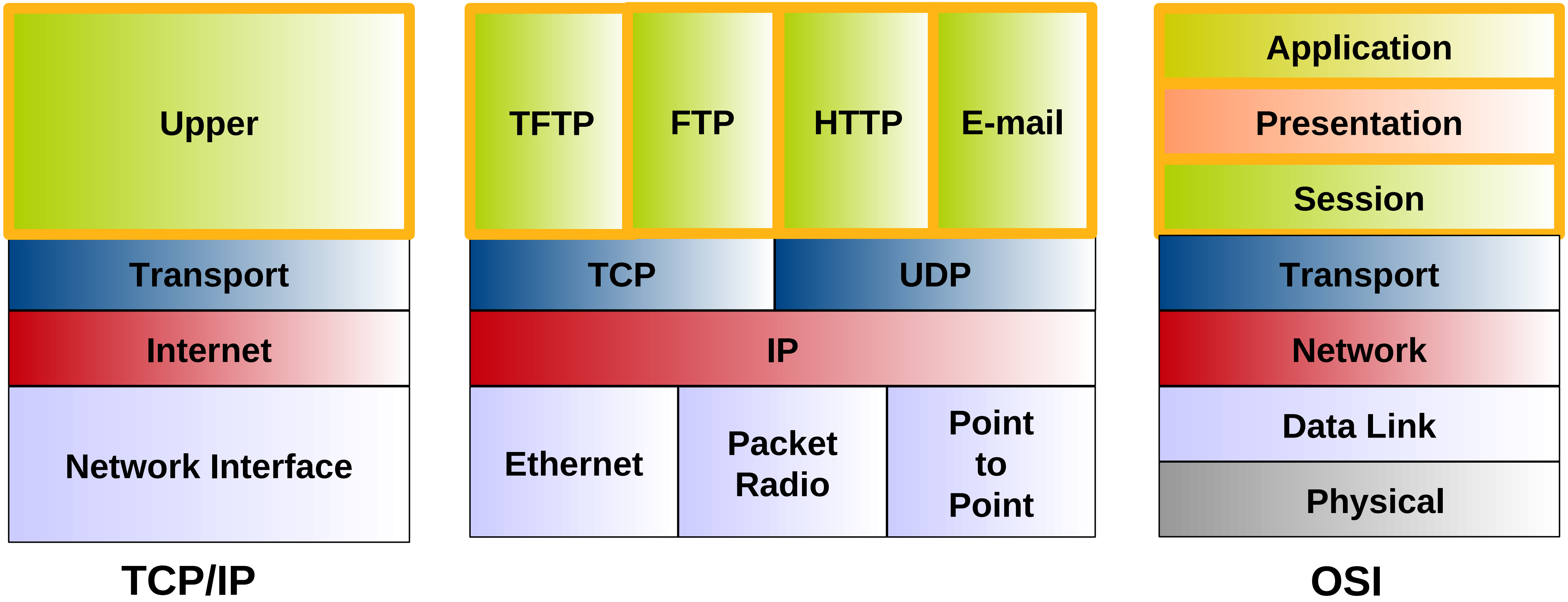
Applications Models

Redes de Comunicações II

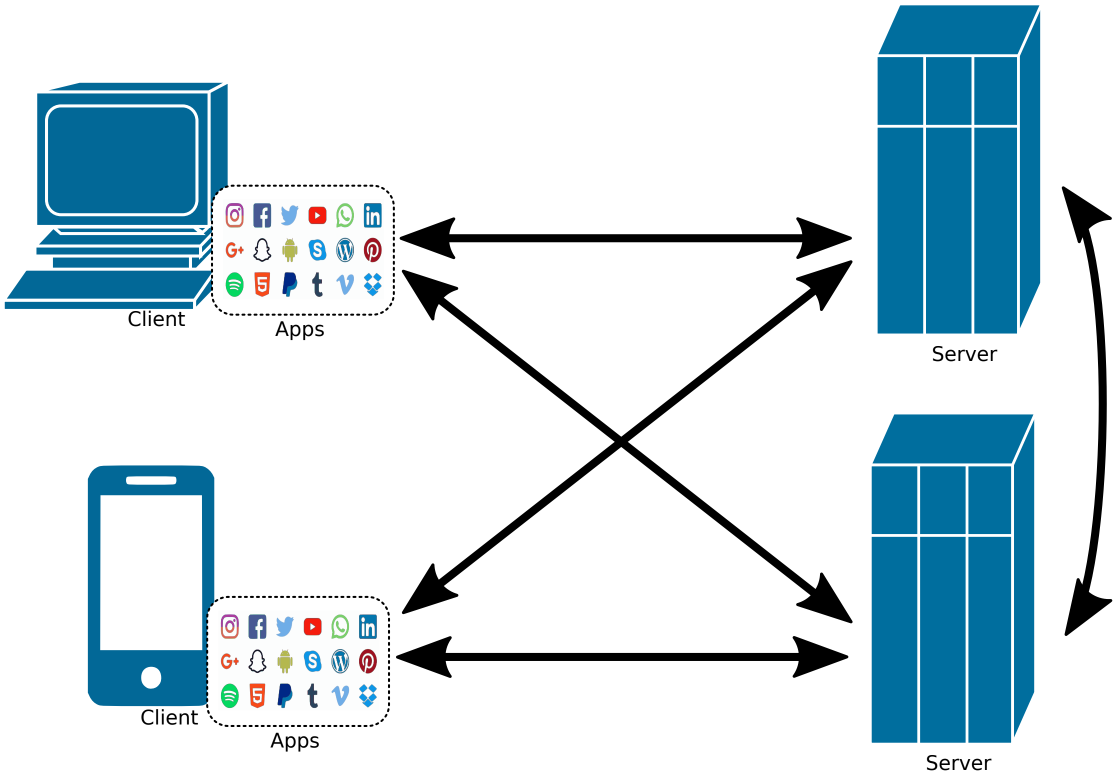
Licenciatura em
Engenharia de Computadores e Informática
DETI-UA

universidade de aveiro
deti.ua.pt

TCP/IP Reference Model



Client-Server Model



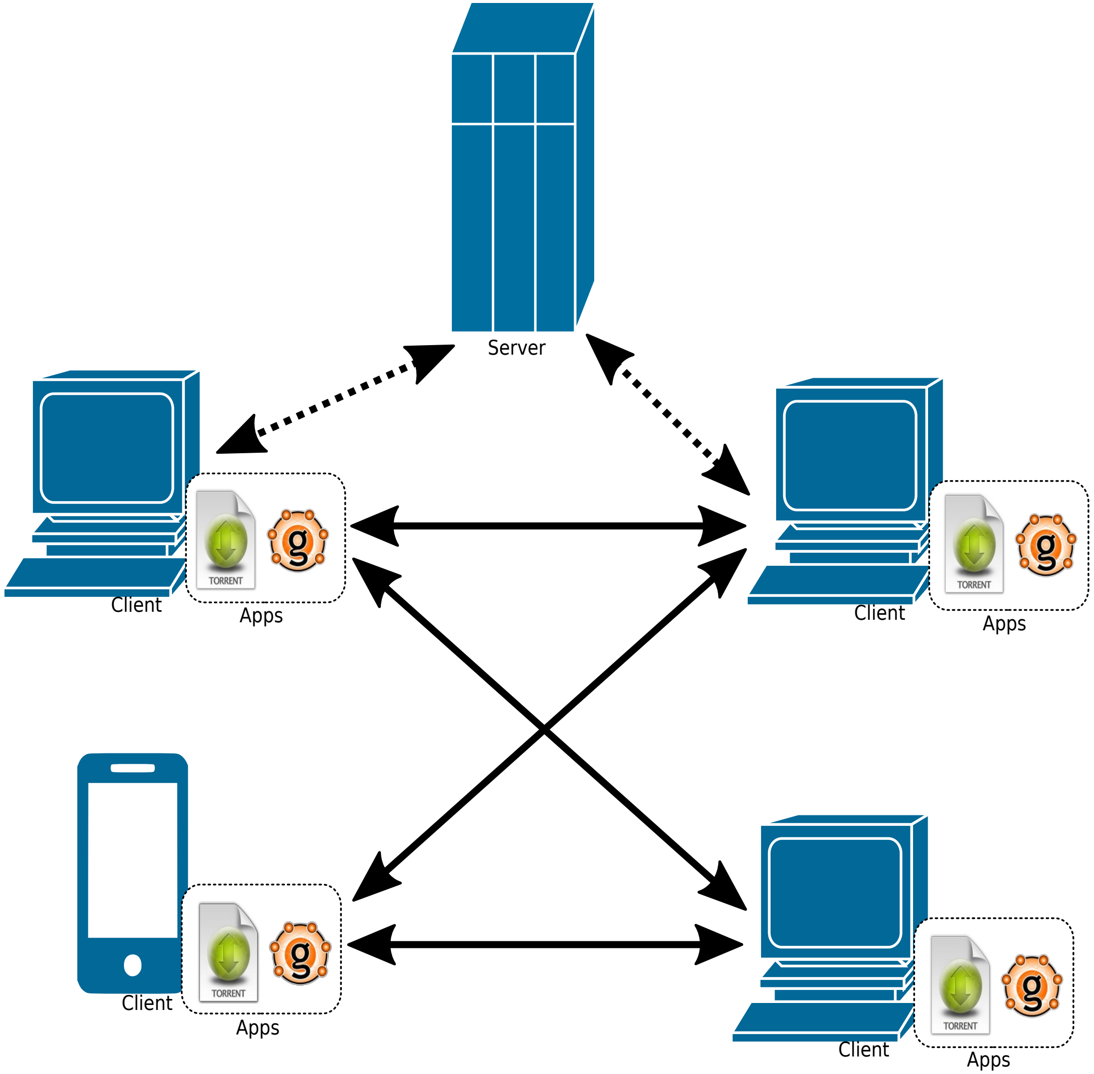
Servers:

- Always ON.
- IP address is always the same or exists a static association between a name and a dynamic IP address.
- May communicate between them.
  - May act as client.

Clients:

- Communicate with servers.
- Can be ON only when in operation.
- May have dynamic addresses.
- Within this model, they do not communicate between themselves.
  - P2P is another communication model.

P2P Model



Clients:

- Communicate between themselves.
- Can be ON only when in operation.
- May have dynamic addresses.
- Peer discovery may be done within the P2P network or using central servers.

Servers:

- May exist only to bootstrap P2P network.

VoIP
Voice (and Video) over IP

universidade de aveiro

deti.ua.pt

Voice over IP

- Network loss: IP datagram lost due to network congestion (router buffer overflow).
- Delay loss: IP datagram arrives too late for playout at receiver.
  - Delays: processing, queueing in network; end-system (sender, receiver) delays.
  - Typical maximum tolerable delay: 400 ms.
- Loss tolerance: depending on voice encoding, packet loss rates between 1% and 10% can be tolerated.
- Speaker's audio: alternating talk/speech with silent periods.
  - 64 kbps during talk/speech.
  - Packets generated only during talk/speech.
    - 20 msec chunks at 8 Kbytes/sec: 160 bytes data.
- Requires session establishment.
- VoIP protocols/frameworks:
  - Session Initiation Protocol (SIP)
    - Session Description Protocol (SDP)
  - H.323
- VoIP and PSTN interoperability in large/ISP scalable scenarios require complex control frameworks:
  - Media Gateway Controller Protocol (MGCP);
  - H.248/Megaco.

Session Initiation Protocol (SIP)

- Defined by RFC 3261.
- Designed for creating, modifying and terminating sessions between two or more participants.
  - Not limited to VoIP calls.
- Is a text-based protocol similar to HTTP.
  - Transported over UDP or TCP protocols.
    - Security at the transport and network layer provided with TLS (requires TCP) or IPSec.
- Offers an alternative to the complex H.323 protocols.
- Due to its simpler nature, the protocol is becoming more popular than the H.323 family of protocols.
- SIP is a peer-to-peer protocol. The peers in a session are called user agents (UAs):
  - User-agent client (UAC) - A client application that initiates the SIP request.
  - User-agent server (UAS) - A server application that contacts the user when a SIP request is received and that returns a response on behalf of the user.
- A SIP endpoint is capable of functioning as both UAC and UAS.

SIP Functionality

- SIP supports five facets of establishing and terminating multimedia communications:
  - User location - determination of the end system to be used for communication;
  - User availability - determination of the willingness of the called party to engage in communications;
  - User capabilities - determination of the media and media parameters to be used;
  - Session setup - "ringing", establishment of session parameters at both called and calling party;
  - Session management - including transfer and termination of sessions, modifying session parameters, and invoking services.

SIP Clients and Servers

- SIP Clients
  - Phones (software based or hardware).
  - Gateways
  - User Agents
  - A User Agent acts as a
    - Client when it initiates a request (UAC),
    - Server when it responds to a request (UAS).
- SIP Servers
  - Proxy server
    - Receives SIP requests from a client and forwards them on the client's behalf.
    - Receives SIP messages and forward them to the next SIP server in the network.
    - Provides functions such as authentication, authorization, network access control, routing, reliable request retransmission, and security.
  - Redirect server
    - Provides the client with information about the next hop or hops that a message should take and then the client contacts the next-hop server or UAS directly.
  - Registrar server
    - Processes requests from UACs for registration of their current location.
    - Registrar servers are often co-located with a redirect or proxy server.

SIP Messages

- SIP used for Peer-to-Peer Communication though it uses a Client-Server model.
- SIP is a text-based protocol and uses the UTF-8 charset.
- A SIP message is either a **request** from a client to a server, or a **response** from a server to a client.
  - A request message consists of a Request-Line, one or more header fields, an empty line indicating the end of the header fields, and an optional message-body;
  - A response message consists of a Status-Line, one or more header fields, an empty line indicating the end of the header fields, and an optional message-body.
  - All lines (including empty ones) must be terminated by a carriage-return line-feed sequence (CRLF).

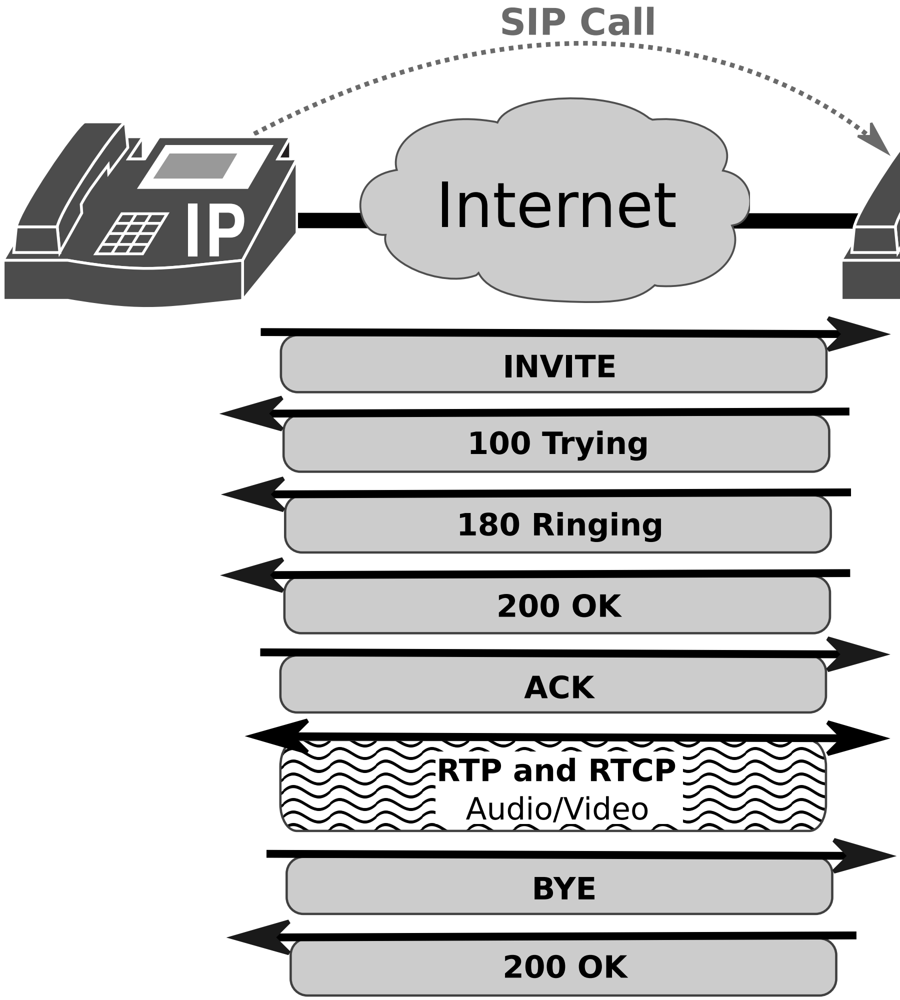
SIP Requests

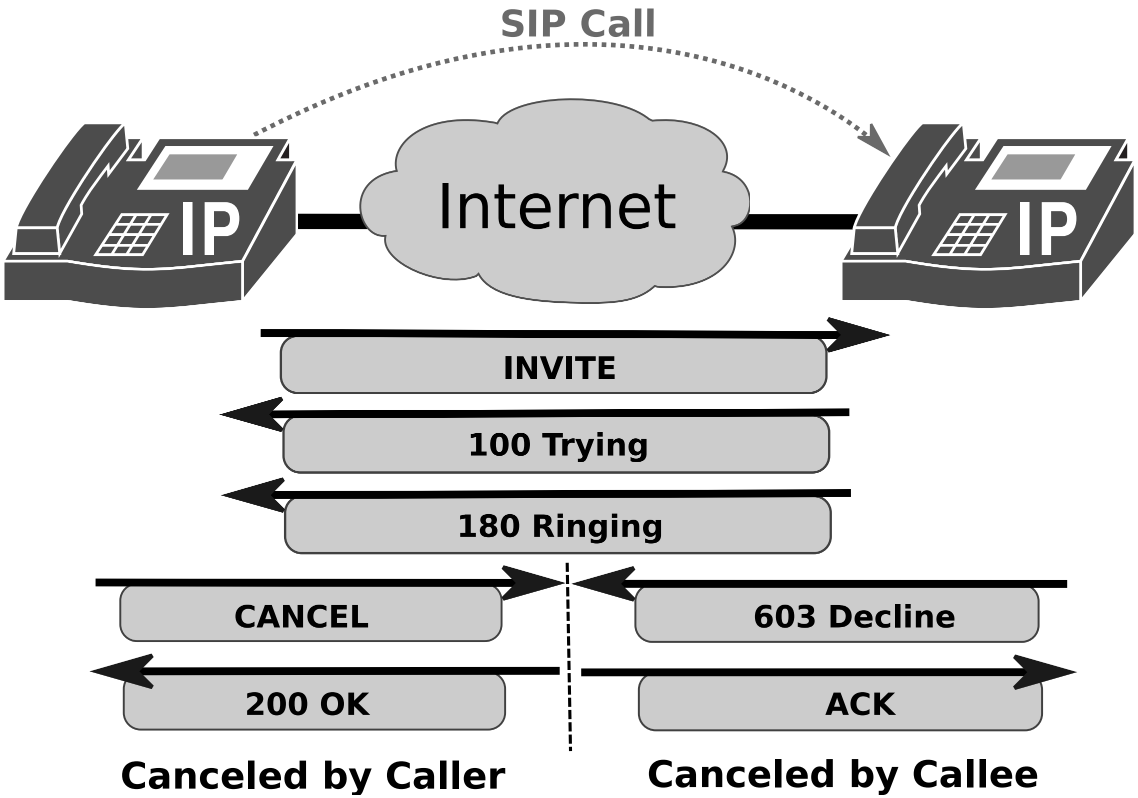
- Requests are also called “Methods”.
- SIP uses SIP Uniform Resource Indicators (URI) to indicate the user or service to which a request is being addressed.
- The general form of a SIP Request-URI is:
  - sip:user:password@host:port;uri-parameters
    - sip:John@doe.com
    - sip:+14085551212@company.com
    - sip:alice@atlanta.com;maddr=239.255.255.1;ttl=15
  - Proxies and other servers route requests based on Request-URI.
- Requests are distinguished by starting with a Request-Line.
  - A Request-Line contains a **Method** name, a **Request-URI**, and **SIP-Version** separated by a single space (SP) character.
    - Request-Line = Method SP Request-URI SP SIP-Version CRLF
  - RFC 3261 defines six methods: INVITE, ACK, OPTIONS, BYE, CANCEL, and REGISTER.
    - SIP extensions provide additional methods: SUBSCRIBE, NOTIFY, PUBLISH, MESSAGE, ...
  - SIP-Version should be “SIP/2.0”.
  - Example:
    - Request-Line: INVITE sip:2001@192.168.56.101 SIP/2.0
- The remaining of a request message is one or more header fields, an empty line indicating the end of the header fields, and an optional message-body.

Session Description Protocol (SDP)

- SIP carries (encapsulates) SDP messages.
- When initiating multimedia teleconferences, VoIP calls, streaming video, or other sessions, is required to transmit to participants media details, transport addresses, and other session description metadata.
- SDP (RFC 4566) provides a standard representation for such information, irrespective of how that information is transported.
  - SDP is purely a format for session description.
  - SDP is intended to be general purpose so that it can be used in a wide range of network environments and applications.
  - SDP does not support negotiation of session content or media encodings.

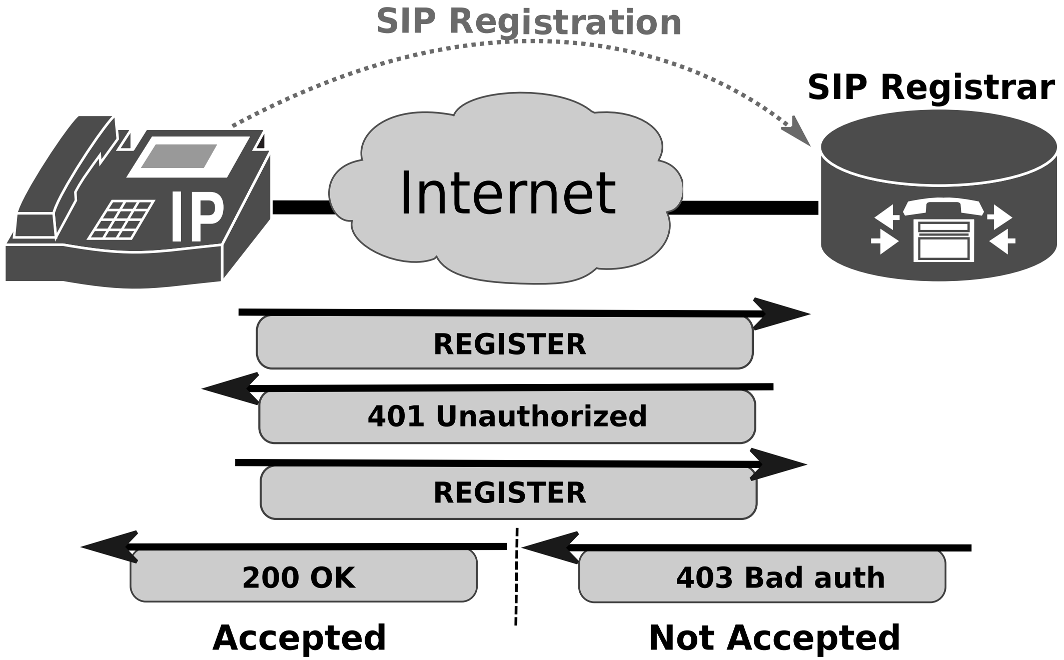
SIP Signaling – Direct Call





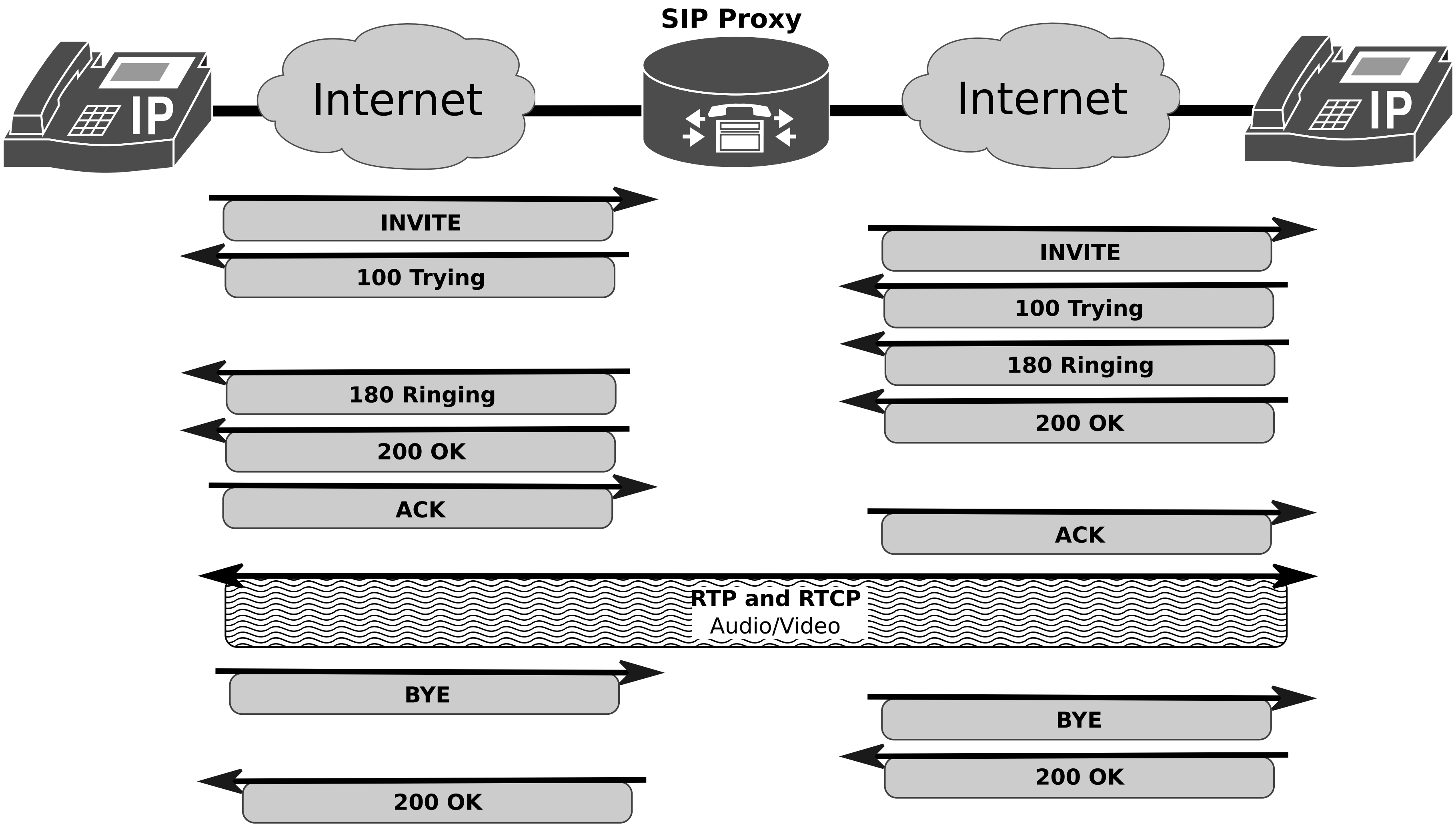
SIP Registrar Server

- SIP Registrar servers store the location of SIP endpoints.
- A user has an account created which allows them to REGISTER contacts with a particular server.
- The account specifies a SIP "Address of Record (AOR)"
- Each SIP endpoint Registers with a Registrar server with a SIP REGISTER request.
  - Using it's Address of Record and Contact address.
- Address of Record is in From header:
  - `From: <sip:Vieira@192.168.56.102>`
- Contact header tells Registrar server where to send messages:
  - `Contact: <sip:Vieira@192.168.56.1:5060>`
- SIP Proxy servers query SIP Registrar servers for routing information.
- The REGISTER message has the field "`Expires:`" to define for how much time the registration should be valid.
- A REGISTER message with an expiration time of zero, field "`Expires:0`", deregisters the user form the server.



- Registration usually requires authentication.
- If REGISTER has no authentication credentials, the SIP Registrar server responds with 401 Unauthorized.
- End-point resends REGISTER with an Authorization header with credentials.
  - `Authorization: Digest username="Vieira", realm="asterisk", nonce="7d88f81c", uri="sip:2001@192.168.56.102", algorithm=MD5, response="b70474b5bbece20a68472e7ad4e37197"`
- Server accepts registration with a 200 OK response.
- Server rejects credentials with a 403 Bad Auth response.

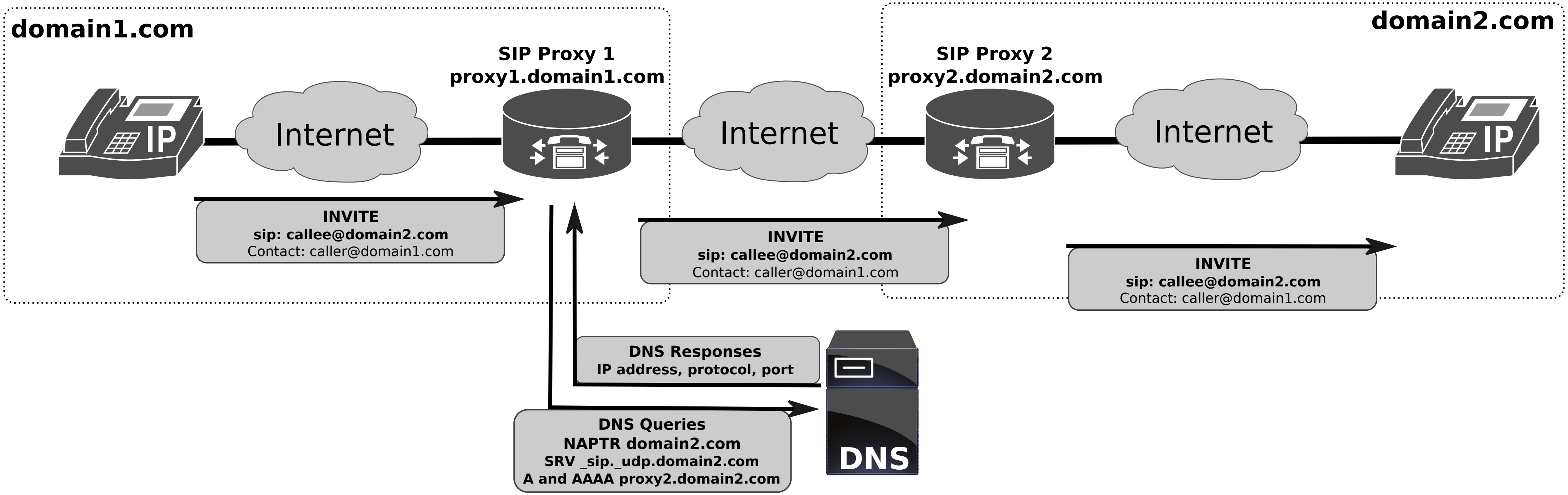
SIP Proxy Server



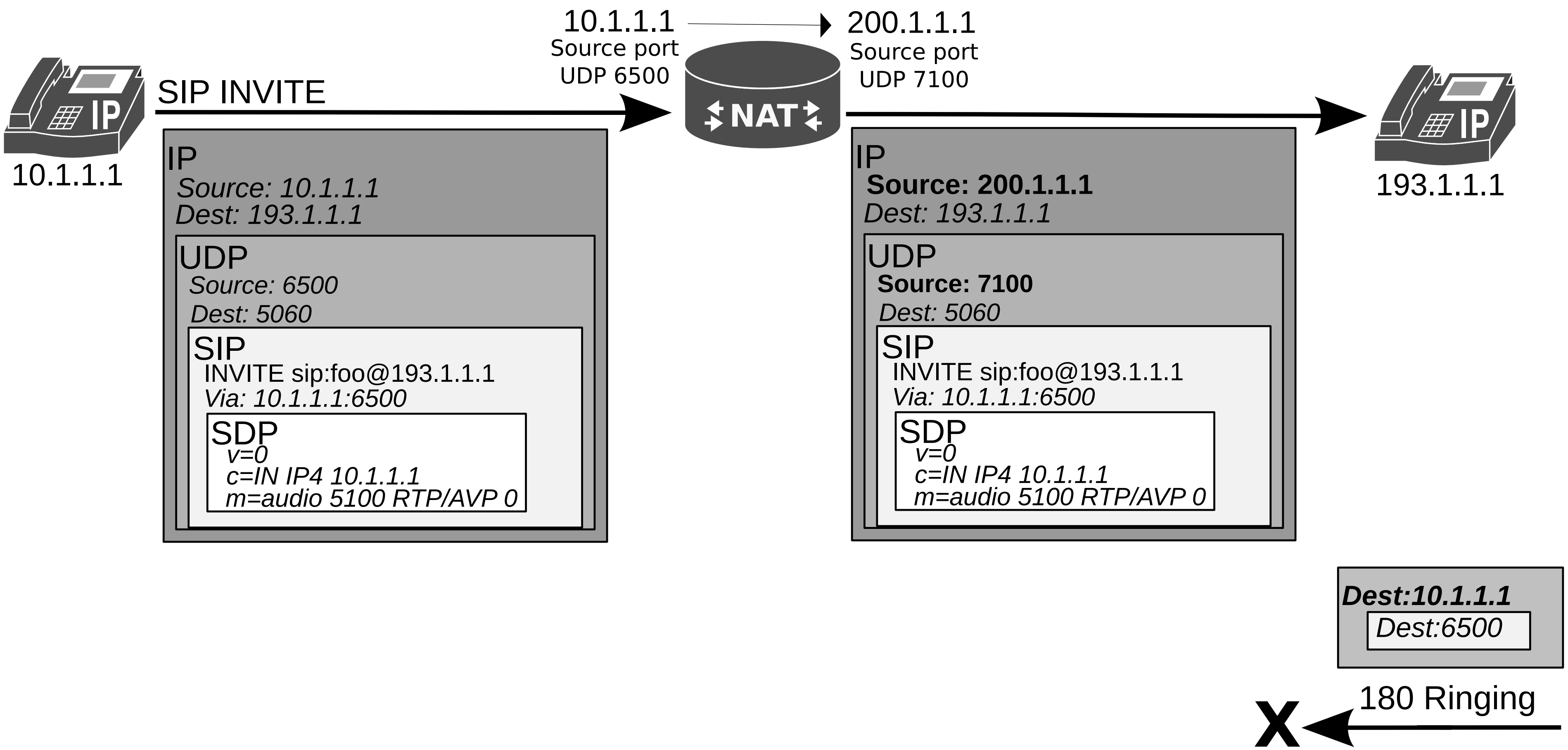
Locating SIP Servers

- RFC 3263 defines a set of DNS procedures to locate SIP Servers.
- SIP elements need to send requests/responses to a resource identified by a SIP URI.
  - The SIP URI may identify the desired target resource or a intermediate hop towards that resource.
  - Requires **Transport protocol**, **IP address** and **Port.**
    - If the URI specifies any of them, then it should be used.
  - Otherwise, must be retrieved from a DNS server.
    - Using **Service (SRV)** and **Name Authority Pointer (NAPTR)** DNS records.
- NAPTR records provide a mapping from a domain name to:
  - A SRV record (that contains the resource responsible server name),
  - And, the specific transport protocol.
- Example:
  - A client/server that wishes to resolve “sip:user@example.com”,
  - Performs a NAPTR query for domain “example.com”,
    - `IN NAPTR 100  50  "s"  "SIP+D2U"  ""  _sip._udp.example.com.`
  - Has UDP as possible transport protocol, performs a SRV query for “_sip._udp.example.com”
    - `IN SRV  0  1  5060  server1.example.com`
    - `IN SRV  0  2  5060  server2.example.com`
  - Has two possible servers, performs A and AAAA queries for the chosen server.

SIP Proxy Forwarding

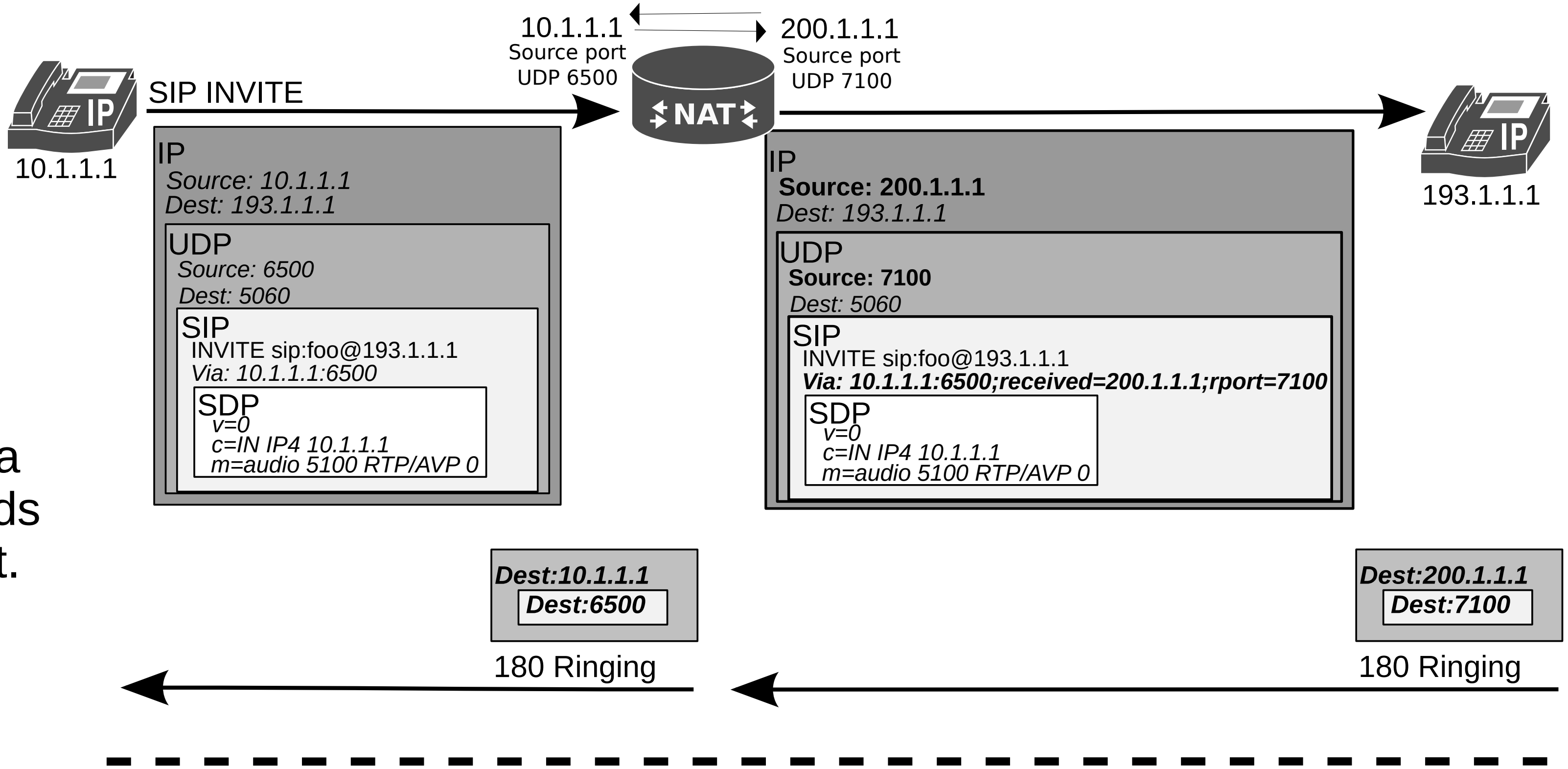


SIP and NAPT

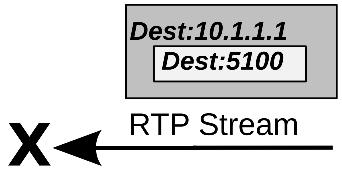


10.1.1.1
Source port
UDP 6500
200.1.1.1
Source port
UDP 7100
SIP INVITE
NAT
10.1.1.1
193.1.1.1
IP
Source: 10.1.1.1
Dest: 193.1.1.1
UDP
Source: 6500
Dest: 5060
SIP
INVITE sip:foo@193.1.1.1
Via: 10.1.1.1:6500
SDP
v=0
c=IN IP4 10.1.1.1
m=audio 5100 RTP/AVP 0
IP
Source: 200.1.1.1
Dest: 193.1.1.1
UDP
Source: 7100
Dest: 5060
SIP
INVITE sip:foo@193.1.1.1
Via: 10.1.1.1:6500
SDP
v=0
c=IN IP4 10.1.1.1
m=audio 5100 RTP/AVP 0
Dest:10.1.1.1
Dest:6500
180 Ringing
X

SIP NAPT Traversal

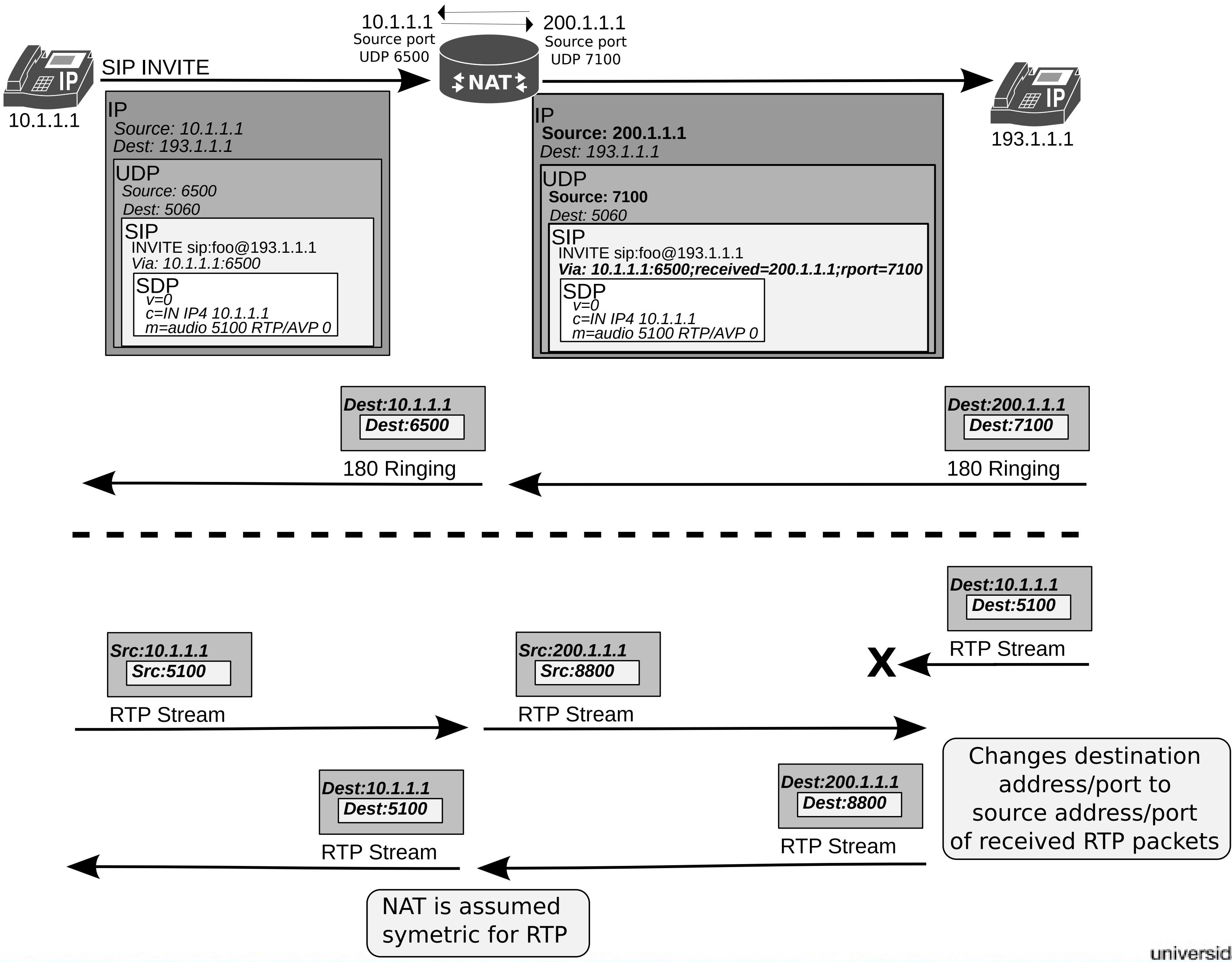


- Symmetric Response Routing (RFC 3581).
- SIP payload is also “translated”, by adding a **received** and **rport** fields with public address/port.
- SDP remains unchanged.



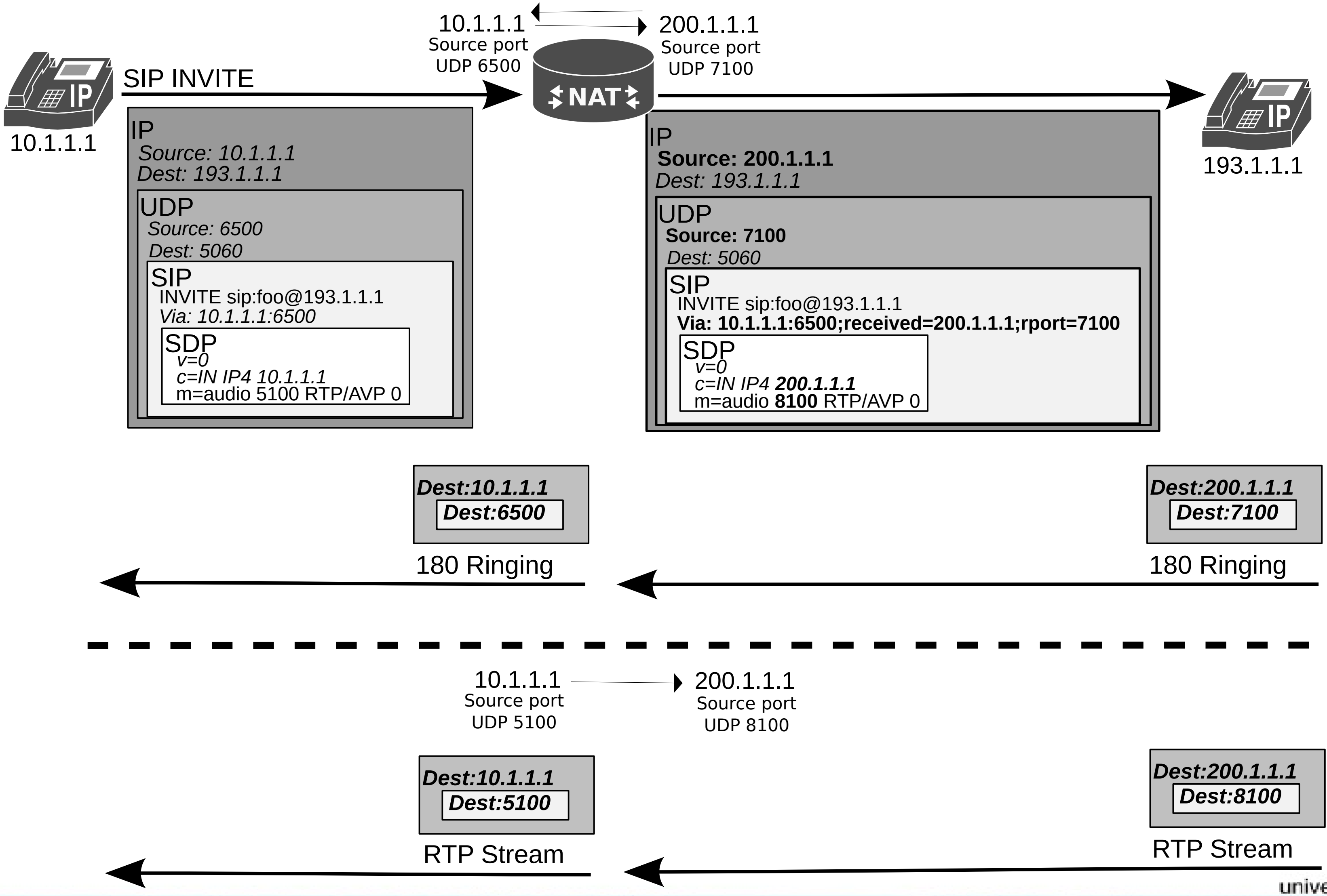
- Media traversal (RTP/RTCP) is still a problem.
  - SDP contents mismatch with public address/port.
  - Possible solutions
    - Let clients (on private network) find out their public address/port and rewrite SDP payload.
      - Manual configuration (when NAT uses static translations).
      - Automatic discovery (when NAT is dynamic) using STUN protocol.
    - Symmetric (RTP/RTCP) NAT (RFC 4961).
    - NAT SIP Application Layer Gateway (ALG).

Symmetric (RTP/RTCP) NAT



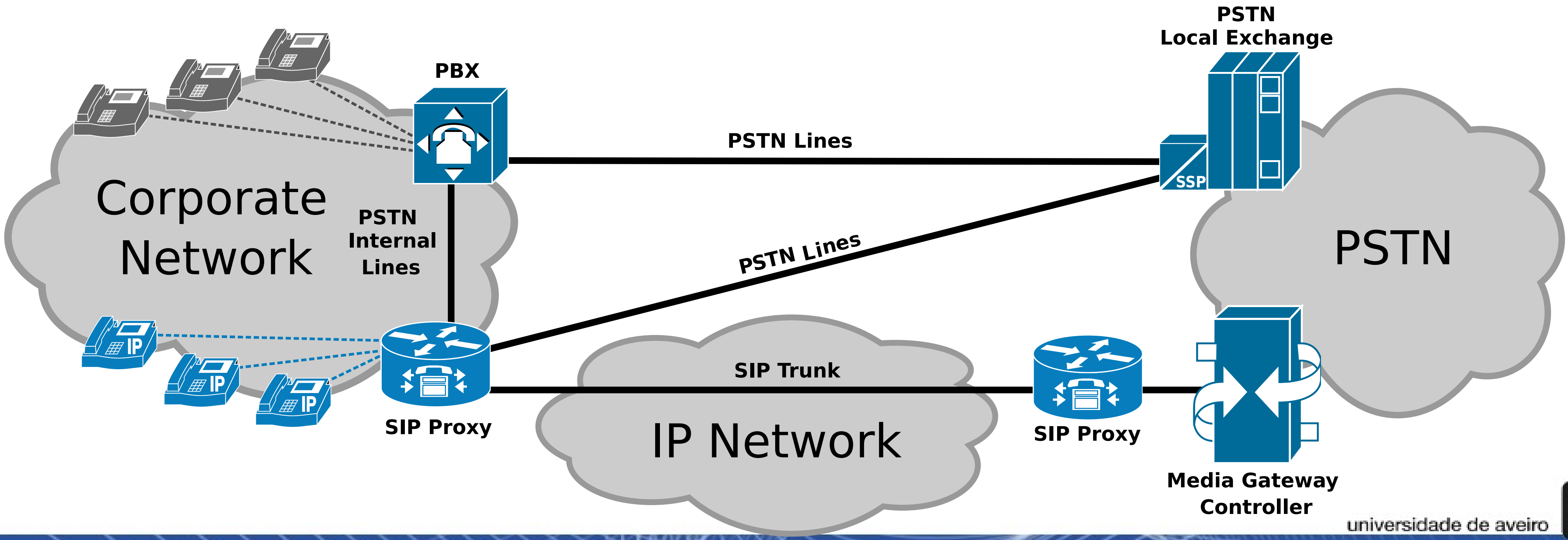
NAT SIP Application Layer Gateway (ALG)

- Required to translate SDP payloads.
- Heavy on NAT gateway.



VoIP and PSTN Connectivity

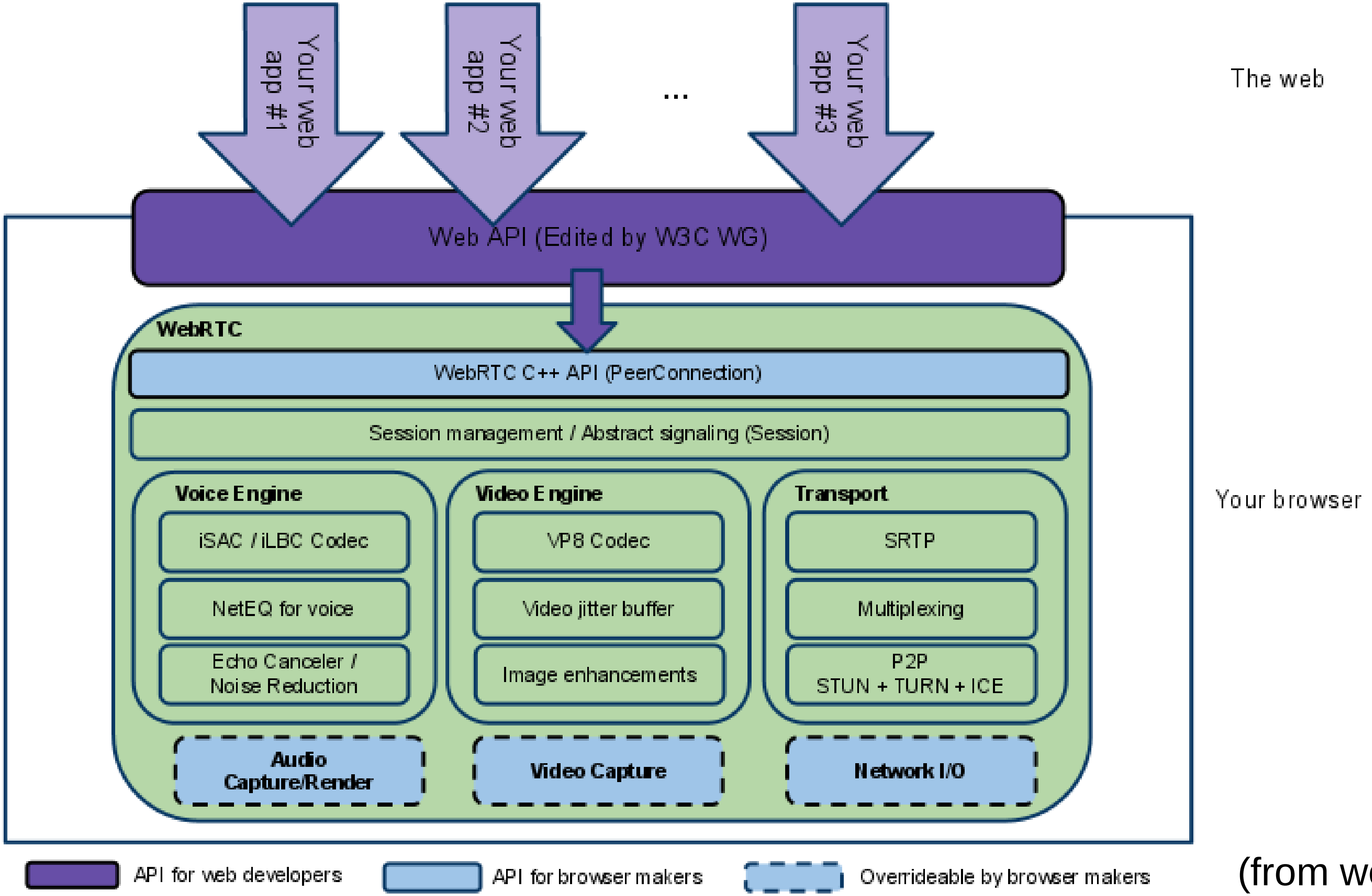
- SIP proxy.
  - With PSTN interface (to ISP or local PBX).
    - Requires multiple PSTN Lines.
    - Not scalable.
  - With SIP trunk to remote SIP proxy.
    - Remote proxy/gateway interfaces with PSTN network.
    - Remote proxy/gateway owned by PSTN ISP or by a third-party entity.
    - Usually TCP/IP transport with a TLS security layer.
    - Scalable!



WebRTC

- WebRTC (Web Real Time Communications) is an open source communication technology.
- Typically used for real-time audio and video communications.
- Provides:
  - Peer-to-peer connections.
    - An instance allows an application to establish peer-to-peer communications with another instance in another browser, or to another endpoint implementing the required protocols.
  - RTP Media transport.
    - Allow a web application to send and receive media stream over a peer-to-peer connection.
  - Peer-to-peer Data transport.
    - Allows a web application to send and receive generic application data over a peer-to-peer connection.
  - Peer-to-peer DTMF.

WebRTC Architecture



(from webrtc.org)