

# Introducción a HTTP y Sockets

Laboratorio de Redes y Sistemas Distribuidos

Eduardo A. Sanchez – 2011

Daniel Moisset – 2012

Marco A. Rocchietti – 2015

FaMAF - Marzo, 2015

# Protocolos



- ¿Qué es un protocolo?

Un protocolo es un formato normalizado o lenguaje que varias computadoras usan para intercambiar datos exitosamente.

- ¿Porqué es necesario?

Sin un protocolo, no hay comunicación. Por ejemplo, si dos personas necesitan hablar, pero hablan diferente idiomas, ellos serían incapaces de comunicarse. Las dos personas necesitarían hablar el mismo idioma para comunicarse.

- Protocolos de aplicación:

Hoy, en Internet hay miles de protocolos que son utilizados; sin embargo, la mayoría usan pocos solamente. Los protocolos más comunes son: HTTP, SMTP, POP3, FTP, DNS, IMAP, ...

# Protocolo HTTP

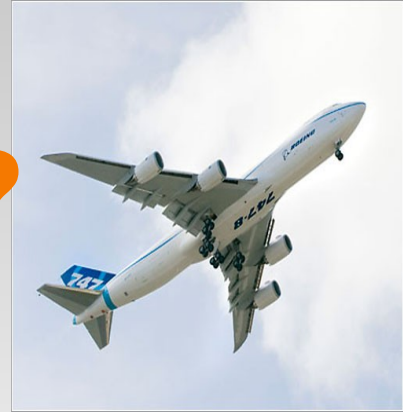


- **HTTP** = *Hyper Text Transfer Protocol*
- Protocolo de **transferencia** usado a través de la WWW.
- Especifica **como** deben ser los mensajes (*request/response*) entre clientes y servidores.
- Utiliza a **TCP** como medio de transporte.
- Protocolo **sin estado**, es decir, que no guarda ninguna información sobre conexiones anteriores.
- Más de una versión en vigencia: HTTP 1.0 y 1.1

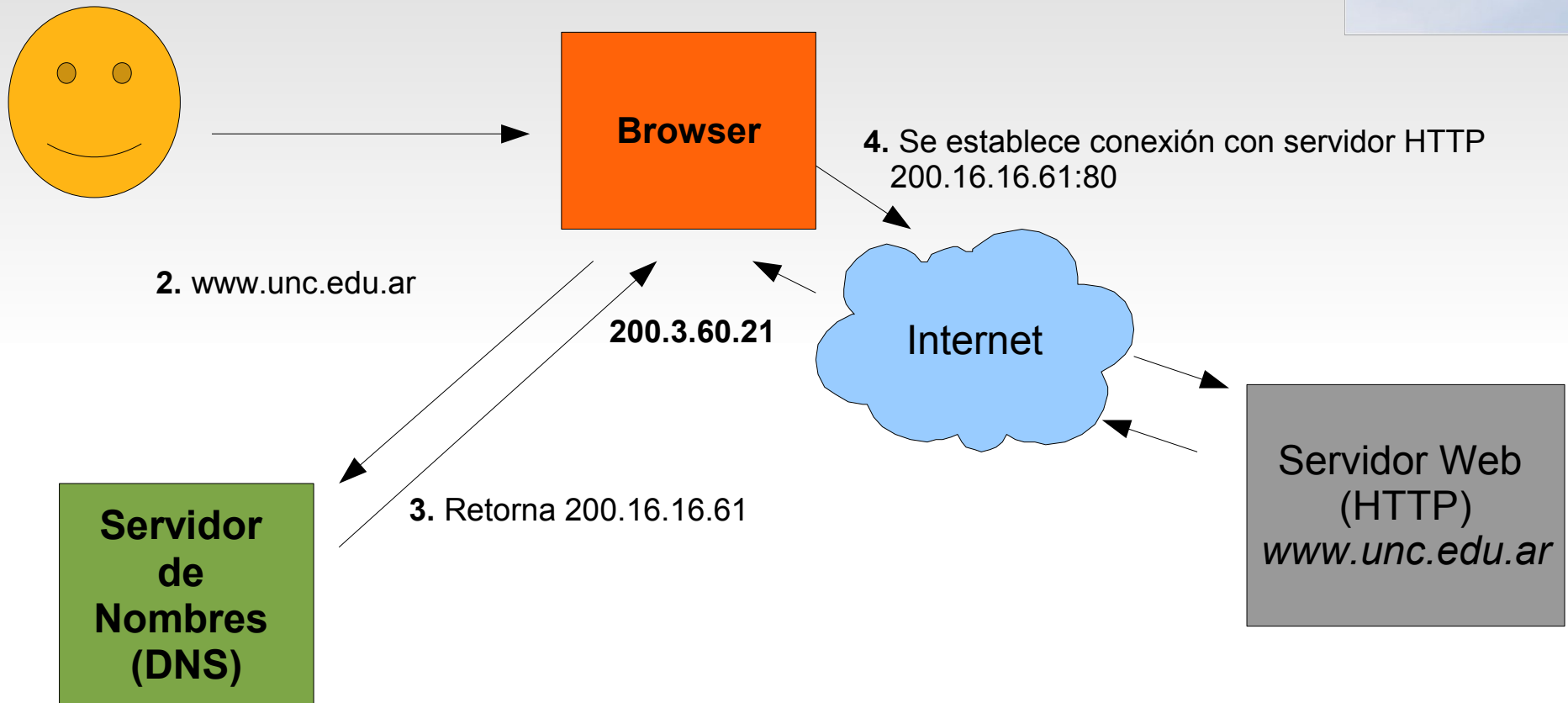
# Algunas definiciones:

- **Browser** = *Navegador*
  - Firefox, IE, Opera, Chrome, etc.
- **Port (TCP)** = *Es un número de 16 bits (e.g. 80)*
  - *Identifica a un programa (servicio) dentro de la computadora.*
- **IP address** = *Dirección de internet.(e.g. 190.7.60.22)*
  - *"Identifica" a una computadora dentro de internet*
- **URL** = *Universal Resource Locator (RFC2396)*
  - *protocol://hostname[:port]/path-and-file-name*
  - *ejemplo: http://www.google.com/search*
- **DNS** = *Domain Name Server*
  - *Ejemplos: 200.45.191.35 ó 8.8.8.8*

# ¿Cómo funciona la web?



1. <http://www.unc.edu.ar/seccion/novedades>

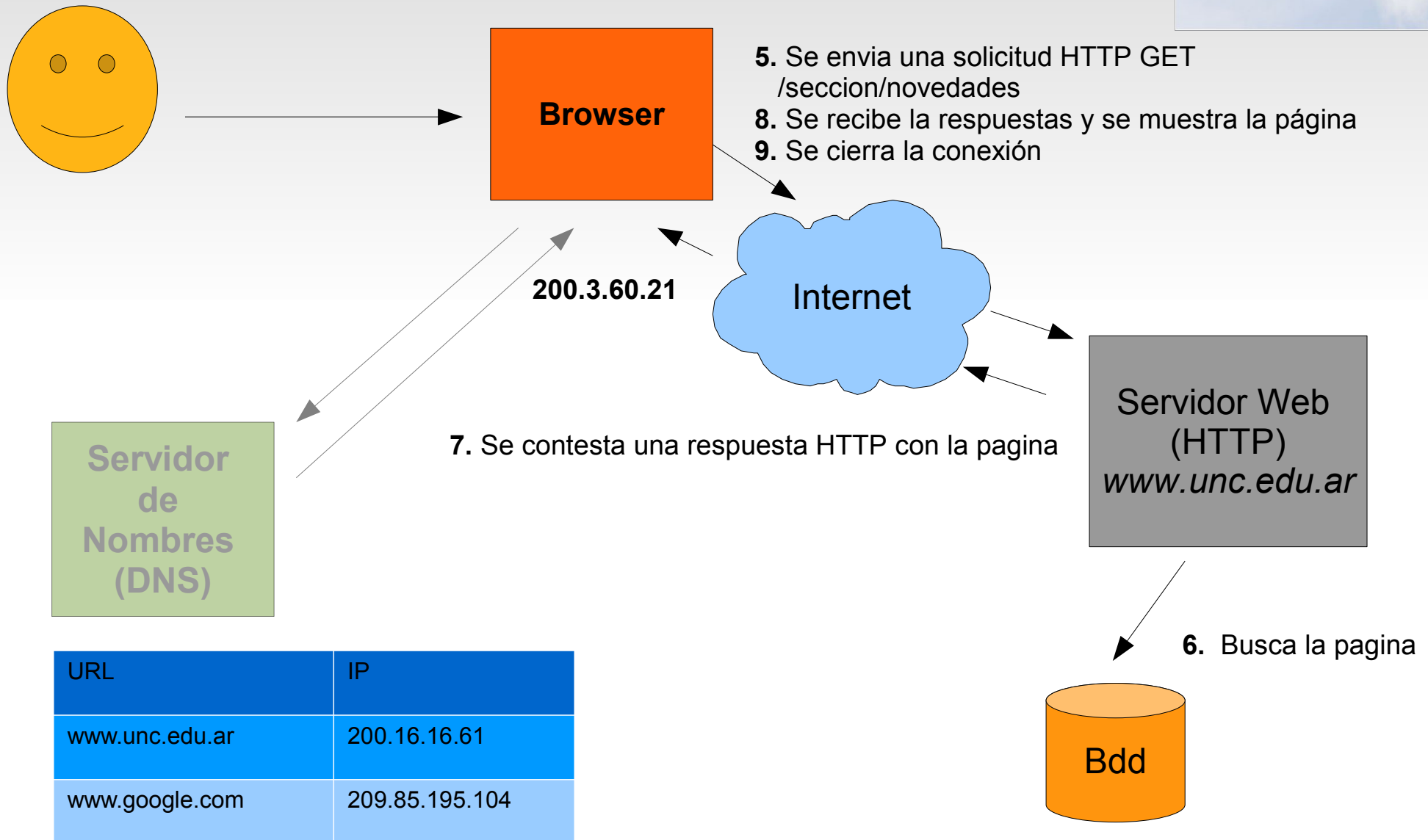


URL	IP
www.unc.edu.ar	200.16.16.61
www.google.com	209.85.195.104

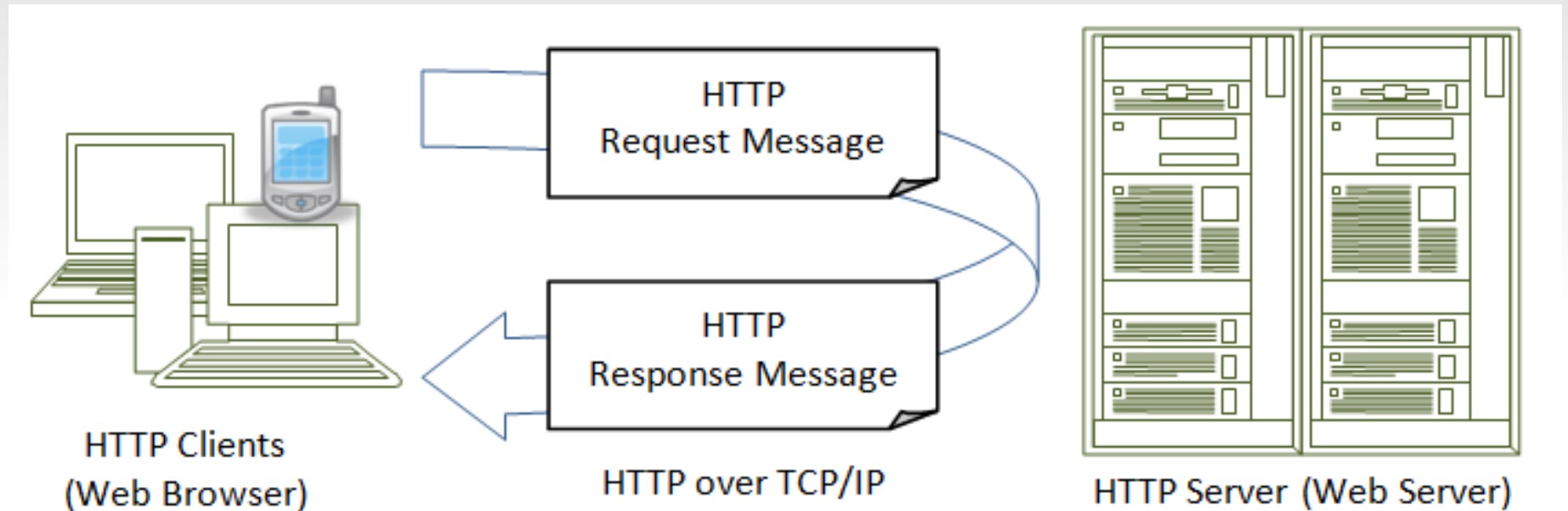
# ¿Cómo funciona la web?



1. <http://www.unc.edu.ar/seccion/novedades>

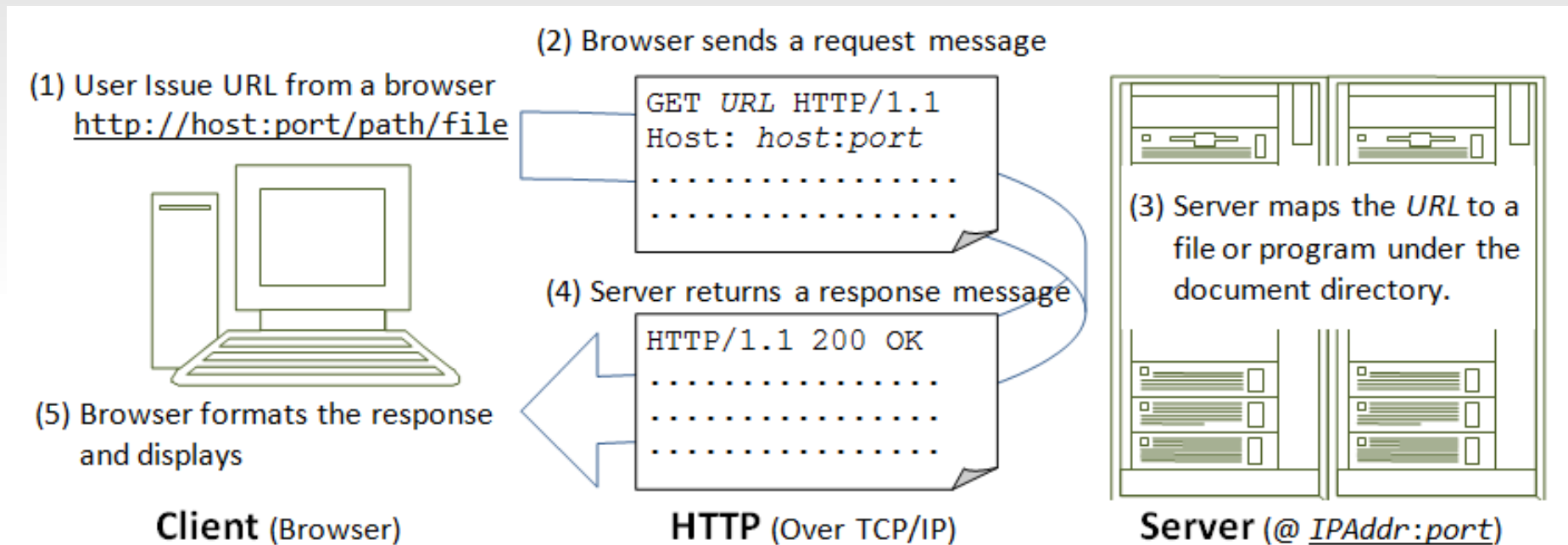


# Protocolo HTTP (cont.)



- × El cliente tira (*pull*) de la información.
- × El servidor nunca toma la iniciativa. Y gmail??

# Protocolo HTTP (cont.)



- (4) La respuesta: el recurso pedido o un mensaje de error.



# Ejemplo Práctico



- telnet www.lmanana.com.ar 80
  - GET http://www.lmcordoba.com.ar/impresa.php HTTP/1.0
  - Seguido de dos(2) fines de línea.

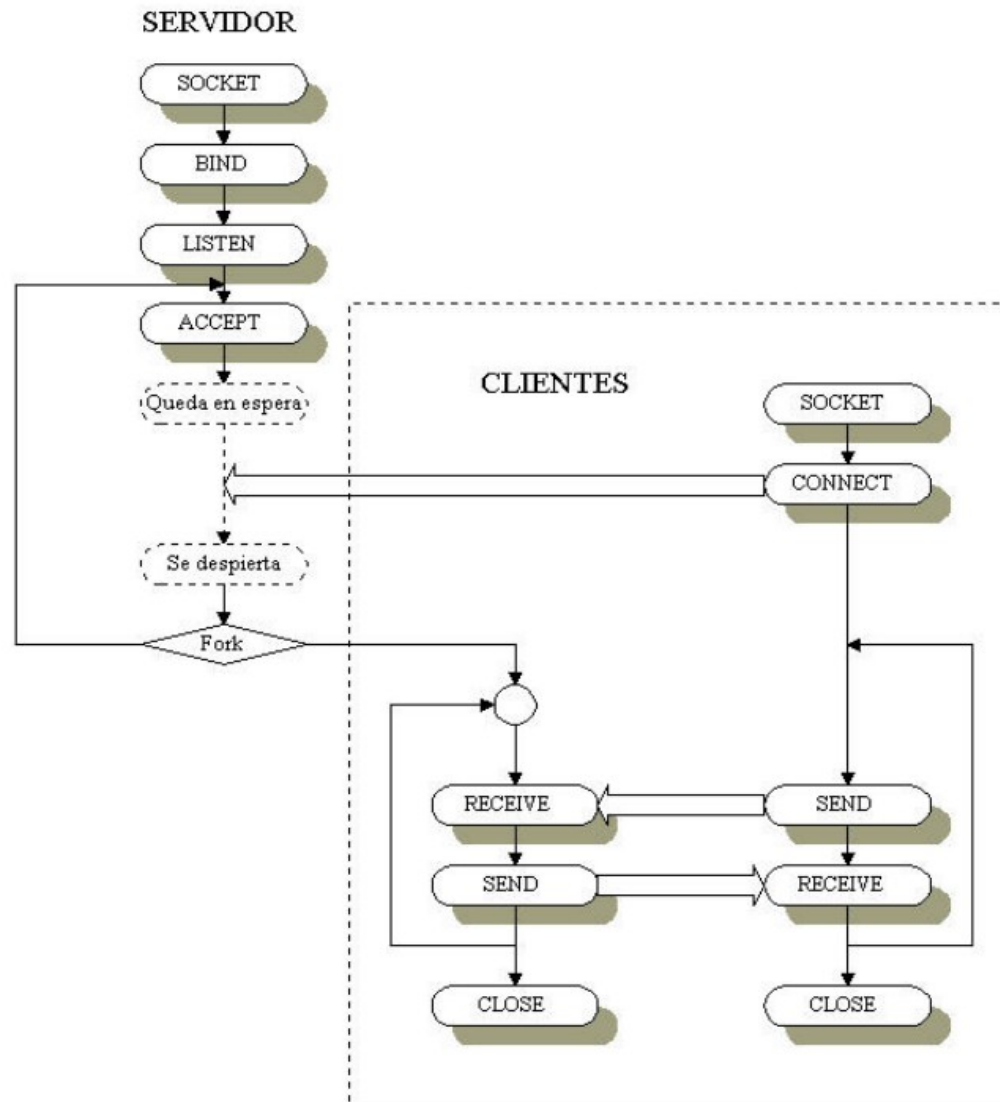
# SOCKETS



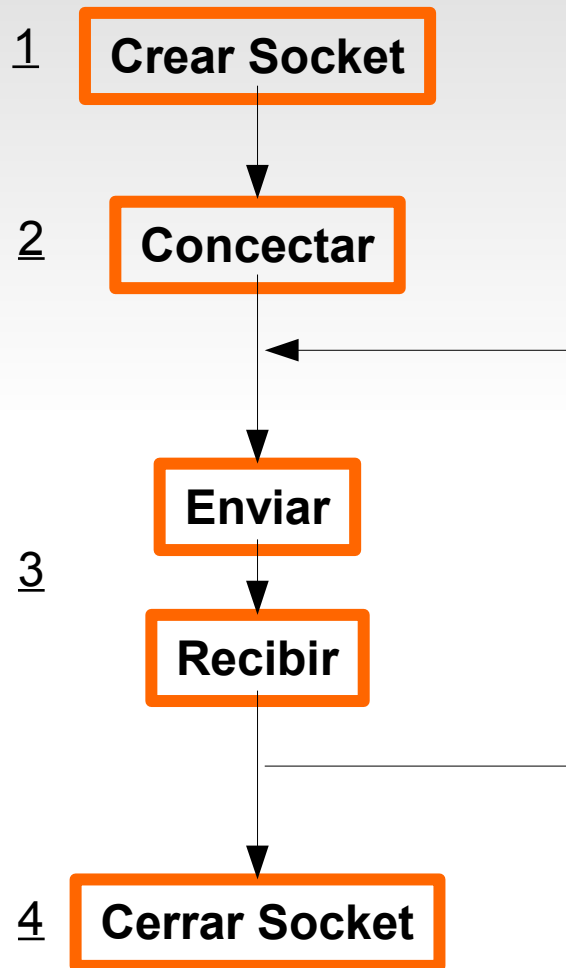
Para establecer una conexión entre dos computadoras usaremos la interfaz que proveen los sockets.

- ¿Qué es un Socket?
  - Un socket es como un pipe pero para comunicar procesos en dos computadoras distintas.
  - Cada computadora tiene una punta del socket.
  - Para armar un socket hace falta un IP y un puerto TCP.

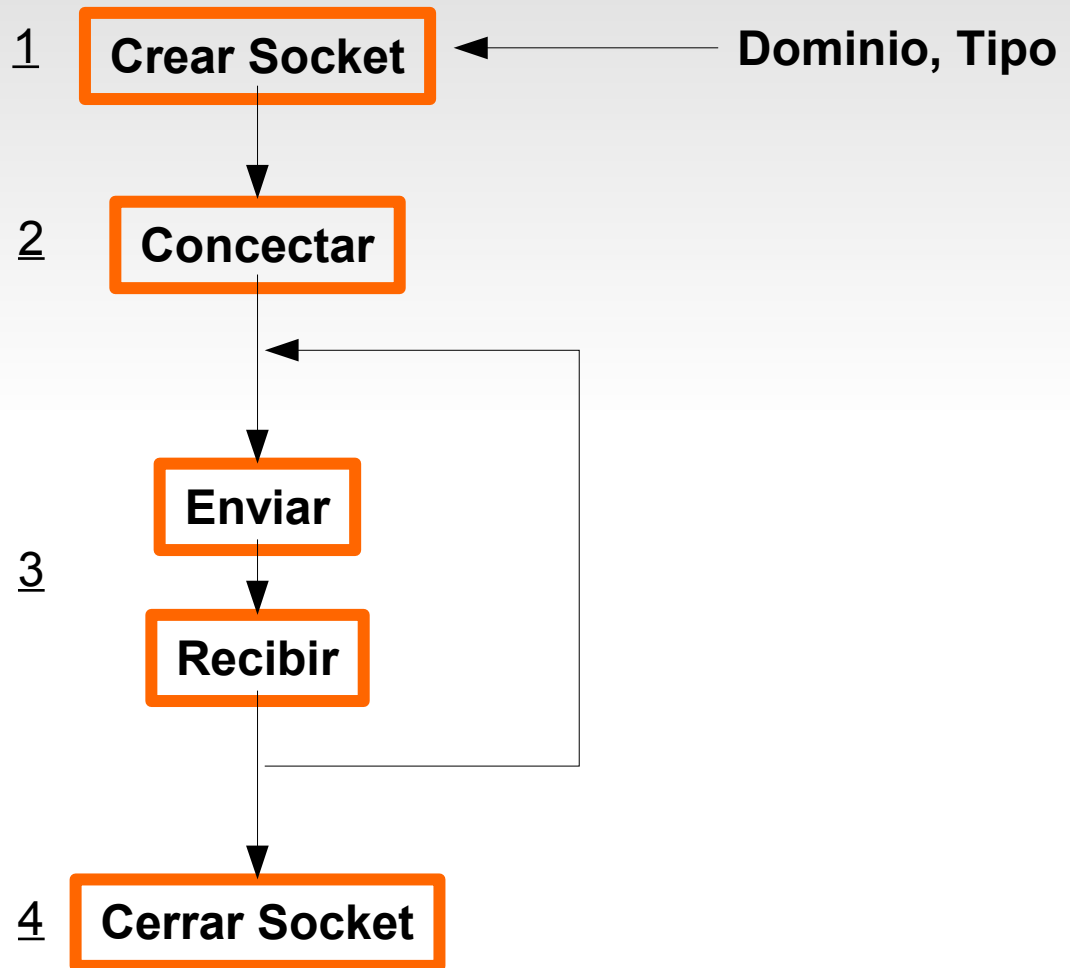
# SOCKETS



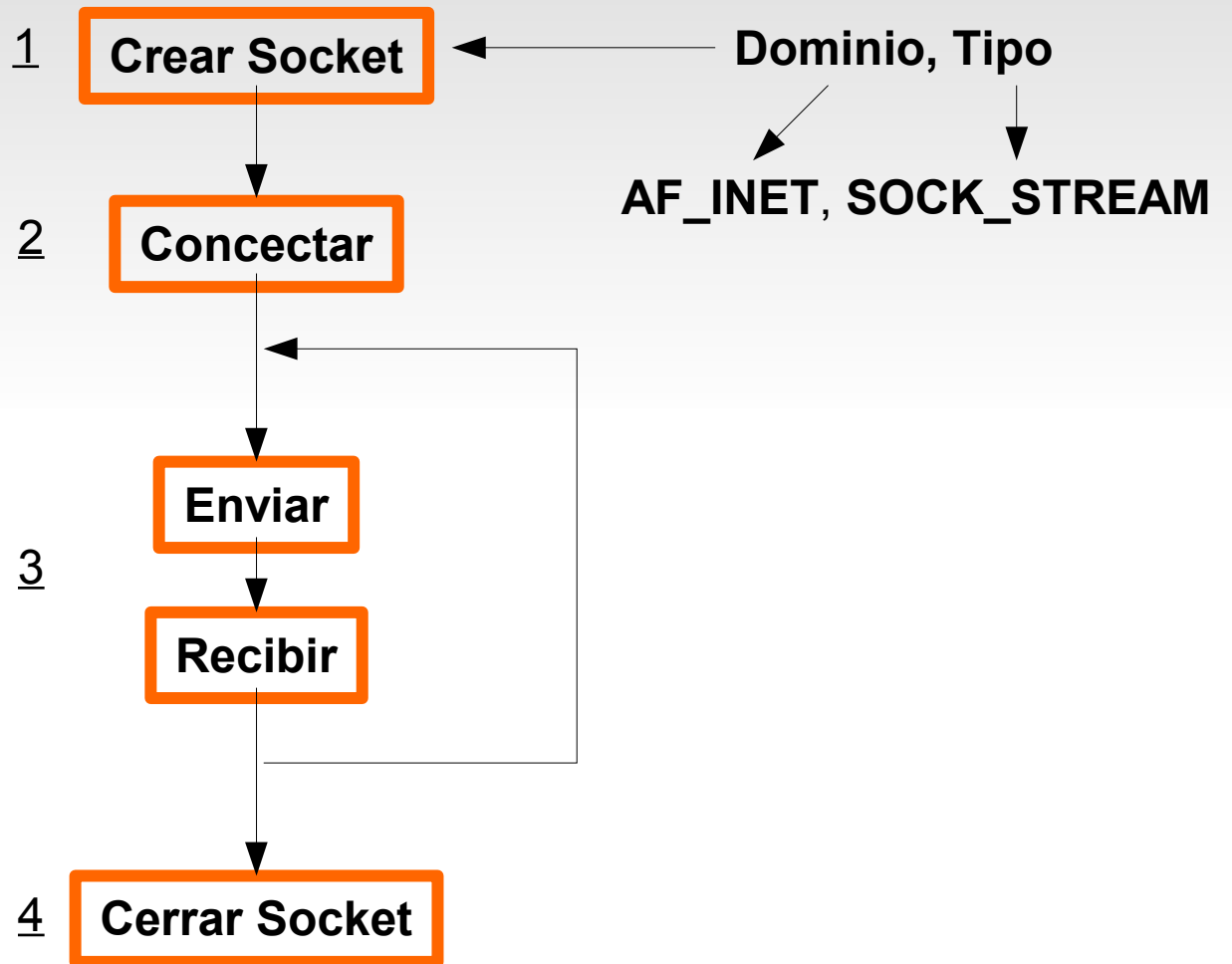
# SOCKETS



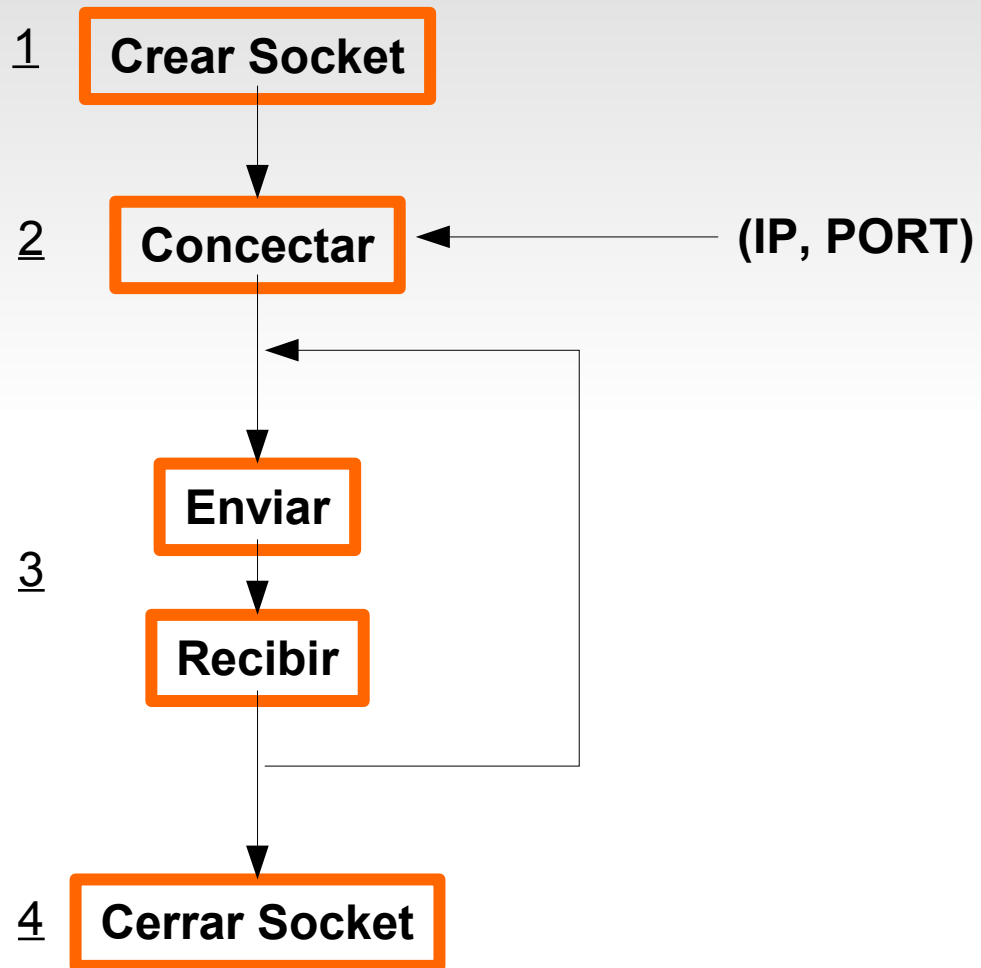
# SOCKETS



# SOCKETS



# SOCKETS



# SOCKETS



En Python:

- Para crear un nuevo socket:  
`import socket`

`s = socket.socket([familia[, tipo[, proto]]]):`

- Para conectar el socket creado a un destino:  
`destino = (ip, port)`  
`s.connect(destino)`

- Obtener el IP a partir de un nombre de host:  
`ip = socket.gethostbyname(nombre_host)`



# Recursos y Referencias

- [http://es.wikipedia.org/wiki/Hypertext\\_Transfer\\_Protocol](http://es.wikipedia.org/wiki/Hypertext_Transfer_Protocol)
- [https://developer.mozilla.org/en/HTTP\\_Logging](https://developer.mozilla.org/en/HTTP_Logging)
- Diagrama de secuencia completo HTTP
- HTTP.html
- HTTP4.html
- Computer Networks, 4<sup>th</sup> Edition, Andrew S. Tanenbaum
- <https://docs.python.org/2/library/socket.html>