

# **Análisis, Simulación y Automatización de Comunicaciones Móviles**

**Trabajo Final de Grado**

Andres La Torre



*A la memoria de Don Felipe, presente en cada página de este proyecto.*



Nada hubiese sido posible sin el constante cariño y apoyo de mis padres y hermana. Mucho menos sin Natalia, mi compañera de fierro en cada etapa de este proyecto, en cada momento de mi vida.

A todos ellos, mi más sincera gratitud.



# Índice

<b>Resumen .....</b>	<b>1</b>
<b>Palabras Claves .....</b>	<b>2</b>
<b>Introducción.....</b>	<b>3</b>
<b>Objetivo del Proyecto.....</b>	<b>4</b>
<b>Destinatarios.....</b>	<b>5</b>
<b>Beneficios .....</b>	<b>5</b>
<b>Estudio Técnico.....</b>	<b>6</b>
<b>Resumen Técnico.....</b>	<b>8</b>
<b>Metodología .....</b>	<b>10</b>
<b>Actividades Realizadas .....</b>	<b>11</b>
<b>Riesgos Esperados y Supuestos Asumidos.....</b>	<b>12</b>
<b>Resultados Alcanzados .....</b>	<b>13</b>
<b>Control de Costos .....</b>	<b>16</b>
<b>Inversión Requerida .....</b>	<b>17</b>
<b>Proyección de Costos de Operación y Mantenimiento.....</b>	<b>17</b>
<b>Análisis de Viabilidad Comercial .....</b>	<b>17</b>
<b>Análisis Financiero.....</b>	<b>18</b>
<b>Estudio Ambiental .....</b>	<b>18</b>
<b>Estudio Social.....</b>	<b>18</b>
<b>Evaluación Económica.....</b>	<b>19</b>
<b>Conclusiones .....</b>	<b>20</b>
<b>Glosario .....</b>	<b>21</b>
<b>Referencias y Bibliografía.....</b>	<b>26</b>
<b>Anexo 1: Introducción a las Llamadas Móviles .....</b>	<b>27</b>
<b>Anexo 2: Llamada Móviles.....</b>	<b>36</b>

<b>Anexo 3: Mensajes CAP2.....</b>	<b>48</b>
<b>Anexo 4: Diagnóstico de las Llamadas Móviles .....</b>	<b>86</b>
<b>Anexo 5: Entorno de Pruebas .....</b>	<b>89</b>
<b>Anexo 6: Diagnóstico del Entorno de Pruebas .....</b>	<b>97</b>
<b>Anexo 7: Reingeniería del Entorno de Pruebas.....</b>	<b>106</b>
<b>Anexo 8: Utilización del Entorno de Pruebas .....</b>	<b>125</b>

## **Resumen**

El presente proyecto surge como una respuesta a la necesidad de comprender la situación actual de las comunicaciones móviles y sus nuevas tendencias, conocimientos fundamentales para detectar y solucionar los problemas que aquejaban a un entorno de pruebas automatizadas, empleado para validar los cambios aplicados sobre la lógica utilizada por una operadora móvil para establecer y tarifar llamadas de voz.

Para alcanzar las metas planteadas se procedió a consultar los diversos estándares utilizados en las comunicaciones móviles, documentando la información obtenida y elaborando un diagnóstico sobre la situación actual y venidera. Con estas nociones se determinaron aquellos aspectos del entorno de pruebas que necesitaban ser mejorados, para luego llevar adelante una reingeniería que lo transformase en una herramienta ágil y simple de utilizar. Finalmente, tras implementar los cambios se establecieron los principales lineamientos a considerar de cara al futuro, buscando garantizar la utilidad del entorno con el transcurso del tiempo.

Los resultados obtenidos fueron sumamente positivos. Desde el punto de vista estrictamente numérico, la simplicidad y la agilidad ofrecida tras la reingeniería del entorno se vio reflejada en un ahorro considerable del tiempo dedicado a la etapa de pruebas. Sin embargo, la consecuencia más relevante del presente proyecto se relaciona a la detallada documentación que se ha proporcionado sobre los aspectos investigados y la reingeniería realizada; lo cual constituye un legado más que interesante para aquellas personas que, en un futuro no muy lejano, deseen comprender que hay detrás de una llamada móvil, en pos de explotar las capacidades del entorno de pruebas o, incluso, ampliarlas.

## **Palabras Claves**

Análisis.

Automatización.

Casos de Prueba.

Comunicaciones Móviles.

Entorno de Pruebas.

Java.

Operadora Móvil.

Red Inteligente.

Reingeniería.

Simulación.

## **Introducción**

Resulta innegable reconocer el constante crecimiento que hay en el mundo de las telecomunicaciones, más aún, si uno se detiene a observar el submundo de las comunicaciones móviles. De hecho, dicho crecimiento es de tal envergadura que uno no termina de adaptarse a una tecnología que aparece otra nueva y mejor.

Sin embargo hay algo que no cambia, la estrecha relación que existe entre las comunicaciones móviles y los sistemas que las respaldan. En efecto, hay sistemas de toda clase detrás de ellas: Desde los de más bajo nivel que permiten enrutar una llamada, mensaje o sesión de datos hacia un destino, pasando por los que proporcionan la tarificación y el cobro del servicio y llegando a las aplicaciones web que brindan la posibilidad de autogestión a los usuarios.

En el caso particular de este proyecto, son de interés los sistemas que permiten enlazar las partes involucradas en las comunicaciones móviles, como así también los que efectúan la tarificación y el cobro del servicio brindado. Esta clase de sistemas se ve sujeta a cambios constantes, no sólo por la vorágine tecnológica previamente mencionada, sino también por el crecimiento ininterrumpido de las ofertas comerciales provenientes de las empresas de comunicación.

Esta situación hace necesaria la aceleración del proceso de puesta en producción de cambios en todos sus aspectos. Esto alcanza la no menos importante etapa de pruebas pre-productivas, donde queda prácticamente descartada la realización de validaciones manuales, mediante el uso de celulares, principalmente por atentar contra la necesidad de aceleración previamente mencionada.

Surgen de esta manera otras alternativas que ofrecen pruebas de carácter automático, logradas mediante el uso de simulaciones. Se trata de entornos o marcos de automatización de pruebas, que tienen por objeto simular de manera sencilla las diversas comunicaciones móviles sobre las que se desea trabajar.

Es el propósito de este proyecto brindar una introducción sobre todo aquello que hay detrás del establecimiento y la tarificación de una llamada móvil, para luego detallar cómo se automatiza la simulación de las mismas en un entorno de pruebas ágil y sencillo de utilizar.

## **Objetivo del Proyecto**

### ***Objetivo General***

Llevar adelante la reingeniería y la posterior utilización de un Framework Automatizado de Pruebas en Java, que permita simular llamadas móviles y generar diversos escenarios de prueba automatizados, todo de manera ágil y simple, acorde a las necesidades actuales de una operadora móvil.

### ***Objetivos Específicos***

Investigar y establecer los lineamientos básicos propios de las comunicaciones móviles, necesarios para marcar un punto de partida en la redefinición del entorno de pruebas existente.

Interpretar y definir los pasos a seguir en lo que a nuevas tecnologías de comunicación móvil se refiere, necesarios para la evolución del entorno de pruebas y para garantizar su utilidad en el futuro.

Redefinir, mejorar e implementar aquellas funcionalidades del entorno de pruebas necesarias para: Simular los componentes de una Red Inteligente y su comunicación, interactuar con los servidores que contienen la lógica del servicio, configurar los flujos de señalización de las llamadas y validar los resultados esperados tras su finalización.

## **Destinatarios**

El presente proyecto está destinado al Centro de Desarrollo RI/PP, donde desempeño mis actividades laborales. El mismo, forma parte de la Dirección de IT de la empresa líder en comunicaciones del país.

Este proyecto surge como una propuesta para atacar la lentitud en la realización de pruebas pre-productivas y para satisfacer la necesidad de automatizar las mismas en un entorno rápido y estable.

## **Beneficios**

La justificación de este proyecto está dada por la necesidad de contar con un entorno automatizado de pruebas para llamadas de voz, que sea ágil y simple de configurar y utilizar. Esto se traduce en múltiples beneficios: Los analistas, responsables de llevar adelante las pruebas, verán su trabajo simplificado; la operadora móvil verá acelerada una de las etapas del proceso de puesta en producción de cambios en sus sistemas; los clientes contarán con la posibilidad de acceder a las últimas ofertas comerciales de manera rápida y concisa.

Además, se lleva a cabo una investigación detallada de los protocolos de capa de aplicación utilizados en las comunicaciones móviles. Dicha investigación conduce a la compilación y el resumen de la información en un único documento; lo cual es bastante útil, considerando que hoy en día esta información se encuentra dispersa por diversos estándares asociados a las distintas entidades reguladoras.

Finalmente, cabe remarcar que no es habitual encontrarse con herramientas como la implementada durante el presente proyecto, es decir, que permitan la simulación de llamadas y a su vez sean fáciles de manejar y adaptar a las necesidades de cambio de una operadora móvil. Precisamente allí radica la novedad y la ventaja sobre otros proyectos: No se trata de una solución enlatada y costosa, es un proyecto in-house que puede ser moldeado de acuerdo a las necesidades del momento, sin la intervención de terceros.

## Estudio Técnico

Para definir la modalidad de las pruebas, se contemplaron las siguientes alternativas:

<b>Probar con Simulaciones (*)</b>	<b>Probar con Celulares</b>
<b>Ventajas</b>	<b>Ventajas</b>
Rápido, ágil y fácil de configurar Automatizable y reutilizable Requiere a una persona en forma esporádica	Son comunicaciones móviles 100% reales
<b>Desventajas</b>	<b>Desventajas</b>
No son comunicaciones móviles 100% reales	Excesivamente lento y arduo de configurar Manual Requiere a dos personas en forma permanente

Para definir el proveedor del entorno de pruebas, se contemplaron las siguientes alternativas:

<b>Solución Propia Actual (*)</b>	<b>Adquisición de Solución de Terceros</b>
<b>Ventajas</b>	<b>Ventajas</b>
No requiere erogación de dinero Amplia experiencia en su uso Código fuente disponible (solución in-house)	Solución robusta e integradora
<b>Desventajas</b>	<b>Desventajas</b>
Necesita ser modificada	Requiere erogación de dinero Requiere una capacitación para utilizarla Código fuente no disponible (solución enlatada)

Para definir la estrategia de trabajo a aplicar sobre el entorno de pruebas, se contemplaron las siguientes alternativas:

<b>Reingeniería de Solución Actual (*)</b>	<b>Corrección y Mejora de Solución Actual</b>
<b>Ventajas</b>	<b>Ventajas</b>
Genera un entorno ágil y simple de usar Cubre los escenarios de prueba tradicionales Solución perdurable en el tiempo	Requiere una menor inversión de tiempo No requiere tener conocimientos adicionales

<b>Desventajas</b>	<b>Desventajas</b>
Requiere una mayor inversión de tiempo	Siguen habiendo problemas de fondo
Requiere investigar el funcionamiento de las RI	No cubre los escenarios de prueba tradicionales
	Solución temporal

Para definir como adquirir los conocimientos teóricos necesarios para llevar adelante la reingeniería del entorno de pruebas, se contemplaron las siguientes alternativas:

<b>Autoaprendizaje (*)</b>	<b>Curso de Capacitación</b>
<b>Ventajas</b>	<b>Ventajas</b>
No requiere una erogación de dinero adicional	Contenidos ya investigados y depurados
Alcance del aprendizaje autodefinido	
Material disponible en la web	
<b>Desventajas</b>	<b>Desventajas</b>
Investigación y depuración requerida	Requiere una erogación de dinero adicional
	Alcance del aprendizaje predefinido
	Material privado de terceros

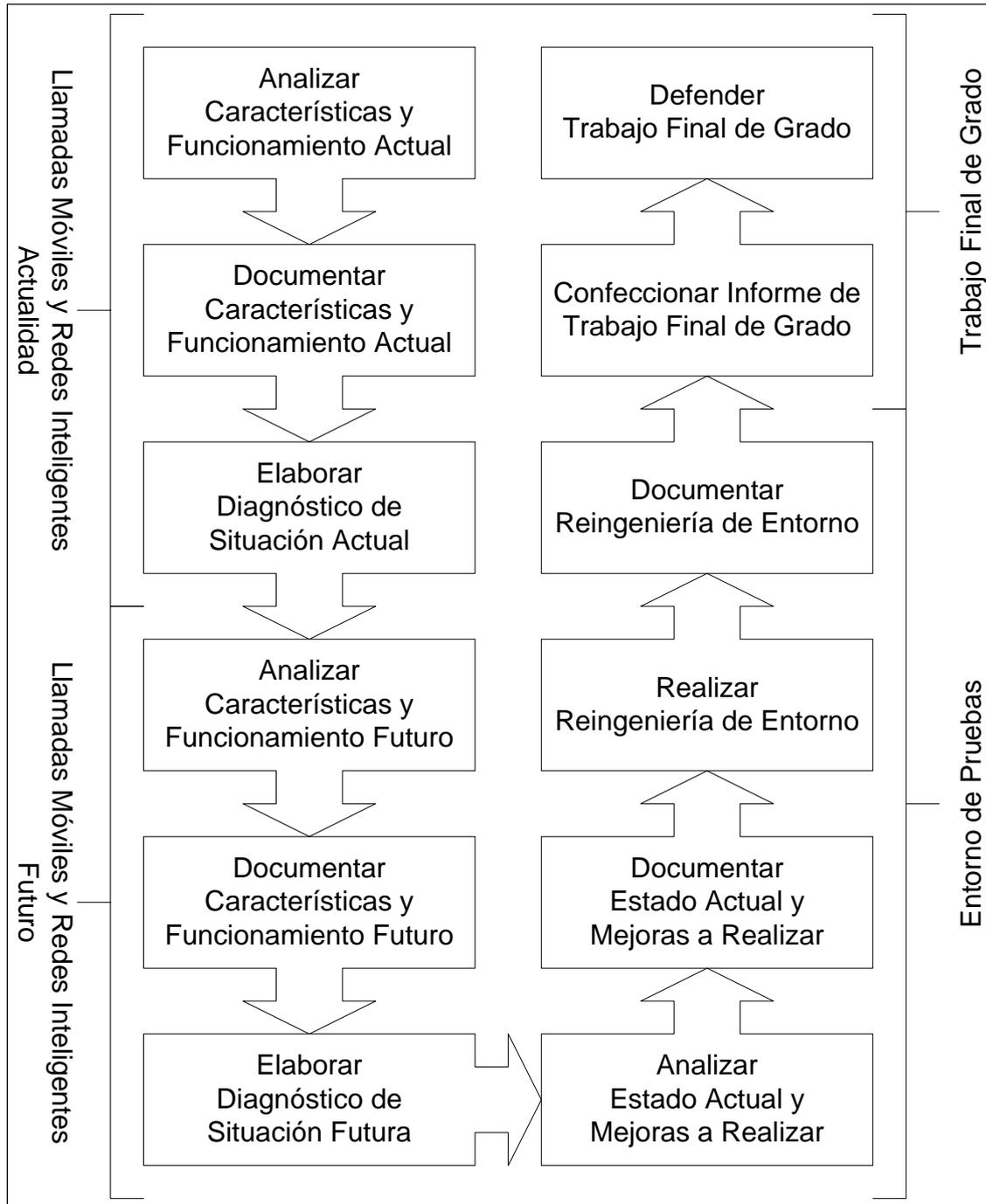
Para definir y diseñar los escenarios a incluir en el entorno de pruebas, se contemplaron las siguientes alternativas:

<b>Proceso Unificado de Rational / UML (*)</b>	<b>Metodología Merise</b>
<b>Ventajas</b>	<b>Ventajas</b>
Enfoque actual y orientado a objetos	Conocimiento amplio de la metodología
Fácil adaptación a cambios en el sistema	
Reutilización de código y componentes	
<b>Desventajas</b>	<b>Desventajas</b>
Conocimiento reducido de la metodología	Poco usada en la actualidad
	Rígida y secuencial
	Complejidad para reutilizar el código

(\*) Alternativas que prevalecieron.

## Resumen Técnico

El siguiente diagrama de bloques refleja las distintas etapas del presente proyecto:



El estudio técnico realizado arrojó los siguientes resultados:

- Las pruebas del sistema se realizan mediante el uso de simulaciones, en lugar de celulares.
- El entorno de pruebas que se utiliza se basa en la solución propia actual, en lugar de la adquisición de una solución de terceros.
- La estrategia de trabajo que se aplica sobre el entorno de pruebas consiste en realizar una reingeniería completa del mismo, en lugar de sólo realizar cambios y mejoras menores.
- Los conocimientos necesarios para llevar adelante la reingeniería del entorno de pruebas se adquieren mediante el autoaprendizaje, en lugar de la realización de un curso de capacitación.
- La definición y el diseño de los escenarios de pruebas incluidos en el entorno se realizan aplicando el Proceso Unificado de Rational (apoyado en UML), en lugar de la Metodología Merise.

El Framework Automatizado de Pruebas en Java se encarga de simular llamadas móviles y validar los resultados, mediante la implementación de estas funcionalidades:

- Simulación de los componentes de una Red Inteligente y su comunicación.
- Interacción con los servidores que contienen la lógica del servicio y los datos en los que esta se apoya (SO Solaris / Sybase ASE).
- Definición y configuración de diferentes flujos de mensajes de señalización, propios de una llamada móvil.
- Validación de registros y notificaciones recibidas tras una llamada móvil.
- Definición de los datos iniciales por defecto.
- Validación de los resultados esperados.
- Casos de prueba (test cases).

## **Metodología**

La primera etapa del presente proyecto está vinculada con el análisis y diagnóstico de las características y funcionalidades, actuales y venideras, de las llamadas móviles y las redes inteligentes.

Para llevar a cabo estas actividades, se consultaron los diversos estándares disponibles en la web y se documentaron aquellas características que son de interés para el presente proyecto (Ver Anexo 1: Introducción a las Llamadas Móviles, Anexo 2: Llamadas Móviles y Anexo 3: Mensajes CAP2).

Una vez finalizada esta compilación, se procedió a organizarla en un único documento y a elaborar un diagnóstico de la situación, tanto actual como venidera. Dicho diagnóstico permitió saber, por un lado, dónde estamos parados hoy en día y, por el otro, hacia dónde vamos, piedras angulares para la realización de la segunda parte de este proyecto (Ver Anexo 4: Diagnóstico de las Llamadas Móviles).

Precisamente, esta segunda etapa está vinculada con la reingeniería de un entorno automatizado de pruebas en Java (basado en JUnit), con el objetivo de convertirlo en uno ágil y simple.

En este caso, se analizaron cada uno de los eslabones que conforman el entorno de pruebas sobre el que se trabajó, documentando todos los aspectos que son de interés de cara al futuro, sobre todo aquellos a mejorar y en los que se basa la reingeniería (Ver Anexo 5: Entorno de Pruebas y Anexo 6: Diagnóstico del Entorno de Pruebas).

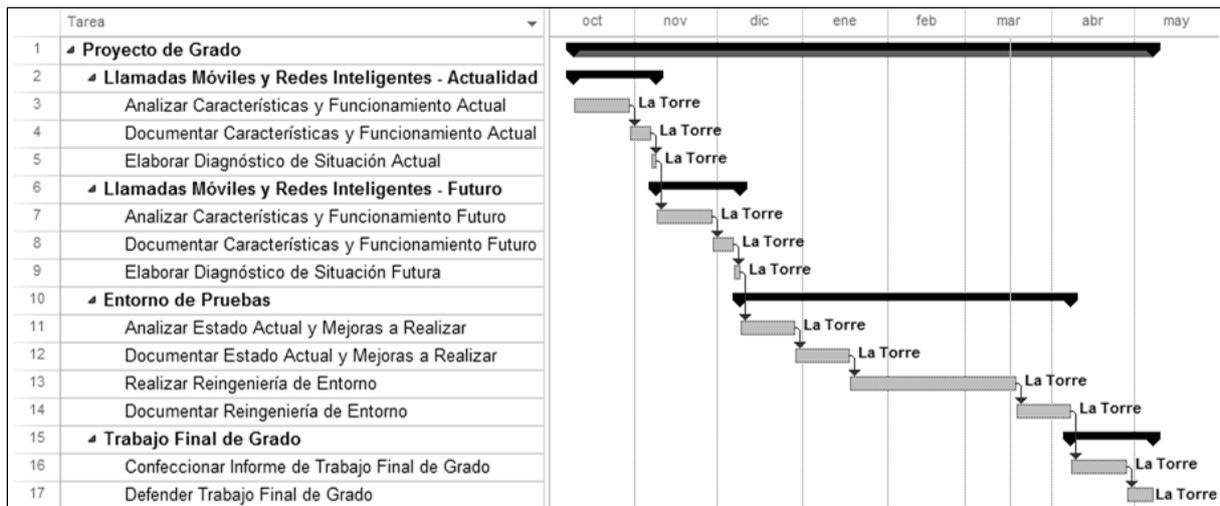
Una vez finalizado el relevamiento, se comenzaron a aplicar los cambios sobre el entorno de pruebas, para lograr convertirlo en uno acorde a las necesidades actuales de la operadora móvil (Ver Anexo 7: Reingeniería del Entorno de Pruebas).

Por último, completados los cambios, se procedió a crear una serie de escenarios vinculados a las llamadas móviles más tradicionales, con el objetivo de probar el correcto funcionamiento del entorno y definir las pautas básicas para su utilización (Ver Anexo 8: Utilización del Entorno de Pruebas).

## Actividades Realizadas

A continuación se detallan las actividades realizadas como parte del presente proyecto:

Tarea	Duración	Comienzo	Fin	RRHH	Pred
1 <b>Proyecto de Grado</b>	210 días	lun 10/10/16	dom 07/05/17	La Torre	
2 <b>Llamadas Móviles y Redes Inteligentes - Actualidad</b>	30 días	lun 10/10/16	mar 08/11/16	La Torre	
3 Analizar Características y Funcionamiento Actual	20 días	lun 10/10/16	sáb 29/10/16	La Torre	
4 Documentar Características y Funcionamiento Actual	8 días	dom 30/10/16	dom 06/11/16	La Torre	3
5 Elaborar Diagnóstico de Situación Actual	2 días	lun 07/11/16	mar 08/11/16	La Torre	4
6 <b>Llamadas Móviles y Redes Inteligentes - Futuro</b>	30 días	mié 09/11/16	jue 08/12/16	La Torre	
7 Analizar Características y Funcionamiento Futuro	20 días	mié 09/11/16	lun 28/11/16	La Torre	5
8 Documentar Características y Funcionamiento Futuro	8 días	mar 29/11/16	mar 06/12/16	La Torre	7
9 Elaborar Diagnóstico de Situación Futura	2 días	mié 07/12/16	jue 08/12/16	La Torre	8
10 <b>Entorno de Pruebas</b>	120 días	vie 09/12/16	vie 07/04/17	La Torre	
11 Analizar Estado Actual y Mejoras a Realizar	20 días	vie 09/12/16	mié 28/12/16	La Torre	9
12 Documentar Estado Actual y Mejoras a Realizar	20 días	jue 29/12/16	mar 17/01/17	La Torre	11
13 Realizar Reingeniería de Entorno	60 días	mié 18/01/17	sáb 18/03/17	La Torre	12
14 Documentar Reingeniería de Entorno	20 días	dom 19/03/17	vie 07/04/17	La Torre	13
15 <b>Trabajo Final de Grado</b>	30 días	sáb 08/04/17	dom 07/05/17	La Torre	
16 Confeccionar Informe de Trabajo Final de Grado	20 días	sáb 08/04/17	jue 27/04/17	La Torre	14
17 Defender Trabajo Final de Grado	10 días	vie 28/04/17	dom 07/05/17	La Torre	16



## Riesgos Esperados y Supuestos Asumidos

### **Riesgos Humanos**

Imposibilidad de cumplir con las horas diarias de trabajo pactadas y por consiguiente con los tiempos de presentación del proyecto.

*Plan de Mitigación:* Para evitar no poder cumplir con el acuerdo de dos horas diarias de trabajo durante los días laborales, se ha previsto recuperar las horas perdidas durante los fines de semana y feriados.

*Plan de Contingencia:* En caso de no cumplir con la fecha final de presentación del proyecto, se ha previsto solicitar tres meses adicionales para completarlo.

*Resultado:* El plan de mitigación concebido fue suficiente para evitar los riesgos previstos.

### **Riesgos Técnicos**

Obsolescencia de los estándares usados en las comunicaciones móviles, con posterior caída en desuso del entorno de pruebas.

*Plan de Mitigación:* Para evitar que el entorno de pruebas caiga rápidamente en desuso en caso de confirmarse la obsolescencia en cuestión, se ha previsto como parte del proyecto analizar las pautas básicas a seguir en cuanto a nuevas tecnologías y estándares se refiere.

*Plan de Contingencia:* En caso de producirse dicha obsolescencia, se ha previsto tomar las pautas analizadas en el plan de mitigación y con ellas, llevar a cabo una nueva reingeniería del entorno para adaptarlo a los nuevos estándares, todo de manera ágil.

*Resultado:* El plan de mitigación concebido fue suficiente para evitar los riesgos previstos.

### **Riesgos Financieros y Económicos**

No se presentaron riesgos financieros y económicos de consideración.

### **Riesgos Físicos y Naturales**

No se presentaron riesgos físicos y naturales de consideración.

## Resultados Alcanzados

### **Mejora en los Tiempos de Creación y Ejecución de Pruebas**

La determinación de los resultados alcanzados se realizó gracias a la comparación de los tiempos empleados para probar dos grupos de proyectos, abordados por igual cantidad de equipos de trabajo.

Para definir los tiempos de las pruebas realizadas sobre el entorno original, se tomaron los valores promedio de aquellos desarrollos encarados antes de llevarse a cabo la reingeniería.

A su vez, para el caso de las pruebas manuales, se contemplaron los tiempos de otro grupo de trabajo, responsable de llevar adelante una prueba integral sobre los mismos desarrollos, aunque valiéndose de llamadas reales.

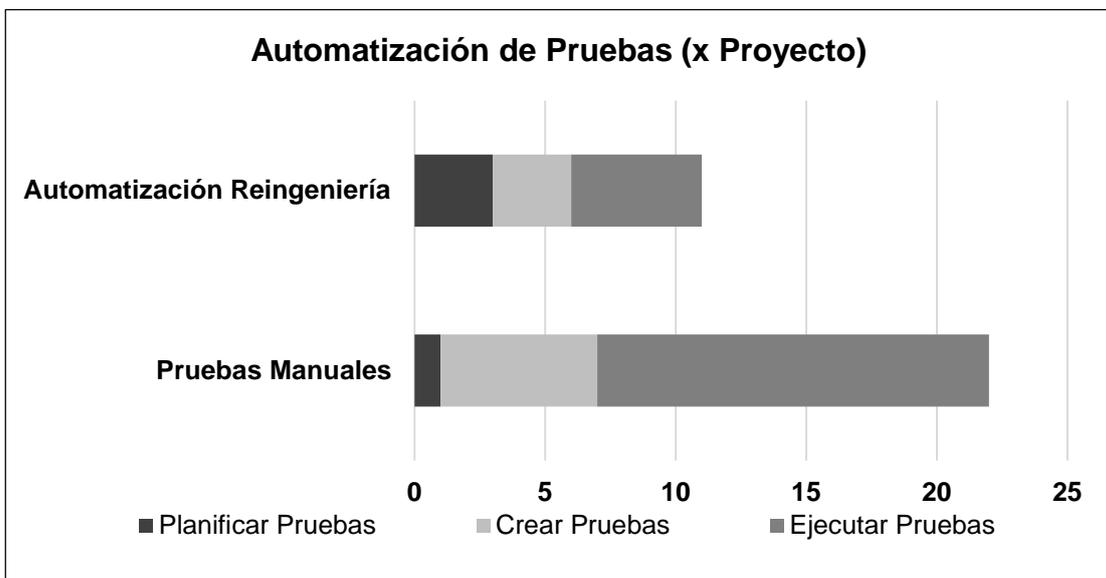
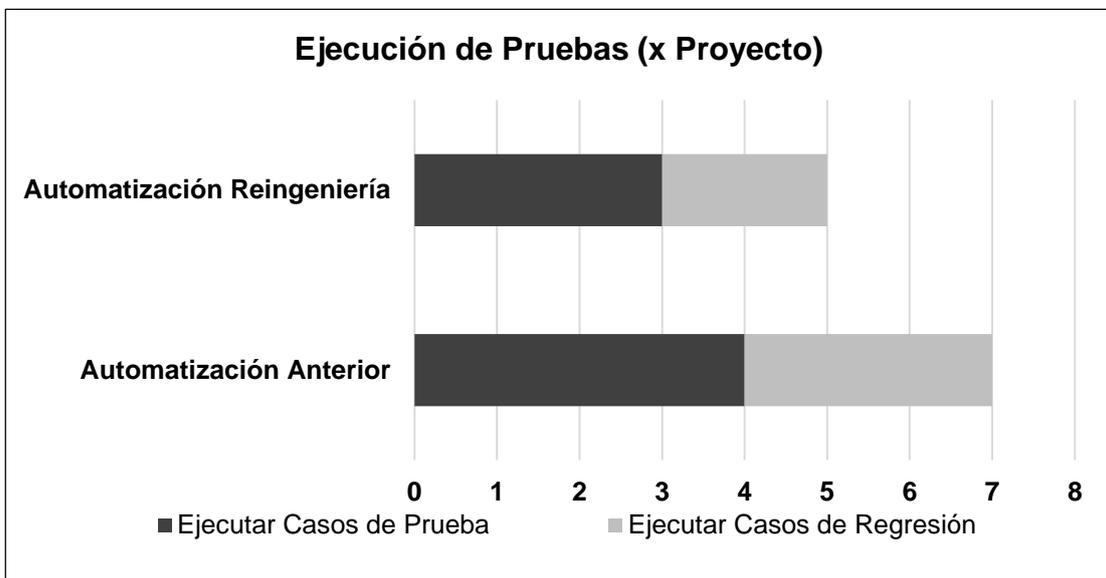
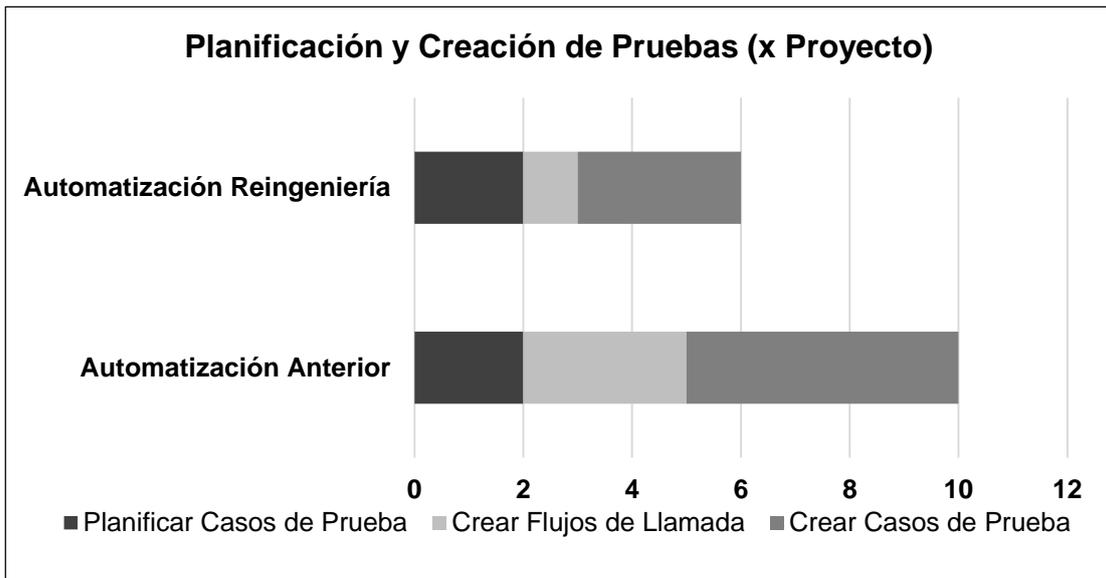
Finalmente, a la hora de establecer los tiempos tras la reingeniería, se tomaron los resultados promedio arrojados en los proyectos encarados tras su implementación.

<b>Ítem</b>	<b>Pruebas Manuales</b>	<b>Automatización Anterior</b>	<b>Automatización Reingeniería</b>
Planificar casos de prueba	1 día	2 días	2 días
Crear flujos de llamada	0 días	3 días	1 día
Crear casos de prueba	6 días	5 días	3 días
Ejecutar casos de prueba	10 días	4 días	3 días
Ejecutar pruebas de regresión	5 días	3 días	2 días
<b>Total</b>	<b>22 días</b>	<b>17 días</b>	<b>11 días</b>

La comparación de los tiempos previos y posteriores a la reingeniería refleja una aceleración en la creación de nuevos flujos de llamada y casos de prueba en el entorno, asociada principalmente a la simplificación de dichas tareas.

Además, los tiempos de ejecución de las pruebas se han visto ligeramente mejorados, sobre todo porque los casos de prueba creados se caracterizan por ser robustos y perdurables en el tiempo.

En cuanto a la comparación con las pruebas manuales, si bien estas demandan menos tiempo de planificación, es a la hora de la ejecución que se ven los beneficios, principalmente cuando dicha tarea ha de ser repetida tras la corrección de un error.



**Resultados No Cuantificables**

Existe un resultado asociado al presente proyecto que no puede ser cuantificado; no obstante, vale la pena mencionarlo por su impacto determinante. El mismo se vincula a las investigaciones realizadas sobre los estándares actuales de las comunicaciones móviles y las tendencias futuras que giran en torno a ellas.

Dichas investigaciones ha sido altamente positivas, principalmente porque han permitido:

- Comprender, con un alto nivel de detalle, cómo es que funcionan las comunicaciones móviles actuales, piedra angular para detectar y corregir aquellos puntos débiles del entorno de pruebas utilizado.
- Demostrar que el entorno en cuestión, lejos de caer en la obsolescencia con la implementación de las nuevas tendencias, muestra un gran potencial para adaptarse al cambio de paradigmas.
- Documentar las características actuales y los lineamientos futuros de las llamadas móviles, dejando clara y concisa evidencia de su funcionamiento, tanto para miembros actuales como futuros de la empresa.

## Control de Costos

Los siguientes recursos, necesarios para el proyecto, se poseían con antelación:

- Locales de Trabajo: Uno fue proporcionado en 2008 por la empresa a la que se destina el proyecto, mientras que el otro es un domicilio particular.
- Notebook Asus: Fue adquirida en 2015 para uso particular.
- Sun Workstation: Fue proporcionada en 2008 por la empresa a la que se destina el proyecto, para desempeñar tareas laborales inherentes a la misma.

El siguiente recurso es parte de la inversión inicial:

- PC Dell: Proporcionada por la empresa a la que se destina el proyecto, para desempeñar tareas laborales inherentes a la misma.

Los siguientes recursos son gratuitos o pertenecen a la empresa destinataria:

- Software de Terceros: Eclipse Luna 4.4.2, Java JDK 1.7.0, Apache MVN 3.3.1 y Apache SVN 1.8.10.
- Software de Empresa: TINPtmpl-520400, TINPbase-520400 y TINSppvoice-410400.
- Software de Terceros con Licencia de Empresa: Sybase ASE 15.7.

Los siguientes recursos, necesarios para el proyecto, son valorizables:

- Suministros de Librería (hojas, encuadernación y CDs): \$1500
 

<i>Resmas de papel:</i>	\$300
<i>Cartuchos de tinta negra:</i>	\$600
<i>CDs:</i>	\$50
<i>Encuadernaciones:</i>	\$550
- Recursos Humanos (analista de desarrollo y pruebas): \$24000
 

<i>Días de trabajo efectivo (sin preparación de TFG):</i>	120
<i>Horas trabajadas por día:</i>	2
<i>Cantidad de recursos humanos:</i>	1
<i>Honorarios por hora trabajada:</i>	\$100

Por lo tanto, el costo presupuestado para todo el proyecto es de: \$25500. El mismo se ha mantenido inalterable durante todas las actividades realizadas.

## Inversión Requerida

El presente proyecto posee una única inversión inicial, a cuenta de la empresa destinataria:

- PC Dell (Procesador Intel Core i5, 8GB de RAM y Windows 7): \$7000

## Proyección de Costos de Operación y Mantenimiento

Los costos proyectados para el funcionamiento óptimo del proyecto son:

- Costos Fijos: \$3000
  - Luz:* \$500
  - Gas:* \$500
  - Agua:* \$500
  - Telefonía e Internet:* \$1500
- Costos Variables: \$25500
  - Suministros de Librería:* \$1500
  - Recursos Humanos:* \$24000

## Análisis de Viabilidad Comercial

El resultado de este proyecto es la reingeniería de un entorno de pruebas automatizado, el cual es parte de una solución in-house no destinada a la comercialización.

De todas formas, con el propósito de llevar adelante los análisis financieros y económicos posteriores, se le asigna una valoración estimativa. La misma equivale a la tercera parte del costo de una licencia para utilizar una de las herramientas de automatización de pruebas más conocida en el mercado: HP UTF (QTP). Dicho costo es de aproximadamente \$120000.

Por lo tanto, el valor comercial del presente proyecto es: \$40000.

## Análisis Financiero

A continuación se detalla el análisis de punto de equilibrio y sus resultados:

- CF (Costos Fijos): \$10000 (incluye inversión inicial)
- CV (Costos Variables): \$25500
- VT (Ventas Totales): \$40000

Las ventas necesarias para no tener pérdidas ni ganancias son de: \$27586,21.

$$PE = \frac{CF}{1 - \frac{CV}{VT}} = \frac{\$10000}{1 - \frac{\$25500}{\$40000}} = \frac{\$10000}{0,3625} = \$27586,21$$

La utilidad neta es del 31%. El 69% restante se destina al pago de los costos.

$$PE = \frac{CF}{VT - CV} = \frac{\$10000}{\$40000 - \$25500} = \frac{\$10000}{\$14500} = 0,69 = 69\%$$

## Estudio Ambiental

El proyecto tiene un impacto ambiental muy limitado que no permite tomar medidas para reducirlo. Dicho impacto está vinculado con dos aspectos:

- Uso de electricidad, gas y agua para realizar la reingeniería del entorno.
- Impresión de toda la documentación vinculada al proyecto.

## Estudio Social

El presente proyecto tiene un impacto indirecto sobre la comunidad. El hecho de pasar a contar con un entorno automatizado de pruebas para llamadas de voz, ágil y simple de utilizar, se traduce en múltiples beneficios desde el plano social:

- Los analistas, responsables de llevar adelante las pruebas, verán su trabajo simplificado.
- La operadora móvil verá acelerada una de las etapas del proceso de puesta en producción de cambios en sus sistemas.
- Los clientes contarán con la posibilidad de acceder a las últimas ofertas comerciales de manera rápida y concisa.

## Evaluación Económica

A continuación se detalla la evaluación económica realizada, la cual permite establecer que el presente proyecto es viable:

Períodos	0	1	2	3	4
Valor del Sistema		\$10.000,00	\$10.000,00	\$10.000,00	\$10.000,00
<b>Total Ingresos</b>		<b>\$10.000,00</b>	<b>\$10.000,00</b>	<b>\$10.000,00</b>	<b>\$10.000,00</b>
Impuestos y Servicios por Operar		\$750,00	\$750,00	\$750,00	\$750,00
Gastos en Suministros de Librería		\$375,00	\$375,00	\$375,00	\$375,00
Salarios para Recursos Humanos		\$6.000,00	\$6.000,00	\$6.000,00	\$6.000,00
<b>Total Egresos</b>		<b>\$7.125,00</b>	<b>\$7.125,00</b>	<b>\$7.125,00</b>	<b>\$7.125,00</b>
Compra de PC Dell	\$7.000,00				
<b>Total Inversiones</b>	<b>\$7.000,00</b>				
No Aplicable					
<b>Total Valor Residual</b>					
<b>Flujo de Fondos Netos</b>	<b>-\$7.000,00</b>	<b>\$2.875,00</b>	<b>\$2.875,00</b>	<b>\$2.875,00</b>	<b>\$2.875,00</b>

Cálculos del Proyecto					
<b>Tasa de Descuento (PF Galicia)</b>					<b>9,08%</b>
<b>Tasa Interna de Retorno (TIR)</b>					<b>23,30%</b>
<b>Valor Actual Neto (VAN)</b>					<b>\$2.297,86</b>
VAN Periódico	\$2.297,86	\$2.635,68	\$2.416,28	\$2.215,15	\$2.030,75
Prd. de Recupero Actualizado	-\$7.000,00	-\$4.364,32	-\$1.948,04	\$267,11	\$2.297,86
Prd. de Recupero Sin Actualizar	-\$7.000,00	-\$4.125,00	-\$1.250,00	\$1.625,00	\$4.500,00
<b>Prd. Recupero Inversión (PRI)</b>					<b>3</b>
<b>Rentabilidad Inmediata (RI)</b>					<b>0,41</b>
Beneficios	\$0,00	\$9.167,58	\$8.404,46	\$7.704,86	\$7.063,49
Costos	\$7.000,00	\$6.531,90	\$5.988,18	\$5.489,71	\$5.032,74
<b>Relación Beneficio/Costo (B/C)</b>					<b>1,08</b>
<b>Retorno de la Inversión (ROI)</b>					<b>0,08</b>

## **Conclusiones**

El objetivo principal del presente proyecto consistía en llevar adelante la reingeniería de un entorno de pruebas automatizadas, basado en la simulación de llamadas móviles. Dicha actividad debía traducirse en una mayor agilidad y simpleza a la hora de utilizarlo, acorde a las necesidades actuales de una operadora móvil.

Analizando los resultados obtenidos tras la redefinición y la mejora de las funcionalidades de dicho entorno, se puede observar una interesante mejora en los tiempos destinados a las pruebas de sistema. Tras su implementación, estas se realizan un 35% más rápido, dando sobradas muestras de haber alcanzado el comportamiento deseado

Por otro lado, también se planteó como objetivo investigar sobre la situación actual de las comunicaciones móviles y sus nuevas tendencias, buscando de esta manera poder definir un punto de partida y los pasos a seguir de cara al futuro.

Dichas investigaciones fueron altamente positivas, ya que no sólo colaboraron con la detección y corrección de los puntos débiles del entorno de pruebas, sino que también permitieron demostrar su gran potencial para adaptarse al cambio de paradigmas que se está dando en las telecomunicaciones.

Finalmente, la detallada documentación de los aspectos investigados y de la reingeniería realizada en el presente proyecto, constituyen un legado más que interesante para aquellos miembros de la empresa que, en un futuro no muy lejano, decidan tomar la posta y llevar el entorno de pruebas un poco más allá.

## Glosario

*1G (First Generation)*: Primera generación de tecnologías vinculadas a las comunicaciones móviles, caracterizada por ser analógica y soportar llamadas de voz.

*2G (Second Generation)*: Segunda generación de tecnologías vinculadas a las comunicaciones móviles, caracterizada por ser digital y soportar llamadas de voz y datos.

*3G (Third Generation)*: Tercera generación de tecnologías vinculadas a las comunicaciones móviles, caracterizada por ampliar las capacidades de su antecesora en materia de transmisión de datos.

*4G (Fourth Generation)*: Cuarta generación de tecnologías vinculadas a las comunicaciones móviles, caracterizada por generar un cambio radical en la infraestructura de la red, con el objetivo de ampliar las capacidades de su antecesora en materia de transmisión de datos.

*AMPS (Advanced Mobile Phone System)*: Estándar definido para las primeras comunicaciones móviles, realizadas mediante el uso de terminales analógicas (1G).

*Automatización de Prueba*: Uso de un software especial para ejecutar pruebas y comparar los resultados obtenidos con los esperados, todo de manera automática.

*BCSM (Basic Call State Model)*: Modelo compuesto por una serie de estados predefinidos en los que puede encontrarse una llamada, junto con los eventos que disparan los cambios de uno a otro. Se trata de un concepto fundamental en las Redes Inteligentes.

*CAP2 (CAMEL Application Part Phase 2)*: Protocolo de capa de aplicación por excelencia del estándar SS7, utilizado principalmente para poder establecer y tarifar llamadas móviles.

*Caso de Prueba*: Conjunto de condiciones o variables bajo las cuáles un analista determinará si una aplicación, un sistema o una característica de éstos es parcial o completamente satisfactoria.

*CGI (Cell Global Identifier)*: Identificador unívoco de las celdas, principales responsables de brindar la cobertura necesaria para el establecimiento de comunicaciones móviles.

*Comunicación Móvil:* Comunicación que se da cuando el emisor y el receptor están, o pueden estar, en movimiento. La movilidad de estos dos elementos que se encuentran en los extremos de la comunicación hace que no sea factible la utilización de cables y, por lo tanto, utilicen una comunicación vía radio.

*CTP (Combined Test Platform):* Plataforma que integra el SCP y el SDP con el objeto de simplificar las distintas etapas de desarrollo y prueba, vinculadas a la lógica de servicio ofrecida por una determinada operadora móvil.

*Diameter:* Protocolo de red diseñado para ofrecer servicios de autenticación, autorización y contabilización (AAA) a aplicaciones con acceso a redes o de tipo IP Móvil.

*Eclipse:* Entorno integrado de desarrollo (IDE), principalmente utilizado para crear aplicaciones basadas en el lenguaje de programación Java.

*Entorno de Pruebas:* Estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos concretos de software, que puede servir de base para la organización y ejecución de pruebas.

*GPRS (General Packet Radio Service):* Servicio que permite la transmisión de datos mediante la conmutación de paquetes. En las comunicaciones móviles, es utilizado, entre otras cosas, para acceder a páginas web y servicios de mensajería sobre una Red IP.

*GSM (Global System for Mobile Communication):* Estándar definido para las comunicaciones móviles, realizadas por teléfonos celulares sobre una red digital (2G).

*IMS (IP Multimedia Subsystem):* Conjunto de especificaciones que describen la arquitectura necesaria para soportar la telefonía y los servicios de multimedia en Redes IP.

*IMSI (International Mobile Subscriber Identity):* Identificador unívoco de los suscriptores de una operadora móvil, utilizado por la Red Inteligente para saber hacia dónde enrutar el mensaje IDP y, de esta forma, establecer una comunicación móvil.

*In-house:* Anglismo utilizado para referirse a algo que es de carácter interno, propio de la empresa.

*IP (Internet Protocol)*: Principal protocolo de comunicación para enviar y recibir datos a través de redes, gracias a su capacidad para interconectarlas. Es la base de la familia de protocolos de internet y prácticamente se lo utiliza en todas las redes de telecomunicación, incluidas (recientemente) las de comunicación móvil (4G).

*Java*: Lenguaje de programación de propósito general, concurrente, orientado a objetos y diseñado específicamente para tener tan pocas dependencias de implementación como fuese posible.

*JTAF (Java Test Automation Framework)*: Entorno de pruebas basado en Java (JUnit), cuyo principal objetivo es permitir la ejecución de pruebas dentro de un marco controlado, en forma automática y basándose en la simulación de llamadas móviles.

*JUnit*: Conjunto de clases (framework) utilizadas en programación para hacer pruebas unitarias de aplicaciones Java. Es decir, en función de algún valor de entrada se evalúa el valor de retorno esperado; si la clase cumple con la especificación, entonces JUnit devolverá que el método de la clase pasó exitosamente la prueba; en caso contrario, devolverá un fallo en el método correspondiente.

*Llamada Móvil Desviada*: Llamada móvil que ha sido desviada a un tercero en forma automática. Se la conoce como Mobile Forwarding (MF).

*Llamada Móvil Entrante*: Llamada móvil vista desde la perspectiva de quien la recibe. Se la conoce como Mobile Terminated (MT).

*Llamada Móvil Saliente*: Llamada móvil vista desde la perspectiva de quien la emite. Se la conoce como Mobile Originated (MO).

*LTE Advanced (Long Term Evolution Advanced)*: Estándar definido para las comunicaciones móviles de alta velocidad, realizadas mediante el uso de terminales móviles y de datos (4G).

*NGC (Next Generation Charging)*: Framework perteneciente a JTAF, responsable de proveer los medios para crear y enviar o recibir y validar los mensajes CAP2 intercambiados en una llamada móvil.

*Operadora Móvil*: Compañía encargada de brindar servicios de comunicación inalámbrica o móvil, controlando todos los elementos necesarios para su venta y entrega: Espectro de radio, infraestructura de red, tarifación, atención al cliente, aprovisionamiento, etc.

*Portabilidad Numérica:* Funcionalidad que permite conservar el número telefónico cuando se cambia de operadora móvil dentro de un mismo país. Busca fomentar una competencia más equitativa entre los proveedores de servicios de comunicación móvil.

*PPT (Prepaid Services Testing):* Framework perteneciente a JTAF, responsable de proveer los medios para poder crear distintos flujos de llamada, los cuales son aplicados en los diversos casos de prueba donde se las simula.

*Proceso Unificado Rational:* Proceso de desarrollo de software que, junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos.

*Protocolo:* Sistema de reglas que permiten la comunicación entre dos o más partes de un sistema. Dichas reglas definen la sintaxis, semántica y sincronización de la comunicación.

*Red Inteligente:* Arquitectura de red estándar, tanto para la telecomunicación fija como móvil. Permite a cada operadora diferenciarse de las demás a través de los servicios de valor agregado que esta provee, los cuales son definidos en sus propios equipos, en lugar de los equipos de conmutación de la red o de las terminales móviles.

*Reingeniería:* Reestructuración de un proyecto, de una empresa o de una institución con el fin de mejorar sus resultados o sus beneficios.

*Roaming:* Anglismo utilizado para referirse a un suscriptor que utiliza la red de otra operadora móvil (nacional o internacional) para comunicarse.

*SCP (Service Control Point):* Plataforma separada de la red de conmutación y perteneciente a una determinada operadora móvil, responsable de alojar la lógica del servicio ofrecido.

*SDP (Service Data Point):* Plataforma responsable de proporcionar los datos persistentes en los que se apoya la lógica de servicio de una determinada operadora móvil (base de datos).

*Señalización:* Uso de señales para el control de las comunicaciones. Intercambio de información relativo al establecimiento y control de un circuito de telecomunicación y la gestión de la red.

*Simulación de Llamadas Móviles:* Imitación del flujo de mensajes de señalización intercambiados dentro de una red inteligente, necesario para el establecimiento y la tarificación de una llamada móvil.

*SMS (Short Message System):* Servicio disponible en las terminales móviles, responsable de permitir el envío y la recepción de mensajes de texto.

*SQL (Structured Query Language):* Lenguaje utilizado para la interacción con las bases de datos, el cual permite la manipulación de los objetos y datos allí alojados.

*SRP (Specialized Resource Point):* Plataforma responsable de proporcionar funciones complementarias dentro de una red de conmutación, como la reproducción de anuncios de voz o la recolección de tonos DTMF ingresados por los usuarios.

*SS7 (Signalling System Number 7):* Conjunto de protocolos de señalización telefónica empleado en la mayoría de las redes telefónicas mundiales. Su principal propósito es el establecimiento y finalización de llamadas, pero también tiene otros usos como: Traducir numeraciones, brindar los mecanismos de tarificación prepaga y permitir el envío de SMS.

*SSP (Service Switching Point):* Plataforma responsable de interceptar la señal de la llamada proveniente de la red de conmutación y enrutarla hacia una determinada operadora móvil, buscando de esta manera recibir las instrucciones sobre cómo proceder.

*STP (Signal Transfer Point):* Plataforma responsable de transportar las distintas consultas y respuestas enviadas entre los equipos de conmutación de la red y la lógica de servicio de una determinada operadora móvil.

*Sybase ASE:* Sistema de gestión de datos, de alta escalabilidad y rendimiento, con soporte a grandes volúmenes de datos, transacciones y usuarios.

*UMTS (Universal Mobile Telecommunications System):* Evolución y ampliación del estándar GSM para las comunicaciones móviles (3G).

*VoLTE (Voice over Long Term Evolution):* Servicio de llamadas de voz ofrecido sobre una red LTE Advanced (IP).

## Referencias y Bibliografía

La bibliografía de consulta principal fue un documento interno y no publicado, de mi autoría. El mismo establece un diagnóstico sobre la situación actual de las pruebas pre-productivas efectuadas en el área en la que desempeño mis actividades laborales.

También se referenciaron, de manera constante, diversos estándares de la telecomunicación, publicados por los siguientes organismos:

- 3GPP (Third Generation Partnership Project): <http://www.3gpp.org>
- ETSI (European Telecommunications Standards Institute): <http://www.etsi.org>
- IETF (Internet Engineering Task Force): <http://www.ietf.org>

Además, se consultaron diversos artículos en español, inglés y francés, publicados en el siguiente sitio web:

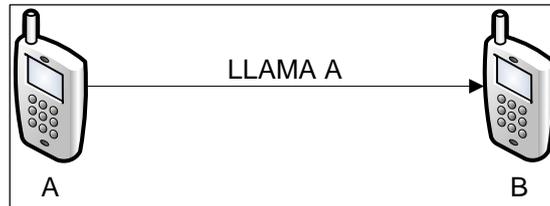
- Wikipedia (The Free Encyclopedia): <https://www.wikipedia.org>

Por último y en menor medida, se hizo referencia a los siguientes libros y publicaciones:

- Redes de Computadoras (4ta Edición - 2003):  
Andrew Tanenbaum (Editorial Pearson Educación)
- Redes de Computadoras, Internet e Interredes (1ra Edición - 1997):  
Douglas Comer (Editorial Prentice-Hall Hispanoamericana)
- Redes Globales de Información con Internet y TCP/IP (3ra Edición - 1996):  
Douglas Comer (Editorial Pearson Educación)
- El Lenguaje Unificado de Modelado (1ra Edición - 2000):  
Grady Booch, James Rumbaugh e Ivar Jacobson (Editorial Addison-Wesley)
- Evaluación y Gestión de Proyectos (2da Edición - 2003):  
José Cuozzo y Gustavo Astini (Editorial IUA)
- Administración (6ta Edición - 2000):  
Stephen Robbins y May Coulter (Editorial Pearson Educación)
- Economía Política (1ra Edición - 2001):  
Daniel Gattás (Editorial Advocatus)

## Anexo 1: Introducción a las Llamadas Móviles

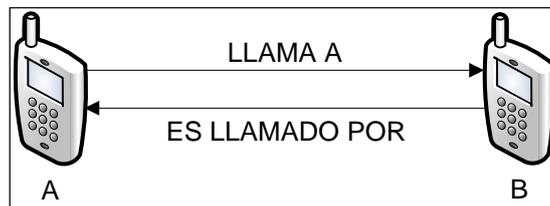
A simple vista, una llamada móvil entre “A” y “B” puede considerarse como algo bastante sencillo y directo:



Lamentablemente esto no es así, ya que existe un sinnúmero de variantes que proporcionan un alto nivel de complejidad. Es por ello que, en el presente documento, se irán detallando cada una de estas variantes, buscando dejar bien claro todo lo que hay que tener en cuenta cuando se realiza una llamada móvil.

### Llamadas Móviles Basadas en su Dirección

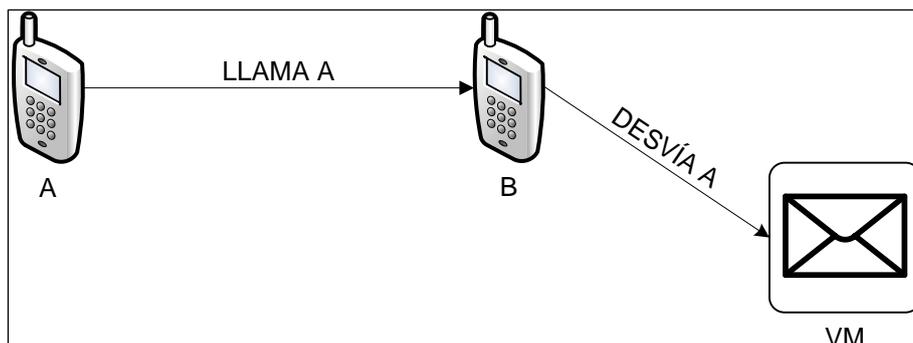
El primer ejemplo de como una simple llamada puede adquirir complejidad, es empezar a verla no solo desde la perspectiva de “A”, sino también desde la de “B”:



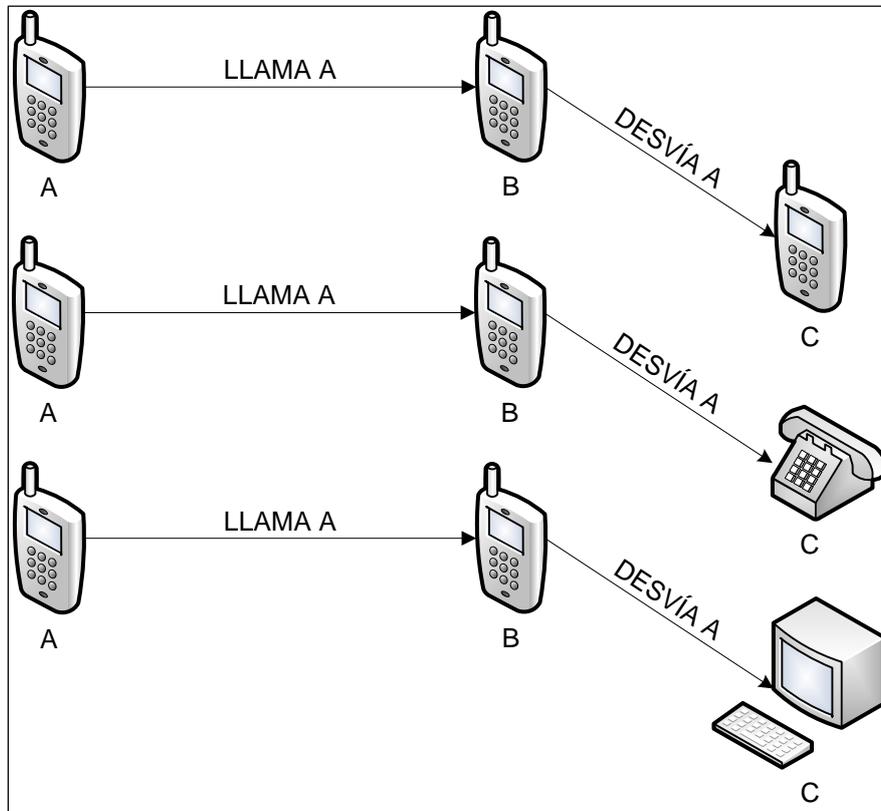
Esto significa que cada llamada tiene dos partes o piernas, pudiendo ser:

- Una llamada móvil saliente (MO), basada en la perspectiva de “A”.
- Una llamada móvil entrante (MT), basada en la perspectiva de “B”.

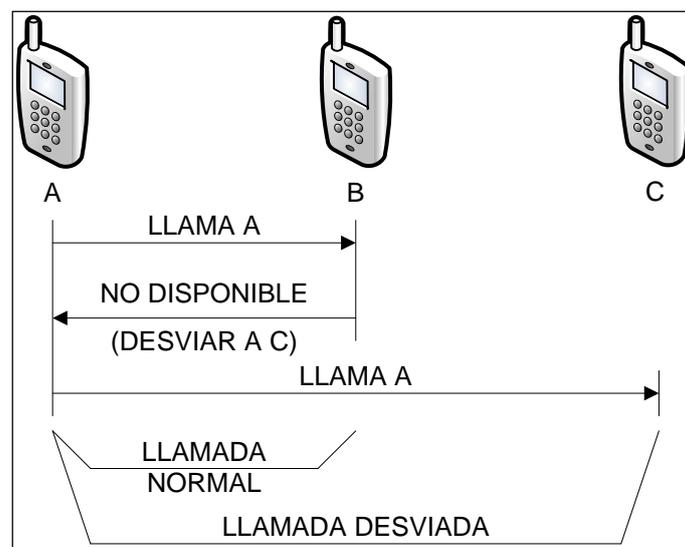
Otro aspecto a considerar es el caso en el que “B” no esté disponible para aceptar la llamada y, probablemente, llegue a su contestador (voice mail):



Esta situación es conocida como desvío de llamada, porque precisamente se la desvía a un tercero; el cual, además de un contestador, puede ser otro teléfono móvil, una línea fija o incluso una computadora (“C”):



Desafortunadamente, el desvío de llamadas no es tan simple como se muestra en el gráfico anterior. De hecho, mientras “B” no está disponible, la llamada original es finalizada y da lugar al establecimiento de una nueva entre “A” y “C”:



Esto significa que cada llamada también puede ser:

- Una llamada móvil normal (MO o MT), donde “A” llama a “B”.
- Una llamada móvil desviada (MF), donde “A” llama a “B” y se lo desvía a “C”.

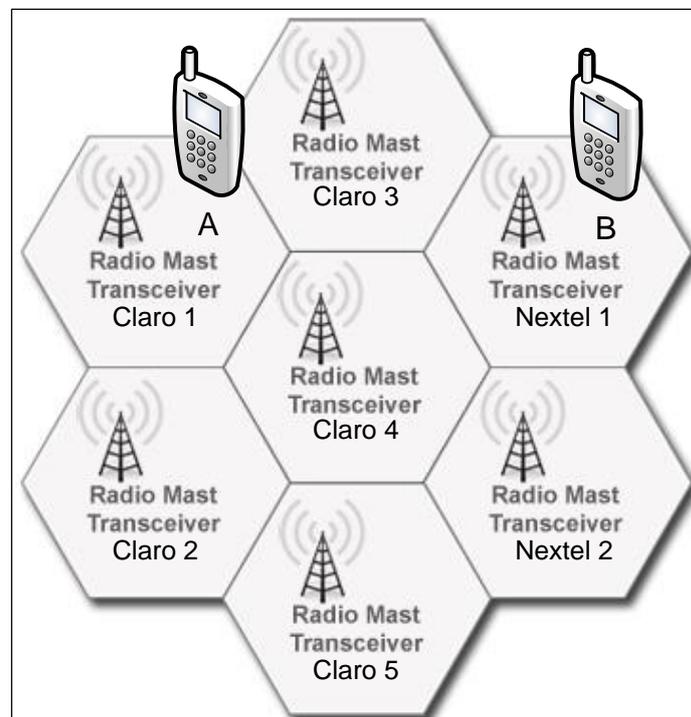
**Basadas en su dirección, las llamadas móviles pueden clasificarse en:**

- **Llamada Móvil Saliente (MO: Mobile Originated).**
- **Llamada Móvil Entrante (MT: Mobile Terminated).**
- **Llamada Móvil Desviada (MF: Mobile Forwarding).**

### ***Llamadas Móviles Basadas en su Ubicación***

Las comunicaciones móviles son logradas gracias a la existencia de una red celular, la cual está compuesta de celdas. Cada una de ellas brinda un área de cobertura, la cual termina donde comienza la de la próxima.

Teniendo esto en cuenta, cada terminal móvil se registra en una celda específica y, cuando deja su área de cobertura, se registra en la siguiente. Esta es la clave de las comunicaciones móviles, ya que de esta forma una terminal puede estar “en cualquier lugar” y ser capaz de hacer o recibir llamadas:



Entonces, para que las comunicaciones móviles funcionen es importante determinar la ubicación de “A” y “B”, la cual está dada por la celda en la que se

registran. Esto hace que sea necesario, para las operadoras móviles, poder identificar cada celda unívocamente, recurriendo para ello a un identificador global llamado CGI (Cell Global Identifier), el cual se compone de:

- Código de País: Mobile Country Code (MCC).
- Código y Tipo de Red: Mobile Network Code + Type (MNC + 0:GSM/1:3GPP).
- Código de Área: Location Area Code (LAC).
- Identificador de Celda: Cell Id (CI)

Cada uno de estos componentes es mandatorio en la conformación del CGI, tal como se puede observar en los siguientes ejemplos:

- CGI 7220310130001587:  
*MCC: 722 (Argentina).*  
*MNC: 0310 (Claro GSM).*  
*LAC: 1300 (Ciudad de Córdoba).*  
*CI: 01587 (Celda 1587).*
- CGI 7220200503300077:  
*MCC: 722 (Argentina).*  
*MNC: 0200 (Nextel GSM).*  
*LAC: 5033 (Ciudad de Buenos Aires).*  
*CI: 00077 (Celda 77).*
- CGI 7360031220100314:  
*MCC: 736 (Bolivia).*  
*MNC: 0031 (Tigo 3GPP).*  
*LAC: 2201 (Ciudad de La Paz).*  
*CI: 00314 (Celda 314).*

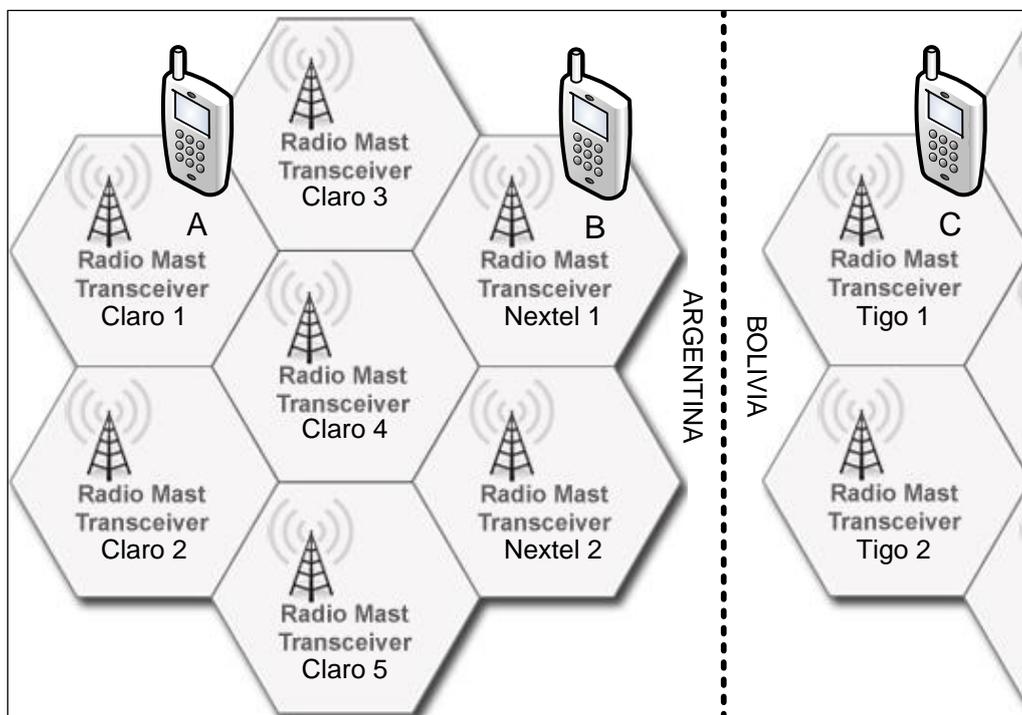
Cada operadora móvil tiene asignados sus propios MCC y MNC y, a su vez, cada uno de sus suscriptores tiene los mismos valores asignados en el IMSI (International Mobile Subscriber Identity) que los identifica.

Durante una llamada, los valores de la operadora (MCC y MNC) son comparados contra los de la celda usada, con el objetivo de saber si la misma le pertenece o no. En caso de que le pertenezca, la llamada es considerada como local (home). Si pertenece a otra operadora del mismo país, la llamada es considerada como roaming

nacional (itinerante). Por último, de pertenecer a cualquier operadora de otro país, la llamada es considerada como roaming internacional.

Por ejemplo, si “A”, “B” y “C” tienen una línea de Claro Argentina:

- Si “A” está registrado en una celda que pertenece a Claro Argentina, se lo considera un número local (home).
- Si “B” está registrado en una celda que pertenece a Nextel Argentina, se lo considera un número en roaming nacional.
- Si “C” está registrado en una celda que pertenece a Tigo Bolivia, se lo considera un número en roaming internacional.



**Basadas en su ubicación, las llamadas móviles pueden clasificarse en:**

- **Locales (Home).**
- **En Roaming (Nacionales o Internacionales).**

### ***Llamadas Móviles Basadas en su Destino***

Una vez determinadas la dirección y la ubicación de la llamada, existe sólo una complicación extra a considerar: El destino. La determinación del mismo sólo adquiere relevancia en la parte saliente de la llamada (MO), cuando esta es analizada desde la

perspectiva del que la realiza (en el caso de las llamadas entrantes, el destino no es una incógnita a develar, ya que se las está analizando desde su perspectiva).

Cada suscriptor en una operadora móvil tiene asociado un número, el cual es usado para realizar llamadas. Por lo tanto, la definición del destino está generalmente ligada al número marcado, el cual está compuesto por:

- Prefijo Internacional (ICP: International Call Prefix): 00, 002, 01.
- Código de País (CC: Country Code): 549, 591, 57.
- Código de Área (AC: Area Code): 351, 11, 2657.
- Número de Suscriptor (SN: Subscriber Number): 6999888, 66111222.

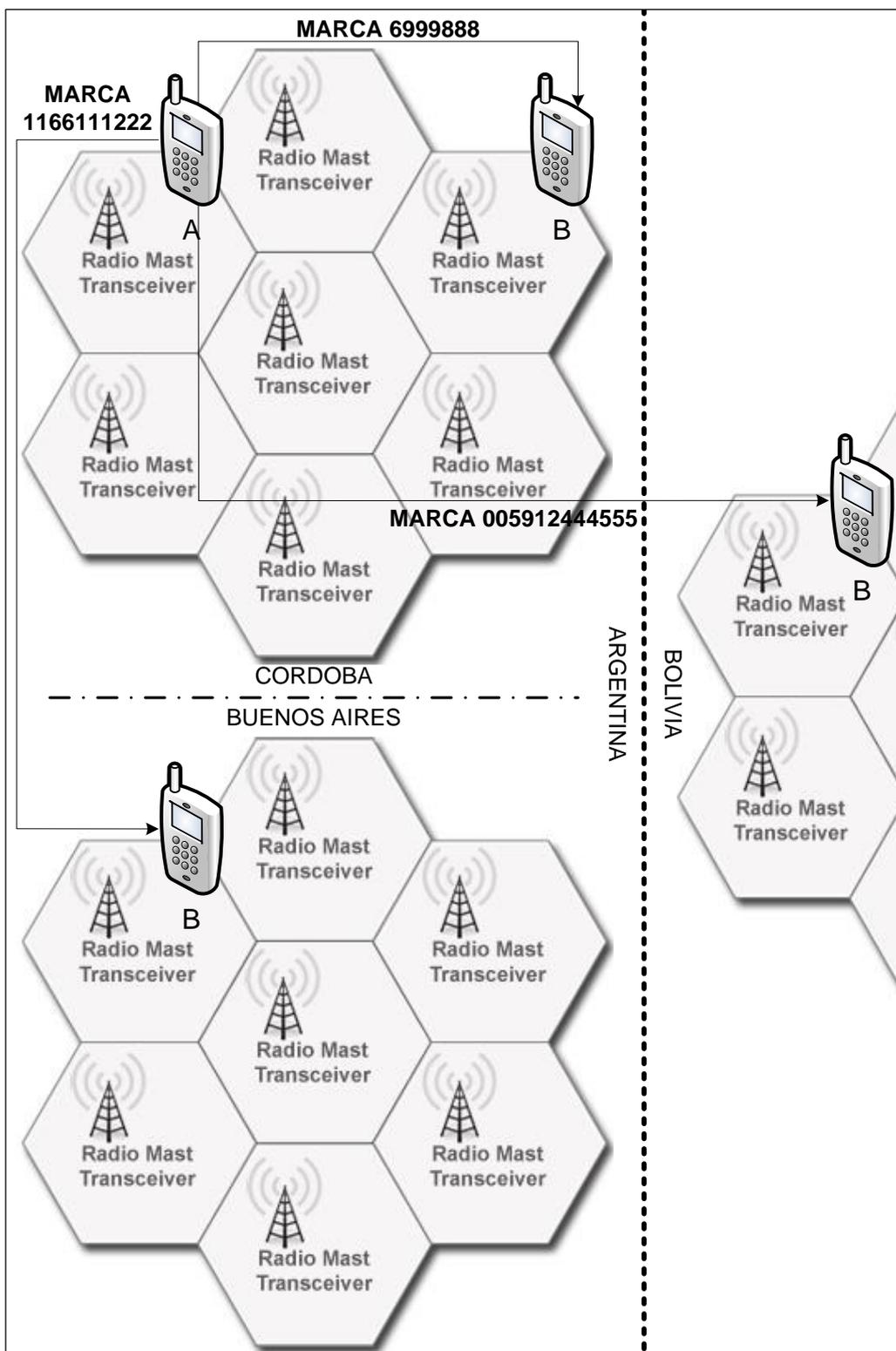
El ICP es un código de salida que debe ser marcado para poder llamar a un destino internacional, el cuál puede ser representado por el símbolo "+". El CC es un código que identifica a un país determinado, el cual es requerido para poder llamar a un destino internacional. El AC es un código que identifica a un área determinada de un país, el cual es requerido para poder llamar a un destino de otra región.

Asumiendo que "A" tiene un número de teléfono y "B" tiene tres:

- A1: 5493516777666 (CC: 549 – Argentina / AC: 351 – Ciudad de Córdoba).
- B1: 5493516999888 (CC: 549 – Argentina / AC: 351 – Ciudad de Córdoba).
- B2: 5491166111222 (CC: 549 – Argentina / AC: 11 – Ciudad de Buenos Aires).
- B3: 5912444555 (CC: 591 – Bolivia / AC: 2 – Ciudad de La Paz).

"A" tendrá distintas formas de marcar cada uno de los números de "B":

- Para 5493516999888, puede elegir marcar:  
*6999888: Porque pertenece al mismo CC y AC.*  
*3516999888: Porque pertenece al mismo CC, pero desea especificar el AC.*  
*005493516999888: Porque desea usar el formato internacional.*
- Para 5491166111222, puede elegir marcar:  
*1166111222: Porque pertenece al mismo CC.*  
*005491166111222: Porque desea usar el formato internacional.*  
*(No puede marcar 66111222, porque pertenece a un AC diferente).*
- Para 5912444555, sólo puede marcar:  
*005912444555: Porque tiene que usar el formato internacional.*  
*(No puede marcar 2444555, porque pertenece a un CC diferente).*



Como se menciona previamente, el destino de una llamada está generalmente ligado al número marcado; sin embargo, “generalmente” no significa “siempre”. En este caso, esto es así porque la Red Inteligente realiza previamente un análisis y formateo del número marcado, basado en la ubicación de quien realiza la marcación.

La siguiente tabla refleja el análisis y el formateo realizado sobre cada tipo de llamada, emitida por un número que pertenece a una operadora móvil argentina (por ejemplo, con el IMSI 722031999888777):

Ubicación	Marcado	Formateo RI	Estado
MCC: 722 - Argentina	6999888	<b>351</b> 6999888	Llamada local establecida
MCC: 722 - Argentina	3516999888	<b>351</b> 6999888	Llamada local establecida
MCC: 722 - Argentina	005493516999888	<b>351</b> 6999888	Llamada local establecida
MCC: 722 - Argentina	<b>00</b> 5912444555	5912444555	Llamada internacional establecida
MCC: 722 - Argentina	5493516999888	-	Llamada fallida: ICP no provisto
MCC: 722 - Argentina	5912444555	-	Llamada fallida: ICP no provisto
MCC: 736 - Bolivia	6999888	<b>549351</b> 6999888	Llamada nacional establecida
MCC: 736 - Bolivia	3516999888	<b>549351</b> 6999888	Llamada nacional establecida
MCC: 736 - Bolivia	005493516999888	<b>549351</b> 6999888	Llamada nacional establecida
MCC: 736 - Bolivia	<b>00</b> 5912444555	5912444555	Llamada internacional establecida
MCC: 736 - Bolivia	5493516999888	-	Llamada fallida: ICP no provisto
MCC: 736 - Bolivia	5912444555	-	Llamada fallida: ICP no provisto

Si "A" quiere realizar una llamada internacional, deberá utilizar el código de salida del país (ICP) o, de lo contrario, no podrá efectuarla. Además, si "A" se encuentra en una celda local (home), el destino será formateado para incluir el código de área (AC), pero no el código de país (CC), ya que se trata de una llamada local. Finalmente, si "A" se encuentra en roaming (itinerante), el destino será formateado para incluir el código de área (AC) y el código de país (CC), porque se trata de una llamada nacional.

El último aspecto a considerar está vinculado con la portabilidad numérica. Dicha portabilidad es la capacidad de un cliente de mantener su número de línea cuando decide cambiar de operadora móvil.

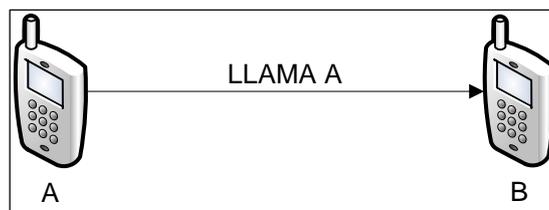
La portabilidad numérica hace que sea necesario distinguir los destinos que están vinculados a clientes de la misma operadora (on-net) de los que no lo están (off-net). Para lograr esto, la Red Inteligente se encarga de enviar, al momento de establecer la llamada, un número de ruteo (routing number) que se asocia a la operadora a la que pertenece el destino; excepto en el caso de las llamadas internacionales. Algunos ejemplos de este RN enviado por la Red Inteligente son: 555 para Claro, 666 para Personal, 777 para Movistar y 888 para Nextel.

**Basadas en su destino, las llamadas móviles pueden clasificarse en:**

- **Local (Home + Destino con el mismo CC).**  
*On-net (Destino perteneciente a la misma operadora móvil).*  
*Off-net (Destino perteneciente a otra operadora móvil).*
- **Nacional (Roaming + Destino con el mismo CC).**  
*On-net (Destino perteneciente a la misma operadora móvil).*  
*Off-net (Destino perteneciente a otra operadora móvil).*
- **Internacional (Destino con otro CC).**

### Conclusión

A simple vista, una llamada móvil entre “A” y “B” puede considerarse como algo bastante sencillo y directo:



Lamentablemente esto no es así, ya que existe un sinnúmero de variantes que proporcionan un alto nivel de complejidad. Estas se reflejan en la siguiente tabla:

Dirección	Ubicación	Destino
MO	Home	Local On-net
MO	Home	Local Off-net
MO	Home	Internacional
MO	Roaming	Nacional On-net
MO	Roaming	Nacional Off-net
MO	Roaming	Internacional
MT	Home	No Aplicable
MT	Roaming	No Aplicable
MF	No Aplicable	Nacional On-net
MT	No Aplicable	Nacional Off-net
MT	No Aplicable	Internacional

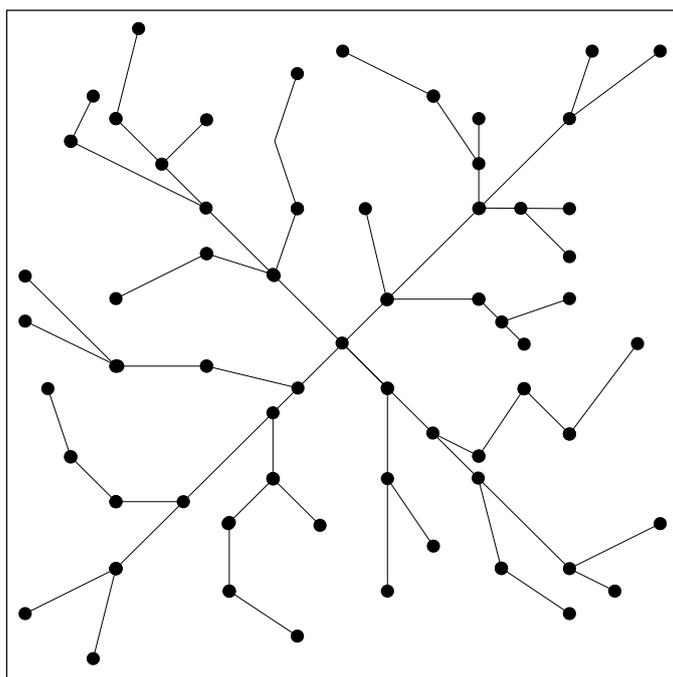
## Anexo 2: Llamadas Móviles

Existe una expresión que se menciona asiduamente en el submundo de las comunicaciones móviles: Red Inteligente. En esta sección se detallan sus principales características, explicando todo lo que existe entre “A” y “B” a la hora de establecer una llamada móvil.

### ***Red Inteligente (IN)***

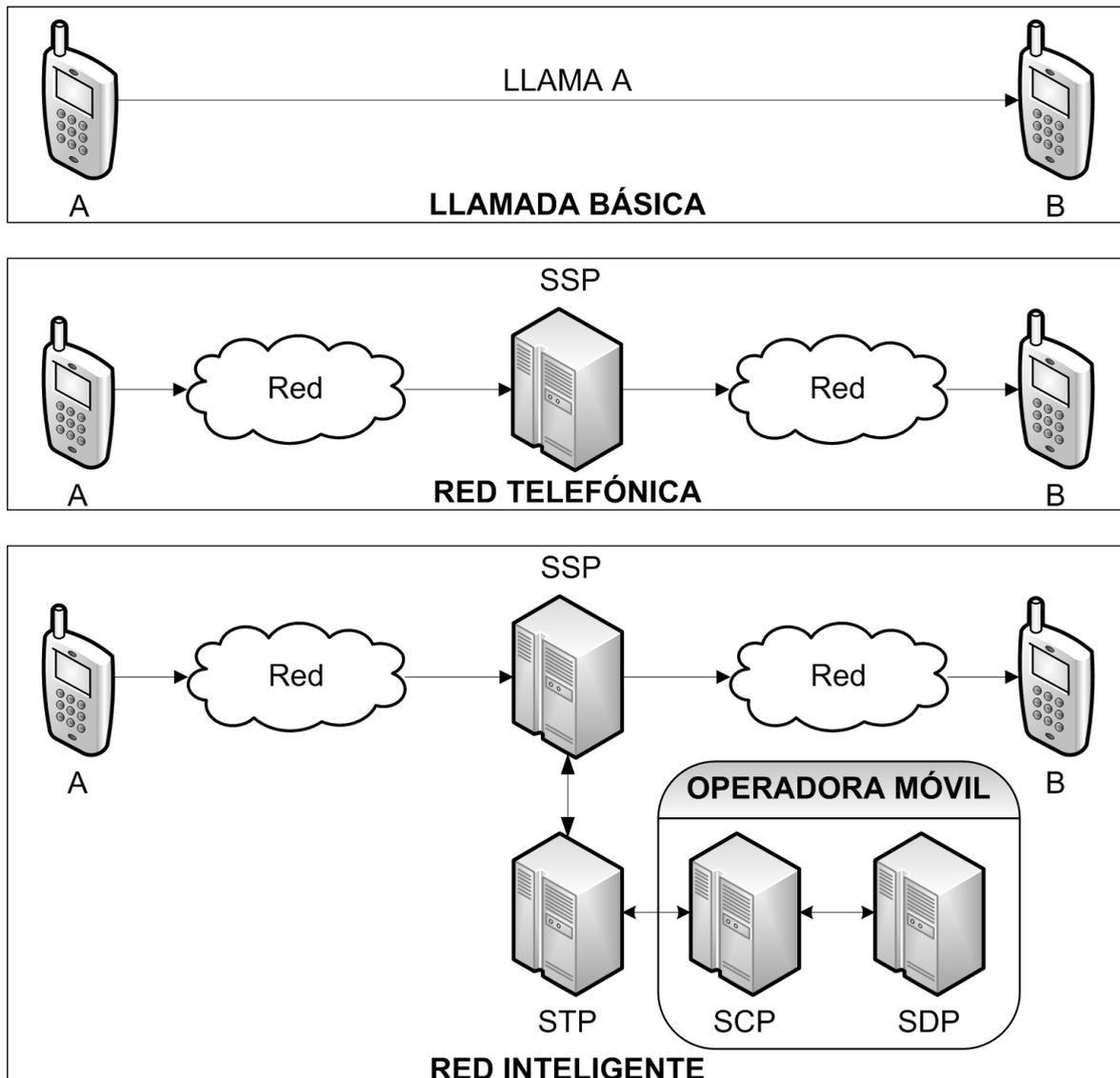
La palabra red hace referencia a cualquier grupo o sistema interconectado. Esta definición es bastante amplia, ya que en ella se pueden alinear: Redes de televisión, redes económicas, redes sociales, redes neuronales, redes de computadoras, etc.

Sin embargo, sólo son relevantes para el presente documento las redes de telecomunicación. Estas se definen como una colección de nodos y enlaces, conectados de forma tal que permiten la telecomunicación (comunicación a distancia) entre terminales:



Básicamente, los enlaces de transmisión conectan los nodos. Estos últimos pasan la señal por los enlaces adecuados hasta llegar al destino deseado. Cualquier nodo intermedio actúa como un switch (conmutador), enrutando la señal hacia el nodo de destino. Algunos ejemplos de redes de telecomunicación son: Redes de computadoras, Internet y redes telefónicas.

El concepto de Red Inteligente está basado en el de las redes telefónicas, pero con una gran diferencia: La lógica del servicio está separada de las funcionalidades de conmutación (switching), permitiendo que se puedan agregar o modificar los servicios ofrecidos sin necesidad de rediseñar dicha funcionalidad. En otras palabras, la Red Inteligente utiliza centros de conmutación como las redes telefónicas tradicionales, pero separa la configuración de los servicios de valor agregado y los deja del lado de las operadoras móviles:



La Red Inteligente está compuesta por los siguientes elementos:

- SSP (Service Switching Point): Responsable de interceptar la señal de la llamada proveniente de la red y enrutarla hacia el SCP de la operadora móvil, para así conocer cómo se debe proceder. Una vez que reciben las instrucciones, conectan (o rechazan) la llamada con el destino.

- STP (Signal Transfer Point): Transporta las distintas consultas y respuestas enviadas entre el SSP y el SCP. A veces, alberga la lógica que define los RN utilizados en la portabilidad numérica.
- SCP (Service Control Point): Plataforma separada de la red de conmutación, la cual contiene la lógica del servicio ofrecido por la operadora móvil.
- SDP (Service Data Point): Proporciona los datos persistentes en los que se apoya la lógica de servicio; es decir, las bases de datos.

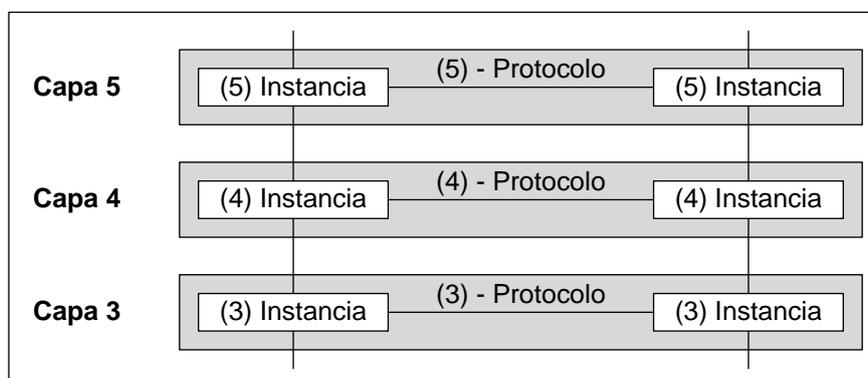
**La Red Inteligente es todo aquello que se encuentra entre “A” y “B”, permitiendo que estos se comuniquen y manteniendo toda la configuración para ello del lado de la operadora móvil.**

**A pesar de que tiene muchos componentes que no han sido mencionados, los más importantes son el SSP y el SCP. Esto se debe a que su interacción es vital para definir si la llamada puede llevarse a cabo o no.**

### **Sistema de Señalización Numero 7 (SS7)**

Una de las claves de cualquier red de telecomunicación es el diálogo entre los componentes físicos. Dicho diálogo se lleva a cabo utilizando un estándar, el cual está basado en un sistema particionado en varias capas de abstracción. Cada capa es una colección de funciones similares que provee ciertos servicios a la capa superior y, a su vez, recibe otros de la inferior.

El beneficio de contar con capas de abstracción es que se pueden ocultar los detalles de la implementación de un grupo determinado de funcionalidades, reduciendo la complejidad en cualquier solución comunicacional asociada:

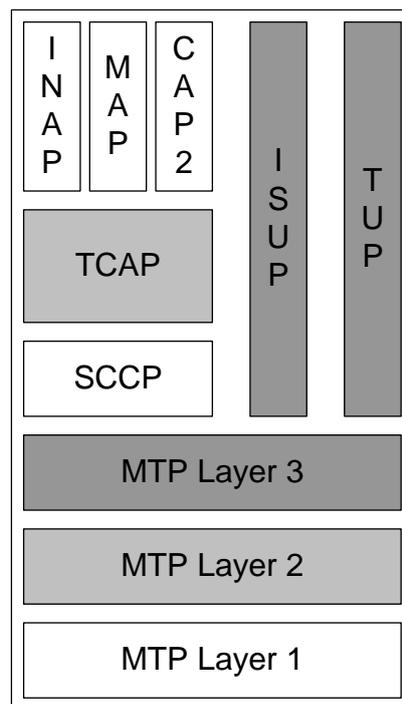


La idea es que todos los componentes de una red que deseen mantener un dialogo con otros tengan las mismas capas de abstracción y, por cada una de ellas, utilicen los mismos protocolos (o lenguajes) para intercambiar datos.

El modelo OSI es el estándar de referencia para definir las capas de abstracción en las redes de telecomunicaciones:



En el caso de la Red Inteligente, el estándar de referencia se llama Sistema de Señalización Número 7 (SS7), el cual se basa en el modelo OSI de abstracción:



Las capas de abstracción más importantes en SS7 son:

- MTP 1 (Message Transfer Part 1): Física.
- MTP 2 (Message Transfer Part 2): Enlace de Datos.
- MTP 3 (Message Transfer Part 3): Red.
- SCCP (Signalling Connection Control Part): Transporte.
- TCAP (Transaction Capabilities Application Part): Sesión.
- INAP (Intelligent Network Application Protocol): Presentación y Aplicación.

Existe otro protocolo de capa de aplicación, fruto de la extensión de INAP: CAP. Significa CAMEL Application Part, donde CAMEL es por Customized Application for Mobile Enhanced Logic. Con el tiempo, este protocolo ha evolucionado en diferentes fases, siendo la segunda de ellas la más utilizada para las llamadas móviles (CAP2).

**La interacción entre los distintos componentes de una Red Inteligente es llevada a cabo utilizando una pila de protocolos, cada uno de ellos perteneciente a una determinada capa de abstracción.**

**Esta pila está basada en el estándar SS7.**

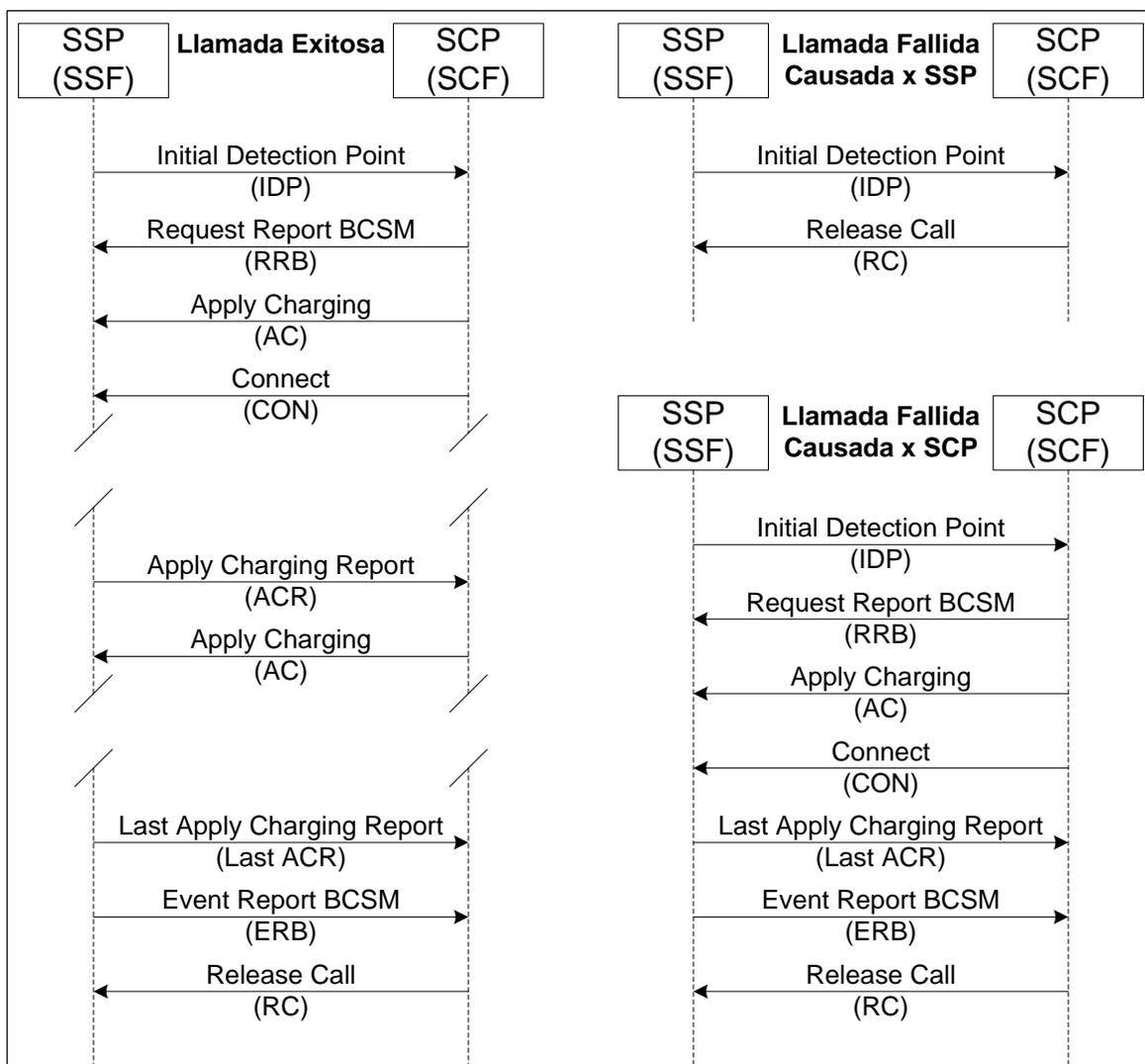
### ***CAMEL Fase 2 (CAP2) - Flujos Principales***

Cada protocolo vinculado al estándar SS7 tiene su propia complejidad, teniendo en cuenta que mientras más baja es la capa a la que pertenece, más cerca está de los "0s y 1s". Para el caso particular de este documento, son de interés los protocolos de capa de aplicación, más precisamente el protocolo CAP2.

Este protocolo es el que se utiliza en las llamadas móviles para intercambiar información entre los dos componentes principales de una Red Inteligente: SSP (switch) y SCP (lógica del servicio). Gracias a este intercambio de mensajes, una llamada móvil puede establecerse y tarifarse en línea (o rechazarse).

A continuación se encuentran algunos ejemplos de los mensajes utilizados en:

- Una llamada exitosa.
- Una llamada fallida porque el SSP proporcione datos incorrectos al SCP.
- Una llamada fallida porque el SCP proporcione datos incorrectos al SSP.



*Initial Detection Point (IDP):* Es el primer mensaje enviado por el SSP al SCP, sin el cual no puede existir una llamada móvil. El mismo posee toda la información proveniente de la Red Inteligente, necesaria para que la operadora móvil defina como proceder: Conectar o rechazar la llamada. Entre la información provista, se encuentra: El número de “A”, el número de “B”, la celda donde se encuentra “A”, las capacidades de la red, la fecha de la llamada, etc.

*Request Report BCSM (RRB):* Es un mensaje enviado por el SCP para indicarle al SSP cuáles son los eventos, ocurridos durante una llamada, que han de serle reportados. Es decir, el SSP se encarga de monitorear los estados por los que pasa una llamada y los eventos que desencadenan cambios; cuando uno de ellos ocurre, de haber sido solicitado por el SCP, será reportado para que éste último actúe en consecuencia. Este mensaje sólo se envía cuando la operadora considera que están dadas todas las condiciones para conectar la llamada, por lo que la primera vez va

acompañado de otros dos mensajes: Apply Charging (AC) y Connect (CON). Algunos de los eventos que se pueden monitorear son: Fallas en el enrutamiento de la llamada, destino ocupado, falta de respuesta en el destino, desconexión de la llamada, etc.

*Apply Charging (AC)*: Es un mensaje enviado por el SCP para indicarle al SSP a quien se le cobra la llamada, junto con el tiempo que debe transcurrir hasta que este último se reporte con información sobre el estado de la misma. Por lo tanto, este mensaje trabaja en conjunto con el Apply Charging Report (ACR), conformando un tándem que se repite hasta el final de la llamada: El AC le indica al SSP que informe el estado de la llamada dentro de "X" segundos y, transcurrido ese tiempo, el SSP realiza esto último mediante el envío de un ACR.

*Connect (CON)*: Es un mensaje enviado por el SCP con el objeto de indicarle al SSP con que destino debe conectar la llamada, junto con la información necesaria para llevar esto a cabo. Es enviado en caso de que alguno de los datos que el SSP informó en el IDP deba ser cambiado, por ejemplo: El número de "B" ha sido reformateado, la llamada ha de ser desviada a un tercero ("C"), etc. En caso de que la llamada se pueda conectar con la misma información que vino en el IDP, se puede usar en su lugar el mensaje *Continue (CUE)*.

*Apply Charging Report (ACR)*: Es un mensaje enviado por el SSP, informando al SCP si la llamada está activa y el tiempo que ha transcurrido desde su comienzo. Por lo tanto, trabaja en conjunto con el Apply Charging (AC), como se mencionó previamente. Además, este mensaje también sirve para indicar que la llamada ha finalizado, informando que la misma ha dejado de estar activa.

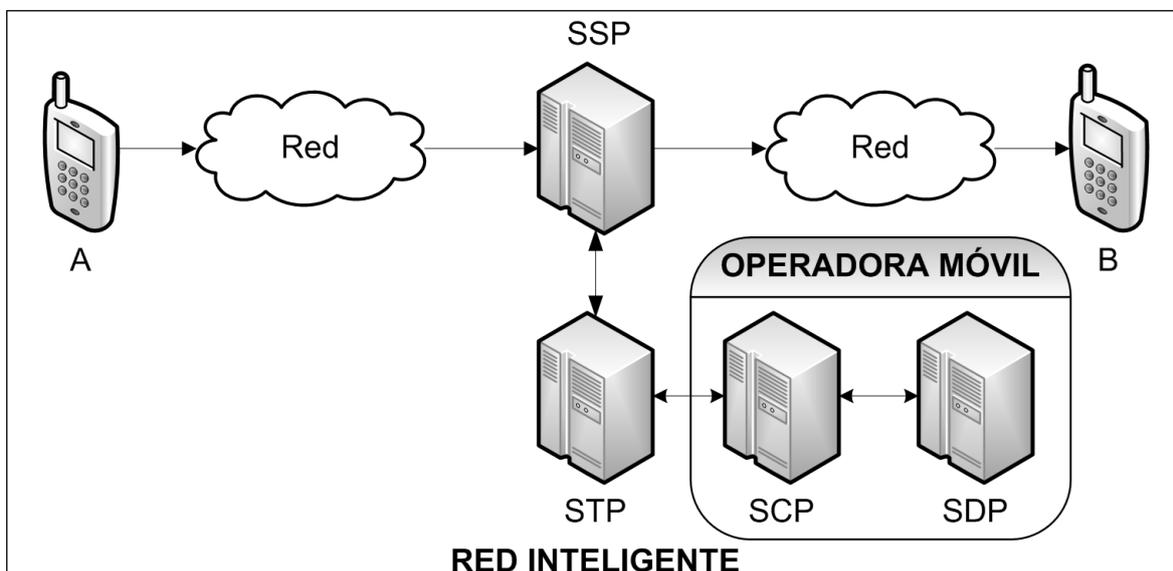
*Event Report BCSM (ERB)*: Es un mensaje enviado por el SSP, informando la ocurrencia de alguno de los eventos que el SCP pidió monitorear (en el RRB), para que este último actúe en consecuencia.

*Release Call (RC)*: Es un mensaje enviado por el SCP para indicarle al SSP que la llamada ha terminado, por lo que todas sus conexiones en la Red Inteligente han de ser liberadas. Es utilizado tanto para finales exitosos como fallidos, tales como la falta de fondos para hacer la llamada, errores en la tarifación, información incorrecta en el IDP, etc. Es por esto último que siempre indica la causa del fin de la llamada.

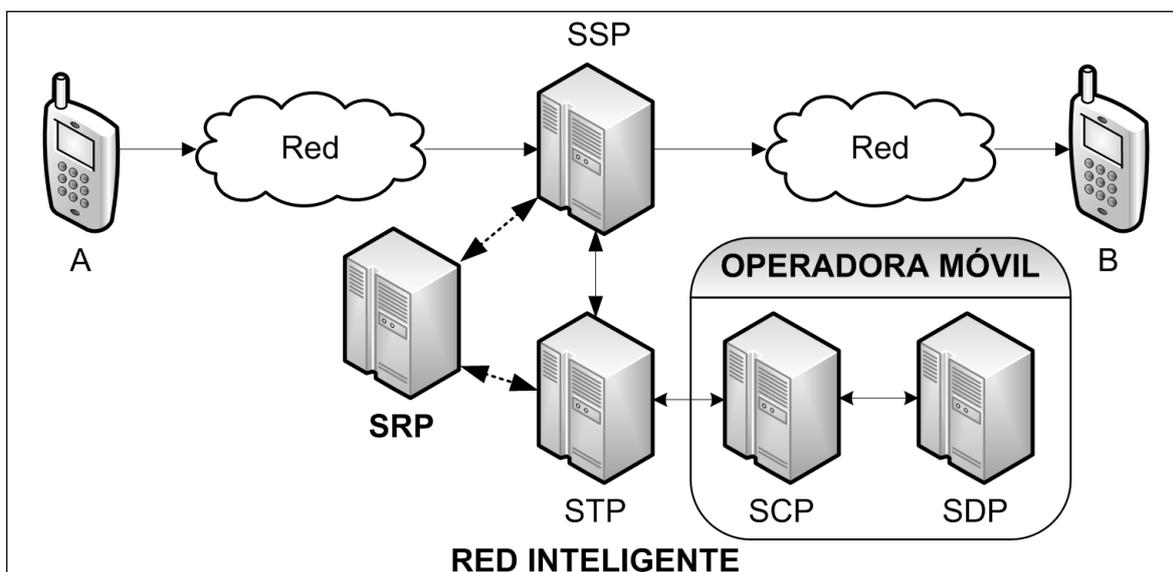
Para más detalles sobre los mensajes intercambiados en el protocolo CAP2, referirse al Anexo 3: Mensajes CAP2.

### CAMEL Fase 2 (CAP2) - Flujos Alternativos

Hasta el momento se han mencionado los mensajes intercambiados entre el SSP y el SCP, principales actores en la mayoría de llamadas móviles:

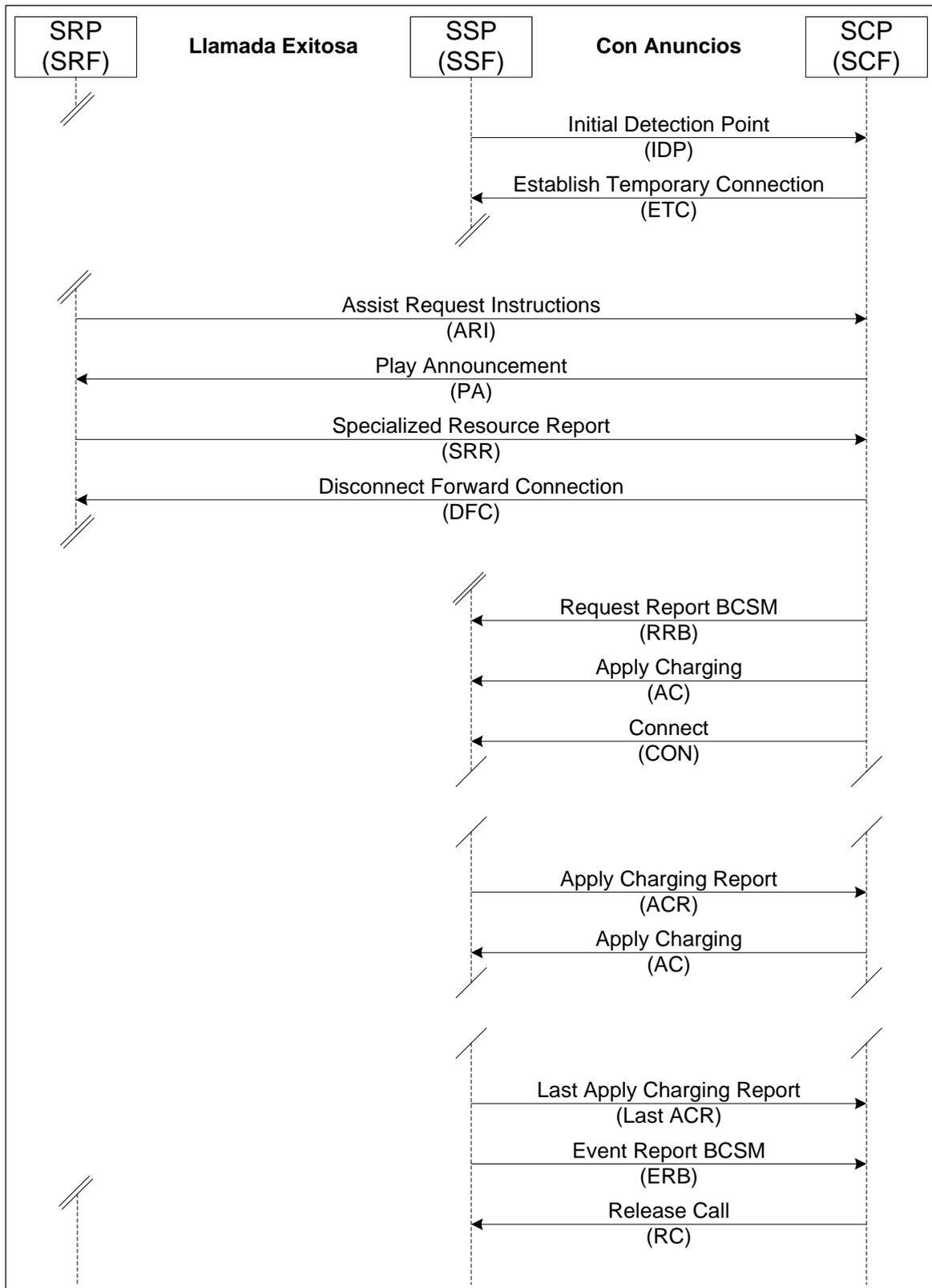


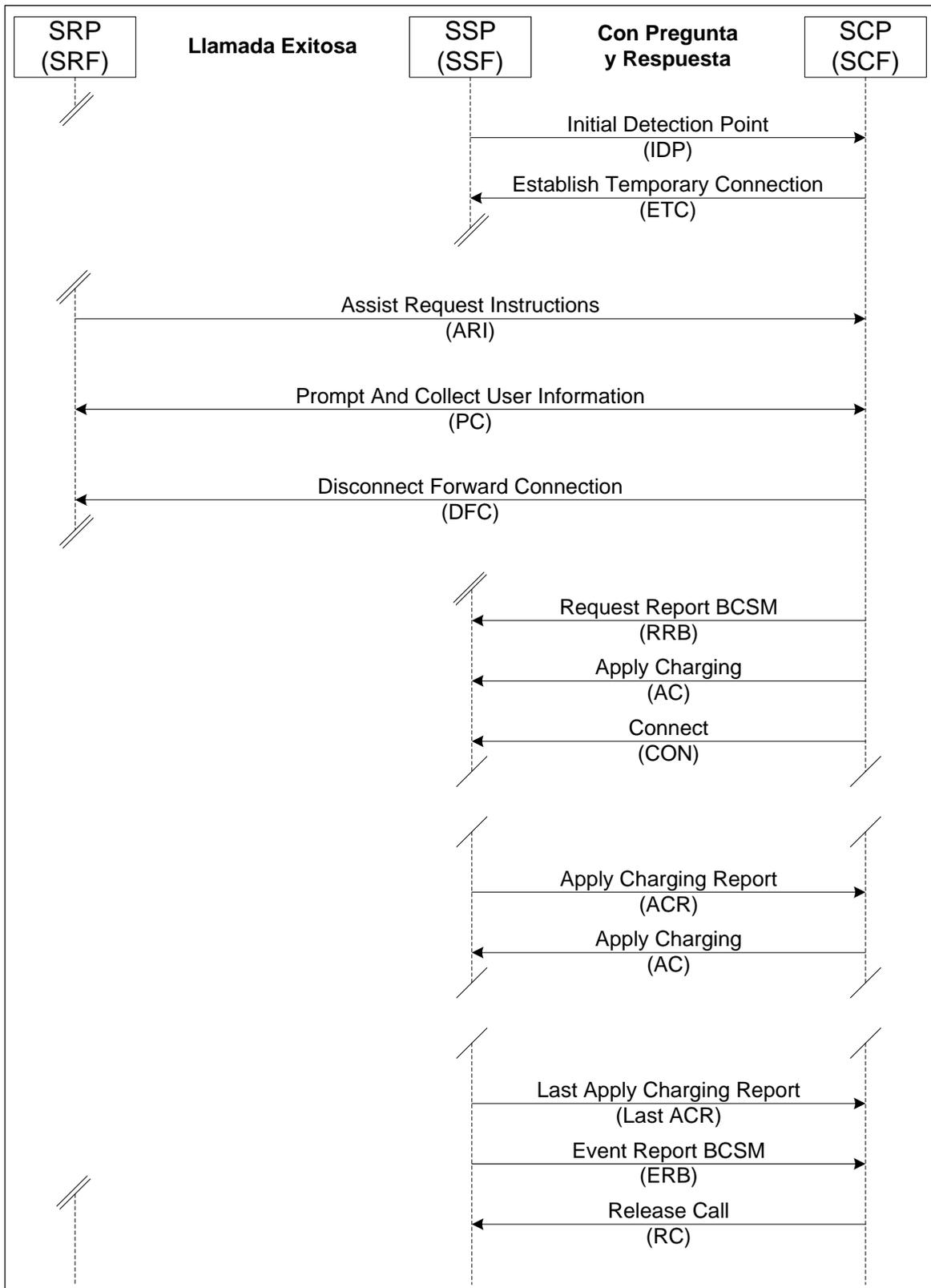
Sin embargo, existe una serie de flujos alternativos que requieren la presencia de un periférico inteligente, anexo al SSP. Dicho periférico es conocido como Specialized Resource Point (SRP) y cumple funciones complementarias, como reproducir anuncios de voz o coleccionar tonos DTMF ingresados por los usuarios:

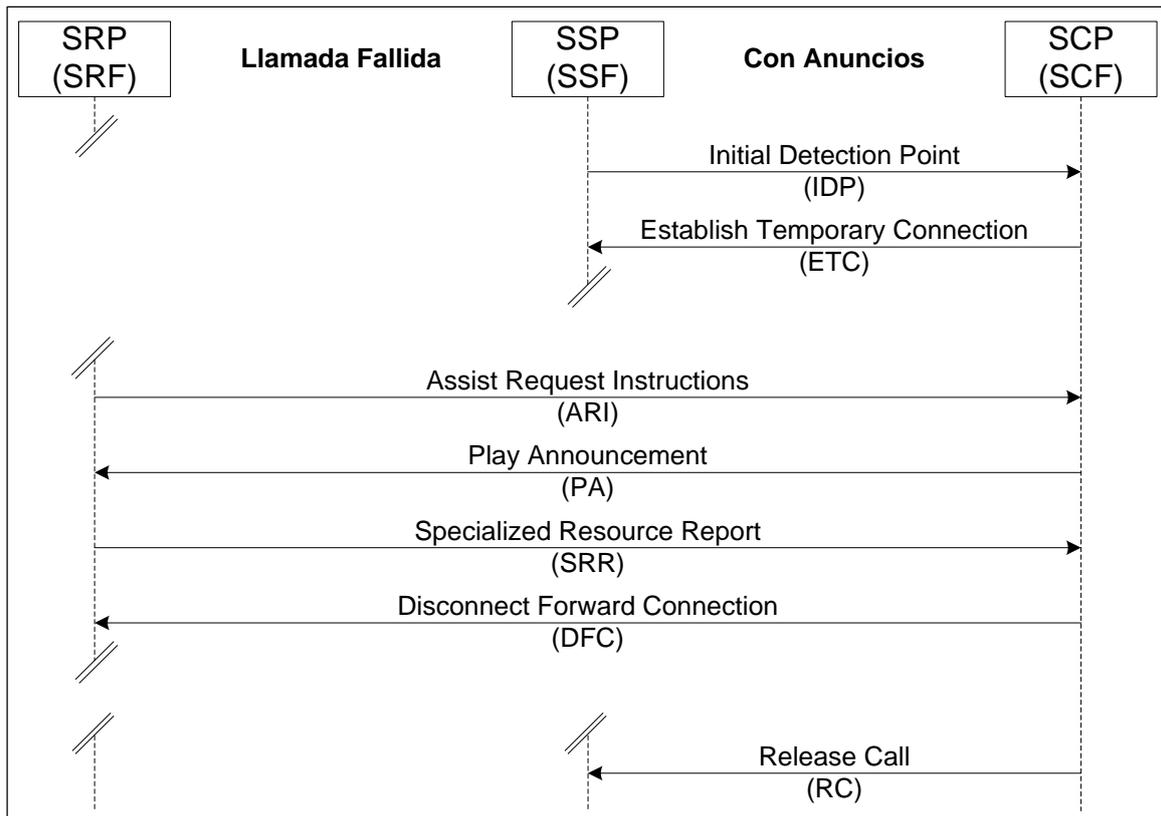


A continuación se encuentran algunos ejemplos de los mensajes utilizados en:

- Una llamada exitosa con anuncios.
- Una llamada exitosa con pregunta y respuesta (prompt and collect).
- Una llamada fallida con anuncios.







*Establish Temporary Connection (ETC)*: Es un mensaje enviado por el SCP, para solicitarle al SSP la creación de una conexión temporal con el SRP. Dicha conexión será utilizada para reproducir un anuncio destinado al usuario o para realizarle una pregunta y esperar su respuesta.

*Assist Request Instructions (ARI)*: Es un mensaje enviado por el SRP al SCP, tras haber sido contactado por el SSP. En él se detallan las capacidades del periférico inteligente, en base a los recursos que tiene a su disposición. En caso de que no exista un SRP externo, se puede utilizar el mensaje *Connect to Resource (CTR)* en lugar del tándem ETC/ARI.

*Play Announcement (PA)*: Es un mensaje enviado por el SCP al SRP, con el objeto de poder interactuar con el usuario final mediante la reproducción de un anuncio. Todas las características para realizar la reproducción (mensaje, duración y repeticiones) son detalladas en este mensaje.

*Specialized Resource Report (SRR)*: Es un mensaje enviado por el SRP, para indicarle al SCP que al usuario se le ha reproducido el anuncio solicitado.

*Prompt and Collect User Information (PC)*: Es un mensaje enviado por el SCP al SRP, con el objeto de poder interactuar con el usuario final y tomar decisiones en consecuencia. Establece un canal de comunicación en ambos sentidos, donde al usuario se le reproducen ciertos anuncios (o preguntas) y este responde, mediante el uso de tonos DTMF.

*Disconnect Forward Connection (DFC)*: Es un mensaje enviado por el SCP, para indicarle al SRP y al SSP que la conexión temporal que se estableció ha de ser terminada.

Para más detalles sobre los mensajes intercambiados en el protocolo CAP2, referirse al Anexo 3: Mensajes CAP2.

## Anexo 3: Mensajes CAP2

### *Initial Detection Point (IDP)*

Es el primer mensaje enviado por el SSP al SCP, sin el cual no puede existir una llamada móvil. El mismo posee toda la información proveniente de la Red Inteligente, necesaria para que la operadora móvil defina como proceder: Conectar o rechazar la llamada. Puede contener los siguientes argumentos:

*Invoke Id:* Define el número de secuencia de la operación invocada.

*Operation Code:* Define el código asociado a la operación (IDP - Initial DP).

*Service Key:* Define la lógica de servicio que el SCP debe utilizar. Es obligatorio para llamadas MO, MT y MF. Normalmente está configurado en 1.

*Called Party Number:* Define quien recibe la llamada y sus características. No es aplicable a llamadas MO (utilizan el *Called Party BCD Number*) y es obligatorio para llamadas MT y MF. Incluye estos datos:

- Nature of Address Indicator:
  - 1 - Subscriber Number.
  - 2 - Unknown Number.
  - 3 - National Number.
  - 4 - International Number.
  - 5 - Network-specific Number.
- Internal Network Number Indicator:
  - 0 - Routing to Internal Network Number Allowed.
  - 1 - Routing to Internal Network Number Not Allowed.
- Numbering Plan Indicator:
  - 1 - ISDN Numbering Plan.
  - 3 - Data Numbering Plan.
  - 4 - Telex Numbering Plan.
- Address Signals: Número de quien recibe la llamada.

*Calling Party Number:* Define quien hace la llamada y sus características. Es obligatorio para llamadas MO y condicional (incluido de estar disponible) para llamadas MT y MF. Incluye estos datos:

- Nature of Address Indicator:
  - 1 - Subscriber Number.
  - 2 - Unknown Number.
  - 3 - National Number.
  - 4 - International Number.
  - 5 - Network-specific Number.
- Calling Party Number Incomplete Indicator:
  - 0 - Complete.
  - 1 - Incomplete.
- Numbering Plan Indicator:
  - 1 - ISDN Numbering Plan.
  - 3 - Data Numbering Plan.
  - 4 - Telex Numbering Plan.
- Address Presentation Restricted Indicator:
  - 0 - Presentation Allowed.
  - 1 - Presentation Restricted.
  - 2 - Address Not Available.
- Screening Indicator:
  - 0 - User Provided - Not Verified.
  - 1 - User Provided - Verified and Passed.
  - 2 - User Provided - Verified and Failed.
  - 3 - Network Provided.
- Address Signals: Número de quien hace la llamada.

*Calling Party's Category:* Define la categoría de quien hace la llamada. Es obligatorio para llamadas MO y condicional (incluido de estar disponible) para llamadas MT y MF. Incluye estos datos:

- Calling Party's Category:
    - 01 - French Operator.
    - 02 - English Operator.
    - 03 - German Operator.
    - 04 - Russian Operator.
-

- 05 - Spanish Operator.
- 10 - Ordinary Calling Subscriber.
- 11 - Calling Subscriber with Priority.
- 12 - Data Call (Voice Band Data).
- 13 - Test Call.
- 15 - Payphone.

*IPSSP Capabilities:* Define las características del periférico inteligente (SRP) anexo al SSP. Es condicional (incluido de estar disponible) para llamadas MO, MT y MF.

Incluye estos datos:

- IP Routing Address:
  - 0 - Not Supported.
  - 1 - Supported.
- Voice Back:
  - 0 - Not Supported.
  - 1 - Supported.
- Voice Information Via Speech Recognition:
  - 0 - Not Supported.
  - 1 - Supported.
- Voice Information Via Voice Recognition:
  - 0 - Not Supported.
  - 1 - Supported.
- Generation of Voice Announcements from Text:
  - 0 - Not Supported.
  - 1 - Supported.

*Location Number:* Define la ubicación del equipo de conmutación (MSC o VLR) que maneja la llamada. Es obligatorio para llamadas MO, condicional (incluido de estar disponible) para llamadas MT y no es aplicable para llamadas MF. Incluye estos datos:

- Nature of Address Indicator:
  - 1 - Subscriber Number.
  - 2 - Unknown Number.
  - 3 - National Number.

- 4 - International Number.
- 5 - Network-specific Number.
- Internal Network Number Indicator:
  - 0 - Routing to Internal Network Number Allowed.
  - 1 - Routing to Internal Network Number Not Allowed.
- Numbering Plan Indicator:
  - 1 - ISDN Numbering Plan.
  - 3 - Data Numbering Plan.
  - 4 - Telex Numbering Plan.
- Address Presentation Restricted Indicator:
  - 0 - Presentation Allowed.
  - 1 - Presentation Restricted.
  - 2 - Address Not Available.
- Screening Indicator:
  - 0 - User Provided - Not Verified.
  - 1 - User Provided - Verified and Passed.
  - 2 - User Provided - Verified and Failed.
  - 3 - Network Provided.
- Address Signals: Número de ubicación asociado por la red al equipo en cuestión.

*Original Called Party Id:* Define quién recibió originalmente la llamada y sus características, en un escenario de desvío de llamada (si "A" llama a "B" y este desvía a "C", entonces el IDP de la llamada desviada tendrá a "B" en este argumento). No es aplicable para llamadas MO y MT y es condicional (incluido de estar disponible) para llamadas MF. Incluye estos datos:

- Nature of Address Indicator:
  - 1 - Subscriber Number.
  - 2 - Unknown Number.
  - 3 - National Number.
  - 4 - International Number.
  - 5 - Network-specific Number.
- Numbering Plan Indicator:

- 1 - ISDN Numbering Plan.
- 3 - Data Numbering Plan.
- 4 - Telex Numbering Plan.
- Address Presentation Restricted Indicator:
  - 0 - Presentation Allowed.
  - 1 - Presentation Restricted.
  - 2 - Address Not Available.
- Address Signals: Número de quien recibió originalmente la llamada.

*Higher Layer Compatibility:* Define los medios para validar la compatibilidad con los diferentes servicios ofrecidos por una operadora móvil. Es condicional (incluido de estar disponible) para llamadas MO, MT y MF. Incluye estos datos:

- Coding Standard:
  - 0 - CCITT Standard.
  - 1 - ISO/IEC Standard.
  - 2 - National Standard.
  - 3 - Standard Defined for the Network.
- Interpretation:
  - 4 - First High Layer Characteristics Identification.
- Presentation Protocol Profile:
  - 1 - High Layer Protocol Profile.
- High Layer Characteristics Identification:
  - 01 - Telephony.
  - 04 - Facsimile Group 2 and 3.
  - 33 - Facsimile Group 4.
  - 36 - Teletex Service - Basic and Mixed Operation Mode.
  - 40 - Teletex Service - Basic and Processable Operation Mode.
  - 49 - Teletex Service - Basic Operation Mode.
  - 50 - Syntax Based Videotex.
  - 51 - International Videotex.
  - 53 - Telex Service.
  - 56 - Message Handling Systems.

65 - OSI Application.

96 - Audio Visual.

*Bearer Capability*: Define los requerimientos de conexión y transmisión que la red puede soportar. Es obligatorio para llamadas MO y condicional (incluido de estar disponible) para llamadas MT y MF. Incluye estos datos:

- Coding Standard:
  - 0 - CCITT Standard.
  - 1 - ISO/IEC Standard.
  - 2 - National Standard.
  - 3 - Standard Defined for the Network.
- Information Transfer Capability:
  - 00 - Speech.
  - 08 - Unrestricted Digital.
  - 09 - Restricted Digital.
  - 16 - 3.1 KHz Audio.
  - 17 - Unrestricted Digital with Tones or Announcements.
  - 24 - Video.
- Transfer Mode:
  - 0 - Circuit Mode.
  - 1 - Packet Mode.
- Information Transfer Rate:
  - 16 - 64 Kbit/s Circuit Mode.
  - 17 - 128 Kbit/s Circuit Mode.
  - 19 - 384 Kbit/s Circuit Mode.
  - 21 - 1536 Kbit/s Circuit Mode.
  - 23 - 1920 Kbit/s Circuit Mode.
  - 32 - Packet Mode Calls.
- Layer 1 Identifier:
  - 1 - Layer 1.
- User Information Layer 1 Protocol:
  - 1 - CCITT Standard Rate Adaption V.110 and X.30.

- 2 - Recommendation G.711 M-law.
- 3 - Recommendation G.711 A-law.
- 4 - Recommendation G.721 32 Kbit/s ADPCM.
- 5 - Recommendations H.221 and H.242.
- 7 - Non-CCITT Standard Rate Adaption.
- 8 - CCITT Standard Rate Adaption V.120.
- 9 - CCITT Standard Rate Adaption X.31.

*Event Type BCSM:* Define el evento (del Basic Call State Model) que dispara la operación invocada. Es obligatorio para llamadas MO, MT y MF. Pude ser configurado con uno de estos valores:

- 01 - Origination Attempt Authorized.
- 02 - Collected Info.
- 03 - Analyzed Info.
- 04 - Route Select Failure.
- 05 - O Called Party Busy.
- 06 - O No Answer.
- 07 - O Answer.
- 08 - O Mid Call.
- 09 - O Disconnect.
- 10 - O Abandon.
- 12 - Termination Attempt Authorized.
- 13 - T Called Party Busy.
- 14 - T No Answer.
- 15 - T Answer.
- 17 - T Disconnect.
- 18 - T Abandon.

*Redirecting Party Id:* Define quien desvía la llamada y sus características, en un escenario de desvío de llamada (si "A" llama a "B" y este desvía a "C", entonces el IDP de la llamada desviada tendrá a "B" en este argumento). No es aplicable para llamadas MO y MT y es condicional (incluido de estar disponible) para llamadas MF. Incluye estos datos:

- Nature of Address Indicator:

- 1 - Subscriber Number.
- 2 - Unknown Number.
- 3 - National Number.
- 4 - International Number.
- 5 - Network-specific Number.
- Numbering Plan Indicator:
  - 1 - ISDN Numbering Plan.
  - 3 - Data Numbering Plan.
  - 4 - Telex Numbering Plan.
- Address Presentation Restricted Indicator:
  - 0 - Presentation Allowed.
  - 1 - Presentation Restricted.
  - 2 - Address Not Available.
- Address Signals: Número de quién desvía la llamada.

*Redirection Information:* Define las características vinculadas al desvío de la llamada. No es aplicable para llamadas MO y MT y es obligatorio para llamadas MF. Incluye estos datos:

- Original Redirection Reason:
  - 0 - Unknown or Not Available.
  - 1 - User Busy.
  - 2 - No Reply.
  - 3 - Unconditional.
- Redirecting Indicator:
  - 0 - No Redirection.
  - 1 - Call Rerouted.
  - 2 - Call Rerouted - All Redirection Information Presentation Restricted.
  - 3 - Call Diverted.
  - 4 - Call Diverted - All Redirection Information Presentation Restricted.
  - 5 - Call Rerouted - Redirection Number Presentation Restricted.
  - 6 - Call Diverted - Redirection Number Presentation Restricted.
- Redirecting Reason:

- 0 - Not Available.
  - 1 - User Busy.
  - 2 - No Reply.
  - 3 - Unconditional.
  - 4 - Deflection During Alerting.
  - 5 - Deflection Immediate Response.
  - 6 - Mobile Subscriber Not Reachable.
- Redirection Counter:
- 1 - First Redirection.
  - 2 - Second Redirection.
  - 3 - Third Redirection.
  - 4 - Fourth Redirection.
  - 5 - Fifth Redirection.

*IMS/*: Define la identificación internacional móvil de un suscriptor, proporcionada mediante su tarjeta SIM (o equivalente) para identificarlo unívocamente dentro de la red celular. Es obligatorio para llamadas MO, MT y MF. Está configurado con la combinación del código de país (MCC), código de red (MNC) y número de identificación de suscriptor móvil (MSIN).

*Subscriber State*: Define el estado de la terminal móvil perteneciente a quien recibe la llamada. No es aplicable a llamadas MO y es condicional (incluido de estar disponible) para llamadas MT y MF. Puede ser configurado con uno de estos valores:

- 1 - CAMEL Busy.
- 2 - Network Determined Not Reachable.
- 3 - Assumed Idle.

*Location Information*: Define la ubicación exacta de quien realiza la llamada (MO) o de quien la recibe (MT). Es obligatorio para llamadas MO, condicional (incluido de estar disponible) para llamadas MT y no es aplicable para llamadas MF. Incluye estos sub-argumentos:

Age of Location Information: Define el tiempo de llamada transcurrido en segundos. Normalmente está configurado en 0, si la terminal móvil está en contacto con la red, o en 30, si la llamada ha sido desviada a un contestador (voice mail).

VLR Number: Define el VLR (Visitor Location Register) que maneja la llamada y sus características. Incluye estos datos:

- Numbering Plan Indicator:
  - 1 - ISDN Numbering Plan.
  - 3 - Data Numbering Plan.
  - 4 - Telex Numbering Plan.
- Nature of Address Indicator:
  - 0 - Unknown.
  - 1 - International Number.
  - 2 - National Number.
  - 3 - Network-specific Number.
  - 4 - Short Code.
- Address Digits: Número del VLR que maneja la llamada.

CGI / SAI / LAI: Define la forma de determinar la ubicación. Normalmente puede ser mediante una CGI (Cell Global Id), SAI (Service Area Id) o LAI (Location Area Id).

Cell Id: Define la CGI, SAI o LAI en la que la terminal móvil está registrada. Incluye estos datos:

- MCC (Mobile Country Code).
- MNC (Mobile Network Code).
- LAC (Location Area Code).
- CI (Cell Id).

*Extended Basic Service Code*: Define el tipo de servicio utilizado, el cual puede ser “Teleservice”, para conexiones entre dos terminales móviles (llamadas), o “Bearer Service”, para conexiones entre una terminal móvil y un APN (sesiones GPRS). Es obligatorio para llamadas MO, MT y MF. Incluye estos datos:

- Teleservice Code:
  - 000 - All Teleservices.
  - 016 - All Speech Transmission Services.
  - 017 - Telephony.
  - 019 - Emergency Calls.
  - 032 - All Short Message Services.

- 033 - Short Message Mobile Terminated.
- 034 - Short Message Mobile Originated.
- 096 - All Facsimile Transmission Services.
- 097 - Facsimile Group 3 and Alter Speech.
- 098 - Automatic Facsimile Group 3.
- 099 - Facsimile Group 4.
- 112 - All Data Teleservices.
- 128 - All Teleservices except Short Message Services.

*Call Reference Number:* Define el valor unívoco que identifica a una llamada. Es obligatorio para llamadas MO, MT y MF. Normalmente está configurado con la combinación del identificador del MSC y su contador de llamadas, en formato hexadecimal.

*MSC Address:* Define el MSC (Mobile Switching Center) que maneja la llamada y sus características. Es obligatorio para llamadas MO, MT y MF. Incluye estos datos:

- Numbering Plan Indicator:
  - 1 - ISDN Numbering Plan.
  - 3 - Data Numbering Plan.
  - 4 - Telex Numbering Plan.
- Nature of Address Indicator:
  - 0 - Unknown.
  - 1 - International Number.
  - 2 - National Number.
  - 3 - Network-specific Number.
  - 4 - Short Code.
- Address Digits: Número del MSC que maneja la llamada.

*Called Party BCD Number:* Define quien recibe la llamada y sus características. Es obligatorio para llamadas MO y no es aplicable para llamadas MT y MF (utilizan el *Called Party Number*). Incluye estos datos:

- Type of Number Indicator:
  - 0 - Unknown.
  - 1 - International Number.

- 2 - National Number.
- 3 - Network-specific Number.
- 4 - Short Code.
- Numbering Plan Identification:
  - 1 - ISDN Numbering Plan.
  - 3 - Data Numbering Plan.
  - 4 - Telex Numbering Plan.
- Address Digits: Número de quien recibe la llamada.

*Time and Timezone:* Define la fecha, hora y zona horaria de la llamada. Es obligatorio para llamadas MO, MT y MF. La fecha es configurada con el formato “AAAAMMDD”, la hora con el formato “HHMMSS” y la zona horaria con el formato “GMT +/- ZH Offset” (“GMT + 1H”, “GMT - 3H”, etc).

A continuación se muestran ejemplos de IDPs extraídos de llamadas MO, MT y MF:

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Invoke ID	0	
Operation Code	0	IDP - Initial DP
Service key	1	
Called party number (Q.763)	N/A	Not Applicable
+ Odd/even indicator	N/A	Not Applicable
+ Nature of address indicator	N/A	Not Applicable
+ Internal network number indicator	N/A	Not Applicable
+ Numbering plan indicator	N/A	Not Applicable
+ Spare	N/A	Not Applicable
+ Address signals	N/A	Not Applicable
Calling party number (Q.763)		
+ Odd/even indicator	1	odd
+ Nature of address indicator	4	international number
+ Calling party number incomplete indicator	0	complete
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	1	presentation restricted
+ Screening indicator	1	user provided, verified and passed
+ Address signals		5491133469216
Calling party's category (Q.763)		
+ Calling party's category (Q.763)	10	ordinary calling subscriber
IPSSP capabilities		
+ IP routing add	0	not supported
+ Voice back	0	not supported
+ Voice information via speech recognition	0	not supported
+ Voice information via voice recognition	0	not supported
+ Generation of voice announcements from text	0	not supported
+ Reserved	0	0

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Location number (Q.763)		
+ Odd/even indicator	0	even
+ Nature of address indicator	4	international number
+ Internal network number indicator	1	routing to internal network number not allowed
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	1	presentation restricted
+ Screening indicator	3	network provided
+ Address signals		543200130020
Original Called Party ID (Q.1218)	N/A	Not Applicable
+ Odd/even indicator	N/A	Not Applicable
+ Nature of address indicator	N/A	Not Applicable
+ Spare	N/A	Not Applicable
+ Numbering plan indicator	N/A	Not Applicable
+ Address presentation restricted indicator	N/A	Not Applicable
+ Spares	N/A	Not Applicable
+ Address signals	N/A	Not Applicable
High layer compatibility (Q.931)		
+ Extension bit	1	Not continued
+ Coding standard	0	CCITT standard
+ Interpretation	4	First HLC identification
+ Pres protocol profile	1	High layer profil
+ Extension bit	1	Not continued
+ High layer charact	1	Telephony
Bearer capability		
+ Bearer capability (Q.931)		
+ Extension bit	1	Not continued
+ Coding standard	0	CCITT standard
+ Info transfer capability	0	Speech
+ Extension bit	1	Not continued
+ Transfer mode	0	Circuit mode
+ Info transfer rate	16	64 kbits circuit mode
+ Extension bit	1	Not continued
+ Layer1 id	1	Layer 1
+ User info layer 1 protocol	3	Recommendation G.711 A-law
Event type BCSM	2	Collect info
Redirecting Party ID (Q.1218)	N/A	Not Applicable
+ Odd/even indicator	N/A	Not Applicable
+ Nature of address indicator	N/A	Not Applicable
+ Spare	N/A	Not Applicable
+ Numbering plan indicator	N/A	Not Applicable
+ Address presentation restricted indicator	N/A	Not Applicable
+ Spares	N/A	Not Applicable
+ Address signals	N/A	Not Applicable
Redirection Information (Q.763)	N/A	Not Applicable
+ Original redirection reason	N/A	Not Applicable
+ Spare	N/A	Not Applicable
+ Redirecting indicator	N/A	Not Applicable
+ Redirecting reason	N/A	Not Applicable
+ National use	N/A	Not Applicable
+ Redirection counter	N/A	Not Applicable
IMSI		
+ MCC/MNC/MSIN		722310006665472

CAP GSM 09.78		MO Call	
CAMEL Components	Value	Description	
Subscriber state	N/A	Not Applicable	
Location information			
+ Age Of Location Information	0	MS currently in contact with the Network	
+ VLR number			
+ Numbering plan indicator	1	ISDN/Telephony Numbering Plan (Rec CCITT E.164)	
+ Nature of address indicator	1	international number	
+ Extension bit	1	no extension	
+ CC/NDC/SN	543200130020		
+ Cell global id or service area id or LAI			
+ Cell Id Fixed Length			
+ MCC	722		
+ MNC	203		
+ LAC	1110		
+ CI	21272		
Ext basic service code			
+ Teleservice			
+ Teleservice code	17	telephony	
Call reference number			
+ Dump	40 1D 41 32 01 12 35		
MSC address			
+ Numbering plan indicator	1	ISDN/Telephony Numbering Plan (Rec CCITT E.164)	
+ Nature of address indicator	1	international number	
+ Extension bit	1	no extension	
+ CC/NDC/SN	543200130020		
Called party BCD number			
+ Extension bit	1	no extension	
+ Type of number	0	Unknown	
+ Numbering plan identification	1	ISDN/telephony numbering plan (Rec. E.164/E.163)	
+ Address digits	5493516450746		
Time and timezone			
+ YYYY/MM/DD	20090102		
+ HH:MM:SS	171812		
+ Time zone	136	GMT - 2H	

CAP GSM 09.78		MT Call	
CAMEL Components	Value	Description	
Invoke ID	0		
Operation Code	0	IDP - Initial DP	
Service key	1		
Called party number (Q.763)			
+ Odd/even indicator	1	odd	
+ Nature of address indicator	4	international number	
+ Internal network number indicator	0	routing to internal network number allowed	
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)	
+ Spare	0	-	
+ Address signals	5491133469216		

CAP GSM 09.78	MT Call	
CAMEL Components	Value	Description
Calling party number (Q.763)		
+ Odd/even indicator	1	odd
+ Nature of address indicator	4	international number
+ Calling party number incomplete indicator	0	complete
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	0	presentation allowed
+ Screening indicator	3	network provided
+ Address signals		5493516450746
Calling party's category (Q.763)		
+ Calling party's category (Q.763)	10	ordinary calling subscriber
IPSSP capabilities		
+ IP routing add	0	not supported
+ Voice back	0	not supported
+ Voice information via speech recognition	0	not supported
+ Voice information via voice recognition	0	not supported
+ Generation of voice announcements from text	0	not supported
+ Reserved	0	0
Location number (Q.763)		
+ Odd/even indicator	0	even
+ Nature of address indicator	4	international number
+ Internal network number indicator	1	routing to internal network number not allowed
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	1	presentation restricted
+ Screening indicator	3	network provided
+ Address signals		543200130020
Original Called Party ID (Q.1218)	N/A	Not Applicable
+ Odd/even indicator	N/A	Not Applicable
+ Nature of address indicator	N/A	Not Applicable
+ Spare	N/A	Not Applicable
+ Numbering plan indicator	N/A	Not Applicable
+ Address presentation restricted indicator	N/A	Not Applicable
+ Spares	N/A	Not Applicable
+ Address signals	N/A	Not Applicable
High layer compatibility (Q.931)		
+ Extension bit	1	Not continued
+ Coding standard	0	CCITT standard
+ Interpretation	4	First HLC identification
+ Pres protocol profile	1	High layer profil
+ Extension bit	1	Not continued
+ High layer charact	1	Telephony
Bearer capability		
+ Bearer capability (Q.931)		
+ Extension bit	1	Not continued
+ Coding standard	0	CCITT standard
+ Info transfer capability	0	Speech
+ Extension bit	1	Not continued
+ Transfer mode	0	Circuit mode
+ Info transfer rate	16	64 kbits circuit mode
+ Extension bit	1	Not continued
+ Layer1 id	1	Layer 1
+ User info layer 1 protocol	3	Recommendation G.711 A-law
Event type BCSM	12	Term attempt authorized

CAP GSM 09.78	MT Call	
CAMEL Components	Value	Description
Redirecting Party ID (Q.1218)	N/A	Not Applicable
+ Odd/even indicator	N/A	Not Applicable
+ Nature of address indicator	N/A	Not Applicable
+ Spare	N/A	Not Applicable
+ Numbering plan indicator	N/A	Not Applicable
+ Address presentation restricted indicator	N/A	Not Applicable
+ Spares	N/A	Not Applicable
+ Address signals	N/A	Not Applicable
Redirection Information (Q.763)	N/A	Not Applicable
+ Original redirection reason	N/A	Not Applicable
+ Spare	N/A	Not Applicable
+ Redirecting indicator	N/A	Not Applicable
+ Redirecting reason	N/A	Not Applicable
+ National use	N/A	Not Applicable
+ Redirection counter	N/A	Not Applicable
IMSI		
+ MCC/MNC/MSIN		722310006665472
Subscriber state	0	Assumed Idle
Location information		
+ Age Of Location Information	0	MS currently in contact with the Network
+ VLR number		
+ Numbering plan indicator	1	ISDN/Telephony Numbering Plan (Rec CCITT E.164)
+ Nature of address indicator	1	international number
+ Extension bit	1	no extension
+ CC/NDC/SN		543200130020
+ Cell global id or service area id or LAI		
+ Cell Id Fixed Length		
+ MCC		722
+ MNC		203
+ LAC		1110
+ CI		21272
Ext basic service code		
+ Teleservice		
+ Teleservice code	17	telephony
Call reference number		
+ Dump		42 5E 22 34 02 1A 3D
MSC address		
+ Numbering plan indicator	1	ISDN/Telephony Numbering Plan (Rec CCITT E.164)
+ Nature of address indicator	1	international number
+ Extension bit	1	no extension
+ CC/NDC/SN		543200130020
Called party BCD number	N/A	Not Applicable
+ Extension bit	N/A	Not Applicable
+ Type of number	N/A	Not Applicable
+ Numbering plan identification	N/A	Not Applicable
+ Address digits	N/A	Not Applicable
Time and timezone		
+ YYYY/MMDD		20090102
+ HH:MM:SS		142812
+ Time zone	136	GMT - 2H

CAP GSM 09.78	MF Call	
CAMEL Components	Value	Description
Invoke ID	0	
Operation Code	0	IDP - Initial DP
Service key	1	
Called party number (Q.763)		
+ Odd/even indicator	0	even
+ Nature of address indicator	4	international number
+ Internal network number indicator	0	routing to internal network number allowed
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Spare	0	-
+ Address signals		105493543513036F
Calling party number (Q.763)		
+ Odd/even indicator	1	odd
+ Nature of address indicator	4	international number
+ Calling party number incomplete indicator	0	complete
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	0	presentation allowed
+ Screening indicator	1	user provided, verified and passed
+ Address signals		5493541669351
Calling party's category (Q.763)		
+ Calling party's category (Q.763)	10	ordinary calling subscriber
IPSSP capabilities		
+ IP routing add	0	not supported
+ Voice back	0	not supported
+ Voice information via speech recognition	0	not supported
+ Voice information via voice recognition	0	not supported
+ Generation of voice announcements from text	0	not supported
+ Reserved	0	0
Location number (Q.763)	N/A	Not Applicable
+ Odd/even indicator	N/A	Not Applicable
+ Nature of address indicator	N/A	Not Applicable
+ Internal network number indicator	N/A	Not Applicable
+ Numbering plan indicator	N/A	Not Applicable
+ Address presentation restricted indicator	N/A	Not Applicable
+ Screening indicator	N/A	Not Applicable
+ Address signals	N/A	Not Applicable
Original Called Party ID (Q.1218)		
+ Odd/even indicator	1	odd
+ Nature of address indicator	4	international number
+ Spare	0	-
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	1	presentation restricted
+ Spares	0	0
+ Address signals		5493543513036
High layer compatibility (Q.931)		
+ Extension bit	1	Not continued
+ Coding standard	0	CCITT standard
+ Interpretation	4	First HLC identification
+ Pres protocol profile	1	High layer profil
+ Extension bit	1	Not continued
+ High layer charact	1	Telephony

CAP GSM 09.78	MF Call	
CAMEL Components	Value	Description
Bearer capability		
+ Bearer capability (Q.931)		
+ Extension bit	1	Not continued
+ Coding standard	0	CCITT standard
+ Info transfer capability	0	Speech
+ Extension bit	1	Not continued
+ Transfer mode	0	Circuit mode
+ Info transfer rate	16	64 kbits circuit mode
+ Extension bit	1	Not continued
+ Layer1 id	1	Layer 1
+ User info layer 1 protocol	3	Recommendation G.711 A-law
Event type BCSM	2	Collect info
Redirecting Party ID (Q.1218)		
+ Odd/even indicator	1	odd
+ Nature of address indicator	4	international number
+ Spare	0	-
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	1	presentation restricted
+ Spares	0	0
+ Address signals		5493543513036
Redirection Information (Q.763)		
+ Original redirection reason	1	User busy
+ Spare	0	-
+ Redirecting indicator	3	Call diverted
+ Redirecting reason	1	User busy
+ National use	0	0
+ Redirection counter	1	
IMSI		
+ MCC/MNC/MSIN		722310006665480
Subscriber state	0	Assumed Idle
Location information	N/A	Not Applicable
+ Age Of Location Information	N/A	Not Applicable
+ VLR number	N/A	Not Applicable
+ Numbering plan indicator	N/A	Not Applicable
+ Nature of address indicator	N/A	Not Applicable
+ Extension bit	N/A	Not Applicable
+ CC/NDC/SN	N/A	Not Applicable
+ Cell global id or service area id or LAI	N/A	Not Applicable
+ Cell Id Fixed Length	N/A	Not Applicable
+ MCC	N/A	Not Applicable
+ MNC	N/A	Not Applicable
+ LAC	N/A	Not Applicable
+ CI	N/A	Not Applicable
Ext basic service code		
+ Teleservice		
+ Teleservice code	17	telephony
Call reference number		
+ Dump		41 37 01 36 03 0F 4C

CAP GSM 09.78	MF Call	
CAMEL Components	Value	Description
MSC address		
+ Numbering plan indicator	1	ISDN/Telephony Numbering Plan (Rec CCITT E.164)
+ Nature of address indicator	1	international number
+ Extension bit	1	no extension
+ CC/NDC/SN		543200130020
Called party BCD number	N/A	Not Applicable
+ Extension bit	N/A	Not Applicable
+ Type of number	N/A	Not Applicable
+ Numbering plan identification	N/A	Not Applicable
+ Address digits	N/A	Not Applicable
Time and timezone		
+ YYYY/MM/DD		20090102
+ HH:MM:SS		120812
+ Time zone	136	GMT - 2H

### **Request Report Basic Call State Model (RRB)**

Es un mensaje enviado por el SCP para indicarle al SSP cuáles son los eventos, ocurridos durante una llamada, que han de serle reportados. Es decir, el SSP se encarga de monitorear los estados por los que pasa una llamada y los eventos que desencadenan cambios; cuando uno de ellos ocurre, de haber sido solicitado por el SCP, será reportado para que éste último actúe en consecuencia. Puede contener los siguientes argumentos:

*Invoke Id:* Define el número de secuencia de la operación invocada.

*Operation Code:* Define el código asociado a la operación (RRB - Request Report BCSM Event).

*BCSM Event #:* Define los eventos (del Basic Call State Model) que deben ser monitoreados y quien los genera (Leg 1: El que hace la llamada / Leg 2: El que recibe la llamada). Este se repetirá por cada evento que se desee controlar. Incluye estos datos:

- Event Type BCSM:
  - 01 - Origination Attempt Authorized.
  - 02 - Collected Info.
  - 03 - Analyzed Info.
  - 04 - Route Select Failure.
  - 05 - O Called Party Busy.
  - 06 - O No Answer.

- 07 - O Answer.
- 08 - O Mid Call.
- 09 - O Disconnect.
- 10 - O Abandon.
- 12 - Termination Attempt Authorized.
- 13 - T Called Party Busy.
- 14 - T No Answer.
- 15 - T Answer.
- 17 - T Disconnect.
- 18 - T Abandon.
- Monitor Mode:
  - 0 - Interrupted.
  - 1 - Notify and Continue.
  - 2 - Transparent.
- Leg Id / Sending Side:
  - 1 - Leg 1.
  - 2 - Leg 2.

A continuación se muestran ejemplos de RRBs extraídos de llamadas MO y MT:

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Invoke ID	1	
Operation Code	23	RRB - Request Report BCSM Event
Bcsm event 1		
+ Event type BCSM	9	O disconnect
+ Monitor Mode	0	Interrupted
+ Leg ID		
+ Sending Side	1	Leg 1
Bcsm event 2		
+ Event type BCSM	9	O disconnect
+ Monitor Mode	0	Interrupted
+ Leg ID		
+ Sending Side	2	Leg 2

CAP GSM 09.78	MT Call	
CAMEL Components	Value	Description
Invoke ID	1	
Operation Code	23	RRB - Request Report BCSM Event
Bcsm event 1		
+ Event type BCSM	17	T disconnect
+ Monitor Mode	0	Interrupted
+ Leg ID		
+ Sending Side	1	Leg 1
Bcsm event 2		
+ Event type BCSM	17	T disconnect
+ Monitor Mode	0	Interrupted
+ Leg ID		
+ Sending Side	2	Leg 2
Bcsm event 3		
+ Event type BCSM	14	T no answer
+ Monitor Mode	1	Notify and continue
+ Leg ID		
+ Sending Side	2	Leg 2
Bcsm event 4		
+ Event type BCSM	13	T called party busy
+ Monitor Mode	1	Notify and continue
+ Leg ID		
+ Sending Side	2	Leg 2

### **Apply Charging (AC)**

Es un mensaje enviado por el SCP para indicarle al SSP a quien se le cobra la llamada, junto con el tiempo que debe transcurrir hasta que este último se reporte con información sobre el estado de la misma. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (AC - Apply Charging).

*ACH Billing Charging Characteristics*: Define las condiciones que el SSP debe tener en cuenta para poder informar sobre el estado de una llamada. Básicamente, se indica a quien se le está cobrando la llamada y en cuanto tiempo se debe informar al SCP sobre el estado de la misma. Este tiempo lo configura la operadora móvil y, si bien suele ser de 60 o 120 segundos, depende principalmente de los controles de tarificación que se llevan a cabo: Balance disponible del suscriptor, tipo de llamada, cambios en la banda horaria, etc. Incluye estos datos:

- Time Duration Charging / Max Call Period Duration.
- Party to Charge / Sending Side:
  - 1 - Leg 1.

## 2 - Leg 2.

A continuación se muestran ejemplos de ACs extraídos de llamadas MO y MT:

CAP GSM 09.78		MO Call	
CAMEL Components	Value	Description	
Invoke ID	2		
Operation Code	35	AC - Apply Charging	
ACH billing charging characteristics			
+ Time duration charging			
+ Max call period duration	1200		
+ Party to charge			
+ Sending Side	1	Leg 1	

CAP GSM 09.78		MT Call	
CAMEL Components	Value	Description	
Invoke ID	2		
Operation Code	35	AC - Apply Charging	
ACH billing charging characteristics			
+ Time duration charging			
+ Max call period duration	1200		
+ Party to charge			
+ Sending Side	2	Leg 2	

### Connect (CON)

Es un mensaje enviado por el SCP con el objeto de indicarle al SSP con que destino debe conectar la llamada, junto con la información necesaria para llevar esto a cabo. Es enviado en caso de que alguno de los datos que se informaron en el IDP deba ser cambiado. Si la llamada se puede conectar con la misma información que vino en el IDP, es posible usar en su lugar el mensaje *Continue (CUE)*. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (CON - Connect).

*Destination Routing Address*: Define el destino con el que se debe conectar la llamada y sus características. Es obligatorio para llamadas MO, MT y MF. Incluye estos datos:

- Nature of Address Indicator:
  - 1 - Subscriber Number.
  - 2 - Unknown Number.
  - 3 - National Number.
  - 4 - International Number.

- 5 - Network-specific Number.
- Internal Network Number Indicator:
  - 0 - Routing to Internal Network Number Allowed.
  - 1 - Routing to Internal Network Number Not Allowed.
- Numbering Plan Indicator:
  - 1 - ISDN Numbering Plan.
  - 3 - Data Numbering Plan.
  - 4 - Telex Numbering Plan.
- Address Signals: Número del destino al que se conecta la llamada.

A continuación se muestran ejemplos de CONs extraídos de llamadas MO, MT y MF:

CAP GSM 09.78		MO Call	
CAMEL Components	Value	Description	
Invoke ID	3		
Operation Code	20	CON - Connect	
Destination Routing Address			
+ Odd/even indicator	1	odd	
+ Nature of address indicator	4	international number	
+ Internal network number indicator	0	routing to internal network number allowed	
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)	
+ Spare	0	-	
+ Address signals		5493413516578	

CAP GSM 09.78		MT Call	
CAMEL Components	Value	Description	
Invoke ID	3		
Operation Code	20	CON - Connect	
Destination Routing Address			
+ Odd/even indicator	1	odd	
+ Nature of address indicator	4	international number	
+ Internal network number indicator	0	routing to internal network number allowed	
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)	
+ Spare	0	-	
+ Address signals		5493543513036	

CAP GSM 09.78	MF Call	
CAMEL Components	Value	Description
Invoke ID	3	
Operation Code	20	CON - Connect
Destination Routing Address		
+ Odd/even indicator	1	odd
+ Nature of address indicator	4	international number
+ Internal network number indicator	0	routing to internal network number allowed
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Spare	0	-
+ Address signals		105493543513036

### **Apply Charging Report (ACR)**

Es un mensaje enviado por el SSP, informando al SCP si la llamada está activa y el tiempo que ha transcurrido desde su comienzo. Por lo tanto, también sirve para indicar que la misma ha finalizado. Puede contener los siguientes argumentos:

*Invoke Id:* Define el número de secuencia de la operación invocada.

*Operation Code:* Define el código asociado a la operación (ACR - Apply Charging Report).

*Call Result:* Define el estado de la llamada, informando a quien se le está cobrando, el tiempo transcurrido desde su inicio y si la misma sigue activa o no. Incluye estos datos:

- Party to Charge / Receiving Side:
  - 1 - Leg 1.
  - 2 - Leg 2.
- Time Information / Time if No Tariff Switch.
- Leg Active:
  - 0 - False.
  - 1 - True.

A continuación se muestran ejemplos de ACRs extraídos de llamadas MO y MT:

CAP GSM 09.78		MO Call	
CAMEL Components	Value	Description	
Invoke ID	1		
Operation Code	36	ACR - Apply Charging Report	
Call result			
+ Party to charge			
+ Receiving Side	1	Leg 1	
+ Time Information			
+ Time if no tariff switch	1200		
+ Leg active	1	true	

CAP GSM 09.78		MT Call	
CAMEL Components	Value	Description	
Invoke ID	1		
Operation Code	36	ACR - Apply Charging Report	
Call result			
+ Party to charge			
+ Receiving Side	2	Leg 2	
+ Time Information			
+ Time if no tariff switch	0		
+ Leg active	0	false	

### **Event Report Basic Call State Model (ERB)**

Es un mensaje enviado por el SSP, informando la ocurrencia de alguno de los eventos que el SCP pidió monitorear, para que este último actúe en consecuencia. Puede contener los siguientes argumentos:

*Invoke Id:* Define el número de secuencia de la operación invocada.

*Operation Code:* Define el código asociado a la operación (ERB - Event Report BCSM).

*Event Type BCSM:* Define el evento (del Basic Call State Model) que estaba siendo monitoreado y ha ocurrido. Puede ser configurado con uno de estos valores:

- 01 - Origination Attempt Authorized.
- 02 - Collected Info.
- 03 - Analyzed Info.
- 04 - Route Select Failure.
- 05 - O Called Party Busy.
- 06 - O No Answer.
- 07 - O Answer.
- 08 - O Mid Call.
- 09 - O Disconnect.

- 10 - O Abandon.
- 12 - Termination Attempt Authorized.
- 13 - T Called Party Busy.
- 14 - T No Answer.
- 15 - T Answer.
- 17 - T Disconnect.
- 18 - T Abandon.

*Event Specific Information:* Define los detalles del evento reportado. Incluye este sub-argumento:

Cause IE (Information Element): Define las causas y las características del evento reportado. Incluye estos datos:

- Coding Standard:
  - 0 - CCITT Standard.
  - 1 - ISO/IEC Standard.
  - 2 - National Standard.
  - 3 - Standard Defined for the Network.
- Location:
  - 00 - User.
  - 01 - Private Network Serving the Local User.
  - 02 - Public Network Serving the Local User.
  - 03 - Transit Network.
  - 04 - Public Network Serving the Remote User.
  - 05 - Private Network Serving the Remote User.
  - 07 - International Network.
  - 10 - Network Beyond Interworking Point.
- Cause Value:
  - 001 - Unallocated Number.
  - 002 - No Route to Transit Network.
  - 003 - No Route to Destination.
  - 004 - Send Special Information Tone.
  - 005 - Misdialed Trunk Prefix.
  - 006 - Channel Unacceptable.

- 007 - Call Awarded in Established Channel.
- 008 - Preemption.
- 009 - Preemption - Circuit Reserved for Reuse.
- 016 - Normal Call Clearing.
- 017 - User Busy.
- 018 - No User Responding.
- 019 - No Answer from User - User Alerted.
- 020 - Subscriber Absent.
- 021 - Call Rejected.
- 022 - Number Changed.
- 023 - Redirection to New Destination.
- 025 - Exchange Routing Error.
- 026 - Non-selected User Clearing.
- 027 - Destination Out of Order.
- 028 - Invalid Number Format - Address Incomplete.
- 029 - Facility Rejected.
- 030 - Response to Status Enquiry.
- 031 - Normal - Unspecified.
- 034 - No Circuit/Channel Available.
- 038 - Network Out of Order.
- 039 - Permanent Frame Mode Connection OoS.
- 040 - Permanent Frame Mode Connection Oper.
- 041 - Temporary Failure.
- 042 - Switching Equipment Congestion.
- 043 - Access Information Discarded.
- 044 - Requested Circuit/Channel N/A.
- 046 - Precedence Call Blocked.
- 047 - Resource Unavailable - Unspecified.
- 049 - Quality of Service Not Available.
- 050 - Requested Facility Not Subscribed.
- 057 - Bearer Capability Not Authorized.
- 058 - Bearer Capability Not Available.
- 063 - Service or Option N/A - Unspecified.

- 065 - Bearer Capability Not Implemented.
- 066 - Channel Type Not Implemented.
- 069 - Requested Facility Not Implemented.
- 070 - Only Restricted Digital Bearer Capability Supported.
- 079 - Service or Option Not Implemented - Unspecified.
- 081 - Invalid Call Reference Value.
- 082 - Identified Channel Does Not Exist.
- 084 - Call Identity in User.
- 085 - No Call Suspended.
- 086 - Call with Requested Call Identity has Cleared.
- 088 - Incompatible Destination.
- 091 - Invalid Transit Network Selection.
- 095 - Invalid Message - Unspecified.
- 096 - Mandatory Information Element is Missing.
- 097 - Message Type Non-existent / Not Implemented.
- 098 - Message Incompatible with Call State or Message Type.
- 099 - IE/Parameter Non-existent or Not Implemented.
- 100 - Invalid Information Element Contents.
- 101 - Message Not Compatible With Call State.
- 102 - Recovery on Timer Expiry.
- 103 - Parameter Non-existent / Not Implemented - Passed On.
- 110 - Message with Unrecognized Parameter - Discarded.
- 111 - Protocol Error - Unspecified.
- 127 - Interworking - Unspecified.

*Leg Id:* Define quien es el que genera el evento reportado (Leg 1: El que hace la llamada / Leg 2: El que recibe la llamada). Incluye estos datos:

- Receiving Side:
  - 1 - Leg 1.
  - 2 - Leg 2.

*Miscellaneous Call Info:* Define la información adicional del evento reportado. Incluye estos datos:

- Message Type:

0 - Request.

1 - Notification.

A continuación se muestran ejemplos de ERBs extraídos de llamadas MO y MT:

CAP GSM 09.78		MO Call	
CAMEL Components	Value	Description	
Invoke ID	2		
Operation Code	24	ERB - Event Report BCSM	
Event type BCSM	5	O called party busy	
Event specific information			
+ Cause IE (Q.850)			
+ Extension bit	1	Not continued	
+ Coding standard	0	CCITT standard	
+ Spare	0	spare	
+ Location	2	Public network serving local user	
+ Extension bit	1	Not continued	
+ Cause value	21	Call rejected	
Leg ID			
+ Receiving Side	1	Leg 1	
Misc call info			
+ Message type	1	Notification	

CAP GSM 09.78		MT Call	
CAMEL Components	Value	Description	
Invoke ID	2		
Operation Code	24	ERB - Event Report BCSM	
Event type BCSM	17	T disconnect	
Event specific information			
+ Cause IE (Q.850)			
+ Extension bit	1	Not continued	
+ Coding standard	0	CCITT standard	
+ Spare	0	spare	
+ Location	1	Private network serving local user	
+ Extension bit	1	Not continued	
+ Cause value	16	Normal call clearing	
Leg ID			
+ Receiving Side	2	Leg 2	
Misc call info			
+ Message type	0	Request	

### **Release Call (RC)**

Es un mensaje enviado por el SCP para indicarle al SSP que la llamada ha terminado, por lo que todas sus conexiones en la Red Inteligente han de ser liberadas. Es utilizado tanto para finales exitosos como fallidos. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (RC - Release Call).

*Cause IE (Information Element)*: Define las causas y las características de la terminación de la llamada. Incluye estos datos:

- Coding Standard:
  - 0 - CCITT Standard.
  - 1 - ISO/IEC Standard.
  - 2 - National Standard.
  - 3 - Standard Defined for the Network.
  
- Location:
  - 00 - User.
  - 01 - Private Network Serving the Local User.
  - 02 - Public Network Serving the Local User.
  - 03 - Transit Network.
  - 04 - Public Network Serving the Remote User.
  - 05 - Private Network Serving the Remote User.
  - 07 - International Network.
  - 10 - Network Beyond Interworking Point.
  
- Cause Value:
  - 001 - Unallocated Number.
  - 002 - No Route to Transit Network.
  - 003 - No Route to Destination.
  - 004 - Send Special Information Tone.
  - 005 - Misdialed Trunk Prefix.
  - 006 - Channel Unacceptable.
  - 007 - Call Awarded in Established Channel.
  - 008 - Preemption.
  - 009 - Preemption - Circuit Reserved for Reuse.
  - 016 - Normal Call Clearing.
  - 017 - User Busy.
  - 018 - No User Responding.
  - 019 - No Answer from User - User Alerted.
  - 020 - Subscriber Absent.

- 021 - Call Rejected.
- 022 - Number Changed.
- 023 - Redirection to New Destination.
- 025 - Exchange Routing Error.
- 026 - Non-selected User Clearing.
- 027 - Destination Out of Order.
- 028 - Invalid Number Format - Address Incomplete.
- 029 - Facility Rejected.
- 030 - Response to Status Enquiry.
- 031 - Normal - Unspecified.
- 034 - No Circuit/Channel Available.
- 038 - Network Out of Order.
- 039 - Permanent Frame Mode Connection OoS.
- 040 - Permanent Frame Mode Connection Oper.
- 041 - Temporary Failure.
- 042 - Switching Equipment Congestion.
- 043 - Access Information Discarded.
- 044 - Requested Circuit/Channel N/A.
- 046 - Precedence Call Blocked.
- 047 - Resource Unavailable - Unspecified.
- 049 - Quality of Service Not Available.
- 050 - Requested Facility Not Subscribed.
- 057 - Bearer Capability Not Authorized.
- 058 - Bearer Capability Not Available.
- 063 - Service or Option N/A - Unspecified.
- 065 - Bearer Capability Not Implemented.
- 066 - Channel Type Not Implemented.
- 069 - Requested Facility Not Implemented.
- 070 - Only Restricted Digital Bearer Capability Supported.
- 079 - Service or Option Not Implemented - Unspecified.
- 081 - Invalid Call Reference Value.
- 082 - Identified Channel Does Not Exist.
- 084 - Call Identity in User.

- 085 - No Call Suspended.
- 086 - Call with Requested Call Identity has Cleared.
- 088 - Incompatible Destination.
- 091 - Invalid Transit Network Selection.
- 095 - Invalid Message - Unspecified.
- 096 - Mandatory Information Element is Missing.
- 097 - Message Type Non-existent / Not Implemented.
- 098 - Message Incompatible with Call State or Message Type.
- 099 - IE/Parameter Non-existent or Not Implemented.
- 100 - Invalid Information Element Contents.
- 101 - Message Not Compatible With Call State.
- 102 - Recovery on Timer Expiry.
- 103 - Parameter Non-existent / Not Implemented - Passed On.
- 110 - Message with Unrecognized Parameter - Discarded.
- 111 - Protocol Error - Unspecified.
- 127 - Interworking - Unspecified.

A continuación se muestran ejemplos de RCs extraídos de llamadas MO y MT:

CAP GSM 09.78		MO Call	
CAMEL Components	Value	Description	
Invoke ID	5		
Operation Code	22	RC - Release Call	
Cause IE (Q.850)			
+ Extension bit	1	Not continued	
+ Coding standard	0	CCITT standard	
+ Spare	0	spare	
+ Location	1	Private network serving local user	
+ Extension bit	1	Not continued	
+ Cause value	17	User busy	

CAP GSM 09.78		MT Call	
CAMEL Components	Value	Description	
Invoke ID	5		
Operation Code	22	RC - Release Call	
Cause IE (Q.850)			
+ Extension bit	1	Not continued	
+ Coding standard	0	CCITT standard	
+ Spare	0	spare	
+ Location	1	Private network serving local user	
+ Extension bit	1	Not continued	
+ Cause value	16	Normal call clearing	

### **Establish Temporary Connection (ETC)**

Es un mensaje enviado por el SCP, para solicitarle al SSP la creación de una conexión temporal con el SRP. Dicha conexión será utilizada para reproducir un anuncio destinado al usuario o para realizarle una pregunta y esperar su respuesta. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (ETC - Establish Temporary Connection).

*Assisting SSPIP Routing Address*: Define el SRP con el que se desea establecer la conexión y sus características. Es obligatorio y puede incluir el *Correlation Id* y el *SCF Id*, en caso de que no sean proporcionados por separado.

*Correlation Id*: Define el código de correlación que permitirá la vinculación con el futuro mensaje ARI que se recibirá como respuesta. Es opcional, ya que puede estar incluido en el *Assisting SSPIP Routing Address*.

*SCF Id*: Define el SCP que procesará el futuro mensaje ARI que se recibirá como respuesta. Es opcional, ya que puede estar incluido en el *Assisting SSPIP Routing Address*.

*Service Interaction Indicators*: Define el tipo de conexión requerida, es decir, si esta será en ambos sentidos o unidireccional. Es opcional, pero en caso de no ser proporcionado se asume que la conexión será en ambos sentidos.

A continuación se muestra un ejemplo de ETC extraído de una llamada MO:

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Invoke ID	1	
Operation Code	17	ETC - Establish Temporary Connection
Assisting SSPIP routing address (Q.763)		
+ Odd/even indicator	1	odd
+ Nature of address indicator	4	international number
+ Calling party number incomplete indicator	0	complete
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	0	presentation allowed
+ Screening indicator	0	user provided, not verified
+ Address signals		099116842755
Service interaction indicators two		
+ Bothway through connection ind	1	Bothway path not required

### **Assist Request Instructions (ARI)**

Es un mensaje enviado por el SRP al SCP, tras haber sido contactado por el SSP. En él se detallan las capacidades del periférico inteligente, en base a los recursos que tiene a su disposición. En caso de que no exista un SRP externo, se puede utilizar el mensaje *Connect To Resource (CTR)* en lugar del tándem ETC/ARI. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (ARI - Assist Request Instructions).

*Correlation Id*: Define el código de correlación que permitirá la vinculación con el mensaje ETC que corresponda. Es obligatorio.

*IP Available*: Define si existe o no un periférico inteligente (SRP) anexo al SSP. Es opcional.

*IPSSP Capabilities*: Define las características del periférico inteligente (SRP) anexo al SSP. Es opcional.

A continuación se muestra un ejemplo de ARI extraído de una llamada MO:

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Invoke ID	1	
Operation Code	16	ARI - Assist Request Instructions
Correlation id (Q.763)		
+ Odd/even indicator	1	odd
+ Nature of address indicator	4	international number
+ Calling party number incomplete indicator	0	complete
+ Numbering plan indicator	1	ISDN (Telephony) numbering plan (Recommendation E.164)
+ Address presentation restricted indicator	0	presentation allowed
+ Screening indicator	0	user provided, not verified
+ Address signals		16842755
IP available	1	true
IPSSP capabilities		
+ IP routing add	1	supported
+ Voice back	0	not supported
+ Voice information via speech recognition	0	not supported
+ Voice information via voice recognition	0	not supported
+ Generation of voice announcements from text	1	supported
+ Reserved	0	0

### **Play Announcement (PA)**

Es un mensaje enviado por el SCP al SRP, con el objeto de poder interactuar con el usuario final mediante la reproducción de un anuncio. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (PA - Play Announcement).

*Information To Send*: Define el anuncio, tono o texto que se le enviará al usuario final, a través del SRP. Es obligatorio e incluye uno de los siguientes sub-argumentos:

In-band Info: Define la información a enviar por el mismo canal utilizado para la llamada de voz (anuncio). Incluye estos datos:

- Message Id (Elementary Message Id / Text / Elementary Message Ids / Variable).
- Number of Repetitions.
- Duration.
- Interval.

Display Information: Define el texto a mostrar en la pantalla del usuario final.

*Disconnect From IP Forbidden*: Define si se mantiene o no la conexión del SRP con el usuario final, una vez que la información le ha sido enviada. Es obligatorio.

*Request Announcement Complete*: Define si un mensaje SRR debe ser enviado o no al SCP, una vez que la información ha sido transmitida al usuario final. Es obligatorio.

A continuación se muestra un ejemplo de PA extraído de una llamada MO:

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Invoke ID	2	
Operation Code	47	PA - Play Announcement
Information to send		
+ Inband info		
+ Elementary message ID	1116	
+ Number of repetitions	0	
+ Duration	15	
+ Interval	0	
Disconnect from IP forbidden	1	true
Request announcement complete notification	1	true

### **Specialized Resource Report (SRR)**

Es un mensaje enviado por el SRP, para indicarle al SCP que al usuario se le ha reproducido el anuncio solicitado. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (SRR - Specialized Resource Report).

A continuación se muestra un ejemplo de SRR extraído de una llamada MO:

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Invoke ID	2	
Operation Code	49	SRR - Specialized Resource Report

### **Prompt And Collect User Information (PC)**

Es un mensaje enviado por el SCP al SRP, con el objeto de poder interactuar con el usuario final y tomar decisiones en consecuencia. Establece un canal de comunicación en ambos sentidos, donde al usuario se le reproducen ciertos anuncios (o preguntas) y este responde, mediante el uso de tonos DTMF. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (PC - Prompt And Collect User Information).

*Collected Info / Collected Digits*: Define las reglas que se aplicarán para coleccionar las respuestas provistas por el usuario final. Es obligatorio y puede incluir estos datos:

- Minimum Number of Digits.
- Maximum Number of Digits.
- End of Reply Digit.
- Cancel Digit.
- Start Digit.
- First Digit Timeout.
- Inter-digit Timeout.
- Error Treatment.
- Interruptible Announcement Indicator.

- Voice Information.
- Voice Back.

*Disconnect From IP Forbidden:* Define si se mantiene o no la conexión del SRP con el usuario final, una vez que la información le ha sido enviada. Es obligatorio.

*Information To Send:* Define el anuncio, tono o texto que se le enviará al usuario final, a través del SRP. Es obligatorio e incluye uno de los siguientes sub-argumentos:

In-band Info: Define la información a enviar por el mismo canal utilizado para la llamada de voz (pregunta). Incluye estos datos:

- Message Id (Elementary Message Id / Text / Elementary Message Ids / Variable).
- Number of Repetitions.
- Duration.
- Interval.

Display Information: Define el texto a mostrar en la pantalla del usuario final.

*Digits Response:* Define la respuesta del usuario final. Es obligatorio.

A continuación se muestra un ejemplo de PC extraído de una llamada MO:

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Invoke ID	2	
Operation Code	48	PC - Prompt And Collect User Information
Collected info		
+ Collected digits		
+ Minimum number of digits	1	
+ Maximum number of digits	1	
+ Interruptable announcement indicator	1	true
+ Voice information	0	false
+ Voice back	0	false
Disconnect from IP forbidden	1	true
Information to send		
+ Inband info		
+ Elementary message IDs [0]	1124	
+ Elementary message IDs [1]	1185	
+ Elementary message IDs [2]	1201	
+ Elementary message IDs [3]	1186	
+ Elementary message IDs [4]	1202	
+ Elementary message IDs [5]	1192	
+ Number of repetitions	0	
+ Duration	15	
+ Interval	0	
Digits Response	1	

**Disconnect Forward Connection (DFC)**

Es un mensaje enviado por el SCP, para indicarle al SRP y al SSP que la conexión temporal que se estableció ha de ser terminada. Puede contener los siguientes argumentos:

*Invoke Id*: Define el número de secuencia de la operación invocada.

*Operation Code*: Define el código asociado a la operación (DFC - Disconnect Forward Connection).

A continuación se muestra un ejemplo de DFC extraído de una llamada MO:

CAP GSM 09.78	MO Call	
CAMEL Components	Value	Description
Invoke ID	3	
Operation Code	18	DFC - Disconnect Forward Connection

## **Anexo 4: Diagnóstico de las Llamadas Móviles**

A través del tiempo se han ido aplicando mejoras continuas a las comunicaciones móviles, siempre con el objetivo de enriquecer la experiencia del usuario. Estas mejoras han estado principalmente ligadas al incremento constante de las prestaciones solicitadas por los clientes; quienes ya no sólo se conforman con realizar llamadas, sino que también desean contar con la posibilidad de enviar mensajes, hacer video llamadas, tener acceso constante a la web, etc.

Para ir alcanzando estas mejoras, los organismos internacionales que regulan la evolución de las comunicaciones móviles han ido definiendo una serie de generaciones: 1G, 2G, 3G, 4G, etc. Cada una de estas generaciones agrupa un número de requerimientos particulares y se asocia a diversos estándares para definir su implementación; tales como: AMPS (Advanced Mobile Phone System) para 1G, GSM (Global System for Mobile Communications) para 2G, UMTS (Universal Mobile Telecommunications System) para 3G, LTE Advanced (Long Term Evolution Advanced) para 4G, etc.

Las generaciones 1G, 2G y 3G se han caracterizado por el uso de una Red Inteligente, centrada en la conmutación de circuitos (circuit switching), debido a que es necesario pasar por un circuito de conmutadores predefinido (switches) para llegar a un destino determinado. Para lograr este tipo de comunicación se han ido utilizado diversos patrones que, de una u otra forma, están basados en el estándar SS7 (AMPS, GSM y UMTS) y, por ende, están vinculados a los protocolos de capa de aplicación de la familia CAMEL (CAP2, CAP3 y CAP4).

Sin embargo, a partir de la generación 4G se ha planteado la necesidad de utilizar la Red IP (en detrimento de la Red Inteligente), con el objetivo primordial de aumentar la velocidad de transmisión de datos. La particularidad de esta red es que se centra en la conmutación de paquetes (packet switching) en lugar de la de circuitos. Por lo tanto, su utilización requiere la existencia de un estándar totalmente distinto a los anteriores, descartando de esta manera el uso de SS7. Si bien existen diversas alternativas, la mayoría de las operadoras móviles se ha inclinado por el estándar LTE Advanced.

A la luz de estos acontecimientos, sería lógico pensar que las comunicaciones 4G le ponen una fecha de fin al uso de las Redes Inteligentes, SS7 y CAP2. Sin embargo, para que esto realmente ocurriese se estaría necesitando:

- Que las operadoras móviles implementen una red de telecomunicaciones totalmente nueva, incluyendo antenas, equipos y tendidos.
- Que todos los usuarios tengan terminales 4G, declarando obsoletas aquellas que sean 2G y 3G.
- Que las operadoras cambien toda la lógica de servicio que se utiliza para controlar y tarifar las llamadas y la transmisión de datos.

La dificultad técnica y económica que esto representa es la principal motivación por la que la mayoría de las operadoras móviles termina optando por soluciones de compromiso o mixtas. Por un lado, se lleva adelante el proceso de implementación de una nueva Red IP para sus servicios de datos y navegación web (GPRS). Por el otro, se mantiene la Red Inteligente actual para sus servicios de llamadas de voz y mensajería (SMS), teniendo que trabajar sólo en la vinculación de esta última con la primera.

Este tipo de solución es conocida como Circuit Switched Fallback, ya que en caso de detectarse el establecimiento de una llamada de voz, esta cae sobre la antigua red basada en conmutación de circuitos (Red Inteligente), en lugar de utilizar la de conmutación de paquetes (Red IP). De esta forma, las operadoras móviles descartan la necesidad de implementar un costoso IMS (IP Multimedia Subsystem) y, en su lugar, solo actualizan ligeramente sus conmutadores (MSC) para soportar dicha variante.

Sin embargo, para el poco probable caso de que la operadora móvil quiera contar con un IMS y, de esta forma, brindar un servicio de llamadas de voz sobre una Red IP (VoLTE), habría que contemplar lo siguiente:

- En VoLTE interactúan un IMS con un OCS (Online Charging System), equivalentes al SSP y al SCP de una Red Inteligente.
- Para dicha interacción se utiliza el protocolo de capa de aplicación Diameter Ro, en lugar de CAP2.

- Para evitar tener que cambiar la lógica del servicio ofrecido por la operadora, se requieren traductores de Diameter Ro a CAP2 y viceversa.
- Ninguna solución actual de VoLTE cubre la totalidad de los flujos de llamada que hoy en día puede ofrecer CAP2.

A modo de conclusión, se puede decir que, dentro de las comunicaciones móviles, las llamadas de voz seguirán realizándose sobre un Red Inteligente. Los elevados costos y tiempos de implementación de una Red IP para este tipo de comunicación hacen prácticamente inviable el cambio. Sin embargo, incluso cuando este se pudiese implementar, sería necesario valerse de traductores para aplicar la lógica del servicio existente. De esta manera, se puede aseverar que estándares como SS7 y protocolos como CAP2 seguirán estando vigentes por un tiempo prolongado.

## **Anexo 5: Entorno de Pruebas**

Hasta el momento se han proporcionado todos los detalles necesarios para comprender lo que ocurre detrás del establecimiento de una llamada móvil en una Red Inteligente. Por ejemplo, para que “A” y “B” se puedan comunicar es necesario que se establezca un diálogo entre los equipos de conmutación de la red (SSP) y los que contienen la lógica del servicio ofrecido por la operadora móvil (SCP), utilizando para ello los diversos mensajes del protocolo CAP2 (SS7).

También se ha mencionado que el uso de una Red Inteligente le permite a la operadora móvil proporcionar su propia lógica de servicio en una plataforma privada, separada de los equipos de conmutación. Esto posibilita la realización de cambios, en pos de alterar o ampliar la oferta comercial destinada al cliente, sin afectar en el proceso a los equipos de conmutación ni a las terminales que estos utilizan para comunicarse.

Como en todo sistema, cuando estas modificaciones ocurren, no sólo hay que contemplar las etapas destinadas a su análisis, diseño y desarrollo, sino que también es necesario tener en cuenta la estrategia que se utilizará para probar y validar que todo funcione adecuadamente. Esto último abre la puerta a un submundo gigantesco dentro del desarrollo de sistemas, el de las pruebas (testing).

Existen diversas estrategias para probar cambios en un sistema: Pruebas estáticas o dinámicas, manuales o automáticas, unitarias, funcionales, de integración, de regresión, de sistemas o de aceptación, etc. Sin embargo, en el caso del presente documento, no interesa tanto la estrategia a utilizar, sino más bien el hecho de que, tarde o temprano, habrá que llevar adelante un conjunto determinado de llamadas para validar el correcto funcionamiento de la lógica del servicio ofrecido.

Es evidente que la ejecución de pruebas mediante la realización de llamadas reales es bastante lenta y contraproducente, ni hablar si se necesita repetirlas varias veces. Surge de esta manera la imperiosa necesidad de contar con una alternativa, preferentemente basada en la simulación de llamadas, que se aplique a las diversas etapas de prueba, brindando simpleza y agilidad al proceso.

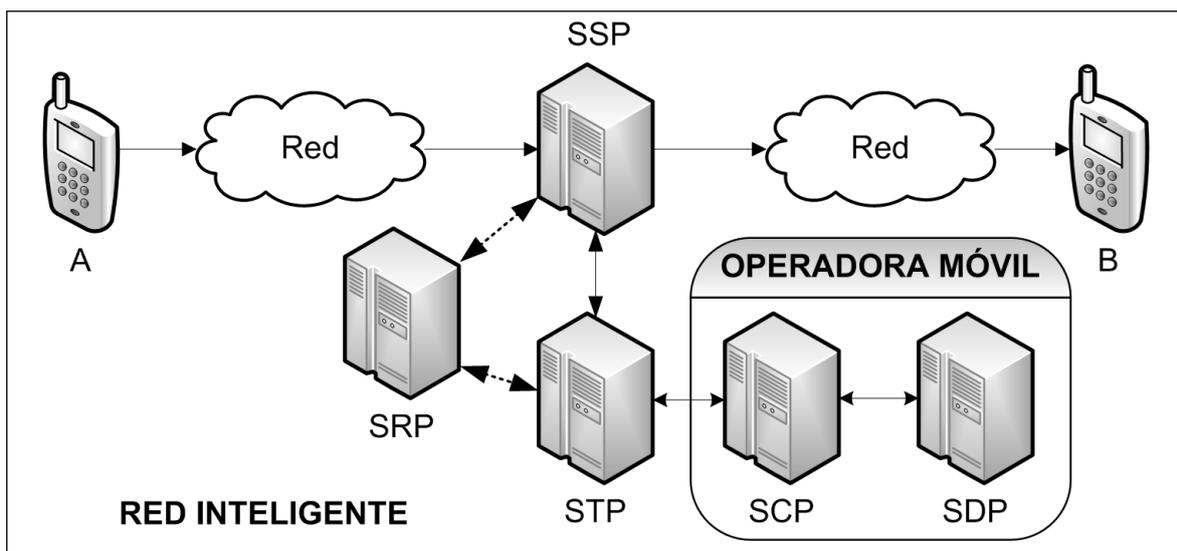
Incluso, se puede ir más lejos y enmarcar la simulación de llamadas dentro de un entorno de pruebas automatizadas, el cual permita:

- Simular los componentes de una Red Inteligente y su comunicación.
- Interactuar con los servidores que contienen la lógica del servicio y los datos en los que esta se apoya (SCP y SDP).
- Definir y configurar diferentes flujos de mensajes de señalización CAP2.
- Validar los registros y las notificaciones que se recibe tras una llamada móvil.
- Definir los datos iniciales por defecto.
- Validar los resultados esperados.
- Proporcionar diferentes casos de prueba automatizados (test cases).

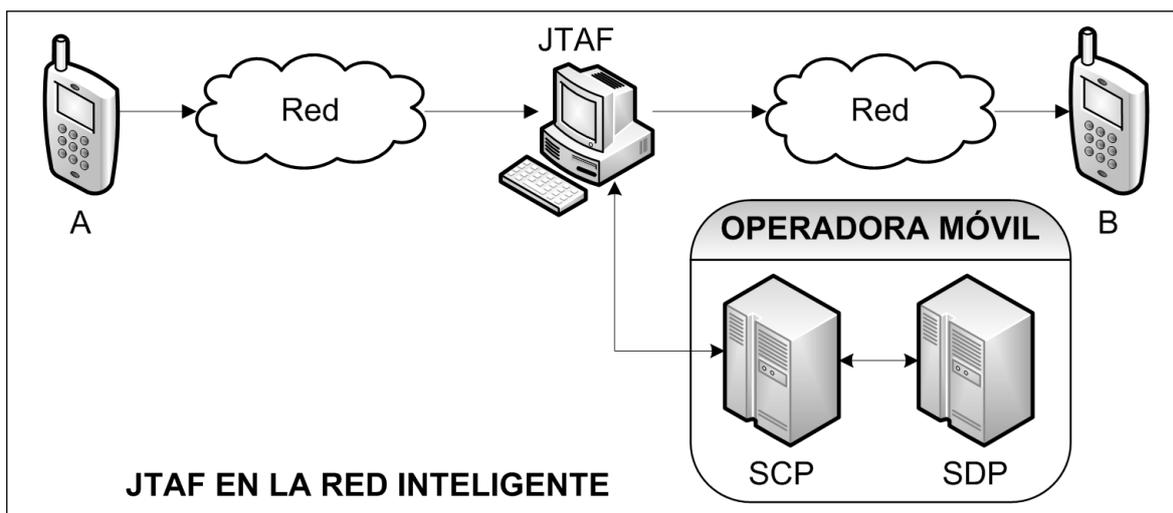
En el caso particular del sector donde desempeño mis tareas laborales, se cuenta con un entorno de pruebas que cumple, en mayor o menor medida, con las características previamente mencionadas. Este recibe el nombre de JTAF (Java Test Automation Framework), se basa en la simulación de llamadas y está implementado sobre JUnit, lo cual permite crear diferentes casos de prueba y llevar adelante una ejecución automatizada de los mismos.

### ***JTAF y la Red Inteligente***

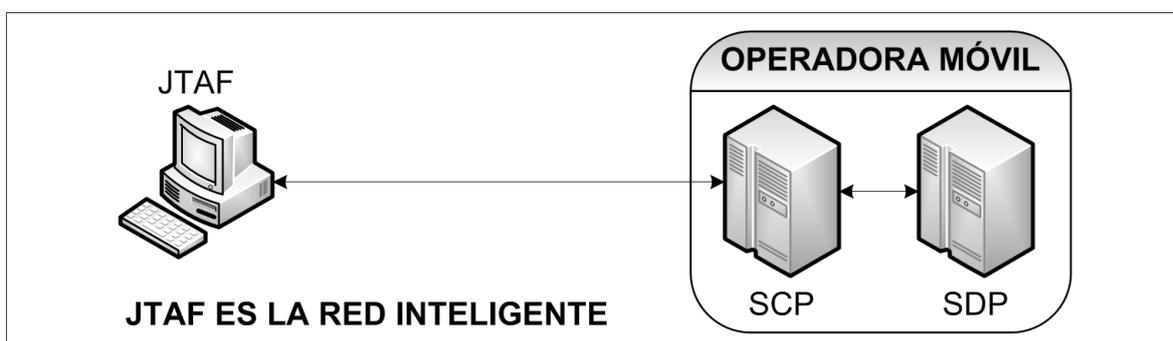
Dentro del presente documento, el siguiente diagrama es el que mejor representa los componentes de una Red Inteligente:



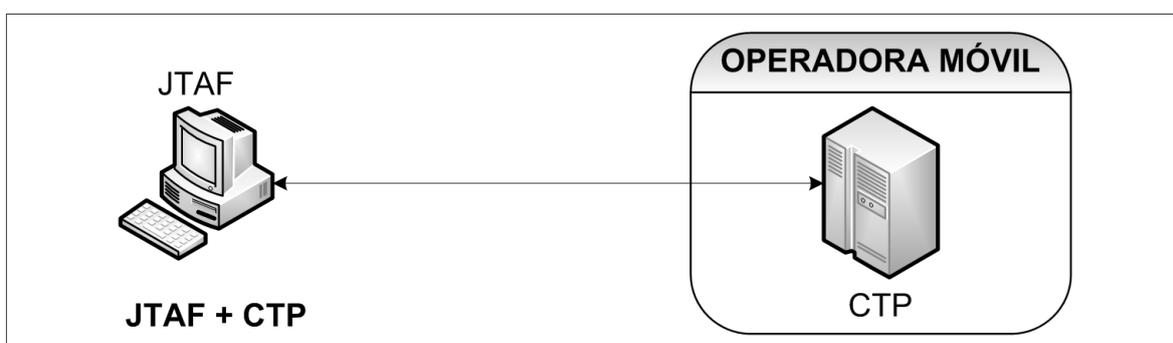
Para incorporar el uso de simulaciones, con el fin de probar cambios introducidos a la lógica de servicio (en el SCP), JTAF ha de ser integrado de la siguiente manera:



Básicamente, JTAF se encarga de simular el conjunto compuesto por SSP (Service Switching Point), SRP (Specialized Resource Point) y STP (Signal Transfer Point). Sin embargo, como el objetivo final es probar la lógica del servicio ofrecido, también es posible prescindir de las terminales móviles y de los enlaces de red involucrados en la llamada, simplificando aún más el diagrama:



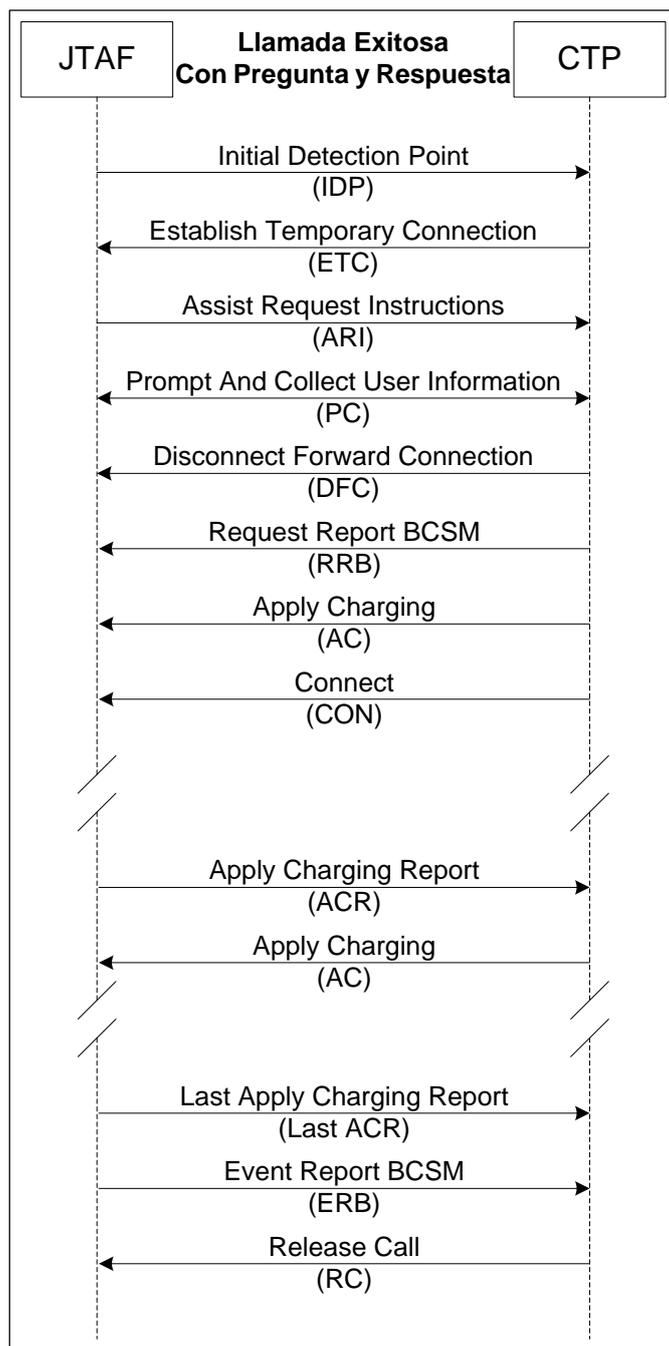
Por último, para facilitar las pruebas basadas en la simulación de llamadas dentro de un entorno controlado, es posible combinar el SCP (Service Control Point) y el SDP (Service Data Point) en una sólo equipo, conocido con el nombre de CTP (Combined Test Platform). De esta manera, se llega al siguiente diagrama final:

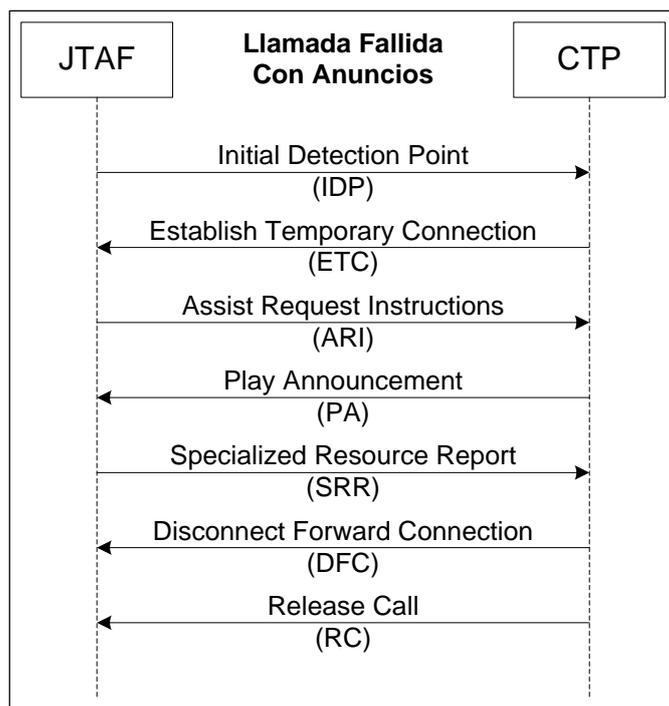


Por lo tanto, a la hora de probar con simulaciones, sólo hay que preocuparse por el diálogo que llevan adelante JTAF y el CTP. Como ya se ha mencionado, dicho diálogo se realiza mediante el intercambio de mensajes dentro de un flujo CAP2.

**JTAF y los Flujos CAP2**

Una llamada móvil está compuesta por un flujo de mensajes intercambiados entre los distintos integrantes de una Red Inteligente. Con la introducción de JTAF, estos se reducen a dos y, por ende, los diálogos se simplifican. Por ejemplo:





Por lo tanto, para simular la mayoría de los flujos de mensajes CAP2 utilizados en llamadas móviles, JTAF debe poder:

- Crear y enviar:

*Initial Detection Point (IDP).*

*Assist Request Instruction (ARI).*

*Prompt And Collect User Information - Response (PC - Response).*

*Specialized Resource Report (SRR).*

*Apply Charging Report (ACR).*

*Event Report BCSM (ERB).*

- Recibir y validar:

*Establish Temporary Connection (ETC).*

*Prompt And Collect User Information - Query (PC - Query).*

*Play Announcement (PA)*

*Disconnect Forward Connection (DFC).*

*Request Report BCSM (RRB).*

*Apply Charging (AC).*

*Connect (CON).*

*Release Call (RC).*

### **Principales Características de JTAF**

JTAF está compuesto por dos frameworks o marcos que, trabajando en conjunto, permiten la simulación de llamadas móviles dentro de un entorno de pruebas controlado: NGC (Next Generation Charging) y PPT (Prepaid Services Testing).

NGC contiene una serie de proyectos esenciales (core), responsables de proveer la mayoría de las funcionalidades de apoyo al entorno de pruebas. Se destacan:

- El acceso a bases de datos y líneas de comandos de los servidores, para la ejecución de instrucciones y captura de resultados.
- Los objetos necesarios para crear los diversos mensajes CAP2, encapsularlos siguiendo los protocolos usados en las capas inferiores (TCAP, SCCP, etc) y codificarlos siguiendo el estándar SS7.
- La máquina de estados que permite el intercambio de eventos con la plataforma de la operadora (SCP), la cual también cuenta con una propia. Dicho intercambio incluye eventos SS7, registros con los datos de la llamada, notificaciones de la plataforma, estadísticas, alarmas, etc.
- La herramienta gráfica que permite la ejecución de los diversos casos de prueba creados, junto con varios métodos útiles para la validación de resultados.

PPT contiene una serie de configuraciones comunes que hacen posible la simulación de llamadas móviles, junto con los diferentes casos de prueba que las utilizan. Se destacan:

- Los objetos necesarios para generar los comandos SQL que han de ser ejecutados en las bases de datos, principalmente para definir las configuraciones iniciales y obtener los datos resultantes, vitales para la ejecución de cada caso de prueba.
- La definición de los detalles de las llamadas a simular, como los números de "A" y "B", el IMSI, la celda utilizada, los saldos disponibles, etc.
- Los flujos de mensajes CAP2 intercambiados en cada uno de los tipos de llamada móvil a simular.

- Los datos inherentes a cada mensaje CAP2 que ha de ser creado y enviado o recibido y validado, como parte de un flujo.
- Los casos de prueba, destinados a validar el correcto funcionamiento de la lógica del servicio.

En resumen, NGC provee los medios para crear y enviar o recibir y validar los mensajes CAP2 intercambiados en una llamada móvil. A su vez, PPT usa esos medios para poder crear distintos flujos de llamada, los cuales son aplicados en los diversos casos de prueba donde son simuladas. Por lo tanto, de estos dos frameworks, PPT es el que resulta de interés para este documento, ya que es allí donde ocurre la creación de los flujos de llamada y de los casos de prueba que los utilizan.

### ***Principales Problemas de JTAF***

Si bien JTAF es utilizado constantemente como una herramienta para llevar adelante pruebas automatizadas sobre la lógica del servicio ofrecido, este presenta una serie de problemas y dificultades que impactan de manera negativa y constante sobre proceso de desarrollo e implementación de cambios.

Estos problemas están relacionados con la imposibilidad de generar nuevos casos de prueba que, no sólo garanticen su perdurabilidad en el tiempo y su reutilización, sino que también la de aquellos escenarios que ya existen en el entorno.

Cada vez que se necesita probar la lógica del servicio, se tiene que invertir una excesiva cantidad de tiempo en definir cómo crear los nuevos escenarios de prueba y los flujos de llamada que estos necesitarán para funcionar; tratando a la vez de no destruir lo existente. Como el tiempo no es un lujo del cual siempre se puede disponer, muchas veces se terminan creando pruebas que son incompletas e incluso destruyen la compatibilidad hacia atrás del entorno.

Las principales razones por los que se llega a esta situación tan dramática están ligadas a los siguientes acontecimientos:

- La definición de los datos iniciales de las llamadas a probar (los números de "A" y "B", el IMSI, la celda utilizada, los saldos disponibles, etc) está distribuida en varios archivos, generando confusiones, errores y redundancias constantes.

- La creación y utilización de flujos de mensajes CAP2, propios de una llamada, también se encuentra desperdigada por varios archivos. Existen cientos de alternativas diferentes y muy difíciles de comprender, lo cual se traduce en varios flujos de llamada inválidos, inutilizados o repetidos.
- La implementación de los distintos mensajes CAP2 es muy rígida y admite pocas variantes. De esta manera, las opciones de configuración de escenarios de prueba son muy limitadas.

Básicamente, la creación de nuevos casos de prueba se ha convertido en un proceso lento y engorroso, con una constante tendencia a generar código redundante y frágil. Los flujos de llamada creados son tan endebletes que ante la menor modificación rompen los casos de prueba que los utilizan, tanto nuevos como existentes.

Por último y como si no fuera suficiente con estas dificultades, también se puede incluir en la lista de problemas la existencia de una incompatibilidad entre JTAF y las nuevas tendencias comunicacionales, vinculadas con la implementación de Redes IP y al ofrecimiento de servicios de voz sobre las mismas (VoLTE). JTAF está diseñado para utilizar protocolos como CAP2 y, en principio, no soportaría el uso de aquellos basados en la conmutación de paquetes, como el caso de Diameter Ro.

## **Anexo 6: Diagnóstico del Entorno de Pruebas**

JTAF es un entorno de pruebas que presenta una gran robustez y flexibilidad a la hora de ejecutar pruebas sobre la lógica del servicio ofrecido por la operadora móvil. Sin embargo, cualquier virtud que se le haya atribuido a lo largo del presente documento pierde relevancia como consecuencia del círculo vicioso que se ha instalado a la hora de diseñar y crear casos de prueba perdurables en el tiempo.

Las pruebas demoran mucho tiempo en ser desarrolladas, principalmente porque cuesta mucho entender por dónde comenzar; a su vez, estas no validan (o no pueden validar) todos los cambios introducidos en el sistema, rompen los antiguos escenarios, etc. Todo esto desemboca en una merma en la calidad del proceso de validación y, por consiguiente, los desarrollos salen a producción con errores graves que pudieron haberse evitado.

Por lo tanto, no serían las nuevas tecnologías las que pueden marcar el fin de JTAF como herramienta de pruebas, sino que seguir en esta espiral de autodestrucción. Es precisamente aquí donde toma vuelo la propuesta de realizar una reingeniería del entorno de pruebas, que resalte sus virtudes y lo despoje de sus miserias.

Resulta evidente que realizar una redefinición de la manera de establecer los datos iniciales de las llamadas a simular, simplificar la creación de los flujos de mensajes CAP2 que en ellas se intercambian, flexibilizar y ampliar la configuración de estos mensajes y brindar una serie de conceptos básicos a la hora de crear pruebas, sería más que suficiente para salvar el framework. Por añadidura, esto robustecería la lógica del servicio ofrecido por la operadora móvil, lo cual se traduciría en una mejor comunicación para sus clientes y en una mayor calidad de vida para la comunidad en la que estos se encuentran inmersos.

Si bien la mencionada reingeniería del entorno de pruebas constituye el tema central del presente documento, antes de abordar sus detalles y como parte de este diagnóstico, se brindarán algunos conceptos básicos que demuestran que la otra gran amenaza, instigada por las Redes IP y las llamadas de voz montadas sobre ellas, no es tal. En realidad, se trata de una nueva oportunidad de mejora, de ampliar los horizontes de JTAF para convertirlo en una herramienta de pruebas aún más potente.

## **JTAF y la Red IP**

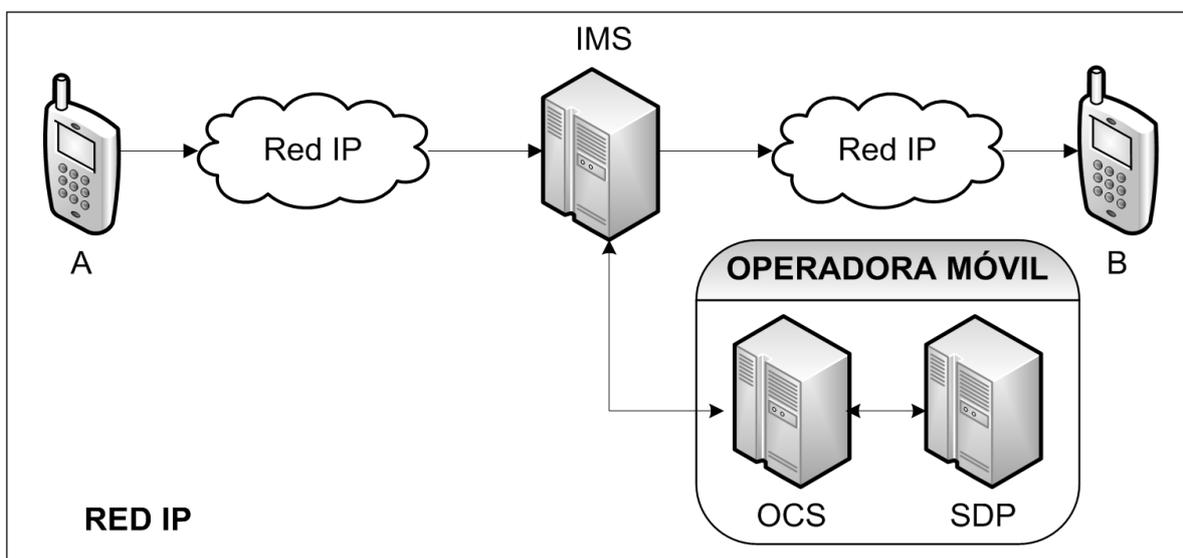
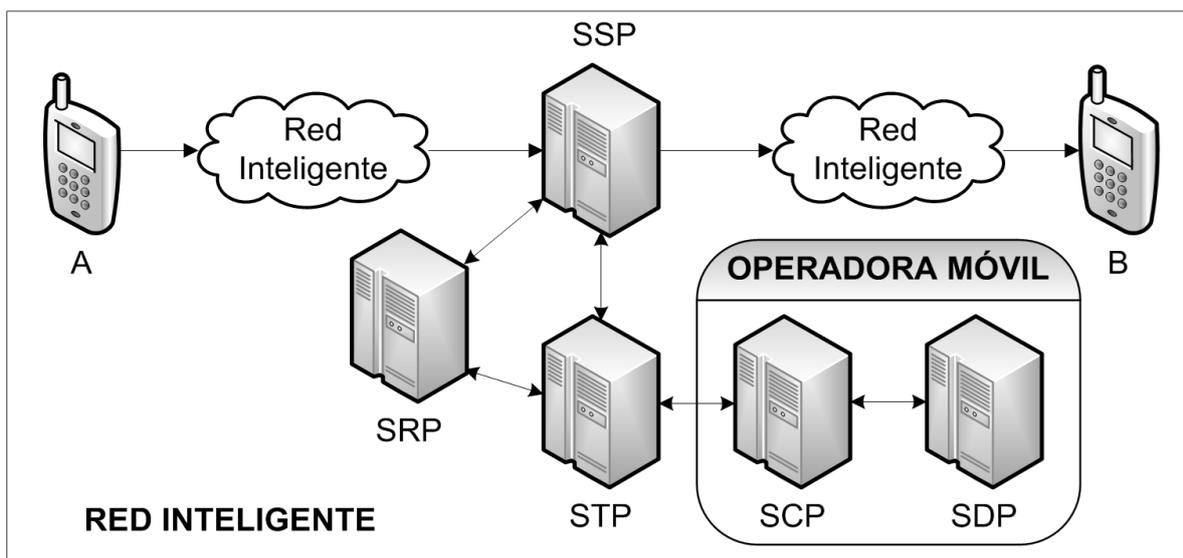
Como se ha mencionado con anterioridad en este documento, el futuro de las comunicaciones móviles yace en el uso de las Redes IP, en detrimento de las Redes Inteligentes. Un apresurado análisis de la situación marcaría este hecho como el fin de los protocolos basados en señalización y, por consiguiente, de los entornos de prueba como JTAF, ya que estos validan la lógica del servicio ofrecido mediante el uso de mensajes CAP2 (SS7).

Sin embargo y como también se ha indicado previamente, el alto costo técnico y económico que implica cambiar la red utilizada para las llamadas móviles y los servicios de mensajería ha dado lugar a soluciones de compromiso: Las Redes Inteligentes se mantienen para estos servicios y las nuevas implementaciones se destinan a la transmisión de datos y a la navegación web, vía GPRS. De esta manera, se sigue garantizando un uso continuo y prolongado de protocolos como CAP2 y, por ende, del entorno de pruebas en cuestión.

De todas formas y pese a su poca probabilidad de ocurrencia, si una operadora móvil decidiese tomar el costoso camino de soportar llamadas de voz sobre una red LTE Advanced (VoLTE), sería necesario:

- Reemplazar todo el tendido de Redes Inteligentes, las cuales se basan en la conmutación de circuitos, por otro de Redes IP, apoyado en la conmutación de paquetes.
- Reemplazar el tándem conformado por SSP (Service Switching Point), SRP (Specialized Resource Point) y STP (Signal Transfer Point), propio de los entornos basados en el uso de señalización, por un IMS (IP Multimedia Subsystem).
- Reemplazar la lógica de servicio alojada en el SCP (Service Control Point) por una nueva provista en un OCS (Online Charging System).

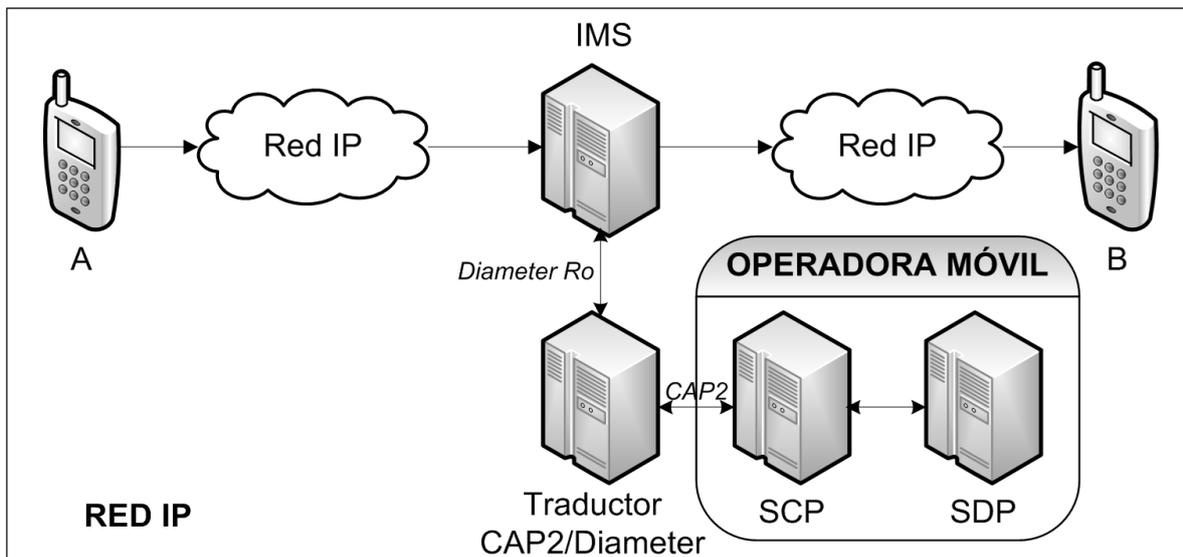
Estos reemplazos o transiciones se pueden observar claramente en los siguientes gráficos:



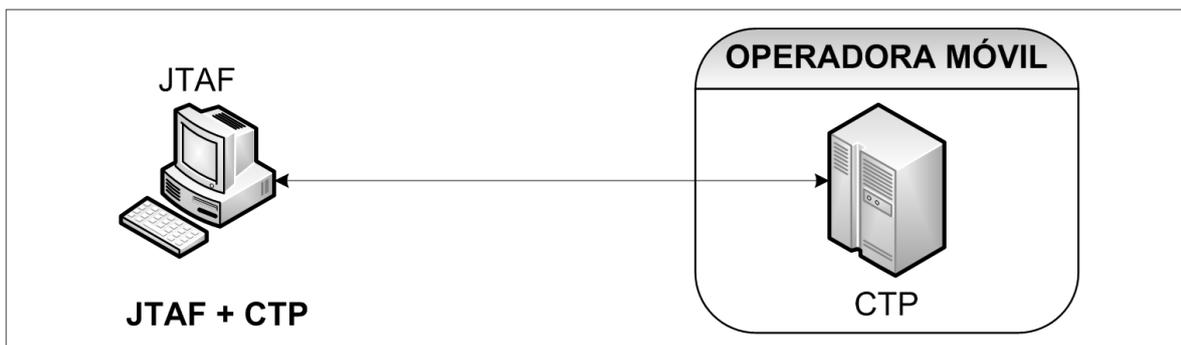
Entre los aspectos a reemplazar durante la transición de tecnologías se menciona a la lógica del servicio, alojada en el SCP. Esto se debe a que la misma está desarrollada para trabajar con el protocolo de capa de aplicación CAP2, mientras que las redes LTE Advanced (IP) trabajan con el protocolo Diameter Ro.

Sin embargo, optar por este camino implica correr un altísimo e innecesario riesgo. Cambiar la robustez y solidez que ofrece la lógica actual, construida durante años y probada en infinidad de veces, por una nueva, desconocida y con un sinnúmero de posibilidades de falla, es algo totalmente inaceptable; más aún si se tiene en cuenta la escalabilidad de cualquier problema que pudiese ocurrir en una operadora móvil que cuenta con millones de clientes.

Por lo tanto, surge la alternativa de mantener la lógica del servicio actual, pero sumándole una especie de traductor de CAP2 a Diameter Ro y viceversa. De esta manera, la representación de la Red IP se ampliaría y se vería de la siguiente manera:



Volviendo a JTAF, este sigue teniendo por objetivo probar la lógica del servicio ofrecido (en el SCP), por lo que en esta oportunidad haría las veces del tándem compuesto por el IMS y su traductor. Además, aquí también se puede prescindir de las terminales móviles y de los enlaces de red involucrados en la llamada (de tipo IP en esta oportunidad) y, a la vez, combinar el SCP y el SDP en un CTP. Por lo tanto, todo lo mencionado desemboca en el siguiente gráfico final:



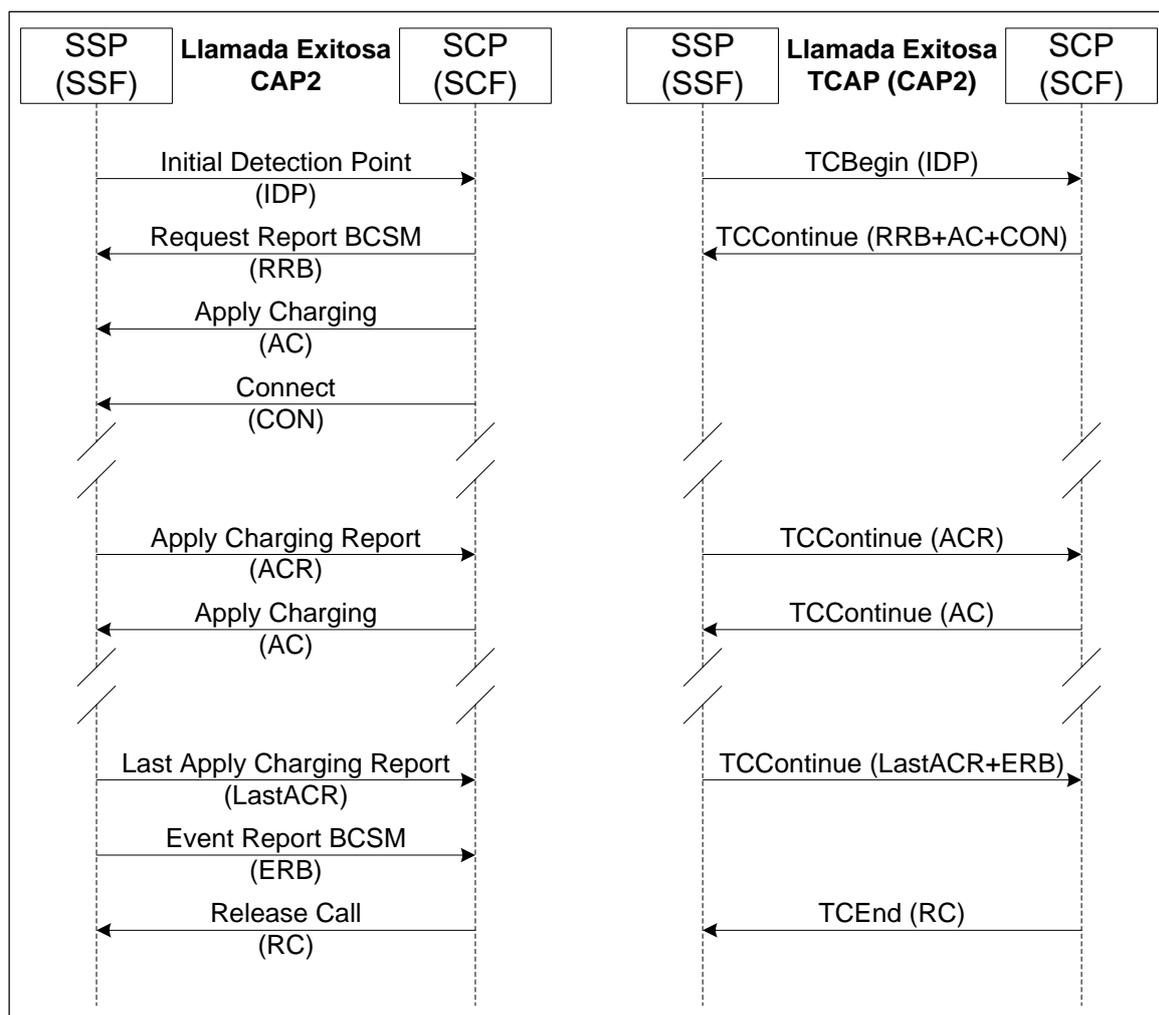
Esto implica que la implementación de una red LTE Advanced (IP) para ofrecer llamadas de voz no tiene ningún impacto directo sobre la lógica del servicio ofrecido. Como JTAF se comunica directamente con dicha lógica a la hora de probarla, esta implementación tampoco tendría un impacto sobre el mismo, garantizando la continuidad de su utilización, al menos mientras el servicio de llamadas de voz siga existiendo.

### JTAF y los Flujos Diameter Ro

Si bien la continuidad de JTAF como entorno de pruebas está garantizada, existe la remota posibilidad que una operadora móvil desee probar la lógica del servicio llegando a través del traductor. Esto implicaría la utilización de Diameter Ro en las comunicaciones y, por consiguiente, podría significar una postergación de CAP2.

Sin embargo, aún en esta situación tan extrema y poco probable se puede sostener que JTAF seguiría vigente. Simplemente sería cuestión de desarrollar un traductor interno, el cual convierta los flujos actuales de CAP2 a Diameter Ro.

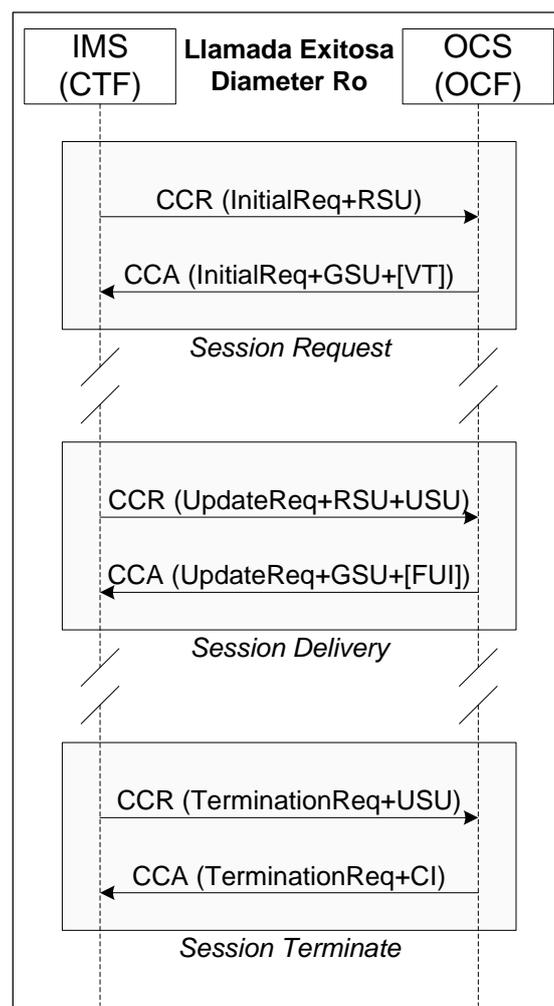
Para alcanzar este objetivo, primero hay que tener presente que Diameter Ro no puede compararse directamente con CAP2. Esto se debe a que el primero, al brindar servicios de autenticación, autorización y contabilización, puede establecer sesiones e intercambiar datos dentro de una misma transacción, mientras que el segundo no. En SS7, esta función pertenece al protocolo TCAP, sobre el cual se implementa CAP2:



Como se puede observar, TCAP utiliza tres tipos de mensajes diferentes durante una llamada:

- TCBegin: Para solicitar el inicio de una nueva transacción, asociada a una llamada de voz en particular (IDP).
- TCContinue: Para autorizar el establecimiento de la llamada e intercambiar mensajes sobre su estado (CON, RRB/ERB, AC/ACR).
- TCEnd: Para solicitar el fin de la llamada (RC).

Para el protocolo Diameter Ro, la situación es totalmente distinta a la de aquellos que se basan en SS7. En este caso, son dos los mensajes que se pueden intercambiar durante una llamada o sesión: El IMS envía CCRs (Credit Control Requests) y el OCS responde con CCAs (Credit Control Answers). Un flujo de mensajes relativamente normal puede verse reflejado en el siguiente gráfico:



Cada uno de estos mensajes contiene una serie de atributos con sus valores (AVPs), los cuales proporcionan toda la información necesaria para llevar adelante una llamada (origen, destinatario, ubicación, etc). Sin embargo, existe un atributo que tiene preponderancia sobre el resto y se lo conoce como CC-Request-Type. Su valor es el que permite determinar en qué momento de la llamada se está:

- Para solicitar el inicio de una sesión, el IMS envía un CCR con el atributo configurado como INITIAL\_REQUEST. Si todas las validaciones son correctas, el OCS autoriza el establecimiento de la sesión respondiendo con un CCA, el cual usa la misma configuración para el atributo en cuestión.
- Mientras la sesión siga activa, el IMS y el OCS intercambiarán CCRs y CCAs, pero en esta oportunidad configurando el atributo como UPDATE\_REQUEST.
- Para finalizar una sesión, el IMS envía un CCR con el atributo configurado como TERMINATION\_REQUEST. El OCS responde con un CCA con la misma configuración, finalizando el intercambio de mensajes y liberando la conexión.

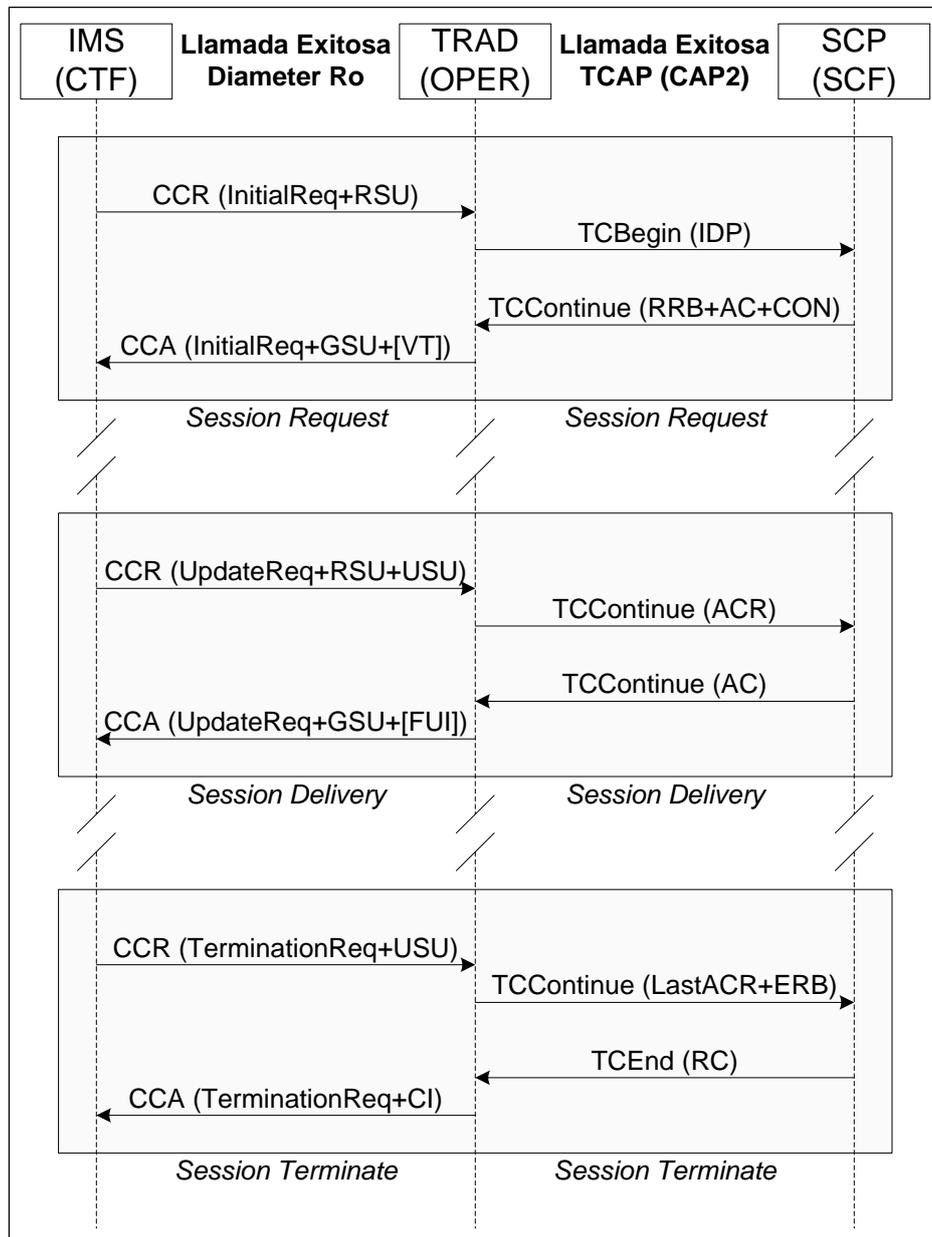
Teniendo presente todas las características definidas hasta el momento para los protocolos Diameter Ro, TCAP y CAP2, resulta sencillo poder identificar las equivalencias que existen entre ellos. Por ejemplo:

- El mensaje CCR (InitialReq) equivale al TCBegin (IDP).
- El mensaje CCA (InitialReq) equivale al TCContinue (RRB+AC+CON).
- El mensaje CCR (UpdateReq) equivale al TCContinue (ACR).
- El mensaje CCA (UpdateReq) equivale al TCContinue (AC).
- El mensaje CCR (TerminationReq) equivale al TCContinue (LastACR+ERB).
- El mensaje CCA (TerminationReq) equivale al TCEnd (RC).

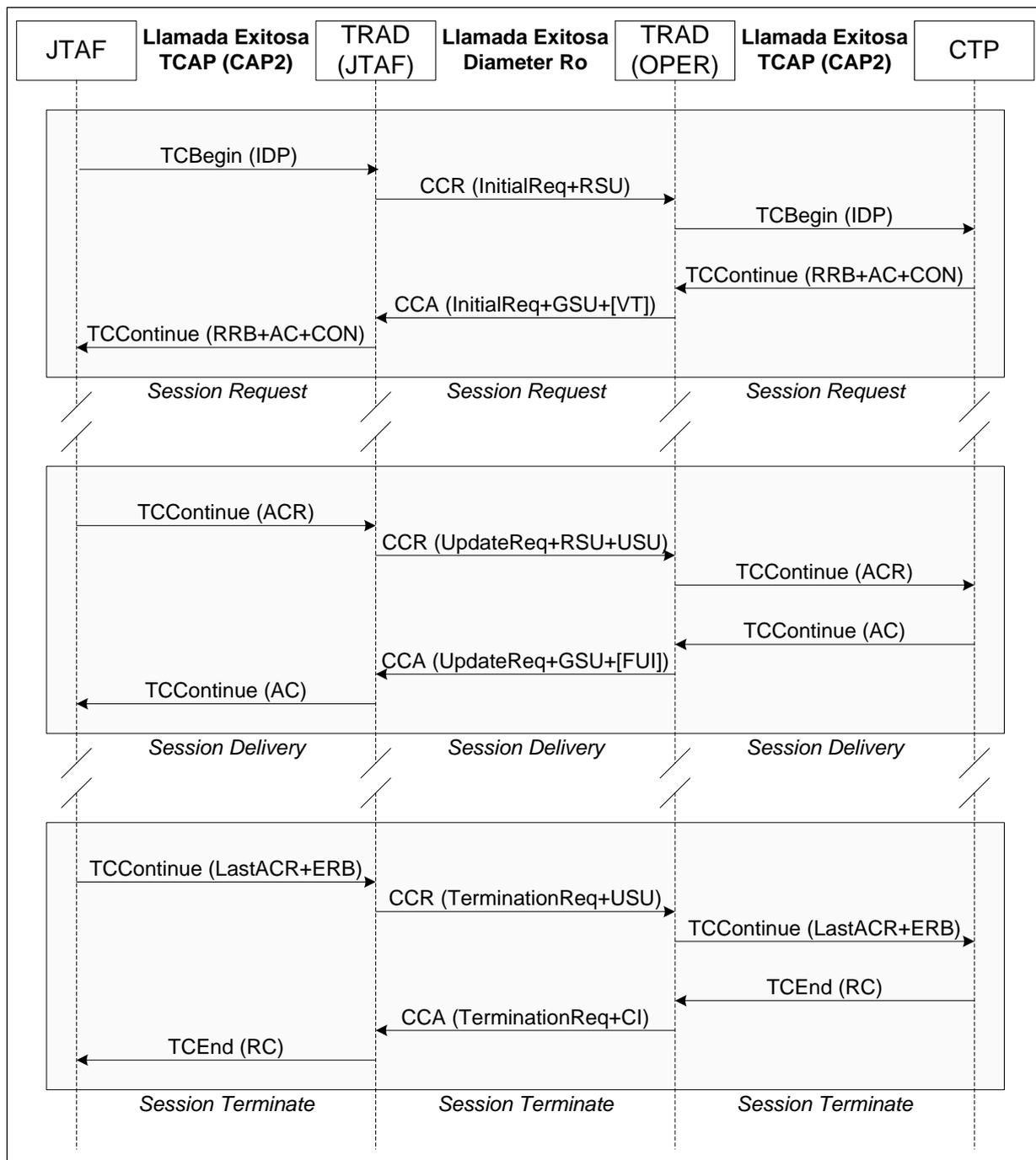
Cualquier traductor que se desee implementar deberá tomar los diversos atributos y valores de los mensajes Diameter Ro, determinar con ellos en qué momento de la llamada se está, definir cuál es el mensaje CAP2 que le corresponde, crearlo y encapsularlo usando TCAP. A su vez, deberá ser capaz de hacer el proceso inverso, extrayendo los mensajes CAP2 encapsulados con TCAP y definiendo con

ellos cual es el mensaje Diameter Ro equivalente, junto con los diversos atributos y valores que han de ser incluidos.

El siguiente gráfico refleja cómo sería el intercambio de mensajes en una Red IP que incluye un traductor entre el IMS (IP Multimedia Subsystem) y el SCP (Service Control Point):



Si a los componentes definidos en este gráfico se le incorporan JTAF y su propio traductor, se obtendría el siguiente intercambio de mensajes:



De esta manera, es posible concluir que JTAf, lejos de desaparecer o caer en lo obsolescencia, se podría convertir en una herramienta de pruebas aún más poderosa con tan solo incluir una serie de ligeros ajustes. Básicamente, esta herramienta permitiría probar la lógica de servicio no sólo en forma directa y utilizando mensajes CAP2, sino que también a través de un traductor asociado al IMS y usando mensajes Diameter Ro.

## Anexo 7: Reingeniería del Entorno de Pruebas

Como se ha descrito previamente, la necesidad de llevar adelante una reingeniería del entorno de pruebas se basó, principalmente, en aquellos problemas detectados a la hora de:

- Definir las condiciones iniciales de la llamada a simular.
- Crear y utilizar diversos flujos de llamada.
- Configurar e implementar los distintos mensajes CAP2 en un flujo.

Es precisamente sobre estos tres ejes que se pone el foco a continuación; primero describiendo como se gestaron, para luego exponer los cambios que se llevaron a cabo en pos de solucionar sus respectivas falencias.

### **La Génesis del Problema**

En un comienzo, la definición de las condiciones iniciales de una llamada se realizaba en la clase *CallInfo.java*, la cual sólo soportaba las configuraciones necesarias para llevar adelante una llamada saliente básica. A su vez, la definición de los resultados esperados, entregados por la plataforma en un CDR (Call Details Record), se hacía en la clase *CallDetails.java*, preparada para validar únicamente una llamada sin costo.

Estas dos clases servían de entrada (input) para la definición de los flujos, lo cual ocurría en la clase *BasicCallFlows.java*. Al comienzo, esta soportaba un único flujo de mensajes propios de una llamada saliente tradicional, junto con el control del CDR enviado por la plataforma.

Por último, para poder definir la secuencia de mensajes del flujo, la clase anterior se apoyaba en una serie de métodos implementados en *CAP2TestModule.java*. Estos constituían una especie de capa de abstracción, la cual ocultaba, en pos de simplificar la tarea del programador, la creación y el envío de mensajes CAP2 hacia la plataforma, junto con la recepción y la validación de sus respuestas.

Con el paso del tiempo se fue agregando código para soportar diversos tipos de llamada, lo cual implicaba la definición de nuevos datos iniciales y resultados esperados. Lamentablemente estas acciones se realizaban en forma indiscriminada,

trabajando indistintamente con las clases *CallInfo.java* y *CallDetails.java*, de acuerdo al criterio del programador de turno.

En poco tiempo, lograr un comportamiento inicial determinado pasó a ser una tarea complicada, ya que no quedaba claro ni dónde ni cómo hacerlo; incluso, cambiar algo en una de estas clases tenía altas probabilidades de romper una funcionalidad existente en la otra. El resultado fue contundente: Dos clases cargadas de definiciones y valores totalmente distintos a su propósito original, que convertían en una ordalía entender que es lo que realmente ocurría en cada una de ellas y para que servían.

En cuanto a la definición de los flujos, tampoco escapó de la suerte de las clases anteriormente citadas. Prontamente fue necesario soportar la simulación de diversos tipos de llamada que distaban mucho de aquella originalmente definida en *BasicCallFlows.java*. La necesidad de simular llamadas con cobro revertido, con anuncios y notificaciones, que fuesen desconectadas por errores de tarificación o que se conectaran con un IVR, desembocó en la existencia de una decena de nuevas clases con sus propios flujos; algunos de ellos duplicados, otros obsoletos o con errores.

Sin un claro control de estos flujos, cada vez que había que probar un nuevo tipo de llamada se terminaba optando por crear algo desde cero, dejando de lado la posibilidad de reutilizar funcionalidades existentes.

Finalmente, la capa de abstracción implementada en *CAP2TestModule.java* rápidamente demostró que, pese a sus intenciones iniciales de simplificar la elaboración de flujos de llamada, lejos de ayudar se había convertido en un problema. Esto se debía principalmente a la poca flexibilidad que ofrecían sus métodos y al hecho de que cualquier intento por cambiarlos terminaba por romper varios casos de prueba existentes.

Las consecuencias de esta problemática fueron la constante duplicación de los métodos de abstracción y la eliminación de algunas de sus validaciones que, por algún motivo difícil de detectar, habían dejado de funcionar. Básicamente, la abstracción que se quería lograr dejó de ser tal cuando estos métodos cesaron de ser reutilizables a la hora de definir un nuevo flujo.



```

813     }
814     public void setCgrUnchargedCost(int cost)
815     {
816         cgrUnchargedCost = cost;
817     }
818     public int getCgrUnchargedCost()
819     {
820         return cgrUnchargedCost;
821     }
822
823     public void setCgrFlag(int flag)
824     {
825         useContingencyCdr = flag;
826     }
827
828     public void setUseSubPeriodicCycleCDRFieldFlag(int flag)
829     {
830         useSubPeriodicCycleCDRField = flag;
831     }
832
833     public void setCgrUniqueId(String id)
834     {
835         cgrUniqueId = id;
836     }
837
838     public String getUniqueId()
839     {
840         return cgrUniqueId;
841     }
842
843     public void setProtocolType(ProtocolType type)
844     {
845         if (type == ProtocolType.INAP) Métodos obsoletos (Hace 8 años que no se usa el protocolo INAP)
846         {
847             cellName = "";
848             smPackageId = 1;
849             destinationName = "";
850             locationId = "";
851             callingPartyCategory = 0;
852             cdrBNumber = "";
853         }
854     }
855
856     public void setDurationAtPrimaryDiscountRate(int duration)
857     {
858         durationAtPrimaryDiscountRate = duration;
859     }
860
861     public int getDurationAtPrimaryDiscountRate()

```

```

243
244     public void setBNumber(String callId)
245     {
246         bNumber = callId;
247         cdrBNumber = callId;
248         this.callId = callId;
249     }
250
251     public void setBNumber(String callId, String origCalledNumber)
252     {
253         bNumber = origCalledNumber;
254         cdrBNumber = origCalledNumber;
255         this.callId = callId;
256     }
257
258     public void setBNumber(String callId, String origCalledNumber, String cdrOrigCalledNumber)
259     {
260         bNumber = origCalledNumber;
261         cdrBNumber = cdrOrigCalledNumber;
262         this.callId = callId;
263     }
264
265     public void setNumberB(String bNumber, String cdrCallId, String cdrBNumber)
266     {
267         this.bNumber = bNumber;
268         this.callId = cdrCallId;
269         this.cdrBNumber = cdrBNumber;
270     }
271
272     public void setRoutingNumber (String routingNumber)
273     {
274         this.routingNumber = routingNumber;
275     }
276
277     public String getRoutingNumber ()
278     {
279         return routingNumber;
280     }
281
282     public void setCountryCode (String countryCode)
283     {
284         this.countryCode = countryCode;
285     }
286
287     public String getCountryCode()
288     {
289         return countryCode;
290     }
291

```

Métodos duplicados

```

2243
2244 // Peridic Autorenewal
2245 protected int subPeriodicCycle = 0;
2246
2247 // CDR report
2248 protected int pkgDefId = 0;
2249 protected int pkgQuantityLeft = 0;
2250 protected int featureSubTypeId = 0;
2251 protected String pkgExpiryDate = "1970-01-01";
2252
2253 // CDR package
2254 protected String suspendeDate = "2079-06-06";
2255 protected String frozenDate = "2079-06-06";
2256 protected int balanceType = 0;
2257 protected int balanceDefId = 0;
2258 protected int finalBalanceSubscriberAccount = 0;
2259 protected int deltaBalanceSubscriberAccount = 0;
2260
2261 // Usage Call Types Counter
2262 protected String finalUsageCallTypeCounters = " ";
2263
2264 // With Immediate Bonus CDR
2265 protected boolean withImmBonusCDR = false;
2266
2267 protected boolean withLoyaltyEnabledCDR = false;
2268
2269 // MT Call
2270 protected boolean mtCall = false;
2271
2272 // Roaming Call
2273 protected boolean roamingCall = false;
2274
2275 // International Call
2276 protected boolean internationalCall = false;
2277
2278 // Call Forwarding
2279 protected boolean callForwarding = false;
2280
2281 // Video Call
2282 protected boolean videoCall = false;
2283 protected booleanivrCall = false;
2284 protected boolean casualRechargeMenu = false;
2285
2286 protected boolean insufficientFunds = false;
2287 protected int announcementId = 0;
2288
2289 protected boolean includeNotification = false;
2290 protected String notificationText = "";
2291 protected ArrayList<String> notifications = null;
    
```

**Variables que tendrían que haberse creado en CallInfo.java**

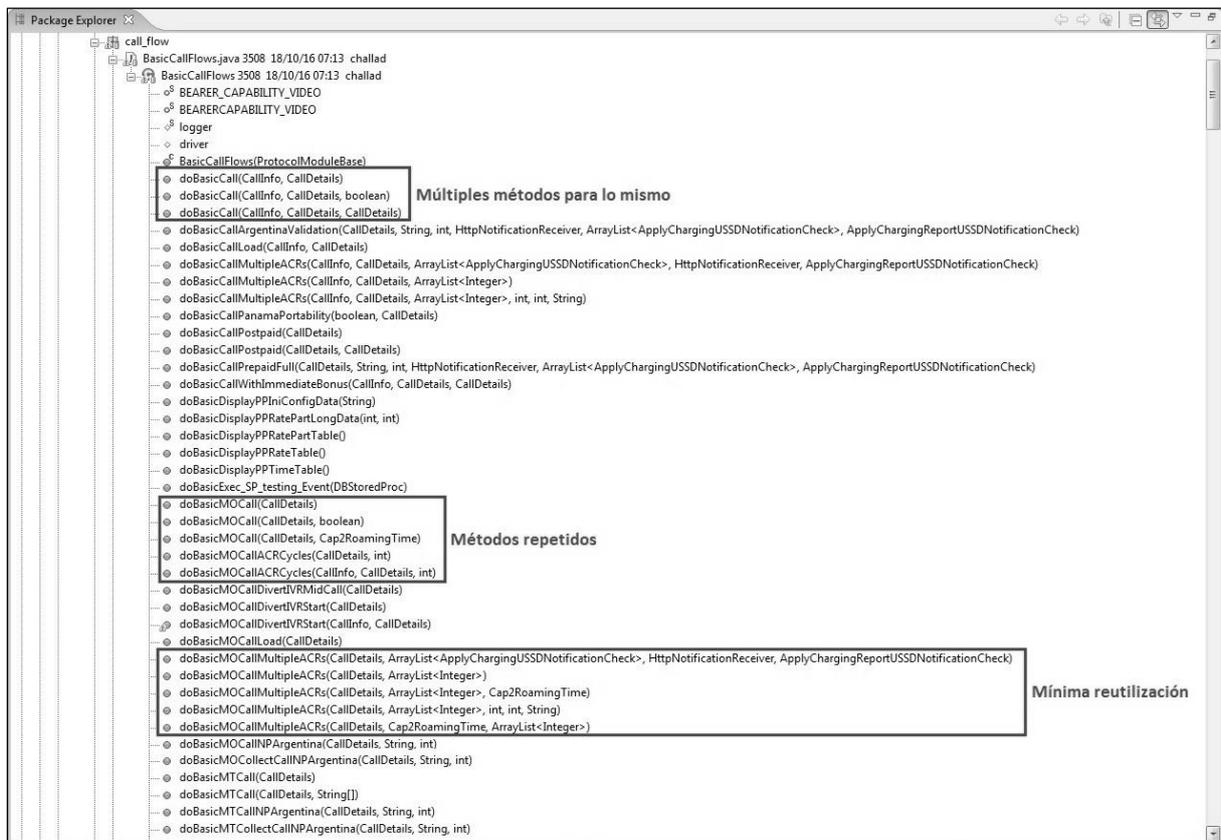
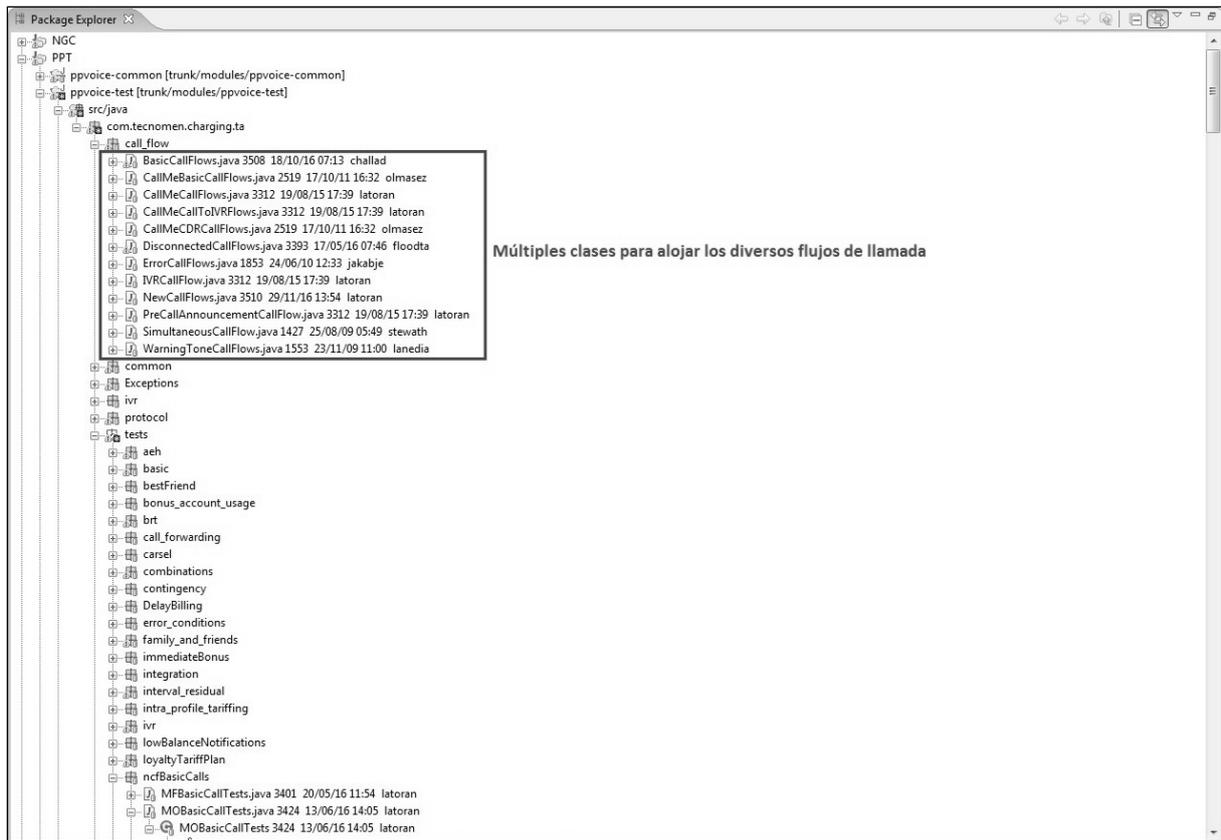
```

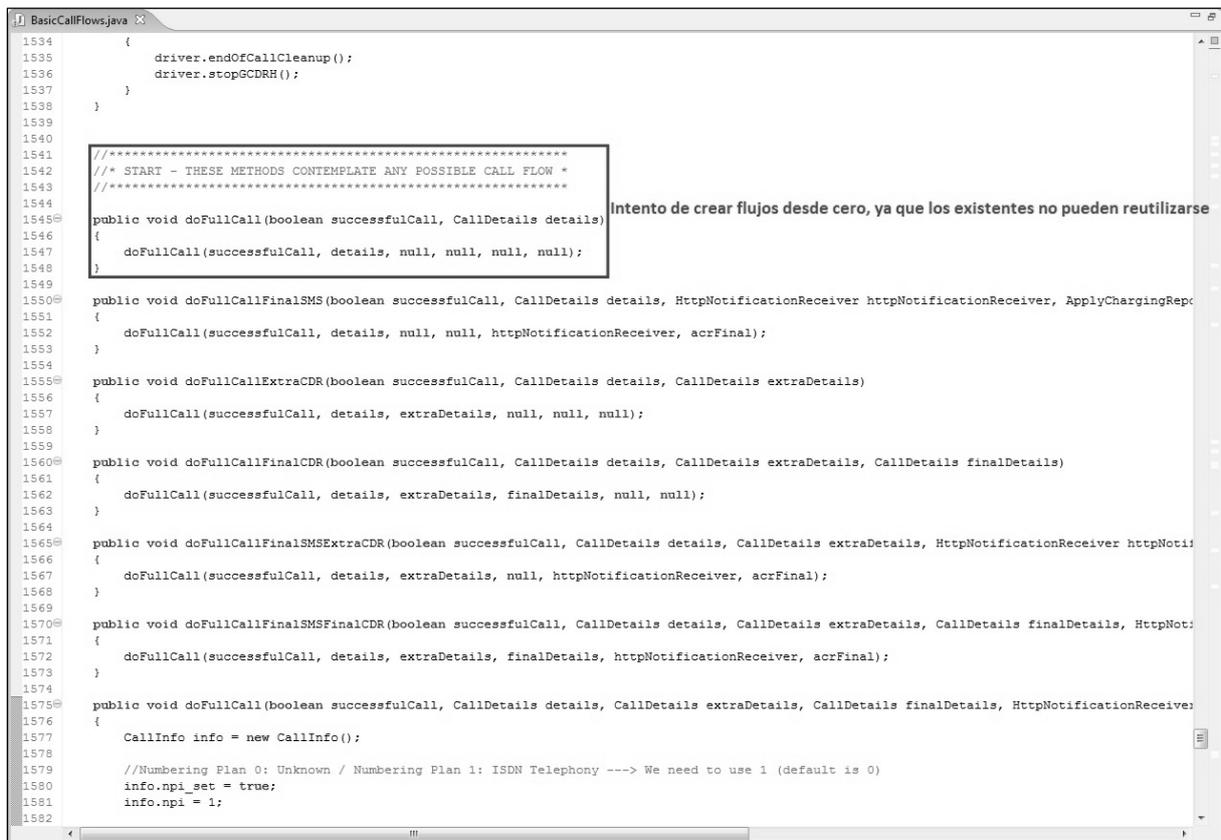
40 O_ABANDON(10),
41 TERM_ATTEMPT_AUTHORIZED(12),
42 T_CALLED_PARTY_BUSY(13),
43 T_NO_ANSWER(14),
44 T_ANSWER(15),
45 T_MIDCALL(16),
46 T_DISCONNECT(17),
47 T_ABANDON(18);
48
49 int type;
50 EventType(int i) { type = i; }
51 public int getType() {return type; }
52
53
54 public CallInfo()
55 {
56 }
57
58 /* information used for building IDP */
59 public int direction = MO_CALL;
60 public EventType idpEventType = EventType.COLLECTED_INFO;
61 public Integer callType = null;
62 public Integer callingPartysCategory = 15;
63 public boolean useLocationId = true;
64 public String locationNumber = "4443";
65 public String[] cellIdFixedLength = {"353", "12", "07400", "04898"};
66 public String vlrNumber = null;
67
68 public Calendar time = Calendar.getInstance();
69 public int timezone = 0;
70 public boolean internationalCall = false;
71 public byte[] bearerCapability = null;
72
73 public String subscriberPrefix = "061";
74
75 // for inap mt calls ?
76 public String tldnId = "000";
77
78 //TODO: figure out what this is...
79 public boolean withAbandon = false;
80 public boolean interruptModeInBusyNoanswer = false;
81
82 // For call forwarding
83 public boolean callForwarding = false;
84 public String callingPartyNumber = "6661932220"; //Why? --> Because the A Number is not validated for CF calls.
85
86 // For collect calls
87 public boolean collectCall = false;
88
89 // For Warning Tones
    
```

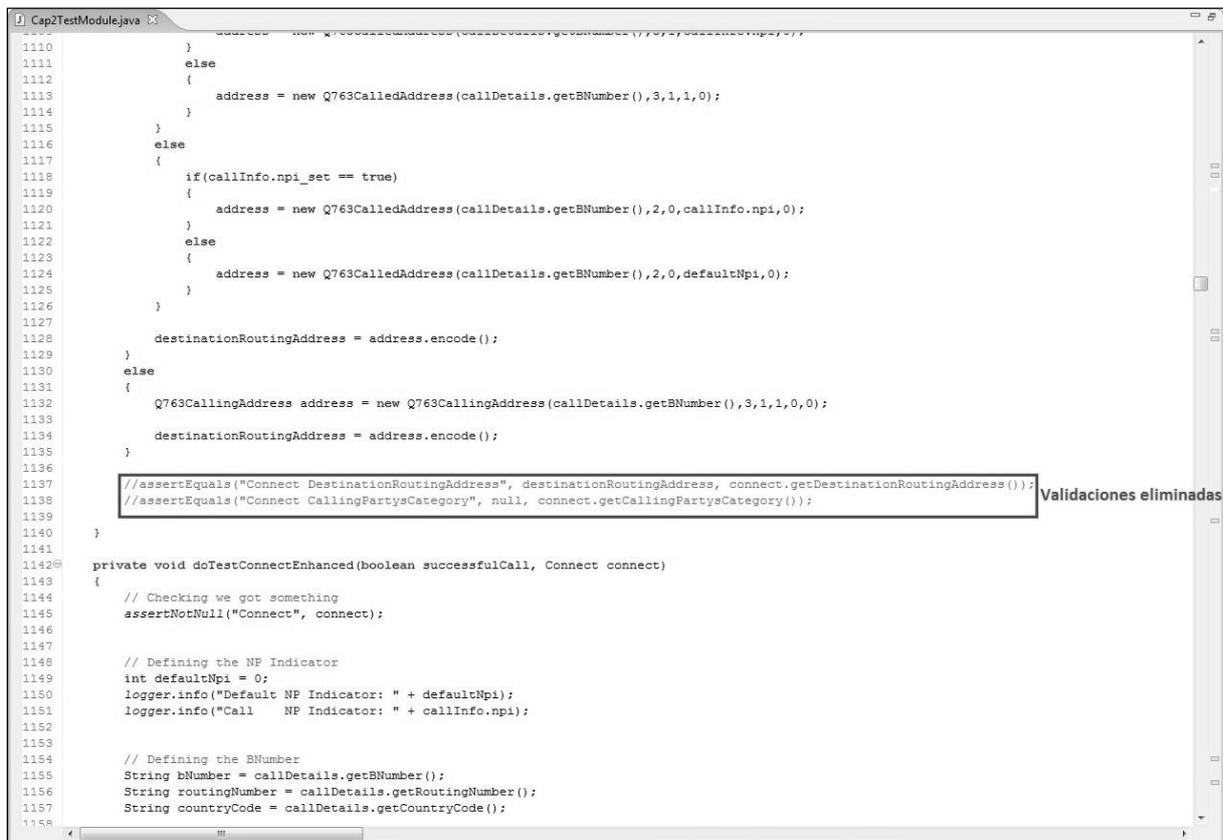
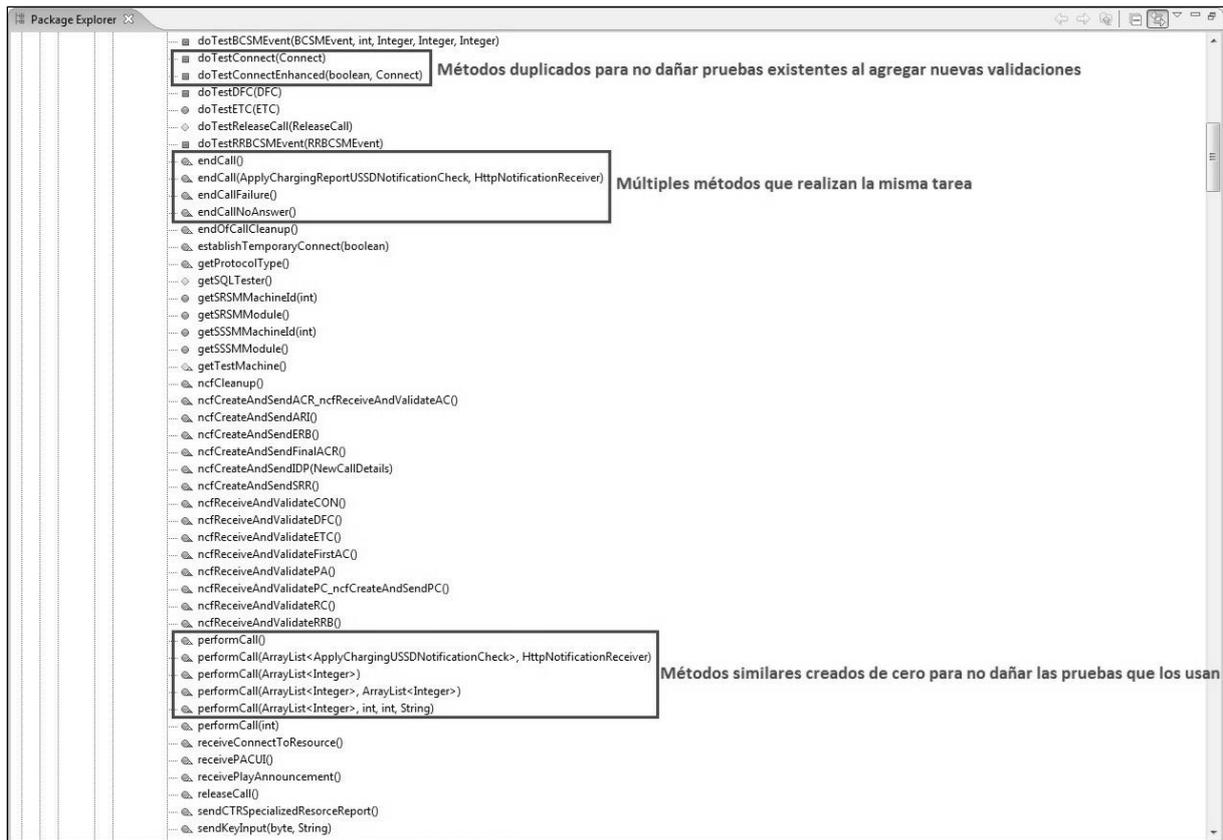
**Constructores que no hacen nada**

**Variables sin métodos Get y Set**

**Parámetros innecesarios**







```

1335     * @param etc
1336     */
1337     public void doTestETC(ETC etc)
1338     {
1339         assertNotNull("ETC not received", etc);
1340
1341         assertNotNull("ETC AssistingRoutingSSPIFAddress", etc.getAssistingSSPIFRoutingAddress());
1342         // TODO - assert other parameters
1343     }
1344
1345     /**
1346     * Create an ARI...
1347     * @param invokeID
1348     * @param assistingSSPIFRoutingAddress
1349     * @return
1350     * @throws Exception
1351     */
1352     protected ARI createARI(int invokeID, byte[] assistingSSPIFRoutingAddress) throws Exception
1353
1354     private void doTestDFC(DFC dfc)
1355     {
1356         assertNotNull("DFC not received", dfc);
1357     }
1358
1359     /**
1360     * Verify the release call isn't null
1361     * TODO: Should this verify the release cause?
1362     * @param rc
1363     */
1364     protected void doTestReleaseCall(ReleaseCall rc)
1365     {
1366         assertNotNull("ReleaseCall", rc);
1367     }
1368
1369     /**
1370     *
1371     * @param invokeID
1372     * @param digits
1373     * @return
1374     */
1375     protected ReceivedInfo createReceivedInfoDigits(byte invokeID, byte[] digits)
1376     {
1377         ReceivedInfo receivedInfo = new ReceivedInfo();
1378         receivedInfo.setInvokeID(invokeID);
1379         receivedInfo.setChoice(Cap2.ReceivedInformation_choice.DIGITS_RESPONSE);
1380         receivedInfo.setDigitsResponse(digits);
1381         return receivedInfo;
1382     }
1383
1384     /**
1385     * Creates an IDP based on the class's CallInfo and CallDetails.
1386     * @return
1387     * @throws Exception
1388     */
1389     protected IDP createIDP() throws Exception
1390     {
1391         IDP idp = new IDP();
1392
1393         idp.setInvokeID((byte)0);
1394         idp.setServiceKey(1);
1395
1396         // Adding check here for the case when no A# in IDP
1397         if (callDetails.getANumber() != "")
1398         {
1399             String callingNumber = callInfo.callForwarding? callInfo.callingPartyNumber : callDetails.getANumber();
1400             Q763CallingAddress callingPartyNumber = new Q763CallingAddress(callingNumber,3,1,1,0,0);
1401             idp.setCallingPartyNumber(callingPartyNumber.encode());
1402         }
1403
1404         int i = callInfo.internationalCall? 4 : 3;
1405
1406         // Adding check here for the case when no B# in IDP
1407         if (callDetails.getBNumber() != "")
1408         {
1409             Q763CalledAddress calledPartyNumber = new Q763CalledAddress(callDetails.getBNumber(),i,1,1,0);
1410             idp.setCalledPartyNumber(calledPartyNumber.encode());
1411
1412             int type = callInfo.internationalCall? 1 : 0;
1413             Gsm0408BcdNumber calledPartyBCDNumber = new Gsm0408BcdNumber(callDetails.getBNumber(),1,type);
1414             idp.setCalledPartyBCDNumber(calledPartyBCDNumber.encode());
1415         }
1416         idp.setEventTypeBCSM(cap2EventType(callInfo.idpEventType));
1417
1418         Imsi imsi = new Imsi(callDetails.getImsi());
1419         idp.setImsi(imsi.encode());
1420
1421         if (callInfo.useLocationId)
1422         {
1423             Q763CallingAddress locationNumber = new Q763CallingAddress(callInfo.locationNumber,4,0,1,0,0);
1424             idp.setLocationNumber(locationNumber.encode());
1425         }
1426
1427         LocationInformation locationInformation = new LocationInformation();
1428
1429         if (callInfo.useLocationId)
1430         {
1431             locationInformation.setCallIDCIChoice(Cap2.CallIDCIChoice.CELL_ID_FIXED_LEMGTN);
1432         }
1433     }
    
```

Validaciones mínimas

Validaciones inexistentes

```

715
716     /**
717     * Creates an IDP based on the class's CallInfo and CallDetails.
718     * @return
719     * @throws Exception
720     */
721     protected IDP createIDP() throws Exception
722     {
723         IDP idp = new IDP();
724
725         idp.setInvokeID((byte)0);
726         idp.setServiceKey(1);
727
728         // Adding check here for the case when no A# in IDP
729         if (callDetails.getANumber() != "")
730         {
731             String callingNumber = callInfo.callForwarding? callInfo.callingPartyNumber : callDetails.getANumber();
732             Q763CallingAddress callingPartyNumber = new Q763CallingAddress(callingNumber,3,1,1,0,0);
733             idp.setCallingPartyNumber(callingPartyNumber.encode());
734         }
735
736         int i = callInfo.internationalCall? 4 : 3;
737
738         // Adding check here for the case when no B# in IDP
739         if (callDetails.getBNumber() != "")
740         {
741             Q763CalledAddress calledPartyNumber = new Q763CalledAddress(callDetails.getBNumber(),i,1,1,0);
742             idp.setCalledPartyNumber(calledPartyNumber.encode());
743
744             int type = callInfo.internationalCall? 1 : 0;
745             Gsm0408BcdNumber calledPartyBCDNumber = new Gsm0408BcdNumber(callDetails.getBNumber(),1,type);
746             idp.setCalledPartyBCDNumber(calledPartyBCDNumber.encode());
747         }
748         idp.setEventTypeBCSM(cap2EventType(callInfo.idpEventType));
749
750         Imsi imsi = new Imsi(callDetails.getImsi());
751         idp.setImsi(imsi.encode());
752
753         if (callInfo.useLocationId)
754         {
755             Q763CallingAddress locationNumber = new Q763CallingAddress(callInfo.locationNumber,4,0,1,0,0);
756             idp.setLocationNumber(locationNumber.encode());
757         }
758
759         LocationInformation locationInformation = new LocationInformation();
760
761         if (callInfo.useLocationId)
762         {
763             locationInformation.setCallIDCIChoice(Cap2.CallIDCIChoice.CELL_ID_FIXED_LEMGTN);
764         }
765     }
    
```

Código difícil de entender

El uso de variables de CallInfo.java y métodos de CallDetails.java genera confusiones

### **Definición de las Condiciones Iniciales**

Lo primero que se decidió corregir fue la definición de los datos de una llamada, tanto iniciales como finales, lo cual pasó a ocurrir en una única clase, a la que se llamó: *NewCallDetails.java*.

Dicha clase tiene por objetivo contener aquellas variables que van a definir las características de la llamada y, a la vez, son incluidas en el CDR enviado al final de la misma. Entre ellas se encuentran el número de suscriptor y su IMSI, la celda que toma la llamada, la duración de la misma, el saldo inicial y final, la tarifa que se aplica, etc.

También se definen, en otro grupo, aquellas variables que sirven de soporte a la llamada simulada. Estas ofrecen la posibilidad de contar con CDRs adicionales, validar anuncios y notificaciones al finalizar la llamada, soportar la portabilidad numérica, etc.

Cada una de estas variables tiene asociados sus métodos get y set, los cuales constituyen la única forma en que estas pueden ser manipuladas desde los casos de prueba. A su vez, con estas mismas variables se define la estructura del CDR esperado, el cual será comparado contra el que llegue de la plataforma para validar si el resultado obtenido ha sido o no el esperado.

Finalmente, es en esta clase donde se deben listar cada uno de los tipos de llamada que serán soportados por el entorno de pruebas.

### **Creación y Utilización de Flujos CAP2**

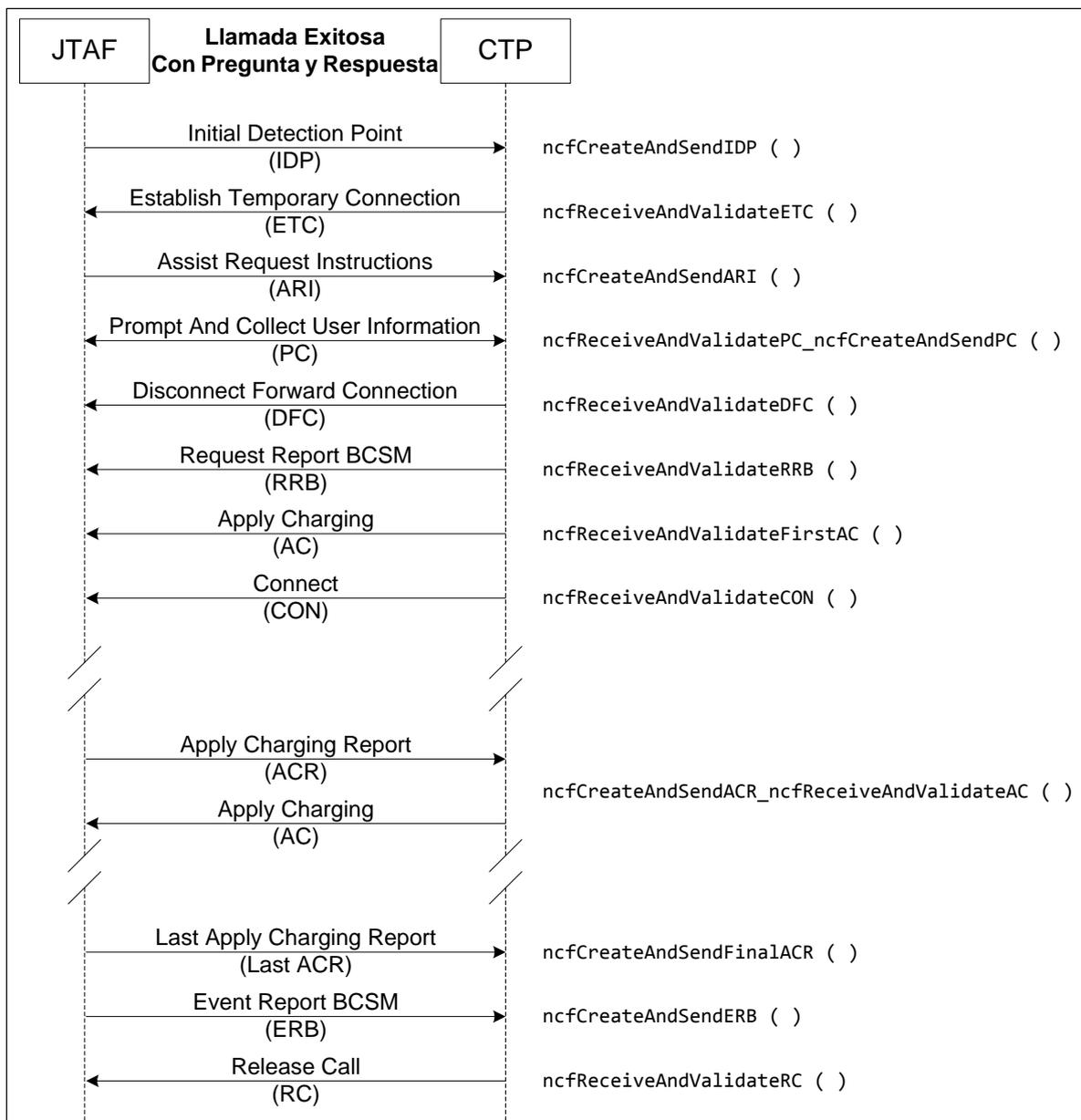
El segundo aspecto que se tuvo en cuenta durante la reingeniería fue la eliminación de la proliferación de flujos de llamada por distintos archivos. Para ello se decidió crear la clase *NewCallFlows.java* e ir por la ambiciosa solución de contar con un solo método, el cual contase con la flexibilidad suficiente para soportar todos los flujos CAP2.

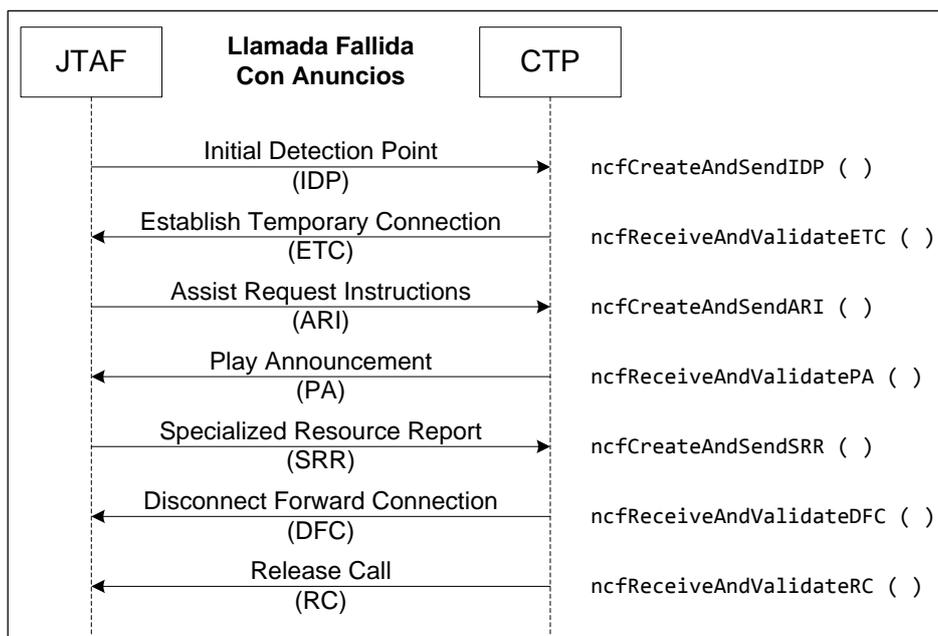
Usando como base el conocimiento adquirido durante el estudio del protocolo en cuestión, se logró determinar la existencia de un patrón común a todos los flujos de llamada disponibles. Este patrón está basado en una situación extrema donde todos los mensajes CAP2 son intercambiados durante la llamada.

Por lo tanto, se procedió primero a proporcionar un método separado para cada uno de los mensajes que se intercambiarían en esta situación extrema, para luego ponerlos en el orden correspondiente. Finalmente, por cada uno de ellos se definieron los diversos tipos de llamada que los invocarían o que los ignorarían.

De esta manera, cada vez que se define un nuevo tipo de llamada en *NewCallDetails.java*, se requiere conocer los mensajes intercambiados. Hecho esto, sólo basta con ir a *NewCallFlows.java* y, por cada uno de los métodos invocados, definir si el tipo de llamada en cuestión lo incluirá o no.

Siguiendo los flujos de mensajes más representativos del presente documento, se listan a continuación los métodos que corresponden a cada uno de ellos:





### Configuración e Implementación de Mensajes CAP2

La implementación de los métodos que representan a cada uno de los mensajes CAP2 se llevó adelante en la clase *Cap2TestModule.java*. De esta forma, se buscó reutilizar toda la lógica del entorno que conecta los casos de prueba y sus flujos con las máquinas de estados, definidas en el framework NGC (Next Generation Charging).

En cuanto a los métodos que allí existían, al formar parte de un intento fallido por brindar una capa de abstracción al programador, se procedió a usarlos sólo como una fuente de consulta inicial, para luego declarados obsoletos.

Nuevamente, basado en los conocimientos adquiridos durante el estudio del protocolo CAP2, se procedió a distinguir, por cada mensaje, los atributos que podían llegar a variar en base al tipo de llamada de aquellos que no. Luego se procedió a asociar los distintos valores, configuraciones o parametrizaciones que dichos atributos pueden tener para los diversos tipos de llamada. De esta manera, se buscó poder brindarle al programador menos experimentado una capa de abstracción, que le evita tener que profundizar en su conocimiento del protocolo.

A la vez, como se realizó un trabajo minucioso y bien detallado en pos de no dejar ningún aspecto librado al azar, se proporcionaron en el mismo código una serie de explicaciones básicas para cada atributo. De esta forma, se intentó brindar también la flexibilidad con la que un desarrollador experimentado puede llegar a querer contar, a la hora de configurar un flujo con un alto nivel de complejidad.



```

1 package com.tecnomen.charging.ta.common;
2
3 @import static com.tecnomen.test.Assert.fail;[]
4
5
6 -----*/
7 /* NEW CALL DETAILS (NewCallDetails.java)
8 /*-----*/
9
10
11
12
13
14 public class NewCallDetails
15 {
16     /*----- CONSTRUCTORS -----*/
17
18     // Basic Constructors
19     public NewCallDetails()
20     {
21     }
22
23     public NewCallDetails(Prepaid.Version version)
24     {
25         this.version = version;
26     }
27
28     public NewCallDetails(NewCallDetails object)
29     {
30         this.accountNumber = object.accountNumber;
31         this.acDuration = object.acDuration;
32         this.acrTimeoutDuration = object.acrTimeoutDuration;
33         this.activeReservationCount = object.activeReservationCount;
34         this.airCost = object.airCost;
35         this.airTimeCharged = object.airTimeCharged;
36         this.aNumber = object.aNumber;
37         this.atiStatusCodeTCAP = object.atiStatusCodeTCAP;
38         this.balanceDeltas = object.balanceDeltas;
39         this.balanceReservations = object.balanceReservations;
40         this.bNumber = object.bNumber;
41         this.bNumberPrefixCON = object.bNumberPrefixCON;
42         this.bNumberPrefixIDP = object.bNumberPrefixIDP;
43         this.bonusAmount = object.bonusAmount;
44         this.bonusPercentage = object.bonusPercentage;
45         this.bonusType = object.bonusType;
46         this.callDuration = object.callDuration;
47         this.callDurationCDR = object.callDurationCDR;
48         this.callForwardingReason = object.callForwardingReason;
49         this.callId = object.callId;
50         this.callingPartyCategory = object.callingPartyCategory;
51         this.callOrigId = object.callOrigId;
52         this.callReference = object.callReference;
53         this.callType = object.callType;
54         this.callTypeId = object.callTypeId;

```

Código claro y ordenado, en una sola clase

```

219 }
220
221
222
223 /*----- ENUMS -----*/
224
225 // Call Types
226 public enum CallType
227 {
228     // ncfBasicCalls
229     MO_REGULAR_ERB_O_DISCONNECT_LEG_1(1),
230     MO_REGULAR_ERB_O_DISCONNECT_LEG_2(2),
231     MO_REGULAR_ACR_INACTIVE_DURATION_0(3),
232     MO_REGULAR_IDP_ERROR(4),
233     MO_REGULAR_SUB_NO_FUNDS(5),
234     MO_REGULAR_SUB_NO_FUNDS_ANNOUNCEMENT(6),
235     MO_REGULAR_ACR_TIMEOUT(7),
236     MO_REGULAR_MAX_TIMEOUT(8),
237     MO_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1(9),
238     MO_ROAMING_ERB_O_DISCONNECT_LEG_1(10),
239     MO_ROAMING_INTER_ERB_O_DISCONNECT_LEG_1(11),
240     MO_ROAMING_PANAMA_ERB_O_DISCONNECT_LEG_1(12),
241     MT_REGULAR_ERB_T_DISCONNECT_LEG_1(13),
242     MT_REGULAR_ERB_T_DISCONNECT_LEG_2(14),
243     MT_REGULAR_ERB_T_NO_ANSWER_LEG_2(15),
244     MT_REGULAR_ERB_T_CALLED_PARTY_BUSY_LEG_2(16),
245     MT_REGULAR_ACR_INACTIVE_DURATION_0(17),
246     MT_REGULAR_IDP_ERROR(18),
247     MT_REGULAR_SUB_NO_FUNDS(19),
248     MT_REGULAR_ACR_TIMEOUT(20),
249     MT_REGULAR_MAX_TIMEOUT(21),
250     MT_ROAMING_ERB_T_DISCONNECT_LEG_1(22),
251     MF_REGULAR_ERB_O_DISCONNECT_LEG_1(23),
252     MF_REGULAR_ERB_O_DISCONNECT_LEG_2(24),
253     MF_REGULAR_ACR_INACTIVE_DURATION_0(25),
254     MF_REGULAR_IDP_ERROR(26),
255     MF_REGULAR_SUB_NO_FUNDS(27),
256     MF_REGULAR_ACR_TIMEOUT(28),
257     MF_REGULAR_MAX_TIMEOUT(29),
258     MF_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1(30),
259     MF_ROAMING_ERB_O_DISCONNECT_LEG_1(31),
260     MF_ROAMING_INTER_ERB_O_DISCONNECT_LEG_1(32),
261     // ncfEmergencyCalls
262     MO_EMERGENCY_IDLE(33),
263     MO_EMERGENCY_IDP_ERROR(34),
264     MO_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_1(35),
265     MO_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_2(36),
266     MO_CDR_EMERGENCY_ACR_INACTIVE_DURATION_0(37),
267     MO_CDR_EMERGENCY_IDP_ERROR(38),

```

Múltiples tipos de llamada soportados

```

1201
1202 }
1203
1204
1205 /*----- VARIABLES -----*/
1206
1207 // General Variables
1208 protected Prepaid.Version version = Prepaid.Version.PP4110;
1209
1210
1211 // CDR Variables
1212 protected VoiceField.Transaction_type.Id transactionType = VoiceField.Transaction_type.Id.TINcdr_OutCall;
1213 protected int externalTransactionType = 0;
1214 protected VoiceField.Service_type.Id serviceType = VoiceField.Service_type.Id.Prepaid;
1215 protected int srcType = 1;
1216 protected String cdrDate = Field.Tag.IGNORED.toString();
1217 protected String callReference = Field.Tag.IGNORED.toString();
1218 protected String subId = "061932220";
1219 protected String imsi = "678900";
1220 protected String srrId = "20";
1221 protected String accountNumber = "5544";
1222 protected int subclass = 0;
1223 protected int profileId = 0;
1224 protected int expiryType = 0;
1225 protected int callingPartyCategory = 10;
1226 protected String callId = "35361932370";
1227 protected String originalNumber = "35361932370";
1228 protected String redirectingNumber = "";
1229 protected VoiceField.Call_forwarding_reason.Id callForwardingReason = VoiceField.Call_forwarding_reason.Id.Unknown;
1230 protected String homeMscAddressCDR = "35361900000";
1231 protected String visitedMscAddressCDR = "35361900000";
1232 protected int operatorId = -1;
1233 protected int callOrigId = 0;
1234 protected int callTypeId = 2;
1235 protected int callDurationCDR = 300;
1236 protected int airTimeCharged = 300;
1237 protected int tollTimeCharged = 300;
1238 protected int itelTimeCharged = 300;
1239 protected int tariffId = 0;
1240 protected int periodicTariffId = 0;
1241 protected int subPeriodicCycle = 0;
1242 protected VoiceField.Tariffing_method.Id tariffingMethod = VoiceField.Tariffing_method.Id.Normal;
1243 protected String locationId = "2720010740004898";
1244 protected String cellName = "TRL-ROM";
1245 protected String destinationName = "TRL-NAT";
1246 protected String timebandName = "ROM-NAT-T";
1247 protected String rateName = "ROM-NAT-R";
1248 protected int tzAdjustment = 0;
1249 protected int airCost = 50000;
1250

```

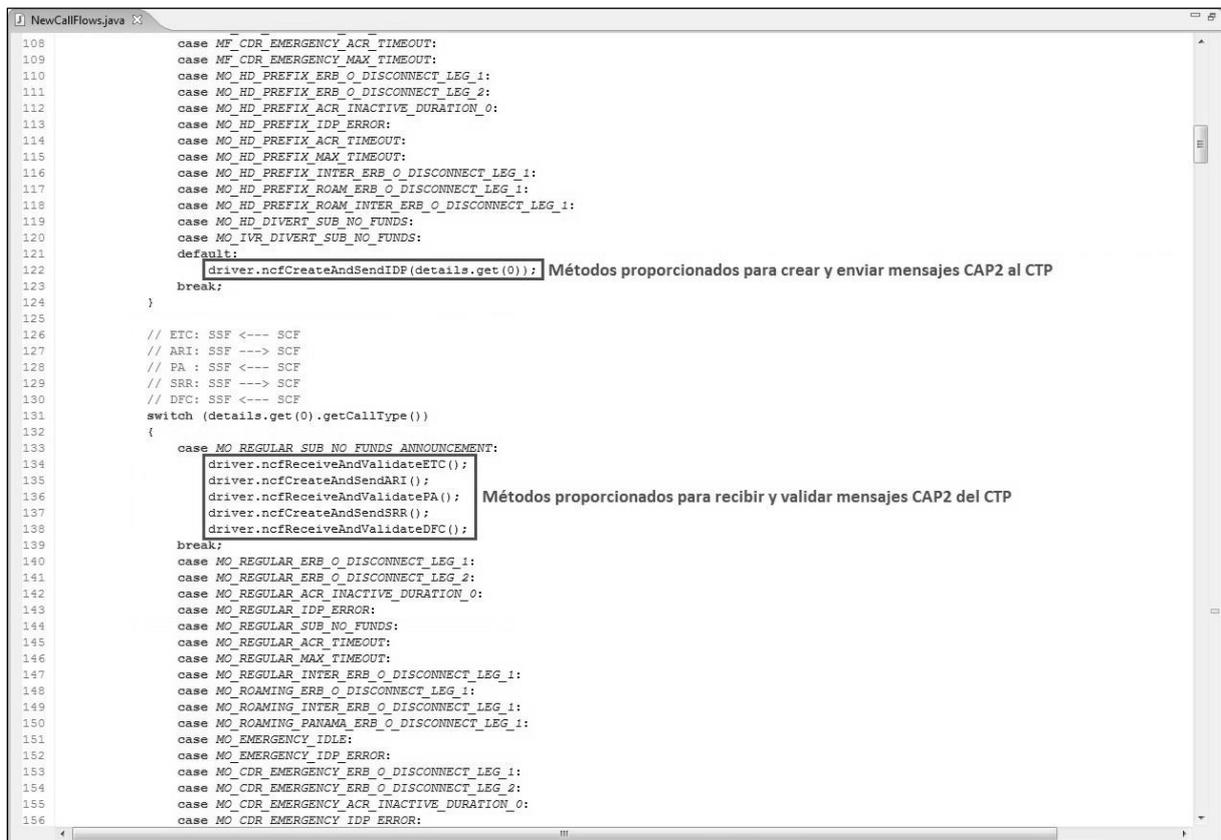
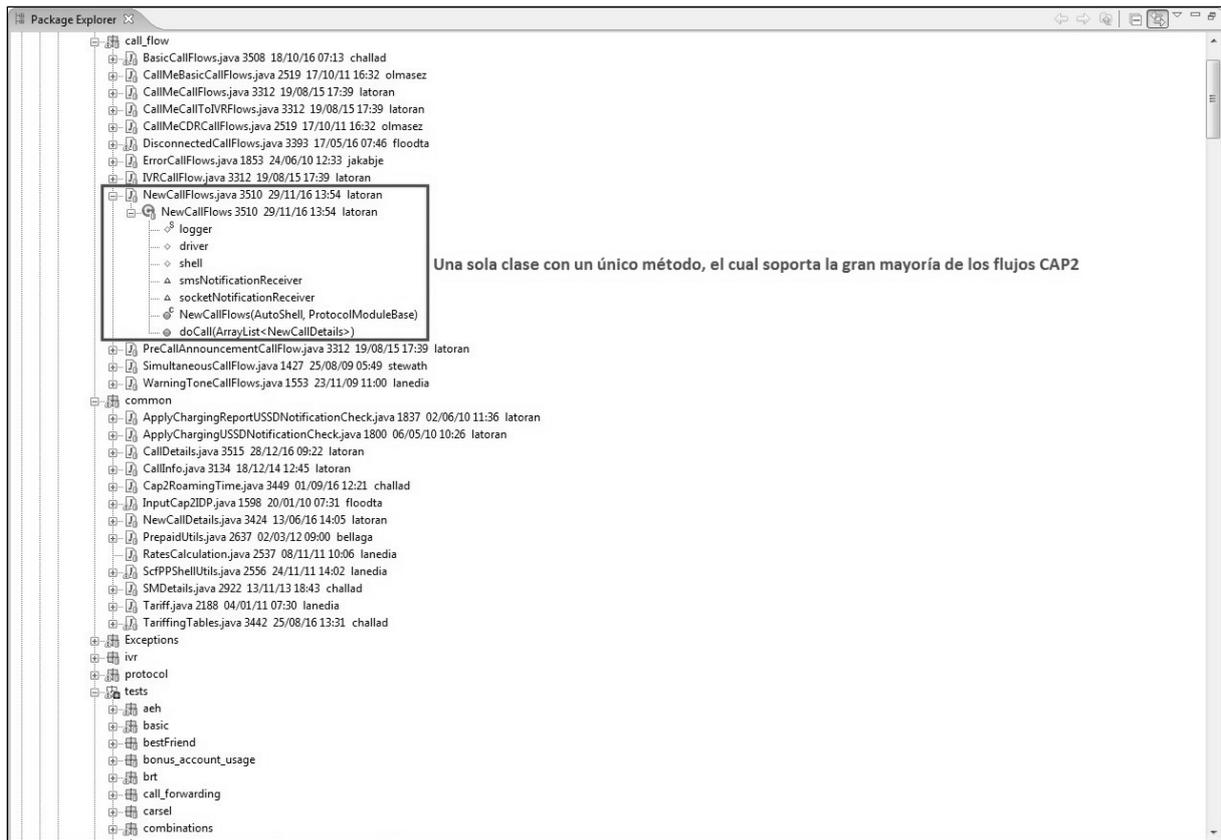
Variables claras y organizadas en grupos

```

292 int callType;
293 CallType(int type) { callType = type; }
294 public int getId() { return callType; }
295 }
296
297
298
299 /*----- GETTERS AND SETTERS -----*/
300
301 // General Getters and Setters
302 public Prepaid.Version getVersion()
303 {
304     return version;
305 }
306
307 public void setVersion(Prepaid.Version version)
308 {
309     this.version = version;
310 }
311
312
313 // CDR Getters and Setters
314 public VoiceField.Transaction_type.Id getTransactionType()
315 {
316     return transactionType;
317 }
318
319 public void setTransactionType(VoiceField.Transaction_type.Id transactionType)
320 {
321     this.transactionType = transactionType;
322 }
323
324 public int getExternalTransactionType()
325 {
326     return externalTransactionType;
327 }
328
329 public void setExternalTransactionType(int externalTransactionType)
330 {
331     this.externalTransactionType = externalTransactionType;
332 }
333
334 public VoiceField.Service_type.Id getServiceType()
335 {
336     return serviceType;
337 }
338
339 public void setServiceType(VoiceField.Service_type.Id serviceType)
340 {

```

Getters y Setters como única forma de manipular las variables



```

504     case MF_EMERGENCY_IDP_ERROR:
505     case MF_CDR_EMERGENCY_ACR_INACTIVE_DURATION_0:
506     case MF_CDR_EMERGENCY_IDP_ERROR:
507     case MO_HD_PREFIX_ACR_INACTIVE_DURATION_0:
508     case MO_HD_PREFIX_IDP_ERROR:
509     case MO_HD_DIVERT_SUB_NO_FUNDS:
510     case MO_IVR_DIVERT_SUB_NO_FUNDS:
511         // Skip
512         break;
513     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
514     case MO_REGULAR_ERB_O_DISCONNECT_LEG_2:
515     case MO_REGULAR_ACR_TIMEOUT:
516     case MO_REGULAR_MAX_TIMEOUT:
517     case MO_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
518     case MO_REGULAR_INTER_ERB_O_DISCONNECT_LEG_2:
519     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
520     case MO_REGULAR_PANAMA_ERB_O_DISCONNECT_LEG_1:
521     case MO_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_1:
522     case MO_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_2:
523     case MO_CDR_EMERGENCY_ACR_TIMEOUT:
524     case MO_CDR_EMERGENCY_MAX_TIMEOUT:
525     case MT_REGULAR_ERB_T_DISCONNECT_LEG_1:
526     case MT_REGULAR_ERB_T_DISCONNECT_LEG_2:
527     case MT_REGULAR_ACR_TIMEOUT:
528     case MT_REGULAR_MAX_TIMEOUT:
529     case MT_REGULAR_ERB_T_DISCONNECT_LEG_1:
530     case MF_REGULAR_ERB_O_DISCONNECT_LEG_1:
531     case MF_REGULAR_ERB_O_DISCONNECT_LEG_2:
532     case MF_REGULAR_ACR_TIMEOUT:
533     case MF_REGULAR_MAX_TIMEOUT:
534     case MF_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
535     case MF_REGULAR_ERB_O_DISCONNECT_LEG_1:
536     case MF_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
537     case MF_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_1:
538     case MF_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_2:
539     case MF_CDR_EMERGENCY_ACR_TIMEOUT:
540     case MF_CDR_EMERGENCY_MAX_TIMEOUT:
541     case MO_HD_PREFIX_ERB_O_DISCONNECT_LEG_1:
542     case MO_HD_PREFIX_ERB_O_DISCONNECT_LEG_2:
543     case MO_HD_PREFIX_ACR_TIMEOUT:
544     case MO_HD_PREFIX_MAX_TIMEOUT:
545     case MO_HD_PREFIX_INTER_ERB_O_DISCONNECT_LEG_1:
546     case MO_HD_PREFIX_ROAM_ERB_O_DISCONNECT_LEG_1:
547     case MO_HD_PREFIX_ROAM_INTER_ERB_O_DISCONNECT_LEG_1:
548     default:
549         driver.ncfCreateAndSendACR_ncfReceiveAndValidateAC();
550     break;
551 }
552

```

Los métodos son incluidos o ignorados dependiendo del flujo de llamada

```

822     assertNull("Unexpected SOCKET Notification received", unexpectedSocketNotification);
823 }
824
825 // Receives and Validates CDRs
826 if (details.get(0).getIncludeCDR())
827 {
828     if (driver.getTestCDRs())
829     {
830         for (int i = 0; i < details.size(); i++)
831         {
832             NewCDR cdr = driver.receiveCDR();
833             assertNull("Expected CDR not received ", cdr);
834             PPSAssert.assertCdrEquals(details.get(i).cdr(), CdrHandler.decodeCdrToString(cdr.getAsnidata()));
835         }
836
837         NewCDR cdrUnexpected = driver.receiveCDR();
838         assertNull("Unexpected CDR received ", cdrUnexpected);
839     }
840 }
841 else
842 {
843     if (driver.getTestCDRs())
844     {
845         NewCDR cdrUnexpected = driver.receiveCDR();
846         assertNull("Unexpected CDR received ", cdrUnexpected);
847     }
848 }
849 }
850 catch (Exception e)
851 {
852     fail(e);
853 }
854 finally
855 {
856     // Stops CDRs Handler
857     driver.stopGCDRH();
858
859     // Stops SMS Notifications Handler
860     if (smsNotificationReceiver != null)
861     {
862         smsNotificationReceiver.shutdown();
863     }
864
865     // Stops SOCKET Notifications Handler
866     if (socketNotificationReceiver != null)
867     {
868         socketNotificationReceiver.shutdown();
869     }
870

```

Validación de múltiples CDRs

```

1418         if (sqlTester != null)
1419         {
1420             sqlTester.cleanup();
1421         }
1422     }
1423 }
1424
1425
1426
1427 /*-----*/
1428 /* ONLY FOR NEW CALL DETAILS & NEW CALL FLOWS - START */
1429 /*-----*/
1430
1431 // IDP: SSF ---> SCF
1432 public void ncfCreateAndSendIDP(NewCallDetails details) throws Exception
1433 {
1434     newCallDetails = details;
1435
1436     ncfInvokeIdOut = 0;
1437     ncfInvokeIdIn = 1;
1438     ncfInvokeIdSrf = 1;
1439     ncfCorrelationNumber = "";
1440     Cap2TestMachine machine = allocateTestMachine();
1441
1442     IDP idp = new IDP();
1443     idp.setInvokeID((byte)ncfInvokeIdOut);
1444
1445     // ServiceKey
1446     switch (newCallDetails.getCallType())
1447     {
1448     case MO_REGULAR_IDP_ERROR:
1449     case MO_EMERGENCY_IDP_ERROR:
1450     case MO_CDR_EMERGENCY_IDP_ERROR:
1451     case MT_REGULAR_IDP_ERROR:
1452     case MF_REGULAR_IDP_ERROR:
1453     case MF_EMERGENCY_IDP_ERROR:
1454     case MF_CDR_EMERGENCY_IDP_ERROR:
1455     case MO_HD_PREFIX_IDP_ERROR:
1456         // ServiceKey = 5
1457         idp.setServiceKey(5);
1458         break;
1459     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
1460     case MO_REGULAR_ERB_O_DISCONNECT_LEG_2:
1461     case MO_REGULAR_ACR_INACTIVE_DURATION_0:
1462     case MO_REGULAR_SUB_NO_FUNDS:
1463     case MO_REGULAR_SUB_NO_FUNDS_ANNOUNCEMENT:
1464     case MO_REGULAR_ACR_TIMEOUT:
1465     case MO_REGULAR_MAX_TIMEOUT:
1466     case MO_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
    
```

Métodos creados en la misma clase, para reutilizar la máquina de estados

La máquina de estados funciona bien y no formó parte de la reingeniería

```

1523     case MT_REGULAR_ERB_T_CALLED_PARTY_BUSY_LEG_2:
1524     case MT_REGULAR_ACR_INACTIVE_DURATION_0:
1525     case MT_REGULAR_IDP_ERROR:
1526     case MT_REGULAR_SUB_NO_FUNDS:
1527     case MT_REGULAR_ACR_TIMEOUT:
1528     case MT_REGULAR_MAX_TIMEOUT:
1529     case MT_ROAMING_ERB_T_DISCONNECT_LEG_1:
1530     case MF_REGULAR_ERB_O_DISCONNECT_LEG_1:
1531     case MF_REGULAR_ERB_O_DISCONNECT_LEG_2:
1532     case MF_REGULAR_ACR_INACTIVE_DURATION_0:
1533     case MF_REGULAR_IDP_ERROR:
1534     case MF_REGULAR_SUB_NO_FUNDS:
1535     case MF_REGULAR_ACR_TIMEOUT:
1536     case MF_REGULAR_MAX_TIMEOUT:
1537     case MF_ROAMING_ERB_O_DISCONNECT_LEG_1:
1538         // CalledPartyNumber = B Number + National Number + Routing To Internal Network Number Allowed + ISDN Numbering Plan + 0
1539         calledPartyNumber = new Q763CalledAddress(newCallDetails.getBNumberPrefixIDP() + newCallDetails.getBNumber(), 3, 0, 1, 0);
1540         idp.setCalledPartyNumber(calledPartyNumber.encode());
1541         break;
1542     case MF_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
1543     case MF_ROAMING_INTER_ERB_O_DISCONNECT_LEG_1:
1544         // CalledPartyNumber = B Number + International Number + Routing To Internal Network Number Allowed + ISDN Numbering Plan + 0
1545         calledPartyNumber = new Q763CalledAddress(newCallDetails.getBNumberPrefixIDP() + newCallDetails.getBNumber(), 4, 0, 1, 0);
1546         idp.setCalledPartyNumber(calledPartyNumber.encode());
1547         break;
1548     case MF_EMERGENCY_IDLE:
1549     case MF_EMERGENCY_IDP_ERROR:
1550     case MF_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_1:
1551     case MF_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_2:
1552     case MF_CDR_EMERGENCY_ACR_INACTIVE_DURATION_0:
1553     case MF_CDR_EMERGENCY_IDP_ERROR:
1554     case MF_CDR_EMERGENCY_ACR_TIMEOUT:
1555     case MF_CDR_EMERGENCY_MAX_TIMEOUT:
1556         // CalledPartyNumber = B Number + Unknown National + Routing To Internal Network Number Allowed + ISDN Numbering Plan + 0
1557         calledPartyNumber = new Q763CalledAddress(newCallDetails.getBNumberPrefixIDP() + newCallDetails.getBNumber(), 2, 0, 1, 0);
1558         idp.setCalledPartyNumber(calledPartyNumber.encode());
1559         break;
1560     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
1561     case MO_REGULAR_ERB_O_DISCONNECT_LEG_2:
1562     case MO_REGULAR_ACR_INACTIVE_DURATION_0:
1563     case MO_REGULAR_IDP_ERROR:
1564     case MO_REGULAR_SUB_NO_FUNDS:
1565     case MO_REGULAR_SUB_NO_FUNDS_ANNOUNCEMENT:
1566     case MO_REGULAR_ACR_TIMEOUT:
1567     case MO_REGULAR_MAX_TIMEOUT:
1568     case MO_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
1569     case MO_ROAMING_ERB_O_DISCONNECT_LEG_1:
1570     case MO_ROAMING_INTER_ERB_O_DISCONNECT_LEG_1:
1571     case MO_ROAMING_PANAMA_ERB_O_DISCONNECT_LEG_1:
    
```

Valores y configuraciones de los atributos de un mensaje CAP2 basados en el tipo de llamada

```

J Cap2TestModule.java x
2775     break;
2776 } Complejidad oculta al desarrollador principiante (Ej: Los tiempos en CAP2 van en décimas de segundo)
2777
2778 // TimeINoTariffSwitch = Accumulated Duration
2779 acr.setTimeInformation_choice(TimeInformation_choice.TIME_IF_NO_TARIFF_SWITCH);
2780 acr.setTimeINoTariffSwitch(accumulatedDuration*10);
2781
2782 // LegActive = True
2783 acr.setCallActive(true);
2784
2785 machine.sendApplyChargingReport(acr);
2786 ncfInvokeIdOut++;
2787
2788 if (accumulatedDuration < newCallDetails.getMaxDuration())
2789 {
2790     // Separates ACR from AC
2791     sleepMs(10);
2792
2793     // Receives and Validates AC
2794     ApplyCharging ac = machine.receiveApplyCharging(getBasicTimeoutMs());
2795     assertNotNull("Apply Charging - Not Received ", ac);
2796     assertEquals("ApplyCharging - InvokeId - Not Matching ", (byte)ncfInvokeIdIn, ac.getInvokeID());
2797
2798     // MaxCallPeriodDuration = AC Duration
2799     intervalDuration = accumulatedDuration + newCallDetails.getAcDuration() < newCallDetails.getMaxDuration() ? newCallDetails.getAcDuration() :
2800     assertNotNull("Apply Charging - Max Call Period Duration - Not Received ", ac.getMaxCallPeriodDuration());
2801     assertEquals("Apply Charging - Max Call Period Duration - Not Matching ", intervalDuration*10, ac.getMaxCallPeriodDuration().intValue());
2802
2803     // SendingSide
2804     switch (newCallDetails.getCallType())
2805     {
2806     case MT_REGULAR_ERB_T_DISCONNECT_LEG_1:
2807     case MT_REGULAR_ERB_T_DISCONNECT_LEG_2:
2808     case MT_REGULAR_MAX_TIMEOUT:
2809     case MT_ROAMING_ERB_T_DISCONNECT_LEG_1:
2810         // SendingSide = Leg 2
2811         assertNotNull("Apply Charging - Sending Side Choice - Not Received ", ac.getSendingSideID_choice());
2812         assertEquals("Apply Charging - Sending Side Choice - Not Matching ", SendingSideID_choice.SENDING_SIDE_ID, ac.getSendingSideID_choice
2813         assertEquals("Apply Charging - Sending Side - Not Received ", ac.getSendingSideID());
2814         assertEquals("Apply Charging - Sending Side - Not Matching ", LegType.LEG2, ac.getSendingSideID());
2815     break;
2816     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
2817     case MO_REGULAR_ERB_O_DISCONNECT_LEG_2:
2818     case MO_REGULAR_MAX_TIMEOUT:
2819     case MO_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
2820     case MO_ROAMING_ERB_O_DISCONNECT_LEG_1:
2821     case MO_ROAMING_INTER_ERB_O_DISCONNECT_LEG_1:
2822     case MO_ROAMING_PANAMA_ERB_O_DISCONNECT_LEG_1:
2823     case MO_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_1:

```

```

J Cap2TestModule.java x
3021     default:
3022         // EventTypeBCSM = O Disconnect
3023         erb.setEventTypeBCSM(EventTypeBCSM.O_DISCONNECT);
3024     break;
3025 }
3026
3027 // CauseIE
3028 switch(newCallDetails.getCallType())
3029 {
3030     case MO_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_1:
3031     case MO_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_2:
3032     case MF_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_1:
3033     case MF_CDR_EMERGENCY_ERB_O_DISCONNECT_LEG_2:
3034         // CauseIE = Not Continued + CCITT standard + Spare + Public Network Serving the Local User + Not Continued + Cause Value = Normal Call C
3035         erb.setEventSpecificInformationBCSM_choice(EventSpecificInformationBCSM_choice.O_DISCONNECT_SPECIFIC_INFO);
3036         erb.setEventSpecificInformationBCSM(new byte[] {(byte)0x82, (byte)0x90});
3037         // 0x82 -- 0x90 = 10000010 -- 10010000 = 1 (Extension Bit = Not Continued) + 00 (Coding Standard = CCITT standard) + 0 (Spare = Spare) +
3038     break;
3039     case MT_REGULAR_ERB_T_DISCONNECT_LEG_1:
3040     case MT_REGULAR_ERB_T_DISCONNECT_LEG_2:
3041     case MT_ROAMING_ERB_T_DISCONNECT_LEG_1:
3042         // CauseIE = Not Continued + CCITT standard + Spare + Private Network Serving the Local User + Not Continued + Cause Value = Normal Call
3043         erb.setEventSpecificInformationBCSM_choice(EventSpecificInformationBCSM_choice.T_DISCONNECT_SPECIFIC_INFO);
3044         erb.setEventSpecificInformationBCSM(new byte[] {(byte)0x81, (byte)0x90});
3045         // 0x81 -- 0x90 = 10000001 -- 10010000 = 1 (Extension Bit = Not Continued) + 00 (Coding Standard = CCITT standard) + 0 (Spare = Spare) +
3046     break;
3047     case MT_REGULAR_ERB_T_NO_ANSWER_LEG_2:
3048         // CauseIE = Not Continued + CCITT standard + Spare + Private Network Serving the Local User + Not Continued + Cause Value = No Answer Fo
3049         erb.setEventSpecificInformationBCSM_choice(EventSpecificInformationBCSM_choice.T_NO_ANSWER_SPECIFIC_INFO);
3050         erb.setEventSpecificInformationBCSM(new byte[] {(byte)0x81, (byte)0x93});
3051         // 0x81 -- 0x93 = 10000001 -- 10010011 = 1 (Extension Bit = Not Continued) + 00 (Coding Standard = CCITT standard) + 0 (Spare = Spare) +
3052     break;
3053     case MT_REGULAR_ERB_T_CALLED_PARTY_BUSY_LEG_2:
3054         // CauseIE = Not Continued + CCITT standard + Spare + Private Network Serving the Local User + Not Continued + Cause Value = User Busy
3055         erb.setEventSpecificInformationBCSM_choice(EventSpecificInformationBCSM_choice.T_BUSY_SPECIFIC_INFO);
3056         erb.setEventSpecificInformationBCSM(new byte[] {(byte)0x81, (byte)0x91});
3057         // 0x81 -- 0x91 = 10000001 -- 10010001 = 1 (Extension Bit = Not Continued) + 00 (Coding Standard = CCITT standard) + 0 (Spare = Spare) +
3058     break;
3059     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
3060     case MO_REGULAR_ERB_O_DISCONNECT_LEG_2:
3061     case MO_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
3062     case MO_ROAMING_ERB_O_DISCONNECT_LEG_1:
3063     case MO_ROAMING_INTER_ERB_O_DISCONNECT_LEG_1:
3064     case MO_ROAMING_PANAMA_ERB_O_DISCONNECT_LEG_1:
3065     case MF_REGULAR_ERB_O_DISCONNECT_LEG_1:
3066     case MF_REGULAR_ERB_O_DISCONNECT_LEG_2:
3067     case MF_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1:
3068     case MF_ROAMING_ERB_O_DISCONNECT_LEG_1:
3069     case MF_ROAMING_INTER_ERB_O_DISCONNECT_LEG_1:

```

## Anexo 8: Utilización del Entorno de Pruebas

Para poder validar el correcto funcionamiento del entorno de pruebas tras su reingeniería, se crearon inicialmente treinta y cuatro flujos de llamada básicos, los cuales eran utilizados en treinta y ocho casos de prueba.

Hoy en día el entorno de pruebas cuenta con alrededor de sesenta flujos de llamada diferentes, utilizados en más de quinientos cincuenta casos de prueba.

Con el objetivo de graficar los cambios realizados sobre el entorno y de representar la forma en que se trabaja sobre él, se eligieron los tres escenarios principales que se pueden dar cuando se crea un nuevo caso de prueba:

- Utilizar un tipo de llamada existente.
- Crear un tipo de llamada nuevo.
- Crear un tipo de llamada nuevo con cambios en los atributos del mensaje.

### Caso 1 - Llamada Saliente de 30 Segundos

Paso 1:

Detectar en *NewCallDetails.java* cual es el tipo de llamada existente que mejor representa el escenario a probar. En este caso se trata del valor por defecto, una llamada normal:

```

1294     protected int     acrTimeoutDuration    = 200;
1295     protected int     callDuration         = 300;
1296     protected int     maxDuration          = 7200;
1297     protected CallType callType            = CallType.MO_REGULAR_ERB_O_DISCONNECT_LEG_1;

```

```

225     // Call Types
226     public enum CallType ← Listado de tipos de llamada ↑
227     {
228         // ncfBasicCalls
229         MO_REGULAR_ERB_O_DISCONNECT_LEG_1(1), ← Tipo de llamada por defecto
230         MO_REGULAR_ERB_O_DISCONNECT_LEG_2(2),   (MO Regular)
231         MO_REGULAR_ACR_INACTIVE_DURATION_0(3),
232         MO_REGULAR_IDP_ERROR(4),
233         MO_REGULAR_SUB_NO_FUNDS(5),
234         MO_REGULAR_SUB_NO_FUNDS_ANNOUNCEMENT(6),
235         MO_REGULAR_ACR_TIMEOUT(7),
236         MO_REGULAR_MAX_TIMEOUT(8),
237         MO_REGULAR_INTER_ERB_O_DISCONNECT_LEG_1(9),
238         MO_ROAMING_ERB_O_DISCONNECT_LEG_1(10),
239         MO_ROAMING_INTER_ERB_O_DISCONNECT_LEG_1(11),
240         MO_ROAMING_PANAMA_ERB_O_DISCONNECT_LEG_1(12),
241         MT_REGULAR_ERB_T_DISCONNECT_LEG_1(13),
242         MT_REGULAR_ERB_T_DISCONNECT_LEG_2(14),

```

**Paso 2:**

Definir en *NewCallDetails.java* cuales son los detalles (variables) que sería necesario modificar para lograr el escenario deseado. En este caso lo único que habría que cambiar son las variables vinculadas a la duración y al costo de la llamada:

1292	protected String	visitedMscAddress	= "35361900000";
1293	protected int	acDuration	= 120;
1294	protected int	acrTimeoutDuration	= 200;
1295	protected int	callDuration	= 300;
1296	protected int	maxDuration	= 7200;

1230	protected String	homeMscAddressCDR	= "35361900000";
1231	protected String	visitedMscAddressCDR	= "35361900000";
1232	protected int	operatorId	= -1;
1233	protected int	callOrigId	= 0;
1234	protected int	callTypeId	= 2;
1235	protected int	callDurationCDR	= 300;
1236	protected int	airTimeCharged	= 300;
1237	protected int	tollTimeCharged	= 300;
1238	protected int	itelTimeCharged	= 300;
1239	protected int	tariffId	= 0;
1240	protected int	periodicTariffId	= 0;
1241	protected int	subPeriodicCycle	= 0;

1244	protected String	cellName	= "IRL-HOM";
1245	protected String	destinationName	= "IRL-NAT";
1246	protected String	timebandName	= "HOM-NAT-T";
1247	protected String	rateName	= "HOM-NAT-R";
1248	protected int	tzAdjustment	= 0;
1249	protected int	airCost	= 50000;
1250	protected int	tollCost	= 50000;
1251	protected int	itelCost	= 0;
1252	protected int	tax	= 0;

1262	protected int	bonusPercentage	= 0;
1263	protected int	bonusAmount	= 0;
1264	protected int[]	originalBalances	= {10000000, 0};
1265	protected int[]	balanceDeltas	= { -100000, 0};
1266	protected int[]	balanceReservations	= { 0, 0};
1267	protected int	activeReservationCount	= -1;
1268	protected int	totalPackageBalance	= 0;
1269	protected int	totalPackageDelta	= 0;
1270	protected String	packageDetails	= " ";
1271	protected int	smPackageId	= 0;
1272	protected String	cgrUniqueId	= "";
1273	protected int	cgrUnchargedCost	= 0;

**Paso 3:**

Identificar los métodos set a utilizar para cambiar el valor por defecto de las variables:

1116	public int	getCallDuration()	
1120			
1121	public void	setCallDuration(int callDuration)	
1122	{		
1123		this.callDuration = callDuration;	
1124	}		

```
544⊕ public int getCallDurationCDR()␣
548
549⊖ public void setCallDurationCDR(int callDurationCDR)
550 {
551     this.callDurationCDR = callDurationCDR;
552 }
553
554⊕ public int getAirTimeCharged()␣
558
559⊖ public void setAirTimeCharged(int airTimeCharged)
560 {
561     this.airTimeCharged = airTimeCharged;
562 }
563
564⊕ public int getTollTimeCharged()␣
568
569⊖ public void setTollTimeCharged(int tollTimeCharged)
570 {
571     this.tollTimeCharged = tollTimeCharged;
572 }
573
574⊕ public int getItelTimeCharged()␣
578
579⊖ public void setItelTimeCharged(int itelTimeCharged)
580 {
581     this.itelTimeCharged = itelTimeCharged;
582 }
```

```
684⊕ public int getAirCost()␣
688
689⊖ public void setAirCost(int airCost)
690 {
691     this.airCost = airCost;
692 }
693
694⊕ public int getTollCost()␣
698
699⊖ public void setTollCost(int tollCost)
700 {
701     this.tollCost = tollCost;
702 }
703
704⊕ public int getItelCost()␣
708
709⊖ public void setItelCost(int itelCost)
710 {
711     this.itelCost = itelCost;
712 }
```

```
1158⊕ public int getBalanceDelta(Account account)␣
1162
1163⊖ public void setBalanceDelta(Account account, int balance)
1164 {
1165     this.balanceDeltas[account.getId()] = balance;
1166 }
```

#### Paso 4:

Crear el caso de prueba, modificando las variables de *NewCallDetails.java* con los métodos set y llamando al flujo de llamada único, definido en *NewCallFlows.java*:

```

NewCallDetails.java  NewCallFlows.java  Cap2TestModule.java  MOBasicCallTests.java
1166
1167  /*-----*/
1168  /* TEST CASES */
1169  /*-----*/
1170
1171  public void test_001_MO_CALL_30_SECONDS()
1172  {
1173      // CallDetails
1174      callDetail.setCallDuration(30);
1175      callDetail.setCallDurationCDR(30);
1176      callDetail.setAirTimeCharged(30);
1177      callDetail.setTollTimeCharged(30);
1178      callDetail.setItelTimeCharged(30);
1179      callDetail.setAirCost(10000);
1180      callDetail.setTollCost(10000);
1181      callDetail.setItelCost(0);
1182      callDetail.setBalanceDelta(Account.MAIN_ACCOUNT, -20000);
1183
1184      try
1185      {
1186          // PreCall
1187          setConfig();
1188
1189          // Call
1190          getNewCallTester().doCall(callDetails);
1191
1192          // PostCall
1193          getResult();
1194      }
1195      catch (Exception e)
1196      {
1197          fail(e);
1198      }
1199  }
1200

```

## Caso 2 - Llamada Saliente sin Release Call

### Paso 1:

Definir en *NewCallDetails.java* el nuevo tipo de llamada que se debe crear:

```

277  MF_CDR_EMERGENCY_MAX_TIMEOUT(48),
278  // ncfHelpdeskPrefixCalls
279  MO_HD_PREFIX_ERB_O_DISCONNECT_LEG_1(49),
280  MO_HD_PREFIX_ERB_O_DISCONNECT_LEG_2(50),
281  MO_HD_PREFIX_ACR_INACTIVE_DURATION_0(51),
282  MO_HD_PREFIX_IDP_ERROR(52),
283  MO_HD_PREFIX_ACR_TIMEOUT(53),
284  MO_HD_PREFIX_MAX_TIMEOUT(54),
285  MO_HD_PREFIX_INTER_ERB_O_DISCONNECT_LEG_1(55),
286  MO_HD_PREFIX_ROAM_ERB_O_DISCONNECT_LEG_1(56),
287  MO_HD_PREFIX_ROAM_INTER_ERB_O_DISCONNECT_LEG_1(57),
288  // ncfDivertCalls
289  MO_HD_DIVERT_SUB_NO_FUNDS(58),
290  MO_IVR_DIVERT_SUB_NO_FUNDS(59),
291  // ncfNoReleaseCall
292  MO_REGULAR_NO_RELEASE_CALL(60);
293
294  int callType;
295  CallType(int type) { callType = type; }
296  public int getId() { return callType; }

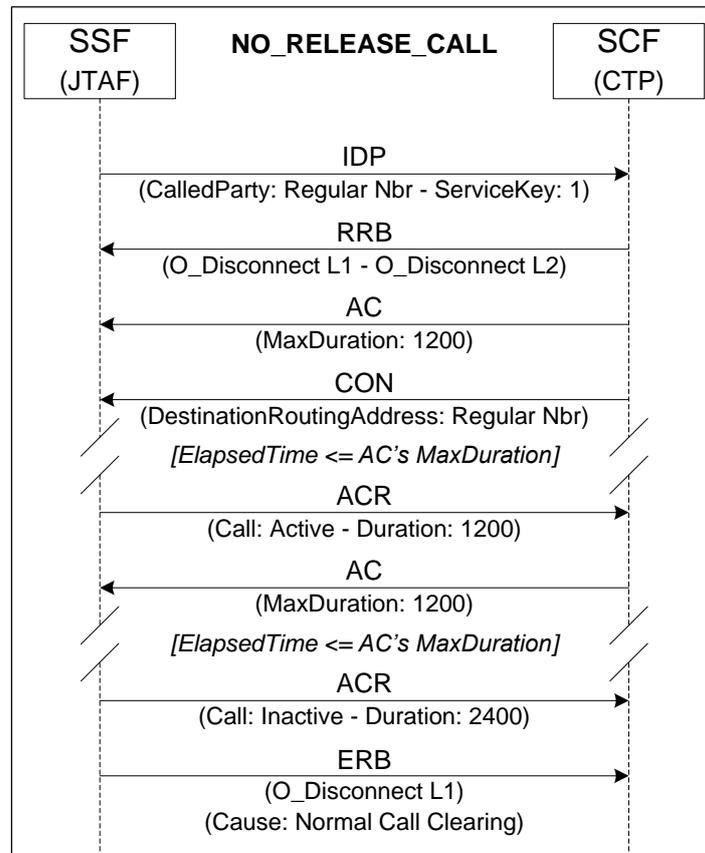
```

← Listado de tipos de llamada

← Nuevo tipo de llamada

**Paso 2:**

Determinar el flujo de mensajes CAP2, intercambiado en el nuevo tipo de llamada:

**Paso 3:**

Agregar el nuevo tipo de llamada al flujo único, definido en *NewCallFlows.java*. Este es como el de la llamada normal (*MO\_REGULAR\_ERB\_O\_DISCONNECT\_LEG\_1*), con la excepción de que no habrá un Release Call. Por lo tanto, la definición de cómo será el flujo de mensajes CAP2 se puede realizar siguiendo esta premisa:

```

118     case MO_HD_DIVERT_SUB_NO_FUNDS:
119     case MO_IVR_DIVERT_SUB_NO_FUNDS:
120     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
121     case MO_REGULAR_NO_RELEASE_CALL: ← Crear y enviar el IDP, igual
122     default:                                     que en una llamada normal
123         driver.ncfCreateAndSendIDP(details.get(0));
124         break;
125     }
126
127     // ETC: SSF <--- SCF
128     // ARI: SSF ---> SCF
129     // PA : SSF <--- SCF
130     // SRR: SSF ---> SCF
131     // DFC: SSF <--- SCF
132     switch (details.get(0).getCallType())
133     {
134     case MO_REGULAR_SUB_NO_FUNDS_ANNOUNCEMENT:
  
```

```

419 // CON: SSF <--- SCF
420 switch (details.get(0).getCallType())
421 {
422     case MO_REGULAR_IDP_ERROR:
423     case MO_REGULAR_SUB_NO_FUNDS:
424     case MO_REGULAR_SUB_NO_FUNDS_ANNOUNCEMENT:
425     case MO_EMERGENCY_IDP_ERROR:
426     case MO_CDR_EMERGENCY_IDP_ERROR:
427     case MT_REGULAR_IDP_ERROR:
428     case MT_REGULAR_SUB_NO_FUNDS:
429     case MF_REGULAR_IDP_ERROR:
430     case MF_REGULAR_SUB_NO_FUNDS:
431     case MF_EMERGENCY_IDP_ERROR:
432     case MF_CDR_EMERGENCY_IDP_ERROR:
433     case MO_HD_PREFIX_IDP_ERROR:
434     case MO_IVR_DIVERT_SUB_NO_FUNDS:
435         // Skip
436         break;
437     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
438     case MO_REGULAR_NO_RELEASE_CALL:

```

Recibir y validar el CON, igual que en una llamada normal

```

700 // RC: SSF <--- SCF
701 switch (details.get(0).getCallType())
702 {
703     case MO_EMERGENCY_IDLE:
704     case MF_EMERGENCY_IDLE:
705     case MO_IVR_DIVERT_SUB_NO_FUNDS:
706     case MO_REGULAR_NO_RELEASE_CALL:
707         // Skip
708         break;
709     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:

```

No recibir ni validar el RC, a diferencia de una llamada normal

#### Paso 4:

Crear el caso de prueba, modificando la variable que define el tipo de llamada:

```

NewCallDetails.java  NewCallFlows.java  Cap2TestModule.java  MOBasicCallTests.java X
1166
1167 /*-----*/
1168 /* TEST CASES */
1169 /*-----*/
1170
1171 public void test_002_MO_CALL_NO_RELEASE()
1172 {
1173     // CallDetails
1174     callDetail.setCallType(CallType.MO_REGULAR_NO_RELEASE_CALL);
1175
1176     try
1177     {
1178         // PreCall
1179         setConfig();
1180
1181         // Call
1182         getNewCallTester().doCall(callDetails);
1183
1184         // PostCall
1185         getResult();
1186     }
1187     catch (Exception e)
1188     {
1189         fail(e);
1190     }
1191 }
1192

```

### Caso 3 - Llamada Saliente a un Short Code

#### Paso 1:

Repetir los primeros tres pasos del segundo caso, tanto en *NewCallDetails.java* como en *NewCallFlows.java*. Sin embargo, en esta oportunidad el tipo de llamada se llama `MO_REGULAR_SHORT_CODE` y se comporta igual que una llamada normal:

```

287     MO_HD_PREFIX_ROAM_INTER_ERB_O_DISCONNECT_LEG_1(57),
288     // ncfDivertCalls
289     MO_HD_DIVERT_SUB_NO_FUNDS(58),
290     MO_IVR_DIVERT_SUB_NO_FUNDS(59),
291     // ncfNoReleaseCall
292     MO_REGULAR_NO_RELEASE_CALL(60),
293     // ncfShortCode
294     MO_REGULAR_SHORT_CODE(61);

```

Nuevo tipo de llamada

```

709     // RC: SSF <--- SCF
710     switch (details.get(0).getCallType())
711     {
712         case MO_EMERGENCY_IDLE:
713         case MF_EMERGENCY_IDLE:
714         case MO_IVR_DIVERT_SUB_NO_FUNDS:
715         case MO_REGULAR_NO_RELEASE_CALL:
716             // Skip
717             break;
718         case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:
719         case MO_REGULAR_SHORT_CODE:

```

Crear y recibir mensajes igual que en una llamada normal

#### Paso 2:

La diferencia con el caso anterior es que resulta necesario cambiar las características del mensaje IDP, para soportar el llamado a un Short Code (\*444, \*555, etc). Por lo tanto hay que modificar dicho mensaje CAP2 en la clase *Cap2TestModule.java*, para empezar a soportar el nuevo tipo de llamada:

```

2344     case MO_HD_PREFIX_ROAM_INTER_ERB_O_DISCONNECT_LEG_1:
2345         // CalledPartyBCDNumber = B Number + ISDN Numbering Plan + Inter Number
2346         calledPartyBCDNumber =
2347             new Gsm0408BcdNumber
2348             (
2349                 newCallDetails.getBNumberPrefixIDP()+newCallDetails.getBNumber(),
2350                 1,
2351                 1);
2352         idp.setCalledPartyBCDNumber(calledPartyBCDNumber.encode());
2353         break;
2354     case MO_REGULAR_SHORT_CODE:
2355         // CalledPartyBCDNumber = B Number + ISDN Numbering Plan + Short Code
2356         calledPartyBCDNumber =
2357             new Gsm0408BcdNumber
2358             (
2359                 newCallDetails.getBNumberPrefixIDP()+newCallDetails.getBNumber(),
2360                 1,
2361                 4);
2362         idp.setCalledPartyBCDNumber(calledPartyBCDNumber.encode());
2363         break;
2364     case MO_REGULAR_ERB_O_DISCONNECT_LEG_1:

```

Nuevo tipo de llamada

Tipo de número del destinatario (B): Internacional (1)

Tipo de número del destinatario (B): Short Code (4)

### Paso 3:

Crear el caso de prueba, modificando las variables que definen el tipo de llamada y el número del destinatario:



```
1166
1167  /*-----*/
1168  /* TEST CASES */
1169  /*-----*/
1170
1171  public void test_003_MO_CALL_SHORT_CODE()
1172  {
1173      // CallDetails
1174      callDetail.setCallType(CallType.MO_REGULAR_SHORT_CODE);
1175      callDetail.setBNumber("555");
1176      callDetail.setCallId("555");
1177      callDetail.setOriginalNumber("555");
1178
1179      try
1180      {
1181          // PreCall
1182          setConfig();
1183
1184          // Call
1185          getNewCallTester().doCall(callDetails);
1186
1187          // PostCall
1188          getResult();
1189      }
1190      catch (Exception e)
1191      {
1192          fail(e);
1193      }
1194  }
1195
```