

MARCO DE REFERENCIA PARA NEGOCIACIÓN Y MONITOREO DE ACUERDOS
DE NIVEL DE SERVICIO SLA, EN AMBIENTES CLOUD

LUIS FERNANDO GOMEZ ORTEGA.

UNIVERSIDAD DEL VALLE.
FACULTAD DE INGENIERÍAS
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
SANTIAGO DE CALI

2016

MARCO DE REFERENCIA PARA NEGOCIACIÓN Y MONITOREO DE ACUERDOS
DE NIVEL DE SERVICIO SLA, EN AMBIENTES CLOUD

LUIS FERNANDO GÓMEZ ORTEGA.

DIRECTOR:

Ph.D. JOHN ALEXANDER SANABRIA

UNIVERSIDAD DEL VALLE.
FACULTAD DE INGENIERÍAS
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
SANTIAGO DE CALI

2016

Contenido

1.	Resumen.....	13
2.	Problema.....	15
2.1.	Definición del problema	15
2.2.	Evidencia	15
2.3.	Antecedentes.....	16
2.4.	Justificación.	17
3.	Estado del arte	19
3.1.	Computación <i>Grid</i>	19
3.1.1.	Elementos fundamentales de las infraestructuras <i>grid</i>	20
3.1.2.	<i>Grid</i> semántica.....	23
3.2.	Computación <i>Cloud</i>	24
3.2.1.	Servicios <i>Cloud</i>	26
3.3.	Acuerdo de nivel de servicio SLA	29
3.3.1.	Ciclo de vida de un SLA.	30
3.3.2.	SLA En computación <i>Cloud</i>	33
3.4.	Proceso de negociación.....	35
3.4.1.	Partes del proceso de negociación.	35
4.	Investigación	39
4.1.	Objetivos.....	39
4.1.1.	Objetivo general.	39
4.1.2.	Objetivos específicos.....	39
4.2.	Resultados esperados	39
4.3.	Metodología	40
4.3.1.	Alcance del trabajo de grado.	40

4.3.2.	Actividades	43
5.	Actores involucrados en el proceso.....	44
6.	Escenarios de interacción entre actores.....	47
6.1.	Escenario de interacción en la oferta de servicios IaaS y SaaS actual.....	47
6.2.	Escenarios emergentes	52
7.	Caracterización de los servicios IaaS y SaaS.....	54
7.1.	Caracterización de servicios IaaS	54
7.1.1.	Selección de los proveedores IaaS.....	54
7.1.2.	Comparación de servicios de los proveedores IaaS.....	57
7.1.3.	Comparación de documentos SLA IaaS.....	62
7.2.	Caracterización de los servicios SaaS.....	65
7.2.1.	Selección de los proveedores SaaS.....	65
7.2.2.	Comparación de servicios de los proveedores SaaS.....	65
7.2.3.	<i>Office 365</i>	67
7.2.4.	Comparación de documentos SLA proveedores SaaS.	69
8.	Desarrollo de ontologías para los dominios IaaS y SaaS.....	80
8.1.	Selección de la metodología	80
8.2.	Desarrollo de ontologías para servicios IaaS	85
8.3.	Desarrollo de ontología para servicios SaaS.....	90
9.	Marco de negociación.	92
9.1.	Definición del espacio de negociación	92
9.2.	Protocolos de negociación relacionados.	97
9.3.	Marco de negociación propuesto.	101
10.	Resultados del trabajo:	107
11.	Conclusiones y trabajo futuro.....	112

11.1.	Conclusiones.	112
11.2.	Trabajo futuro.....	119
12.	Referencias.	120

Figuras

Figura 1. Elementos principales de los entornos <i>grid</i> , según (Foster, Kesselman, & Tuecke, 2001).....	21
Figura 2. Pirámide de servicios <i>cloud</i> adaptada de (Youseff et al., 2008).	26
Figura 3. Pirámide servicios <i>cloud</i> , en dos configuraciones comunes.	27
Figura 4: Esquema de negociación directa entre usuario final y proveedor de servicios.....	36
Figura 5: Esquema de negociación entre usuario final y proveedor por medio de un bróker.	36
Figura 6: Esquema de negociación entre usuario final y proveedores por medio de múltiples bróker.....	37
Figura 7: Mercado de infraestructura para el comercio de servicios <i>cloud</i> . Tomado de (Buyya et al., 2008)	38
Figura 8: Escenario de interacción consumidor, proveedor, auditor tomado de (Liu et al., 2013).....	38
Figura 9. Entidades involucradas en la investigación.....	41
Figura 10. Ciclo de negociaciones de SLA	42
Figura 11. Panel de filtros usado por http://www.intelcloudfinder.com en su configuración de búsqueda rápida. Fuente: tomado de http://www.intelcloudfinder.com/quicksearch agosto de 2013.....	49
Figura 12. Panel de filtros usado por el servicio https://www.computenext.com/ Fuente: tomado de https://www.computenext.com/ agosto de 2013.....	49
Figura 13. Panel para búsqueda de servicios SaaS para Pymes. Tomado de http://www.eoi.es/saassometro/web/inicio/seleccionArea visitada 24 de agosto de 2013 ...	50
Figura 14. Filtros aplicados en www.findthebest.com fuente: Tomado de http://cloud-computing.findthebest.com/ Agosto de 2013.	51
Figura 15. Filtros aplicados en https://www.saasmax.com fuente: Tomado de https://www.saasmax.com Agosto de 2013.....	52
Figura 16. Características generales del servicio.....	66
Figura 17. Características generales del servicio <i>Office 365 Small Business</i>	67
Figura 18. Ontología para la definición del dominio SLA para servicios IaaS.....	89

Figura 19. Ontología para la definición del dominio SLA para servicios SaaS.....	91
Figura 20. Elementos del espacio de negociación.....	93
Figura 21. Marco general de negociación. Fuente Kijthaweesinpoon & Zhou, 2008.....	98
Figura 22. Conflict-Aware Negotiation Protocol Fuente: Netto 2012	99
Figura 23. Esquema del marco de negociación propuesto.	104
Figura 24. Escenario de intersección de ofertas.	105
Figura 25. Escenario de disjunto de ofertas.....	105

Tablas.

Tabla 1: Conciliación entre el ciclo de vida de SLA 3 fases y ciclo de vida de SLA de 6 fases.	31
Tabla 2. Actividades del proyecto.	43
Tabla 3. Comportamiento del cuadrante mágico de Gartner.	56
Tabla 4. Instancias estándar de <i>Amazon</i>	57
Tabla 5. Ejemplo de máquinas virtuales <i>Azure</i>	58
Tabla 6. Ejemplo de instancias <i>Google Compute Engine</i>	59
Tabla 7. Ejemplo de Servicio IaaS de <i>GoGrid</i>	59
Tabla 8. Servicio IaaS de <i>Rackspace</i>	60
Tabla 9. Comparación de periodos de facturación entre proveedores.	61
Tabla 10. Penalidad por tiempo de actividad de conectividad mensual.	63
Tabla 11. Penalidad por tiempo de actividad de instancia de rol.	63
Tabla 12. Penalidades por incumplimiento de disponibilidad.	64
Tabla 13. Penalidades por tiempo de inactividad.	73
Tabla 14. Días añadidos por porcentaje de actividad mensual.	78
Tabla 15. Elementos del protocolo de negociación bilateral multipasos presentado por...	100

Dedicatoria.

Quiero dedicar este trabajo como es natural a mi familia.

Mi padre Fernando Gómez Giraldo.

Mi madre Dora Inés Ortega.

Mi abuela Dioselina Cruz.

Mis hermanos.

Wilson Gómez.

Andrés Gómez.

Valentina Gómez.

Mis sobrinos.

Ian Daniel Gómez.

Alan Gómez.

A los demás familiares y amigos que me apoyaron para conseguir este gran logro, infinitas gracias.

También quiero dedicar este trabajo a una persona sin la cual no hubiera sido posible salir de muchos momentos difíciles.

Marta Cecilia Sandino Rodríguez.

Mil gracias por enseñarme tanto y por tus sabios consejos.

Agradecimientos.

Habiendo todos los días tanto para agradecer, este día quiero agradecer a Dios por la oportunidad que me da de nombrar algunas personas sin las cuales este logro no hubiera sido posible, a las cuales les agradezco profundamente haber creído en mí y permitirme ser la persona que soy hoy.

A mis Abuelos, padres, hermanos, sobrinos y demás familiares.

Gracias, sin ustedes esto no hubiera sido posible ni siquiera en sueños.

A mi mentor, amigo y hermano.

Leonardo Yunda.

Gracias por todas las oportunidades.

A una gran mujer.

Martha Lucía Duque Ramírez.

Gracias por tu amistad, apoyo y por creer en mí.

A mis hermanos de corazón.

Edward Fernando Mosquera, Andrés Uzuga Naranjo, Walther Trujillo, Geiner Añasco, Gloria Mayence Añasco y Douglas Niño.

Gracias son los mejores.

A la familia Ostos Ortiz.

Olga Lucia Ostos, Don Jorge Ostos, Doña Laude Ortiz.

Gracias por ser ángeles que Dios puso en mi camino, por su amistad sincera y por ser mi familia cuando estuve lejos de casa.

A un Ángel.

Martha Cecilia Sandino.

Gracias por tanto.

A los del clan.

Victoria Saavedra, Yelitza Palacio, Julián David Villegas, Billy Paolo Pórtela, Oscar Guevara, Jesús Antonio Escobar, Carlos Escobar Marmolejo y Jorge Parra.

Por haberme brindado su amistad y todos los buenos momentos.

A mis amigos de la Universidad Santiago de Cali.

Claudia Liliana Zúñiga, Steve Rodríguez Guerrero y Julián Hernández.

Gracias por su gran amistad.

A los que me encontré después.

Carlos Ramírez, Simena Dinas y Oscar Ceballos.

Gracias por enseñarme tanto.

A quienes me dieron la oportunidad.

Leonardo Yunda, Marco Rodríguez, Andrés Alarcón y Andrés Millán.

Gracias por creer en mí.

A mis profesores.

En especial mi director John Alexander Sanabria.

Gracias Maestro, sin tu ayuda y apoyo esto no hubiera sido posible.

A una persona muy especial.

Nelcy Bastidas.

Gracias por todo, por tu preocupación y por tu cariño.

A un colega.

Ideen Santana Pérez.

Gracias por tu apoyo en la escritura de artículos y en la parte semántica.

A una gran profesional y amiga.

Myriam Leonor Torres Pérez.

Gracias por haberme dado el tiempo para terminar esta última etapa.

A mis compañeros de trabajo de la Escuela de Ciencias de la Salud de la Universidad Nacional Abierta y a Distancia.

Gracias por todo.

A todos los anteriores infinitas gracias por sus contribuciones directas e indirectas en el desarrollo de esta investigación, Dios los bendiga siempre.

1. Resumen.

La computación distribuida ha sido explotada bajo diferentes paradigmas desde hace algún tiempo, en la actualidad la computación *grid* y *cloud*, desde sus respectivos puntos de vista son los ambientes predominantes en las infraestructuras distribuidas de computación. Las infraestructuras *grid* presentan un comportamiento comunitario que promueve la creación de organizaciones virtuales de usuarios, que comparten recursos computacionales, para conseguir una meta común. Las organizaciones virtuales están compuestas por diferentes actores como personas, empresas, instituciones y recursos computacionales que se encuentran distribuidos a través de múltiples dominios administrativos.

Los acuerdos de nivel de servicio (SLA) por sus siglas en español son difíciles de cumplir en entornos *grid* por su perspectiva no comercial, pues ninguna de las partes está obligada a cumplir con un contrato para la satisfacción de usuarios, sin embargo es deseable tener acuerdos mínimos que permitan estimar por ejemplo, si una computación lanzada al *grid* va a tomar días o meses en ejecución, de acuerdo con la cantidad de recursos disponibles para dicha computación.

Las infraestructuras *cloud* por su parte, pueden ser tomadas como un aspecto evolutivo de la computación *grid* y proveen diferentes tipos de servicios bajo demanda a través de un modelo económico de pago por los recursos que se consumen. La característica de pagar por lo que se utiliza, implica que en un momento determinado se tengan que aprovisionar recursos suficientes para atender picos de demanda de un usuario. Estos altos volúmenes de demanda pueden oscilar desde algunos minutos hasta días incluso. Una vez se disminuye la demanda de los recursos, el excedente de los mismos es contraído o liberado.

La naturaleza comercial y fluctuante del *cloud* hace que requiera mecanismos complejos para asegurar que el cliente pague por lo que realmente consume y que los recursos puedan ser aprovisionados y suspendidos de manera automática y en tiempo real.

Desde la perspectiva de *cloud*, los SLA son contratos entre clientes y proveedores de servicios, cuyo fin es establecer un mínimo de condiciones esperadas por el cliente que el proveedor debe cumplir para satisfacer sus necesidades. El proveedor de servicios está obligado a responder por los compromisos consignados en el documento SLA, de ocurrir lo contrario puede ser penalizado.

La presente propuesta de investigación, propone un marco de referencia que permite mejorar el proceso de negociación en ambientes *cloud*, para lo cual se definió un conjunto de actores basados en agentes, los cuales se pueden enmarcar fácilmente en cualquier proceso de negociación.

Por otra parte se desarrollaron mecanismos semánticos para la comunicación de los agentes que representan a los actores. Para el desarrollo de dichos mecanismos se crearon ontologías que definen los dominios clave del proceso de negociación. Para este trabajo los dominios más importantes son el dominio de servicios PaaS, dominio de servicios SaaS y el dominio SLA. Para poder llegar a los conceptos base de las ontologías se serializaron caracterizaciones que también se ilustran en el trabajo.

Por último los aspectos antes mencionados se incluyen en un protocolo de negociación de recursos *cloud*, que se puede ajustar a subastas y dinámica de libre mercado. Cada una de las etapas del trabajo encuentra justificación teórica y también se soporta en elementos implementados actualmente en la industria de informática de servicios *cloud*.

2. Problema

2.1. Definición del problema

Los documentos SLA que son utilizados por proveedores *cloud*, no incorporan características que faciliten su negociación, y monitoreo en tiempo de ejecución.

2.2. Evidencia

La computación *cloud* puede ser considerada como una evolución de *grid*, para prestar servicios de forma pública, la cual tiene como características que se paga por lo que se consume y está abierta al público en general con la única restricción de realizar el pago correspondiente.

Por la naturaleza utilitaria de la computación *cloud*, los proveedores SaaS, PaaS, IaaS y usuarios finales, deben establecer bases sobre las cuales se aprovisiona el servicio. Dichas bases son establecidas en un documento llamado SLA, el cual ilustra claramente el servicio que ofrece el proveedor para cumplir los requerimientos manifestados por el cliente.

Por definición los recursos en *cloud* tienen la flexibilidad de crecer o decrecer en demanda y tiempo real, lo que hace indispensable que los SLA sean monitoreados para que se cumpla a cabalidad lo que se define en ellos. Sin embargo es una tarea difícil de realizar dado que el monitoreo y evaluación debe hacerse en tiempo real de ejecución, lo que implica, que se requiera de una plataforma dinámica que interopere fácilmente con diferentes proveedores *cloud*.

Los proveedores de software como servicio SaaS dependen completamente de *cloud* para su negocio, en dicho sentido se requiere que los SLA sean flexibles y se puedan ajustar a diversos tipos de requerimiento en el tiempo, en contraste con lo que pasa con proveedores SaaS muchas compañías están utilizando computación *cloud* principalmente para rebajar

costos en la ejecución de sistemas que no son críticos en su operación (More, 2009)

Por otra parte (Wu & Buyya, 2010) cita que el 85% de los proveedores de IaaS no contemplan acciones legales para el incumplimiento de SLA. Lo que va en decremento de la confianza que los consumidores pueden tener en ellos, si se tiene en cuenta que los proveedores SaaS en muchos casos deben cubrir penalidades por el no cumplimiento de servicio según lo pactado en los SLA.

Para los usuarios de servicios IaaS, no es fácil encontrar factores diferenciadores entre muchos de los proveedores de ese tipo de servicios, debido a que su oferta es muy similar entre sí. La calidad del servicio (QoS) por sus siglas en español y en especial el cumplimiento de los SLA deben ser entonces elementos discriminatorios para la elección de un proveedor sobre otro (Bouchenak, 2010).

De acuerdo con lo expresado se evidencia que los SLA son un factor fundamental en *cloud* debido a su naturaleza comercial. Los temas de monitoreo y negociación, han sido tratados en trabajos relacionados que serán nombrados en el numeral 2.3 antecedentes. Sin embargo el enfoque de este trabajo difiere de los demás en que se propone un marco de referencia que se enfoca en la negociación y monitoreo de documentos SLA entre proveedores SaaS, IaaS y usuarios como un conjunto y no como entidades aisladas.

2.3. Antecedentes.

El trabajo con SLA, ha tenido una importante aplicación en plataformas basadas en *cloud computing*, gracias a los acuerdos que se pueden alcanzar entre proveedor y usuario a la hora de contratar un servicio. De otro lado, en ecosistemas *grid* también se mostró interés en trabajar acuerdos de nivel de servicio, sin embargo solo hasta hace algunos años se empezó a observar la intención clara de incorporar SLA como mecanismo para pactar la provisión de recursos computacionales a los usuarios de estos ambientes. Esta incorporación tardía se debe en parte a que las plataformas *grid* no siguen un modelo

económico al momento de proveer recursos a los usuarios. A continuación se nombran algunos trabajos relacionados con SLA en entornos *grid* y *cloud*.

(Dumitrescu & Foster, 2005) presenta el problema del acceso a recursos compartidos de almacenamiento y computaciones, en infraestructuras *grid* a través de SLA basados en Maui y WS-Agreement con GRUBER, el cual es un mediador que provee agentes.

Por otra parte (Dobson & Sanchez-Macian, 2006) proponen la definición de una ontología que encierre los términos tanto de QoS como de SLA, de modo que se obtengan beneficios en selección de servicios, monitoreo de violaciones a los SLA y adaptación QoS.

(Padgett, 2005) muestra una arquitectura destinada a la gestión de SLA que aborda negociación y reserva de recursos, como también monitoreo del cumplimiento del SLA mediante un motor de decisiones basado en reglas.

(Macías & Guitart, 2010) muestra el uso de documentos SLA como herramientas para aumentar los ingresos de proveedores *cloud*, por medio de funciones que le permiten calcular las penalidades y el máximo beneficio, incluso dejando de cumplir algunos SLA.

2.4. Justificación.

El desarrollo de la presente propuesta busca aportar en la investigación sobre acuerdos de nivel de servicios en computación *cloud*, tanto entre usuarios finales y proveedores SaaS, como entre proveedores IaaS y proveedores SaaS. Este trabajo busca presentar un marco de referencia que permita la automatización del proceso de negociación y monitoreo de lo pactado en los SLA.

Desde lo económico es una realidad que el negocio que gira en torno a las infraestructuras *cloud* crece rápidamente todos los días y cada vez más se encuentran proveedores de servicios IaaS, SaaS y PaaS. Los proveedores de dichos servicios están interesados en

maximizar sus ganancias, en dicho sentido el conservar y hacer nuevos clientes se hace muy importante. Por ejemplo si un proveedor incumple lo pactado con un cliente, este estará insatisfecho y no volverá pero aún más grave es el hecho de que el proveedor puede hacerse de mala fama que puede repercutir en sus ganancias, en el lado opuesto un proveedor que cumple los compromisos adquiridos con sus clientes puede ver potenciadas sus ganancias por publicidad positiva.

Del lado de los consumidores de servicios *cloud* también se requiere maximizar el beneficio, para estos es importante que los proveedores cumplan con los acuerdos de nivel de servicios para tener los resultados que esperan en el tiempo estimado. Así mismo, el cumplimiento de los contratos por parte de los proveedores genera confianza y satisfacción en los clientes, quienes se ven beneficiados de la buena prestación de servicios.

Debido al poco tiempo de desarrollo del mercado de los servicios *cloud* es importante enfocar esfuerzos para mejorar sus debilidades. El desarrollo del presente trabajo puede contribuir al mejor entendimiento de este tipo de entornos y de los desafíos que plantea en el largo plazo.

3. Estado del arte

El siguiente estado del arte comprende una revisión bibliográfica de temas afines con los entornos *cloud*. En primera instancia se presenta la influencia de la computación *grid* en el nacimiento y desarrollado de la computación *cloud*. En la sección 3.1.1 se presentan algunos conceptos arquitecturales de las plataformas *grid*, así como también algunos trabajos alrededor de la *grid* semántica en la sección 3.1.2. Seguidamente en la sección 3.2. Se presentan algunos conceptos alrededor de *cloud* y los diferentes “-as-a-Service” planteados al momento en ambientes *cloud*. La sección 3.3. Se hace un recorrido por los acuerdos de nivel de servicio SLA por sus siglas en español. Por ultimo en la sección 3.4. Se hace un recorrido por el proceso de negociación.

3.1. Computación *Grid*

Sin lugar a dudas la computación *grid*, tiene un papel fundamental en la evolución y desarrollo de la computación *cloud*, muchos de los problemas técnicos que enfrentan quizá son iguales en ambos, y si se tiene en cuenta el corto tiempo de madures de la computación *cloud* muchos de los servicios que esta presta actualmente son posibles por tener como columna vertebral herramientas desarrolladas en el auge de la computación *grid* (Foster, Zhao, Raicu, & Lu, 2008a).

Los ecosistemas *grid* se caracterizan por ser ambientes computacionales a gran escala que orquestan plataformas de hardware y software con el propósito de atender las necesidades de almacenamiento y procesamiento de problemas computacionalmente intensivos. A través de servicios centrados en Internet. *grid* se considera como una evolución de los sistemas distribuidos tradicionales en el cual se promueve el uso compartido de plataformas tecnológicas, de manera que varias organizaciones pueden participar y beneficiarse de ella, a diferencia de los sistemas distribuidos donde la infraestructura suele ser de un solo propietario.(Castro, 2006).

La computación *grid* es muy similar la computación *cloud*, sin embargo se diferencian en que el uso de los recursos *cloud* no se restringe a los miembros que conforman una organización virtual, sino a cualquier entidad o persona que pueda pagar por el acceso a sus servicios principalmente de almacenamiento y computo.

El desarrollo de la computación *grid*, ha sido impulsado por el gran avance de las ciencias, que requieren cada vez mayores recursos computacionales para poder ejecutar tareas complejas. En un principio su utilización se enfocó en problemas científicos como cálculos complejos, investigación en física, astronomía y modelos climáticos, sin embargo con el tiempo se ha convertido en foco de interés para el sector empresarial y gubernamental. Influenciando el crecimiento y desarrollo de infraestructuras *cloud* que presentan servicios similares con un nivel de cobertura más alto. (Jimenez, 2007), (Folling, Grimme, Lepping, & Papaspyrou, 2010).

3.1.1. Elementos fundamentales de las infraestructuras *grid*.

Según (Foster, Kesselman, & Tuecke, 2001) el problema que cubre *grid* es “compartir recursos coordinadamente y resolver problemas en organizaciones virtuales multi-institucionales”. Bajo esa definición del problema se pueden apreciar dos conceptos básicos organizaciones virtuales y recursos compartidos, cuando las organizaciones virtuales comparten recursos es importante tener en cuenta aspectos como la interoperabilidad y seguridad.

En la Figura 1, se muestran los elementos básicos encontrados en la consideración de Foster, donde destacan las organizaciones virtuales, los recursos compartidos e implícitamente las arquitecturas.

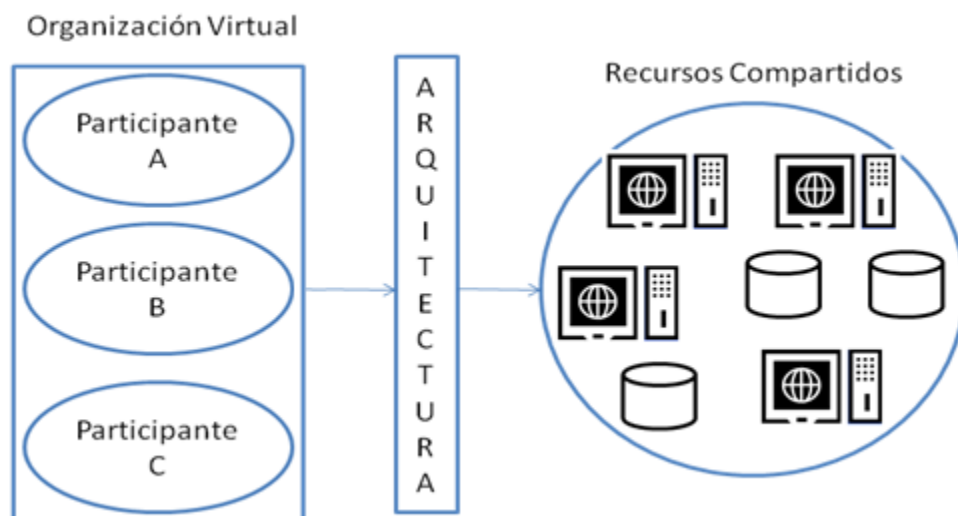


Figura 1. Elementos principales de los entornos *grid*, según (Foster, Kesselman, & Tuecke, 2001)

Organizaciones Virtuales.

Las organizaciones virtuales de usuarios (VO), están conformadas por individuos, instituciones y/o grupos que definen condiciones, reglas y políticas para compartir recursos. Cada miembro de la VO, comparte recursos propios y tiene acceso a los recursos de los demás miembros en demanda (Foster et al., 2001). Por otra parte (Denemark, Jankowski, Meyer, Ruda, & Wolniewicz, 2005) plantean que los participantes de una organización virtual pueden organizarse de forma jerárquica, con el fin de poder manejar usuarios.

La capacidad de tener organizaciones que comparten recursos hace que la computación en malla sea flexible, sin embargo compartir recursos no siempre es considerado una ventaja si se tiene en cuenta que se pueden compartir recursos con un potencial competidor. En dicho sentido se hacen importantes las arquitecturas donde se abordan temas de seguridad, control de acceso y calidad de servicio entre otras cosas.

Recursos compartidos.

Los recursos compartidos en entornos *grid* pueden ser de diferentes tipos como procesamiento, almacenamiento, software, documentos, etc. Dichos recursos generalmente

son heterogéneos y suelen estar dispersos en diferentes ubicaciones geográficas lo cual requiere la interconexión por medio de redes de alta velocidad. (Talia, 2002)

Arquitectura.

(Chervenak, Foster, Kesselman, Salisbury, & Tuecke, 2000) propone un modelo de arquitectura basado en 4 principios básicos que son:

1. Mecanismos de neutralidad.
2. Políticas de neutralidad.
3. Compatibilidad con infraestructura *grid*.
4. Información de infraestructura uniforme.

Tomado como punto de partida los principios expuestos anteriormente el autor expone una arquitectura de dos niveles dividida en componentes de alto nivel y un conjunto de servicios los cuales a su vez pueden ser genéricos o particulares.

Por otra parte con el trabajo realizado en infraestructuras *grid* se han desarrollado varios entornos de software para su aprovechamiento como CONDOR¹, UNICORE², LEGION³, GLOBUS⁴, entre otros. La variedad de entornos sugiere también variedad en las arquitecturas utilizadas. En cierto modo todos los desarrollos resuelven los problemas que ataca *grid* desde la perspectiva de Foster. La arquitectura abierta de servicios *GRID* OGSA es un referente en la búsqueda de estandarización de *grid* en diferentes aspectos. (Talia, 2002), (Joseph, Ernest, & Fellenstein, 2004), (Foster et al., 2001).

OGSA es una arquitectura abierta orientada a servicios, que atiende a la necesidad de estandarización y definición de un conjunto de capacidades y comportamientos que se enfocan en problemas clave de la computación *grid*, OGSA presenta un modelo de capas

¹ www.cs.wisc.edu/condor

² www.unicore.org

³ legion.virginia.edu

⁴ www.globus.org/toolkit

que se enfoca en brindar un conjunto de servicios estandarizados que pueden ser usados para administrar infraestructura y ofrecer mecanismos de administración a sus usuarios (Alper et al., 2006)

3.1.2. *Grid* semántica

La *grid* semántica tiene su inspiración en la Web semántica, la cual según (Berners-lee, Hendler, & Lassila, 2002), es una extensión de la Web actual que permite a las máquinas y programas de computadora realizar tareas de forma inteligente. Para conseguir ese objetivo la Web debe incorporar maneras de expresar conocimiento, tiene que poder representar el conocimiento en su estructura.

Para expresar conocimiento formalmente Berners-Lee (2001) promueve el uso de ontologías las cuales según (Gruber, 1995) son “una especificación explícita y formal de una conceptualización”, en un sentido más computacional pueden ser consideradas como esquemas de metadatos que proveen vocabularios controlados para conceptos de un dominio en particular (Maedche & Staab, n.d.). Por otra parte para incorporar las ontologías a la estructura de las páginas Web se usa XML un lenguaje de marcado que permite crear nuevas etiquetas con reglas sencillas que permiten incluir fácilmente los conceptos definidos en las ontologías (Bosak & Bray, 1999). Por último para realizar tareas en forma inteligente destaca el uso de agentes que puedan realizar mejores búsquedas y recomendaciones valiéndose de los datos adicionales que se incluyen en las páginas. El uso de los elementos citados anteriormente en la Web es conocido como Web semántica.

Teniendo en cuenta lo anterior, (Goble & De Roure, 2004) definen la *grid* semántica como una extensión de la *grid* actual en el que la información y los servicios tienen un significado bien definido, mejorando el trabajo cooperativo entre computadores y personas. Definición que se ajusta a la de Web semántica emitida por W3C⁵ donde define la Web semántica como una extensión de la Web actual mejorando el trabajo cooperativo entre computadores y personas. (Roure & Hendler, 2004) expresa una visión de *grid* como la

⁵ <http://www.w3.org>

infraestructura futura para soportar e-ciencia, dotada de gran facilidad de uso y con alto grado de automatización que permita la colaboración a escala global.

3.1.2.1. S-OGSA.

La arquitectura de sistemas OGSA, proporciona un conjunto de servicios y capacidades y comportamientos enfocados en los problemas que debería resolver la computación *grid*, Los ecosistemas *grid* dependen en gran parte de la gestión eficiente de los recursos que los conforman, como consecuencia se tiene la proliferación de metadatos.

La mayor parte de los metadatos de los ecosistemas *grid* se localizan en los *middleware*, librerías u orígenes de datos, estos metadatos son usados para satisfacer necesidades puntuales de cada *middleware* lo que genera dependencia excesiva de personas en el desarrollo, mantenimiento y configuración de los mismos. Sumando el bajo nivel de interoperabilidad que se consigue. (O Corcho et al., 2006).

OGSA semántica o S-OGSA, extiende la arquitectura OGSA brindando un conjunto de mecanismos livianos que permitan en uso de semántica y servicios de conocimiento en forma asociada. SOGSA se enfoca en 6 principios de diseño que son:

- Baja complejidad de los elementos de la arquitectura.
- Extensibilidad del framework.
- Uniformidad de mecanismos.
- Diversidad de capacidades semánticas.
- Heterogeneidad de la representación semántica.
- Ilustración de servicios.

3.2. Computación *Cloud*.

Algunos autores como (Armbrust et al., 2010), (MacVittie, Murphy, Silva, & Salchow, 2010), (Vouk, 2004) y (Foster, Zhao, Raicu, & Lu, 2008b) entre otros, comparten que la definición de *cloud* aun es borrosa y presenta ambigüedad, sin embargo también hay consenso en que la computación *cloud*, incluye aspectos básicos como escalabilidad de hardware y software, pago por lo que se usa y virtualización.

(Armbrust et al., 2009) destacan 3 aspectos fundamentales del *cloud*, 1. La capacidad de aparentar poseer recursos inagotables, 2. La eliminación de preacuerdo en cuanto al uso de los recursos los cuales pueden ser incrementados o reducidos según la demanda del usuario y 3. La propuesta de pagar únicamente por lo que se consume en tiempos muy cortos.

Por otro lado la computación en la *cloud* tiene un fuerte componente comercial (Vaquero et al., 2009) destaca que la definición de *cloud* debe enfatizar en los beneficios económicos, (Chang, Bacigalupo, Wills, & De Roure, 2010) clasifica 8 modelos de negocio en torno a la computación en la nube y sus aplicaciones, usando el CCM de Jericho Fórum⁶.

(Vaquero et al., 2009) propone la siguiente definición basado en aspectos comúnmente expresados en definiciones de otros autores. “las infraestructuras *cloud* son un conjunto de recursos virtualizados fácil de usar y acceder tales como hardware, plataformas de desarrollo y/o servicios. Estos recursos pueden ser reconfigurados dinámicamente para ajustarse a cargas variables (escalable), promoviendo la utilización óptima de recursos. Este conjunto de recursos es típicamente explotado con un modelo de pago por uso donde los proveedores ofrecen garantías por medio de acuerdos de nivel de servicios hechos a medida”

Por otro lado es fácil encontrar que algunos vendedores usan las siglas SaaS software como servicios, IaaS infraestructura como servicios o PaaS plataforma como servicios para hacer referencia a los servicios *cloud* que ofrecen, sin embargo (Youseff, Butrico, & Da Silva, 2008), propone una ontología, donde presenta 5 capas que comprenden aplicaciones, entornos de software, infraestructura de software, kernel de software y hardware, que se

⁶ <https://www.opengroup.org/jericho/index.htm>

ilustra en la figura 2. Destaca en esa misma figura que la capa que otros autores solo citan como IaaS puede dividirse en 3 tipos de servicios además de IaaS, se encuentra DaaS y CaaS.

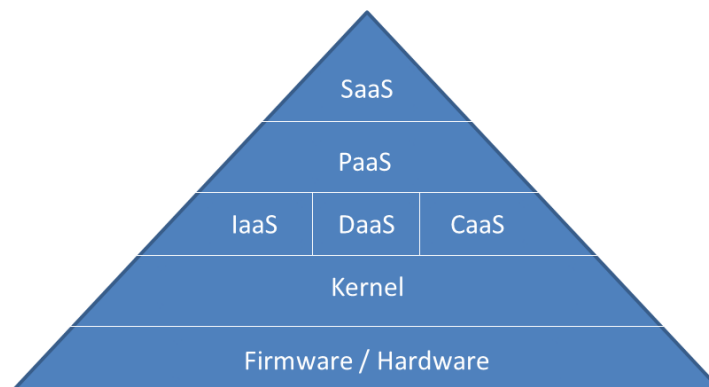


Figura 2. Pirámide de servicios *cloud* adaptada de (Youseff et al., 2008).

3.2.1. Servicios *Cloud*.

La siguiente sección aclara los conceptos SaaS, PaaS e IaaS, tomando como referencia la pirámide mostrada en la Figura 2, donde se muestran los servicios que se pueden ofrecer en *cloud*, en una jerarquía donde se diferencia claramente que la infraestructura soporta las plataformas y el software.

En la figura 3, se muestra la pirámide de servicios *cloud* en sus configuraciones más comunes, en el lado izquierdo se ve la pirámide completa con los 3 servicios, en la parte derecha se observa el servicio IaaS que soporta el servicios SaaS.

La configuración de servicios del lado derecho es mucho más común dado que cualquier plataforma SaaS debe ser soportada por un servicio IaaS, la capa correspondiente al servicio PaaS es opcional.

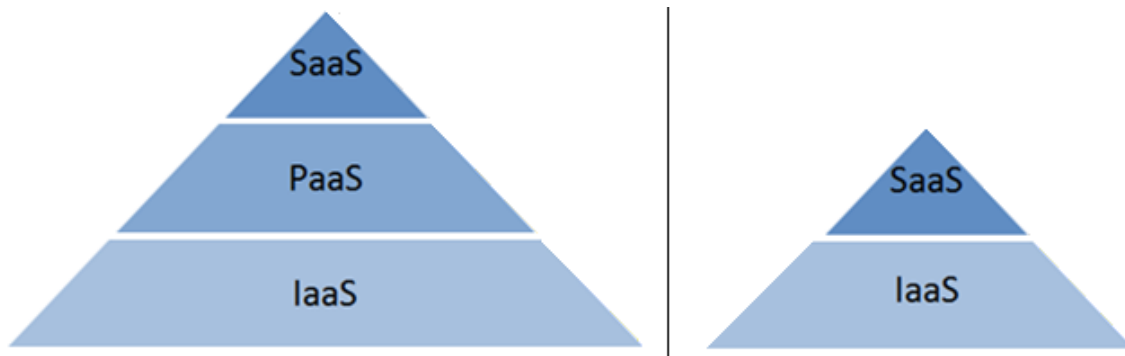


Figura 3. Pirámide servicios *cloud*, en dos configuraciones comunes. Fuente: El autor.

Software como servicio SaaS.

Software como servicio, su particularidad es que el usuario paga por el uso de software de terceros, en dicho sentido el usuario tiene premisos de usar y personalizar el software despreocupándose de la plataforma y de los recursos computacionales que consuma la aplicación. (Vaquero, Rodero-Merino, Caceres, & Lindner, 2008), (MacVittie et al., 2010).

Infraestructura como servicios IaaS.

IaaS es el servicio de provisión de recursos computacionales como procesamiento y almacenamiento. Bajo este esquema, los usuarios que acceden a este tipo de servicio tienen total libertad de personalizar la plataforma y software que deseen. Los aspectos concernientes a escalabilidad del hardware, aumento y disminución de recursos, se realiza bajo demanda y de forma transparente al usuario final. (Vaquero et al., 2008), (MacVittie et al., 2010).

Tomando en cuenta lo destacado por (Armbrust et al., 2010) y la promesa de simular recursos ilimitados. Es de suponer que los centros de datos destinados para *cloud*, son de gran tamaño, también es posible tener centros de datos de tamaño medio que ofrecen servicios *cloud*, que para poder soportar exitosamente su operación deben contratar servicios IaaS de otros proveedores más grandes cuando tienen picos de demanda.

Plataformas como servicios PaaS.

Las plataformas como servicios presentan un nivel de abstracción que permite desarrollar aplicaciones sobre una plataforma de software determinada, los proveedores PaaS ofrecen lenguajes de programación y entornos de desarrollo con un conjunto de librerías bien definidas, por ejemplo Java EE, IBM WebSphere, Oracle, *Google Apps*, .NET, BizTalk (Vaquero et al., 2008), (MacVittie et al., 2010), (Youseff et al., 2008).

Los usuarios de servicios PaaS, generalmente son desarrolladores de aplicaciones, que esperan desplegarlas en entornos *cloud* (Youseff et al., 2008), el uso de servicios PaaS libera al usuario de la preocupación por licencias de uso y en muchos casos de la construcción de módulos. Sin embargo el hecho de no poder utilizar componentes y elementos propios de otras plataformas limita los resultados finales.

Almacenamiento de datos como servicio DaaS.

El almacenamiento de datos como servicio más conocido como DaaS, se enfoca en brindar al usuario la opción de almacenar datos en discos remotos a cualquier hora y desde cualquier lugar, (Youseff et al., 2008) considera que la disponibilidad, escalabilidad y consistencia de datos son características de suma importancia para el servicio DaaS pero pueden presentar conflictos para presentarse juntas, en consecuencia los proveedores dan mayor importancia a alguna sobre las otras de manera que se pueden tener servicios basados en archivos, bases de datos relacionales replicadas y almacenes llave valor, por ejemplo los RDBMS da más importancia a la consistencia de los datos, mientras los almacenes llave valor prestan más atención a la disponibilidad de los datos sobre la consistencia de los mismos.

Comunicaciones como servicio CaaS.

Según (Youseff et al., 2008) las infraestructuras de comunicaciones se constituyen en elementos fundamentales para asegurar la calidad del servicio (QoS), en dicho sentido

CaaS toma un papel protagónico dado que el servicio debe ser configurable, planeado, predecible y confiable para poder lograr los objetivos del cliente. Por otro lado el servicio CaaS debe soportar requerimientos como seguridad de redes, aprovisionamiento dinámico de capas virtuales para el aislamiento de tráfico o ancho de banda dedicado.

3.3. Acuerdo de nivel de servicio SLA

Un SLA es un documento (Contrato) entre un consumidor y un proveedor de servicios, en el que el primero manifiesta sus necesidades y el segundo indica cómo va a cumplir dichas necesidades, especificando la calidad de servicio aceptable, generalmente esta última está asociada con costos, aun cuando se puede incluir cláusulas penales por el incumplimiento de los acuerdos. (Wustenhoff, 2002), (Menasce & Casalicchio, 2004). Por su parte, (Patel, Ranabahu, & Sheth, 2009) consideran que desde la perspectiva del proveedor de servicios es imposible cumplir con todas las expectativas de los clientes, por lo tanto se debe realizar una negociación, al final de la misma se genera el SLA.

Por otro lado (Jin et al., 2002) expone 9 componentes generales que podrían tener los SLA orientado a los Servicios Web.

1. *Propósito*: donde se describe la razón de la creación del SLA.
2. *Participantes*: donde se describe los participantes que involucra el SLA y sus respectivos roles.
3. *Periodo de validez*: define un periodo en el tiempo durante el cual el SLA será válido.
4. *Ámbito*: define los servicios cubiertos por el SLA.
5. *Restricciones*: define los pasos necesarios para que se pueda prestar el servicio.
6. *Objetivo del nivel de servicio*: generalmente el contrato suscrito por cliente y proveedor de servicios incluye un conjunto de indicadores como disponibilidad, rendimiento y confiabilidad. Cada uno de los indicadores deberá tener una meta a alcanzar.
7. *Penalidades*: Indica lo que ocurre si el proveedor de servicios incumple los

objetivos del SLA.

8. *Servicios opcionales*: contiene los servicios que no son tomados por el usuario al momento de generar el SLA, pero que el proveedor de servicios podría cubrir como una extensión.
9. *Administración*: describe la responsabilidad de las partes en la supervisión de los procesos con el fin de cumplir con los objetivos planteados.

3.3.1. Ciclo de vida de un SLA.

Dado que el SLA es un documento que tiene implicaciones directas sobre la prestación del servicio, es importante ubicar el ciclo de vida de este documento y las fases del mismo para poder garantizar QoS.

En (Cardellini, Casalicchio, & Branco, 2011) se muestra un ciclo de vida para los SLA de 3 fases, propuesto por Ron, S., & Aliko, P. en su artículo “*Service Level Agreement*”:

1. Creación.
2. Operación.
3. Remoción.

El ciclo de vida anterior se lleva a un mayor nivel de detalle por “*Sun Microsystems Internet Data Center Group*” en 2002 expresando un ciclo de vida de 6 pasos:

1. Descubrimiento del proveedor de servicios.
2. Definición del SLA.
3. Establecer el acuerdo.
4. Monitoreo de violaciones del SLA.
5. Terminación de SLA
6. Imposición de sanciones por violación del SLA.

La tabla 1 ilustra la conciliación entre los dos ciclos de vida como se puede observar se conserva el ciclo de vida de 3 fases pero con el ciclo de vida de 6 fases se la mucha más claridad y detalle.

Tabla 1: Conciliación entre el ciclo de vida de SLA 3 fases y ciclo de vida de SLA de 6 fases. Fuente: (Wu & Buyya, 2010). Service Level Agreement (SLA) in Utility Computing Systems.

Ciclo de vida de 3 fases.	Ciclo de vida de 6 fases.
Fase de creación.	Descubrimiento de proveedor de servicio.
	Definición de SLA.
	Establecer el acuerdo.
Fase de operación.	Monitoreo de violaciones del SLA.
Fase de remoción.	Terminación de SLA.
	Imposición de sanciones por violación del SLA.

A continuación se realiza una breve descripción de cada una de las fases del ciclo de vida propuesto.

Fase de descubrimiento del proveedor de servicio: En computación *cloud* es posible que un proveedor utilice recursos de otros proveedores, así que es importante conocer cuántos recursos se encuentran disponibles.

Definición del SLA: En la definición del SLA es necesario conocer los requerimientos del cliente y los aspectos del SLA que será firmado, términos del servicio que incluyen parámetros de calidad de servicio (QoS). Esta etapa se consigue mediante el proceso de negociación entre las partes involucradas proveedor y consumidor.

Establecer el acuerdo: En esta etapa se construye una plantilla del SLA, la plantilla debe contener todos los componentes del SLA.

Monitoreo de violaciones del SLA: El propósito de esta etapa es determinar si el SLA se cumplió o hubo alguna violación. Existen tres cuestiones principales en el monitoreo que son:

1. Que parte se encarga del proceso.
2. Como se puede asegurar justicia entre las partes.
3. Como se definen los límites de la violación del SLA.

Terminación de SLA: La terminación es un aspecto muy importante es el momento en que se decide que el SLA ya no tienen validez, esta situación se puede presentar por varios factores como una violación, renegociación o cumplimiento del tiempo acordado para el servicio.

Imposición de sanciones por violación del SLA: En esta etapa se imponen las sanciones respectivas por las violaciones a SLA, de manera que la parte que incumpla deberá cubrir los perjuicios de la parte afectada. Sin embargo existen inconvenientes a la hora de fijar los montos de las penalidades.

Por otra parte (Shu & Meina, 2010) manifiestan que en general los SLA cuentan con 5 etapas principales que son:

1. Desarrollo del servicio.
2. Negociación y mercadeo
3. Desarrollo
4. Implementación

5. Evaluación.

Sin embargo en su trabajo propone un manejador de SLA de dos momentos antes y después de la ejecución, antes de ejecución se realizan las siguientes tareas registros del SLA, consulta y negociación del contrato, en tiempo de ejecución se ejercen tareas de monitoreo y control.

3.3.2. SLA En computación *Cloud*.

Los documentos SLA son usados en computación *grid* y en *cloud* con diferentes enfoques, sin embargo su objetivo es el mismo establecer un documento donde se contemplan los requisitos mínimos de calidad de servicio, que son exigencia del consumidor y que el proveedor debe cumplir con más o menos rigurosidad según el caso.

Como ya se ilustro en el apartado referente a computación *cloud*, esta presenta una orientación comercial y se soporta en varios modelos de negocio que destacan el hecho de pagar por lo que usa, en ese sentido es difícil mantener un SLA en el tiempo dificultando el acuerdo entre cliente y proveedor.

(Wu & Buyya, 2010) muestra el manejo de SLA en computación *cloud* de una manera más compleja, cuando se involucran proveedores SaaS, usuarios finales y proveedores IaaS, donde el proveedor SaaS debe cumplir con las exigencias de sus clientes y el proveedor o proveedores IaaS deben cumplir con las exigencias de los proveedores SaaS.

Tomando como base el trabajo de diferentes autores, se tienen dos servicios *cloud* donde la aplicación de SLA merece la pena ser estudiada, dichos servicios son:

1. Servicios IaaS.
2. Servicios SaaS.

En el primer caso los proveedores de infraestructura deben cumplir las demandas de los proveedores SaaS que soportan su operación sobre recursos arrendados bajo demanda. En el segundo caso los proveedores de SaaS deben satisfacer las demandas de los usuarios finales.

(Wu & Buyya, 2010) también muestra el modelo de multas aplicado a las violaciones de SLA por parte de los proveedores de SaaS, los cuales no solo deben cubrir las demandas requeridas por los cliente sino también compensarlos por aceptar servicios en esas condiciones. En contraste según datos citados, el 85% de los proveedores de IaaS no contemplan acciones legales para el incumplimiento de SLA.

De otro lado al igual que su contraparte en computación *cloud* los SLA que se usan, son acuerdos donde se destacan entre otros los recursos computacionales necesarios para que una tarea pueda llevarse a cabo.

Los SLA en ecosistemas *grid* a menudo presentan soporte para descubrimiento, negociación y reserva de recursos, sin embargo muchos carecen de soporte para adaptación en tiempo de ejecución, lo que podría deberse a su naturaleza no comercial. (Padgett, 2005). De manera contraria un SLA usado para servicios de computación debe incluir soporte para negociación dado que una de sus características fundamentales es la generación de un pago.

3.4. Proceso de negociación

La negociación es el proceso mediante el cual las partes involucradas en un conflicto logran llegar a un acuerdo de forma que todos los actores que intervienen obtengan beneficios mutuos, en computación *cloud* se pueden encontrar diferentes tipos de negociación en los que intervienen varios actores.

3.4.1. Partes del proceso de negociación.

La negociación se compone esencialmente de dos partes. 1. Protocolo de negociación especifica reglas para la interacción entre las partes (intercambio de mensajes, condiciones para llegar a un acuerdo etc.), 2. Estrategias de negociación que definen el plan de acción que puede tomar un participante durante la negociación, el protocolo de negociación debe ser público y usado por todos los participantes por otro lado las estrategias son privadas y propias de cada participante orientadas a conseguir el mayor beneficio posible. (Lomuscio, Wooldridge, & Jennings, 2003)

Escenarios de negociación.

Desde la perspectiva de la negociación de servicios de computación *cloud* se pueden observar diferentes escenarios donde es posible identificar los actores involucrados en dicho proceso. (Alhamad, Dillon, & Chang, 2010) muestra 3 escenarios que son expuestos a continuación.

Escenario 1.

La figura 4 ilustra un escenario de negociación directa donde un proveedor de servicios expone una plantilla de SLA en la cual expresa todos los criterios tomados en cuenta por el

para la prestación del servicio como duración del contrato, facturación y tiempos de respuesta, de otro lado el usuario puede tomar varias decisiones como:

1. Aceptar los términos propuestos por el proveedor y firmar.
2. Renegociar el SLA.
3. Declinar terminar el proceso con dicho proveedor.

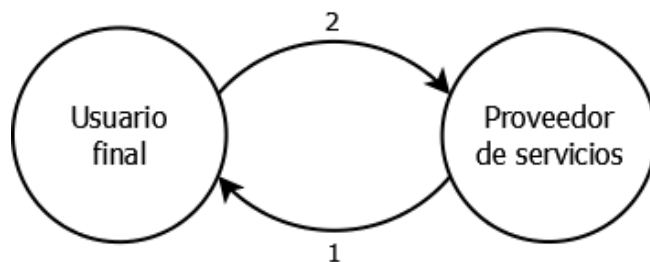


Figura 4: Esquema de negociación directa entre usuario final y proveedor de servicios.

Fuente: El autor.

Escenario 2.

En el presente escenario se encuentra un actor extra no identificado en el primer escenario. Como se observa en la figura 5, el cliente es representado por un intermediario (bróker) el cual está encargado de encontrar al proveedor que más se ajuste a los requerimientos de brindados por consumidor final.

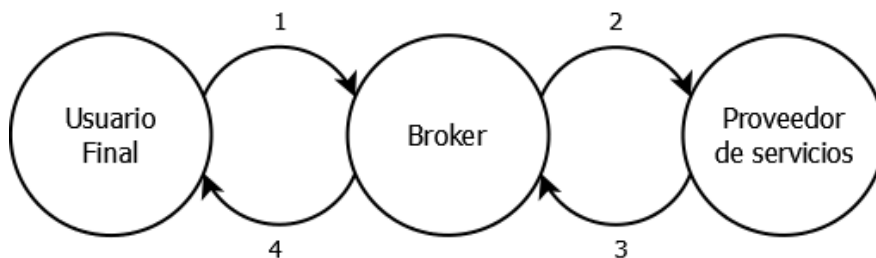


Figura 5: Esquema de negociación entre usuario final y proveedor por medio de un bróker.

Fuente: El autor.

Escenario 3.

En este escenario es posible que un usuario final utilice a diferentes brókeres especializados en asuntos específicos como se muestra en la figura 6, por ejemplo en se puede tener un cliente con la necesidad de adquirir infraestructura y además utilizar algunos programas como servicio para la operación que tenga en mente, en ese caso se puede usar un bróker para IaaS y otro para SaaS.

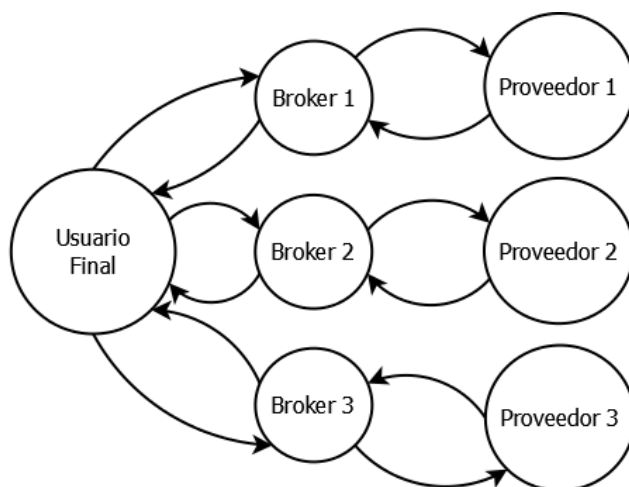


Figura 6: Esquema de negociación entre usuario final y proveedores por medio de múltiples bróker. Fuente: El autor.

En el trabajo de (Buyya, Yeo, & Venugopal, 2008) se muestra un acercamiento a lo que debería ser un mercado orientado a la administración de recursos, donde se regule la demanda y oferta de servicios. En dicho trabajo se expone una arquitectura para un mercado de intercambio global de recursos donde naturalmente se encuentran implicados otros actores diferentes a los que ya se han nombrado.

En la figura 7, se pueden apreciar además de los actores conocidos proveedor, consumidor y bróker, otros actores responsables proporcionar equilibrio al mercado en este caso particular el directorio, el banco y la subasta. Los cuales permiten que el funcionamiento de un mercado de servicios *cloud* sea posible según el autor.

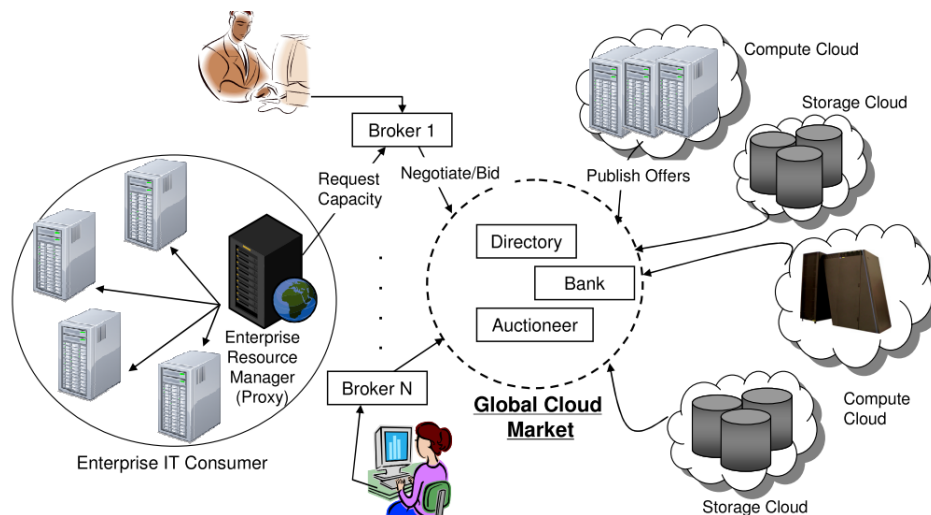


Figura 7: Mercado de infraestructura para el comercio de servicios *cloud*. Tomado de (Buyya et al., 2008)

Por otra parte (Liu, Tong, Mao, & Bohn, 2013) también proponen 3 diferentes escenarios de interacción 1. El de consumidor y proveedor ya mostrado en la figura 4, el escenario de consumidor, bróker, proveedor mostrado en la figura 5 con una ligera variación que incluye al Carrier y un tercer escenario aun no contemplado que se muestra en la Figura 8, donde el auditor tiene relación directa con el proveedor y el consumidor de servicios.

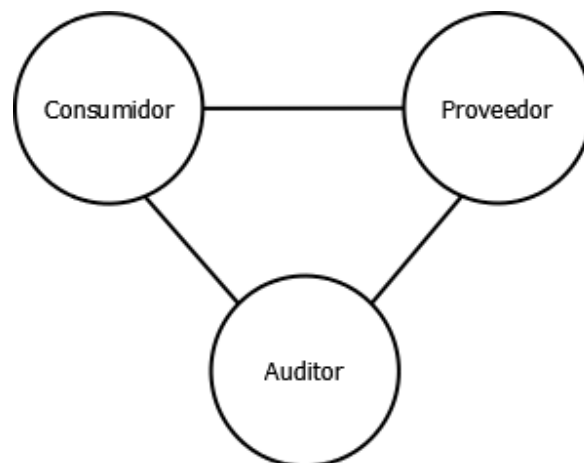


Figura 8: Escenario de interacción consumidor, proveedor, auditor tomado de (Liu et al., 2013).

4. Investigación

4.1. Objetivos

4.1.1. Objetivo general.

- Proponer un marco de referencia para la negociación y monitoreo de acuerdos de nivel de servicio (SLA), en ambientes *cloud*.

4.1.2. Objetivos específicos

- Definir los actores involucrados dentro del marco de referencia.
- Establecer dos escenarios posibles de interacción entre los actores.
- Definir los mecanismos que permitan la interacción entre los actores.
- Desarrollar plantilla de SLA. Proponer la definición de un documento que de soporte a los procesos de negociación y monitoreo en ambientes *cloud*.
- Proponer un procedimiento para la negociación de SLA entre usuarios finales y proveedores SaaS.
- Proponer un procedimiento para el monitoreo de SLA entre usuarios finales y proveedores SaaS.
- Proponer un procedimiento para la negociación de SLA entre proveedores SaaS e IaaS.

4.2. Resultados esperados

Con la presente propuesta se esperan los siguientes resultados.

- Un marco de referencia para negociación y monitoreo de acuerdos de nivel de servicio SLA, en ambientes *cloud*.
- Un artículo escrito para ser presentado en alguna revista o medio de difusión.

4.3. Metodología

4.3.1. Alcance del trabajo de grado.

La investigación se realizara para infraestructuras *cloud*, considerando usuarios finales, proveedores SaaS e IaaS. Se deja por fuera el proveedor PaaS, las razones para no abordar el servicio PaaS son las siguientes.

1. El servicio PaaS tiene la participacion mas baja del mercado de los tres servicios principales.
2. La proyeccion de crecimiento del servicio PaaS para los proximos cinco años es inferior al cinco porciento.
3. Muchos usuarios utilizan servicios IaaS para instalar sus plataformas sin contratar servicios PaaS.

En la tabla 2, se presenta el crecimiento de los sevicios *cloud* en los ultimos años, se muestra lo que se vendio en billones de dolares y el porcentaje de crecimiento. Se puede observar claramente que el volumen mas bajo de consumo se situa en los servicios PaaS, aun cuando en 2016 crecio un 21.1%, si se observa la columna de 2016 en billones de dolares los servicios SaaS e IaaS superan ampliamente al Servicio PaaS.

Tabla 2: Crecimiento de servicios *cloud* en 2015 y 2016. Tomado de <http://www.gartner.com/newsroom/id/3188817>

Servicio	2015 - Billones de dólares	2015 % de crecimiento	2016-Billones de dólares	2016 % de crecimiento
SaaS	31.4	15.5	37.7	20.3
PaaS	3.8	16.1	4.6	21.1
IaaS	16.2	31.9	22.4	38.4

Aun cuando todos los servicios *cloud*, han crecido en términos de ventas y de porcentajes es una realidad que los servicios que pueden ser considerados de uso masivo son IaaS y

SaaS. Por tal razón son seleccionados para el presente trabajo. Como se muestra en la figura 9 el presente trabajo involucra las entidades correspondientes a usuarios finales, proveedores IaaS y proveedores SaaS.

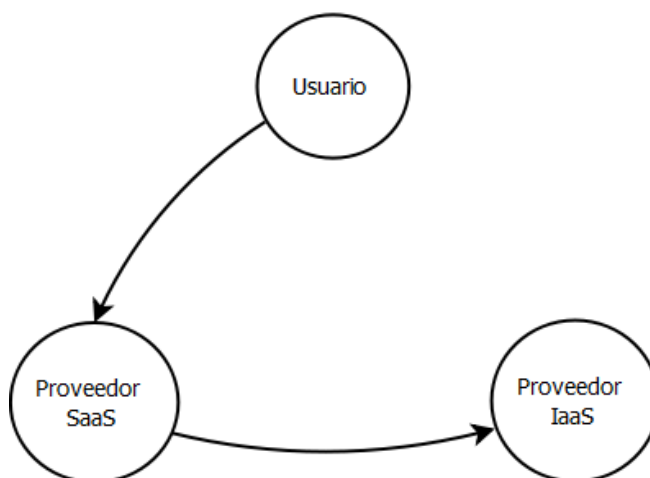


Figura 9. Entidades involucradas en la investigación. Fuente: El autor.

El desarrollo de la investigación se enfoca en la negociación y monitoreo de documentos SLA en infraestructuras *cloud*, dejando por fuera el proceso de reserva de recursos. La decisión de no abordar el proceso de reserva de recursos obedece que a dicho proceso se encuentra por fuera del alcance de la negociación.

Cuando se efectúa la compra de un servicio *cloud* IaaS el usuario final puede configurar una instancia de máquina virtual o puede comprar una máquina preconfigurada, en ambos casos la reserva de recursos está a cargo del proveedor de servicios quien debe aprovisionar las máquinas con las configuraciones solicitadas.

El SLA resultante del proceso de negociación, proporciona los argumentos necesarios para que los sistemas de aprovisionamiento sepan las características particulares de las solicitudes del cliente. No obstante el proceso de aprovisionamiento y reserva de recursos es interno y el proveedor incluso puede hacerlo de forma dinámica.

La Figura 10, ilustra un posible ciclo de negociación donde un usuario puede comprar un

servicio SaaS montado sobre un servicio IaaS, en el cual no es necesario conocer la forma en que se aprovisionan los recursos.

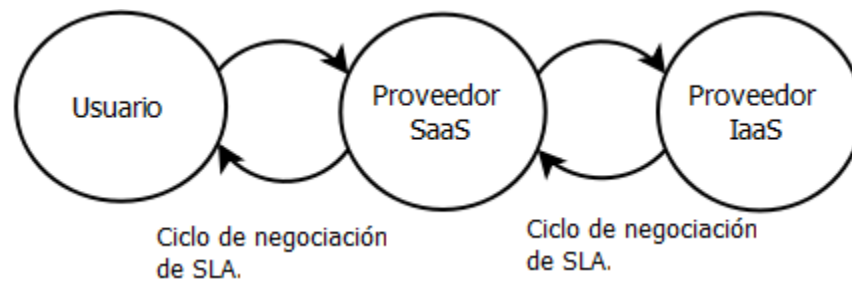


Figura 10. Ciclo de negociaciones de SLA. Fuente: El autor.

Otro aspecto importante que se acota en la investigación es la característica que tienen los proveedores SaaS de recurrir a un mercado de proveedores IaaS, lo que hace dinámico el comportamiento de lo que un proveedor SaaS puede aprovisionar. Sin embargo para el desarrollo del proyecto se tomara una cantidad fija de recursos las cuales pueden estar distribuidas en diferentes proveedores IaaS o en uno solo.

En cuanto al desarrollo de los mecanismos de interacción entre los actores del proyecto, desde el principio se tuvo en cuenta la utilización de ontologías. Sin embargo existen otras formas de lograr dichos mecanismos, en el transcurso de la investigación se examinaron diferentes formas de llegar a una plantilla de SLA como XML, WSLA y desarrollo de ontologías.

Se seleccionó como mecanismo para la interacción el desarrollar ontologías, por presentar un método formal para la definición de un dominio concreto y por la flexibilidad inherente a ellas. En dicho sentido se recurre a la creación de ontologías para definir los dominios SaaS, PaaS y SLA.

4.3.2. Actividades.

Tabla 3. Actividades del proyecto.

#	Nombre de la actividad
1	Revisión general de bibliografía
2	Caracterización de actores involucrados en la negociación y monitoreo de SLA
3	Definición de actores involucrados
4	Selección de escenarios de interacción
5	Recopilación de documentos SLA
6	Caracterización de documentos SLA
7	Propuesta de documento SLA
8	Selección de proveedores SaaS
9	Caracterización del proceso de negociación entre proveedores SaaS y usuarios
10	Propuesta para el procedimiento de negociación
11	Caracterización del proceso de monitoreo de SLA entre proveedores SaaS y usuarios
12	Caracterización de la negociación de SLA entre proveedores SaaS e IaaS
13	Propuesta para gestionar la negociación de SLA entre Proveedores SaaS e IaaS
14	Redacción de informes
15	Caracterización de servicios ofrecidos por proveedores SaaS seleccionados

5. Actores involucrados en el proceso.

Independientemente de si el enfoque es directo o no, por medio de intermediarios o usando el sistema de subasta los actores fundamentales siguen estando presentes. Dichos actores son:

1. El consumidor.
2. Intermediario (Bróker) opcional.
3. El proveedor de servicios.
4. Auditor (Opcional).

Los otros elementos que aparecen en los diferentes enfoques están orientados a problemas más amplios que no están dentro del alcance de este trabajo.

Descripción de las responsabilidades de los actores.

Consumidor o usuario final.

El consumidor de servicios es el actor principal en todos los escenarios de interacción teniendo en cuenta que es la persona u organización que paga por los servicios y en dicho sentido es el actor que inicia la dinámica de la negociación, el consumidor tiene que saber cuál es su necesidad para poder configurar el o los contratos de servicios con los proveedores o brókeres según sea el caso.

Según sus necesidades (Liu et al., 2013) los clasifica como consumidores SaaS, consumidores PaaS y consumidores IaaS. Sin embargo en términos generales su responsabilidad es la misma lo que cambia es su necesidad y en consecuencia el tipo de servicio que desean contratar.

Proveedores *Cloud*.

A diferencia de los usuarios finales los proveedores de servicios *cloud* poseen responsabilidades generales pero dependiendo del servicio ofrecido también tienen responsabilidades específicas, esto se puede deber a que en muchos casos todos los servicios no pueden ser cubiertos por un único proveedor así que los proveedores se especializan en un servicio en particular que puede actuar cohesionado con servicios de otros proveedores.

Los proveedores SaaS despliegan, configuran, mantienen y actualizan el software, por otra parte los proveedores PaaS gestionan la infraestructura computacional y ejecuta el software que permite funcionar a la plataforma tal como pilas de ejecución de software, bases de datos y otros componentes del *middleware*.

Naturalmente los proveedores PaaS también soportan desarrollo, despliegue y gestión de procesos proveyendo herramientas como entornos de desarrollo integrado (IDE) por sus siglas en español, controladores de versiones y kits de desarrollo de software (SDK) por sus siglas en español, además despliega mantiene y gestiona dichas herramientas.

En la capa más baja los proveedores IaaS proveen recursos físicos como que incluyen servidores, redes y almacenamiento. Que ponen al servicio del consumidor utilizando diversos niveles de abstracción como máquinas virtuales e interfaces de red virtuales. Los proveedores IaaS tienen a su cargo la infraestructura física y el software que permite aprovisionar dicha infraestructura como servicio.

Brókeres.

Aun cuando según (Plummer & Kenney, 2009). Los brókeres pueden enfocarse en tres tareas principales que son:

1. Intermediación de servicios *cloud*.
2. Agregación de servicios.

3. Arbitraje de servicios *cloud*.

En este trabajo solo se usaran como intermediarios entre los consumidores de servicios y proveedores.

Los brókeres son los encargados de tomar el SLA de un cliente y encontrar el proveedor que mejor se adapte a sus exigencias entre un conjunto de ellos, el bróker debe tener conocimiento de las necesidades del cliente y por otro lado debe saber que proveedores podrían satisfacer dichas necesidades.

Auditor.

Una vez desplegado el servicio por parte del proveedor es necesario para el consumidor saber si se está cumpliendo con todos los requerimientos del SLA. El auditor es un tercero de confianza que es contratado para verificar que todo se esté cumpliendo de acuerdo al contrato. (Plummer & Kenney, 2009) menciona que esta tarea puede ser realizada por un bróker sin embargo en este trabajo se diferencian los roles de bróker y auditor apegándose más a las responsabilidades planteadas en (Liu et al., 2013).

6. Escenarios de interacción entre actores.

En el mercado de servicios *cloud*, se tienen algunos escenarios de interacción, se puede hacer una división entra los escenarios teóricos no implementados o implementados insipientemente a los cuales se llamara escenarios emergentes, y los escenarios de interacción en la oferta de servicios actual. A continuación se enuncian algunos escenarios y sus características.

En el mercado de servicios *cloud* se encuentran básicamente para este trabajo dos tipos de escenarios de interacción a saber, escenarios teóricos no implementados o implementados insipientemente a los cuales se llamará escenarios emergentes, por otra parte se encuentran escenarios de interacción implementados en la oferta de servicios actual. Ambos tipos de escenario serán ampliados en las secciones siguientes.

6.1. Escenario de interacción en la oferta de servicios IaaS y SaaS actual

En la oferta de servicios actual se encuentran esencialmente cuatro escenarios de interacción.

1. Búsqueda individual de proveedores por parte del usuario final.
2. Servicios de recomendación basados en filtros.
3. Servicios de recomendación basados en ranking.
4. Servicios de intermediarios brókeres que basados en la necesidad del cliente recomiendan proveedores de servicios.

Búsqueda individual de proveedores por parte del usuario final

En este escenario el usurario final realiza el 100% del trabajo, en el que el mismo debe buscar, evaluar y seleccionar el mejor proveedor de servicios, según sea su necesidad. Este escenario podría ser usado por un consumidor de poca experiencia en el mercado de servicios *cloud*, sin embargo por lo reciente de dicha tecnología aun la gran mayoría de usuarios son inexpertos en la compra de esa clase de servicios.

Servicios de recomendación basados en filtros sin ranking.

Los servicios de recomendación basados en filtros permiten al usuario final establecer sus requerimientos en un alto nivel según sea el servicio que necesite, cabe destacar que el usuario debe estar muy consciente de sus necesidades. Este tipo de escenario es mucho más cómodo para el consumidor dado que en una sola parte puede encontrar un grupo considerable de proveedores que pueden cubrir sus necesidades. Sin embargo, la evaluación debe hacerla el propio usuario el servicio solo llega a recomendar pero no evalúa las capacidades del proveedor respecto de las necesidades del consumidor.

Para el caso de IaaS se pueden encontrar sitios como www.intelcloudfinder.com y www.computenext.com ambos usan filtros para poder definir mejor las necesidades del cliente y basados en dichos criterios entregan una lista de proveedores que podrían suplir las demandas del cliente.

En la figura 11, se muestra el panel de filtros de www.intelcloudfinder.com, este servicio cuenta con dos paneles de filtros, el panel de búsqueda rápida que se basa en cajas de chequeo, a medida que el cliente va seleccionando características, se van mostrando los posibles proveedores para las necesidades del cliente, es posible que varios, uno o ninguno pueda suplir las necesidades del cliente.

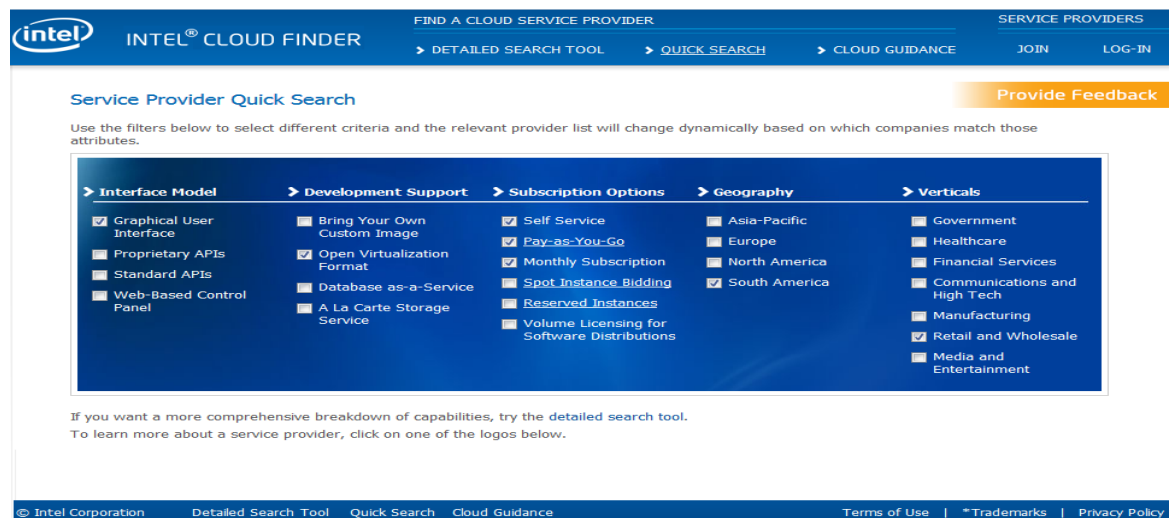


Figura 11. Panel de filtros usado por <http://www.intelcloudfinder.com> en su configuración de búsqueda rápida. Fuente: tomado de <http://www.intelcloudfinder.com/quicksearch> agosto de 2013.

Por otra parte el sitio <https://www.computenext.com/> ofrece una mayor gama de servicios IaaS y SaaS, también su forma de filtrar es diferente a la del ejemplo anterior ya que no tiene cajas de chequeo. En la figura 12, se muestra el panel de filtros que utiliza para obtener los requerimientos del cliente.

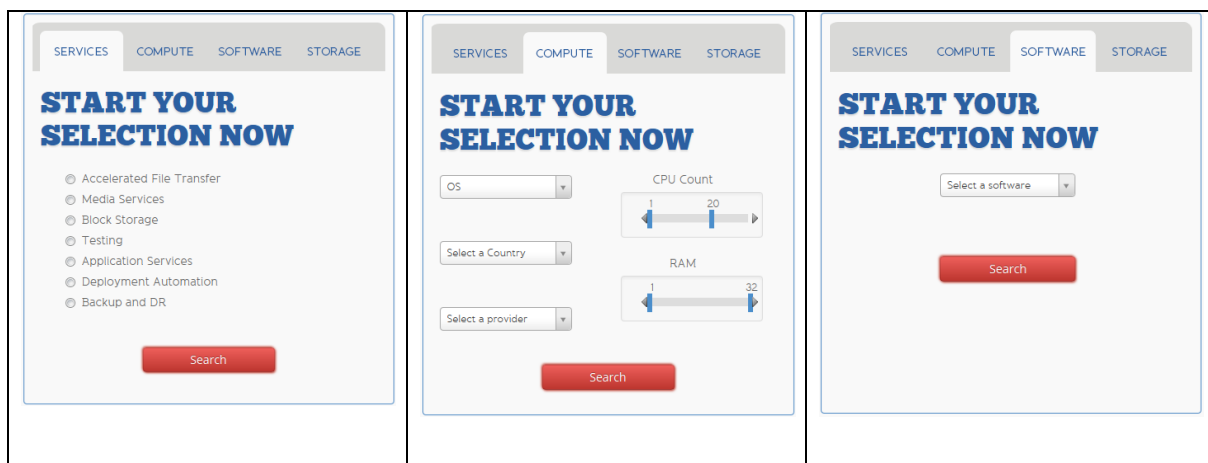


Figura 12. Panel de filtros usado por el servicio <https://www.computenext.com/> Fuente: tomado de <https://www.computenext.com/> agosto de 2013.

En cuanto a los servicios SaaS se encuentran portales como Saasometro disponible en <http://www.eoi.es/saassometro/web/inicio/seleccionArea>. El servicio al igual que los mostrados anteriormente se basa en filtros, pero este es únicamente para buscar software como servicio para Pymes. En la figura 13, se muestra el panel de filtros que tiene donde se pueden observar diferentes tipos de productos software.

The screenshot shows the 'Comparador de aplicaciones web SaaS para Pymes' interface. It features a search bar with the placeholder 'Busca una aplicación para compararla:' and a text input field labeled 'Nombre de la aplicación'. To the right, there are two main sections for filtering. Section 1, titled '¿De qué área funcional quieres usar la herramienta SaaS?', lists functional areas: ERP, CRM, Contabilidad, Tienda Online, Gestión de Proyectos, Facturación, E-learning, Nóminas, Backup, Business Intelligence, and Otros. Section 2, titled 'Selecciona categoría. Para uso:', lists usage categories: Básico, Medio, Avanzado, and Todas. At the bottom, there are navigation arrows and a 'Siguiente' button.

Figura 13. Panel para búsqueda de servicios SaaS para Pymes. Tomado de <http://www.eoi.es/saassometro/web/inicio/seleccionArea> visitada 24 de agosto de 2013

Servicios de recomendación basados en filtros con ranking.

En esta categoría se encuentran servicios que hacen un ordenamiento de los proveedores de manera que dan al usuario final un dato extra para la toma de decisiones el cual es su ponderación en el *ranking*. Dicha ponderación puede ser un factor diferenciador cuando muchos proveedores pueden cumplir con las expectativas de un cliente.

El servicio <http://www.findthebest.com/> utiliza filtros y también *ranking*, en la figura 14, se muestra al lado izquierdo la columna de filtros que se usa en <http://www.findthebest.com/> para la selección de una lista de proveedores que pueden suplir las necesidades de un cliente. El buscador se centra en servicios IaaS para lo cual

se toman en cuenta características como tipo de licencia de software, forma de pago, interface de control, soporte gratuito y sistema operativo.

Por otro lado en la parte derecha se observan los filtros que se utilizan para mejorar las búsquedas de proveedores por medio de una escala de 5 valores donde realizan 5 preguntas.

1. ¿Cuán importante es un sistema operativo compatible?
2. ¿Cuán importante es la interfaz de control del proveedor?
3. ¿Cuán importantes son características del soporte que ofrece el proveedor?
4. ¿Cuán importante es que el proveedor ofrezca soporte gratuito?

Se puede observar que es un servicio bastante sencillo y muy útil para recomendar, sin embargo cuando se ha llegado a la selección de proveedores que pueden cumplir con las necesidades del cliente, el servicio muestra los resultados ordenados por un *ranking*.

The image displays two side-by-side screenshots of the AssistMe interface. The left screenshot shows a sidebar with various filter categories: COMPUTING CATEGORY (All, Cloud Management (18), Infrastructure as a Service (13), Platform as a Service (22)), SOFTWARE LICENSE (All, Open Source (13), Proprietary (55)), SUBSCRIPTION TYPE (All, Pay as you go (21), Pay as you go and Subscription Plans (39), Subscription Plans (19)), CONTROL INTERFACE (All, Web Based Application/Control Panel (53), API (Application Programming Interface) (48), Command Line (18), Graphical User Interface (30)), FREE SUPPORT (All), OPERATING SYSTEMS (All), and POPULAR SEARCHES (Select). The right screenshot shows the main survey area with the title 'AssistMe | Help us recommend the best cloud computing providers for you'. It contains four questions, each with a 5-point scale from 'Unimportant' to 'Very Important': 'How important are the compatible operating systems?', 'How important is the control interface of the cloud computing provider?', 'How important are the support features of the cloud computing provider?', and 'How important is it that the cloud computing provider offer free support?'. An 'AssistMe' button is located at the bottom of the survey area.

Figura 14. Filtros aplicados en [www.findthebest.com](http://cloud-computing.findthebest.com/) fuente: Tomado de <http://cloud-computing.findthebest.com/> Agosto de 2013.

Caso similar al mostrado anteriormente ocurre con el sitio <https://www.saasmax.com> que se especializa en servicios SaaS, donde primero por medio de una serie de filtros se

ubica a los proveedores que pueden suplir la necesidad del cliente, y después los resultados son presentados usando *ranking*. En la figura 15, se muestra el panel que se usa para la selección en saasmax.com.

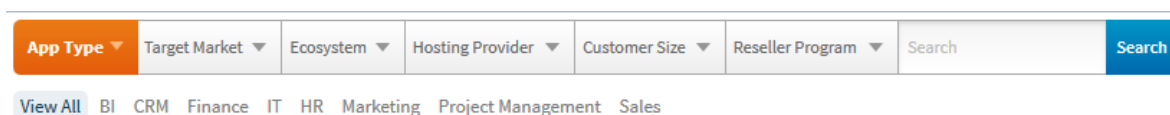


Figura 15. Filtros aplicados en <https://www.saasmax.com> fuente: Tomado de <https://www.saasmax.com> Agosto de 2013.

Servicios de intermediarios Brókeres.

Aunque según Gardner, todos los ejemplos anteriores cabrían dentro del concepto de *bróker*, en este apartado se toma dicho actor como un agente externo que consigue para el cliente el mejor proveedor basado en sus necesidades, en internet se pueden encontrar servicios que cuentan con esas características como <http://www.cloudcompare.ie>.

6.2. Escenarios emergentes

Además de los escenarios de interacción consumidor – proveedor de forma directa, y los servicios de recomendación que se presentan en la actualidad. Algunos artículos abordan modelos alternativos que pueden mejorar el proceso en aplicaciones futuras, los modelos alternativos se plantean desde modelos de mercado, dinámicas de oferta y demanda, y sistemas dirigidos por subastas.

(Wang, Sun, Huang, Wu, & Wang, 2012) manifiesta que la mayoría de empresas de TI venden recursos *cloud* bajo un modelo de precios fijos pero dicho esquema de precios tiene muchos problemas como baja eficiencia, inflexible, más costoso y dificultad en obtener equilibrio de precios de acuerdo con la relación entre la oferta y la demanda en el mercado. Aunque los problemas anteriormente citados existen en la realidad también los autores

aclaran una posible solución puede ser la introducción de modelos económicos y basados en mercado.

(Wang et al., 2012) plantean un sistema de subasta inversa con correspondencia por lotes para que el proceso sea más eficiente, además de implementar doble castigo para impedir el comportamiento fraudulento de los usuarios con el sistema de subasta inversa se obtienen mejoras en la eficiencia, y además se puede lograr tener precios justos para el comprador y obtener beneficios para el vendedor. Para optimización usan un algoritmo evolutivo inmune con aplicación del método de aproximación de Vogel.

(Shi, Xu, Liu, & Wang, 2013) coincide con (Wang et al., 2012) en que indiscutiblemente la computación *cloud* es un paradigma orientado a mercado y en consecuencia se hace necesario un modelo de mercado eficaz y un mecanismo de fijación de precios que eviten problemas de paradigmas anteriores.

(Shi et al., 2013) afirma que aun cuando existen muchos modelos económicos que pueden aplicarse con éxito en computación *cloud* y *grid* pocos investigadores están trabajando en su aplicación real, los autores por su parte presentan la posibilidad de aplicar el modelo (CDA) subasta doble continua por sus siglas en inglés, donde se puede tener interacción permanente entre consumidores y proveedores de servicio.

CDA permite tener un juego de posibilidades mayor dado que soporta negociación uno a uno, uno a muchos y muchos a muchos. La dinámica consiste en que tanto los consumidores como proveedores pueden mejorar su puja u oferta de manera que se tiene un comportamiento desde ambas vías.

7. Caracterización de los servicios IaaS y SaaS.

El presente apartado se centra en la caracterización de los servicios IaaS y SaaS seleccionados para este trabajo. La metodología que se utilizó para la caracterización en ambos casos es idéntica y se describe a continuación.

En primera instancia se buscaron proveedores destacados en el mercado tanto para el caso de IaaS como para el caso de SaaS.

1. Se compararon los servicios ofrecidos por cada proveedor.
2. Se compararon los documentos que pudieran conformar un SLA de cada proveedor.
3. Se destacaron elementos comunes, no comunes y no considerados por parte de los proveedores en cuanto servicio y documentos.

Los elementos encontrados en la caracterización hacen parte de los recursos no ontológicos tenidos en cuenta para el desarrollo de las ontologías correspondientes para cada dominio.

7.1. Caracterización de servicios IaaS

A continuación se describe el proceso realizado según la metodología planteada.

7.1.1. Selección de los proveedores IaaS.

Para la selección de los proveedores se optó por tomar una muestra de los mayores competidores del mercado IaaS, para poder identificar los jugadores más representativos del mercado se utilizó el cuadrante mágico de Gartner, dicho cuadrante se divide en cuatro categorías que son:

1. Líderes.
2. Retadores.

3. Jugadores de Nicho.
4. Visionarios.

El cuadrante resume gráficamente la situación del mercado de un producto tecnológico en un momento determinado, para el caso específico de este trabajo, se consultó el servicio *cloud* IaaS. En el cuadrante se analiza el comportamiento de los mejores 15 proveedores del año y las cuatro categorías se ubican en un plano cartesiano donde cada uno de los cuadrantes representa una categoría.

Como insumo para la selección se tomaron los informes del grupo Gartner del año 2013, 2014 y 2015. En la tabla 4, se muestra el comportamiento del servicio *cloud* IaaS, en últimos tres años. Para la selección de los proveedores se tomaron en cuenta los movimientos en el cuadrante hasta posiciones de liderazgo.

Los proveedores seleccionados fueron:

- *Amazon* con su servicio EC2.
- *GoGrid*.
- *Rackspace*.
- *Microsoft Azure*.
- *Google Compute Engine*.

Como se observa en la tabla *Amazon* y *Microsoft*, son líderes indiscutibles en el mercado. Por otro lado se nota como *Rackspace* pasa de ser visionario a establecerse en un nicho, *GoGrid* aparece establecido en un nicho hasta 2014 y desapareciendo en 2015. Dicha desaparición se debe a que *GoGrid* fue comprada por *DATAPIPE*, competidor líder en gestión de recursos *cloud* con lo que *GoGrid* ahora pate de *DATAPIPE* termina siendo uno de los competidores más poderosos de la industria.

La selección de los servicios de *Google Compute Engine*, merece un análisis especial, el gigante de internet solo aparece en el cuadrante de visionarios hasta 2014 y mejora su

posición en la misma categoría en 2015. La razón de la selección de *Google* se centra en que su oferta de servicios es una de las más completas del mercado y que sin duda pronto estará en mejor posición que la mayoría de sus competidores.

Tabla 4. Comportamiento del cuadrante mágico de Gartner.

Cuadrante	AÑO 2013	AÑO 2014	AÑO 2015
LIDERES	AMAZON	AMAZON	AMAZON
	CSC	MICROSOFT	MICROSOFT
RETADORES	VERIZON		
	TERREMARK		
	SAVVIS		
	DIMENSION DATA		
JUGADORES DE NICHOS	TIER 3	RACKSPACE	RACKSPACE
	JOYENT	VM WARE	VIRTUSTREAM
	FUJITSU	VIRTUSTREAM	INTERROUTE
	VIRTUSTREAM	DIMENSION DATA	CSC
	HP	JOYENT	FUJITSU
	GOGRID	FUJITSU	VERIZON
	IBM	HP	JOYENT
	SOFTLAYER	GOGRID	NTT COMMUNICATIONS
			DIMENSION DATA
VISIONARIOS	MICROSOFT	CENTURY LINK	CENTURY LINK
	RACKSPACE	CSC	GOOGLE
		IBM	VM WARE
		VERIZON TERREMARK	IBM
		GOOGLE	

Además de los movimientos de los proveedores en el cuadrante de Gartner, se tuvo en cuenta la disponibilidad de documentación dado que para algunos proveedores no se

encontró una fuente de datos disponible que permitiera obtener lo necesario para la caracterización.

7.1.2. Comparación de servicios de los proveedores IaaS.

A continuación se realiza la comparación de servicios y documentos que componen un SLA de cada uno de los proveedores.

Servicio IaaS de Amazon EC2.

El servicio IaaS de *Amazon* conocido comercialmente como EC2, se presenta en instancias de máquinas virtuales con diferentes configuraciones. Sin embargo los elementos encontrados en cada configuración son los mismo variando solo los valores para ellos.

Por ejemplo en la tabla 5, se observan instancias estándar de *Amazon*. Los servicios cuentan con un nombre que sugiere su capacidad, definen la cantidad de memoria RAM que disponen, unidades de procesamiento que son equivalente a CPU, almacenamiento con sus respectivas unidades y rendimiento de entrada y salida.

Tabla 5. Instancias estándar de *Amazon*.

Tipo Instancia	RAM	UI EC2	Almacenamiento	Plataforma	Rendimiento E/S
Pequeña	1.7 GB	1	160 GB	32 / 64	Moderado
Media	3.75 GB	2	410 GB	32 / 64	Moderado
Grande	7.5 GB	4	850 GB	64	Alto
Extra grande	15 GB	8	1690 GB	64	Alto

El resto de instancias sin importar su tipología sean microisntancias, instancias con alta

cantidad de memoria, instancias para CPU de alto rendimiento, instancias informáticas para clúster o instancias de alto rendimiento para entrada y salida se definen exactamente en los mismo términos que las instancias estándar.

En Amazon no se encuentra explícitamente el nombre del procesador ni su velocidad, esto por la naturaleza heterogénea de las máquinas usadas para ese propósito. Sin embargo cuentan con una unidad de conversión que estipula que una UI EC2, proporciona la capacidad de CPU equivalente de un procesador *Opteron* 2007 o *Xeon* 2007 de 1,0-1,2 GHz. También es equivalente a un procesador *Xeon* de principios de 2006 de 1,7 GHz.

Servicio IaaS de Azure Máquinas virtuales.

Azure es la plataforma de *Microsoft* que pretende impactar el mercado *cloud* en todos sus posibles sabores, en la tabla 6, se observan los servicios IaaS que tienen que ver únicamente con la presentación de instancias virtuales.

Tabla 6. Ejemplo de máquinas virtuales *Azure*

Tipo instancia	# CPU	Velocidad	RAM
Extra pequeña	1	1 GHz	768 MB
Pequeña	1	1.6 GHz	1.75 GB
Media	2	1.6 GHz	3.5 GB
Grande	4	1.6 GHz	7 GB
Extra grande	8	1.6 GHz	14 GB

En *Azure* también es posible tener máquinas virtuales de *Linux* de manera que no solo se cierra a los productos de MS, la tabla de valores es igual en todos los aspectos tomados en cuenta en el cuadro anterior.

Para el caso de *Azure* también se cuenta con almacenamiento, sin embargo el almacenamiento es variable y se puede mejorar en cada caso. Además cuenta con un calculador de precios que permite configurar máquinas a medida.

Servicio IaaS de Google Compute Engine.

El servicio *Google Compute Engine* difiere de los demás proveedores en que es un servicio de computación de alto rendimiento, en dicho sentido se enfoca en computaciones intensivas en procesamiento. En este servicio en particular el almacenamiento se cobra de forma adicional. En la tabla 7, se ilustra un ejemplo con máquinas de configuración estándar.

Tabla 7. Ejemplo de instancias *Google Compute Engine*.

Tipo de maquina	#CPU	Velocidad	Ram
n1-standard-1	1	2.3 GHz * 2.75	3.75GB
n1-standard-2	2	2.3 GHz * 5.50	7.5GB
n1-standard-4	4	2.3 GHz * 11	15GB
n1-standard-8	8	2.3 GHz * 22	30GB
n1-standard-16	16	2.3 GHz * 44	60GB
n1-standard-32	32	2.3 GHz * 88	120GB

Al igual que *Azure*, cuenta con una calculadora que permite configurar máquinas a medida.

Servicio IaaS de GoGrid Virtual Server.

Los servicios IaaS de *GoGrid* se presentan en configuraciones estándar como se observa en la tabla 8. Para esta compañía el apartado de CPU esta expresado en núcleos de procesador, además de ese detalle no difiere de los demás proveedores en aspectos muy profundos.

Tabla 8. Ejemplo de Servicio IaaS de *GoGrid*.

Producto	RAM	Núcleos	Almacenamiento
X-Small	0.5 GB	0.5	25 GB
Small	1 GB	1	50 GB

Medium	2 GB	2	100 GB
Large	4 GB	4	200 GB
X-Large	8 GB	8	400 GB
XX-Large	16 GB	16	800 GB
XXX-Large	24 GB	24	1.2 TB

Servicio IaaS de *Rackspace* Virtual Server.

El servicio IaaS de *Rackspace*, se presentan en configuraciones estándar como se observa en la tabla 9, para el servicio ofrecido por esta compañía no se encuentra una tendencia diferente a la de los demás proveedores seleccionados para este estudio.

Tabla 9. Servicio IaaS de *Rackspace*.

RAM	Almacenamiento	CPU
512 MB	20 GB	1
1 GB	40 GB	1
2 GB	80 GB	2
4 GB	160 GB	2
8 GB	320 GB	4
15 GB	620 GB	6
30 GB	1200 GB	8

Por otro lado los planes de uso que pueden variar desde el pago por horas hasta años, en la tabla 10, se muestran los diferentes periodos de tiempo ofrecidos por los cuatro proveedores seleccionados. Se puede observar que en los cuatro se tiene una tarifa por hora y por mes, en tres casos se tienen planes a seis meses y un año.

Es importante notar que entre más grande sea el periodo de tiempo que se contrata un servicio menor será su valor, sin embargo la flexibilidad que se espera de los servicios

cloud radica en los contratos de corto tiempo incluso en fracciones de hora caso que no se contempla en los proveedores seleccionados.

Tabla 10. Comparación de periodos de facturación entre proveedores.

Proveedor	Hora	Mes	6 meses	1 año
<i>Amazon</i>	✓	✓	✓	✓
<i>Azure</i>	✓	✓	✓	✓
<i>GoGrid</i>	✓	✓	✓	✓
<i>Rackspace</i>	✓	✓	X	X
<i>Google Compute</i>	✓	✓	✓	✓

Conclusión de la comparación de los servicios IaaS:

De las tablas anteriores se tiene que todas las instancias sin excepción se pueden expresar en términos de Memoria RAM, capacidad de almacenamiento y CPU, en cuanto al servicio de computación. Sin embargo es importante notar que las instancias necesitan sistemas operativos para poder funcionar los más difundidos son *Linux* y *Windows* en diferentes versiones de productos. Para el caso de la investigación actual no es importante determinar cuál sistema operativo es el más utilizado.

En cuanto a los valores y unidades de cada componente se observa que en cuanto a memoria RAM se utilizan MB y GB siendo mayormente utilizado el GB. Por otro lado la unidad de almacenamiento más utilizada es GB, no obstante en uno de los proveedores evaluados se utiliza GB y TB como unidades. En cuanto a la plataforma se tiene que los valores son 32 o 64 bits en todos los proveedores seleccionados.

En el caso de las unidades de computación se tiene algún grado de heterogeneidad, que pueden redundar en la elección inadecuada de un servicio por ejemplo para el caso de Amazon se asume según su documentación que una unidad informática EC2 “proporciona la capacidad de CPU equivalente de un procesador *Opteron* 2007 o *Xeon* 2007 de 1,0-1,2

GHz. También es equivalente a un procesador *Xeon* de principios de 2006 de 1,7 GHz”⁷, sin embargo para otros proveedores no es tan claro pues para el caso de *Azure* se usan velocidades en GHz pero no se relacionan con ningún procesador de manera que no se sabe cuál es el rendimiento real de la instancia, en el caso de *GoGrid* es más borroso aun pues se hace referencia a núcleos de procesador (*Cores*) y no se hace referencia a ningún valor en MHz.

En el servicio de computación también puede ser un factor significativo el ancho de banda disponible para cada instancia, los mecanismos de acceso de entrada y salida, la escalabilidad, fiabilidad y otras medidas que no se aclaran en los sitios web. Por otra parte también pueden jugar un rol fundamental los contratos de los proveedores.

7.1.3. Comparación de documentos SLA IaaS.

A continuación se analizan los documentos SLA, de los diferentes proveedores seleccionados con el fin de verificar cuales son las condiciones de prestación del servicio y los indicadores clave de producto KPI, por sus siglas en español que destacan en cada uno de ellos.

Contrato de nivel de servicio Amazon EC2.

En el contrato de nivel de servicio de *Amazon*⁸, se observa claramente que se enfatiza en la disponibilidad anual del servicio que es de 99.95% y en las penalidades que se aplican para el no cumplimiento de dicho porcentaje de disponibilidad.

La penalidad aplicada por el no cumplimiento de la disponibilidad del servicio es el 10% de la factura durante el periodo de crédito, el contrato es explícito en afirmar que los créditos que se otorgan a los clientes son solo redimibles en cuentas futuras y no representan reembolsos de dinero en ningún caso.

⁷ Citado de: <http://aws.amazon.com/es/ec2/instance-types/>

⁸ <http://aws.amazon.com/es/ec2-sla/>

Contrato de nivel de servicio de proceso de *Microsoft Windows Azure*

El contrato de nivel de servicio de *Microsoft Windows Azure*⁹ también enfatiza en la disponibilidad del servicio teniendo dos tablas para dos diferentes contextos.

1. Niveles de servicio de tiempo de actividad de conectividad mensual

Tabla 11. Penalidad por tiempo de actividad de conectividad mensual.

Porcentaje de tiempo de actividad mensual	Abono de servicio
< 99,95%	10%
< 99%	25%

2. Niveles de servicio de tiempo de actividad de instancia de rol mensual

Tabla 12. Penalidad por tiempo de actividad de instancia de rol.

Porcentaje de tiempo de actividad mensual	Abono de servicio
< 99,9%	10%
< 99%	25%

El contrato es explícito en afirmar que ningún abono de servicio superara el monto de la factura mensual del cliente.

Contrato de nivel de servicio *Google Compute Engine*.

El contrato de nivel de servicio de *Google Compute Engine*¹⁰ no difiere mucho de los demás y enfatiza en el tiempo de disponibilidad del servicio y en el pago de créditos de servicio por los posibles incumplimientos por parte del proveedor. Sin embargo como se

⁹ <http://www.microsoft.com/en-us/download/details.aspx?id=24434>

¹⁰ <https://cloud.google.com/compute/sla>

observa en la tabla 13, considera una disponibilidad desde el 95%. Los demás proveedores tienen su tope mínimo en el 99%. No obstante llegan a cubrir hasta el 50% en créditos de servicio.

Tabla 13. Penalidades por incumplimiento de disponibilidad.

Porcentaje mensual de servicio	Porcentaje mensual de facturación cubierto de no cumplirse el SLO.
99.00% - < 99.95%	10%
95.00% - < 99.00%	25%
< 95.00%	50%

GoGrid SLA.

En este caso el SLA correspondiente al servicio *Cloud Servers* destaca un servicio de monitoreo para garantizar que el servicio se va a presta como realmente se acordó, su SLA es muy contundente al promover que dan un 10.000% de garantía por una falla pero que en cualquier caso el crédito no excederá el valor de la factura mensual.

Dos aspectos destacados en el SLA que son primero seguridad física en los centros de datos quizá es algo que se da por sentado pero en este contrato lo ponen por escrito y el segundo y más importante ingeniería de soporte 24/7 todo el año con unos tiempos de respuesta a requerimientos del cliente bastante buenos bien sea por chat o por teléfono.

Rackspace SLA.

El SLA de esta compañía también enfatiza en los términos que se reconocen como incumplimiento de servicio y las penalidades en las que se incurre de probarse alguna falla, es destacable que hace una referencia a la red, infraestructura, migraciones y host. Dando un porcentaje de crédito según el origen de la falla.

Al igual que los otros documentos SLA, es claro en señalar que el crédito por fallas en el servicio no será superior al monto total de un periodo de facturación.

Conclusión del análisis de los SLA en los servicios IaaS.

Usando el framework SMI los SLA de los proveedores de servicios IaaS seleccionados se enfocan en las categorías financiera y garantía, el enfoque en lo financiero es predecible pues ese aspecto es de particular importancia en los servicios *cloud*, en cuanto a la categoría de garantía también tiene un alto impacto económico pues cumplir con los tiempos del contrato asegura que el mismo se llevara a cabo sin penalidades.

Es destacable que una categoría tan importante como responsabilidad, casi no se considera. Es posible que algunos proveedores cuenten con algunas herramientas de monitoreo pero no se nota de manera explícita en ninguno de los contratos ni en la definición del servicio la posibilidad de auditoria.

7.2.Caracterización de los servicios SaaS.

A continuación se describe el proceso realizado según la metodología planteada.

7.2.1. Selección de los proveedores SaaS.

El mercado de servicios SaaS, está plagado de opciones en diferentes categorías. Sin embargo para este trabajo se seleccionó la categoría de servicios de ofimática donde sus mayores proveedores son *Microsoft* con en servicio *Office 365* y *Google* con el servicio *Apps for Business*.

7.2.2. Comparación de servicios de los proveedores SaaS.

Google Apps for Business.

“Google Apps for Business” es un paquete de aplicaciones Web que comprende correo electrónico, calendarios en línea, almacenamiento en nube, editor de texto, hoja de cálculo, y diapositivas, entre otras. Como se observa en la figura 16, el servicio está orientado a negocios de todo tamaño y en este momento llega a los cinco millones de clientes¹¹.

El servicio se presenta en dos planes que básicamente expresan la forma de facturación seleccionada por el cliente uno por pagos mensuales y otro por pagos anuales los cuales son sujetos a descuentos, por otro lado se puede acceder al servicio desde teléfonos inteligentes, tabletas y pc, dando alta capacidad de acceso a los diferentes documentos que genere la empresa.

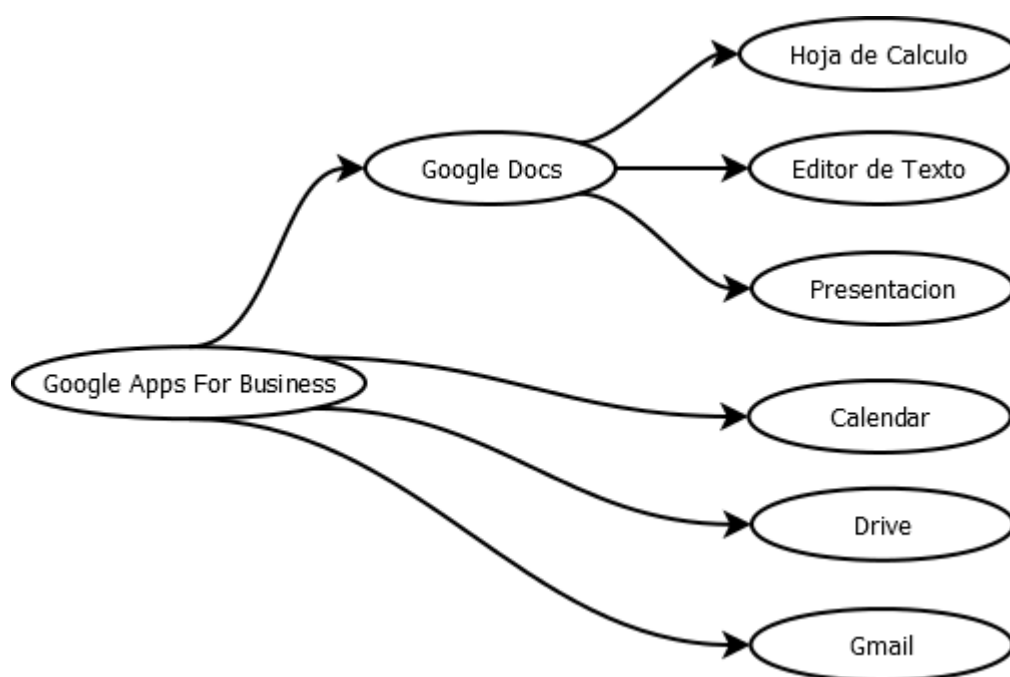


Figura 16. Características generales del servicio. Fuente: El autor.

¹¹ <http://www.google.com/intl/es-419/enterprise/apps/business/>

7.2.3. Office 365.

Office 365 es el nuevo servicio de *Microsoft* que pone sus populares aplicaciones del *Office* que ya se conocen como *Word*, *Excel* y *PowerPoint* en línea. Como se observa en la figura 17. Aun cuando el servicio tiene diferentes focos de mercado en la parte empresarial este trabajo se enfoca en el servicio para compañías medianas y pequeñas. Además de que el servicio se puede acceder en línea también cuenta con las versiones tradicionales fuera de línea, de manera que se puede usar desde internet, en computadores de escritorio y también ofrece acceso desde dispositivos móviles teléfonos y tabletas respectivamente.

El servicio se presenta para pequeñas empresas ¹² en dos variantes, la primera *Office 365* pequeña empresa y en segunda instancia *Office 365* pequeña empresa Premium, también existen planes para empresas de tamaño mediano¹³ y empresas grandes¹⁴.

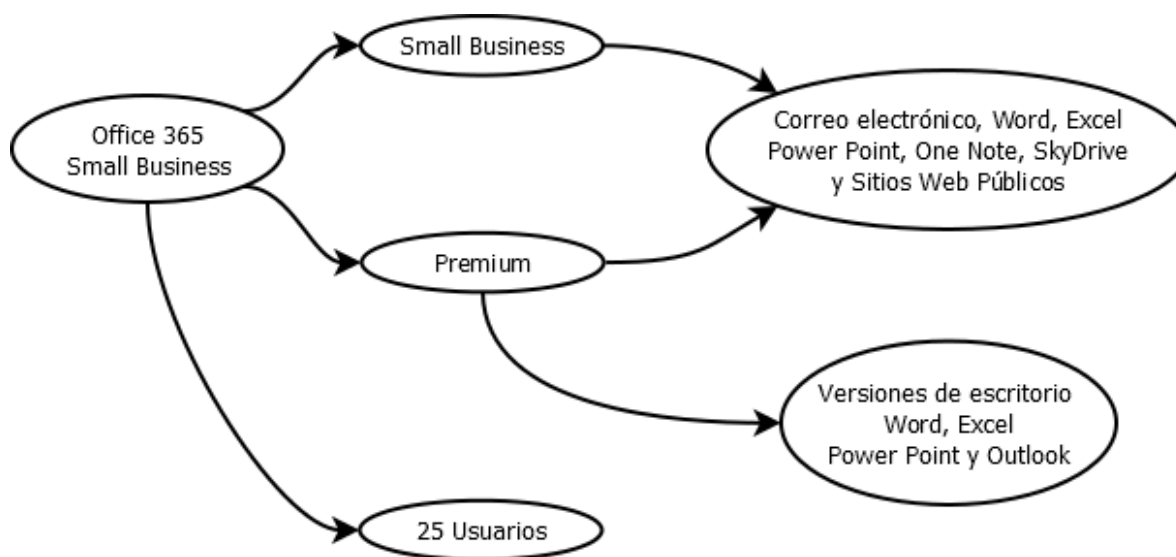


Figura 17. Características generales del servicio *Office 365* Small Business. Fuente: El autor.

¹² <http://Office.microsoft.com/en-us/business/compare-Office-for-business-plans-FX102918419.aspx>

¹³ <http://Office.microsoft.com/en-us/business/compare-Office-for-business-plans-FX102918419.aspx>

¹⁴ <http://Office.microsoft.com/en-us/business/compare-Office-for-business-plans-FX102918419.aspx>

Conclusión de la comparación de los servicios SaaS:

Después de la comparación se tiene que ambos servicios de ofimática en línea ofrecen productos similares, en una suite que de cierto modo tiende a ser compatible con la líder del mercado. Brindando soluciones de hoja de cálculo, editor de texto, editor de presentaciones y correo electrónico. Como los servicios base de la suite.

Por parte de *Google* encuentra su mayor valor en que su servicio es 100% Web y puede ser accedido desde cualquier navegador de internet sin mayores complicaciones. Además de tener todo el ecosistema *Android* a su servicio lo que hace que sea poderosamente atractivo para el uso en dispositivos móviles y aporta mucho valor la integración transparente con otras herramientas de productividad de *Google*.

Tienen un valor único por usuario lo que hace fácil estimar el costo del producto cuando se necesita más cobertura, aun cuando *Google* ofrece un servicio *cloud* de almacenamiento no agrega valor ya que su contraparte también lo ofrece.

Por el lado de *Office 365* además de tener en su versión de escritorio la parte mayoritaria del mercado, agrega valor dando a sus compradores versiones licenciadas de ese producto para 5 equipos en su oferta para pequeñas empresas.

También ofrece versiones móviles para equipos con *Windows Phone*, sin embargo este último sistema operativo no ha logrado tener gran impacto en el mercado de dispositivos móviles.

Retirando las características diferenciadoras que están localizadas solo en la forma de facturación, la oferta de servicios es casi idéntica. Por lo cual se puede determinar que los aspectos más importantes del servicio son tener una suite con cuatro aplicaciones de núcleo que son para ambos hoja de cálculo, editor de texto, editor de presentaciones y correo electrónico.

Por otra parte los esquemas de facturación de ambos aunque diferentes caben en la categoría de facturación por usuarios, en periodos mensuales.

7.2.4. Comparación de documentos SLA proveedores SaaS.

Características del contrato de servicios de *Microsoft* que incluye a los servicios de *Office.com*.

A continuación se realizara un resumen de los aspectos tratados en el contrato de servicios de *Microsoft*¹⁵.

Ámbito del contrato: Se nombran los servicios que están influenciados por el contrato, términos que deben cumplir los usuarios de los servicios donde hacen referencia al código de conducta¹⁶ de MS, también se tocan aspectos como cambio en el contrato y los tipos de cambios que se pueden esperar en el servicio.

Cuentas *Microsoft*: Se enfoca únicamente en las cuentas y los derechos sobre las cuentas de MS.

Contenido: Se enfoca en la propiedad del contenido que se cargue, almacene o transmita en los servicios de MS, por otro lado se tocan temas como el acceso a los contenidos del usuario y las acciones que puede hacer MS con el contenido de los mismos.

Cancelación de los servicios: Trata temas de terminación de servicios por varias causas como infringir el contrato o no acceder a la cuenta por más de 270 días.

Privacidad: En este apartado se encuentran el uso que puede dar MS a la información personal y de equipos de los usuarios.

¹⁵ <http://Windows.microsoft.com/es-xl/Windows-live/microsoft-services-agreement>

¹⁶ <http://Windows.microsoft.com/es-ES/Windows-live/code-of-conduct>

Interrupciones de los servicios y respaldos: Donde se explica que todos los servicios en línea, tienen interrupciones y que MS no es responsable de los daños derivados de estas.

Software: Si para el uso de algún servicio se incluye el uso de software adicional, en esta parte se encuentran los términos y usos aceptados de dicho software.

Términos adicionales de Office.com, MSN Y Bing: Se incluyen algunos términos respecto del uso de los servicios citados, en el tema de contenidos multimedia.

Servicios de pago de Microsoft: se hace referencia a la política de pagos, cargos adicionales y facturación.

Ausencia de garantías: MS no ofrece ningún tipo de garantía sobre los servicios prestados el uso de los servicios se asume bajo cuenta y riesgo del usuario.

Limitación de responsabilidad: MS solo se hace responsable del reconocimiento del valor de un mes, el cliente no tiene derecho a recibir ninguna otra indemnización por daños, perjuicios o pérdidas. Lo que incluye sin que sirva de limitación, el lucro cesante, los daños consecuenciales, especiales, indirectos, incidentales o punitivos.

Entidad contratante de Microsoft: Dependiendo del lugar donde esté ubicada la entidad contratante así mismo se acogerá a diferentes legislaciones en este apartado se indica cual es la legislación por la que se registrará el contrato.

Sitios Web de terceros: MS no se hace responsable por sitios web o servicios de terceros.

Administración de derechos digitales: Para algunos contenidos se puede solicitar de forma automática los derechos de uso multimedia.

Software Microsoft.Net Framework: Aplica licencia de Windows para el uso que se haga del mismo.

Subsistencia: Todas las cláusulas que deban perdurar aun cuando el contrato termine.

Cesión y transferencia: Solo *Microsoft* puede ceder el contrato, en ninguna caso el cliente podrá hacerlo.

Avisos: Se refiere a comunicaciones sobre los servicios.

Interpretación del contrato: El contrato se interpreta según la legislación pertinente *Microsoft* puede cambiar apartes si un tribunal lo exige.

Ausencia de terceros beneficiarios: No existen terceros que se puedan beneficiar del contrato además del cliente.

Componentes de fuentes: Las fuentes solo se permiten para mostrar o imprimir el contenido.

Soporte: Se hace referencia a los diferentes sitios de soporte.

Restricciones a la exportación: Se enfoca en la restricción para transferir productos software a países con algún embargo o prohibición.

Acuerdo de nivel de servicio para servicios en línea de *Microsoft*¹⁷.

1. **Introducción:** Hace referencia a que este acuerdo de nivel de servicio es parte un contrato mayor, dicho contrato ya fue expuesto anteriormente en este trabajo. Esta parte del contrato aplica para varios servicios pero en particular para este trabajo es relevante su aplicación a los servicios de “*Office Web Applications*”.

¹⁷ <http://microsoftvolumelicensing.com/Downloader.aspx?DocumentId=4256>

Se afirma que se da una retribución económica si no se logra mantener el compromiso de nivel de servicio, también se aborda la flexibilidad del contrato en el tiempo si se llega a renovar el servicio.

2. **Definiciones:**

- a. Monto mensual aplicable: hace referencia a los honorarios pagados por el usuario en un mes.
- b. Tiempo de inactividad: son los minutos en el mes durante los cuales el servicio no está disponible, multiplicado por el número de usuarios, con excepción del tiempo de inactividad programado.

Requisitos para tiempo de inactividad en aplicaciones de *Office* web.

Cualquier periodo de tiempo en que los usuarios no pueden utilizar las aplicaciones web para ver o editar un documento de *Office*, almacenado en un sitio “*sharePoint*” con permisos apropiados.

- c. Incidente: se refiere a un único evento o un conjunto de eventos que resultan en tiempo de inactividad.
- d. *Microsoft*: Es la entidad que firma el acuerdo.
- e. Tiempo de inactividad programado: Son los periodos de tiempo programado de inactividad para realizar actividades de mantenimiento y actualización, dichos tiempos programados de inactividad serán publicados al menos cuatro días antes de su ejecución.

- f. Servicio o servicios: son los servicios que aplican a esta parte del acuerdo y pagados por el usuario.
 - g. Crédito de servicio: es el porcentaje del monto mensual aplicable como resultado de una reclamación aprobada por *Microsoft*.
 - h. Nivel de servicio: se entiende como métricas de desempeño acogidas por *Microsoft* para cumplir en la prestación de servicio por ejemplo disponibilidad mensual.
 - i. Minutos de usuario: número de minutos mes, multiplicado por la cantidad de usuarios.
3. **Compromiso de nivel de servicio:** Se establece una fórmula para calcular el porcentaje mínimo de disponibilidad del servicio.

$$\frac{\text{Minutos de usuario} - \text{tiempo de inactividad del servicio}}{\text{Minutos de usuario}} * 100$$

Si el resultado de la formula “Tiempo de actividad” cae por debajo del 99.9% en cualquier mes entonces se puede ser elegible para recibir créditos de servicio.

Tabla 14. Penalidades por tiempo de inactividad.

Porcentaje de actividad	Tiempo de	Crédito de servicio
< 99,9		25%
< 99,0%		50%
< 95,0%		100%

4. **Reclamaciones del servicio:** En esta sección se explica el proceso para reclamar, si se incumple con el tiempo mínimo de actividad.

5. **Limitaciones:** se exponen los casos en el que SLA no aplica, por ejemplo factores fuera de control que resulten de software y hardware de terceros.
6. **Compra de varios servicios:** Si se compra más de un servicio aplicable al SLA y se tiene un incidente que causa que algunos o todos los servicios no estén disponibles se puede reclamar individualmente.
7. **Compra de varios servicios como suite:** En el caso de comprar varios servicios como parte de una suite, los créditos serán calculados por cada servicio.
8. **Excepciones y términos adicionales para servicios particulares:** esta sección cita términos para algunos servicios definidos, en el caso de aplicaciones de *Office* en línea no aplica.

De lo anterior se puede concluir que los documentos que hacen parte de un SLA para servicios SaaS, se centran en las condiciones del contrato, penalidades y servicios ofrecidos. Factores que serán tomados en cuenta para el desarrollo de la ontología propuesta.

Los procesos de negociación automática por su puesto están completamente excluidos dado que hasta el momento se encuentran en un modelo donde el mercado claramente no regula las operaciones.

Características de los términos generales del contrato de *Google Apps for Business*.

A continuación se realizará un resumen de los aspectos tratados en Los términos del servicio de “*Google Apps for Business online*”¹⁸,

1. **Servicios:** Hace referencia en términos generales a la prestación de servicios y las

¹⁸ http://www.google.com/apps/intl/en/terms/premier_terms.html

condiciones mínimas que *Google* exige para la prestación de algunos servicios.

2. **Obligaciones del cliente:** Se centra en las obligaciones que adquiere el cliente al contratar un servicio con *Google*.
3. **Facturación y pagos:** En este apartado destacan los dos planes que se ofrecen al público 1. Un plan flexible donde el cliente no se compromete a adquirir servicios a un plazo determinado pero se factura sobre la base de un mes *Google* cobra al cliente los días que utilizo el servicio, 2. Plan anual donde el cliente recibe un descuento. Por otro lado se tratan todos los aspectos relacionados con los pagos, impuestos y órdenes de compra.
4. **Servicios de soporte técnico:** Existen dos tipos en el documento uno por el cliente y otro por *Google*, en el primer caso el cliente asume el apoyo técnico y en el segundo caso si no es posible hacerlo por parte del cliente se escala a personal de *Google*.
5. **Suspensiones:** *Google* puede suspender cuentas de usuario si este incumple con el acuerdo, también puede hacer por cuestiones de seguridad.
6. **Información confidencial:** Este apartado hace referencia al trato que se le dará a la información que se considera confidencial por cada una de las partes involucradas en el acuerdo.
7. **Derechos propiedad intelectual y características de marca:** Se centra en que ni el usuario tiene propiedad sobre los derechos de autor de las aplicaciones por haber pagado, ni *Google* tiene derechos de autor sobre los documentos propiedad del cliente, el cliente puede acceder a autorizar que se muestren algunos elementos relacionados con su marca pero puede revocar ese privilegio.
8. **Publicidad:** hace referencia a la autorización que da el cliente para que *Google*

pueda incluir su marca en listas de clientes y hacer referencias verbales de la marca del cliente siempre y cuando sea autorice.

9. **Declaraciones garantías y renunciaciones:** Se centra en el poder que tienen ambas partes para suscribir el acuerdo y hace consideraciones sobre descargos de responsabilidad.
10. **Plazos (términos):** Se refieren a los plazos de tiempo por los cuales es válido el contrato y por supuesto el tiempo durante el cual se prestan los servicios. También se toca la renovación en cada uno de los planes y solicitud de terminación de cuentas de usuario.
11. **Terminación:** se refiere a la terminación del contrato se estipula una terminación por incumplimiento, se determina que cualquiera de las dos partes puede terminar el contrato por razones específicas. De terminar el contrato *Google* dará al cliente un plazo prudente para retirar todos sus documentos de su infraestructura.
12. **Indemnización:** Se define en dos vías una por parte del cliente y otra por parte de *Google* y se refiere a que cualquiera de las dos partes indemnizara, defenderá, y eximirá a la otra de toda responsabilidad que tenga que ver con la reclamación de un tercero (aplica para el cliente), o de que los servicios prestados por *Google* violan cualquier principio considerado en la ley aplicable. También se tratan posibles infracciones.
13. **Limitación de responsabilidad:** Se enfoca en el límite de las responsabilidades de cada parte firmante del contrato, así como excepciones.
14. **Varios:** Se centra en varias cosas de diversa naturaleza que pueden no quedar claras en el cuerpo del contrato y que son importantes para la firma del mismo.
15. **Definiciones:** Muestra definiciones usadas en el acuerdo, cuando es necesario se

relacionan otros documentos.

SLA de Google Apps¹⁹.

El documento consigna que la interfaz web de los servicios cubiertos estará disponible al menos un 99.9% del tiempo durante cualquier mes natural, considera penalidades en créditos de servicio si *Google* no cumple con los términos del contrato de nivel de servicio.

En el documento se definen los siguientes elementos para el cálculo de créditos de servicio. Se hace una traducción textual del documento.

1. **Tiempo de Inactividad:** significa, para un dominio, que los errores que afectan al usuario son superiores a un 5%. El Tiempo de Inactividad se calcula en función del porcentaje de errores registrados en el servidor.
2. **Servicios Cubiertos por Google Apps:** se refiere a los siguientes componentes del Servicio: *Gmail*, *GoogleCalendar*, *Google Talk*, *Google Docs y Drive*, Grupos de *Google*, *Google Sites* y *Google Apps Vault*.

Quedan excluidos los siguientes componentes del Servicio: funciones experimentales de *Gmail Labs*, *Google Apps – Postini Services* y el chat de voz o de vídeo de *Gmail*.

3. **Porcentaje de Tiempo de Actividad Mensual:** hace referencia al número total de minutos de un mes natural menos el número de minutos de tiempo de inactividad registrados durante dicho mes, dividido entre el número total de minutos del mes natural.
4. **Servicio:** Se refiere a los servicios de *Google Apps for Business* (también conocido como *Google Apps Premier Edition*), el servicio *Google Apps for Government*, el

¹⁹ <http://www.google.com/apps/intl/es/terms/sla.html>

servicio *Google Apps* for ISPs (también conocido como *Google Apps* Partner Edition), el servicio *Google Apps* for Education (también conocido como *Google Apps* Education Edition) o *Google Apps* Vault (según se aplique) proporcionados por *Google* al Cliente conforme al Acuerdo.

5. Crédito de servicio:

Tabla 15. Días añadidos por porcentaje de actividad mensual.

Porcentaje de Operatividad Mensual	Días de Servicio añadidos al final de la vigencia de este (o el valor monetario equivalente a los días de servicio para los clientes con facturación mensual posterior al uso del servicio), sin coste alguno para el cliente.
< 99,9% - >= 99,0%	3
< 99,0% - >= 95,0%	7
< 95,0%	15

En caso de que se confirme la reclamación del cliente para poder recibir los créditos de servicio deberá hacer la solicitud con un plazo máximo de 30 a partir del momento en que pueda hacerlo efectivo, si el cliente incumple dicho requisito, perderá el derecho a recibir el crédito de servicio.

Es de notar que en el SLA se estipula que el valor de los créditos de servicio para un solo mes de facturación en ningún momento podrá superar 15 días que pueden ser añadidos al final de un periodo de facturación o su valor monetario aplicado a la cuenta mensual del cliente. Los Créditos de Servicio no pueden canjearse por ninguna compensación monetaria ni convertirse en dinero, salvo para los clientes que estén suscritos al plan de facturación mensual de *Google*.

En las exclusiones el SLA enfatiza en que no aplica para los productos estipulados dentro del documento o por factores descritos como “fuerza mayor”

Conclusión del análisis de los SLA en los servicios SaaS.

Después de la comparación de los documentos SLA y contratos, se puede concluir que están enfocados en las restricciones del servicio y en las penalidades por incumplimiento, siendo los contratos mucho más densos que los documentos SLA donde se limitan a presentar las condiciones del servicio y penalidades.

Es destacable que la mayoría de los contratos sean salvavidas por parte del proveedor dejando al usuario a su suerte en muchos aspectos que pueden ser responsabilidad del vendedor. Como es de esperarse el contrato se centra en aspectos como restricciones del servicio, penalidades por incumplimiento del servicio y facturación.

8. Desarrollo de ontologías para los dominios IaaS y SaaS.

El presente capítulo se centra en el desarrollo de las ontologías que serán utilizadas como mecanismo de interacción entre los actores involucrados en el proceso de negociación, para el desarrollo de las ontologías se siguieron diferentes escenarios de la misma metodología.

En adelante se profundizara sobre las metodologías y las ontologías resultantes.

8.1. Selección de la metodología

Para el desarrollo de las ontologías se estudiaron varias metodologías que se exponen a continuación:

METHONTOLOGY muestra el proceso de desarrollo de la ontología, las tareas que recomienda desarrollar antes son:

1. Planificación de tareas.
2. Conocer las metas y el ámbito.
3. Buscar fuentes de conocimiento del dominio.
4. Realizar un modelo conceptual que describa el problema y su solución.
5. Formalizar el modelo conceptual.
6. Reutilizar en la medida de lo posible ontologías desarrolladas.
7. Implementar la ontología en algún lenguaje que permita hacerla computable.
8. Evaluar la ontología.
9. Documentar.
10. Mantener la ontología.

La metodología *METHONTOLOGY* fue utilizada con éxito en la década de los noventa en modelado empresarial y para definir dominios relacionados con la química. *METHONTOLOGY* permite desarrollar una ontología desde un boceto implementando tareas bien definidas para las cuales se recomienda usar algunas técnicas asociadas a cada una. (Fernandez, Gomez-Perez, & Juristo, 1997)

La metodología es muy recomendada para extender ontologías existentes, un tópico muy bien valorado si se tiene en cuenta que en la actualidad se cuenta con una amplia variedad de recursos ontológicos que pueden ser utilizados para ampliar un dominio o como parte de un dominio más grande.

ON-TO-KNOWLEDGE la metodología plantea las siguientes etapas:

1. Definición de metas.
2. Levantamiento de requerimientos.
3. Proceso iterativo de refinamiento.
4. Evaluación de la ontología en el dominio de aplicación.
5. Mantenimiento.

Más que una metodología para el desarrollo de ontologías, *ON-TO-KNOWLEDGE* incluye un conjunto de herramientas que en cierto modo debían soportar tecnologías de la Web semántica. Bajo ese precepto un grupo de instituciones como Telecom británico, la universidad holandesa de Vrije y el instituto AIFB alemán se unieron para el desarrollo del proyecto. (Sure, 2002).

Se utilizó en casos industriales en dominios como seguros, telecomunicaciones y energía. Por otro lado se usó por la compañía de seguros Swiss Life, con la esperanza de construir un sistema de memoria organizacional.

ONTOLOGY DEVELOPMENT 101, es una metodología sencilla pero muy útil. Implementa las siguientes actividades.

1. Determinar el dominio y alcance de la ontología.
2. Reutilizar ontologías existentes.
3. Enumerar términos importantes para la ontología.
4. Definir las clases y jerarquía de clases.
5. Definir las propiedades de las clases (*Slots*).
6. Definir las facetas de los *Slots*.
7. Crear instancias.

Para ilustrar la metodología *ONTOLOGY DEVELOPMENT 101*, los autores utilizaron quizá el ejemplo más memorable en cuanto a creación de ontologías se refiere. Dicho ejemplo es el de la ontología de Pizas. Como tal *ONTOLOGY DEVELOPMENT 101* enmarca un grupo de recomendaciones más que de tareas definidas, por lo cual se adapta muy bien a personas que son principiantes en el desarrollo de las mismas. (Noy & McGuinness, 2001)

En la tabla 16, se ilustra la comparativa adaptada de (Oscar Corcho, Fernández-López, & Gómez-Pérez, 2003), en la tabla original destaca *ON-TO-KNOWLEDGE*, sin embargo se añade la columna de la metodología *NeOn* dado que no existía cuando se desarrolló el trabajo. Como se puede observar la metodología *NeOn* abarca todas las categorías planteadas en la comparativa.

NEON (Suárez de Figueroa Baonza, 2010) es una metodología para la construcción de redes de ontologías, la cual tiene guías que se adaptan a cualquiera de los siguientes escenarios.

1. Desarrollo de la ontología desde cero, desde la especificación de la aplicación.
2. Reutilización y reingeniería de recursos no ontológicos.
3. Reutilización de recursos ontológicos.
4. Reutilización y re-ingeniería de los recursos ontológicos.
5. Reutilización y fusión de recursos ontológicos.
6. Reutilización, fusión y reingeniería de recursos ontológicos.
7. Reutilización de los patrones de diseño de ontologías.
8. Reestructuración de recursos ontológicos.
9. Localización de recursos ontológicos.

Cada uno de los nueve escenarios planteados tiene una metodología definida con tareas que permiten poder cumplir los objetivos propuestos.

Tabla 16: Comparación de metodologías para el desarrollo de ontologías. Adaptado de: (Oscar Corcho, Fernández-López, & Gómez-Pérez, 2003)

Característica			<i>Cyc</i>	<i>Usdhold and King</i>	<i>Gruninger and Fox</i>	<i>KACTUS</i>	<i>METH ONTOL OGY</i>	<i>SENSUS</i>	<i>ONTo Knowl edge</i>	<i>NEON</i>
Proceso de gestión de proyecto	Inicio del proyecto		No	No	No	No	No	No	Si	Si
	Monitoreo y control de proyectos		No	No	No	No	Si	No	Si	Si
	Gestión de calidad de la ontología		No	No	No	No	No	No	Si	Si
Proceso de desarrollo orientado a las ontologías	Proceso predesarrollado	Exploración de conceptos	No	No	No	No	No	No	Si	Si
		Sistema de localización	No	No	No	No	No	No	No	Si
	Proceso de desarrollo	Requerimientos	No	Si	Descrito	Si	Descrito en detalle	Si	Si	Si
		Diseño	No	No	Descrito	Descrito	Descrito en detalle	No	Si	Si
		Implementación	Si	Si	No	Si	Descrito en detalle	Descrito	Si	Si
	Proceso de postdesarrollo	Instalación	No	No	No	No	No	No	No	Si
		Operación	No	No	No	No	No	No	No	Si
		Soporte	No	No	No	No	No	No	No	Si

		Mantenimiento	No	No	No	No	Si	No	Si	Si
		Retiro	No	No	No	No	No	No	No	Si
Proceso integral	Adquisición de conocimiento		Si	Si	Si	No	Descrito en detalle	No	Si	Si
	Verificación y validación		No	Si	Si	No	Descrito en detalle	No	Si	Si
	Gestión de la configuración de la ontología		No	No	No	No	Descrito en detalle	No	Si	Si
	Documentación		Si	Si	Si	No	Descrito en detalle	No	Si	Si
	Entrenamiento		No	No	No	No	No	No	No	Si

Tomando como base el trabajo de (Oscar Corcho, Fernández-López, & Gómez-Pérez, 2003) se observa que las metodologías comparadas son más o menos parejas, teniendo como destacada a *ON-TO-KNOWLEDGE*, sin embargo la metodología *NeOn* posterior a la comparativa tiene un enfoque que cubre todas las categorías de la propuestas superando ampliamente al resto.

Para el desarrollo del presente trabajo se seleccionó la metodología *NeOn* por ser basada en escenarios lo que permite que se adapte a una variedad amplia de situaciones. Para el caso particular de este trabajo ya existen acercamientos de ontologías para definir los servicios IaaS y SaaS, con algunas diferencias que se expondrán en los trabajos relacionados.

Por otra parte la metodología *NeOn*, cuenta con una extraordinaria documentación, y directrices disponibles para que la implementación y desarrollo de la ontología sea dirigido por las tareas establecidas en cada escenario.

Como herramienta para el desarrollo de la ontología se utilizó *Protege*, el cual es un motor de ontologías gratuito disponible en protege.stanford.edu. Desarrollada y mantenida por “*Stanford Center for Biomedical Informatics research at the Stanford University School of Medicine*”. *Protege* es una herramienta ampliamente difundida llegando a formar una comunidad de 244817 usuarios registrados a noviembre de 2015 según datos encontrados en su página oficial.

Entre otras características cuenta con una interfaz intuitiva y una serie de complementos que se pueden utilizar para propósitos diversos como diferentes tipos de gráficos, razonadores, arboles de navegación entre otros.

8.2. Desarrollo de ontologías para servicios IaaS

Como resultado de la caracterización de los servicios ofrecidos por los diferentes proveedores y sus documentos SLA y contratos se obtuvo la ontología que se muestra en la figura 18, las clases derivan de la clase *thing* y están unidas a ella por flechas de línea

sólida. Las subclases están ligadas a las clases de las cuales dependen por flechas de línea discontinua.

Las clases utilizadas en la ontología se pueden categorizar si se quiere en clases que hacen referencia a elementos propios del servicio, clases propias de un documento SLA y clases para el proceso de negociación.

Clases propias del servicio.

AccessMethod: Esta clase tiene como objetivo representar cual o cuales métodos de acceso tiene el servicio, según la caracterización los métodos de acceso más comunes son las API, los *Web Services* y la interfaz de línea de comandos. Cada una de ellas representadas en subclases.

HardwareResource: La clase recursos de hardware tiene que ver con los recursos físicos que componen el servicio, a saber CPU, almacenamiento y memoria RAM cada uno representado por subclases en la ontología.

SoftwareResource: Define los recursos de software que se usan en el aprovisionamiento de la máquina, para el caso particular de esta tesis solo se tuvo en cuenta el sistema operativo que se representa con una subclase. Sin embargo es posible extender la ontología para incluir utilidades, librerías y otras categorías de software sin que ello afecte la concepción dada en este trabajo.

VirtualMachine: La clase que representa la máquina virtual sirve como referencia para saber el nombre del motor de virtualización y que versión se usa. Datos importantes para el proceso de aprovisionamiento que no se toca en este trabajo.

NetworkResources: Hace referencia a los recursos de red utilizados que vienen incluidos en el servicio.

Clases que conforman un documento SLA.

Administration: Esta clase hace referencia a la administración del servicio que para el caso puede ser administrada por el usuario o por un tercero.

Parties: Esta clase establece las partes firmantes del contrato a saber el consumidor y el proveedor.

Penalties: La clase penalidades se encarga de expresar las excepciones para las cuales el servicio presenta penalidades y la compensación por dichas penalidades.

Scope: La clase ámbito define los servicios a los cuales se les aplicara el SLA.

SLO: La clase SLO hace referencia a los objetivos del servicio, para el caso particular mostrado solo existe uno que es disponibilidad, sin embargo hay un conjunto amplio de SLO que pueden ser utilizados haciendo una extensión de la ontología.

ValidityPeriod: Hace referencia al periodo de validez del contrato de servicios, incluye la fecha de inicio y final que suscribe el periodo de tiempo donde SLA será válido para comprador y consumidor.

Clases que soportan el proceso de negociación.

NegotiationParameter: La clase de parámetros de negociación se centra en los valores máximo y mínimo para un servicio y las ofertas mínima y máxima que se pueden ejecutar en el proceso.

Como se puede observar, para el desarrollo de la ontología se abordaron varios aspectos, que hacen que la ontología sea muy funcional, pues define con claridad el dominio IaaS y los elementos necesarios de un SLA para dicho dominio. Además de permitir incorporarse

a una red de ontologías también se pueden eliminar elementos para usarla en un dominio más específico.

Además de las clases y subclases se plantean relaciones entre los conceptos que pertenecen al dominio del servicio, aun cuando el uso que se le atribuye a la ontología en este trabajo está situada en la recopilación de información verdadera, es decir que es tomada directamente de la fuente sin utilizar inferencias. También es posible utilizar la ontología para realizar deducciones usando razonadores.

La ontología presentada en la figura 18, se encuentra disponible en <http://ontohub.org/ontocloud/SLA-IaaS-Template.owl> donde es posible descargarla en archivo XML-OWL, adicionalmente se le ha generado documentación usando la herramienta (Live OWL documentation environment) disponible en <http://www.essepuntato.it/lode>

Cabe resaltar que dentro de la investigación se encontraron ontologías que se reutilizaron pero la idea de poner una red de ontologías que incorporaran el dominio de servicios IaaS junto con el dominio de un SLA es completamente original y no se encontró otra similar.

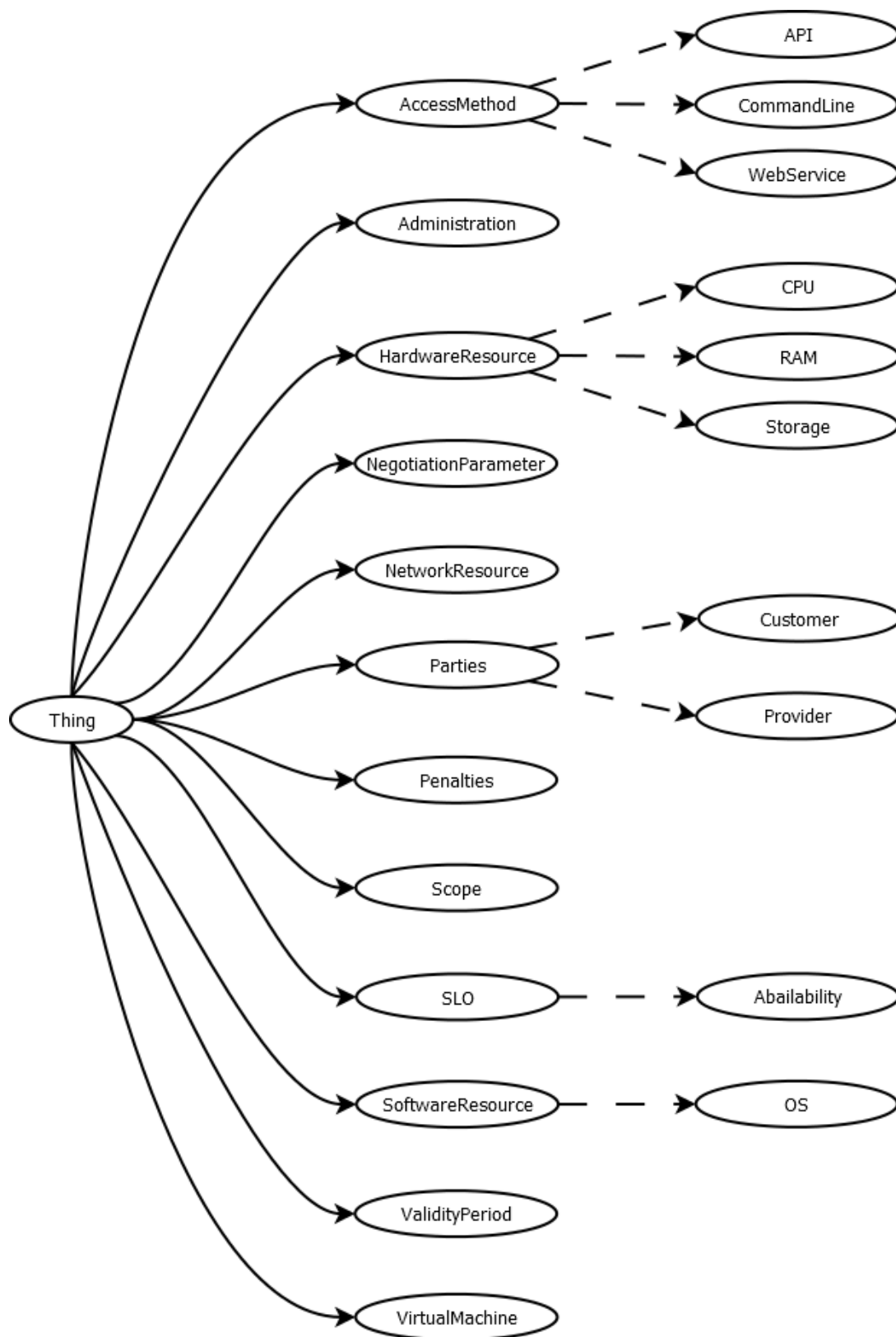


Figura 18. Ontología para la definición del dominio SLA para servicios IaaS. Fuente: El autor.

8.3. Desarrollo de ontología para servicios SaaS

De acuerdo con lo encontrado en la caracterización del servicio seleccionado, es necesario determinar algunas situaciones que se hallaron durante la investigación. En primera medida existe una gran variedad de productos SaaS, algunos se ofrecen de manera modular y otros como un solo producto.

En segunda instancia, los contratos se adaptan al tipo de producto software que se está vendiendo, de esa manera es posible encontrar restricciones del servicio de diferente naturaleza.

Para el caso particular de la ontología para servicios SaaS, se tomó como base la ontología para servicios IaaS, en lo que tiene que ver con el dominio propio del SLA y con el proceso de negociación. Se determinó que el servicio debe quedar definido en la clase *scope* de la misma.

La única extensión que se realizó, se encuentra ubicada en la clase *AccessMethod* donde se quitaron las sub clases *Api* y *CommandLine* y solo se dejó *WebService*. Adicionalmente se incluyó la clase *MobileApp* la cual representa si el servicio tiene acceso por aplicaciones móviles.

La ontología realizada se encuentra disponible en <http://ontohub.org/ontocloud/SLA-SaaS-Template.owl>.

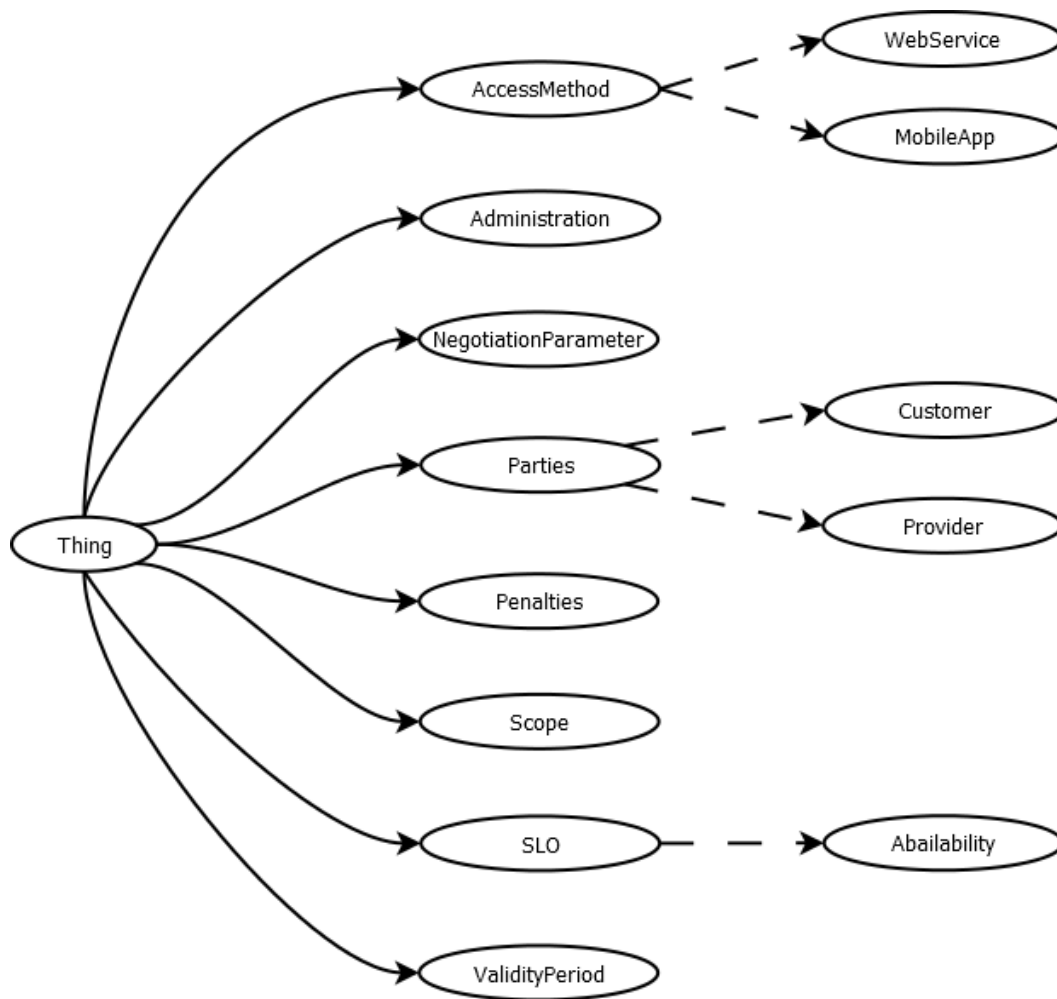


Figura 19. Ontología para la definición del dominio SLA para servicios SaaS. Fuente: El autor.

9. Marco de negociación.

El presente capítulo se centra en la conceptualización y propuesta de un *framework* para la negociación de recursos *cloud*, en el que se incluyen todos los elementos abordados en este trabajo. En un principio se hace referencia al dominio de negociación que sirve como insumo para determinar el comportamiento de los agentes que llevan a cabo el proceso de negociación.

Por otra parte se presenta un protocolo de negociación inspirado en algunos protocolos referentes que se presentan como parte de este apartado. Para terminar se hace referencia al marco de negociación completo.

9.1. Definición del espacio de negociación

(Lomuscio et al., 2003) considera que los mecanismos de negociación dependen en gran medida de las características del espacio de negociación, en la figura 20 se muestra los elementos que deben tenerse en cuenta para definir el espacio de negociación.



Figura 20. Elementos del espacio de negociación. Fuente: El autor.

Dominio de negociación.

En la oferta actual de servicios *cloud* como se nota en capítulos anteriores existen diferentes atributos que son importantes en el momento de decidirse por un servicio determinado. En general el consumidor espera que el proveedor pueda satisfacer sus requerimientos por ejemplo: precio, disponibilidad, garantías entre otros. En dicho sentido el presente trabajo tienen un dominio de negociación de múltiples atributos como el precio, garantía y disponibilidad.

Interacciones.

Las interacciones en el proceso pueden ser de uno a uno, uno a muchos y muchos a muchos. Para el presente trabajo se tendrán interacciones uno a uno, pues si bien el proceso involucra muchos vendedores y muchos compradores en el momento de realizar la negociación se habilitan parejas [vendedor – comprador] de modo que si algún otro agente está interesado en el mismo servicio deberá esperar a que la negociación actual finalice.

Características de los agentes.

El mecanismo natural para un protocolo de negociación automático, con tanta diversidad de oferentes y consumidores es la negociación basada en agentes. En la definición del espacio de negociación que se muestra en la figura 20, se muestran las características que deben tener los agentes.

A continuación se hace una breve descripción de cada una de las características que deben tener los agentes, para poder desempeñar las funciones asignadas dentro del protocolo de negociación.

Rol

En el presente trabajo se consideran agentes con 3 roles. 1. Vendedor, 2. Comprador y 3. Auditor.

Racionalidad.

Los agentes contarán con racionalidad limitada dado que los agentes de este trabajo no van a tener la capacidad de realizar grandes cálculos en tiempo constante, es decir en el presente trabajo los agentes realizarán una cantidad de cálculos limitada con la que deberán decidir si continuar o se retiran de una negociación.

Conocimiento.

En cuanto al conocimiento, los agentes solo tendrán conocimiento sobre los servicios por los cuales desean ofertar.

Compromiso.

Los agentes están obligados a actuar de acuerdo a la o las estrategias de negociación propuestas, y su mayor compromiso es lograr un acuerdo siempre y cuando la negociación se ajuste a lo esperado por las partes.

Comportamiento.

El comportamiento de los agentes en el framework será individual egoísta y tendrán como objetivo maximizar su propio beneficio.

Estrategias de oferta.

Las estrategias dependen de cada agente, sin embargo se propone que en la primera iteración el vendedor proponga su valor máximo de venta con base en el histórico de negociaciones, por otro lado el comprador deberá proponer su valor mínimo de compra de esa manera se puede delimitar el intervalo en el cual es posible llegar a un acuerdo y ajustar las estrategias de ser necesario.

Entorno

El entorno de la negociación es dinámico dado que las condiciones de la misma son cambiantes y los servicios en si también están sujetos a cambios.

Parámetro de evento:

Validación de oferta: las ofertas serán validadas según las reglas que se determinen en el protocolo de negociación.

Visibilidad de oferta:

Las ofertas solo serán visibles en el momento de la negociación por los agentes implicados en el proceso, si se consigue un acuerdo entonces los detalles de la negociación serán guardados en el historial de negociaciones exitosas y serán tomados como base para las ofertas en negociaciones futuras.

Asignación temporal del bien y cierre de la negociación:

Los servicios no serán asignados temporalmente únicamente serán asignados cuando se llegue a un acuerdo por parte de los agentes, la negociación se puede cerrar en varios casos.

- i. Cuando el tiempo para llegar a un acuerdo se excede, en cuyo caso no hay acuerdo.
- ii. Cuando se consigue un acuerdo.

Parámetros de información.

Los agentes involucrados en la negociación, tendrán el historial de negociaciones que son comunes y que deben ser consultados por cada agente para determinar su oferta inicial.

Los agentes consumidores expondrán públicamente los requerimientos que tienen y con base en dichos requerimientos y el historial de negociaciones los agentes vendedores pueden postularse para negociar con el cliente.

Los agentes vendedor y consumidor intercambiaran información sobre los servicios, disponibilidad del servicio, costo y penalidades.

9.2. Protocolos de negociación relacionados.

A continuación se presentan algunos protocolos de negociación que se tomaron como referencia para la propuesta de protocolo que será presentada en el presente trabajo.

En (Kijthaweesinpoon & Zhou, 2008) se propone un marco unificado de negociación para comercio electrónico, que soporta modelos de negociación de diferentes tipos. Los autores manifiestan que muchos de los protocolos usados soportan únicamente negociación distributivas o subastas distributivas que tienen su base en la comparación de precios, siendo el precio solo uno de los aspectos fundamentales de la negociación. Por otro lado en la negociación integrativa se debe promover un intercambio de información que permita al agente tomar decisiones sobre los diferentes atributos de la negociación, incluido el precio.

El marco general que proponen (Kijthaweesinpoon & Zhou, 2008) presenta elementos básicos presentes en casi todas las arquitecturas de negociación agentes como se muestra en la siguiente figura 21, sin embargo su aporte radica en que no es un marco de propósito específico o para un solo tipo de negociación lo que permite adaptar protocolos.

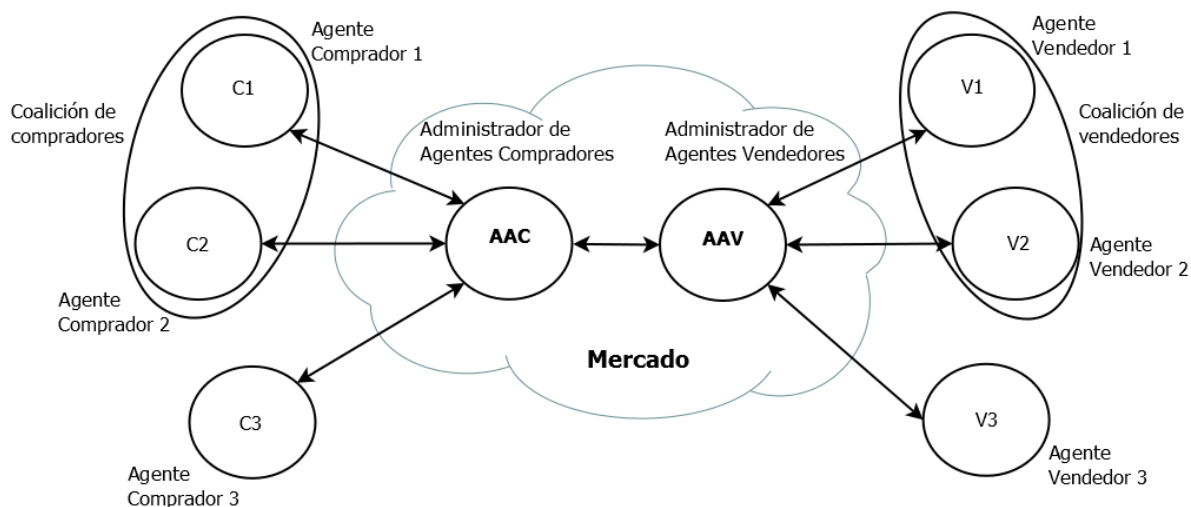


Figura 21. Marco general de negociación. Fuente Kijthaweesinpoon & Zhou, 2008.

En la literatura se encuentran muchos protocolos de negociación, que soportan negociación unilateral, multilateral y coaliciones. Sin embargo como afirma (Kijthaweesinpoon & Zhou, 2008) la mayoría se basa en subastas, en ese tipo de protocolos la negociación solo va en una vía, de manera que los bienes solo son vendidos al mejor postor. Si bien dicho enfoque es válido, para el presente trabajo carece de sentido pues implica que un usuario debería pujar cada vez que desea comprar un servicio mejorando su oferta pero sin recibir una mejora por parte del proveedor.

Por otro lado modelos como el propuesto por (Buyya et al., 2008), tienen un elemento regulador que es el mercado pero se parte de la premisa de que se tiene un mercado global de servicios donde todos los proveedores se acogen a las mismas reglas, escenario que por el momento no es realista.

El *Conflict-Aware Negotiation Protocol* (CANPRO) es un protocolo que según (Netto, 2011) siempre que un proveedor recibe la petición de recursos de un usuario devuelve un mensaje en el que se describe:

1. Si la capacidad requerida se encuentra disponible.
2. El porcentaje de la capacidad requerida que podría estar en conflicto con otros usuarios que están negociando al mismo tiempo.

3. El número de usuarios con los que se puede entrar en conflicto en la negociación.

La mayor ventaja del protocolo es que el usuario tiene más tiempo para determinar si cierra o no la negociación, además de que el protocolo ofrece mensajes que permiten saber si los recursos por negociar presentan alto nivel de conflicto, entendiendo el termino conflicto como un recurso o grupo de recursos que están siendo negociados al mismo tiempo por varios consumidores (Netto, 2011).

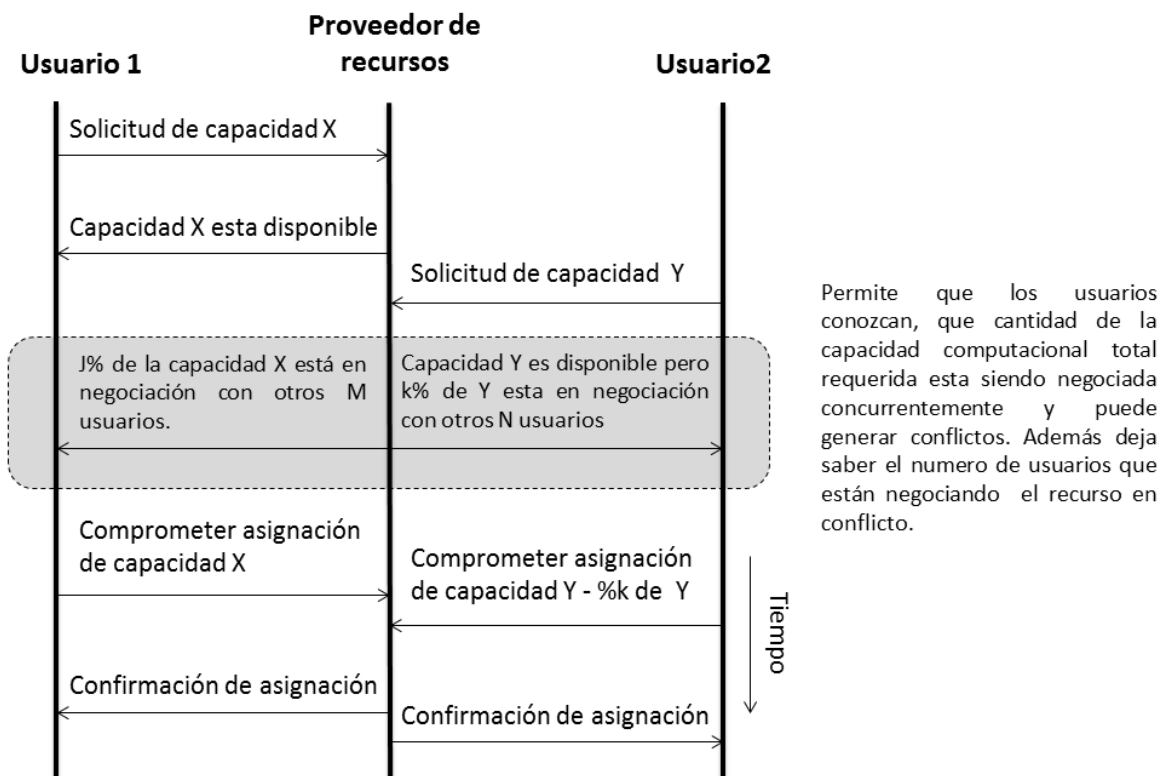


Figura 22. Conflict-Aware Negotiation Protocol Fuente: Netto 2012

En (Pan, 2011) se propone el protocolo de negociación monótono de concesión bilateral multi-pasos. El cual se expone en adelante. En la tabla 17, se observan los elementos que componen el protocolo en la columna de la izquierda se representan los términos formales y en la de la derecha la descripción de cada termino.

Tabla 17. Elementos del protocolo de negociación bilateral multipasos presentado por (Pan, 2011)

Termino Formal	Descripción
a	Agentes que pueden representar a proveedores y consumidores.
$a \in \{c, p\}$	Conjunto al que puede pertenecer el agente
ru_a	Grado en el que el agente no se dará por vencido en la negociación
a^-	Oponente del agente a
T_a	Tiempo límite del agente a , para lograr la negociación
T_{a^-}	Tiempo límite del agente a^- , para lograr la negociación
$T = \{0, 1, \dots, \min\{T_a, T_{a^-}\}\}$	Tiempo en el cual se puede llevar a cabo la negociación
$O_{t,c \rightarrow p}$	Oferta sobre parámetros de QoS multidimensional propuesta por el agente de consumidor c , para el agente proveedor p , en el tiempo t
$Up(O_{t,c \rightarrow p})$	Valor de utilidad
$O_{a,0}$	Oferta inicial de un agente para maximizar su beneficio.

Un agente proveedor recibe una oferta de un agente consumidor con un tiempo para la negociación t ($O_{t,c \rightarrow p}$), este califica la oferta usando su función de utilidad y obtiene un valor de utilidad, si $Up(O_{t,c \rightarrow p})$ es mayor o igual que la propuesta actual del agente p entonces, el agente p acepta la oferta en el tiempo t y la negociación termina exitosamente con un acuerdo.

El agente c puede realizar una evaluación similar a la oferta del agente p , cualquier agente a puede tomar la decisión de aceptar la oferta de su oponente y se alcanzara un acuerdo, si ambos agentes aceptan las ofertas de sus oponentes se puede seleccionar una oferta al azar para el acuerdo, si ninguno de los agentes acepta la oferta de su rival se da una ronda siguiente $t + 1$, en la que los agentes tendrán que volver a realizar una oferta mejorando la

anterior o retirarse si la línea de vida de la negociación es alcanzada, si el tiempo de negociación termina y no se logra el acuerdo el proceso terminara con una falla.

Existen muchos protocolos que se pueden adaptar para realizar el proceso de negociación, sin embargo el trabajo de Pan 2011, es muy afín con el problema tratado en este trabajo por lo que se extrajeron varias ideas de dicho protocolo. Por otra parte existen características deseables a la hora de elegir un protocolo de negociación como:

1. Soportar ofertas y contraofertas.
2. Soportar restricciones de tiempo.
3. Soportar agentes con diferentes roles.
4. Asegurar que en cada iteración se está más cerca de lograr un acuerdo.

9.3.Marco de negociación propuesto.

Habiendo definido el espacio de negociación, se pueden encontrar similitudes con los modelos negociación de multiagentes bajo restricciones de tiempo y negociación uno a uno en dominios mezclados presentados en (Lomuscio et al., 2003), sin embargo el espacio de negociación presentado no se sitúa específicamente en dichos modelos, en el presente trabajo se tomaran diferentes elementos de los modelos citados.

En el proceso de negociación se hace necesario tener reglas claras, que permitan a los actores involucrados saber cuáles son los objetos relacionados en el proceso. Y como deben actuar en cada fase del mismo. Los protocolos de negociación se concentran en protocolos iterativos y de una pasada. Los protocolos iterativos permiten tener mejores resultados, sin embargo toman más tiempo de ejecución. Por otra parte se encuentran los protocolos de una pasada o una sola vez, que en contraste con los iterativos toman mucho menos tiempo de ejecución aun cuando sus resultados no son tan buenos que los de los protocolos iterativos.

Para el presente trabajo se propone un marco de negociación que tenga en cuenta el espacio de negociación, comportamiento de los agentes, naturaleza de los servicios y atributos sobre los cuales se realizará la negociación. Tomando como base el esquema de clasificación en comercio electrónico propuesto por (Lomuscio et al., 2003), se tienen similitudes con los modelos negociación de multiagentes bajo restricciones de tiempo y negociación uno a uno en dominios mezclados.

Protocolo propuesto.

Para el presente trabajo se propone un marco de negociación dividido en dos grandes fases denominadas cara interna y cara externa, basado en algunos modelos y arquitecturas mostrados en capítulos anteriores y el estado del arte.

Para el proceso automático de negociación, se utilizan agentes que representan roles asociados con el conjunto de actores descrito en capítulos anteriores. Los agentes reciben su denominación según el rol que desempeñan.

AV = Agente que actúa en nombre del vendedor.

AC = Agente que actúa en nombre del consumidor.

AB = Agente que actúa como bróker

En adelante se expondrán los elementos del protocolo de negociación propuesto en primera instancia se mostrará la cara externa del marco de negociación, que tiene que ver con todas las precondiciones necesarias para que se pueda llevar a cabo la cara interna del marco de negociación. Por otra parte la cara interna del protocolo corresponde al protocolo propiamente dicho.

Cara externa del marco de negociación.

El protocolo externo se desarrolla como se muestra en la figura 23, esta parte del protocolo se centra en los preparativos para que la negociación se lleve a cabo. Dichos preparativos se expresan en 11 pasos que son expuestos a continuación.

Paso 1: Los compradores expresan sus requerimientos usando la plantilla de SLA propuesta (ontología del dominio respectivo).

Paso 2: Los vendedores expresan los diferentes servicios que pueden prestar por medio de su correspondiente plantilla de SLA (ontología del dominio respectivo).

Paso 3: Creación del agente AC, con base en lo expresado en la ontología.

Paso 4: Creación del agente AV, con base en lo expresado en la ontología.

Paso 5: El AC valida los parámetros de negociación con el archivo histórico de negociaciones y ajusta sus valores de ser necesario.

Paso 6: El AV valida los parámetros de negociación con el archivo histórico de negociaciones y ajusta sus valores de ser necesario.

Paso 7: Los requerimientos de los AC son publicados en un tablero.

Paso 8: El tablero lanza un mensaje a todos los AV.

Paso 9: Los agentes interesados en prestar el servicio se postulan notificando al tablero, el tablero organiza a los AV interesados poniendo en primera posición de negociación a los AV que primero se postulen.

Paso 10: El AC entrega a un intermediario AB el estado inicial de la negociación.

Paso 11: El AV entrega a un intermediario AB el estado inicial de la negociación.

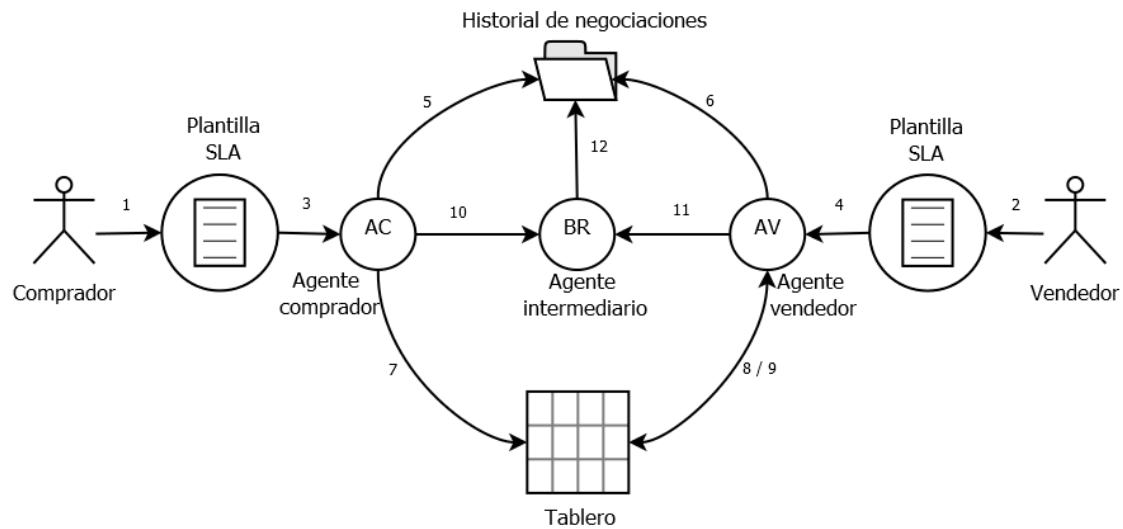


Figura 23. Esquema del marco de negociación propuesto. Fuente: El autor.

Cara interna del marco.

La cara interna del marco de negociación que se propone continuación no presenta restricciones de tiempo, para lograr que los acuerdos puedan ser alcanzados se define un espacio de negociación que está determinado por el valor más alto de venta del AV y el valor más bajo de compra presentado por el AC, este razonamiento permite que se conozca en la primera oferta y contraoferta el área de negociación, en adelante los agentes tienen que buscar llegar a un acuerdo sin embargo no se puede asegurar que puedan lograrlo, dado que sus límites de negociación son privados. De acuerdo con lo anterior se presentan 2 escenarios.

1. Escenario de intercepción.

El escenario de intercepción es el único posible para llegar a un acuerdo, en dicho escenario el AV y el AC, presentan un área común en su dominio de negociación, como se muestra en la figura 24. Es posible llegar a un acuerdo siempre y cuando ambos agentes logren llegar al área de interés común.

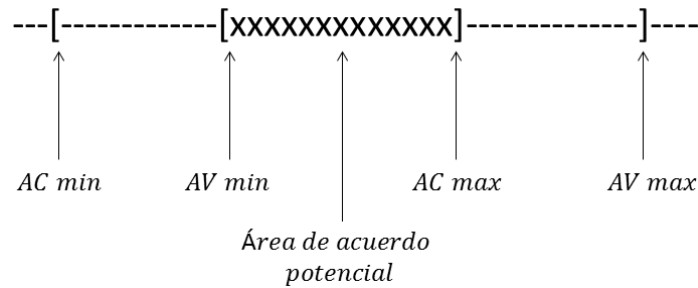


Figura 24. Escenario de intersección de ofertas. Fuente: El autor.

2. Escenario disjunto.

El segundo escenario se presenta cuando el AV y el AC no presentan un área de interés común en cuyo caso nunca podrá haber un acuerdo como se muestra en la figura 25. Para tratar de evitar al máximo conjuntos disjuntos en el dominio de negociación el agente debe basar sus ofertas en el historial de negociaciones de servicios similares, si no existen datos históricos los agentes son libres de realizar la oferta que consideren.

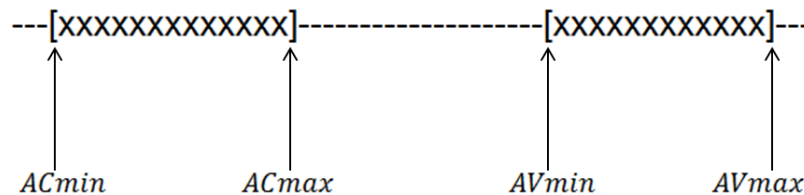


Figura 25. Escenario de disjunto de ofertas. Fuente: El autor.

Definición del protocolo interno de negociación.

Después de haberse desarrollado la cara externa del protocolo, están dadas todas las garantías para que se pueda iniciar un proceso de negociación que potencialmente derivará en un acuerdo.

Cabe destacar que las ontologías en este sentido se utilizan para dos propósitos específicos el primero es para expresar las necesidades del cliente en términos establecidos para el dominio y la segunda es como plantilla del contrato de manera que en un principio se tiene como mecanismo de comunicación entre los agentes y al final como una plantilla para el contrato de nivel de servicios SLA.

La cara interna del protocolo consta de los siguientes cuatro pasos.

Paso 1: El AC comunica al AB los datos para iniciar una interacción con un AV.

Paso 2: El AV comunica al AB los datos para iniciar una interacción con un AC.

Paso 3: El AB evalúa las ofertas de ambos AC y AV, si están en área común se considera logrado el acuerdo. Si las ofertas no se encuentran en área común el AB informa a ambos AV y AC para que mejoren sus ofertas o se retiren de la negociación.

Paso 4: Si se logra un acuerdo el AV confirma la disponibilidad de recursos, asigna los recursos si están disponibles o declina del acuerdo sino existen recursos disponibles, si el AV se retira del acuerdo se dará paso en la negociación a un nuevo AV.

A diferencia del protocolo presentado por (Pan, 2011) en el presente trabajo no se consideran limitaciones de tiempo, para controlar que las interacciones no sean demasiado extensas se propone un tope máximo de 3 ofertas y contra ofertas en las cuales se deberá lograr un acuerdo de lo contrario se da paso a una nueva ronda de negociación con un nuevo AV.

Como se puede observar el protocolo de negociación, encierra diferentes partes que se ajustan al problema de compra y venta de servicios *cloud*, tanto para IaaS como para SaaS. El uso de ontologías es muy importante dado que es un mecanismo natural caso de implementación para que los agentes puedan tener el conocimiento básico sobre el dominio de negociación.

Por otra parte el uso de ontologías permite estandarizar un vocabulario que permita que los proveedores y consumidores puedan expresar sus necesidades de la misma forma y haciendo mucho más ágil la consecución de recursos.

10. Resultados del trabajo:

A continuación se hace un recorrido por los diferentes resultados obtenidos en el trabajo, en primer lugar se citan los objetivos conseguidos por objetivos específicos y en segunda instancia se muestran los objetivos de difusión conseguidos.

Resultados concernientes a los objetivos.

A continuación se enuncian los resultados obtenidos de acuerdo con los objetivos planteados.

1. Definir los mecanismos que permitan la interacción entre los actores.

Se logró definir un conjunto de actores fundamentales para el proceso de negociación como se puede observar con más detalle en el capítulo 5.

Actores.

- Consumidores: Clientes que requieren servicios *cloud*.
- Intermediario (opcional): Persona o compañía que negocia en nombre de un cliente.
- Proveedor de servicio: Compañía que ofrece servicios *cloud*.
- Auditor (opcional): Compañía que verifica que el servicio cumpla con las condiciones del SLA.

2. Establecer dos escenarios posibles de interacción entre los actores.

Como se evidencia en el capítulo 6, se obtuvieron dos categorías en lo que tiene que ver en los escenarios de interacción que se citan a continuación.

- a. Escenarios actuales:

- Búsqueda individual: en este escenario tanto la búsqueda como la adquisición de los recursos se hace por parte del usuario final sin ayuda de herramientas externas.
- Servicios de recomendación basados filtros sin ranking: son servicios de recomendación donde se puede filtrar por criterios predefinidos de utilidad para el usuario final, estos servicios facilitan muchísimo la búsqueda de proveedores.
- Servicios de recomendación con filtros basados en ranking: Facilitan al usuario final realizar una búsqueda de proveedores que cumplan con sus expectativas, haciendo uso de escalas multivaluadas de tres o cinco valores para destacar los proveedores mejor calificados por otros usuarios.

b. Escenarios emergentes:

- Oferta de servicios por subasta inglesa: Se trata de una subasta tradicional donde los interesados ofertan siempre subiendo el valor a pagar por el servicio.
- Oferta de servicios por subasta inversa: Como su nombre lo indica es una subasta donde se debe bajar el valor de la oferta cada vez, en este escenario los que ofertan son los proveedores.
- Escenarios de libre mercado: En el modelo de libre mercado el regulador del servicio es el mismo mercado, por tanto el precio del bien o servicio se establece por una dinámica de oferta y demanda.

3. Definir los mecanismos que permitan la interacción entre los actores.

En los escenarios actuales la interacción se da mayormente por medio de servicios Web o por servicios de intermediarios, sin embargo únicamente se limitan a recomendar. En la propuesta de este trabajo se propone que los actores expresen sus expectativas mediante servicios Web pero que la comunicación sea llevada a cabo por agentes como se ilustra en el capítulo 9. Para lograr dicho fin se proponen tres tipos de agentes.

- Agentes consumidores: Son agentes que representan a los consumidores, de servicios, su función es minimizar la inversión del comprador.

- Agentes proveedores: Son agentes que representan a los proveedores, su función es maximizar la ganancia del proveedor.
- Agentes intermediarios: El agente intermediario garantiza que la información de compradores y vendedores sea privada, compara las ofertas e informa si se debe mejorar la oferta por parte de proveedor y consumidor.
- Tablero: Como elemento adicional se incorporó un tablero cuya principal función es publicar las ofertas para que los compradores y vendedores tengan información inicial del historial de negociaciones.

4. Desarrollar plantilla de SLA. Proponer la definición de un documento que de soporte a los procesos de negociación y monitoreo en ambientes *cloud*.

Para el desarrollo de la plantilla de SLA se encontró que el desarrollo de una ontología, era muy ventajoso sobre otras formas hacerlo como usar un simple documento XML o WSLA. El desarrollo de una ontología para la definición del dominio es muy interesante por sus características de expansión y contracción, reuso y la definición formal del dominio.

Para el desarrollo de ontología se seleccionó como metodología *NEON*, la cual presenta ventajas para el desarrollo de ontologías como su propuesta de diseño por escenarios, plantillas para documentación y la facilidad que da para crear redes de ontologías lo que potencia el reuso de recursos ontológicos y no ontológicos.

Como resultado concreto de esta etapa se obtuvieron dos ontologías que representan la plantilla OWL para servicios IaaS, para servicios PaaS, ambas pueden ser catalogadas como redes de ontologías dado que expresan dominios propios de los servicios *cloud*, negociación, facturación entre otros. Las ontologías anteriormente nombradas se encuentran publicadas en el repositorio ontohub.org cuyas URL se encuentran a continuación.

- Plantilla para servicio IaaS: <http://ontohub.org/ontocloud/SLA-IaaS-Template.owl>
- Plantilla para servicio PaaS: <http://ontohub.org/ontocloud/SLA-SaaS-Template.owl>

5. Proponer un procedimiento para la negociación de SLA entre usuarios finales y proveedores SaaS.

Para conseguir el objetivo planteado anteriormente, se planteó un protocolo de negociación que contempla todos los escenarios planteados en el presente trabajo de grado.

El protocolo se encuentra en el desarrollo del trabajo con todas sus partes, cabe destacar que el protocolo hace parte de la producción original de este trabajo.

Resultados de divulgación.

Durante el desarrollo del trabajo se han logrado diferentes resultados de divulgación, los cuales se ilustran a continuación.

Ponencia oral.

Ponencia en el evento “1er seminario de tecnologías de la Información y Comunicación SETICUCC”. Para dicho evento se envió un artículo titulado “Orquestación de servicios en entornos *grid* y *cloud*”, el cual no fue publicado pero se aceptó la ponencia.

Ponencia aprobada para el evento “1er. Congreso Latinoamericano y Centroamericano de Semilleros, Líderes y Grupos de Investigación”. Que se realizará en abril del presente año en La Habana Cuba. Para dicho evento se envió un artículo titulado “Desarrollo de una ontología para definir el dominio de servicios *cloud* IaaS”

Poster.

Poster aprobado para el evento “1er. Congreso Latinoamericano y Centroamericano de Semilleros, Líderes y Grupos de Investigación”. Que se realizará en abril del presente año en La Habana Cuba. Para dicho evento se envió un artículo titulado “Desarrollo de una ontología para definir el dominio de servicios *cloud* IaaS”

Artículos escritos.

- Desarrollo de una ontología para definir el dominio de servicios *cloud* IaaS. (Artículo presentado a evento y aceptado en revista)
- Ontología para representar atributos del proceso de negociación *cloud* IaaS, mediante documentos SLA. (Artículo en correcciones)
- Protocolo de negociación para servicios *cloud*, basado en agentes. (Artículo en construcción).

Artículos en revista.

- Desarrollo de una ontología para definir el dominio de servicios *cloud* IaaS. (Artículo aceptado para publicación en la Revista de Ingeniería, Investigación y Tecnología Editada por la Universidad Nacional Autónoma del Estado de México.

Capítulos de libro.

Durante el desarrollo de investigación se escribieron dos capítulos de libro, aun cuando el tema del libro no está relacionado con la tesis realizada se incorporaron aspectos del trabajo en el capítulo:

- "Informática Médica bajo la perspectiva de la Web Semántica y la Computación Distribuida" Telesalud E Informática Medica. En: Colombia ISBN: 9789586515535 ed: Universidad Nacional Abierta Y A Distancia Unad, v. , p.244 - 259 ,2013

Otros productos.

En cuanto se publiquen los resultados de los artículos, las figuras originales, tablas y otra producción. Será publicada en la red científica *Research Gate*.

11. Conclusiones y trabajo futuro

En el siguiente apartado se presentan las conclusiones del trabajo y se plantean algunas propuestas de trabajo futuro.

11.1. Conclusiones.

En el presente trabajo se propuso un marco de referencia para la negociación de acuerdos de nivel de servicio SLA, en ambientes *cloud*. El marco de referencia propuesto es completamente nuevo y para el mismo se definieron los actores involucrados en el proceso y se establecieron los posibles escenarios de interacción de los mismos, se definieron los mecanismos de para la interacción entre los actores y se establecieron mecanismos para el desarrollo del proceso de negociación.

En el presente trabajo definió un conjunto de actores relevantes en el proceso de negociación de servicios *cloud*, que son los proveedores, consumidores, intermediarios y auditores. El conjunto de actores representado puede variar en diferentes trabajos pero establece una base para comenzar en cualquier proceso de negociación.

Aun cuando los actores principales de proceso son el proveedor y el consumidor, para efectos del proceso de negociación automática, se requieren de otros actores de mucha importancia para dinamizar la consecución de acuerdos y por ende cerrar la negociación de manera satisfactoria.

En el desarrollo de la presente investigación se establecieron los escenarios de interacción para los actores, se encontró que actualmente en la oferta de precio fijo existen cuatro categorías de interacción entre actores que se citan a continuación.

Búsqueda individual de proveedores por parte del usuario final: En la búsqueda manual el usuario final debe hacer la búsqueda, seleccionar el proveedor que mejor se ajuste a sus necesidades y comprar el servicio que requiera.

Servicios de recomendación basados en filtros: En este tipo de búsqueda el usuario final hace uso de servicios Web que filtran resultados de proveedores según los criterios suministrados por el usuario final, quedando de parte del usuario la selección del proveedor y la compra de los servicios requeridos.

Servicios de recomendación basados en ranking: Este tipo de servicios de recomendación extiende el uso de filtros, proveyendo al usuario final de una lista organizada con los proveedores mejor calificados por otros clientes. De esa manera el usuario tiene el poder de los filtros para acotar la búsqueda y como información adicional las calificaciones de otros usuarios que pueden llegar a ser útiles para tomar la decisión de un proveedor sobre otro.

Servicios de intermediarios brókeres: Los servicios de intermediarios son servicios donde un tercero actúa en nombre de un comprador de manera que según los requerimientos dados por el cliente selecciona un proveedor y hace la compra o solo recomienda el proveedor dependiendo de lo pactado con el cliente.

Las categorías presentadas anteriormente aportan en la discusión de infraestructuras *cloud*, pues no se encontró evidencia de una categorización para las formas de interacción implementadas actualmente y se constituyen en un hallazgo original de este trabajo.

Adicionalmente a las categorías de servicios implementados, se encontró que existen otras formas de interacción entre usuarios las cuales se denominaron alternativas en este trabajo. La razón de llamarlas alternativas obedece a que aun cuando son posibles de implementar no se encuentra que hayan tenido un impacto real en el mercado, por supuesto que lo anterior se debe al modelo de precio fijo.

Entre las formas de interacción alternativa se encuentran las siguientes:

Subasta inglesa: En ese tipo de subasta los consumidores aspiran a comprar un bien el cual se vende al mejor postor. Para que un comprador pueda hacerse del bien o servicio subastado debe mejorar la oferta del resto de sus competidores. Claramente representa un beneficio para para el vendedor dado que las ofertas van en un solo sentido y solo pueden mejorar.

Subasta inversa: El modelo de subasta inversa como su nombre lo indica se desarrolla al contrario que una subasta inglesa tradicional, en esta subasta el comprador define los requerimientos y una oferta tope, de allí en adelante los posibles proveedores deben ir bajando los precios. Este tipo de subasta beneficia al comprador, sin embargo se ve únicamente en grandes negocios sobre todo en el sector estatal.

Modelos de libre mercado: En los modelos de libre mercado los precios de los bienes o servicios se valorizan o devalaran atendiendo a dinámicas de oferta y demanda, en dicho sentido el precio de los bienes no es fijo y no existe un beneficio claro para alguna de las partes.

En el desarrollo del trabajo también se logró definir los mecanismos para la interacción entre los actores. En las implementaciones actuales las interacciones se dan por medio de servicios Web y el consumidor debe hacer mucho trabajo manual. Sin embargo en un ambiente de negociación automático la forma natural de interacción es por medio de agentes los cuales representan roles específicos dentro del protocolo.

Los agentes garantizan tiempos de respuesta muy buenos y gran flexibilidad, uno de los aportes más importantes del trabajo en ese sentido es definir el espacio de negociación para los agentes lo cual hace que se tenga claridad sobre las características que deben tener a la hora de ser implementados.

No se encontró evidencia de la definición del espacio de negociación de agentes, en un protocolo de negociación que se adapte a entornos *cloud*. De manera esta parte se constituye en un aporte muy importante para la discusión de sobre la negociación automática de servicios *cloud*.

En esta tesis se desarrollaron plantillas para documentos SLA, que soportan los procesos de negociación y monitoreo en ambientes *cloud*. Se utilizaron ontologías para definir los dominios de interés para el documento SLA dichos dominios comprenden los servicios SaaS, servicios IaaS, SLA y características concernientes con la negociación.

Para llegar al desarrollo de las ontologías primero se caracterizó una muestra de servicios IaaS y SaaS. En la caracterización de servicios IaaS se encontró, que tienen como entidades base recursos computacionales y recursos software que son presentados a los usuarios por medio de máquinas virtuales. Por otra parte existen diferentes métodos de acceso a los recursos y tipos de facturación.

En cuanto a los servicios SaaS, se puede añadir que existe una gran oferta dada esta condición los términos de uso y contratos difieren según el sector en el cual se encuentra el producto software, por tal cantidad de oferta se decidió tomar para este trabajo servicios de ofimática en línea.

Se analizaron dos servicios líderes de ofimática en línea y se encontró una oferta de servicios muy similar en ese aspecto no hubo sorpresas. En cuanto a los contratos de servicios también se observó un comportamiento muy similar, sin embargo es de extrañar que los contratos de servicio sean un salva vidas para los proveedores en todos los casos.

Además de los servicios se analizaron los contratos de servicios y SLA, tanto de servicios IaaS como SaaS seleccionados. Se encontró una tendencia muy marcada a definir las penalidades, forma de facturación y términos de uso. De otra parte lo que tiene que ver con el derecho de consumidor como la auditabilidad y el compromiso con el medio ambiente están absolutamente por fuera del alcance de los documentos encontrados.

Las caracterizaciones de los servicios y contratos fueron un insumo valioso para la investigación, y pueden llegar a ser de gran interés para trabajos futuros, por parte del presente trabajo sin la caracterización no se hubiera llegado al desarrollo de las ontologías.

En el trabajo se desarrollaron dos redes de ontologías para definir tres dominios, para desarrollar las redes de ontologías se decidió usar la metodología *NeOn*, que por su enfoque basado en escenarios es perfecta para la creación de redes ontológicas desde cero u a partir de recursos ontológicos y no ontológicos existentes.

Para el desarrollo de la ontología de servicios IaaS, se tomaron recursos ontológicos existentes y recursos no ontológicos, siguiendo la metodología se logró crear una red de ontologías nueva que define el dominio IaaS y cubre el dominio SLA, la parte del dominio SLA se obtuvo a partir de recursos no ontológicos que fueron fruto de la caracterización y la investigación teórica.

Para el desarrollo de la ontología de dominio SaaS, se tomó como base la ontología de servicios IaaS, suprimiendo por supuesto la parte correspondiente a dicho servicio y poniendo en su lugar las clases correspondientes al dominio SaaS. Siguiendo la metodología se pudo conseguir una red de ontologías para definir el dominio SaaS y cubrir el dominio SLA.

Para el trabajo realizado el desarrollo de las ontologías es uno de los aportes más importantes pues es el mecanismo que da sentido al protocolo, es la forma de expresar requerimientos, el mecanismo para expresar el contrato y son la base de conocimiento inicial de los agentes. En dicho sentido el desarrollo de las ontologías es el gran factor integrador que permite que el protocolo se pueda llevar a cabo.

Por otra parte aun cuando en el documento se citan recursos ontológicos que abarcan parte de servicios SaaS e IaaS, no se evidencia una red de ontologías que cubra también el dominio SLA y que tenga las funciones específicas que se le ha dado en este trabajo. Por dicha razón es un gran aporte en la generación de SLA semánticos, es una forma innovadora de integrar semántica a los procesos de negociación.

En esta tesis se planteó un protocolo de negociación que se adapta a la negociación de servicios *cloud*, para esto se analizaron diferentes protocolos de negociación basados en

agentes, se encontró que el protocolo de concesión bilateral multipasos, se acerca mucho a la aproximación de negociación que se presenta en este trabajo.

La concepción de la negociación en este trabajo se ciñe a que el proceso de negociación es un mecanismo para resolver conflictos de intereses donde todas las partes implicadas deben estar dispuestas a ceder para que puedan verse beneficiadas, es decir debe haber un gana – gana.

El protocolo de negociación que se planteó en el trabajo tiene elementos del protocolo de concesión bilateral, sin embargo el protocolo propuesto difiere de dicho protocolo en que se han establecido escenarios y reglas que se ajustan a mercados *cloud*, así como la implementación de mecanismos como el tablero que sirve para que los agentes puedan ajustar su información inicial.

Por otra parte el protocolo contempla el uso de ontologías para una gran parte de funciones, en diferencia al resto de protocolos que se centran en el proceso de negociación únicamente dejando por fuera los mecanismos previos que llevan a que el proceso se desarrolle de forma satisfactoria.

Aun cuando existen protocolos de negociación basados en agentes que sirvieron de inspiración para el protocolo que se presenta actualmente, no se evidencia un protocolo de negociación con el alcance del que se propuso. Lo que hace que sea un protocolo nuevo que puede servir para trabajos futuros.

11.2. Trabajo futuro

Para el trabajo futuro se recomienda realizar la implementación de los agentes que permitan probar el protocolo de negociación presentado en este trabajo. De manera que se puedan obtener datos que permitan implementar mejoras en el protocolo.

El protocolo se trató de implementar en herramientas de simulación sin embargo muchos de los motores de simulación que se encontraron se centraban en problemas espaciales, para los cuales el problema que se propuso en este trabajo no se adaptaba, por otro lado en uno de los simuladores *cloud* más populares CloudSim no está soportado el proceso de negociación de infraestructura y se centra en operación del servicio.

Las ontologías presentadas en el trabajo pueden servir como punto de partida para trabajos futuros donde se requiera definir dominios de negociación y dominios de servicios *cloud*. Se espera que en el futuro se aproveche este insumo en redes de ontologías.

12.Referencias.

- Alhamad, M., Dillon, T., & Chang, E. (2010). SLA-Based Trust Model for Cloud Computing. *4th IEEE International Conference on Digital Ecosystems and Technologies*, 606–610. <http://doi.org/10.1109/DEST.2010.5610586>
- Alper, P., Corcho, O., Kotsiopoulos, I., Missier, P., Bechhofer, S., Kuo, D., & Goble, C. (2006). S-OGSA as a Reference Architecture for OntoGrid and for the Semantic Grid. *Management*. Retrieved from <http://oa.upm.es/5677/>
- Armbrust, M., Fox, A., Griffith, R., Joseph, A. d., Katz, R., Konwinski, A., ... Zaharia, M. (2010). A View of Cloud Computing. *Communications of the ACM*, 53(4), 50 – 58.
- Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., ... Zaharia, M. (2009). Above the clouds: A Berkeley view of cloud computing. *University of California, Berkeley, Tech. Rep. UCB*, 07–013. <http://doi.org/10.1145/1721654.1721672>
- Berners-lee, T. I. M., Hendler, J., & Lassila, O. R. A. (2002). The Semantic Web. *Knowledge Creation Diffusion Utilization*, (May 2001).
- Bosak, J., & Bray, T. (1999). XML and the Second-Generation WEB. *Scientific American*, 280(5), 89–93. <http://doi.org/10.1038/scientificamerican0599-89>
- Bouchenak, S. (2010). Automated control for SLA-aware elastic clouds. In *Proceedings of the Fifth International Workshop on Feedback Control Implementation and Design in Computing Systems and Networks* (pp. 27–28). ACM. Retrieved from <http://dl.acm.org/citation.cfm?id=1791210>
- Buyya, R., Yeo, C. S., & Venugopal, S. (2008). Market-oriented cloud computing: Vision, hype, and reality for delivering IT services as computing utilities. In *Proceedings - 10th IEEE International Conference on High Performance Computing and Communications, HPCC 2008* (pp. 5–13). <http://doi.org/10.1109/HPCC.2008.172>
- Cardellini, V., Casalicchio, E., & Branco, K. (2011). Performance and Dependability in Service Computing: Concepts, Techniques and Research Directions. 19. Retrieved from <http://books.google.com/books?hl=en&lr=&id=HJdHZ2ekg8cC&oi=fnd&pg=PR1&q=Performance+and+Dependability+in+Service+Computing++Concepts+,+Techniqu>

es+and+Research+Directions&ots=-CM-

ak1LAe&sig=CHkhl1YCQi3VbI4L9R72FbMII0\http://books.google.com/books?hl=en

- Castro, H. (2006). Grid Computing: promesa de los sistemas distribuidos. *Revista Sistemas ACIS*, Edición, 98. Retrieved from <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Grid+computing:+promesa+de+los+sistemas+distribuidos#0>
- Chang, V., Bacigalupo, D., Wills, G., & De Roure, D. (2010). A Categorisation of Cloud Computing Business Models. *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, 509–512. <http://doi.org/10.1109/CCGRID.2010.132>
- Chervenak, A., Foster, I., Kesselman, C., Salisbury, C., & Tuecke, S. (2000). The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets. *Journal of Network and Computer Applications*, 23(3), 187–200. <http://doi.org/10.1006/jnca.2000.0110>
- Corcho, O., Alper, P., Kotsiopoulos, I., Missier, P., Bechhofer, S., & Goble, C. (2006). An overview of S-OGSA: A Reference Semantic Grid Architecture. *Web Semantics: Science, Services and Agents on the World Wide Web*, 4(2), 102–115. <http://doi.org/10.1016/j.websem.2006.03.001>
- Corcho, O., Fernández-López, M., & Gómez-Pérez, A. (2003). Methodologies, tools and languages for building ontologies. Where is their meeting point? *Data and Knowledge Engineering*, 46(1), 41–64. [http://doi.org/10.1016/S0169-023X\(02\)00195-7](http://doi.org/10.1016/S0169-023X(02)00195-7)
- Denemark, J., Jankowski, M., Meyer, N., Ruda, M., & Wolniewicz, P. (2005). User Management for Virtual Organizations r ´ Institute of Computer Science , Masaryk University Faculty of Informatics , Masaryk University CoreGRID Technical Report Number TR-0012. *System*.
- Dobson, G., & Sanchez-Macian, A. (2006). Towards unified QoS/SLA ontologies. ... *Computing Workshops*, 2006. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4027032
- Dumitrescu, C. L., & Foster, I. (2005). GRUBER : A Grid Resource Usage SLA Broker, 465–474.

- Fernandez, M., Gomez-Perez, A., & Juristo, N. (1997). *METHONTOLOGY: From Ontological Arts Towards Ontological Engineering. Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering*, 33–40.
- Folling, A., Grimme, C., Lepping, J., & Papaspyrou, A. (2010). Robust Load Delegation in Service Grid Environments. *IEEE Transactions on Parallel and Distributed Systems*, 21(9), 1304–1316. <http://doi.org/10.1109/TPDS.2010.16>
- Foster, I., Kesselman, C., & Tuecke, S. (2001). The Anatomy of the Grid. *Intl J. Supercomputer Applications*. Retrieved from <http://www.cse.nd.edu/courses/cse498k/www/slides/xxiang.pdf>
- Foster, I., Zhao, Y., Raicu, I., & Lu, S. (2008a). Cloud Computing and Grid Computing 360-Degree Compared. *2008 Grid Computing Environments Workshop*, 1–10. <http://doi.org/10.1109/GCE.2008.4738445>
- Foster, I., Zhao, Y., Raicu, I., & Lu, S. (2008b). Cloud computing and grid computing 360-degree compared. *Grid Computing Environments Workshop 2008 (GCE '08)*, 1–10. <http://doi.org/10.1109/GCE.2008.4738445>
- Goble, C., & De Roure, D. (2004). The semantic grid: Myth busting and bridge building. *Ecai 2004: 16th European Conference on Artificial Intelligence, Proceedings*, 110, 1129–1135. <http://doi.org/10.1109/JPROC.2004.842781>
- Gruber, T. (1995). Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*. <http://doi.org/citeulike-article-id:230211>
- Jimenez, C. L. (2007). Datagrid: Infraestructura de computación en malla para integración de aplicaciones y de información, 58–67. Retrieved from http://www.acis.org.co/fileadmin/Revista_99/Tres.pdf
- Jin, L., Jin, L., Machiraju, V., Machiraju, V., Sahai, A., Sahai, A., ... Agreement, S. L. (2002). Analysis on Service Level Agreement of Web Services. *Technology*. Retrieved from <http://www.csd.uoc.gr/~hy565/newpage/docs/pdfs/papers/HPL-2002-180.pdf>
- Joseph, J., Ernest, M., & Fellenstein, C. (2004). Evolution of grid computing architecture and grid adoption models. *IBM Systems Journal*, 43(4), 624–645. <http://doi.org/10.1147/sj.434.0624>
- Kijthaweesinpoon, T., & Zhou, P. L. (2008). An Automated Mechanism for Negotiations in

- E-Commerce Systems. *2008 International Seminar on Business and Information Management*, 194–198. <http://doi.org/10.1109/ISBIM.2008.236>
- Liu, F., Tong, J., Mao, J., & Bohn, R. (2013). NIST Cloud Computing Security Reference Architecture. *NIST Special ...*, 500(299), 1–204. <http://doi.org/500-299>
- Lomuscio, A. R., Wooldridge, M., & Jennings, N. R. (2003). A classification scheme for negotiation in electronic commerce. *Group Decision and Negotiation*, (12), 31–56. Retrieved from http://link.springer.com/chapter/10.1007/3-540-44682-6_2
- Macías, M., & Guitart, J. (2010). Using Resource-level Information into Nonadditive Negotiation Models for Cloud Market Environments. In *Proceedings of NOMS'10* (pp. 325–332). <http://doi.org/10.1109/NOMS.2010.5488485>
- MacVittie, L., Murphy, A., Silva, P., & Salchow, K. (2010). Controlling the Cloud: Requirements for Cloud Computing. *F5 Networks, Inc.*, 15. Retrieved from <http://www.f5networks.es/pdf/white-papers/controlling-the-cloud-wp.pdf>
- Maedche, A., & Staab, S. (n.d.). Ontology Learning for the Semantic Web Ontologies for the Semantic Web. *Workbench*, 1–18.
- Menasce, D. A., & Casalicchio, E. (2004). QoS in grid computing. *Internet Computing, IEEE*, 8(4), 85–87. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1315571
- Netto, M. A. S. (2011). CANPRO : A Conflict-Aware Protocol for Negotiation of Cloud Resources and Services.
- Noy, N., & McGuinness, D. (2001). *ONTOLOGY DEVELOPMENT 101: A guide to creating your first ontology*, 1–25. Retrieved from <http://er.uni-koblenz.de/IFI/AGStaab/Teaching/SS09/sw09/Ontology101.pdf>
- Padgett, J. (2005). Grid Service Level Agreements Combining Resource Reservation and Predictive Run-time Adaptation. 91. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.134.842&rep=rep1&type=pdf>
- Pan, L. (2011). Towards a framework for automated service negotiation in cloud computing. *Cloud Computing and Intelligence Systems (CCIS)*, 2–5. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6045091
- Patel, P., Ranabahu, A., & Sheth, A. (2009). Service Level Agreement in cloud computing.

- Cloud Workshops at OOPSLA*. <http://doi.org/10.1007/978-1-4614-1614-2>
- Plummer, D., & Kenney, F. (2009). Three Types of Cloud Brokerages Will Enhance Cloud Services. Retrieved January 1, 2013, from <https://www.gartner.com/doc/973412?ref=ddisp>
- Roure, D. De, & Hendler, J. A. (2004). E-science: The grid and the semantic web. *IEEE Intelligent Systems*, 19(1), 65–71. <http://doi.org/10.1109/MIS.2004.1265888>
- Shi, X., Xu, K., Liu, J., & Wang, Y. (2013). Continuous Double Auction Mechanism and Bidding Strategies in Cloud Computing Markets. *arXiv Preprint arXiv:1307.6066*, 1–16. Retrieved from <http://arxiv.org/abs/1307.6066>
- Shu, Z. S. Z., & Meina, S. M. S. (2010). An architecture design of life cycle based SLA management. *Advanced Communication Technology (ICACT), 2010 The 12th International Conference on*, 2, 1351–1355. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5440283
- Suárez de Figueroa Baonza, M. del C. (2010). *NeOn Methodology for Building Ontology Networks : Specification , Scheduling and Reuse*.
- Sure, Y. (2002). *ON-TO-KNOWLEDGE* Ontology based Knowledge Management Tools and their Application 1 Introduction 2 Tool Suite 3 Industrial case studies.
- Talia, D. (2002). The open grid services architecture: Where the grid meets the web. *IEEE Internet Computing*, 6(6), 67–71. <http://doi.org/10.1109/MIC.2002.1067739>
- Vaquero, L. M., Rodero-Merino, L., Caceres, J., & Lindner, M. (2008). A break in the clouds. *ACM SIGCOMM Computer Communication Review*, 39(1), 50. <http://doi.org/10.1145/1496091.1496100>
- Vaquero, L. M., Vaquero, L. M., Rodero-merino, L., Rodero-merino, L., Caceres, J., Caceres, J., ... Lindner, M. (2009). A Break in the Clouds: Towards a Cloud Definition. *Computer Communication Review*, 39(1), 50–55. <http://doi.org/10.1145/1496091.1496100>
- Vouk, M. (2004). Cloud Computing – Issues , Research and Implementations. *Journal of Computing and Information Technology*, 235–246. Retrieved from http://hrcak.srce.hr/cit_ojs/index.php/CIT/article/download/1674/1378
- Wang, X., Sun, J., Huang, M., Wu, C., & Wang, X. (2012). A Resource Auction Based Allocation Mechanism in the Cloud Computing Environment. In *Parallel and*

Distributed Processing Symposium Workshops PhD Forum (IPDPSW), 2012 IEEE 26th International (Vol. 84, pp. 2111–2115).
<http://doi.org/10.1109/IPDPSW.2012.260>

Wu, L., & Buyya, R. (2010). Service Level Agreement (SLA) in Utility Computing Systems. *arXiv:1010.2881*, 6, 27. <http://doi.org/10.4018/978-1-60960-794-4.ch001>

Wustenhoff, E. (2002). Service level agreement in the data center. *Sun BluePrints*, (April). Retrieved from <http://anselmo.homeunix.net/Sun-BluePrints/0402/sla.pdf>

Youseff, L., Butrico, M., & Da Silva, D. (2008). Toward a unified ontology of cloud computing. In *Grid Computing Environments Workshop, GCE 2008*. <http://doi.org/10.1109/GCE.2008.4738443>