

# AppTFG

**Jaime Ramos Romero  
Javier Bastarrica Lacalle**

**Grado en Ingeniería Informática  
Facultad de Informática**

UNIVERSIDAD COMPLUTENSE DE MADRID



**Trabajo de Fin de Grado  
Madrid, Junio 2015**

Directores:  
Samer Hassan Collado  
Pablo Ojanguren



# Autorización de Difusión y Uso

Autorizo a la Universidad Complutense de Madrid a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor, tanto la propia memoria, como el código, la documentación y/o el software desarrollado.

Jaime Ramos Romero  
Javier Bastarrica Lacalle

Madrid, Junio 2015

Copyleft by Jaime Ramos Romero and Javier Bastarrica Lacalle, released under the license Creative Commons Attribution Share-Alike International 4.0 available at:

<https://creativecommons.org/licenses/by-sa/4.0/>



# Índice general

	Page
Autorización de Difusión y Uso	III
Abstract	IX
Resumen	XI
<b>1. Introducción</b>	<b>1</b>
1.1. Objetivos del Proyecto . . . . .	1
1.2. Estructura del Documento . . . . .	1
<b>2. Migración de Wave a Android</b>	<b>3</b>
2.1. Estado del Arte . . . . .	3
2.2. Wave . . . . .	3
2.2.1. Google Wave . . . . .	3
2.2.2. Apache Wave . . . . .	3
2.3. Tecnologías y Características de Wave . . . . .	4
2.3.1. Características de Wave . . . . .	4
Federación . . . . .	4
Consistencia en tiempo real . . . . .	4
Escalabilidad . . . . .	4
Modelo Wave . . . . .	5
2.3.2. Servidores Wave . . . . .	6
Wave in a Box . . . . .	6
SwellRT . . . . .	7
2.4. Metodología de Migración . . . . .	8
2.4.1. Objetivo . . . . .	8
2.4.2. Entorno de desarrollo, Construcción y Depuración . . .	8
Eclipse . . . . .	8
Construcción por consola . . . . .	8
Depuración por consola . . . . .	8
2.4.3. Identificación y Solución de Problemas . . . . .	8
Conexion HTTP . . . . .	8
Conexion WebSocket . . . . .	8
Login . . . . .	8

2.4.4.	Organización y Resultados . . . . .	8
	Servicio Android . . . . .	8
	Diagramas y Dependencias . . . . .	8
<b>3.</b>	<b>Creación de aplicación Android (NOMBRE DE APP)</b>	<b>9</b>
3.1.	1ª Parte: Programas Políticos . . . . .	9
3.1.1.	Estado del Arte . . . . .	9
3.1.2.	Intencion . . . . .	9
3.1.3.	Objetivos . . . . .	9
3.1.4.	Usabilidad . . . . .	9
3.2.	2ª Parte: Propuestas y Comparativas . . . . .	9
3.2.1.	Estado del Arte . . . . .	9
3.2.2.	Intencion . . . . .	9
3.2.3.	Objetivos . . . . .	9
3.2.4.	Usabilidad . . . . .	9
3.3.	Tecnologías y Metodologías de la app . . . . .	9
3.3.1.	Base de Datos . . . . .	9
3.3.2.	Service REST . . . . .	9
3.3.3.	Frontend . . . . .	9
<b>4.</b>	<b>Resultados y Conclusiones</b>	<b>11</b>
4.1.	Discusion de Resultados . . . . .	11
4.2.	Conclusiones . . . . .	11
<b>5.</b>	<b>Trabajo a Futuro</b>	<b>13</b>
5.1.	Mejoras . . . . .	13
	<b>Bibliografía</b>	<b>13</b>

## Índice de figuras

2.1. Modelo Conversacional de Wave . . . . .	6
2.2. Cliente Wave In A Box . . . . .	7

## Índice de cuadros

# Abstract

Abstract en inglés.

**Keywords:**

Apache Wave, Collaboration, Android, Apps, Real Time, Politics





# Resumen

Abstract en español.

## **Palabras Clave:**

Apache Wave, Colaboración, Android, Apps, Tiempo Real, Política



# Capítulo 1

## Introducción

### 1.1. Objetivos del Proyecto

Los objetivos son los siguientes:

- 1ª parte: Migración de Wave a Android
  - Estudiar la implementación actual de Wave en Java y GWT.
  - Adaptar la implementación para hacer uso de las características nativas de Android.
- 2ª parte: Creación de una aplicación Android
  - Evaluar posibles ideas de aplicación y estudiar su viabilidad.
  - Evaluar con usuarios su usabilidad.
  - Implementar la aplicación.
  - Testear y evaluar con usuarios el resultado.

### 1.2. Estructura del Documento

- Capítulo 2 - Migración de Wave a Android: Descripción de la tecnología Wave y de la Metodología utilizada para la migración a Android.
- Capítulo 3 - AppTFG:
- Capítulo 4 - Resultados y Conclusiones:
- Capítulo 5 - Trabajo Futuro:



# Capítulo 2

## Migración de Wave a Android

### 2.1. Estado del Arte

### 2.2. Wave

#### 2.2.1. Google Wave

Ideado y presentado en 2009 por ingenieros de Google [1], Wave es a la vez un protocolo de comunicaciones [2] y una plataforma web de código libre, que permiten a sus usuarios comunicarse y colaborar entre sí en tiempo real (Ver sección 2.3.1) y de forma federada (Ver sección 2.3.1) a través de Internet. Inicialmente fue desarrollado con el objetivo de integrar en una sola plataforma servicios ampliamente utilizados como son el correo electrónico, las redes sociales y la mensajería instantánea. Pese al gran entusiasmo generado entre la comunidad de desarrolladores tras su anuncio, en el año 2010 Google anuncia el abandono del proyecto [3] debido a su poca acogida entre los desarrolladores y a que decide reorientar el uso de la tecnología hacia sus plataformas de edición de documentos Google Docs [4] y a su red social Google + [5]. Es en este momento cuando el desarrollo libre del proyecto pasa a manos de la Apache Software Foundation bajo el nombre de Apache Wave.

#### 2.2.2. Apache Wave

Al cambiar de manos su desarrollo en 2010, la tecnología pasa a formar parte de la incubadora de la fundación Apache [6] como software de código libre bajo licencia Apache [7]. Así, se produce el desarrollo de Wave In a Box (WIAB) (Ver sección 2.3.2), plataforma que integra un cliente web sencillo y una implementación de un servidor Wave que cualquiera puede descargar y desplegar en su ordenador.

## **2.3. Tecnologías y Características de Wave**

### **2.3.1. Características de Wave**

Como plataforma de código libre desarrollada para ser utilizada en red, Wave hace uso de distintas tecnologías y protocolos bien conocidos. Entre sus características más destacadas están las siguientes:

#### **Federación**

El Protocolo Wave [2] fue desarrollado para utilizar un modelo federado [8] [9] de comunicación basado en la tecnología XMPP [10] [2]. Se trata por tanto de un modelo descentralizado en el que cualquiera de los participantes en la conversación es libre de actuar tanto como servidor como cliente sin que ello afecte a su participación en la conversación. Además, a diferencia de otras tecnologías (como el correo electrónico) en las que cada participante almacena su propia copia de la conversación y cada vez que hay cambios se debe transmitir la conversación entera a todos los participantes, Wave tiene la ventaja de que actúa de forma que es el servidor de la conversación el único que almacena la copia entera y se encarga de calcular los cambios que se han producido para transmitir solamente dichos cambios por la red a los participantes, con las consiguientes ventajas en términos de latencia que ello conlleva.

#### **Consistencia en tiempo real**

El Protocolo Wave [2] utiliza la tecnología de Transformaciones Operacionales (OT) [11] para garantizar la consistencia en la comunicación en tiempo real entre los participantes. Es decir, cualquier cambio producido por cualquiera de los participantes en la conversación se transmite automáticamente y en tiempo real al resto de los participantes sin pérdida de información y garantizando que los cambios se muestran en el estricto orden en el que se produjeron sin errores [12].

#### **Escalabilidad**

Wave fue desarrollado como un protocolo de alta escalabilidad que permite gestionar la existencia de una gran cantidad de conversaciones y participantes sin que por ello se resienta la productividad del sistema.

### Modelo Wave

Además de definir el protocolo del que hace uso Wave, Google definió un Modelo de Datos Conversacional [13] que refleja la arquitectura de los datos que componen las conversaciones en Wave. Así, a grandes rasgos, podemos ver dichas conversaciones como documentos XML sobre los que los usuarios participantes (cualquiera es libre de unirse a una conversación en cualquier momento) actúan creando nuevos elementos o modificando los ya existentes. Este modelo de datos define una nomenclatura propia para los elementos que componen esta tecnología [14] [9]:

- **Wave:** Conjunto de wavelets (conversaciones).
- **Wavelet:** conjunto de documentos de una conversación y sus participantes.
- **Blip:** documento con el contenido de un mensaje en la conversación. Un blip puede tener otros blips dentro de él y los blips pueden ser publicados o no en función de si su visibilidad se extiende o no al resto de participantes de la conversación respectivamente.
- **Manifiesto conversacional:** documento con metadatos que definen la estructura de una conversación.
- **Hilo conversacional:** conjunto de Blips consecutivos que forman parte de una conversación.
- **Extensiones** [15]: pequeñas aplicaciones que se ejecutan dentro de una Wave y aportan nuevas funcionalidades que no forman parte del modelo conversacional básico. Pueden ser de dos tipos:
  - **Gadget:** aplicación que se ejecuta en el contexto de una Wave y en la que todos sus usuarios participan.
  - **Robot:** aplicación que participa en una Wave a modo de usuario automatizado e interactúa con el contenido pudiendo modificarlo y responder a eventos por acciones de otros usuarios reales.



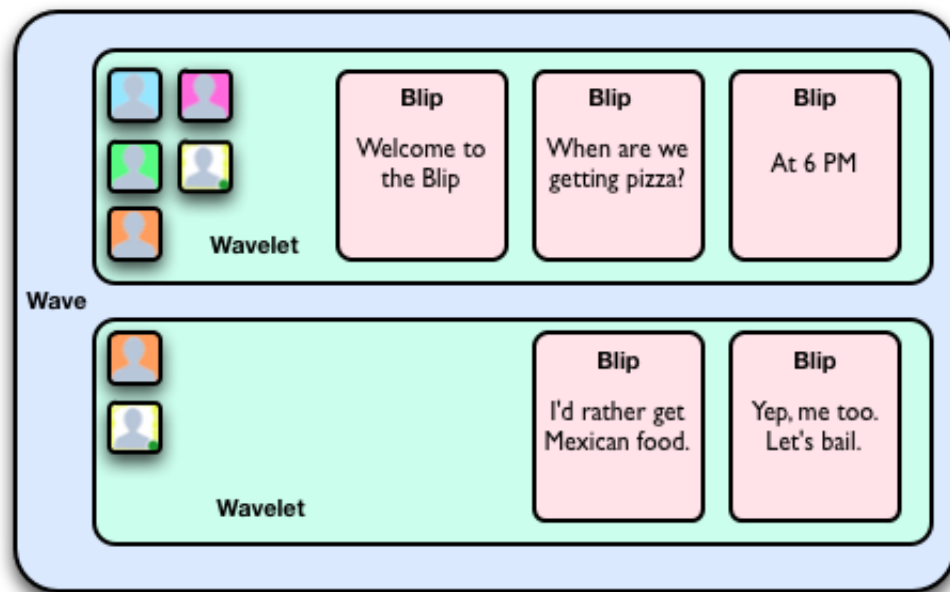


Figura 2.1: Modelo Conversacional de Wave

### 2.3.2. Servidores Wave

#### Wave in a Box

Wave In a Box (WIAB) [16] es el nombre de la implementación de un servidor Wave desarrollado por la Apache Software Foundation tras pasar el proyecto a sus manos en el año 2012. Al igual que el resto del código de la tecnología que heredó de Google, está implementado en Java usando OpenJDK [17]. La instalación trae consigo un cliente web desarrollado en Javascript usando el framework Google Web Toolkit [18]. Este cliente web sirve como prueba de concepto de las funcionalidades básicas del Modelo Conversacional de Wave, pudiendo gestionar waves, usuarios y extensiones. Actualmente cualquiera puede descargar y desplegar WIAB en su ordenador siguiendo los pasos que nos proporcionan en su wiki [19]. La aplicación se distribuye en forma de código fuente, accesible entre otras formas desde su repositorio de GitHub [20]. Existen asimismo servidores de prueba ya desplegados en Internet sobre los que se puede observar el funcionamiento de WIAB [21].

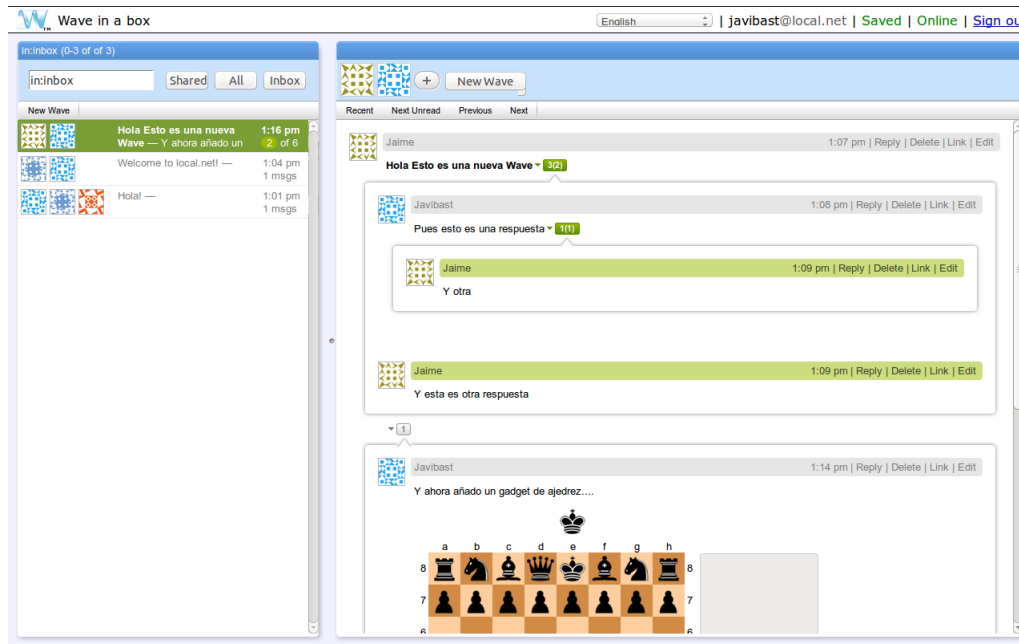


Figura 2.2: Cliente Wave In A Box

### SwellRT

Como parte del proyecto europeo P2PValue [22] existe SwellRT, un fork de WIAB que amplía las características de éste último añadiendo un nuevo modelo de datos (Modelo de Datos Colaborativo) más allá del Modelo de Datos Conversacional de Wave original. Proporciona también un API escrito en Java que permite trabajar sobre los datos de ese nuevo modelo en forma de tres tipos básicos: mapas, listas y strings. Es por tanto un framework de colaboración en tiempo real que basa su funcionamiento en Apache Wave y cuyo principal propósito es permitir la integración de la tecnología Wave en otras aplicaciones, que podrán compartir objetos (de los tipos antes mencionados) de forma federada y en tiempo real. Su código fuente está disponible en GitHub [23], así como sus instrucciones de instalación (Ver el Readme en GitHub).

Para este proyecto se ha usado el framework SwellRT como base para la migración de la tecnología de Apache Wave a la plataforma Android [24]. Se pretende con esto que SwellRT haga uso de las funcionalidades nativas de Android.

## **2.4. Metodología de Migración**

### **2.4.1. Objetivo**

### **2.4.2. Entorno de desarrollo, Construcción y Depuración**

Eclipse

Construcción por consola

Depuración por consola

### **2.4.3. Identificación y Solución de Problemas**

Conexion HTTP

Conexion WebSocket

Login

### **2.4.4. Organización y Resultados**

Servicio Android

Diagramas y Dependencias

# Capítulo 3

## Creación de aplicación Android (NOMBRE DE APP)

### 3.1. 1ª Parte: Programas Políticos

3.1.1. Estado del Arte

3.1.2. Intencion

3.1.3. Objetivos

3.1.4. Usabilidad

### 3.2. 2ª Parte: Propuestas y Comparativas

3.2.1. Estado del Arte

3.2.2. Intencion

3.2.3. Objetivos

3.2.4. Usabilidad

### 3.3. Tecnologías y Metodologías de la app

3.3.1. Base de Datos

3.3.2. Service REST

3.3.3. Frontend



# Capítulo 4

## Resultados y Conclusiones

### 4.1. Discusion de Resultados

### 4.2. Conclusiones



# Capítulo 5

## Trabajo a Futuro

### 5.1. Mejoras





# Bibliografía

- [1] Inc. Google. Meet Google Wave.  
<http://googlecode.blogspot.com.es/2009/05/hello-world-meet-google-wave.html>.
- [2] Inc. Google. Google Wave Federation Protocol Over XMPP.  
<http://wave-protocol.googlecode.com/hg/spec/federation/wavespec.html>.
- [3] Inc. Google. End of Google Wave.  
<https://support.google.com/answer/1083134?hl=en>.
- [4] Google Docs.  
<https://drive.google.com/>.
- [5] Google+.  
<https://plus.google.com/>.
- [6] Apache. Apache Wave (Incubating).  
<http://incubator.apache.org/wave/about.html>.
- [7] Apache. Apache License 2.0.  
<http://www.apache.org/licenses/LICENSE-2.0>.
- [8] Inc. Google. Wave Federation.  
<http://www.waveprotocol.org/federation>.
- [9] Google. Google Wave Federation Architecture White Paper.  
<http://wave-protocol.googlecode.com/hg/whitepapers/google-wave-architecture/google-wave-architecture.html>.
- [10] Peter Saint-Andre. Extensible messaging and presence protocol (xmpp): Core. 2011.  
RFC 6120 Available at <http://tools.ietf.org/html/rfc6120>.
- [11] Understanding Operational Transformation.  
<http://www.codecommit.com/blog/java/understanding-and-applying-operational-transformation>.

- [12] Inc. Google. Google Wave Operational Transformation.  
<http://www.waveprotocol.org/whitepapers/operational-transform>.
- [13] Inc. Google. Google Wave Conversation Model.  
<https://wave-protocol.googlecode.com/hg/spec/conversation/convspec.html>.
- [14] Inc. Google. Google Wave API Overview.  
<http://www.waveprotocol.org/wave-apis>.
- [15] Inc. Google. Google Wave Extensions.  
<http://www.waveprotocol.org/wave-apis/extensions>.
- [16] Apache. Wave In A Box.  
<http://www.waveprotocol.org/wave-in-a-box/>.
- [17] Oracle. OpenJDK.  
<http://openjdk.java.net/>.
- [18] Inc. Google. Google Web Toolkit.  
<http://www.gwtproject.org/>.
- [19] Apache. Install WIAB.  
<https://cwiki.apache.org/confluence/display/WAVE/Install+WIAB>.
- [20] Apache. WIAB Repository.  
<https://github.com/apache/incubator-wave>.
- [21] WIAB Server Example.  
<http://waveinabox.net/>.
- [22] P2PValue. P2P Value European Project.  
<http://www.p2pvalue.eu/>.
- [23] P2PValue. SwellRT, a real-time federated collaboration framework.  
<https://github.com/P2Pvalue/swellrt>.
- [24] Inc. Google. Android Developers Site.  
<http://developer.android.com/index.html>.