



## HACIA UNA WEB INDEPENDIENTE DEL DISPOSITIVO MEDIANTE CC/PP

José Luis Ferrer Riera, Anna Calveras Augé  
Departamento de Ingeniería Telemática, UPC  
Grupo de Redes Inalámbricas

*jfer6322@alu-etsetb.upc.es, anna.calveras@mat.upc.es*

*Resumen- Composite Capabilities/Preferences Profiles (CC/PP) es el nuevo lenguaje estándar creado por el World Wide Web Consortium (W3C) para que la gran diversidad de dispositivos que disponen, o dispondrán, de un acceso a Internet (móvil, PDA, PC, TV...), sean capaces de expresar sus capacidades y las preferencias del usuario mediante perfiles, tal y como su nombre indica. Para expresar estas características, los perfiles CC/PP emplean Resource Descripción Framework (RDF), otro estándar del W3C y pilar de la web semántica, el cual, a su vez, se encuentra construido sobre Extensible Markup Language (XML).*

La especificación User Agent Profile (UAProf) definida por la Open Mobile Alliance (OMA), antiguo WAPForum, utiliza CC/PP para la descripción de los teléfonos móviles. Se puede entender como el primer gran desarrollo CC/PP y ya se encuentra incluido en los últimos dispositivos móviles, implicando la existencia actual de millones de dispositivos que usan CC/PP.

Mediante CC/PP los dispositivos tienen la habilidad de enviar la información CC/PP conjuntamente con las peticiones HTTP a los servidores, de manera que éstos puedan procesar la información CC/PP y realizar la adaptación o selección de contenidos adecuados a las características del dispositivo, consiguiendo así una Web independiente del dispositivo, acercándose cada vez más a uno de las principales metas del W3C: El Acceso Universal a la Web.

En este artículo se pone de manifiesto la problemática actual de la negociación de contenidos utilizando HTTP/1.1 y se describe la especificación CC/PP, el estándar propuesto por W3C para solucionar esta carencia, justificando sus claves de diseño. También se describe el estándar de la OMA, UAProf, la primera implementación de CC/PP para la descripción de los terminales móviles que se encuentra incluido en la nueva especificación WAP 2.0.

*Palabras clave- CC/PP, UAProf, RDF, Adaptación de contenidos, móvil, Web*

### ¿POR QUÉ CC/PP?

Con la puesta en funcionamiento de los sistemas móviles de tercera generación (UMTS) y la gran utilización de redes basadas en el estándar IEEE 802.11 (Wireless LAN), se consigue una mayor capacidad en los enlaces inalámbricos, permitiendo la existencia actual de una gran heterogeneidad de dispositivos, como son los teléfonos móviles o las PDAs, capaces de acceder a multitud de contenidos web antes impensables: páginas con contenidos multimedia (audio, imágenes y video) o páginas con contenidos script, p.e. Javascript.

Actualmente, la especificación HTTP/1.1 (Hypertext Transfer Protocol)[1] permite la negociación de contenidos basada en servidor, donde la selección de la mejor representación para la respuesta HTTP es realizada por el servidor.

La selección se basa en las diferentes representaciones de la respuesta y de los contenidos de los siguientes campos de la cabecera de petición:

**-Accept.** El campo Accept se utiliza para especificar qué tipos MIME (Multipurpose Internet Mail Extensions) son aceptados con la respuesta.

**-Accept-Charset.** Este campo se puede usar para indicar qué tipos de caracteres se aceptan en la respuesta.

**-Accept-Encoding.** Este campo restringe la codificación de los contenidos que se aceptan en la respuesta.

**-Accept-Language.** Utilizado para restringir el grupo de lenguajes naturales que son preferidos como respuesta a una petición.

**-User-Agent.** Contiene la información que identifica al cliente que inicia la petición (user agent).



Un posible ejemplo de cabecera de petición sería:

```
User-Agent: Mozilla/4.04 (X11; I; SunOs 5.4 sun4m)
Accept: image/gif, image/x-xbitmap, image/jpeg, image/
png, */*
Accept-Encoding: gzip Accept-Language: es, en;
q=0.7, fr; q=0.6
Accept-Charset: iso-8859-1, *, utf-8
```

En la petición anterior, el servidor reconoce, por ejemplo, que el cliente prefiere contenidos en español, seguidos de contenidos en inglés y como última instancia, en francés. El parámetro q indica la preferencia, y su valor se encuentra entre 0 y 1, mínima y máxima preferencia, respectivamente. También es habitual utilizar el contenido del campo user agent para conocer las características correspondientes al navegador utilizado.

Este tipo de negociación basada en el servidor presenta varias desventajas:

- Es imposible determinar por parte del servidor de manera precisa qué contenido será el mejor para cualquier usuario, ya que se desconocen todas las capacidades reales del dispositivo y las que tiene activadas el *user agent*.
- Resulta ineficiente que el cliente esté transmitiendo sus capacidades en cada petición, y mucho más cuando nos encontramos en un entorno móvil donde se dispone de poco ancho de banda.
- Los parámetros deducibles a partir de las cabeceras HTTP, resultan insuficientes para una correcta descripción del dispositivo, p.e. se desconocen la totalidad de preferencias activadas por el usuario en el navegador.

La solución para la creación de contenidos web independientes del dispositivo propuesta por el World Wide Web Consortium (W3C) [2], la recomendación Composite Capabilities/Preferences Profiles (CC/PP) 1.0 [3], es el lenguaje estándar con el que los dispositivos pueden expresar sus capacidades y preferencias de usuario.

Mediante CC/PP es posible la creación de perfiles donde se expresen las características del dispositivo, agente y/o preferencias del usuario. Cuando el cliente realiza una petición al servidor, le indica su perfil CC/PP, el cual es procesado por el servidor y utilizado de guía para la adaptación de los contenidos adecuándose a las características del dispositivo que ha realizado la petición (ver figura 1).

Además, CC/PP ha sido diseñado para ser utilizado en entornos móviles, evitando en cada momento el envío de información redundante. Gracias a CC/PP, como se comprobará a continuación, se obtiene la flexibilidad que no existe en la negociación de contenidos basada en servidor utilizada en HTTP/1.1.



Figura 1. Adaptación de contenidos para distintos tamaños de pantalla

## EL PERFIL CC/PP

CC/PP está basado en RDF (Resource Description Framework) [4], otro estándar creado por el W3C. Aunque el objetivo de este documento no es la descripción detallada de RDF, es necesaria una breve introducción a sus principales características para una correcta comprensión de la composición de los perfiles CC/PP.

RDF utiliza XML (Extensible Markup Language) [5] para ser interpretado por los dispositivos y con el fin de definir vocabularios o esquemas. En cada esquema RDF se encuentran las definiciones de los atributos que representarán las propiedades de los dispositivos (en el caso de CC/PP). El modelo RDF [6] se basa en los tres objetos siguientes:

- **Recurso.** Cualquier elemento que pueda tener URI (Uniform Resource Identifier) [7] es un recurso, incluyendo documentos HTML, páginas web enteras o documentos XML. Todos los elementos descritos mediante expresiones RDF se denominan recursos.
- **Propiedad.** Una propiedad es un aspecto específico, característica, atributo o relación utilizada para

describir un recurso. Ejemplos de propiedades serían autor o título.

- **Declaración.** Una declaración en RDF está formada por un recurso junto a una propiedad definida y, con su respectivo valor. Estas tres partes son conocidas como sujeto, predicado y objeto, respectivamente.

A través de las declaraciones RDF formadas por tripletas (sujeto, predicado y objeto) es posible la creación de frases de forma similar a cualquier lenguaje. Además, estas frases RDF pueden ser procesadas por los dispositivos para la obtención de su significado. Por esta razón RDF se considera la base para la creación de la web semántica. Un simple ejemplo sería la frase:

" José Luis Ferrer es el autor del recurso  
http://www.mat.upc.es/~jferrer/ "

representada en RDF/XML como:

```
<rdf:RDF xmlns:s="http://www.mat.upc.es/~jferrer/
mi_esquema">
  <rdf:Description about="http://www.mat.upc.es/~ferrer/">
    <s:Autor>José Luis Ferrer</s:Autor>
  </rdf:Description>
</rdf:RDF>
```

El esquema RDF se indica mediante la extensión de XML con la declaración de un XML namespace:

```
<rdf:RDF xmlns:s="http://www.mat.upc.es/~jferrer/
mi_esquema">
```

En la URI indicada por el namespace XML *s*, [http://www.mat.upc.es/~jferrer/mi\\_esquema](http://www.mat.upc.es/~jferrer/mi_esquema), se encuentran las definiciones de los atributos correspondientes al esquema (en el ejemplo se utiliza la propiedad Autor).

Gracias a estar basado en RDF, CC/PP consigue ser flexible y extensible. La flexibilidad la consigue debido a que es posible la definición de un esquema por cualquier usuario, vendedor o fabricante, permitiendo así la descripción de los nuevos dispositivos que vayan apareciendo tan solo añadiendo el nuevo vocabulario. Mientras que la extensibilidad la consigue por otra razón similar, se pueden ir introduciendo nuevos atributos en los vocabularios para describir otras propiedades añadidas a los dispositivos. Además, al estar expresado en XML, distintos dispositivos y elementos (proxies, gateways...) pueden intercambiar descripciones basadas en CC/PP.

El perfil CC/PP, está compuesto por un número de declaraciones en RDF/XML donde se indican los valores de las propiedades utilizadas para definir las capacidades del dispositivo y las preferencias del usuario. CC/PP no proporciona ningún tipo de vocabulario estándar para la composición de los perfiles CC/PP. Sin embargo, gracias a CC/PP se posee el lenguaje para expresar las características de dispositivo y las preferencias del usuario.

El perfil CC/PP se constituye de una jerarquía de dos niveles:

- Un perfil con un número de componentes.
- Uno o más atributos por cada componente existente.

Los componentes equivalen a categorías donde se agrupan distintos atributos, los cuales son considerados como las propiedades. Ejemplos de nombres de componentes serían: *SoftwarePlatform*, *HardwarePlatform* o *Application*. Cada uno de estos componentes tendría sus determinados atributos. Por ejemplo, el componente *Application* podría tener los atributos: versión, nombre de la aplicación, vendedor, tipos de archivo soportados,... Todos estos atributos estarían declarados en el vocabulario o esquema, referidos mediante los nombres de espacio XML.

Un ejemplo de declaración de varios atributos de un componente en RDF sería:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns#" xmlns:ccpp="http://www.w3.org/
2002/11/08-ccpp-schema#" xmlns:ej="http://
www.ejemplo.com/vocabulario#">
  <rdf:Description rdf:about="http://
www.ejemplo.com/perfil#MiPerfil">
    <ccpp:component>
      <rdf:Description rdf:about="http://
www.ejemplo.com/perfil#Hardware">
        <rdf:type rdf:resource="http://
www.ejemplo.com/
vocabulario#PlataformaHardware"/>
        <ej:AnchoPantalla>320</ej:AnchoPantalla>
        <ej:AltoPantalla>200</ej:AltoPantalla>
      </rdf:Description>
    </ccpp:component>
  </rdf:Description>
</rdf:RDF>
```

Aparte de su expresión en XML, RDF suele representarse en un modelo gráfico, como el mostrado en la figura 2.

## ARQUITECTURA

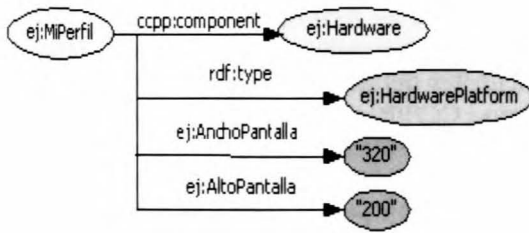


Figura 2. Representación Gráfica RDF

Los atributos de cada componente pueden ser añadidos directamente con su valor en el perfil CC/PP, o pueden ser especificados mediante referencia hacia un perfil por defecto del usuario. Éstos se indican mediante URIs, reduciendo la información a enviar con el perfil, muy importante en entornos móviles. Los atributos de un componente se indican hacia sus valores por defecto mediante *ccpp:defaults* o *ccpp:Defaults*, definidos en el namespace **ccpp**. En el ejemplo anterior, indicando la URI con los valores por defecto de los atributos de un componente, quedaría como sigue:

```
<ccpp:component>
  <rdf:Description rdf:about="http://
www.ejemplo.com/ perfil#Hardware">
    <rdf:type rdf:resource="http://www.ejemplo.com/
schema#PlataformaHardware"/>
    <ccpp:defaults rdf:resource="http://
www.ejemplo.com/PerfilHardware#HWDefault"/>
  </rdf:Description>
</ccpp:component>
```

Donde, los valores por defecto de Hardware están en la URI indicada por *ccpp:defaults* expresados mediante RDF:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns" xmlns:ej="http://www.ejemplo.com/
vocabulario#">
  <rdf:Description rdf:about="http://www.ejemplo.com/
PerfilHardware#HWDefault">
    <rdf:type rdf:resource="http://www.ejemplo.com/
vocabulario#PlataformaHardware"/>
    <ej:AnchoPantalla>320</ej:AnchoPantalla>
    <ej:AltoPantalla>200</ej:AltoPantalla>
  </rdf:Description>
</rdf:RDF>
```

Si los atributos de un componente aparecen mediante referencia en su valor por defecto, pero también aparece en el perfil un atributo indicando un valor, el valor añadido dentro del perfil toma preferencia sobre su valor por defecto.

El contexto de uso más simple de CC/PP contemplado por el W3C [8], el mostrado en la figura 3, es el caso en que el cliente envía su perfil de preferencias y capacidades en la cabecera HTTP, aplicando una extensión del protocolo HTTP. El servidor procesa el perfil CC/PP del cliente y le responde con los contenidos adaptados o adecuados según su perfil.



Figura 4. Caso de perfil CC/PP indicado por URI.

Una de las premisas de diseño de CC/PP es su utilización en entornos móviles con dispositivos como PDA's o teléfonos móviles, generalmente hasta el momento en redes de baja capacidad. Es por esta razón, que una de las características más interesantes de las que ofrece CC/PP es la de no enviar siempre el perfil completo, con todos los valores del atributo. Simplemente envía la referencia a un perfil CC/PP indicado por la URI donde se encuentra el perfil. Gracias a este método, se reduce considerablemente la cantidad de información a enviar. Tal y como se puede observar en la figura 4, lo único que debe hacer el servidor es obtener el perfil del repositorio indicado por el cliente. Existe otra ventaja importante de CC/PP, ésta reside en que proporciona un método para indicar en una petición tan solo los atributos que han cambiado desde la última petición.

Los distintos dispositivos que se encuentren entre el cliente y el servidor, como proxies o gateways,

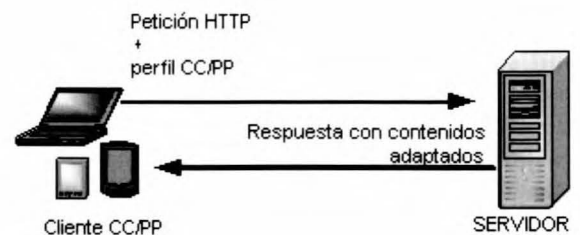


Figura 3. Caso más simple de uso CC/PP

también pueden modificar el perfil CC/PP que había incluido el cliente. Por ejemplo, podrían añadir las preferencia en el tipo de codificación que admiten o cualquier tipo de datos no admitidos por un firewall que existiera entre el cliente y el servidor.

## CC/PP EXCHANGE PROTOCOL (CC/PPEX)

CC/PP Exchange Protocol (CC/PPex) [9] es el protocolo propuesto por el W3C para intercambiar los perfiles CC/PP de la forma más efectiva posible. Este protocolo está basado en el HTTP Extension Framework [10], un mecanismo genérico de extensión para HTTP/1.1, diseñado para ser compatible con las aplicaciones HTTP existentes. A pesar de que éste es un protocolo propuesto por el W3C para la transferencia de descripciones CC/PP, es totalmente independiente de CC/PP y no intenta ser el estándar para el intercambio de información CC/PP.

CC/PPex se basa en introducir nuevos campos en las cabeceras HTTP. Concretamente se añaden los siguientes:

- **Profile.** Este campo es una lista de referencias. Cada una representa un objeto correspondiente de la descripción CC/PP. Las referencias de la lista pueden ser URIs absolutas o una codificación MD5 en base64 del valor que se encuentra en un campo *Profile-diff* de la propia cabecera.

- **Profile-diff.** Empleado para indicar las diferencias en el perfil respecto al perfil anterior. Se permiten varios campos *Profile-diff* en una misma cabecera HTTP de petición. Obviamente, indicar en las cabeceras tan solo las diferencias del perfil respecto la última petición, resulta mucho más eficaz que enviar toda la descripción entera.

- **Profile-Warning.** Este campo pertenece a la cabecera de respuesta de la extensión de HTTP realizada. Se utiliza para transportar información de aviso, como por ejemplo las indicaciones de transformación de contenidos o no.

A continuación, se muestra un ejemplo de petición:

```
M-GET http://www.ejemplo.com HTTP/1.1
Host: www.w3.org
Man: "http://www.w3.org/1999/06/24-CCPPexchange";
ns=99
99-Profile:"http://www.aaa.com/hw", "http://
www.bbb.com/sw" , "1-uKhJE/AEeeMzFSejsYshHg=="
99-Profile-Diff-1: <?xml version="1.0"?>
<RDF xmlns="http://www.w3.org/TR/1999/PR-rdf-
syntax-19990105#" xmlns:PRF="http://www.w3.org/TR/
WD-profile-vocabulary#">
```

```
<Description ID="SoftwarePlatform"
PRF:Sound="On"/>
</RDF>
```

En este caso se trata de una petición obligatoria extremo-a-extremo, ya que se utiliza el método *M-GET (Mandatory GET)* y se incluye el campo *Man* en la cabecera. Esta petición indica un diferencial de perfil (*profile-diff*) y dos referencias indirectas. Para referirse al *profile-diff* se ha aplicado el algoritmo MD5 seguido de una codificación en base64 al valor del campo *profile-diff* y, finalmente insertar el número de *profile-diff* al inicio del resultado. Mediante este método cada diferencial de perfil queda totalmente identificado por este valor que se incluye en el campo *Profile*. Examinando tan solo este campo, proxies y gateways pueden realizar la búsqueda en la tabla de caché de manera más eficiente.

## UAPROF

User Agent Profile (UAProf) [11] es el estándar desarrollado por Open Mobile Alliance (OMA) [12], antiguo WAP forum. Esta especificación permite a los dispositivos WAP (Wireless Application Protocol) el intercambio de información sobre las preferencias y capacidades (User Agent Profile), entre el cliente WAP, elementos intermedios de la red y el servidor de origen. UAProf se puede entender como el primer desarrollo a gran escala de CC/PP. UAProf utiliza CC/PP para crear un vocabulario estándar que, actualmente, es usado por todos los dispositivos con la especificación 2.0 de WAP [13]. A parte de crear un vocabulario, la especificación de UAProf también incluye protocolos de intercambio de los perfiles y una codificación binaria para éstos, logrando así una transmisión eficiente de la información sobre el enlace inalámbrico.

UAProf se encuentra en un contexto de uso basado en una arquitectura extremo a extremo, tal y como puede comprobarse en la figura 5.

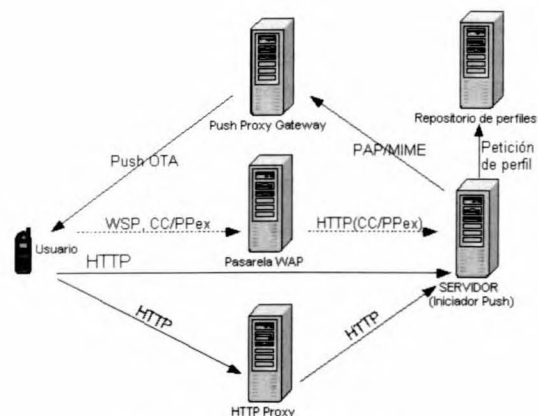


Figura 5. Arquitectura UAProf

Los dispositivos WAP pueden hacer peticiones al servidor de origen mediante la capa de protocolos WAP, o utilizando HTTP directamente sobre el enlace sin hilos, conocido como Wireless Profiled HTTP (W HTTP). Si el cliente se conecta utilizando la capa de protocolos WAP, se hace necesaria la introducción de una pasarela WAP para que realice la petición HTTP al servidor. Opcionalmente, como se observa en la figura 5, puede utilizarse el CC/PP Exchange Protocol (CC/PPex) para la transmisión del perfil.

El servidor es el encargado de indicar al Proxy WAP el envío de los contenidos al cliente sobre el enlace móvil (Over The Air, OTA), utilizando Push Access Protocol (PAP) [14]. La acción Push se utiliza en WAP porque el encargado de entregar los contenidos al cliente no es el servidor, siendo imposible la utilización de un protocolo *stateless*, a base de peticiones y respuestas, como HTTP. Gracias a esta acción, el proxy envía la respuesta al cliente sin que este último haya iniciado la petición al proxy.

Cuando se inicia una sesión WSP (Wireless Session Protocol), el cliente WAP introduce la información de sus preferencias y capacidades en los campos *profile* y *profile diff* de la cabecera de petición de conexión. La pasarela WAP, responde a la petición añadiendo el campo *profile warning* a la cabecera de la respuesta. Si este campo toma el valor 100 (OK), entonces la información del perfil del usuario es cacheada por la pasarela WAP y será válida para todo el tiempo de vida de la sesión. En la especificación de UAProf, se incluyen los métodos mediante los cuales el cliente puede modificar, suspender o reiniciar su perfil guardado en la sesión. La pasarela WAP, tal y como ocurría en CC/PP, puede añadir información sobre sus características y preferencias al perfil del usuario.

En el caso que la petición del cliente sea sobre W-HTTP, los nuevos campos en la cabecera en los que se transmite la información de las capacidades y preferencias (en caso de que se transmita) son: *x wap profile* y *x wap profile diff*. Mientras que el nuevo campo en la cabecera de las respuestas es *x wap profile warning*.

```
GET http://www.ejemplo.com HTTP/1.1
Host: localhost
x-wap-profile:"http://www.aaa.com/hw","1-uKhJE/AEeeMzFSejsYshHg=="
x-wap-profile-diff:1; <?xml version="1.0"?>
  <RDF xmlns="http://www.w3.org/TR/1999/PR-rdf-syntax-19990105#" xmlns:PRF="http://www.w3.org/TR/WD-profile-vocabulary#">
    <Description ID="SoftwarePlatform" PRF:Sound="On"/>
  </RDF>
```

En cambio, si el transporte de la información CC/PP se realiza sobre WSP, se añaden dos definiciones de nuevos campos en la cabecera de petición y un campo más en la cabecera de respuesta:

- **Profile**. Este campo tan solo contiene una URL (Uniform Resource Locator), que hace referencia a un perfil.

- **Profile-Diff**. Este campo contiene información sobre las preferencias y capacidades codificada en WBXML (Wireless Binary XML).

- **Profile-Warning**. Campo utilizado en las cabeceras de respuesta para indicar incidencias.

Cada petición puede tener varios campos *Profile* y varios *Profile-Diff*. Estas peticiones las realiza el cliente a la pasarela WAP, siendo ésta la que crea las peticiones HTTP, pasando de WSP a HTTP, utilizando la información de la petición y la información de las cabeceras en caché durante la sesión.

La sintaxis especificada para los nuevos campos es la siguiente:

- *Profile* = Número\_entero(WSP) URI.

El número entero de ocho bits indica el nombre del campo del perfil. Un ejemplo de este campo sería: *Profile: 0x05 http://www.aaa.com 0x00*. Donde *0x00* indica el final del string.

- *Profile-Diff* = Número\_entero(WSP) Longitud Perfil-CCPP.

Por ejemplo: *Profile-Diff: 0xB6 0x0A 0x01 0x05 0x04..... 0xB6* es el número que indica el nombre del campo de diferencial de perfil, la longitud es de diez octetos (*0x0A*) y a continuación se incluye la información CC/PP codificada en WBXML.

- *Profile-Warning* = Número\_entero(WSP) (Código\_de\_Aviso|(longitud Código\_de\_Aviso URI\_con\_warning Fecha)).

En el campo de aviso, si todo ha ido correctamente tan solo se encuentran los valores del número entero identificando al campo y el código de aviso correspondiente a 100:0x90. Mientras que si ha ocurrido algún error en alguna URI, ésta es indicada junto con su código de error y la fecha.

La pasarela WAP, al recibir la petición del cliente, la procesa y transforma a su equivalente HTTP, añadiendo la información CC/PP utilizando el CC/PP Exchange Protocol para indicar las capacidades y preferencias al servidor (ver figura 6). A parte de realizar esta transformación, también realiza caché de cabeceras para determinar los cambios que va realizando el cliente en su perfil. Durante el tiempo de vida de una sesión (WSP), la pasarela WAP realiza la

gestión de las cabeceras para conocer en todo momento el perfil del cliente.

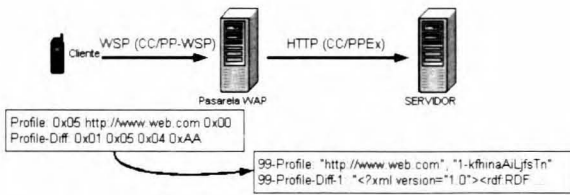


Figura 6. Transformación de CC/PP-WSP a CC/PPEx en la pasarela WAP

Como se ha comentado inicialmente, UAPProf proporciona los métodos para que el cliente modifique su perfil durante la sesión, sin necesidad de reiniciarla. La pasarela WAP realiza la caché de cabeceras para crear las peticiones HTTP correspondientes, utilizando información de las cabeceras en caché y de las peticiones que reciba del cliente

A diferencia de la recomendación CC/PP, en UAPProf sí se indican los pasos a seguir para la obtención, construcción y modificación de perfiles. En primer lugar, se obtienen los valores de los atributos indicados por sus URIs de referencia. A continuación, se obtienen los valores de los documentos indicados por los campos *Profile* y *Profile diff*. Finalmente, en caso de que aparezcan múltiples descripciones de un atributo, se determina su valor final mediante las reglas de resolución especificadas para cada atributo. Existen tres tipos:

- **Locked.** Indica que el valor que toma el atributo, es el primer valor donde aparece indicado.
- **Override.** La última descripción del atributo es el valor válido.
- **Append.** El valor final consiste en una lista de todas las ocurrencias del atributo en cuestión.

Si no existiera ninguna regla, los valores proporcionados por los diferenciales de perfil (*profile diff*), se sobrescriben sobre los anteriores.

## VOCABULARIO UAPROF

El vocabulario especificado por UAPProf está compuesto por seis tipos de componentes distintos:

- **HardwarePlatform.** En esta clase se encuentran las propiedades que describen adecuadamente las características hardware del terminal. Entre éstas se incluye, el tipo de dispositivos, tamaño de la pantalla, ...

- **SoftwarePlatform.** Grupo de propiedades que hacen referencia a la plataforma sobre la que opera el dispositivo. Proporciona información sobre el sistema operativo, los codificadores de video y audio soportados, y las preferencias de lenguaje del usuario.

- **BrowserUA.** Características del navegador web.

- **NetworkCharacteristics.** En este grupo se encuentran los atributos sobre la infraestructura de la red.

- **WapCharacteristics.** Características que pertenecen a las capacidades WAP soportadas por el dispositivo.

- **PushCharacteristics.** Propiedades push específicas que soporta el dispositivo. Por ejemplo, los tipos MIME soportados, el tamaño máximo de mensaje recibido,...

Todos los atributos definidos en el último esquema UAPProf se encuentran en:

<http://www.openmobilealliance.org/tech/profiles/ccppschemata-20030226.html>

## CONCLUSIONES

CC/PP se ha presentado como la solución estándar creada por el W3C para la descripción de las características de cualquier dispositivo. Gracias al hecho de estar basado en RDF, es posible la creación de vocabularios donde se encuentren definidos los atributos (propiedades) agrupados por componentes. En la recomendación no se define ningún vocabulario estándar, sino que se proporciona la plataforma para su desarrollo. Utilizando CC/PP, OMA ha creado UAPProf 1.1 como un vocabulario estándar para describir los teléfonos móviles. Aparte de la creación de un vocabulario propio, en UAPProf también se especifican unas reglas de resolución a la hora de la composición del perfil CC/PP, ya que la metodología a seguir para la formación del perfil no se define en la recomendación CC/PP.

Ambos estándares, CC/PP y UAPProf, exponen los distintos protocolos a utilizar para el transporte e intercambio de perfiles entre los dispositivos y elementos intermedios (como proxies y gateways). La principal premisa de diseño de los protocolos ha sido, obviamente, la optimización de la utilización del canal, ya que los enlaces inalámbricos utilizados para el acceso a Internet no poseen, de momento, un gran ancho de banda y, además, se dispone de baterías limitadas en los dispositivos. El W3C propone el CC/PP Exchange Protocol basado en la extensión de HTTP, como un protocolo que puede ser utilizado (no es el único) para este transporte. Mientras que UAPProf especifica el transporte de los perfiles mediante WSP o W-HTTP.

Por otra parte, gracias a RDF, CC/PP consigue ser extensible y flexible, pero esta flexibilidad de CC/PP se convierte en su principal problema. Si se quiere compatibilidad con el mayor número de dispositivos, no pueden existir tantos vocabularios como dispositivos o fabricantes. Por lo tanto es lógico que se piense en definir vocabularios estándares para la descripción de dispositivos del mismo tipo. Asimismo, las definiciones de las propiedades tienen que coincidir si una misma propiedad aparece en varios esquemas. Será necesaria la creación de vocabularios donde una propiedad determinada siempre tenga el mismo significado semántico, de lo contrario será imposible la correcta interpretación de descripciones de distintos dispositivos.

Para finalizar, remarcar la utilidad de CC/PP para su uso en los procesos de adaptación de contenidos, consiguiendo avanzar hacia el acceso universal a la Web, uno de los principales objetivos del W3C. Gracias a su definición, es posible la caracterización de cualquier tipo de dispositivo mediante propiedades definidas en esquemas, que son utilizadas por los servidores para realizar las adaptaciones de contenidos, sin necesidad de almacenar multitud de contenidos para los distintos tipos de dispositivos existentes. UAProf ha sido el primer gran desarrollo de CC/PP, ya implementado en los teléfonos con WAP 2.0. El proceso de elaboración de la recomendación final de CC/PP se ha prolongado durante 5 años, pero la aparición de ésta el pasado 15 de enero de 2004, sirve para que todas las empresas implicadas en la fabricación de dispositivos y desarrollo de aplicaciones independientes del dispositivo puedan implementar, sin controversias, CC/PP.

## BIBLIOGRAFÍA Y REFERENCIAS

- [1] Hypertext Transfer Protocol HTTP/1.1. Request for Comments 2616. <http://www.ietf.org/rfc/rfc2616.txt>
- [2] World Wide Web Consortium W3C. <http://www.w3.org>
- [3] Composite Capabilities/Preference Profiles (CC/PP): Structure and Vocabularies 1.0. W3C Recommendation 15 January 2004 <http://www.w3.org/TR/CCPP-struct-vocab/>
- [4] Resource Description Framework (RDF).
- [5] Extensible Markup Language (XML). <http://www.w3.org/XML/http://www.w3.org/RDF/>

- [6] Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation 22 February 1999. <http://www.w3.org/TR/REC-rdf-syntax>
- [7] Uniform Resource Identifiers (URI): Generic Syntax. Request for Comments 2396. <http://www.ietf.org/rfc/rfc2396.txt>
- [8] Composite Capabilities/Preference Profiles: Requirements and Architecture. <http://www.w3.org/TR/CCPP-ra>
- [9] CC/PP Exchange protocol base on HTTP Extension Framework. <http://www.w3.org/TR/NOTE-CCPPexchange>
- [10] H. Frystyk, P. Leach, S. Lawrence. HTTP Extension Framework. Internet Draft. <http://www.w3.org/Protocols/HTTP/ietf-http-ext/draft-frystyk-http-extensions-03.txt>
- [11] OMA/WAP Forum User Agent Profile 1.1 specification, version 20-October-2001 <http://www.wapforum.org/tech/documents/WAP-248-UAPProf-20011020-a.pdf>
- [12] Open Mobile Alliance. <http://www.openmobilealliance.org>
- [13] Wireless Application Protocol (WAP) 2.0 [http://www.openmobilealliance.org/tech/affiliates/wap/technical\\_wap2\\_0-20020813.zip](http://www.openmobilealliance.org/tech/affiliates/wap/technical_wap2_0-20020813.zip)
- [14] Wireless Application Protocol (WAP) Push Access Protocol Specification, Versión 29 April 2001. WAP-247-PAP-20010429-a

## AUTORES



José Luis Ferrer Riera,  
[jfer6322@alu-etsetb.upc.es](mailto:jfer6322@alu-etsetb.upc.es)

*Realizando PFC en el Grupo de Redes Inalámbricas del Departamento de Ingeniería Telemática (UPC) sobre la utilización de CC/PP y UAProf para la adaptación de contenidos web en entornos móviles.*



Anna Calveras Augé,  
[anna.calveras@entel.upc.es](mailto:anna.calveras@entel.upc.es)

*Dr. Ingeniero de Telecomunicaciones por la Universidad Politécnica de Cataluña (UPC). Profesor Titular de Universidad en el Departamento de Ingeniería Telemática. Forma parte del Grupo de Redes Inalámbricas de dicho departamento.*