal	Accept proposal

En la Figura 17 se muestra la gráfica comparativa de los resultados de este caso. El primer conjunto de barras muestra los resultados de la negociación entre los agentes G y E sin utilizar el traductor. Y el segundo muestra los resultados para estos agentes utilizando el traductor.

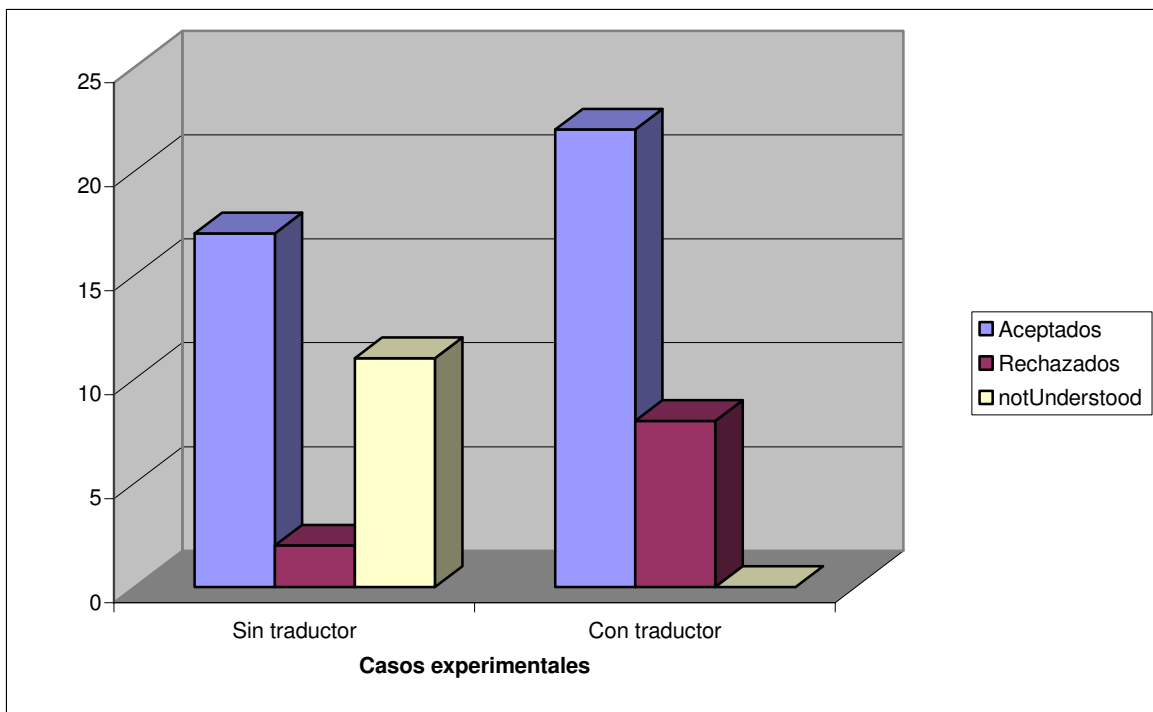


Figura 17. Gráfica de resultados del experimento 2

Análisis del resultado

Como se observa en la tabla 7, las negociaciones realizadas entre los agentes G y E sin el traductor tienen como resultados 17 acuerdos aceptados, 2 rechazados y terminaron por falta de entendimiento. Por el contrario, los resultados con el traductor fueron de 22 acuerdos aceptados y rechazados, no hubo negociaciones que terminan por falta de entendimiento.

Los resultados de la gráfica de la Figura 17 muestran que para un caso de heterogeneidad media el número de fallas de la negociación por problemas de heterogeneidad en el lenguaje se pueden reducir hasta un 100% utilizando el traductor.

5.5 Experimento 3: heterogeneidad baja

Objetivo

Evaluar la metodología de solución para el caso de heterogeneidad baja. De acuerdo a la clasificación, un caso de heterogeneidad baja es aquel en el que ambos agentes manejan casi el mismo conjunto de primitivas. Por lo tanto para este caso se utilizaron las primitivas del agente B, con tres tipos de variantes. Una variante de observación es una diferencia en las primitivas de inicio del ciclo, la segunda variante es una diferencia en las primitivas dentro del ciclo y una tercera variante es una diferencia en las primitivas al final del ciclo.

Procedimiento

1. Se definen los agentes de negociación.

Evaluación de la metodología

Lenguaje del agente B	Variante 1 diferencia al principio del ciclo	Variante 2 diferencia en el ciclo	Variante 3 diferencia al final del ciclo
CFP	RFQ	CFP	CFP
Propose	Propose	Propose	Propose
Accept	Accept	Accept	Agree
Reject	Reject	Reject	Reject
Modify	Modify	Counter offer	Modify
Acknowledge	Acknowledge	Acknowledge	Acknowledge
Terminate	Terminate	Terminate	Terminate
Withdraw	Withdraw	Withdraw	Withdraw

2. Se clasifican las primitivas de acuerdo a las clases de la ontología.

Agente	Starter	Reactor	Completer
B (Buyer)	CFP	Propose Modify Withdraw Acknowledge	Accept Reject Terminate
Variante 1	RFQ	Propose Modify Withdraw Acknowledge	Accept Reject Terminate
Variante 2	CFP	Propose Counter offer Withdraw Acknowledge	Accept Reject Terminate
Variante 3	CFP	Propose Modify Withdraw Acknowledge	Agree Reject Terminate

3. Se alinean las primitivas de negociación en el diagrama de estados finitos.

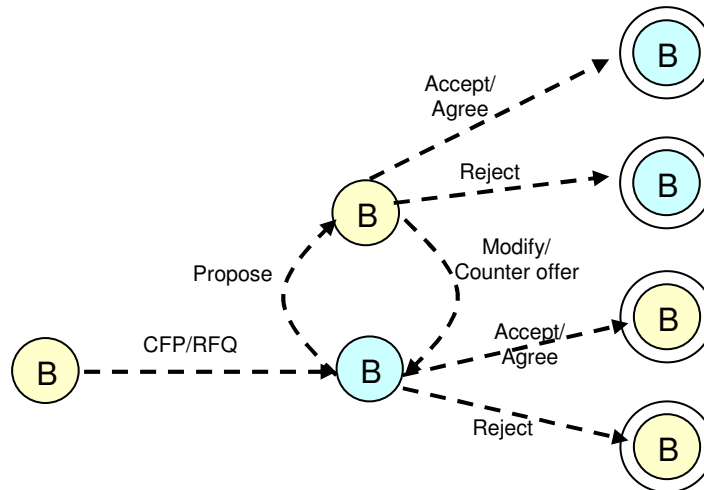


Figura 18. Diagrama de estados finitos, agente B

4. Se establecen las relaciones de sinónimos y similares para las primitivas que pueden ocasionar problemas de entendimiento.

B		Variante
CFP	isSynonymOf	RFQ
Modify	isSynonymOf	Counter offer
Accept	isSynonymOf	Agree

5. Se publican las primitivas en la ontología con sus respectivas propiedades.

6. Se ejecuta el proceso de negociación. Se realizaron 30 pruebas con el agente B como comprador y cada una de las variantes como vendedor, los valores de los datos de precio del comprador, precio del vendedor y cantidad, se generaron al azar.

Resultado

En la tabla 8 se muestran los resultados del experimento.

Tabla 8. Resultados del experimento 3

Prueba	AL PRINCIPIO Sin traducción Resultado	EN EL CICLO Sin traducción Resultado	AL FINAL Sin traducción Resultado	TODOS LOS CASOS Con traducción Resultado
1	notUnderstood	notUnderstood	notUnderstood	Accept
2	notUnderstood	Accept	notUnderstood	Accept
3	notUnderstood	Accept	notUnderstood	Accept
4	notUnderstood	Accept	notUnderstood	Accept
5	notUnderstood	Accept	notUnderstood	Accept
6	notUnderstood	notUnderstood	notUnderstood	Accept
7	notUnderstood	notUnderstood	notUnderstood	Accept
8	notUnderstood	Accept	notUnderstood	Accept
9	notUnderstood	Accept	notUnderstood	Accept
10	notUnderstood	Reject	Reject	Reject
11	notUnderstood	notUnderstood	notUnderstood	Accept
12	notUnderstood	Accept	notUnderstood	Accept
13	notUnderstood	Accept	notUnderstood	Accept
14	notUnderstood	Accept	notUnderstood	Accept
15	notUnderstood	Accept	notUnderstood	Accept
16	notUnderstood	notUnderstood	notUnderstood	Accept
17	notUnderstood	notUnderstood	notUnderstood	Accept
18	notUnderstood	notUnderstood	notUnderstood	Accept
19	notUnderstood	Accept	notUnderstood	Accept
20	notUnderstood	notUnderstood	notUnderstood	Accept
21	notUnderstood	notUnderstood	notUnderstood	Accept
22	notUnderstood	notUnderstood	notUnderstood	Accept
23	notUnderstood	Accept	notUnderstood	Accept
24	notUnderstood	Accept	notUnderstood	Accept
25	notUnderstood	notUnderstood	notUnderstood	Accept
26	notUnderstood	Accept	notUnderstood	Accept
27	notUnderstood	Accept	notUnderstood	Accept
28	notUnderstood	notUnderstood	Reject	Reject
29	notUnderstood	notUnderstood	notUnderstood	Accept
30	notUnderstood	notUnderstood	notUnderstood	Accept

La tabla 8 muestra que las negociaciones entre el agente B y la primera variante, tienen resultados de fuerte afectación con respecto al logro de acuerdos, todos los resultados fueron de falla por falta de entendimiento cuando no se utiliza el traductor. Las

negociaciones entre el agente B y la segunda variante, donde la diferencia se da durante el ciclo intermedio del protocolo de negociación, tuvieron 15 acuerdos aceptados, 1 rechazado y 14 fallas por falta de entendimiento cuando no se utiliza el traductor. Las negociaciones entre el agente B y la tercera variante, donde la diferencia se da al final del ciclo, específicamente en la primitiva de acuerdo; los resultados fueron de 28 fallas por falta de entendimiento y 2 rechazos cuando no se utiliza el traductor.

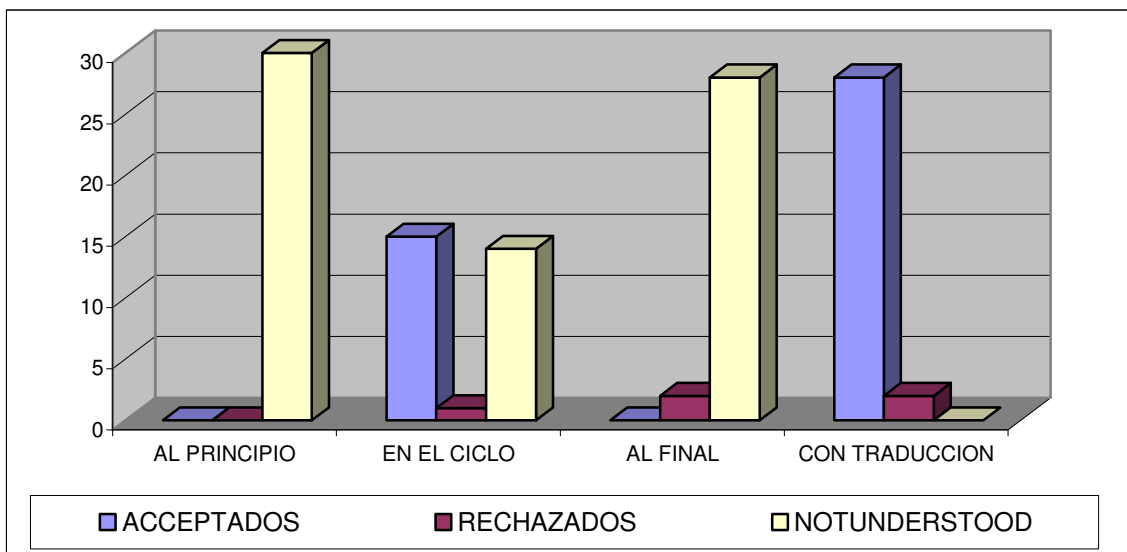


Figura 19. Gráfica de resultados del experimento 3

Análisis del resultado

Como se observa en la gráfica de la Figura 19, cuando hay una diferencia en la primitiva de inicio, a pesar de que las demás primitivas sean iguales, el resultado de las negociaciones sin el traductor terminan todas por falta de entendimiento. Otro resultado importante es cuando la primitiva distinta es la de acuerdo al final del ciclo, ya que no se lograrán acuerdos a pesar de que la negociación se haya realizado. En las negociaciones para estas tres variantes, se pueden lograr mejoras de hasta 100%.

5.6 Discusión de los resultados

Los resultados de las pruebas experimentales muestran que el caso más difícil es el de agentes con mayor heterogeneidad en los lenguajes de comunicación, ya que si se tratan de ejecutar las negociaciones sin el uso del traductor, todas fallan. Para este caso se puede lograr hasta un 90% de reducción de fallas por falta de entendimiento cuando se incorpora el uso del traductor con apoyo de la ontología.

El caso de heterogeneidad media, mostró que aún sin el uso del traductor se pueden lograr algunos acuerdos y rechazos de las negociaciones, sin embargo, también se presentan casos de falla por falta de entendimiento. Para este caso, se pueden reducir hasta el 100% del número de fallas al incorporar el traductor con apoyo de la ontología.

En el caso de heterogeneidad baja, se observan tres variantes. La primera variante es una diferencia en la primitiva de inicio. Este caso tiene una fuerte afectación en el resultado de las negociaciones, ya que sin el uso del traductor todas fallan. La segunda variante consiste de una diferencia en una de las primitivas del ciclo, en este caso se pueden lograr algunos acuerdos y rechazos aun sin la utilización del traductor; pero también existen varias fallas por falta de entendimiento. La tercera variante consiste de una diferencia en la primitiva de acuerdo al final del ciclo. Este caso también tiene una fuerte afectación en el resultado de las negociaciones, ya que a pesar de que éstas se realizan nunca hay acuerdos cuando no se utiliza el traductor. Para todos estos casos se puede lograr una reducción de las fallas por falta de

entendimiento de hasta el 100%, ya que solamente hay diferencia en muy pocas primitivas.

En la tabla 9 se muestran las características de las pruebas experimentales, los cuales se presentaron en la tabla comparativa de trabajos relacionado (ver tabla 2) de la sección 2.1.1 de este documento. El objetivo de esta tabla es mostrar las características que se cumplen en cada caso de experimentación.

Tabla 9. Características de las pruebas experimentales

Exp.	Descripción	Interopera- bilidad	Adaptabilidad	Bilateral	Multilateral	Basado en Agentes
1	Heterogeneidad Alta	✓	✓	✓	✗	✓
2	Heterogeneidad Media	✓	✓	✓	✗	✓
3	Heterogeneidad Baja	✓	✓	✓	✓	✓

Capítulo 6

Conclusiones y trabajos futuros

En este capítulo se presentan las conclusiones del desarrollo de esta investigación doctoral, y se sugieren los trabajos futuros.

6.1 Conclusiones

En este trabajo se presenta una solución al problema de comunicación entre agentes de negociación que usan lenguajes heterogéneos; mediante el desarrollo de una ontología compartida, que permite que se describan, publiquen y se creen las relaciones de equivalencia entre las primitivas.

Las principales contribuciones de esta tesis son las siguientes:

1. Se desarrolló una ontología compartida, mediante la cual se describen, publican y comparten las primitivas de negociación de los agentes que manejan lenguajes de comunicación heterogéneos. Se definieron un conjunto de relaciones de equivalencia entre las primitivas de negociación, a través de las cuales se realizan las traducciones durante el intercambio de mensajes en los procesos de negociación.
2. Se estudiaron varias metodologías para el desarrollo de ontologías, sin embargo, no se encontró una metodología apropiada para este trabajo, por lo cual fue necesario desarrollar una metodología propia (ver sección 3.1.2). Los autores presentan sus propuestas y sus ejemplos, pero en cada caso se necesitan hacer ciertas adecuaciones. Se realizó una integración entre las metodologías que ofrecen más detalle y partiendo de esto se presentan una serie de etapas que fueron seguidas para llegar al diseño final de la ontología.
3. Se realizó un análisis de los sistemas de negociación publicados en revistas especializadas, los cuales reportan el diseño y uso de sus primitivas de negociación (ver inciso 5, sección 3.1.2). Con base en este análisis se identificaron los conceptos clave y considerando el objetivo de la ontología - servir como medio de resolución de problemas por entendimiento -, se definieron los tipos de relaciones de equivalencia entre las primitivas.
4. Se propuso una metodología para poblar la ontología mediante la cual los dueños de los agentes que manejan

lenguajes de comunicación heterogéneos y que desean realizar negociaciones entre ellos, pueden publicar sus primitivas de negociación.

5. Para probar la solución, se desarrolló una plataforma de ejecución de procesos de negociación utilizando la arquitectura orientada a servicios Web, y se incorporó un módulo de traducción que sirve como un intermediario que traduce solamente cuando se requiere, reduciendo el número de traducciones innecesarias.
6. Se realizó un análisis de todos los posibles casos de procesos de negociación entre el conjunto de agentes con lenguajes de comunicación heterogéneos, para seleccionar los más representativos. Como resultado de este análisis se desarrolló una fórmula que permite determinar para cierto conjunto de agentes de negociación heterogéneos, cuántos enlaces de comunicación tendrán problemas por falta de entendimiento (ver sección 4.2).
7. El conjunto de pruebas experimentales que se ejecutaron muestran que la solución desarrollada mejora la comunicación entre agentes de negociación heterogéneos reduciendo los casos de falla debido a falta de entendimiento, para el caso de mayor heterogeneidad hasta un 90% y aumentando el número de acuerdos.

La conclusión final de este trabajo de investigación es que la heterogeneidad en el lenguaje de comunicación entre agentes de negociación se puede resolver mediante la incorporación de una ontología, que sirva como un vocabulario compartido entre los agentes.

6.2 Trabajos futuros

Como resultado de este trabajo de investigación doctoral, se han identificado un conjunto de trabajos a futuro, con los cuales se daría continuidad a este tema. En esta sección se describen de forma breve estas propuestas.

Extender la aplicación de la ontología en otros aspectos de la negociación:

En este trabajo se desarrolló una ontología compartida para la traducción de mensajes durante el proceso de negociación. Esta ontología actualmente se emplea para la traducción de la primitiva dentro del mensaje transmitido; sin embargo, sería de mucha utilidad extender el uso de esta ontología para los siguientes casos:

- a. Una extensión inmediata del uso de esta ontología es para la parte de los parámetros del mensaje, los cuales pueden poseer heterogeneidad en el número, el formato, hasta el significado de los parámetros.
- b. Desarrollar un mecanismo para la validación de secuencias de mensajes entre agentes, y utilizarla para resolver problemas por heterogeneidad en los protocolos de negociación.

Automatización de la metodología para la integración de múltiples agentes:

En esta tesis se presenta una metodología para la integración de múltiples agentes de negociación que manejan lenguajes de

comunicación heterogéneos. Esta metodología actualmente consiste de una serie de pasos que se ejecutan de forma manual, con ayuda de algunas herramientas de soporte. Sin embargo, sería de gran interés y ayuda la automatización de esta metodología, para lo cual se sugieren los siguientes trabajos:

- a. Desarrollar un sistema para la clasificación automática de las primitivas de negociación, partiendo del análisis de las descripciones proporcionadas por sus dueños, empleando técnicas de lenguaje natural y algoritmos de clasificación.
- b. Desarrollar un sistema que identifique y genere de manera automática las relaciones entre las primitivas de negociación, considerando la clasificación a la que pertenecen y el uso pretendido de la primitiva dentro del protocolo de negociación.
- c. Desarrollar un módulo de validación sintáctica para todas las primitivas proporcionadas por los desarrolladores de los agentes.

Aplicar la metodología para la integración de múltiples agentes a otros casos:

En esta tesis se propone una metodología basada en la incorporación de una ontología, a través de la cual se favorece la interoperabilidad en la comunicación entre agentes de negociación electrónica heterogéneos. Sería de bastante interés la aplicación de esta metodología a otro tipo de casos que tengan la necesidad de integrar múltiples sistemas heterogéneos.

Referencias

- [1] Chavez A., Maes P., "Kasbah: An Agent Marketplace for Buying and Selling Goods". Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, London, UK, April 1996.
- [2] Bichler M., Kersten G., Strecker S., "Towards a Structured Design of Electronic Negotiations". Group Decision and Negotiation, Vol. 12, No. 4, pp. 311-335, 2003.
- [3] Su S., Huang C., Hammer J., Huang Y., Li H., Wang L., Liu Y., "An Internet-based negotiation server for e-commerce". The VLDB Journal 10, pp. 72-90, 2002.
- [4] Müller H. J., "Negotiation Principles, Capítulo 7, Foundations of Distributed Artificial Intelligence". G.M.P. O'Hare, and N.R. Jennings, John Wiley & Sons, 1996.
- [5] Smith R. y Davis R., "Frameworks for cooperation in distributed problem solving". Readings in Distributed Artificial Intelligence, San Mateo, CA, 1980.
- [6] Smith R. G., "The contract net protocol: High-level communication and control in a distributed problem solver". Readings in Distributed Artificial Intelligence, San Mateo, CA, 1988.
- [7] Pruitts, D. G., "Negotiation Behaviour", Academic Press, New York, 1981.
- [8] Jelassi M., Foroughi A., "Negotiation support systems: an overview of design issues and existing softwares". Decision Support Systems Vol. 5 No. 2, pp. 167-181, 1989.

- [9] Kersten G., Noronha S., "WWW-based negotiation support: design, implementation and use". *Decision Support Systems*, Vol. 25, Issue 2, pp. 135-54, 1999.
- [10] Maes, P., Guttman R. H., y Moukas, A. G., "Agents that Buy and Sell". *Communications of the ACM*, Vol. 42, pp. 81-91, 1999.
- [11] Maxsim Tsvetovatyy, Maria Gini, Bamshad Mobasher, y Zbigniew Wieckowski, "MAGMA: An agent-based virtual market for electronic commerce". *Journal of Applied Artificial Intelligence*, pp. 501-524, 1997.
- [12] Guttman, R. H., Moukas, A. G., y Maes, P., "Agent-mediated electronic commerce: a survey". *Knowledge Engineering Review*, pp. 143–152, 1998.
- [13] Rodríguez, J. A., Noriega, P., Sierra, C., y Padget, J., "A Java-based electronic auction house". *Proceedings of the Second International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*, 1997.
- [14] Wurman, Peter R., Wellman, Michael P., y Walsh William E., "The Michigan Internet AuctionBot: A configurable auction server for human and software agents". In *proceedings of the Second International Conference on Autonomous Agents*, pp. 301–308, 1998.
- [15] Moai Technologies, <http://www.moai.com>. Fecha de consulta 05 de Julio de 2006.
- [16] Perfect Commerce Inc., <http://www.perfect.com>. Fecha de consulta 05 de Julio de 2006.
- [17] Ozro Inc., <http://www.eyedeas.net/clients/ozro>. Fecha de consulta 05 de Julio de 2006.
- [18] Thiessen, E. M., Loucks, D. P., and Stedinger, J. R., "Computer-Assisted Negotiations of Water Resources Conflict", *Group Decision and Negotiation*, Vol. 7, pp. 109-129, 1998.

- [19] Kumar M., Feldman S., Business negotiation on the Internet. In Proceedings of the INET 98 Internet Summit, Geneva, 1998.
- [20] Benyoucef M., Keller R., "An Evaluation of Formalisms for Negotiations in E-Commerce". In proceedings of the Workshop on Distributed Communities on the Web, Springer LNCS 1830, Quebec, Canada, pp. 45-54, 2000.
- [21] Benyoucef M. y Keller R. K., "A Conceptual Architecture for a Combined Negotiation Support System". In proceedings of the Eleventh International Workshop on Database Expert Systems Applications, Greenwich, London, pp 1015-1019, 2000.
- [22] Benyoucef M., Bassil S., y Keller R. K., "Workflow modeling of combined negotiations in e-commerce". In Proceedings of the Fourth International Conference on Electronic Commerce Research, pp. 348-359, 2001.
- [23] Leyman F. y Roller D., "Workflow Based Applications", IBM Systems Journal, Vol. 36, No. 1, 1997.
- [24] BEA WebLogic, <http://www.bea.com>. Fecha de consulta 05 de Julio de 2006.
- [25] Finning T., Fritzson R., y McEntire R., "KQML as an agent communication language". In proceedings of the 3rd International Conference on Information and Knowledge Management, ACM Press, pp. 456-463, 1994.
- [26] FIPA, Foundation for Intelligent Physical Agents. FIPA Specifications, 2003; disponible en <http://www.fipa.org/specifications/index.html>
- [27] Uschold M., Grüninger M., "Ontologies: Principles, Methods and Applications", Knowledge Engineering Review, Vol. 11 No. 2, pp. 93-155, 1996.

- [28] Sonia V. Rueda, Alejandro J. García, Guillermo R. Simari, "Argument-based Negotiation among BDI Agents". Journal of Computer Science and technology, Vol. 2 No. 7, 2002.
- [29] Stanislav Pokraev, Zlatko Zlatev, Rogier Brusee, Pascal van Eck, "Semantic Support for Automated Negotiation with Alliances". In Proceedings of the 6th International Conference on Enterprise Information Systems, Portugal, pp. 244-249, 2004.
- [30] Haifei Li, Chunbo Huang and Stanley Y., "Design and Implementation of Business Objects for Automated Business Negotiations". Group Decision and Negotiation, Kluwer Academic Publisher, Vol. 11, pp. 23-44, 2002.
- [31] Chang M. K., and Woo C., "A Speech-Act-Based Negotiation Protocol: Design, Implementation, and Test Use". ACM Transactions on Information Systems, Vol. 12, No. 4, pp.360-382, 1994.
- [32] Crawshay-Williams, R., "Methods of Criteria of reasoning: An Inquiry into the Structure of Controversy". Editorial Routledge and Kegan Paul, London, 1957.
- [33] Naes Arne, "Interpretation and Preciseness: A Contribution to the theory of Communication". Editorial: Oslo, I kommisjon hos J. Dybwad, 1953.
- [34] Carbogim, D., Robertson, D., and Lee J., "Argument-based applications to knowledge engineering". The Knowledge Engineering Review, Vol. 5, Issue 2, pp. 119-149, 2000.
- [35] Pease A., Colton S., Smail A., Lee J., "Semantic Negotiation: Modeling Ambiguity in Dialogue". In Proceedings of Edilog 2002, the 6th Workshop on the semantics and pragmatics of dialogue, 2002.

- [36] Bartolini, C., Preist, C. & Jennings, N., "A Generic Software Framework for Automated Negotiation". HP Labs Technical Report HPL-2002-2, 2002.
- [37] Gruber T. R., "A Translation Approach to Portable Ontology Specifications". Knowledge Acquisition, Academic Press, Vol. 5, No. 2, pp. 199-220, 1993.
- [38] Guarino N., "Formal Ontologies and Information Systems". In Proceedings of the International Conference on Formal Ontology in Information Systems, IOS Press, Ámsterdam, 1998.
- [39] Dieter Fensel, Ontologies, "A Silver Bullet for Knowledge Management and Electronic Commerce", Editorial Springer, 1^{ra} edición, 2001.
- [40] Gennari J., Musen M., Fergerson R., Grosso W., Crubézy M., Eriksson H., Noy N., y Tu S., "The evolution of Protégé-2000: An environment for knowledge-based systems development". International Journal of Human-Computer Studies, Vol. 58 No. 1, pp. 89-123, 2003.
- [41] Knublauch H., Horridge M., Musen M., Rector A., et-al, "The Protégé OWL Experience". In proceedings of the Workshop on OWL: Experiences and Directions, Galway, Ireland, 2005.
- [42] Fernández M., Gómez Pérez A., Juristo N., "METHONTOLOGY: From Ontological Art Towards Ontological Engineering". Proceedings of the Symposium on Ontological Engineering of AAI. California, 1997.
- [43] Paolucci M., Kawamura T., Payne T., Sycara K., "Semantic Matching of Web Services Capabilities". In Proceedings of the 1st International Semantic Web Conference (ISWC), 2002.
- [44] Ramos-Quintana F., Frausto Solís J., Camargo-Santacruz F., "A Methodology for Modeling Interactions in Cooperative Information Systems Using Coloured Petri Nets". International

Journal of Software Engineering and Knowledge Engineering
(IJSEKE), Vol. 12, No. 6, pp. 619-636, 2002.

[45] Mendenhall, Scheaffer, Wackerly, "Estadística Matemática con
Aplicaciones", Grupo Editorial Iberoamérica

Anexo A. Lenguajes de los agentes

Combinaciones por pares de agentes que manejan lenguajes de comunicación heterogéneos.

CASO	AGENTES	LENG 1	LENG 2	PRIMITIVAS IGUALES
1	b, a	CFP Propose Accept Terminate Reject Acknowledge Modify Withdraw	Initial_offer RFQ Accept Reject Offer Counter-offer	2
2	c, a	accept-offer what-is-price what-is-item add-sell-agent add-buy-agent add-potential-customers add-potential-sellers agent-terminated deal-made	Initial_offer RFQ Accept Reject Offer Counter-offer	1
3	d, a	Requests_Add Authorize_Add Require Demand Accept Reject Unable Require-for Insist_for Demand_for	Initial_offer RFQ Accept Reject Offer Counter-offer	2
4	e, a	Call for proposal Propose proposal Reject proposal Withdraw proposal Accept proposal Modify proposal Acknowledge message Terminate negotiation	Initial_offer RFQ Accept Reject Offer Counter-offer	2
5	f, a	request-quotation give-quotation order delivered paid	Initial_offer RFQ Accept Reject Offer Counter-offer	0

6	g, a	Accept Proposal Agree Cancel Call for Proposal Confirm Disconfirm Failure Inform Inform If Inform Ref Not Understood Propagate Propose Proxy Query If Query Ref Refuse Reject Proposal Request Request When Request Whenever Subscribe	Initial_offer RFQ Accept Reject Offer Counter-offer	2
7	h, a	Propose Arrange Request Inform Query Command Inspect Answer Refine Modify Change Bid Send Reply Refuse Explain Confirm Promise Commit Accept Reject Grant Agree	Initial_offer RFQ Accept Reject Offer Counter-offer	2
8	c, b	accept-offer what-is-price what-is-item add-sell-agent add-buy-agent add-potential- customers add-potential-sellers agent-terminated deal-made	CFP Propose Accept Terminate Reject Acknowledge Modify Withdraw	1
9	d, b	Requests_Add Authorize_Add Require Demand Accept Reject Unable Require-for Insist_for Demand_for	CFP Propose Accept Terminate Reject Acknowledge Modify Withdraw	2
10	e, b	Call for proposal Propose proposal Reject proposal Withdraw proposal Accept proposal Modify proposal Acknowledge message Terminate negotiation	CFP Propose Reject Withdraw Accept Modify Acknowledge Terminate	0

11	f, b	request-quotation give-quotation order delivered paid	CFP Propose Accept Terminate Reject Acknowledge Modify Withdraw	0
12	g, b	Accept Proposal Agree Cancel Call for Proposal Confirm Disconfirm Failure Inform Inform If Inform Ref Not Understood Propagate Propose Proxy Query If Query Ref Refuse Reject Proposal Request Request When Request Whenever Subscribe	CFP Propose Accept Terminate Reject Acknowledge Modify Withdraw	3
13	h, b	Propose Arrange Request Inform Query Command Inspect Answer Refine Modify Change Bid Send Reply Refuse Explain Confirm Promise Commit Accept Reject Grant Agree	CFP Propose Accept Terminate Reject Acknowledge Modify Withdraw	2
14	d, c	Requests_Add Authorize_Add Require Demand Accept Reject Unable Require-for Insist_for Demand_for	accept-offer what-is-price what-is-item add-sell-agent add-buy-agent add-potential- customers add-potential-sellers agent-terminated deal-made	1
15	e, c	Call for proposal Propose proposal Reject proposal Withdraw proposal Accept proposal Modify proposal Acknowledge message Terminate negotiation	accept-offer what-is-price what-is-item add-sell-agent add-buy-agent add-potential- customers add-potential-sellers agent-terminated deal-made	0

16	f, c	request-quotation give-quotation order delivered paid	accept-offer what-is-price what-is-item add-sell-agent add-buy-agent add-potential- customers add-potential-sellers agent-terminated deal-made	0
17	g, c	Accept Proposal Agree Cancel Call for Proposal Confirm Disconfirm Failure Inform Inform If Inform Ref Not Understood Propagate Propose Proxy Query If Query Ref Refuse Reject Proposal Request Request When Request Whenever Subscribe	accept-offer what-is-price what-is-item add-sell-agent add-buy-agent add-potential- customers add-potential-sellers agent-terminated deal-made	1
18	h, c	Propose Arrange Request Inform Query Command Inspect Answer Refine Modify Change Bid Send Reply Refuse Explain Confirm Promise Commit Accept Reject Grant Agree	accept-offer what-is-price what-is-item add-sell-agent add-buy-agent add-potential- customers add-potential-sellers agent-terminated deal-made	1
19	e, d	Call for proposal Propose proposal Reject proposal Withdraw proposal Accept proposal Modify proposal Acknowledge message Terminate negotiation	Requests_Add Authorize_Add Require Demand Accept Reject Unable Require-for Insist_for Demand_for	2

20	f, d	request-quotation give-quotation order delivered paid	Requests_Add Authorize_Add Require Demand Accept Reject Unable Require-for Insist_for Demand_for	1
21	g, d	Accept Proposal Agree Cancel Call for Proposal Confirm Disconfirm Failure Inform Inform If Inform Ref Not Understood Propagate Propose Proxy Query If Query Ref Refuse Reject Proposal Request Request When Request Whenever Subscribe	Requests_Add Authorize_Add Require Demand Accept Reject Unable Require-for Insist_for Demand_for	3
22	h, d	Propose Arrange Request Inform Query Command Inspect Answer Refine Modify Change Bid Send Reply Refuse Explain Confirm Promise Commit Accept Reject Grant Agree	Requests_Add Authorize_Add Require Demand Accept Reject Unable Require-for Insist_for Demand_for	2
23	f, e	request-quotation give-quotation order delivered paid	Call for proposal Propose proposal Reject proposal Withdraw proposal Accept proposal Modify proposal Acknowledge message Terminate negotiation	0

24	g, e	Call for Proposal Propose Reject Proposal Accept Proposal Terminate negotiation Disconfirm Inform Agree Cancel Confirm Failure Inform If Inform Ref Not Understood Propagate Proxy Query If Query Ref Refuse Request Request When Request Whenever Subscribe	Call for proposal Propose proposal Reject proposal Accept proposal Terminate negotiation Modify proposal Acknowledge message Withdraw proposal	4
25	h, e	Propose Arrange Request Inform Query Command Inspect Answer Refine Modify Change Bid Send Reply Refuse Explain Confirm Promise Commit Accept Reject Grant Agree	Call for proposal Propose proposal Reject proposal Withdraw proposal Accept proposal Modify proposal Acknowledge message Terminate negotiation	4
26	g, f	Accept Proposal Agree Cancel Call for Proposal Confirm Disconfirm Failure Inform Inform If Inform Ref Not Understood Propagate Propose Proxy Query If Query Ref Refuse Reject Proposal Request Request When Request Whenever Subscribe	request-quotation give-quotation order delivered paid	1

27	h, f	Propose Arrange Request Inform Query Command Inspect Answer Refine Modify Change Bid Send Reply Refuse Explain Confirm Promise Commit Accept Reject Grant Agree	request-quotation give-quotation order delivered paid	0
28	h, g	Propose Arrange Request Inform Query Command Inspect Answer Refine Modify Change Bid Send Reply Refuse Explain Confirm Promise Commit Accept Reject Grant Agree	Accept Proposal Agree Cancel Call for Proposal Confirm Disconfirm Failure Inform Inform If Inform Ref Not Understood Propagate Propose Proxy Query If Query Ref Refuse Reject Proposal Request Request When Request Whenever Subscribe	9

Anexo B. Resultados de la investigación

Como resultado de este trabajo de investigación, se publicó el siguiente artículo:

1. "Design of a Shared Ontology Used for Translating Negotiation Primitives". Lecture Notes in Computer Science LNCS-3983, SPRINGER VERLAG, 2006.

Otros artículos publicados con los resultados de este trabajo de investigación son los siguientes:

2. "Ontology Support for Communicating Agents in Negotiation Processes". IEEE Computer Society Conference Proceedings of Fifth International Conference on Hybrid Intelligent Systems, Rio de Janeiro, Brazil, November 6-9, 2005.
3. "An Ontology-based Translator for Communicating Negotiation Agents over Internet", IEEE Looking Forward, 2006.

Se codirigió una tesis de maestría para apoyar este trabajo de investigación.

"Arquitectura de un Mercado Electrónico Utilizando Servicios Web", por el alumno Arturo Pérez Cebreros. Tesis defendida en agosto de 2005.

cenidet

Centro Nacional de Investigación y Desarrollo Tecnológico
Departamento de Ciencias Computacionales

TESIS DOCTORAL

Ontología Compartida de Apoyo en la Comunicación entre Agentes de Negociación que Usan Lenguajes Heterogéneos

presentada por

Maricela Claudia Bravo Contreras

M. en C. en Ciencias de la Computación por el CENIDET

como requisito para la obtención del grado de:

Doctor en Ciencias en Ciencias de la Computación

Director de tesis:

Dr. Joaquín Pérez Ortega

Co-Director de tesis:

Dr. Rodolfo A. Pazos Rangel

Jurado:

Dr. Gerardo Reyes Salgado
Presidente

Dr. Joaquín Pérez Ortega
Secretario

Dr. Juan Frausto Solís
Primer Vocal

Dra. Laura Cruz Reyes
Segundo Vocal

Resumen

Actualmente una de las limitaciones para el comercio electrónico, es el problema de la automatización de la negociación, y en particular el entendimiento entre agentes de negociación que utilizan diferentes lenguajes para comunicarse. En esta tesis se aborda el problema de comunicación entre agentes de negociación que usan lenguajes heterogéneos. Este problema ocurre cuando varios agentes que participan en un proceso de negociación, intercambian mensajes derivados de lenguajes diferentes, los cuales contienen conjuntos de primitivas implícitamente codificadas.

Se presenta una solución al problema, la cual consiste en el desarrollo de una ontología compartida, en la cual se publican y se describen las primitivas de negociación que son utilizadas por los agentes durante la comunicación de forma explícita, utilizando un lenguaje ontológico. Un beneficio de este enfoque de solución es que la traducción de las primitivas de los agentes, solamente se efectúa cuando es requerida, reduciendo el número de traducciones innecesarias. Dicha solución está orientada para ejecutarse en un ambiente abierto de Internet, y consta de una arquitectura basada en servicios Web dentro de la cual se ejecutan los procesos de negociación electrónica entre agentes.

El enfoque de solución se validó con un conjunto de pruebas experimentales que incorporan un grupo de agentes de negociación que usan lenguajes heterogéneos. Los resultados de las pruebas realizadas muestran que la solución desarrollada mejora la comunicación entre agentes de negociación reduciendo hasta un 90% los casos de falla debido a falta de entendimiento, para el caso de mayor heterogeneidad.