EL64E
Redes de Computadores

Protocolos de aplicación

Protocolos de Aplicación

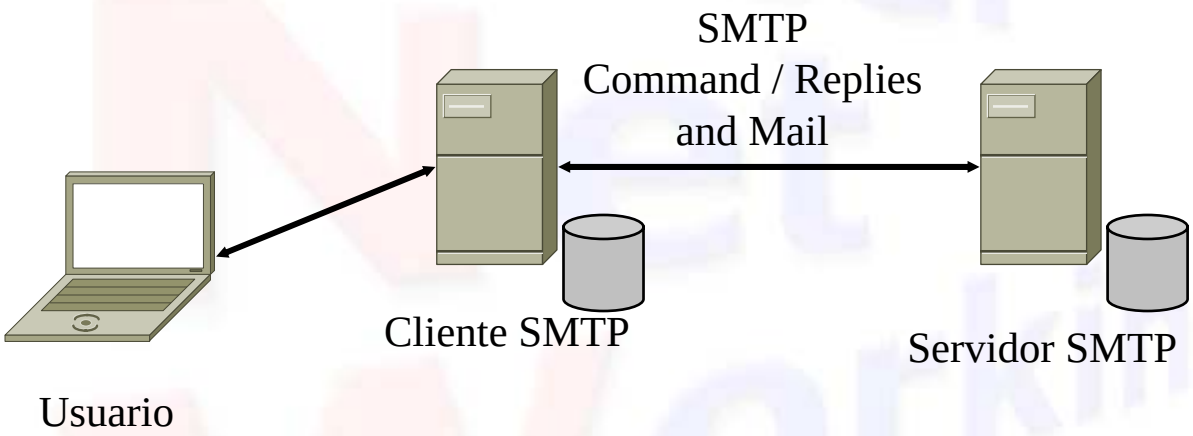
1. SMTP/POP3
2. DNS
3. HTTP
4. WAP
5. Protocolos de Señalización IP: SIP, MGCP
6. RTP

SMTP/POP3

SMTP

- SIMPLE MAIL TRANSFER PROTOCOL
- RFC 821
- Protocolo para el envío de e-mail
- SMTP establece un canal de transmisión bidireccional.
- El servido SMTP puede ser el destino final o uno intermedio.
- El cliente genera comandos SMTP al servidor que son respondidos para inyectarle los correos

Funcionamiento



Ejemplo de Conexión

MAIL FROM:<jsandova@ing.uchile.cl>
250 OK
RCPT TO: <el64e@ing.uchile.cl>
250 OK
RCPT TO: <superman@ing.uchile.cl>
550 No such user here
RCPT TO: <el65l@ing.uchile.cl>
250 OK
DATA
354 Start mail input; end with .
Blah blah blah...
...etc. etc. etc.
.
250 OK

Forwarding

- A veces la información de destino es incorrecta pero el servidor “sabe” cual es el destino correcto.

Ejemplo de Forward

MAIL FROM:<jsandova@ieee.org>

251 User not local; will forward to <jsandova@ing.uchile.cl>

En este caso el servidor es responsable de redirigir el mensaje al destino correcto

o

MAIL FROM:<jsandova@ieee.org>

251 User not local; please try <jsandova@ing.uchile.cl>

En este caso el server rechaza el correo y el cliente debe reenviarlo al destino correcto.

VERIFYING AND EXPANDING

- SMTP provee comandos para verificar un nombre de usuario o expandir una lista de mail.
  - Con el comando VRFY se pude enviar el nombre de usuario y recibir el nombre completo incluido su e-mail
  - Conel comando EXPN se puede enviar el nombre de una lista de distribución e correo y recibir el nombre completo de los usuarios y e-mail de los miembros de la lista.

Ejemplo de Verifying

VRFY jsandova

250 Jorge Sandoval <jsandova@ing.uchile.cl>

O

VRFY jsandova

251 User not local; will forward to <jsandova@ieee.org>

O

VRFY el643

550 String does not match anything.

O

VRFY el643

551 User not local; please try

O

VRFY Gourzenkyinplatz

553 User ambiguous.

Ejemplo de Expanding

EXPN el64e

250 Jorge Sandoval <jsandova@ing.uchile.cl>

250 Redes de Computadores <el64e@ing.uchile.cl>

250 Laboratorio LABNET <labnet@gorrion.die.uchile.cl>250

O

EXPN el64e

502 5.7.0 Sorry, we do not allow this operation

Algunos Comandos

HELLO <hostname>

MAIL FROM: <sender> [<parameters>]

RCPT TO: <recipient> [<parameters>]

DATA

RSET

VRFY

EXPN

HELP

NOOP

QUIT

TURN

Reply Codes By Function Groups

500 Syntax error, command unrecognized
[This may include errors such as command line too long]
501 Syntax error in parameters or arguments
502 Command not implemented
503 Bad sequence of commands
504 Command parameter not implemented

211 System status, or system help reply
214 Help message [Information on how to use the receiver or the meaning of a particular non-standard command; this reply is useful only to the human user]

220 <domain> Service ready
221 <domain> Service closing transmission channel
421 <domain> Service not available, closing transmission channel
[This may be a reply to any command if the service knows it must shut down]

Reply Codes By Function Groups (Cont)

250 Requested mail action okay, completed

251 User not local; will forward to <forward-path>

450 Requested mail action not taken: mailbox unavailable
[E.g., mailbox busy]

550 Requested action not taken: mailbox unavailable
[E.g., mailbox not found, no access]

451 Requested action aborted: error in processing

551 User not local; please try <forward-path>

452 Requested action not taken: insufficient system storage

552 Requested mail action aborted: exceeded storage allocation

553 Requested action not taken: mailbox name not allowed
[E.g., mailbox syntax incorrect]

354 Start mail input; end with <CRLF>.<CRLF>

554 Transaction failed

Ejemplo de envío de un e-mail

220 BBN-UNIX.ARPA Simple Mail Transfer Service Ready
HELO USC-ISIF.ARPA
250 BBN-UNIX.ARPA
MAIL FROM:<Smith@USC-ISIF.ARPA>
250 OK
RCPT TO:<Jones@BBN-UNIX.ARPA>
250 OK
RCPT TO:<Green@BBN-UNIX.ARPA>
550 No such user here
RCPT TO:<Brown@BBN-UNIX.ARPA>
250 OK
DATA
354 Start mail input; end with <CRLF>.<CRLF>
Blah blah blah...
...etc. etc. etc.
.
250 OK
QUIT
221 BBN-UNIX.ARPA Service closing transmission channel

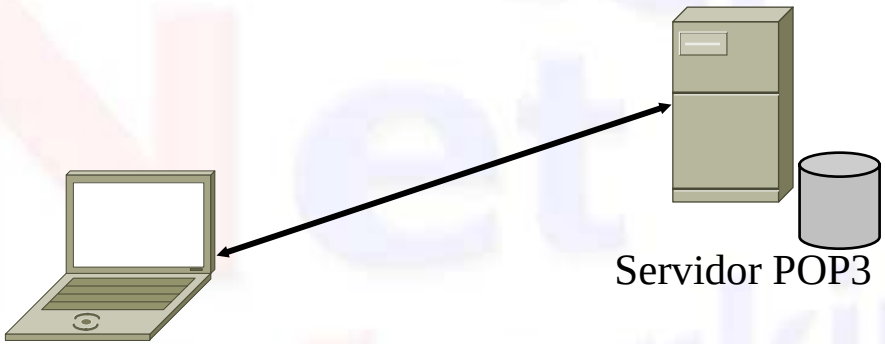
Ejemplo 2 de envío de un e-mail abortado

220 MIT-Multics.ARPA Simple Mail Transfer Service Ready
HELO ISI-VAXA.ARPA
250 MIT-Multics.ARPA
MAIL FROM:<Smith@ISI-VAXA.ARPA>
250 OK
RCPT TO:<Jones@MIT-Multics.ARPA>
250 OK
RCPT TO:<Green@MIT-Multics.ARPA>
550 No such user here
RSET
250 OK
QUIT
221 MIT-Multics.ARPA Service closing transmission channel

POP3

- Post Office Protocol - Versión 3
- RFC 1939
- Protocolo para lectura de e-mail
- POP3 establece un canal de transmisión bidireccional.
- El cliente genera comandos POP3 al servidor que son respondidos para la lectura los correos

Funcionamiento



Validación

- Ingreso Exitoso

  +OK Qpopper (version 4.0.3) at arrayan starting.

  USER jsandova

  +OK Password required for jsandova.

  PASS mipassword

  +OK jsandova has 13 visible messages (0 hidden) in 23450 octets.

- Ingreso fallido

  +OK Qpopper (version 4.0.3) at arrayan starting.

  USER jsandova

  +OK Password required for jsandova

  PASS 1234

  -ERR [AUTH] Password supplied for "jsandova" is incorrect.

Mensajes en el Buzón

- Al Ingreso

  +OK Qpopper (version 4.0.3) at arrayan starting.

  USER jsandova

  +OK Password required for jsandova.

  PASS mipassword

  +OK jsandova has 13 visible messages (0 hidden) in 23450 octets.

- Comando STAT
  - Indica numero de mensajes y largo total en byte
  - Ejemplo

  STAT

  +OK 13 23450

Información de mensajes en el Buzón

- Comando LIST
  - Indica largo total en byte de cada uno de los mensajes
  - Ejemplo

```
LIST
+OK 13 visible messages (23450 octets)
1 538
2 1499
3 957
4 2585
5 2074
6 3374
7 2271
8 2225
9 1464
10 1074
11 1283
12 1678
13 2428
.
```

Lectura de un Mensaje

- La Lectura se realiza utilizando el comndo RETR
- Ejemplo:

```
RETR 1
+OK 538 octets
Date: 16 May 2002 12:10:57 -0400
From: Mail System Internal Data <MAILER-DAEMON@ing.uchile.cl>
Subject: DON'T DELETE THIS MESSAGE -- FOLDER INTERNAL DATA
Message-ID: <1021565457@ing.uchile.cl>
X-IMAP: 0967564211 0000000751
X-UIDL: ='h"!394!!_aJ!!'GV!!
Status: RO

This text is part of the internal format of your mail folder, and is not
a real message.  It is created automatically by the mail system software.
If deleted, important folder data will be lost, and it will be re-created
with the data reset to initial values.

.
```

Lectura Parcial de un Mensaje

- La lectura parcial de un mensaje se realiza utilizando el comando RETR
- Ejemplo (leo el encabezado y 5 primeras líneas):

```
TOP 1 5
+OK Message follows
Date: 16 May 2002 12:10:57 -0400
From: Mail System Internal Data <MAILER-DAEMON@ing.uchile.cl>
Subject: DON'T DELETE THIS MESSAGE -- FOLDER INTERNAL DATA
Message-ID: <1021565457@ing.uchile.cl>
X-IMAP: 0967564211 0000000751
X-UIDL: ='h"!394!!_aJ!!'GV!!
Status: RO

This text is part of the internal format of your mail folder, and is not
a real message.  It is created automatically by the mail system software.
If deleted, important folder data will be lost, and it will be re-created
with the data reset to initial values.

.
```

Eliminación de un Mensaje

- La Eliminación se realiza utilizando el comndo DELE
- Ejemplo:

  DELE 1

  +OK message 1 deleted

Ejemplo de Conexión

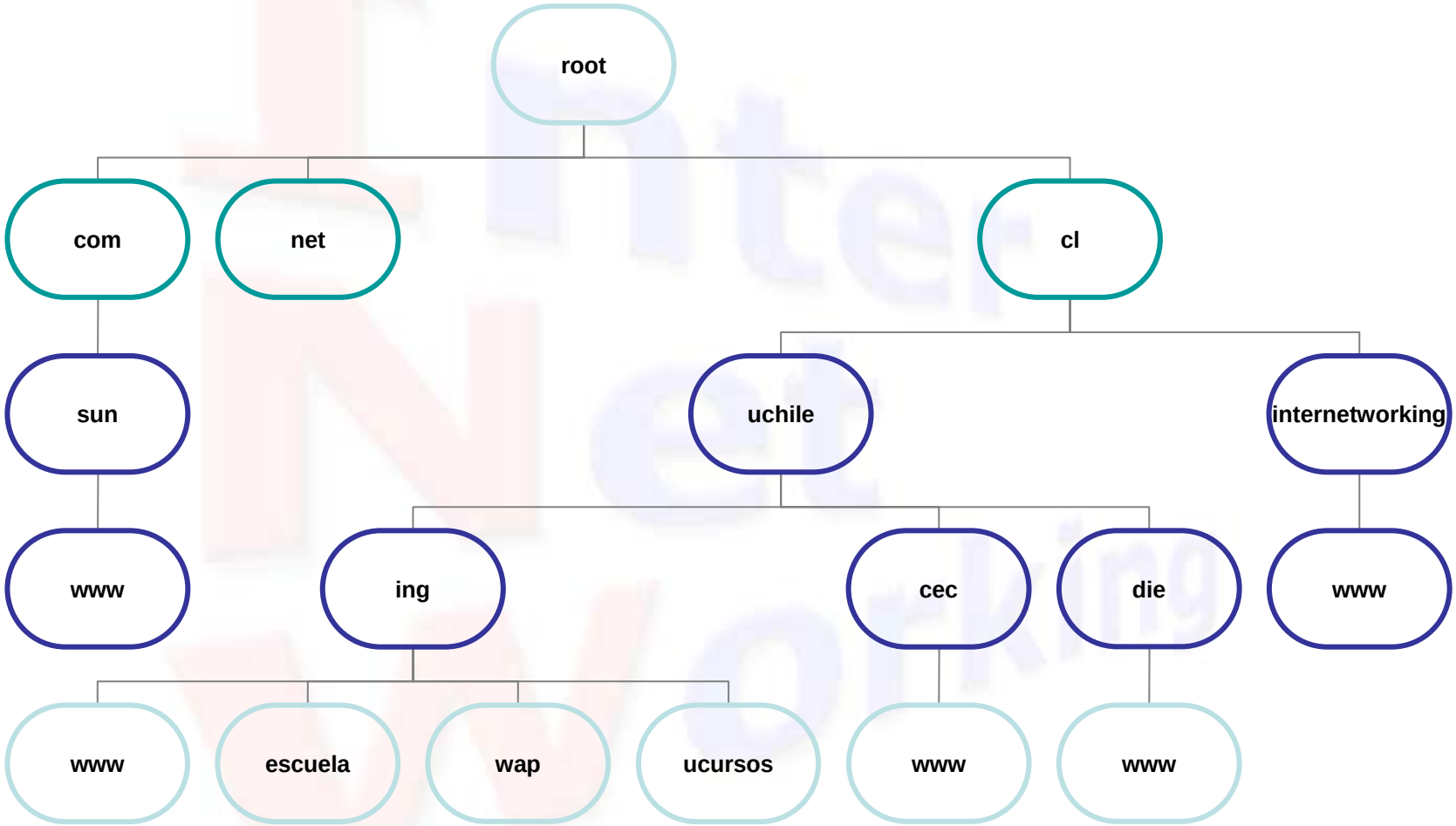
```
+OK Qpopper (version 4.0.3) at arrayan starting.
USER jsandova
+OK Password required for jsandova.
PASS mipassword
+OK jsandova has 13 visible messages (0 hidden) in 23450 octets.
STAT
+OK 13 23450
RETR 1
+OK 538 octets
Date: 16 May 2002 12:10:57 -0400
From: Mail System Internal Data <MAILER-DAEMON@ing.uchile.cl>
Subject: DON'T DELETE THIS MESSAGE -- FOLDER INTERNAL DATA
Message-ID: <1021565457@ing.uchile.cl>
X-IMAP: 0967564211 0000000751
X-UIDL: ='h"!394!!_aJ!!'GV!!
Status: RO

This text is part of the internal format of your mail folder, and is not
a real message.  It is created automatically by the mail system software.
If deleted, important folder data will be lost, and it will be re-created
with the data reset to initial values.

.
DELE 1
+OK message 1 deleted
QUIT
+OK Pop server at arrayan signing off.
```

DNS

DNS

- Aplicación distribuida que permite asociar un nombre a una dirección IP.
- Posee una estructura jerárquica.
- Delegación de autoridad
- BIND. Berkeley Internet Name Domain
- Conceptos:
  - Servidor Primario. Actualiza
  - Servidor Secundario. Informa
  - Servidor Cache. Informa sin autoridad

Estructura jerárquica



Delegación de Autoridad

- Se delegan a subdominios definiendo un record NS.
- Las preguntas sobre hosts de un subdominio son derivadas al DNS del subdominio

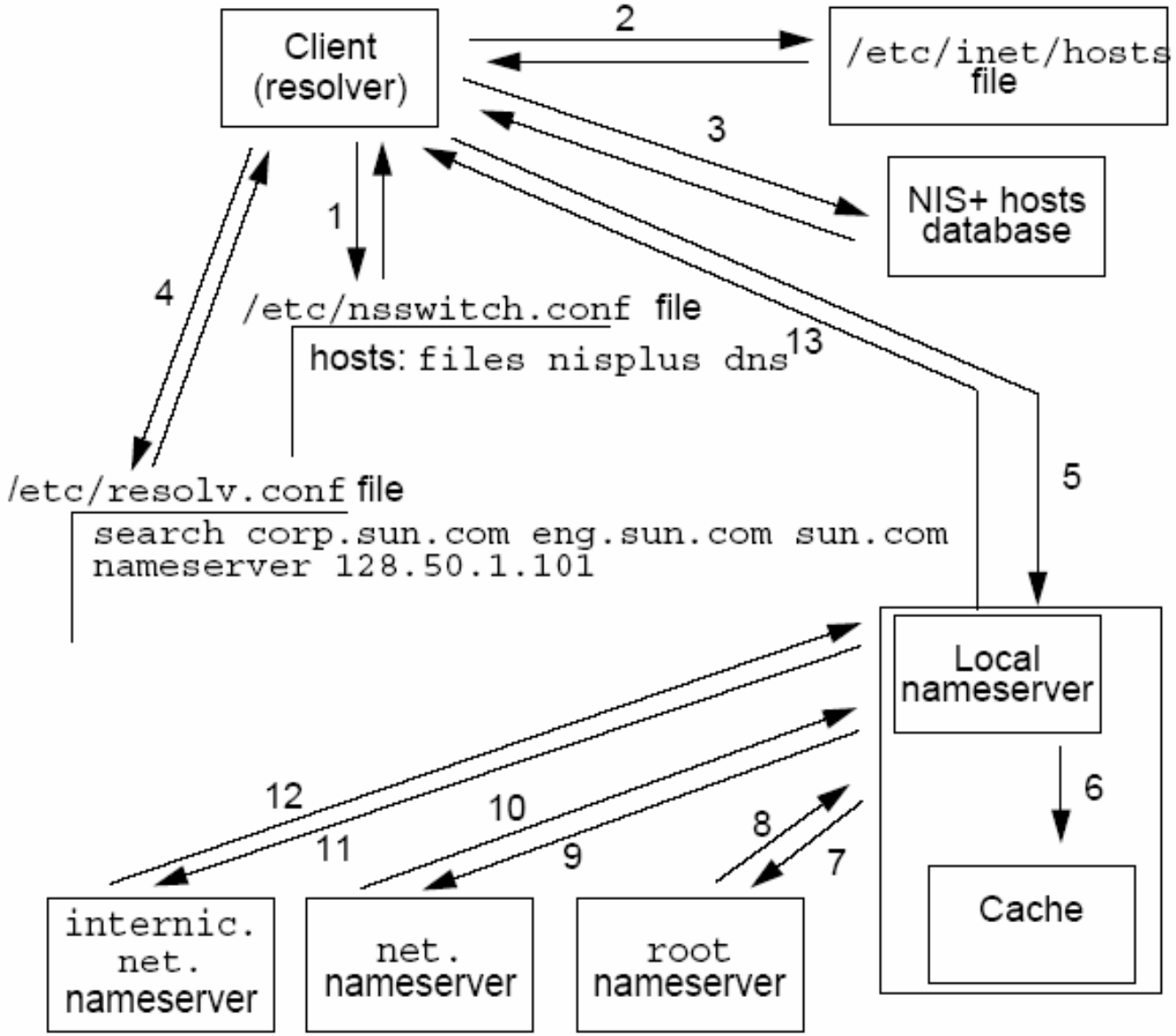
Tipos de Registros

- A Address Record. Para definir un host
- CNAME Canonical Name. Usado para definir una alias de un host.
- NS Name Service. Para definir un DNS
- MX Mail Exchange. Para definir un servidor SMTP
- PTR para definir un dominio inverso
- SOA Start of Authority

SOA Start of Authority

- **Serial Number.** Habitualmente en formato yyyymmddhh para indicar modificación. Se asume una nueva versión si este número se incrementa.
- **Refresh**. Cada cuanto tiempo el secundario refresca.
- **Retry**. Cada cuanto tiempo reintenta si la actualización del secundario fracaza.
- **Expire**. Indica cuanto tiempo puede estar la copia en el secundario sin refrescar.
- **TTL**. Tiempo que permanecerán los registros en los caches de los DNSs.

Ejemplo de resolución de nombres

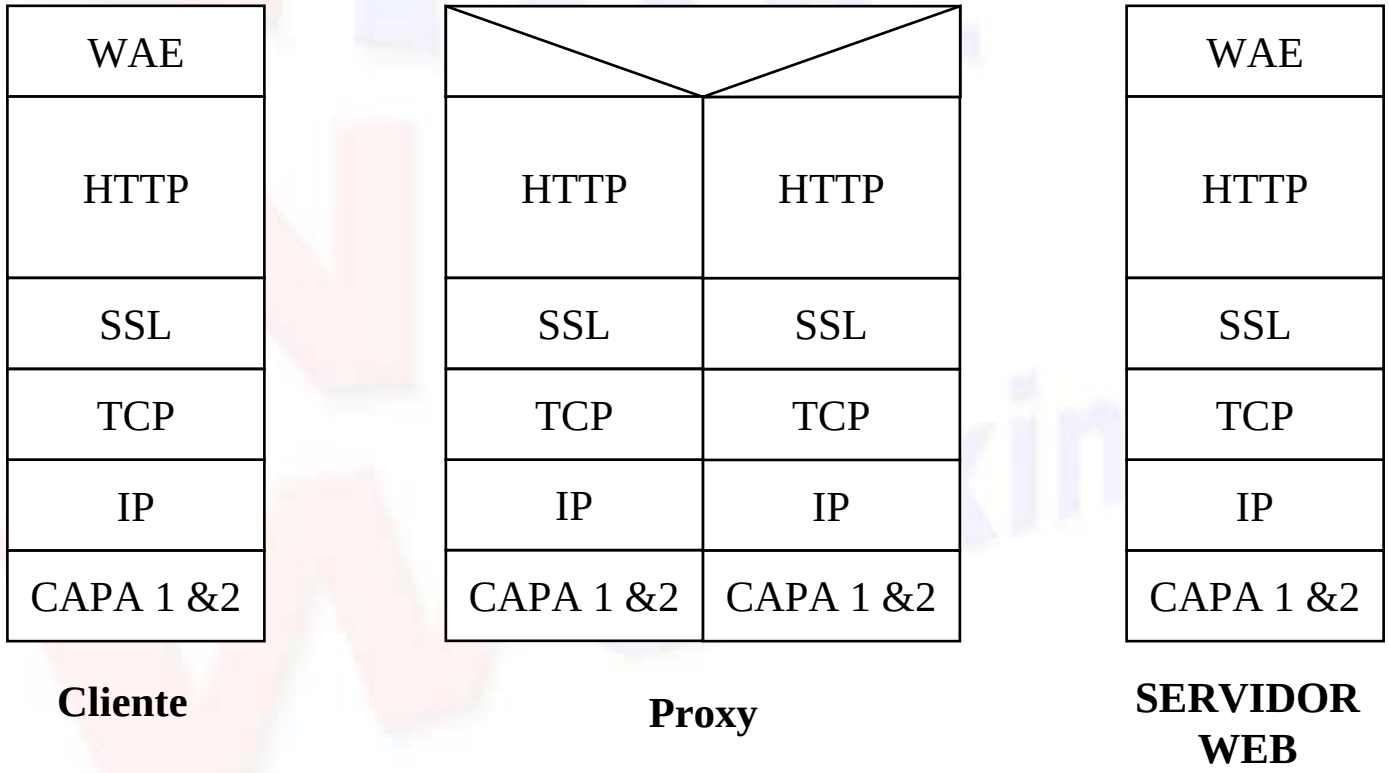


HTTP

HTTP

- HyperText Transfer Protocol
- RFC 2616
- Protocolo pregunta-respuesta

Stack de protocolos



URL

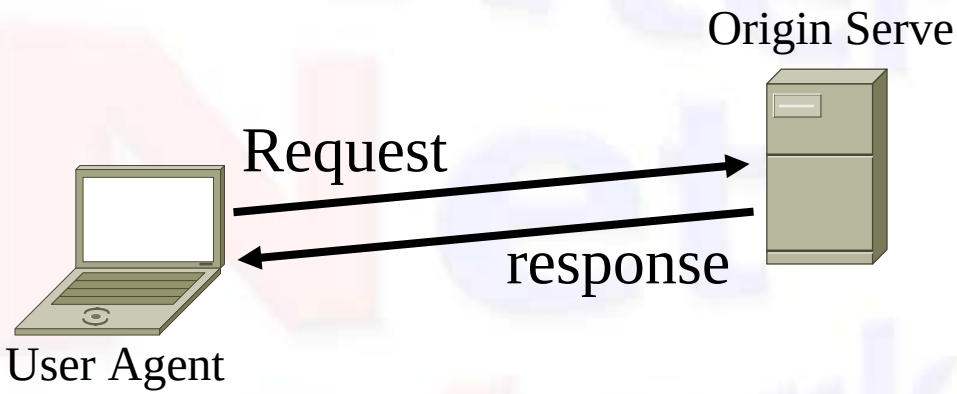
- Formato URL

  http_URL = "http:" "//" host [":" port] [abs_path ["?" query]]

Ejemplo

```
%telnet www.cec.uchile.cl 80
GET http://www.cec.uchile.cl/~jsandova HTTP/1.0

HTTP/1.1 301 Moved Permanently
Date: Thu, 23 May 2002 15:47:19 GMT
Server: Apache/1.3.11 (Unix)
Location: http://www.cec.uchile.cl/~jsandova/
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML><HEAD>
<TITLE>301 Moved Permanently</TITLE>
</HEAD><BODY>
<H1>Moved Permanently</H1>
The document has moved <A
    HREF="http://www.cec.uchile.cl/~jsandova/">here</A>.<P>
</BODY></HTML>
```

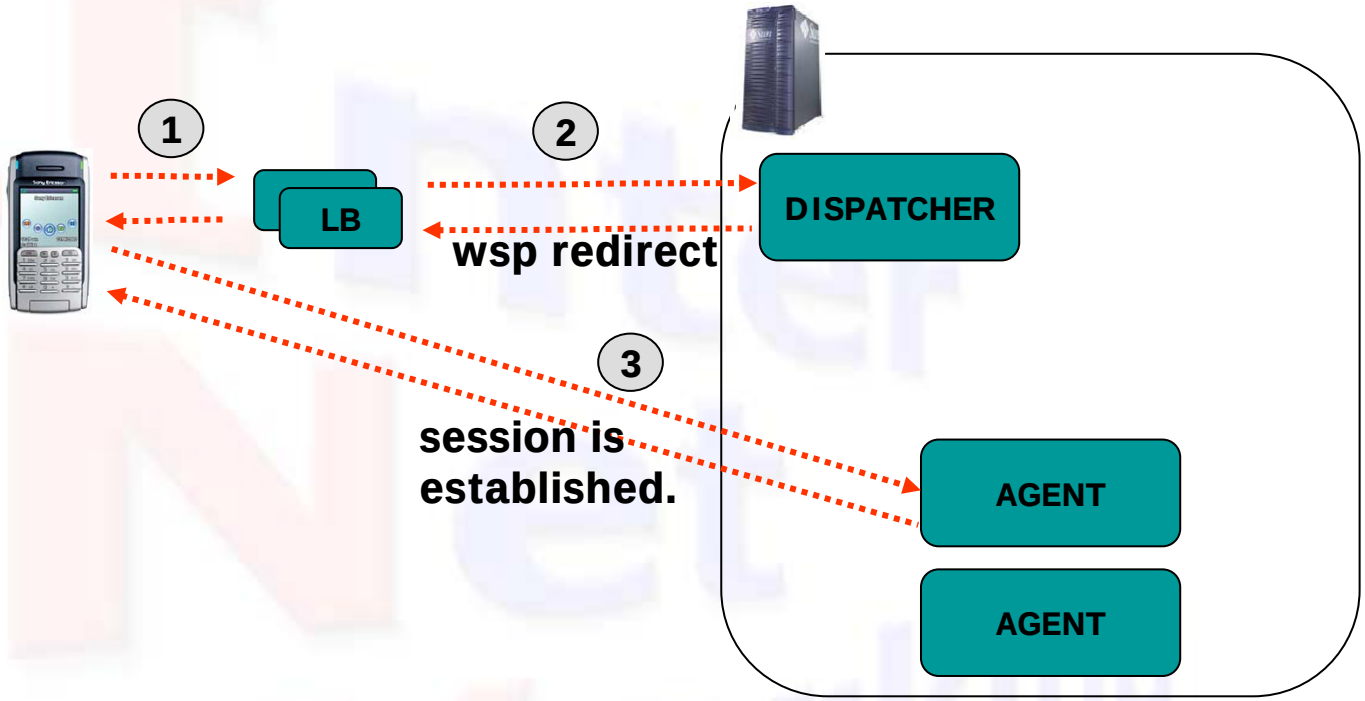
Ejemplo de Conexión



Status Code Definitions

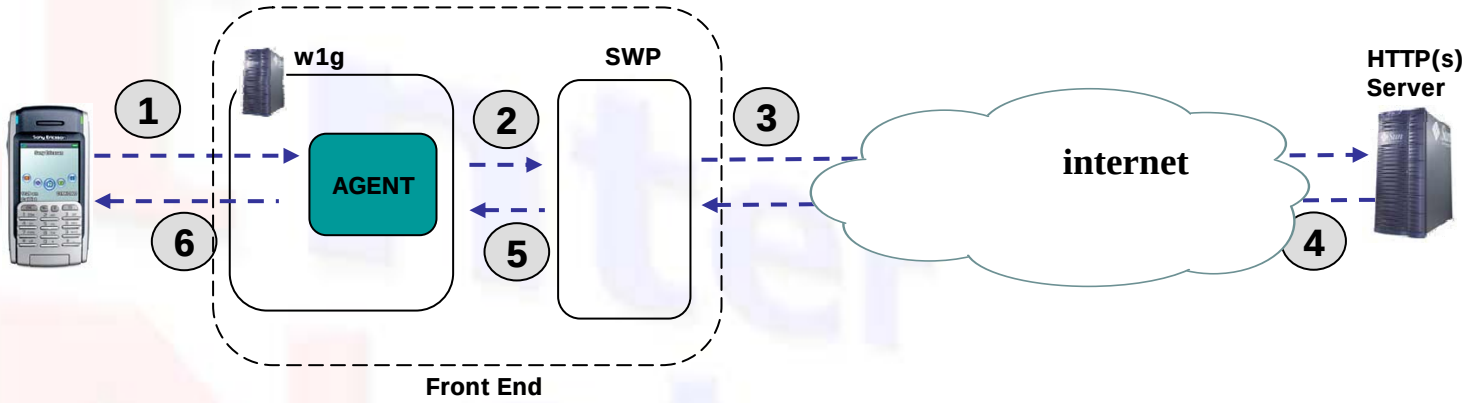
- Informational 1xx
- Successful 2xx
- Redirection 3xx
- Client Error 4xx
- Server Error 5xx

Detalles Establecimiento de Sesión



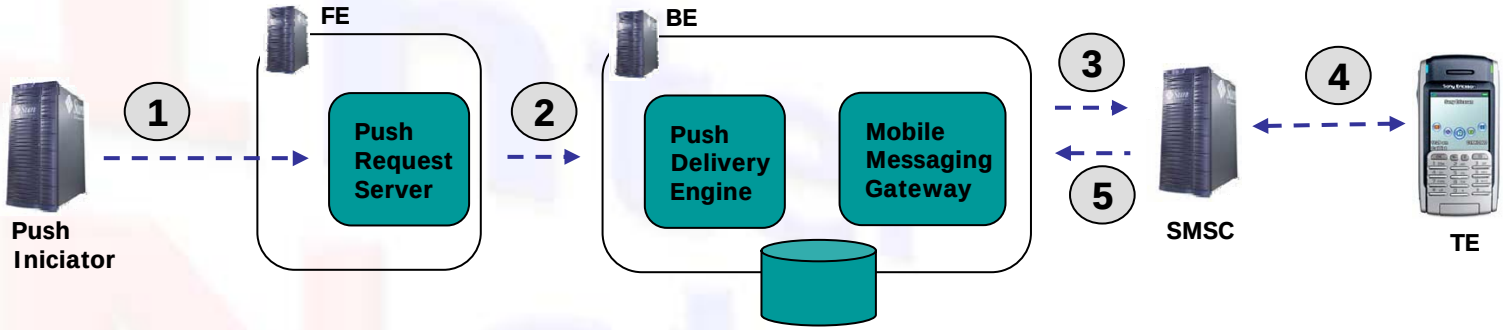
- El terminal se comunica con la dirección del dispatcher en el LB la que es enviada al FE de acuerdo a reglas de balanceo.
- El dispatcher redirige la conexión al Agent que lo atenderá

Detalles Conexión General



- El móvil solicita el contenido al w1g a través del agent asignado.
- El agent solicita el contenido al http server a través del swp (Smart Wireless Proxy).
- La validación del cliente se realiza hacia el BD situada en el BE

Funcionamiento PPG



- El PI inyecta el mensaje push al PPG a través de los FEs. Habitualmente los PI son: MMSc, Campañas Push, Proveedor de Contenido, etc.
- El PPG lo procesa y lo envia a través del MMG hacia el SMSC.

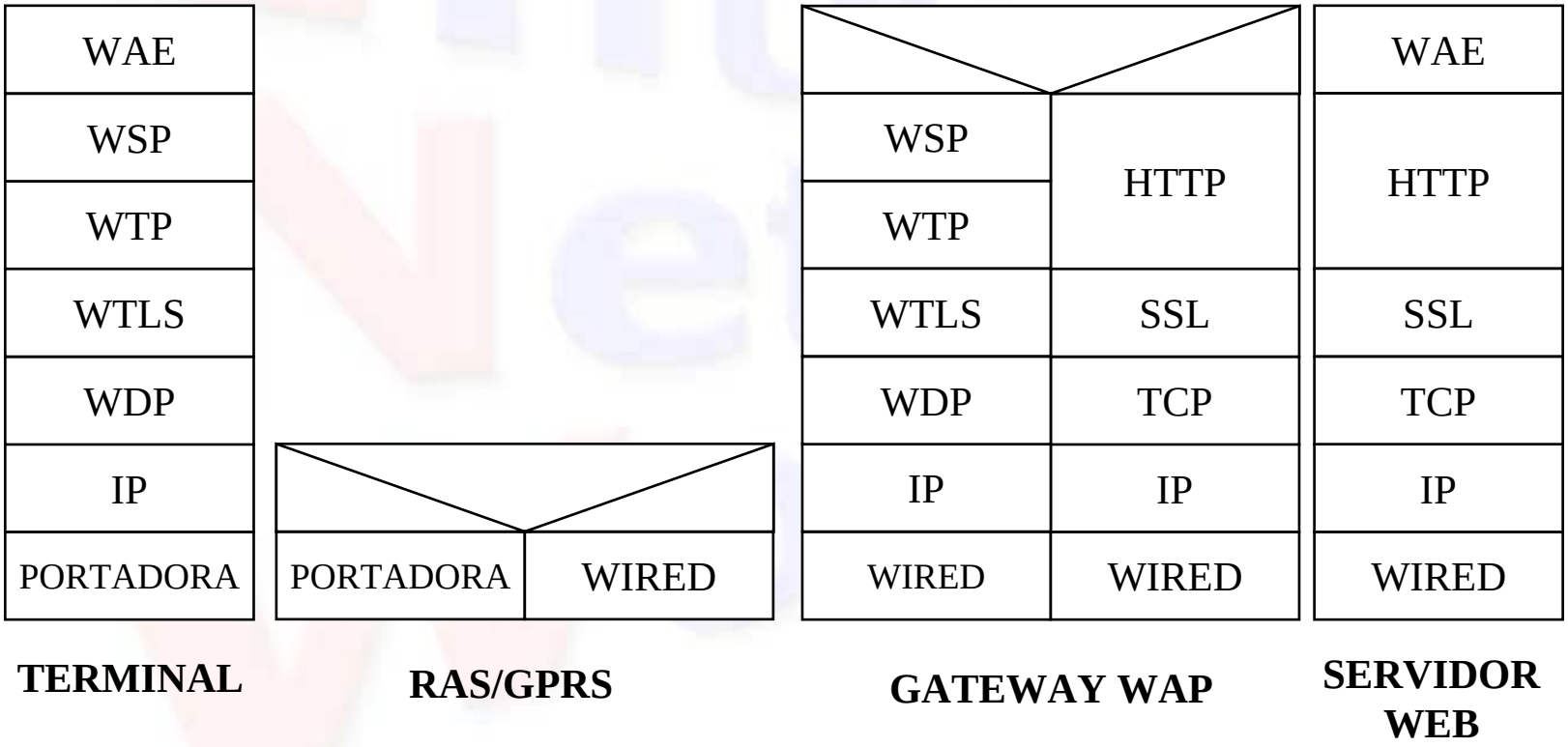
Puertos

- HDTP
  - Dispatcher udp 1905
  - Agent udp 1910 - 1949
- WAP1
  - Unsecure ConnectionLess
    - Dispatcher udp 9200
    - Agent udp 39200-39249
  - Unsecure ConnectionOriented
    - Dispatcher udp 9201
    - Agent udp 49200-49249
  - Secure ConnectionOriented
    - Dispatcher udp 9203
    - Agent udp 49250-49299
- WAP2
    - Unsecure tcp 80, 8080
    - Secure tcp 443

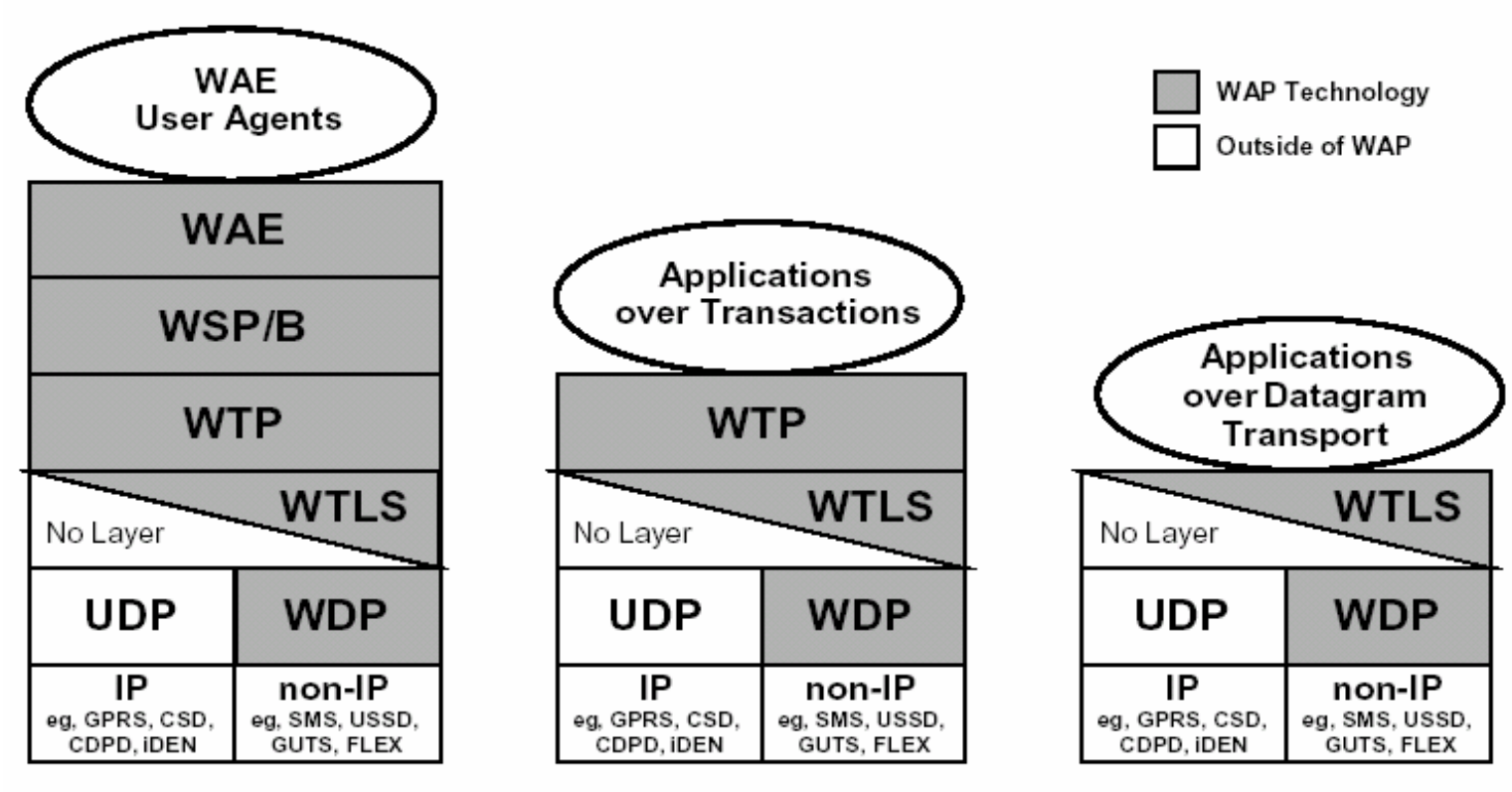
Puertos

- Wap Push PI Access
  - Private PI Pap Unsecure tcp 9001
  - Public PI Pap Unsecure tcp 9002
  - Public PI Pap Secure tcp 9003
  - HTTP Confirmed Push tcp 4035
- Radius Accounting
  - Radius Accounting udp 12080
- SMSC
  - SMPP al SMSC tcp 15001

Stack de protocolos



Stack de protocolos



Stack de Protocolos cont

- WAE: Wireless Application Enviroment
- WSL: Wireless Session Layer
- WTL: Wireless Transsaction Layer
- WTLS: Wireless Transaction Layer Security
- WDP: Wireless Datagram Protocol

XML

What is XML?

- E**X**tensible **M**arkup **L**anguage
- XML is a **markup language** much like HTML.
- XML was designed to **describe data**.
- XML tags are not predefined in XML. You must **define your own tags**.
- XML is **self describing**.
- XML uses a DTD (**Document Type Definition**) to formally describe the data

The main difference between XML and HTML

- XML is not a replacement for HTML.
- XML and HTML were designed with different goals:
- XML was designed to describe data and to focus on what data is.
- HTML was designed to display data and to focus on how data looks.
- HTML is about displaying information, XML is about describing information.

XML is extensible

- The tags used to markup HTML documents and the structure of HTML documents are predefined. The author of HTML documents can only use tags that are defined in the HTML standard.

- XML allows the author to define his own tags and his own document structure.

XML is a complement to HTML

- It is important to understand that XML is not a replacement for HTML. In the future development of the Web it is most likely that XML will be used to structure and describe the Web data, while HTML will be used to format and display the same data.

XML in future Web development

- We have been participating in XML development since its creation. It has been amazing to see how quickly the XML standard has been developed, and how quickly a large number of software vendors have adopted the standard.

- We strongly believe that XML will be as important to the future of the Web as HTML has been to the foundation of the Web. XML is the future for all data transmission and data manipulation over the Web.

An example XML document

```
<?xml version="1.0"?>
<note>
<to>EL64E</to>
<from>Jorge Sandoval</from>
  <heading>Examen</heading>
<body>Recuerden que no hay
  eximición!</body>
</note>
```

Use of Elements vs. Attributes

- Take a look at these examples:
- Using an Attribute for sex:

```
<person sex="female">
   <firstname>Anna</firstname>
   <lastname>Smith</lastname>
</person>
```

- Using an Element for sex:

```
<person> <sex>female</sex>
   <firstname>Anna</firstname>
   <lastname>Smith</lastname>
</person>
```

SIP / MGCP

Voz / Multimedia sobre IP

Session Initiation Protocol (SIP)
IETF RFC 2543

SIP – Session Initiation Protocol

- Protocolo de señalización de sesiones cliente-servidor
  - Objetivo principal es proveer presencia y movilidad
  - Primitivas: establecer sesión, terminar sesión, cambios
- Sobre SIP se pueden definir servicios arbitrarios:
  - Redireccionar llamadas de números desconocidos a la secretaria
  - Responder con una página web si no se está disponible
  - Al recibir una invitación, mandar una imagen JPEG
- Características
  - Codificación textual (compatible con telnet, tcpdump)
  - Programable

SIP – Protocolo Genérico de Presencia

- SIP no está limitado a Telefonía IP
  - SIP sirve para establecer presencia de usuarios
  - Mensajes SIP pueden transportar contenido de señalización arbitrario: mensajería instantánea, imágenes, cualquier tipo MIME
- Para aplicaciones que tengan noción de sesión
  - Juegos en red (Quake II/III, etc.)
  - Video conferencia
  - Sistemas de realidad virtual distribuidos
- Aplicaciones pueden hacer uso de la infraestructura SIP
  - Presencia en mensajería instantánea
  - SIP para appliances

Historia de SIP

- 02/96: draft-ietf-mmusic-sip-00: 15 páginas ASCII, sólo un tipo de requerimiento
- 12/96: -01: 30 páginas, 2 tipos de requerimientos
- 01/99: -12: 149 páginas, 6 métodos
- 03/99: RFC 2543, 153 páginas, 6 métodos
- 11/99: Se forma SIP WG
- 11/00: draft-ietf-sip-rfc2453bis-02, 171 páginas, 6 métodos
- 12/00: la cantidad de trabajo del WG se torna inmanejable
- 04/01: se divide en dos WG: SIP y SIPPING
- 2001: implementaciones SIP disponibles:
  - http://www.columbia.edu/~hgs/sip/implementation.html
  - http://www.pulver.com/sip/products.html

Servidores SIP

- Servidor Proxy SIP
  - Intermedia señalización de llamadas (puede actuar como cliente y servidor)
  - Funciona transaccionalmente (no requiere mantener estado de sesión)
- Servidor de Redireccionamiento SIP
  - Redirecciona llamadas a otros servidores
- Servidor de Registro SIP
  - Acepta pedidos de registro de usuarios
  - Mantiene ubicación del usuario en un Servidor de Ubicación (como el HLR en redes celulares)

Direcciones SIP

- SIP entrega direcciones globales
  - Quien espera ser llamado usa el método SIP REGISTER para asociarse a una dirección
  - Quien llama usa esa dirección para establecer comunicaciones en tiempo-real con alguien quien espera ser llamado
- SIP usa URLs
  - sip:jiri@iptel.org
  - sip:voicemail@iptel.org?subject=llamame
  - sip:ventas@hotel.cl; geo.position:=48.54_-123.84_120
- Debe incluir host, puede incluir nombre de usuario, número de puerto, parámetros como transporte, etc.
- Puede ser parte de página web, tarjeta de visita, etc.
- Espacio de direcciones ilimitado
- Acepta otras URLs (HTTP, mailto:, etc)

Dispositivos Terminales SIP

- Agente de Usuario (UA)
  - Cliente UA (origina llamada)
  - Servidor UA (espera que lo llamen)
  - Puede ser HW o SW

Métodos SIP según RFC 2543

- **INVITE** inicia una sesión
  - Descripción de la sesión incluida en el mensaje
- **ACK** confirma el establecimiento de la sesión
  - Sólo se puede usar con INVITE
- **BYE** termina la sesión
- **CANCEL** cancela un INVITE pendiente
- **OPTIONS** para consultar capacidades
- **REGISTER** asocia una dirección permanente a la ubicación actual. Puede contener datos del usuario (scripts CPL)

Métodos Extendidos de SIP

- **INFO** señalización en medio de la llamada para mandar información fuera de banda (RFC 2976)
  - Para mandar dígitos marcados en el teléfono
- **COMET** se cumple con precondición (draft-ietf-sip-manyfolks-resource)
- **PRACK** Provisional reliable Responses ACK (draft-ietf-sip-100rel)
- **SUBSCRIBE** para mensajería instantánea
- **NOTIFY (**draft-rosenberg-impp-*)

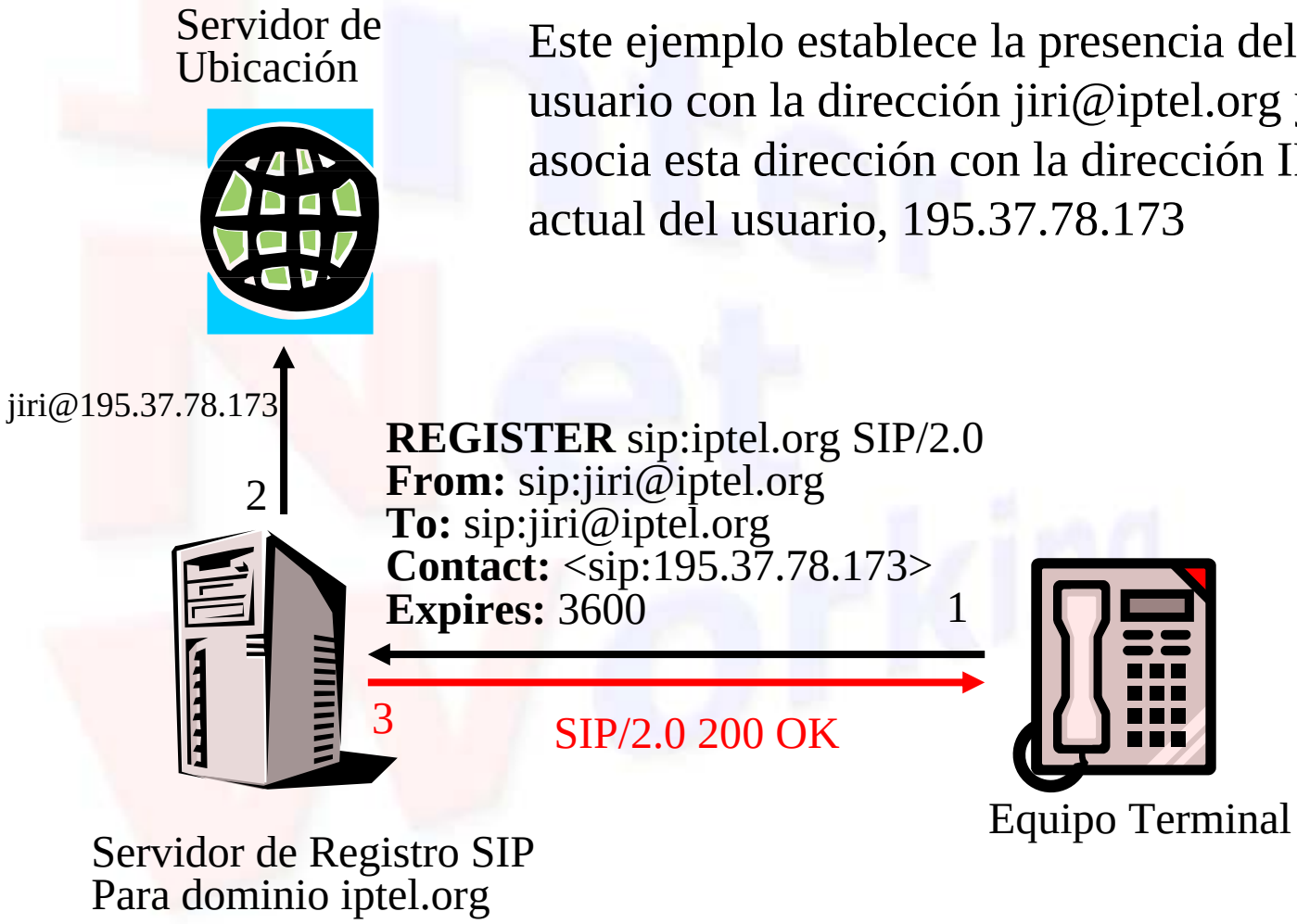
Códigos de Respuestas de SIP

- Prestado de HTTP: xyz texto explanatorio.
- Requiere que receptor entienda “xyz”.
- Códigos x80 y superiores para evitar conflictos con códigos de respuesta de HTTP en el futuro.
- 1yz Información
  - 100 Intentando
  - 180 Sonando (procesado localmente)
  - 181 La llamada ha sido desviada
- 2yz Éxito
  - 200 OK
- 3yz Redirección
  - 300 Múltiples alternativas
  - Traslado permanente
  - Traslado temporal

Códigos de Respuestas de SIP

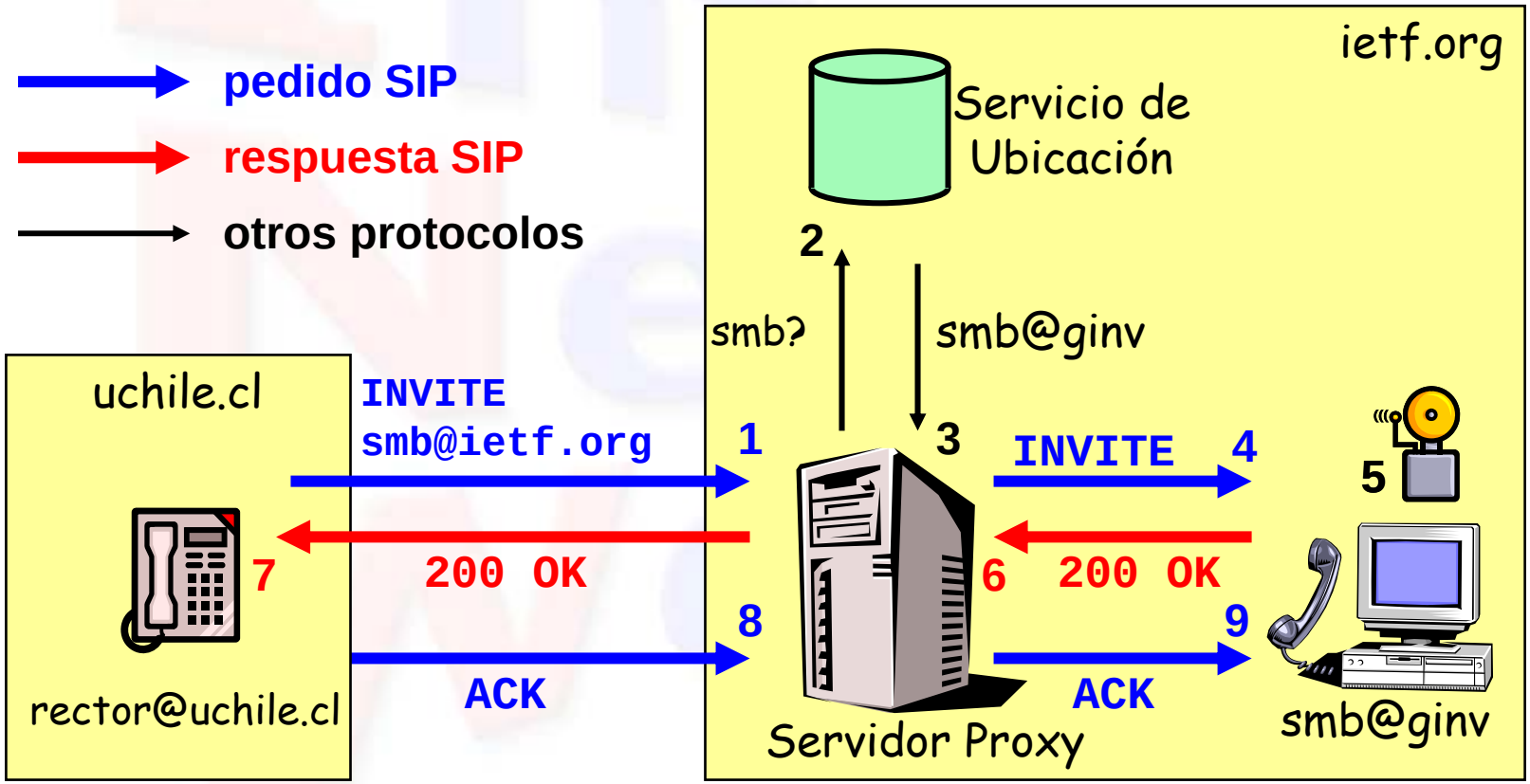
- 4yz Error en el cliente
  - 400 Requerimiento malo
  - 401 No está autorizado
  - 482 Loop detectado
  - 486 Ocupado aquí
- 5yz Falla en servidor
  - 500 Error interno en el servidor
- 6yz Falla Global
  - 600 Ocupado en todas partes

Registro en SIP



Este ejemplo establece la presencia del usuario con la dirección jiri@iptel.org y asocia esta dirección con la dirección IP actual del usuario, 195.37.78.173

Invitación SIP
usando un Servidor Proxy SIP



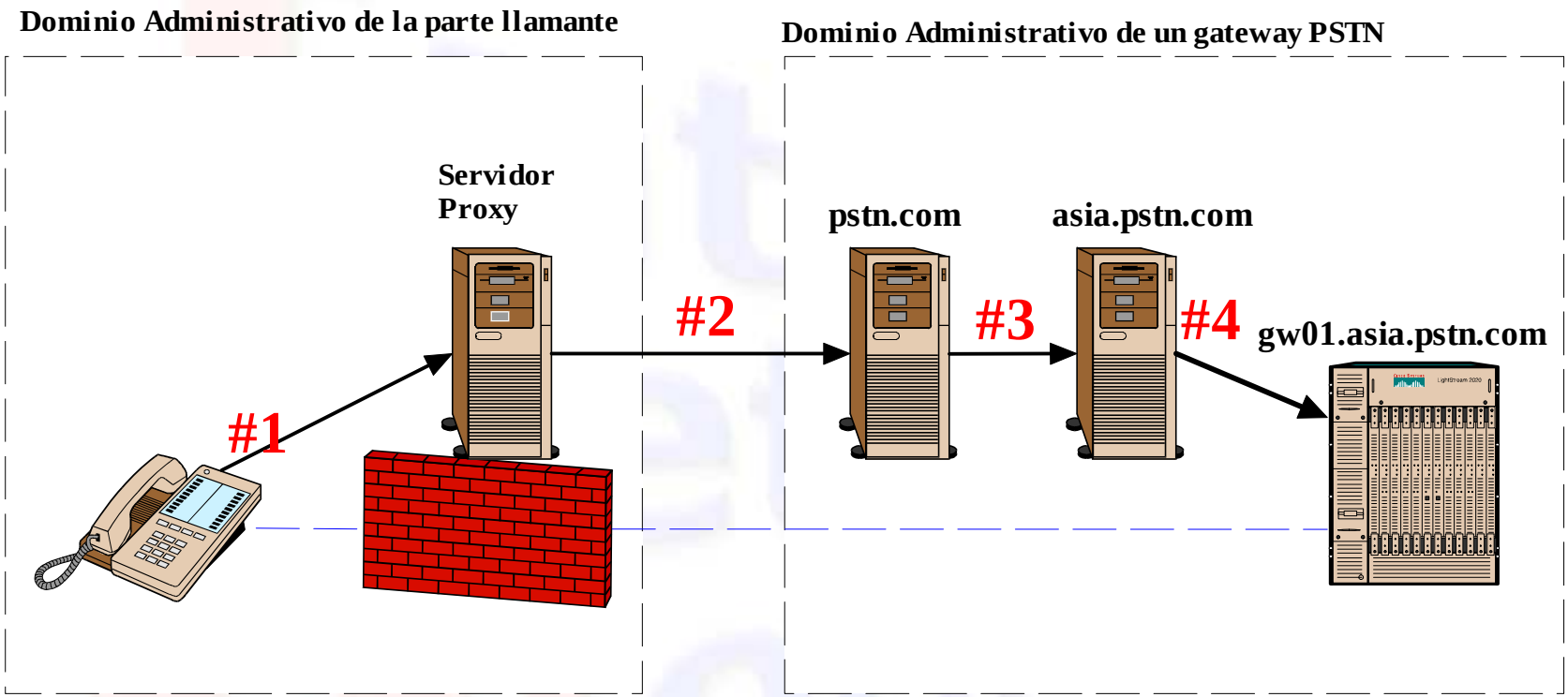
Funcionalidad de Servidor Proxy SIP

- Sirve como punto de encuentro global para llegar a otros equipos terminales.
- Debe enrutar, determinando dónde mandar la señalización (UA/otro proxy/redirección).
- Permite que la función de enrutamiento sea progamable.
- Puede probar varios destinos secuencial o paralelamente.
- Permite agregar lógica arbitraria sobre el protocolo:
  - Preferencias de señalización del usuario
  - AAA
  - Control de firewall
  - Etc.

Encadenamiento de Proxy

- En algunos casos, se puede requerir otro proxy para llamadas salientes
  - Para proveer lógica al procesar llamadas
  - Necesidad de firewall
  - Servicio de ruta de menor costo
  - Teléfonos IP deben conocer dirección del proxy
    - Puede configurarse a mano o se puede obtener con algún protocolo de configuración como DHCP o TFTP.
- Servidores pueden encadenarse arbitrariamente
  - Un servidor corporativo central puede distribuir señalización a servidores departamentales
  - Un usuario podría querer que llamadas a su anexo suenen en su celular
- Servidores deben evitar loops infinitos

Ejemplo de Encadenamiento de Proxy



El proxy de salida de la parte llamante sale a través del firewall

El proxy de entrada en el destino identifica al proxy del país/área que está siendo llamado

El proxy en el área de destino distribuye la carga a uno de los gateways en una granja

Proxy vs. Redirección SIP

- Un servidor SIP puede actuar como proxy para un requerimiento o bien redireccionarlo
- Cuál de los métodos aplica es cuestión de configuración. Puede ser determinado estática o dinámicamente (CPL).
- Redirección es útil cuando un usuario se muda o cambia de proveedor (PSTN: “el número que Ud. marcó no está disponible”).
- Proxy es útil si se requiere AAA, control de firewall. En general, el servidor proxy tiene mayor control.

Session Description Protocol (SDP)

- Provee información suficiente para participar en una sesión multimedia
- SDP incluye la descripción de:
  - Codecs a usar
  - Destino de la información (dirección IP y puerto)
  - Nombre y propósito de la sesión
  - Tiempo que la sesión está activa
  - Información de contacto
- Es, en realidad, un formato de datos y no un protocolo

Ejemplo de SDP

```
v=0
o=jperez 28908044538 289080890 IN IP4 193.175.132.118
s=Tutorial SIP
e=jperez@die.uchile.cl
c=IN IP4 146.83.4.11
t=28908044900 28908045000
m=audio 49170 RTP/AVP 0 98
a=rtpmap:98 L16/11025/2
```

Principios de Diseño de SIP

- Sigue el modelo de estado de IP
  - Más inteligencia y estado en dispositivos terminales
  - El core de la red a lo más mantiene estado transaccional
  - La periferia de la red puede mantener estado de sesiones
  - Beneficios: bajo consumo de memoria y CPU en servidores, alta confiabilidad y escalabilidad sin punto único de falla
- Soporte de UDP
  - Menor tiempo de establecimiento, menos estado que mantener
- INVITEs son idempotentes (datos no pueden estar repartidos en múltiples requerimientos)

Integración con otros protocolos IETF

- SIP fue diseñado para interactuar con otros protocolos definidos por la IETF. Por ejemplo:
  - RSVP – para reservar recursos de la red.
  - RTP (Real-Time Protocol) – para transportar datos en tiempo real y proveer feedback de calidad de servicio.
  - RTSP (Real Time Streaming Protocol) – para controlar la entrega de contenido multimedia.
  - SAP (Session Advertisement Protocol) – para publicar sesiones multimedia vía multicast.
  - SDP (Session Description Protocol) – para describir sesiones multimedia.
  - MIME (Multipurpose Internet Mail Extension) – estándar de facto para describir contenido en Internet.
  - COPS (Common Open Policy Service)

Comparando SIP con H.323

- SIP y H.323 son similares en cuanto a funcionalidad. Ambos ofrecen:
  - Control de llamadas, establecimiento y término de llamadas.
  - Servicios básicos como llamada en espera, transferencia de llamada, caller id, etc.
  - Intercambio de capacidad de los terminales.

Fortalezas de H.323 y SIP

- H.323 sirve para conferencias multimedia sofisticadas, permitiendo aplicaciones como pizarras compartidas, video conferencias, aplicaciones que compartan datos.
- SIP permite crear aplicaciones fácilmente con SIP-CGI (Common Gateway Interface) y SIP-CPL (Call Processing Language)
- SIP permite que el control de la llamada lo tome un tercero. Se está trabajando para agregarle esta funcionalidad a H.323

H.323 vs. SIP

	H.323 (ITU-T)	SIP (IETF)
Adopción	Amplia	Creciente
Señalización	Codificación binaria compleja	Codificación textual simple
Descripción de medios	Sub-protocolos: H.245, H.225, Q.391, RAS, H.450	SDP (tipos y direcciones)
Servidores	Gatekeeper H.323	Roles: Proxy, Registrador, ...
Capa de Transporte	RTP/RTCP sobre UDP	idem
Seguridad	??	Seguro

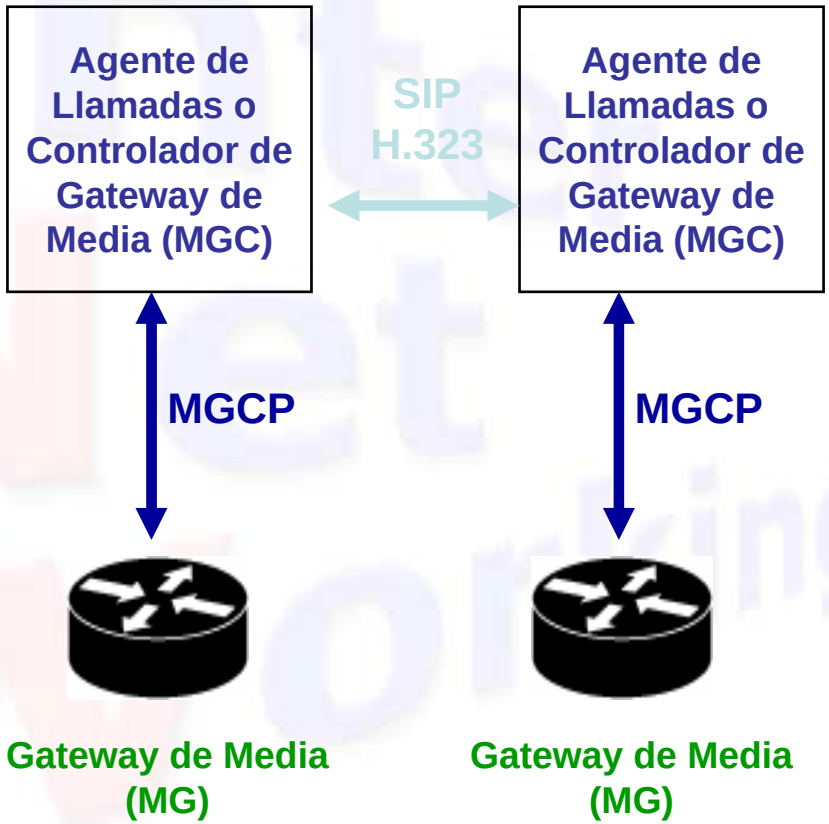
Voz / Multimedia sobre IP

Media Gateway Control Protocol (MGCP)
IETF RFC 2705

¿Qué es MGCP?

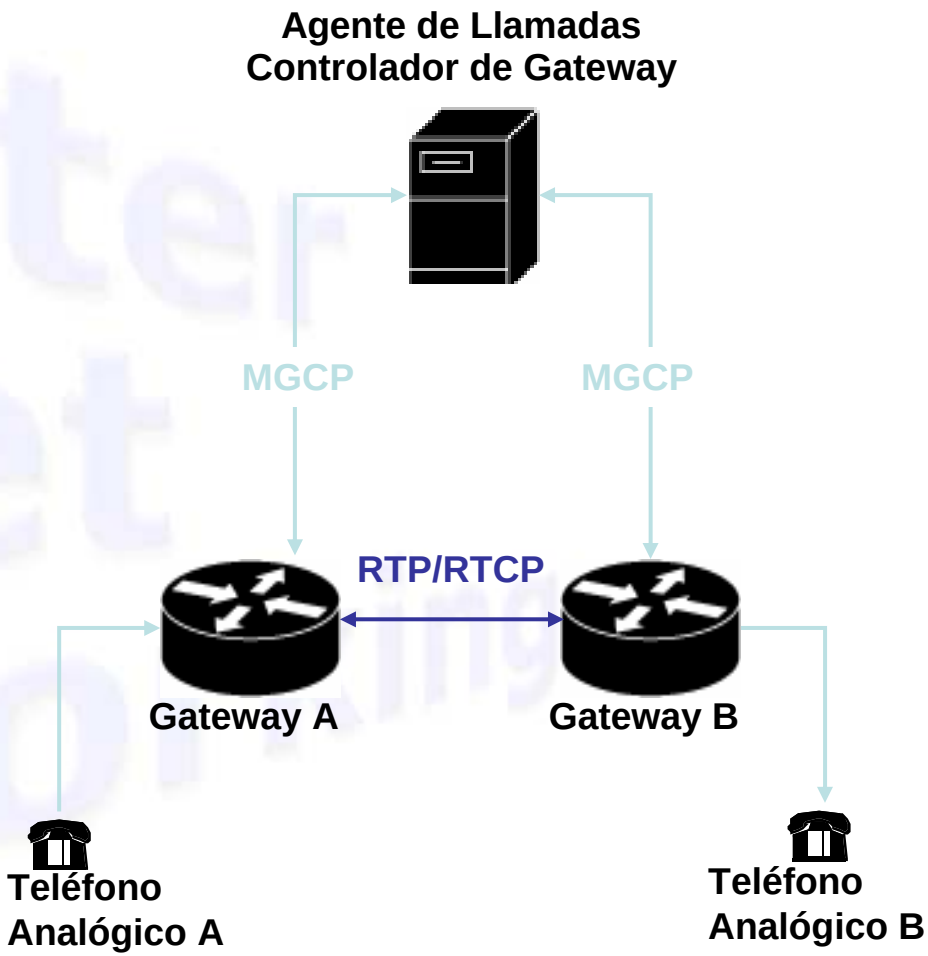
- Protocolo para controlar gateways telefónicos desde elementos de control externos llamados controladores de gateways de media o agentes de llamadas.
- Componentes de MGCP:
  - Gateways
    - para traducción entre redes de conmutación de circuitos y redes de datos.
    - Notifica al agente de llamadas sobre eventos en los terminales.
    - Ejecuta comandos a pedido del agente de llamadas.
  - Agente de llamadas (o Controlador de gateways de media)
    - Provee señalización, control e inteligencia al gateway.
    - Manda y recibe comandos del gateway.

Cómo se relaciona MGCP con SIP, H.323



Flujo de llamada Simplificado

- Cuando el teléfono A es descolgado, Gateway A manda una señal al agente de llamadas.
- Gateway A genera tono de marcar y captura el número discado.
- El número es mandado al agente de llamadas.
- El agente de llamadas determina cómo enrutar la llamada.
- El agente de llamadas manda comandos al Gateway B.
- Gateway B hace que el teléfono B suene.
- El agente de llamadas manda comandos a ambos gateways para establecer sesiones RTP/RTCP.



Comandos MGCP

- Agente de llamadas
  - EndpointConfiguration
  - NotificationRequest
  - CreateConnection
  - ModifyConnection
  - DeleteConnection
  - AuditEndpoint
  - AuditConnection
- Gateway
  - Notify
  - DeleteConnection
  - RestartInProgress

MGCP, SIP y H.323

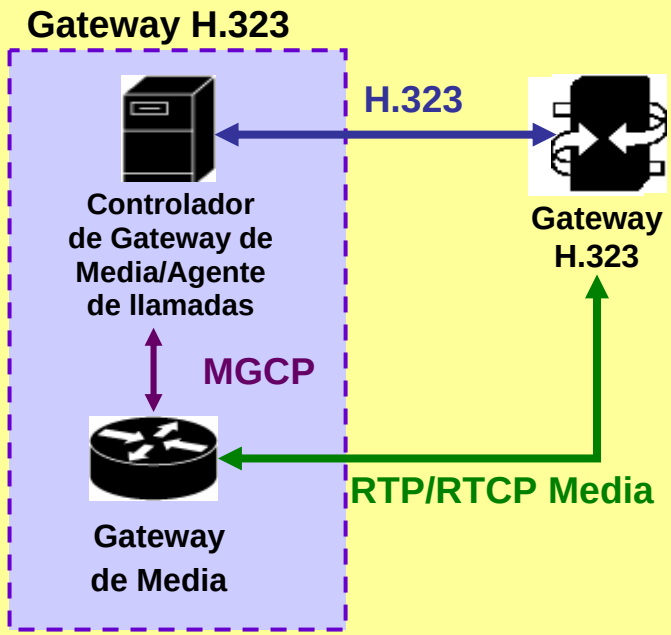
- MGCP separa lo que es control y establecimiento de llamadas de las funciones para la media.
- MGCP no reemplaza a SIP ni a H.323. SIP y H.323 proveen control y establecimiento de llamadas simétrico o peer-to-peer.
- MGCP interopera con SIP y H.323:
  - Un agente de llamadas acepta pedidos para establecer llamadas SIP o H.323
  - El agente de llamadas usa MGCP para controlar el gateway de media
  - El gateway de media establece sesiones de media con otros terminales H.323 o SIP.

En este ejemplo, un gateway H.323 se "descompone" en:

– Un agente de llamadas que ofrece señalización.

– Un gateway que maneja la media.

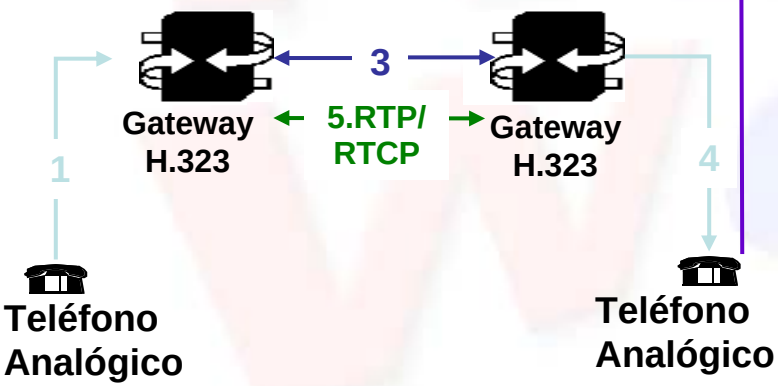
El protocolo MGCP se usa para controlar el gateway.



Ejemplo Comparativo

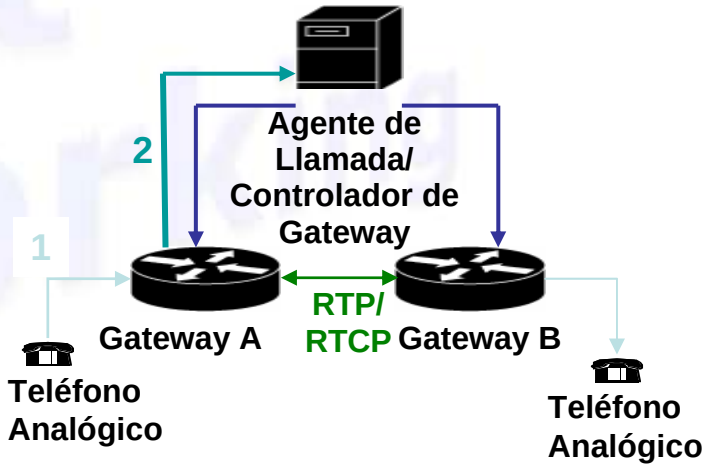
H.323

1. El usuario levanta el teléfono y marca un número.
2. El gateway determina cómo enrutar la llamada.
3. Los dos gateways intercambian información de capacidades.
4. El gateway de destino llama al número discado.
5. Los dos gateways establecen una sesión RTP/RTCP entre ellos.



MGCP

1. El usuario levanta el teléfono y marca un número.
2. El gateway notifica al agente de llamadas sobre el evento en el terminal.
3. El agente de llamadas determina capacidades, información de rutas, y le manda un comando a los gateways para que establezcan una sesión RTP/RTCP entre ellos



¿Qué es MEGACO?

- MEdia GAteway COntrol
- Un protocolo derivado de MGCP que está siendo desarrollado en forma conjunta entre la ITU y la IETF
  - ITU: H.248
  - IETF: RFC 2885
- Tiene algunos comandos adicionales a MGCP y puede usar codificación textual como MGCP y SIP (lo que le gusta a la IETF) o binaria con ASN.1 (como le gusta a la ITU).

Conclusiones

- SIP y H.323 son comparables ya que ambos ofrecen establecer, terminar y controlar llamadas, intercambiar capacidades y servicios telefónicos básicos. Con ellos se construyen redes de VoIP con arquitectura distribuida (preferida por el mundo Internet).
- SIP está ganando terreno y se espera que prevalezca sobre H.323.
- MGCP y MEGACO son protocolos para controlar gateways de media desde agentes de llamadas. Se pueden usar con H.323 o SIP para redes de VoIP con arquitectura centralizada (preferida por el mundo telefónico).

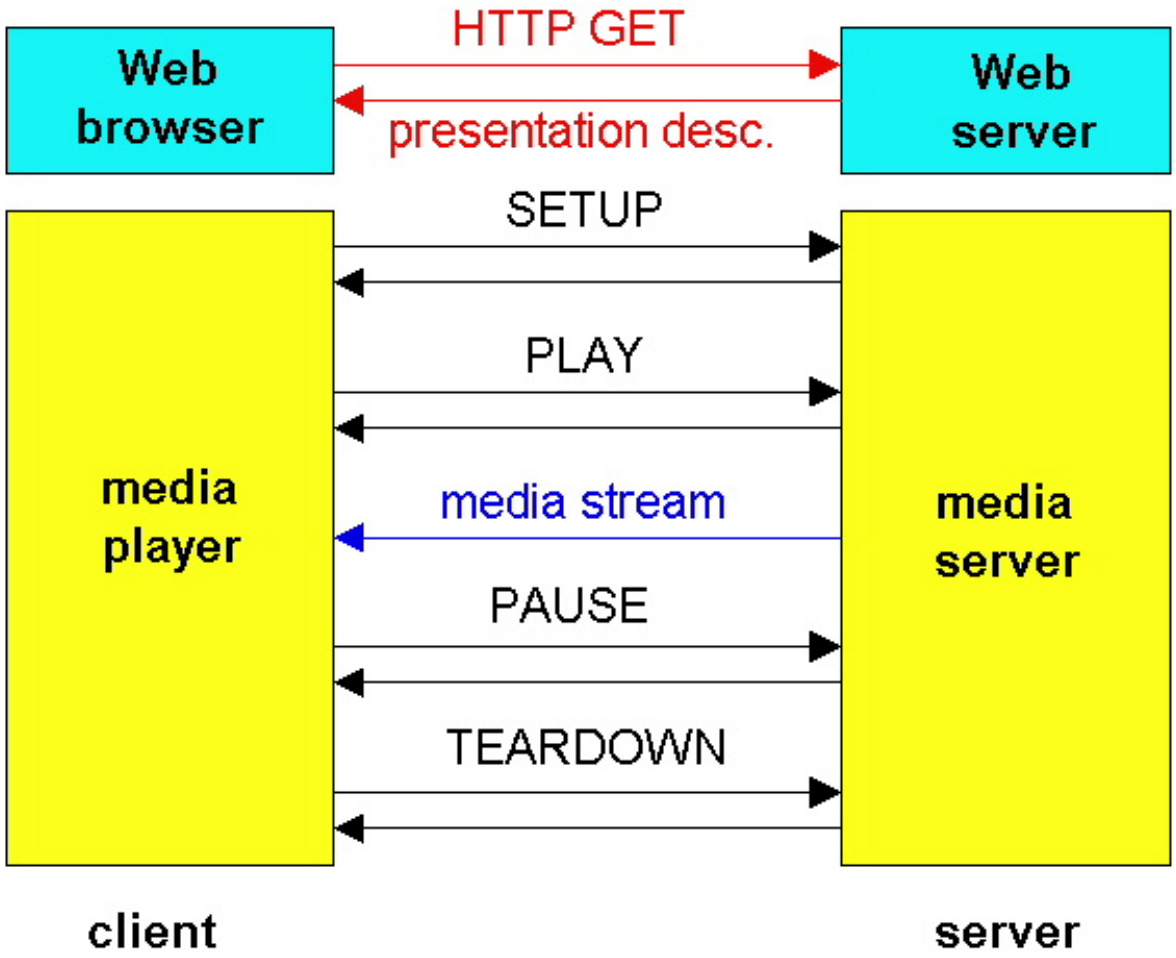
EN CONSTRUCCIÓN

RTSP

Metafile Example

```
<title>Twister</title>
<session>
        <group language=en lipsync>
                <switch>
                    <track type=audio
                            e="PCMU/8000/1"
                            src =
  "rtsp://audio.example.com/twister/audio.en/lofi">
                    <track type=audio
                            e="DVI4/16000/2" pt="90 DVI4/8000/1"
  src="rtsp://audio.example.com/twister/audio.en/hifi">
                </switch>
              <track type="video/jpeg"
  src="rtsp://v
```

RTSP Operation



RTSP Exchange Example

C: SETUP rtsp://audio.example.com/twister/audio RTSP/1.0
Transport: rtp/udp; compression; port=3056; mode=PLAY

S: RTSP/1.0 200 1 OK
Session 4231

C: PLAY rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Session: 4231
Range: npt=0-

C: PAUSE rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Session: 4231
Range: npt=37

C: TEARDOWN
RTSP/1.0

RFCs

- RFC 2326: Real-Time Streaming Protocol (RTSP)
- RFC 2327: Session Description Protocol (SDP)
- RFC 2543: Session Initiation Protocol (SIP)
- RFC 2974: Session Announcement Protocol (SAP)
- RFC 3108: SDP for ATM Bearer Connections
- RFC 3259: A Message Bus for Local Coordination
- RFC 3264: An Offer/Answer Model with Session Description Protocol (SDP)
- RFC 3266: Support for IPv6 in Session Description Protocol (SDP)
- RFC 3388: Grouping of Media Lines in the Session Description Protocol (SDP)
- RFC 3524: Mapping of Media Streams to Resource Reservation Flows