CONTROL DE DESCONEXIÓN DE VIDEO STREAMING MÓVIL EN ARQUITECTURA ANDROID MEDIANTE TÉCNICAS DE REALIDAD AUMENTADA Y MOTORES DE REGLAS CON HERRAMIENTAS OPEN SOURCE

Stalin León Paredes[1], Diego Toapanta Chancusi[2], Diego Marcillo[3], Tatiana Gualotuña Alvarez[4]

1, 2, 3, 4 Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador, stalin_eduardo_leon@hotmail.com, diegotoapantag@gmail.com, dmmarcillo@espe.edu.ec, tmgualotunia@gmail.espe.edu.ec

RESUMEN

El problema para poder implementar visualización de contenidos multimedia en dispositivos móviles en tiempo real se basa en aspectos tales como: limitaciones de memoria, comportamiento irregular de los sistemas de comunicación inalámbricos que dispone el dispositivo debido a rupturas de los enlaces, deterioro de la señal o en su defecto la inexistencia de motores de reglas de negocio que se encarguen de el análisis y el mejoramiento del paso de información. Este articulo está enfocado a describir como tratar de combinar elementos de reglas de negocio en el tratamiento de una señal de videostreaming y el uso de realidad aumentada en un dispositivo cliente para poder interactuar con su entorno el cual se encuentra lleno de puntos de acceso a diferentes redes inalámbricas.

Palabras Clave: Streaming, multimedia, dispositivos móviles, realidad aumentada, reglas de negocio.

ABSTRACT

The problem to implement display multimedia content on mobile devices in real time is based on aspects such as memory limitations, the irregular behavior of wireless communication systems available to the device due to broken links, or signal deterioration failing the lack of business rules engines to be responsible for analyzing and improving the passage of information. This article focuses on describing how to combine elements of business rules in the treatment of a video streaming signal and the use of augmented reality on a client device to interact with their environment which is full of different access point wireless networks.

Keywords: Streaming media, mobile, augmented reality, business rules

1. INTRODUCCIÓN

Los teléfonos móviles ahora ya no sólo sirven para hablar con otra persona o enviar mensajes de textos, hoy son dispositivos poderosos capaces de enviar así como reproducir audio y video, tanto así que se puede decir que son los nuevos ordenadores pero móviles.

La tendencia de hoy del desarrollo de aplicaciones, lenguajes de programación, desarrollo de frameworks y sistemas operativos se están inclinando hacia los dispositivos móviles, a la vanguardia de estos avances se encuentran diversas plataformas como: Symbian OS, Windows Mobile de Micrsoft, iPhone OS de Mac y Android de Google. Siendo la mayoría de estas propietarias limitando a los desarrolladores a generar nuevas aplicaciones, es por esto que Android tiene una gran ventaja ya que es una plataforma de código abierto.

Todas estas tecnologías más los avances en las redes WIFI[1], HSDPA[2], han hecho posible desarrollar nuevas aplicaciones para los dispositivos móviles, en donde la multimedia juega un papel muy importante y depende del servicio de streaming.

Compañías dedicadas al uso de video streaming como Youtube apuestan a la tecnología móvil, pero se enfrentan a problemas de desconexión siendo estos muy molestos para el usuario final. Para estos inconvenientes existe una interesante solución basada en código JAVA[3], la plataforma JADE-LEAP[4], que posee agentes inteligentes a los cuales se les puede encargar cualquier tarea o comportamiento, como por ejemplo ser los protagonistas en el control de la conexión en el servicio streaming. Su ventaja es que no utilizan gran cantidad de memoria en el terminal móvil, además de enviar varios tipos de mensajes sirven para controlar estados de envío y recepción de información.

Como un paso previo a la combinación de reglas de negocio y realidad aumentada sobre la aplicación que permite el control de desconexión de videostreaming se hará un breve repaso sobre conceptos básicos para explicar el objetivo de integrar reglas de negocio mediante Drools y realidad aumentada utilizando framework de código abierto (LOOK!).

Por lo tanto se busca evaluar el uso de la tecnología de agentes basada en JADE que realiza el control de desconexión de videostreaming, pero incluyendo un motor de reglas de negocio como lo es JBoos Drools, como mecanismo de evaluación de diferentes escenarios sobre redes WiFi y la técnica de videostreaming en los dispositivos móviles Android; mediante el uso de motores de reglas de negocio se busca amplificar las diversas posibilidades, con la cuales poder evaluar el comportamiento del paso de una señal de videostreaming y definir las posibilidades que se pueden ejecutar bajo ciertas características de una señal de videostreaming entre un servidor que se encuentra transmitiendo una señal y un cliente que para nuestro caso es una teléfono móvil con tecnología Android que reproducirá la señal emitida por el servidor, esto se lo explicará en la sección 3 de este artículo.

También se busca incluir dentro de una aplicación cliente dedicada a la reproducción de videostreaming, el framework LOOK! que permita a un usuario mantener contacto con el entorno que le rodea y los distintos servidores que se encuentren emitiendo una señal de videostreaming mediante el uso de Realidad Aumentada de forma que un usuario tenga la libertad de escoger el canal de comunicación, como se lo explicara en la sección 3 de este artículo.

2. MÉTODOS

2.1. REGLAS DE NEGOCIO

Dentro del contexto de aplicaciones empresariales, existe el concepto de regla de negocio. Estas reglas de negocio son definidas propiamente por las directivas de la organización y pueden ser condiciones o parámetros de los diferentes servicios que ésta presta. Algunos ejemplos son:

- El precio de un minuto de telefonía celular, según el plan al que pertenezca el usuario
- Las condiciones para aceptar o rechazar una solicitud de crédito
- Los parámetros para realizar descuentos por compra de productos en combo
- Las condiciones para admitir a un estudiante en una Universidad, etc.

Los motores de reglas de negocio o BRMS[5] surgen como una alternativa de solución a la problemática de administrar el cambio de las reglas de negocio en una organización. En particular los BRMS ofrecen:

- Un repositorio común a las aplicaciones donde se guardan las reglas de negocio versionadas
- Herramientas que permiten definir estas reglas tanto a usuarios técnicos (desarrolladores) como a usuarios no técnicos (directivos, expertos de negocio)
- Independencia entre el lenguaje de programación de una aplicación y el lenguaje para expresar las reglas
- Facilidad para definir las reglas de negocio, por categorías, en un lenguaje de alto nivel propio del

[1] Wireless Fidelity
[2] High Speed Downlink Packet Access.
[3] Lenguaje de Programación Orientado a Objetos
[4] Java Agent Development Framework
[5] Business Rule Manager Systems

motor de reglas.

- Un mecanismo de despliegue de las reglas de negocio.

2.1.1. ESTRUCTURA DE BÁSICA DE UNA REGLA DE NEGOCIO

La estructura de una regla de negocio se compone de n nombre, atributos, predicado en lógica de primer orden y acciones a ejecutar tras la evaluación de la regla como muestra la figura 1.

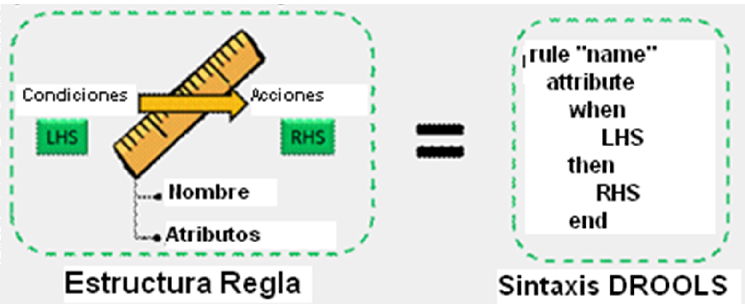

Fig.1 - Estructura regla – Sintaxis Drools

En la sintaxis MVEL[6] en Drools, la interpretación de los campos es la siguiente:

- Name: indica el nombre de la regla que se está creando y debe ser único por paquete en Drools.
- Attributes: se refiere a características propias de la regla, siendo algunos valores: no-loop, duration entre otros (En el ejemplo realizado utilizamos el atributo de no-loop).
- LHS[7], se refiere a las condiciones necesarias para que la regla aplique. Se expresa con lógica de primer orden y generalmente evalúan los atributos de los objetos a los que se les aplicaría la regla.
- RHS[8], se refiere a las acciones a ejecutar cuando el resultado de evaluar los elementos condicionales da verdadero; es decir cuando la regla aplica. En el RHS pueden ir además llamados a métodos java sobre nuestros objetos, en el desarrollo del tutorial llamaremos a métodos setters de nuestros objetos.
- Generalmente las reglas tienen la forma:

  "Clase"(Atributo1 comparador valor &&/|| Atributo2 comparador valor…)

  Un ejemplo de LHS es el siguiente:

  Cheese(type == "stilton" && price < 10 && age == "mature")

Aquí se evalúa sobre un objeto de tipo Cheese si su tipo es "Stilton" y su precio es menor a 10 y su edad es "mature". Si el objeto concreto sobre el que se evalúan estas condiciones posee estas características, se ejecutan las acciones.

2.2. DROOLS

Drools es una implementación de la especificación JSR94[9] que define un conjunto de funciones y procedimientos que debe ofrecer un motor de reglas en Java. La especificación no define un lenguaje para expresar las reglas y por ende, dadas las múltiples implementaciones de un motor de reglas, cada una de estas tiene su propio lenguaje. Organizaciones como la OMG[10] y W3C[11] trabajan actualmente en proyectos para intercambiar información de las reglas de negocio y estandarizar un lenguaje de mayor nivel de abstracción para la integración de organizaciones; pues el auge de la semántica web, reglas de negocio para verticales de negocio como comercio en línea y tendencias de evaluación y diagnóstico de reglas en diversas disciplinas ha llevado a la necesidad de compartir reglas de negocio. La W3C trabaja

[6] MVFLEX Expression Language
[7] Left Hand Side
[8] Right Hand Side
[9] Java Rule Engine API
[10] Object Management Group
[11] World Wide Web Consortium

en el proyecto RIF[12] el cual especifica casos de uso y requerimientos para la definición de un formato para intercambio y traducción de reglas de negocio entre diversos sistemas. La OMG trabaja en el proyecto RuleML que consiste en una serie de iniciativas para definir un lenguaje basado en XML[13] que permita el almacenamiento, intercambio, consulta y uso de aplicaciones web basándose en reglas.

Drools permite que se le enchufen distintos lenguajes para expresar las reglas, actualmente pueden ser escritas en Java, MVEL, Python y Groovy. También permite programación declarativa y es lo suficientemente flexible para evaluar la semántica de un problema con un lenguaje de dominio específico (DSL[14]) basado en XML.

2.2.1. FUNCIONAMIENTO DROOLS

El principal elemento es el **motor de inferencias** el cual evalúa reglas de negocio dados unos datos (hechos) para inferir conclusiones que se reflejan en las acciones a tomar. El proceso de evaluación de hechos vs reglas se denomina "PatternMatching". El siguiente diagrama representa el sistema de reglas como se muestra en la figura 2:

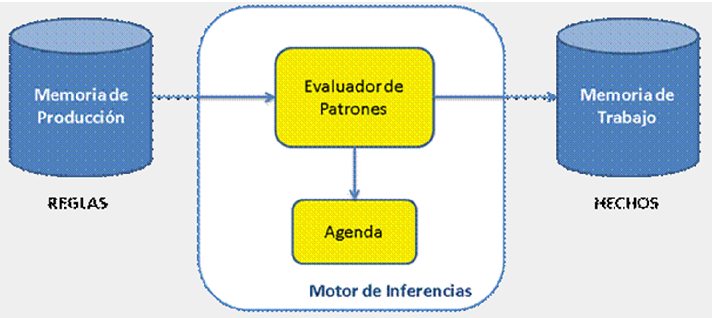

Fig.0- Funcionamiento Drools

- **Memoria de Producción:** Representa el espacio donde son almacenadas las reglas.
- **Memoria de Trabajo (WorkingMemory):** Representa el espacio donde la evaluación de reglas toma lugar.
- **Activación:** Objeto que se crea cuando los hechos satisfacen las condiciones de una regla.
- **Agenda:** Componente que administra la ejecución ordenada de activaciones creadas a partir de la misma aserción sobre mismos hechos mediante una estrategia de solución de conflictos.
- **Hechos:** Datos que se utilizan para evaluar las reglas. Pueden verse como clases java (beans) sobre los cuales se van a aplicar reglas. Deben ser insertados al worikingmemory antes de lanzar las reglas.

Luego de insertados los hechos al workingmemory, se llama al método fireAllRules() que se encarga de revisar las condiciones sobre los objetos y de ejecutar las acciones correspondientes. Por otro lado la creación del WorkingMemory, la inserción de los Facts y el llamado al método de fireAllRules si se hacen desde la aplicación java.

2.2.2. MÉTODOS DE EJECUCIÓN DE REGLAS DE NEGOCIO EN DROOLS

- StatefulKnowledgeSession: Método que se encarga de cargar toda la base de reglas configuradas.
- readKnowledgeBase():
  - Crea una instancia de un objeto de KnowledgeBuilder de clase de un archivo que contiene Sample.drl nuestras reglas de negocio.
  - Esta clase es responsable de análisis léxico y sintáctico los diferentes tipos de archivos de con-

[12]Rule Interchange Format
[13]Extensive Markup Language
[14]Domain Specific Language

figuración.

- fireAllRules(): Método que dispara las reglas configuradas en el archivo .drl.
- KnowledgeRuntimeLogger: Es un derivado de la especializada de la Agenda y WorkingMemorylisteners. Que presta servicios de auditoría de ejecución, cuya salida se pueden ver en una pantalla gráfica.

2.3. REALIDAD AUMENTADA

La realidad aumentada (RA)[15] es el término que se usa para definir una visión directa o indirecta de un entorno físico del mundo real, cuyos elementos se combinan con elementos virtuales para la creación de una realidad mixta en tiempo real. Consiste en un conjunto de dispositivos que añaden información virtual a la información física ya existente, es decir, añadir una parte sintética virtual a lo real. Esta es la principal diferencia con la realidad virtual, puesto que no sustituye la realidad física, sino que sobreimprime los datos informáticos al mundo real.

La realidad aumentada de investigación explora la aplicación de imágenes generadas por ordenador en tiempo real a secuencias de video como una forma de ampliar el mundo real. La investigación incluye el uso de pantallas colocadas en la cabeza, un display virtual colocado en la retina para mejorar la visualización, y la construcción de ambientes controlados a partir sensores y actuadores.

En términos generales la realidad aumentada permite: que la realidad aumentada:

- Combinar elementos reales y virtuales.
- Interacción en tiempo real.
- Se puede registrar eventos en 3D.

En la figura 3 se muestra una abstracción del funcionamiento de realidad aumentada.

Fig.3 -Representación de funcionamiento de realidad aumentada

3. IMPLEMENTACIÓN DE REGLAS DE NEGOCIO Y REALIDAD AUMENTADA EN LA ALPICACIÓN DE CONTROL DE DESCONEXIÓN DE VIDEOSTREAMING.

Una vez analizado el funcionamiento de las reglas de negocio y el framework de realidad aumentada, se procedió a implementar estas nuevas funcionalidades dentro del servidor y del cliente.

Empezamos integrando al servidor proxy las reglas de negocio necesarias para que al momento de formar la cadena de conexión entre VLC y la aplicación, la dirección IP dentro de esta cadena concuerde con la dirección del servidor de videostreaming. Este proceso mostró un resultado exitoso ya que la regla de negocio se ejecutó en el servidor al momento de establecer la conexión, pero comprometió el retardo en la entrega de los paquetes, debido a que las reglas tienen que ejecutarse antes de enviar los paquetes al cliente.

La realidad aumentada dentro de la aplicación cliente también tuvo una implementación básica en donde lo que se buscó fue integrar el framework LOOK! de realidad aumentada para que al momento de enfocar la cámara del dispositivo hacia el servidor de videostreaming, se establezca la conexión de forma automática, sin embargo, en esta ocasión una vez más se vio comprometida la eficacia de la aplicación, esta vez por la lentitud en el procesamiento propio del dispositivo.

3.1. MOTIVOS POR LOS CUALES IMPLEMENTAR REGLAS DE NEGOCIO Y REALIDAD AUMENTA A LA APLICACIÓN DE CONTROL DE DESCONEXIÓN DE VIDEOSTREAMING

[15] Realidad Aumentada

Dado que un usuario que se encuentra reproduciendo una señal de videostreaming no posee la percepción del entorno que lo rodea se vio la necesidad de incorporar ciertas funcionalidades al aplicativo, estas funcionalidades se establecieron en base a lo que se describe en la tabla 1.

Tabla I: Comparación Necesidad Actual con Procesos Propuestos

NECESIDAD	PROCESO
Incrementar la capacidad de control sobre una señal de videostreaming	Implementación de Motores de Reglas de Negocio
Incrementar la percepción del usuario cliente con los servidores que se encuentren emitiendo una señal de videostreaming.	Incorporación de Realidad Aumentada
Análisis de factibilidad entre la versión anterior de control de desconexión de videostreaming y la versión que incorpora JBDrools y Realidad Aumentada con un Framework.	Comparación estadística entre versiones
Propuesta de Investigación futuras	Versión para control de desconexión mediante el uso de Agentes JADE en vía de obsolescencia.

3.2. ESTABLECIMIENTO DE FUNCIONAMIENTO INICIAL

Tanto el APS y el APC funcionaban correctamente durante una sesión de videostreaming tanto en el envío y la recepción de paquetes de audio y video, esto mostró un correcto desempeño, al integrar reglas de negocio el APS empezó a mostrar cierta lentitud debido a la gran cantidad de archivos .log genera como parte del proceso de aprendizaje del aplicativo, sin embargo cuando debía dispararse la regla de negocio lo hizo con normalidad y cumpliendo las expectativas sobre la fusión en el APS que funcionaba con agentes JADE y reglas de negocio con JBoss Drools.

3.3. CONSIDERACIONES GENERALES PARA EL USO DE REGLAS DE NEGOCIO

La transmisión de video utiliza el mecanismo Streaming, el mismo permite la descarga parcial del video en pequeños fragmentos de menor tamaño en el Cliente, mejorando la usabilidad al enviar una secuencia de cuadros de video que presentan bajas demoras al iniciar la visualización y menores requerimientos de almacenamiento.

Durante el proceso de transmisión del video, podrían existir rupturas o cortes especialmente en comunicación con redes inalámbricas; sin embargo, si se privilegia la usabilidad del sistema, es decir, que la descarga sea paulatina a medida en que divide en streams el video a transmitir, sería deseable que al sufrir una desconexión el propio aplicativo identifique el restablecimiento de la comunicación y continúe el proceso de descarga sin que el usuario vuelva a reiniciar la sesión.

Es en este escenario donde los sistemas basados en Agentes de Software como JADE, figuran como alternativa para controlar y mejorar el proceso de transmisión de datos, incrementando de esta forma, la usabilidad del sistema.

Al poseer ya un sistema que permite controlar la desconexión se pudo implementar reglas de negocio a la aplicación servidor para evaluar la cadena de conexión del servidor mediante DROOLS, la regla establecida para evidenciar la ejecución de una regla, fue establecer que al tener un nombre diferente en la cadena de conexión al nombre propio del servidor, la regla se ejecuta y cambia el nombre del servidor en la cadena de conexión sin interferir con la transmisión, recepción de video y sonido.

3.3.1. Consideraciones para programar con realidad aumentada

Antes de programar, con la funcionalidad de la aplicación de realidad aumentada en la aplicación de control de desconexión, se debe plantear lo siguiente:

1. Los tipos de entidades van a ser representadas en la realidad aumentada tales como son EntityData, MyWordEntity.
2. Las características que se escogieron para definir estas entidades dentro del APC fueron que se podía interactuar mediante la cámara del móvil y el entorno que nos rodea.
3. La representación gráfica se la hizo mediante el uso de objetos en tres dimensiones.
4. Las interacciones que se vieron útiles en para cada entidad fueron:
    - Efectos al pulsar
    - Efectos al navegar
    - Efectos al enfocar con la cámara
5. Si es necesario localizar al usuario para obtener la funcionalidad buscada.
6. Y de ser así, que tipo de localización será la adecuada: Relativa, con el sistema inercial o absoluta, con el sistema de localización WiFi.
7. Qué sistema de referencia se utiliza para situar a los elementos y al usuario.
8. Donde está el origen de coordenadas del mundo.
9. Si se necesitan añadir elementos extras de interfaz (botones, menús, cajas de texto, Activitys secundarias, etc.) para completar la funcionalidad requerida.

Una vez analizado lo que se desea implementar es necesario agregar el framework de realidad aumentada LOOK! el cual nos permitirá utilizar todas las facilidades para que la aplicación establezca una conexión entre el entorno y la aplicación de control de videostreaming y de esa forma dar la potestad al usuario para elegir cuál será su canal de comunicación.

Una de las ventajas de incorporar realidad a la aplicación cliente es que el usuario tiene una mayor perspectiva del mundo que lo rodea.

3.3.2. Consideraciones para crear reglas de negocio

Para poder crear reglas de negocio para la aplicación que controla la desconexión de videostreaming se debe migrar el aplicativo a un nuevo tipo de proyecto DROOLS, el cual debe tener todas las librerías que nos ayudaran a crear reglas de negocio asociadas a los métodos ya creados en el aplicativo que controla la desconexión de video streaming mediante el uso de agentes JADE, estas reglas de negoció determinaran un cierto comportamiento en la aplicación previamente ya creada.

3.3.3. Análisis de Procesos Existentes

Una vez que se ha realizado la primera prueba, cuyos resultados son detallados en el numeral 4, se procedió a adaptar y medir el comportamiento que existe al aumentar reglas de negocio al funcionamiento anterior.

El servidor RTSP que se ha usado para las pruebas es VLC[16], el mismo que tiene como objetivo emitir el video codificado en un formato compatible con el dispositivo móvil Android. Se ha decidido utilizar la siguiente configuración: para video el códec h263, tasa de bits de 400 kb/s, 25 fps, ancho=176, alto=144 y el para el audio el códec AAC[17], tasa de bits de 128 kb/s, 1 canal y la tasa de muestreo de 44100.

Al crear reglas de negocio y colocarlas en el servidor que crea los agentes en el APS, las cuales cooperan para determinar cómo debe comportarse el APC mediante las reglas establecidas y continuar con el funcionamiento del control de desconexión de videostreaming.

Uno de los beneficios de DROOLS es que evalúa las reglas de negocio y almacena el comportamiento que existió al ejecutarlas para posteriores eventualidades permitiendo establecer un comportamiento más rápido al evaluar otros escenarios.

[16] VideoLAN Client
[17] Advanced Audio Coding

4. RESULTADOS

4.1. EVALUACION DE RESULTADOS EXPERIMENTALES

Los resultados obtenidos luego de implementar reglas de negocio y realidad aumentada con las siguientes consideraciones:

Se ha utilizado una red inalámbrica con tecnología WiFi con un punto de acceso de 54Mbps, un equipo portátil, procesador AMD de 2.20GHz, 4 GB de RAM y tarjeta inalámbrica Atheros AR5009 802.11 a/g/n WifiAdapter; en el cual se ejecuta el servidor de imágenes y el Agente Proxy Servidor APS; también se ha utilizado un dispositivo móvil Android, modelo NexusOne, Procesador Qualcomm QSD 8250 1 GHz, 512MB RAM, Tarjeta microSD de 4GB expandible hasta 32 GB, WiFi (802.11b/g/n); en el cual se ejecuta el Agente Proxy Cliente APC.

Para realizar la primera prueba se emitió una señal de video para determinar el funcionamiento que la aplicación ya agregada las características de reglas de negocio y realidad aumenta, reproduciendo un video del servidor mientras se ejecutan las reglas de negocio.

En esta prueba la dirección del servidor (URL) debe estar configurada y se debe estar emitiendo el video con VLC.

Al realizar esta prueba se puedo determinar que el video se visualiza con baja calidad debido a que al momento de ejecutar las reglas de negocio se generan archivos de log propios de DROOLS que servirán para el aprendizaje pero que comprometen en velocidad de respuesta al servidor.

Los resultados fueron no satisfactorios, se puede apreciar que la trasmisión del video se vuelve muy lenta debido a gran cantidad de logs que se generan en la APS al ejecutar reglas de negocio.

La segunda prueba que se realizo era para determinar si se puede conjugar en la APC realidad aumentada la cual su respuesta fue satisfactoria debido a que al adicionar el framework de realidad aumenta la aplicación sigue funcionando pero sin obtener el posicionamiento automático de la APS que emite una señal de video streaming como se muestra en la figura 4.

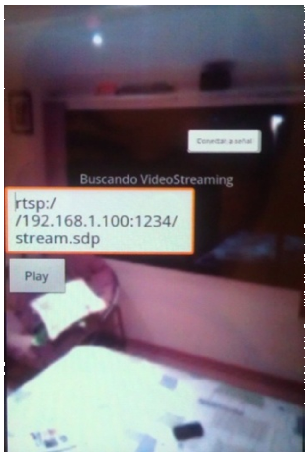

Fig. 4 - APC con realidad aumentada

Se realizaron otras pruebas, donde intenta obtener el posicionamiento de una APS que se encuentra emitiendo una señal de video streaming pero al tratar de conjugar la obtención del posicionamiento a través de la APC se determinó que la incompatibilidad del framework LOOK AR! con la versión ANDROIND con la que fue creada la APC para controlar la desconexión no permite obtener un posicionamiento mediante el uso de realidad aumentada.

Al intentar combinar el entrenamiento propio del framework CNWiFi, se evidencia la incompatibilidad de la versión utilizada para el desarrollo del framework contra la versión de la aplicación del cliente.

Puesto que no se pudo concluir con la investigación al tratar de conjugar reglas de negocio en el APS y realidad aumentada en APC se procedió a determinar únicamente las ventajas o desventajas mediante el uso de reglas de negocio por lo cual se ha tomado dos escenarios, en los que se han realizado las pruebas:

- **Escenario 1**: Versión Aplicación de control de desconexión

- **Escenario 2**: Versión Aplicación de control de desconexión con reglas de negocio.

En ambos escenarios se ha emitido un clip de video (.avi), con una duración de 03:19 minutos, así como también se ha calculado una muestra de los paquetes de audio y otra de video, para cada software.

4.2. CÁLCULO DE LA MUESTRA

Escenario 1 –Streaming-Jade Versión 1.1

Tabla II: ***Escenario 1 –Streaming-Jade Versión 1.1***

	N (# total de Paquetes Recibidos)	n(muestra)
Audio	2446	481
Video	3457	511

Escenario 2 –Jade-Streaming con Drools y Realidad Aumentada

Tabla IIII: ***Escenario 2 –Streaming-Jade Versión 1.1***

	N (# total de Paquetes Recibidos)	n(muestra)
Audio	2446	481
Video	3457	511

Con las muestras obtenidas, se ha calculado los parámetros que sirven para mostrar el rendimiento de los servicios de videostreaming, además permiten hacer una comparación entre los dos escenarios, demostrando fortalezas y debilidades de cada uno.

4.3. ANÁLISIS DE RESULTADOS

Al medir obtener la información durando una emisión de videostreaming se pudo obtener las siguientes graficas comparativas para el retardo de video y el retardo de audio:

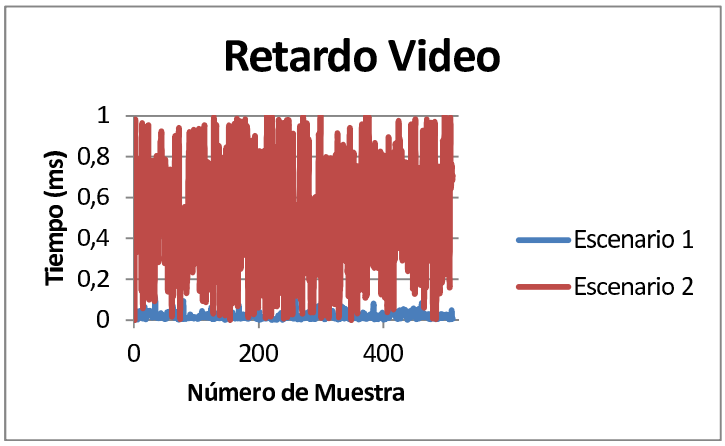

Fig. 5 - Retardo de video

En el gráfico de Retardo de video se muestra que al utilizar reglas de negocio en la APS provoca que el paso de los paquetes se procesen de una manera más lenta que sin utilizar reglas de negocio, se evidencia que al procesar las reglas de negocio la APS crea archivos .log lo cual produce que el paso de paquetes de video afecten su normal paso hacia la APC para su reproducción.

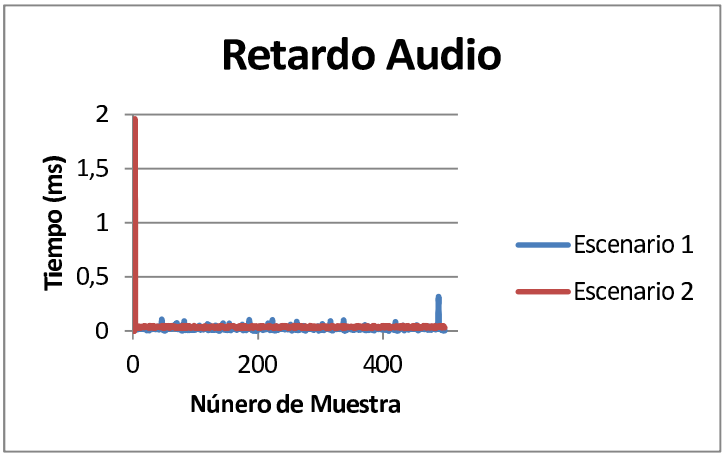

Fig. 6 - Retardo de Audio

En el gráfico de Retardo de audio se muestra que al utilizar reglas de negocio en la APS provoca que el paso de los paquetes se procesen de una manera poco más lenta que sin utilizar reglas de negocio, se evidencia que al procesar las reglas de negocio la APS crea archivos .log lo cual produce que el paso de paquetes de audio afecten su normal reproducción en la APC.

5. CONCLUSIONES Y TRABAJO FUTURO

A pesar del buen funcionamiento de DROOLS como componente de toma de decisiones dentro del servidor proxy, éste sistema cuenta con algunos bugs que aún no han sido arreglados y que se debe tomar en cuenta al momento de generar las reglas de negocio, como es el hecho de que el nombramiento sus clases pueden presentar inconvenientes si se usa una mezcla entre la escritura mayúsculas y minúsculas.

Sin embargo, más allá de estos inconvenientes, DROOLS nos proporciona la funcionalidad suficiente como para generar dentro del proxy un sistema de reglas de negocio capaz de tomar decisiones reconociendo los patrones aprendidos al momento de ejecutar cada una de las reglas.

Debido a la constante actualización de la versión del sistema operativo Android para móviles, y el distanciamiento de flash en estos dispositivos, JADE deja de ser una opción para la comunicación mediante agentes entre el servidor y el cliente, ya que el paso de mensajes con cambios en las tramas no está soportado a partir de la versión 2.3.4.

A esto debemos sumar que el sistema de localización proporcionado por Look! tampoco se puede acoplar a la versión de Android en la que se desarrolló inicialmente la aplicación del cliente. Esto nos lleva a buscar nuevas alternativas de comunicación que sean soportadas por los dispositivos de última generación, dando como pauta el uso de HTML5.

La principal recomendación es migrar la programación a codificación HTML5 aprovechando el uso de sus componentes de video embebidos en el código.

Evitar la manipulación de paquetes de videostreaming ya que las nuevas actualizaciones de Android validan estos paquetes antes de transmitirlos, por lo que modificarlos generaría errores en la aplicación.

Con los lineamientos de HTML5 se puede empezar a validar el código de la aplicación con código HTML5 para que la migración de la aplicación pueda ser exitosa.

6. AGRADECIMIENTOS

A la Ing. Diego Marcillo por su apoyo y consejos brindados a lo largo de este proyecto.

A todas las personas que nos supieron brindar sus experiencias ya que sin su colaboración no hubiera sido posible la realización de este proyecto.

7. REFERENCIAS BIBLIOGRÁFICAS

1.**ESPE - DarioLarenas y Cristina Navarro.**CONTROL DE DESCONEXIÓN EN REDES WIFI PARA DISPOSITIVOS MÓVILES BASADOS EN SISTEMA OPERATIVO ANDROID PARA SERVICIOS DE VIDEOSTREAMING.[Citado el: 5 de 10 de 2011.] *Sangolquí*, Pichincha, Ecuador , 2010.

2. **Universidad de los Andes.**CUPI2 PROJECT(2007) Tutoriales - DROOLS [En línea]. Colombia, 2007. [Citado el: 20 de 03 de 2011.]http://cupi2.uniandes.edu.co/site/index.php/tutoriales/drools

3. Woodrow Barfield, y Thomas Caudell, eds. **Fundamentos de Informática usable y Realidad Aumentada. Mahwah, NJ: Lawrence Erlbaum, 2001**

4.**Universidad Complutense de Madrid - Sergio Bellón Alcarazo, Jorge Creixell Rojo y Ángel Serrano Laguna.** Framework LOOK! de Realidad Aumentada para Android. [En línea] 01 de 07de 2011. [Citado el: 19 de 08 de 2011.] http://www.lookar.net/

5. **Android Developers.**Android Developers. [En línea] http://developer.android.com /.

6. **Jorge Franganillo,** HTML5: el nuevo estándar básico del web

7. **Stephen Cawood y Mark Fiala.** "Realidad Aumentada: A Practical Guide", 2008

8. **Héctor Fuente Pérez**, Guía Rápida de JBoss Drools BRMS 4.0.

9. **Diego Mattei Blog,** Cómo incluir videos en HTML5, http://diegomattei.com.ar/2010/06/29/como-incluir-videos-con-html5/

10. **Salvador Gómez,** Desarrollo en Android, http://sgolver.net/blog/?p=1313