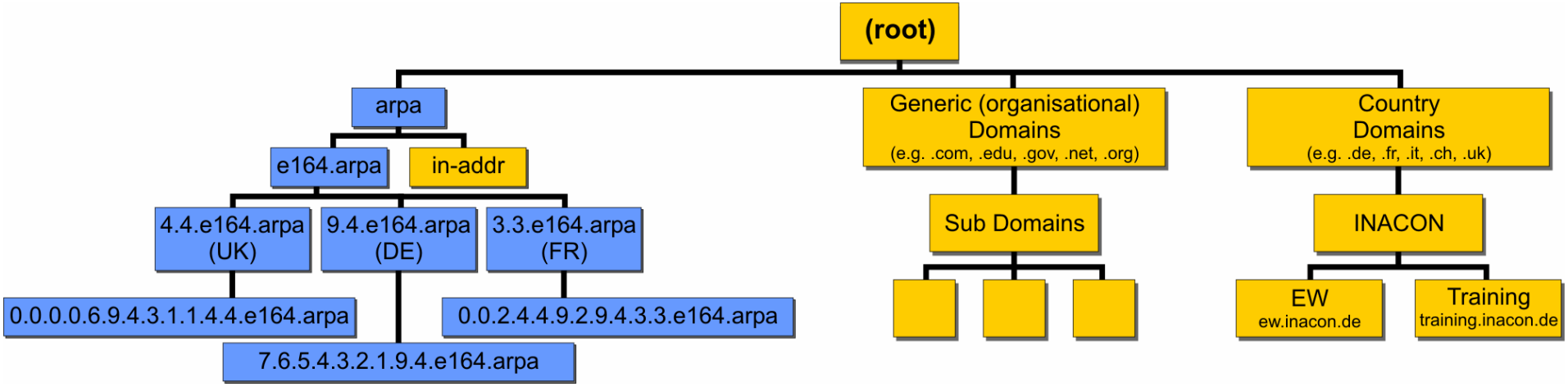
DNS and ENUM Organization Overview

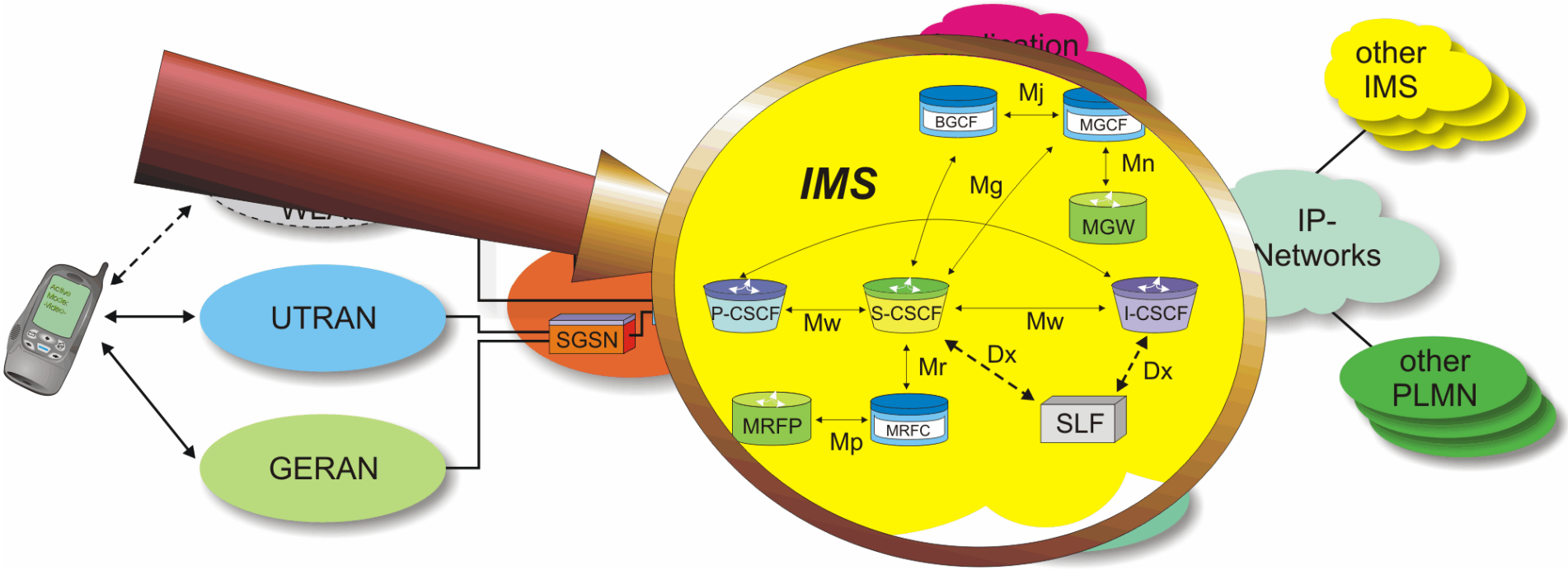


DNS and ENUM Organization Overview

DNS
A system of hierarchically domain name servers used to provide fully qualified domain names (FQDN) or corresponding IP addressed to a requesting host, depending on the type of request.

ENUM
Closely associated to DNS, an ENUM system provides a translation of E164 phone numbers into fully qualified domain names (FQDN). These IP names can then be used to obtain corresponding target addresses from a DNS server.

And what is inside the IMS?

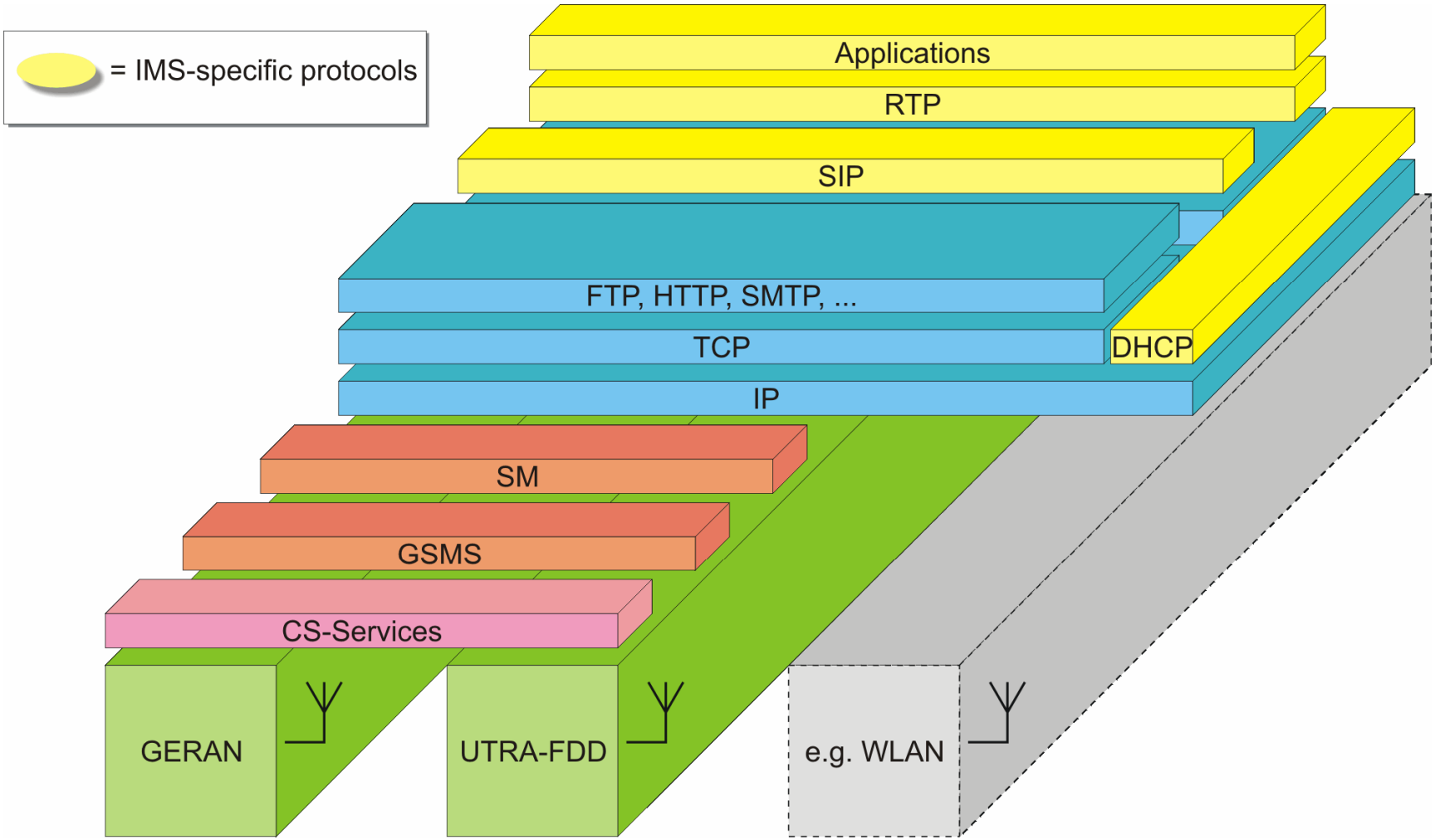


And what is inside the IMS?

The IMS is entirely based on IP as transport protocol. It therefore hosts IP-driven servers of which the majority uses SIP (Session Initiation Protocol) to communicate internally and externally. Still, other protocols are also used within the IMS.
Other network elements within the IMS are media gateways and their controllers.

[3GTS 23.228]

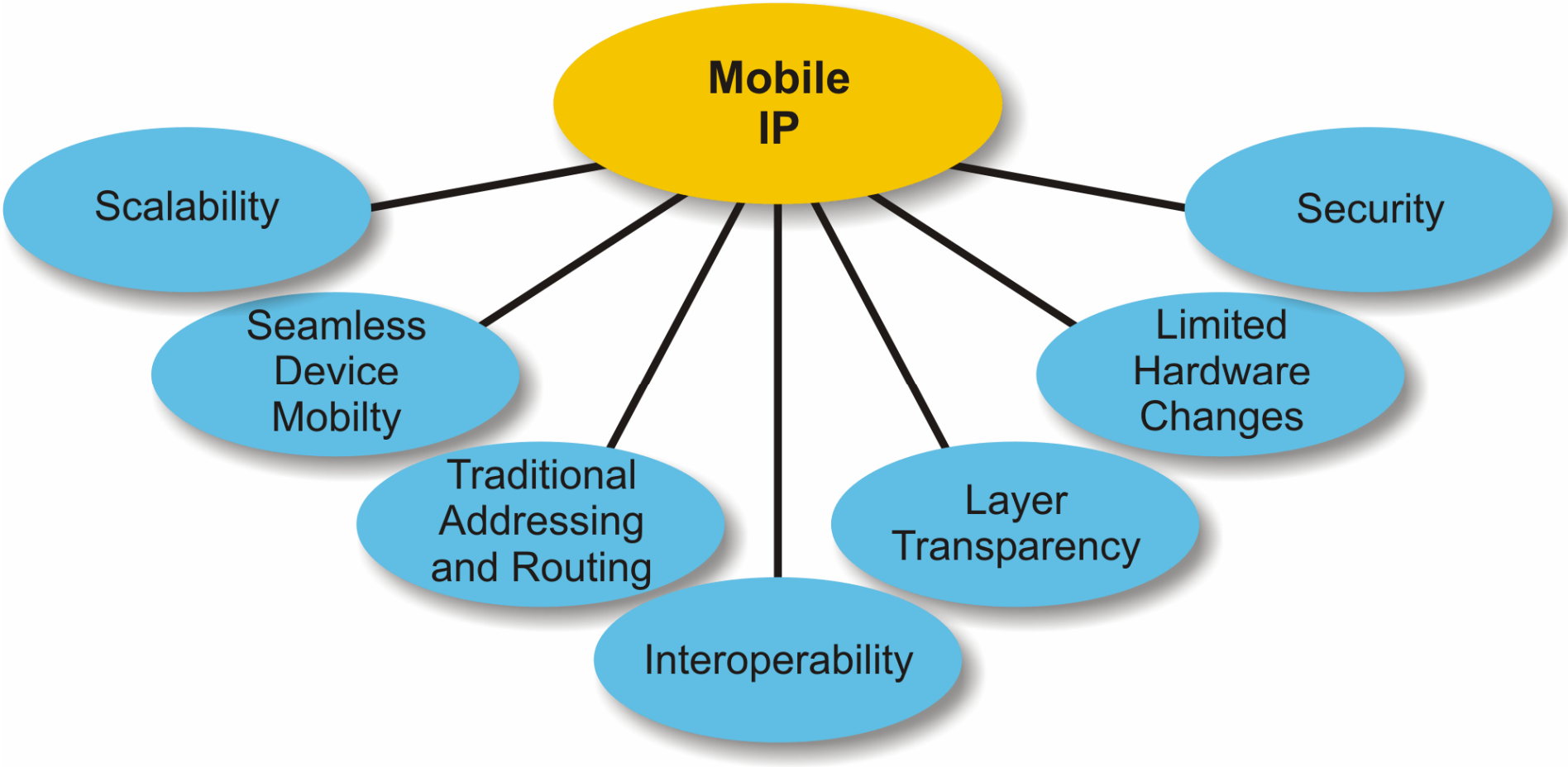
The Perspective of the Mobile Station



The Perspective of the Mobile Station

- ⇒ The figure illustrates in sufficient detail the complexity of a multi-mode mobile station to use the IMS as peer. Again, focus of any 3GPP-compliant mobile station will be the support for GERAN and UTRAN radio access technologies but this does neither preclude mobile vendors nor network operators to allow a mobile station access to the network operator's IMS also through an IEEE 802.11 radio access network.
- ⇒ The figure emphasizes the protocol stack of such a mobile station
- ⇒ Note that despite of all its complexity, the figure still suppresses many complications like the possible need for dual-stack IP-operation (IPv4 and IPv6) or the detailed protocols within the GERAN or UTRA-FDD "black boxes".

Mobile IP



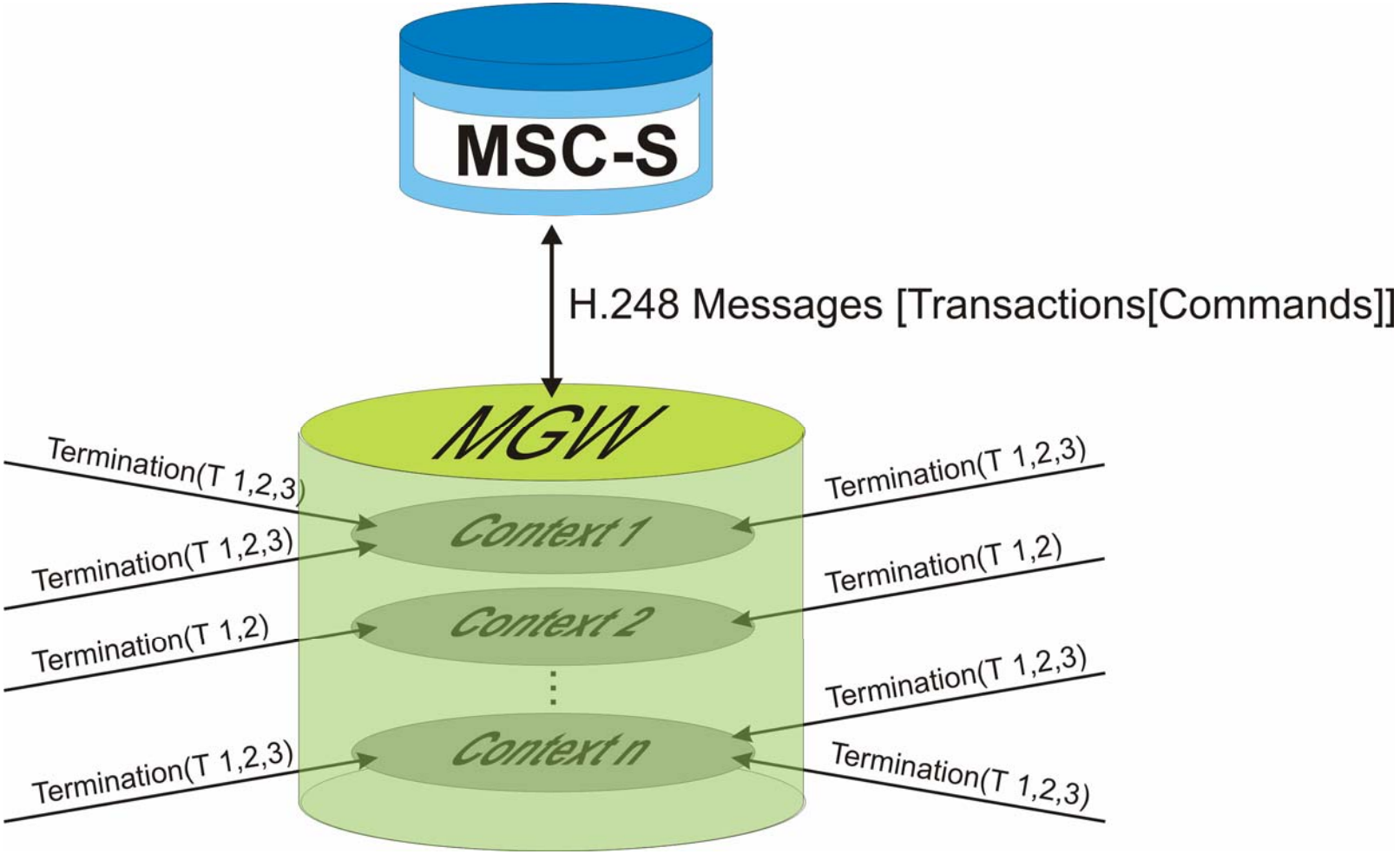
Mobile IP

- **Scalability**
  Allows a device to change from any network to any other. Example: One may detach a notebook from local desktop and move it to another country or simply to the office next door without impact on the operation.
- **Seamless device mobility using the existing device address**
  A mobile device may change the physical network attachment method and location. Still, it will continue to use the existing IP address.
- **Traditional addressing and routing**
  The assignment of IP addresses and the routing remains the same as in legacy IP. No new routing requirements such as host-specific routes are imposed on the internetwork.
- **Interoperability**
  Mobile IP devices can still communicate with existing IP devices, that do not know how Mobile IP works.
- **Layer transparency**
  The changes made for Mobile IP are limited to the network (IP) layer. The transport protocol layer and higher layer protocols will function as in legacy IPv4 systems. Existing connections can even be maintained across a move.
- **Limited hardware changes**
  Necessary software changes to enable Mobile IP are limited to the mobile device and to routers used directly by the mobile device. Any intermediate routers between home and visited network remain unchanged.
- **Security**
  Mobile IP uses a system of redirected messages. It includes authentication procedures.

Mobile IP implements a system where datagrams are generally sent to the mobile's home location and are then forwarded to wherever it is currently located (⇔ visited network).

[RFC 2002]

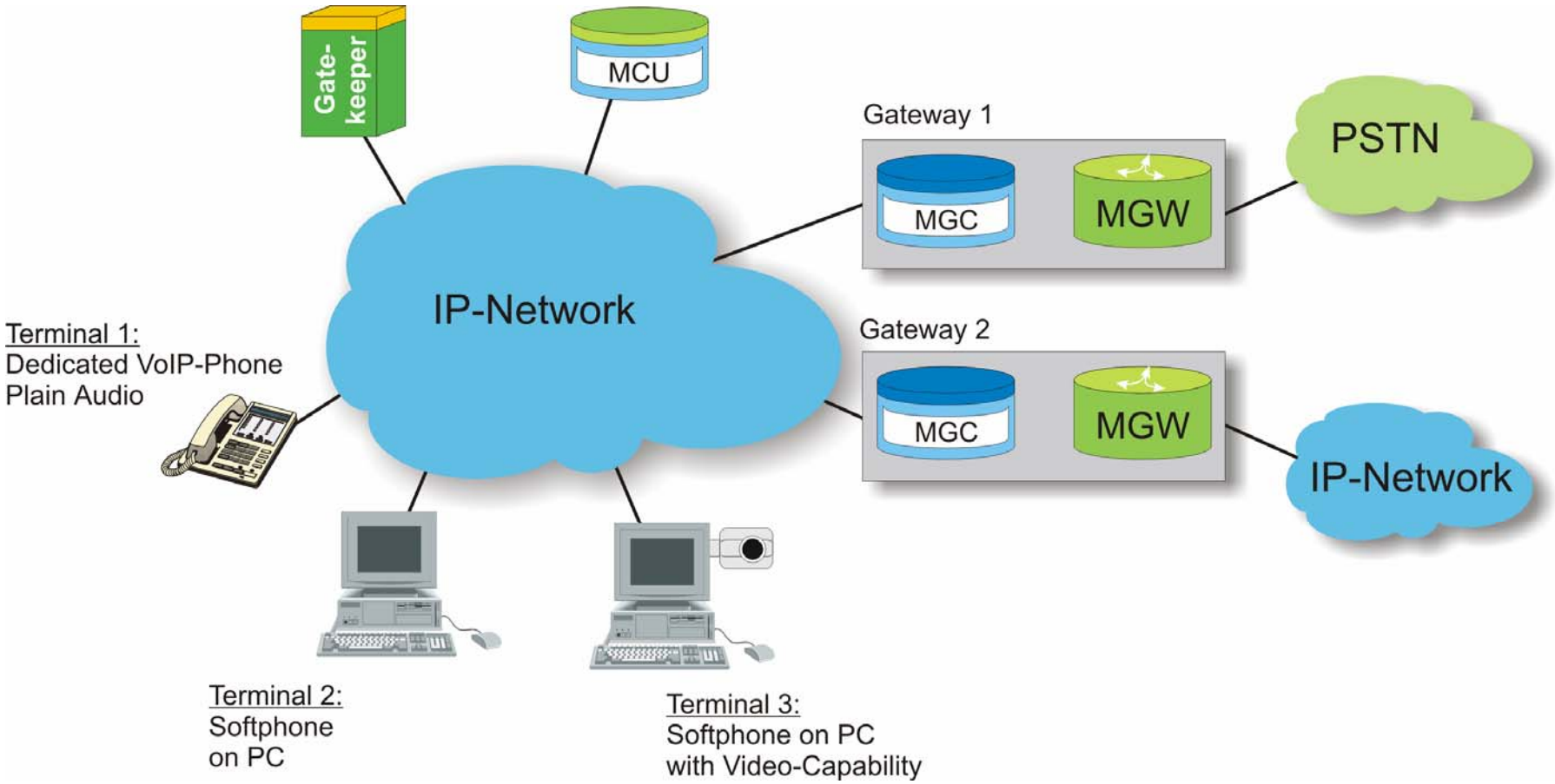
Interworking MSC-S ⇔ MGW



Interworking MSC-S ⇔ MGW

- **The MSC-Server is used to interpret and route the call control signaling.**
- **The relationship between MSC-Server and Media Gateways is not one-on-one but allows for a shared use of one media gateway by different MSC-Servers.**
- **The Media Gateway is required to provide the necessary bearers.**
- **MSC-Server and Media Gateway communicate with each other through H.258/Megaco-messages.**

The H.323-Protocol and Network Architecture



The H.323-Protocol and Network Architecture

Note:

- H.323 was originally targeted at the definition of packet-switched multimedia communication and provides the guidelines for the related network architecture and signaling control functions.
- Nowadays, H.323 is the most frequently used protocol standard for VoIP.
- H.323 is actually an entire standards family which most importantly uses H.225.0 for call control (⇔ similar to ISDN-call setup messages (Q.931)), H.245 for the terminal capabilities exchange (e.g types and versions of codecs) and RAS (Registration, Access Control and Status) for the communication between gatekeeper and terminals.

Network Architecture

The figure illustrates a typical H.323-network configuration which consists of:

H.323-Terminals

One option of H.323-compliant terminals may be dedicated IP-phones with plain audio function. Note that H.323 mandates the availability of audio codecs in an H.323-terminal while the support of video is optional. Other options of H.323-compliant terminals are softphones on a PC which may or may not support video transmission and reception.

Gatekeepers

Gatekeepers are optional network elements that however serve important functions like admission control on the network. If gatekeepers are present in the network, their services have to be used by the other network elements. Terminals register their presence in the network with a gatekeeper and, depending on the gatekeeper's setup, the entire signaling or only part of it may be conducted through the gatekeeper.

Gateways

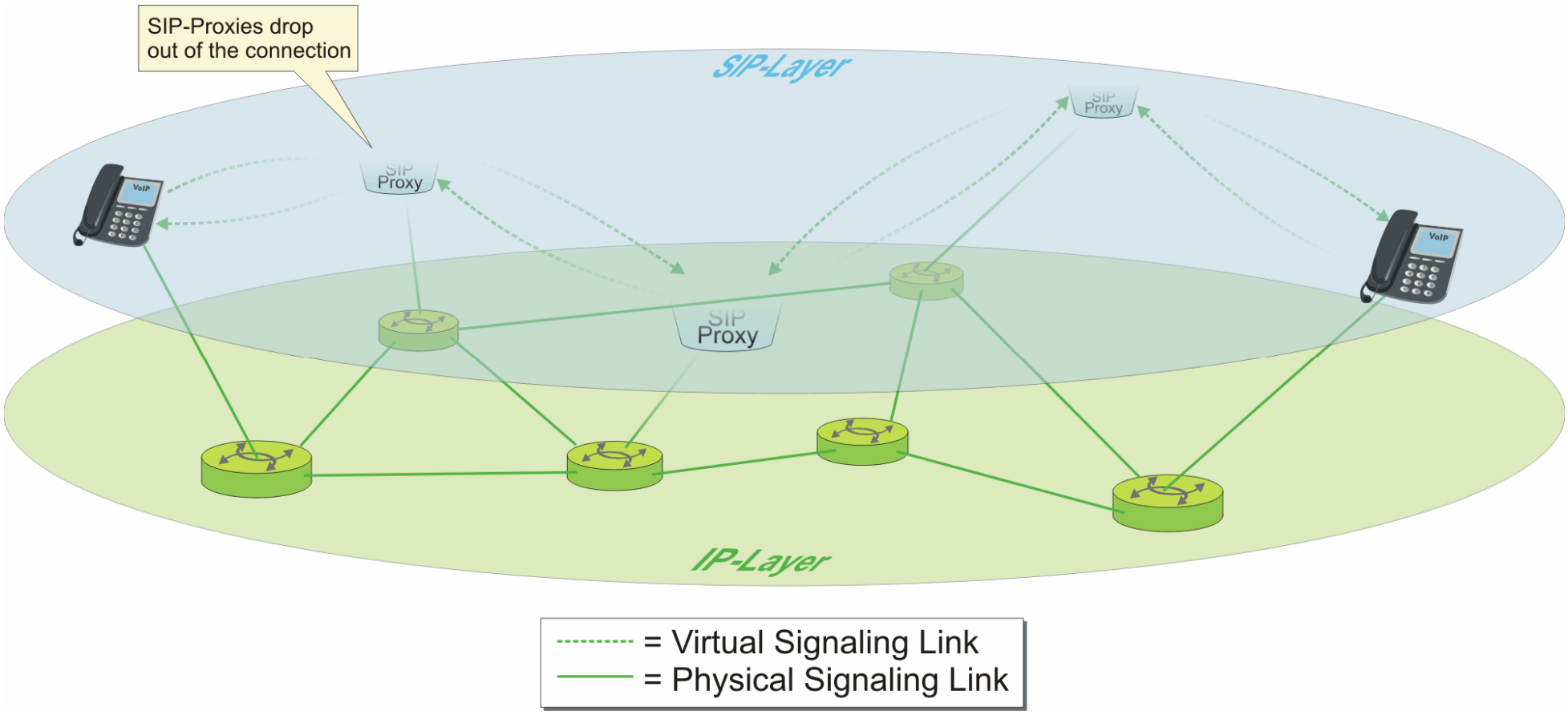
Gateways are network elements that interconnect and interface H.323-networks to the PSTN or to other H.323-networks. Gateways may be split into MGC (Media Gateway Controller) and MGW (Media Gateway).

Multipoint Control Unit

An MCU provides conference call services to allow three or more terminals to participate. An MCU consists of a multipoint controller for the call control functions and optional multipoint processors for handling the media channels.

[ITU-T H.323]

Session Completion Phase



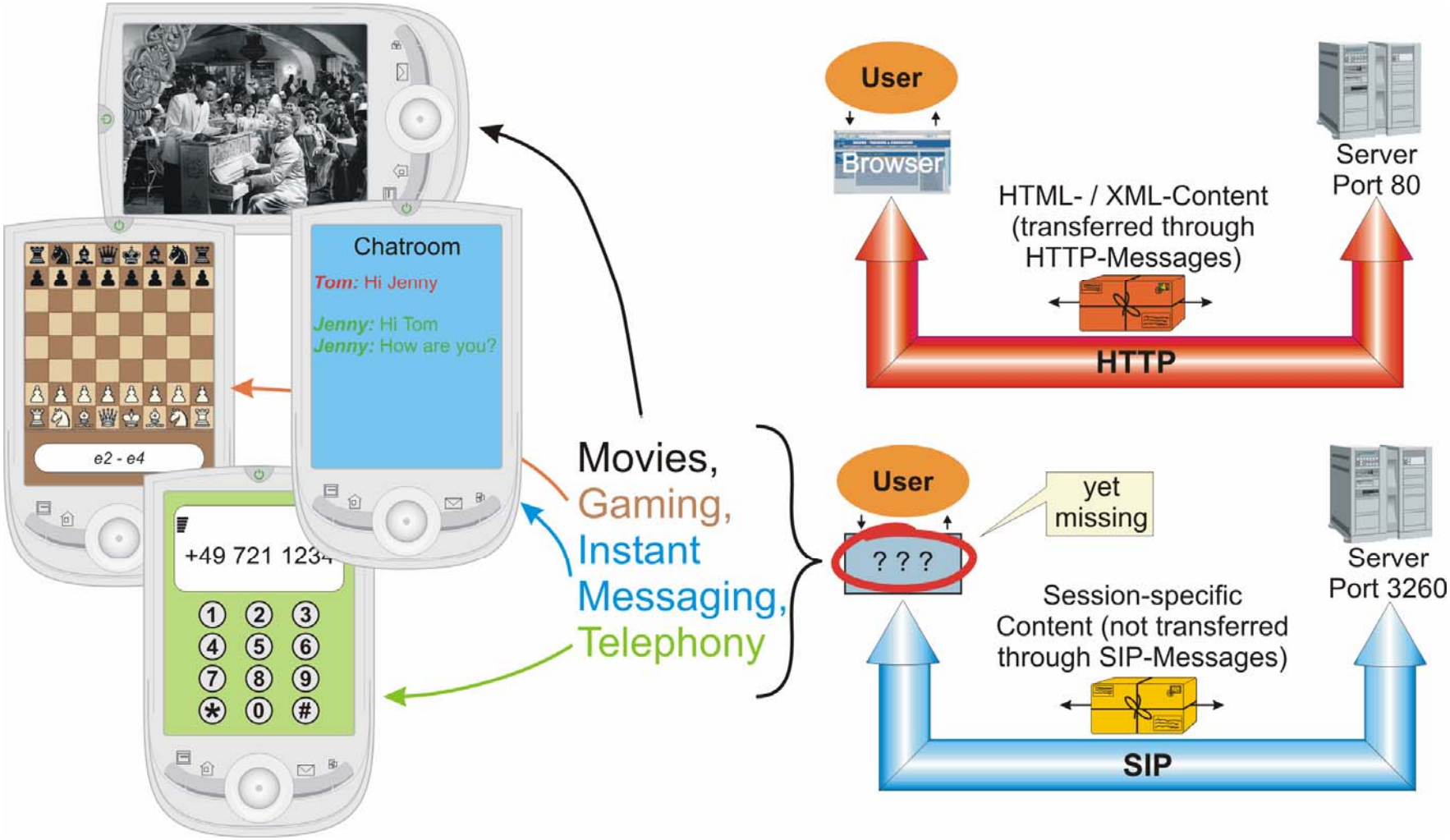
Session Completion Phase

⇒ Note that the SIP-proxy servers drop out of the communication chain. This is the regular way of processing in SIP. That is, after the setup of the communication channel, there is no more involvement required of the proxies.

⇒ This statement is, however, only true if standard conditions apply: No NAT, no media conversion needed, no IP-version Interworking between the two peers…)

Note that operators may also configure certain means to assure that the proxies remain in the signaling chain and are also receiving keep-alive messages periodically from one or both peers.

Comparison between SIP and HTTP



Comparison between SIP and HTTP

- **To get a better feeling about SIP a comparison with HTTP is helpful**
  - ⇒ HTTP is the protocol behind and underneath web browsing and it is extremely prominent on the World Wide Web. HTTP enables *all* applications on the internet that are accessed through a web browser. Typical examples for web browsers are the Internet Explorer or Firefox.
  - ⇒ The message format, the request/response philosophy and many response codes (e.g. 200-OK) of SIP have been inherited from HTTP. One can legitimately say that SIP is at least formally based on HTTP.
  - ⇒ Note that HTTP is the transfer and control protocol for the "content" but the content itself is not HTTP. It consists of HTML-/XML-encoded information which in turn may incorporate other information like JAVA or JAVA-SCRIPT.

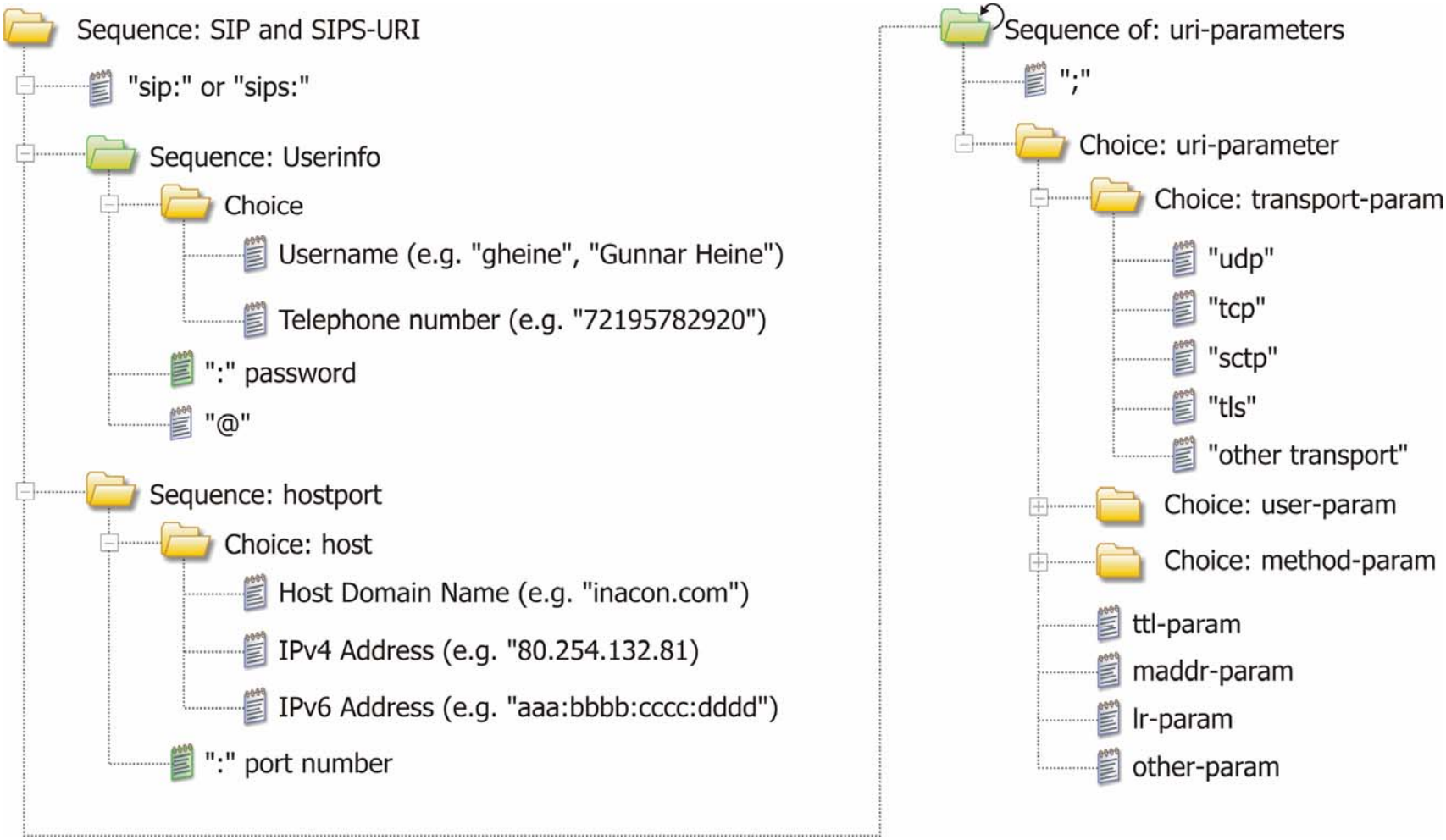
Note:
- In difference to SIP, HTTP is also used to embed the previously mentioned "content" into HTTP-packets for its transfer between peers.
- In HTTP, a "session setup" occurs inherently and static with the first HTTP-message exchange that identifies the capabilities of both peers and which clarifies whether these peers can exchange "content" in the first place. In other words, the actual session setup phase is very simple.

- ⇒ Usually, a client or user accesses an HTTP-server from his/her web browser to download a website from that server to the browser and afterwards the client will, possibly interactive, process the downloaded content through his/her web browser.
- ⇒ SIP on the other hand uses a "user client device" to manage a session between that "user client device" and other "user client devices" or towards an application server.

Problem for SIP:
- The vast number of possible *session types* (e.g. telephone calls, games, location services …) to be established through SIP makes it difficult to provide or define a standardized generic "user client device" like the web browser for HTTP to support all possible session types.
- Still, we are almost convinced that such a definition will occur to enable the development of an unlimited number of applications on top of SIP. After all, it was only the invention of the web browser about 10 years ago with its GUI which made the internet gain speed for applications beyond messaging.

Conclusion: Those who want to use SIP as basis for application development need to clarify in a first step which additional features SIP provides compared to HTTP (e.g. multi-party vs. two-party, user-to-user communication is possible…). In a second step a generic "SIP-browser" needs to be invented that will allow for the necessary economies of scale and that will enable the application development itself rather seamlessly.

The SIP(S)-URI

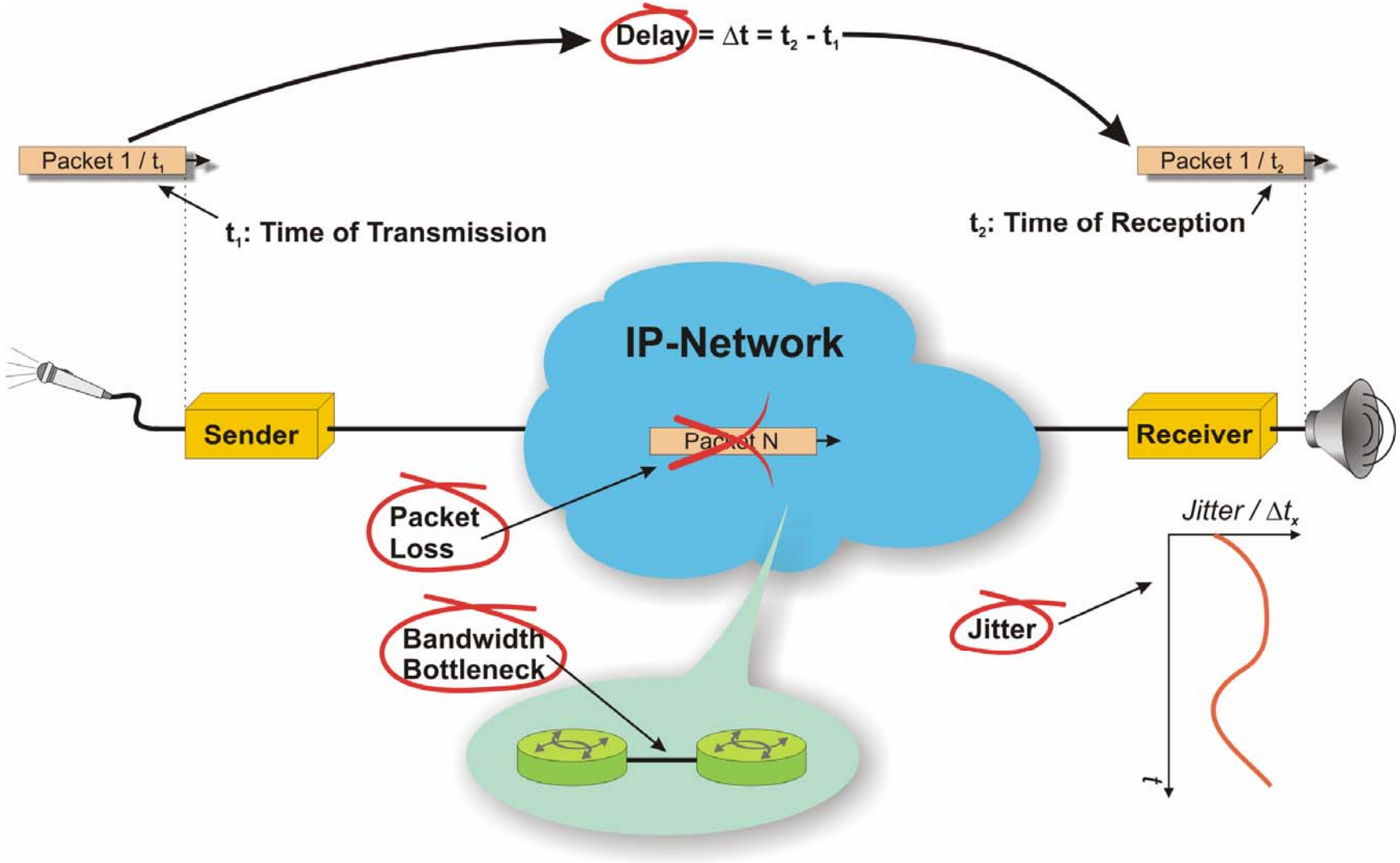


The SIP(S)-URI

- **SIP or SIPS**
  The SIP(s)-URI starts with the text string “SIP:” to indicate that the following identifiers relate to a SIP-URI or with the text string “SIPS:” to indicate that a SIPS-URI follows.
- **Userinfo**
  The presence of the element “userinfo” is optional but highly recommended to identify a user using his / her fully qualified domain name (e.g. baulbrause@cakao.net). The userinfo consists of either a username or a telephone number. Either element may be followed by an optional password which is separated from the userinfo through the character “:”. The use of a password in the Request-URI is not recommended by the specifications. In either case, the userinfo ends with the character “@”.
- **Hostport**
  The presence of the element “hostport” is mandatory. It consists of:
  ⇒ the element “host” which includes either an IP-address (in which case there would be no userinfo) or a host domain name (e.g. cakao.net). Note that the use of IP-addresses as host-ID is not recommended.
  ⇒ optionally a port number where this request shall be sent to at the receiver side (the default port number for SIP is ‘5060’$_{dec}$.
- **uri-parameters**
  The presence of uri-parameters is optional. If one or more uri-parameters are present, their listing is separated from the element “hostport” through the character “;” which is also used to separate different parameters from each other. Different parameters depend on each other. For instance, only when the maddr-parameter (server address of that user) identifies a multicast IP-address, then the “ttl”-parameter (time to live) shall be present.
  The “transport” parameter specifies the transport protocol to be used for SIP-messages relating to this request. The “lr”-parameter indicates that the sending UA uses loose source routing.

  [RFC 3261 (19.1)]

Problems of VoIP



Problems of VoIP

Delay

For real-time applications like voice transmission, the ITU-T G.114-recommendation mandates a one-way latency of ≤ 150 ms. When the latency is higher, people feel annoyed and start to talk over each other.
Inherently, delay is an issue in packet-switched networks, because resources are only provided on demand and usually, there is no guarantee that the resources are available when required. Delays are the consequence. This applies in particular to IP-networks. Altogether, the following sources for delay can be identified in VoIP:

1. Packet Inherent Delay:
   Depending on the technical approach, more or less speech samples need to be collected before a packet is sent to the peer. Usually, the periodicity of the packet transmission is 20 ms which means that the first speech samples in each packet have an inherent delay of 20 ms.
2. Processing Delay:
   The speech samples need to be encoded and compressed before the transmission which requires processing time. The same applies to the receiver side. The amount of processing time depends on the selected encoder/decoder and the available processing resources in both, the sender and the receiver.
3. Transfer Delay:
   Transfer delay relates to the process of actually transmitting a speech packet to the peer through possibly numerous intermediate network elements. The transfer delay may vary (⇔ jitter) which poses additional problems.

Jitter (Packet Arrival Time Variance)

Jitter relates to the variance of the delay time to transfer a speech packet from sender to receiver. Obviously, the receiver needs to find a way to cope with the jitter.

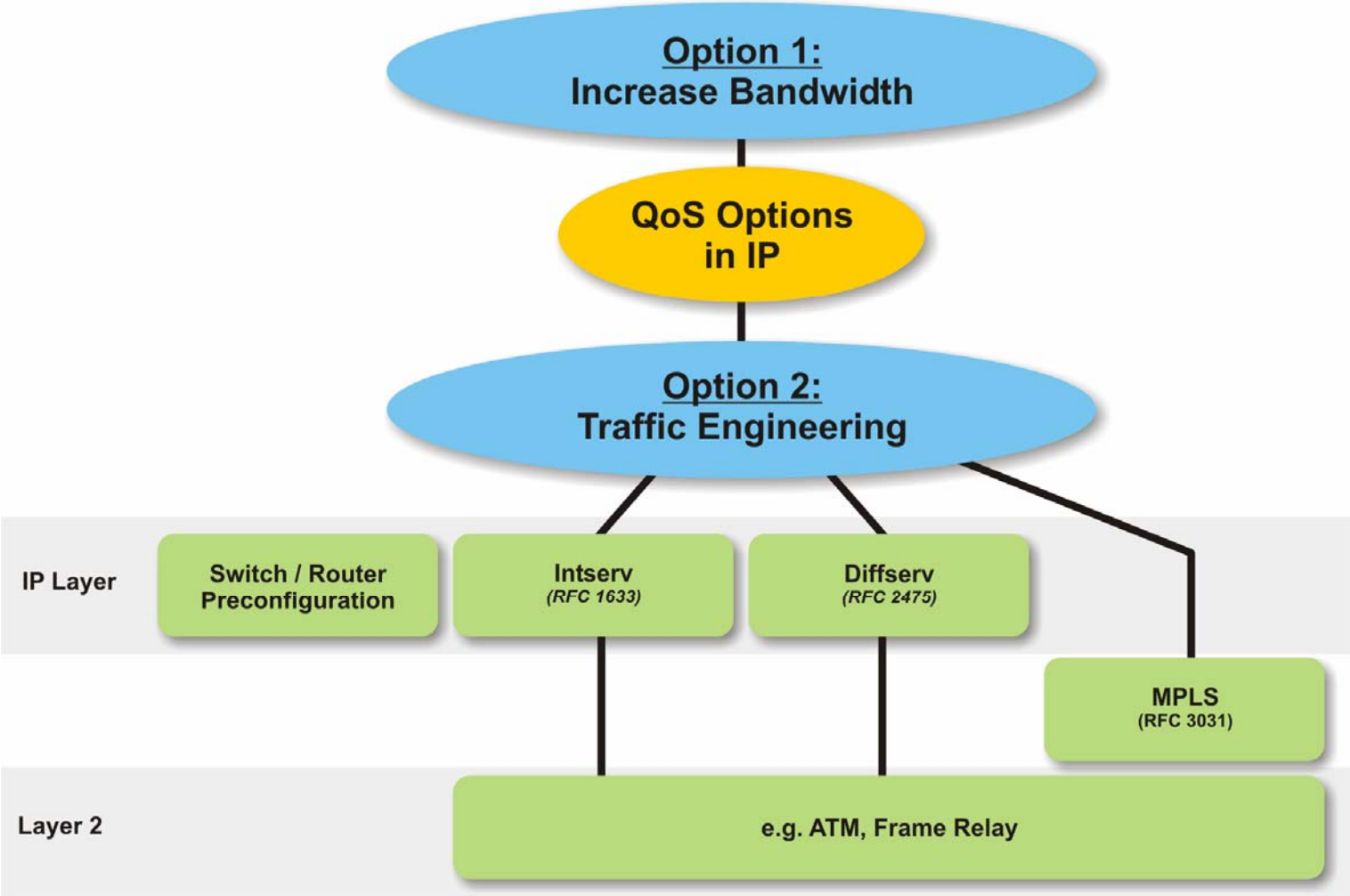
Packet Loss

In an IP-network, packets may be lost or intentionally discarded (e.g. during overload) and it makes no sense to retransmit such a packet because of the time-critical nature of VoIP. A packet which is too late is actually a lost packet.

Bandwidth Bottlenecks

Bandwidth bottlenecks are typical on the internet and are a consequence of its “best effort” and packet-switched characteristics.

QoS Options in IP-Networks



QoS Options in IP-Networks

Within the IETF, there are different strategies how to provide QoS in an IP-network:

- **Large bandwidth reserve**
  Still the most important technique used in real-world networks to control QoS in IP. Under utilized networks (< 50 %) usually provide sufficient reserves for most of the QoS requirements of actual applications.
- **Intserv (RFC 1633)**
  Intserv is based on the end-to-end establishment and reservation of resources prior to any information transfer. In that respect, Intserv establishes something like a tunnel for a specific data flow.
- **Diffserv (RFC 2475)**
  This approach reuses the "Type-of-Service" field in the IP-header to allow the tagging of IP-frames with different QoS-requirements.

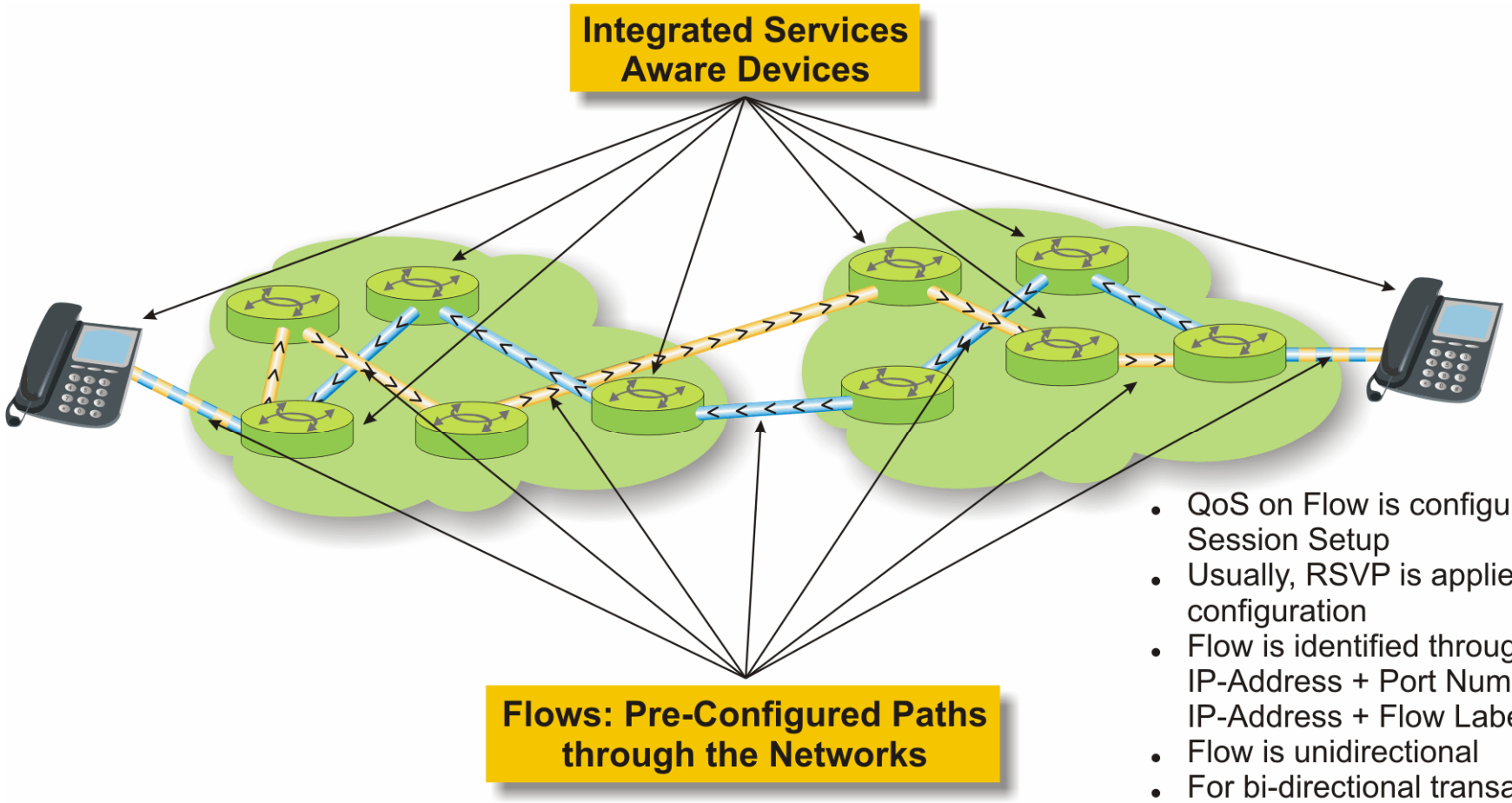
Note: Both strategies require that each device in a network can be configured and setup to provide a given QoS. This is usually not possible on the public internet.

- **Traffic Engineering**
  The ability to route primary paths around known network bottlenecks and points of congestion to optimize network utilization and plan resources based on known demand. Requires mechanisms to precisely measure and control network parameters (⇔ Traffic Inspection). Effective use of Traffic engineering can substantially increase the usable network capacity.

- **Multi Protocol Label Switching, MPLS (RFC 3031)**
  Data transport and routing mechanism, that is transparent to the type of traffic. It provides connection oriented routing for groups of packets (flows) which share the same requirements (QoS, traffic demands) between endpoints of a MPLS domain. Fast traffic switching is enabled, as the Routers do not need to examine IP header or payload information but use the routing label information only. MPLS thus establishes a sort of tunnel for the transported layer 2 traffic. It interfaces to existing routing protocols such as RSVP, OSPF, BGP, LDP.

- [RFC 1633, RFC 2475, RFC 2998, RFC 3031]

Operation of Integrated Services



- QoS on Flow is configured during Session Setup
- Usually, RSVP is applied for configuration
- Flow is identified through
  IP-Address + Port Number (IPv4)
  IP-Address + Flow Label (IPv6)
- Flow is unidirectional
- For bi-directional transactions two flows need to be setup

Operation of Integrated Services

The alternative approach to *Differentiated Services* to provide QoS in an IP-network is called *Integrated Services*, because the QoS-sensitive services can be integrated into an existing best effort IP-network. However, as the figure illustrates, the involved network elements and especially the routers need to be aware of the *Integrated Services* and they need to be able to configure or deny the requested QoS. *Integrated Services* are specified in two variants:

- **Controlled Load Services (RFC 2211)**
  This variant can provide QoS for real-time applications which are still able to cope with a slightly variable QoS. The name stems from the philosophy that controlled load services provide a QoS that is equal to the QoS which is provided by a slightly loaded IP-network. The difference to best effort traffic is that in the case of increasing network load, the controlled load services will still obtain the negotiated QoS. However, using this type of service no specific delay time reservations are possible.

- **Guaranteed Services (RFC 2212)**
  This variant is also aiming at real-time applications but guarantees the requested QoS end-to-end along the path. It kind of simulates an idling network. The most important difference to controlled load services is that guaranteed services allow for “locking” the end-to-end delay time.

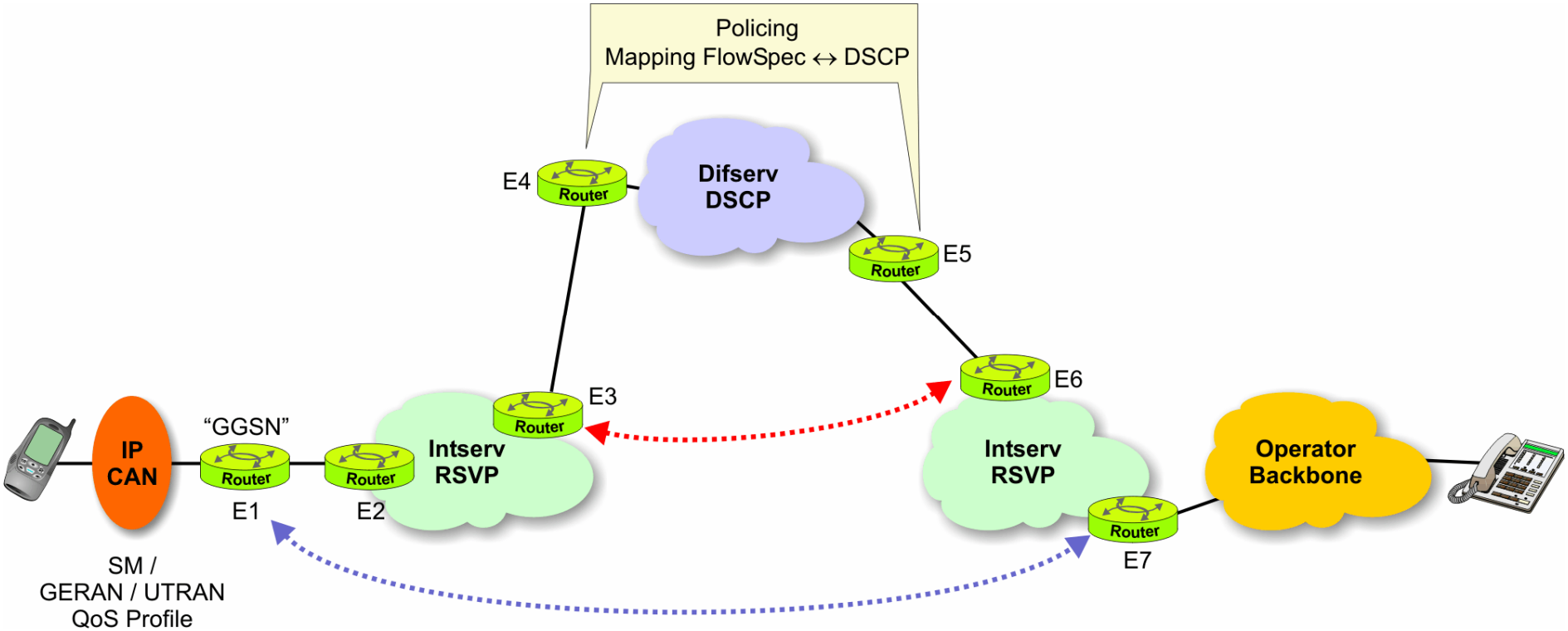
  ⇒ In general, *Integrated Services* are based on the definition and configuration of a flow along the path between sender and receiver. This flow is setup upon session setup.
  ⇒ Typically, the Resource Reservation Protocol (RSVP / RFC 2205) is applied during session setup (e.g. telephone call) to configure the QoS for a specific flow. Alternatively, manual configuration or a network management protocol can be used. No specific solution is specified.
  ⇒ Each network element along the path needs to keep track of the flow identifying information (IPv4 ⇒ Destination IP-address / Transport Protocol (e.g. UDP) / Port Number or IPv6 ⇒ Destination IP-address / Flow Label) and the assigned QoS (priority, delay, buffer sizes…).
  ⇒ IP-frames that belong to a certain flow will be treated according to the configured QoS.

Note. It is not requested that all routers in a network must be IntServ aware. However for a given connection all addressed routers must be able to handle IntServ.

If there is only one router that cannot handle IntServ, this router will fall back to best effort traffic handling.

[RFC 1633, RFC 2210, RFC 2211, RFC 2212]

Routing over mixed IntServ – DiffServ Networks



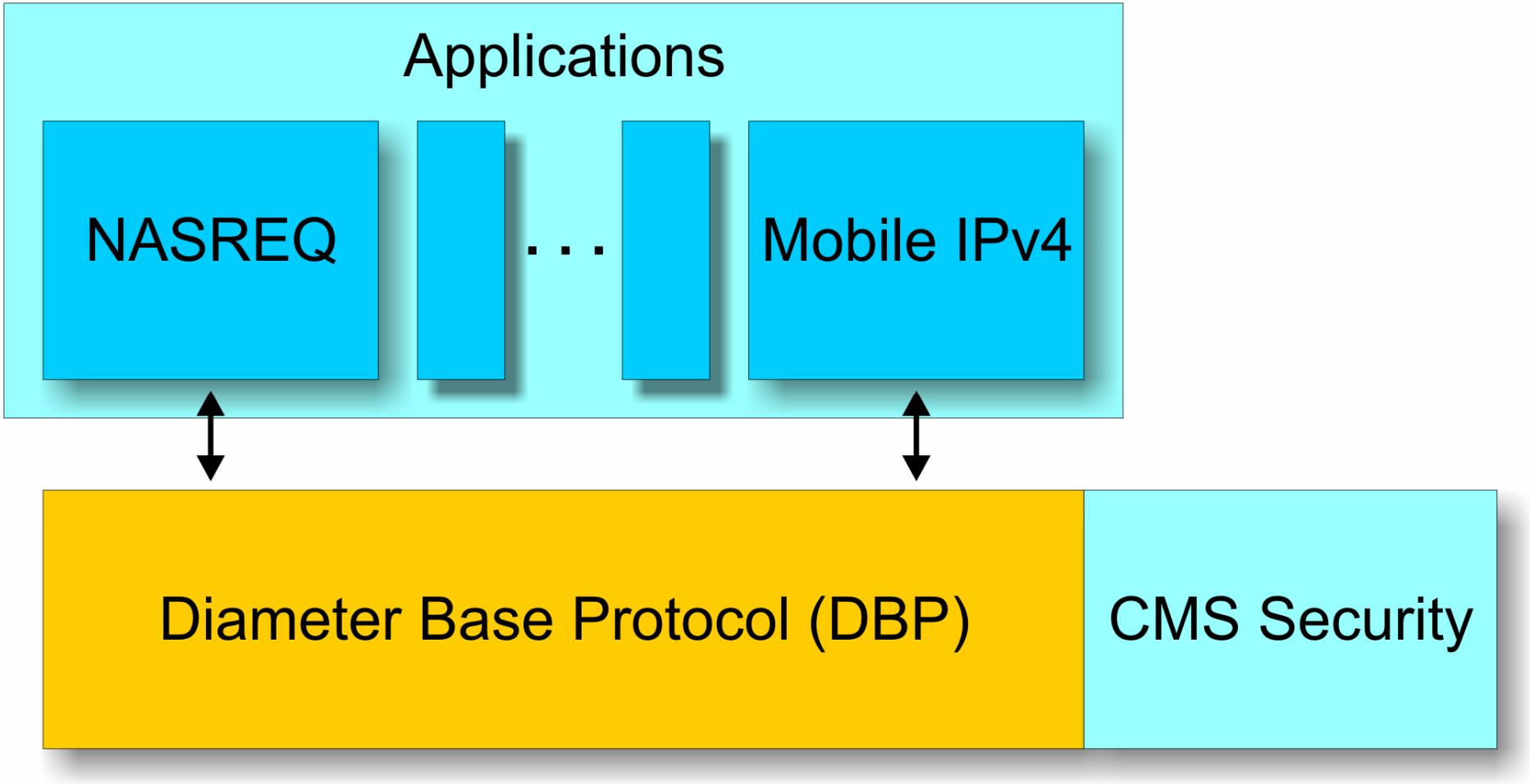
Routing over mixed IntServ – DiffServ Networks

If end to end QoS has to be established, several different types of networks may need to be considered. In our example, we have an access network that terminates at Edge Router E1 (a GGSN in that case), two intermediate Intserv aware networks with the Edge Routers E2, E3, E6 and E7. In between, there is a Diffserv network with the Edge Routers E4 and E5.

Intserv would need to establish a link through the Diffserv domain which is supposed to be not RSVP aware. At the Edge Routers E4 and E5 the Intserv Flowspec will be mapped into a DSCP bit pattern and vice versa. Any RSVP_PATH message is ignored within the Diffserv domain and the packets are forwarded to Edge Router E6 and onwards to the receiving host. The RSVP_RESV message generate will transparently be routed back through the Diffserv domain to the Intserv aware Edge Router E3. E3 will compare the available resources based on the Intserv to Diffserv mapping. If E3 approves the request, the RSVP_RESV message is sent to the original sender. Receipt of that message is interpreted as addmittion for the flow. Also the DSCP marking is learned and will be used for subsequent packets sent for this flow.

As an alternative, an IPsec tunnel could be established through the Diffserv domain. IPsec does not use the DS field in an IP header for the cryptographic calculations. Therefore, modifications of that field by a node has no effect on IPsec's end to end security.

DIAMETER Overview



DIAMETER Overview

Diameter protocol was designed as an improved version of the RADIUS protocol. A goal was to maximize compatibility and ease migration from RADIUS to Diameter. For example, a Diameter message, like a RADIUS message, conveys a collection of attribute-value pairs (AVP).

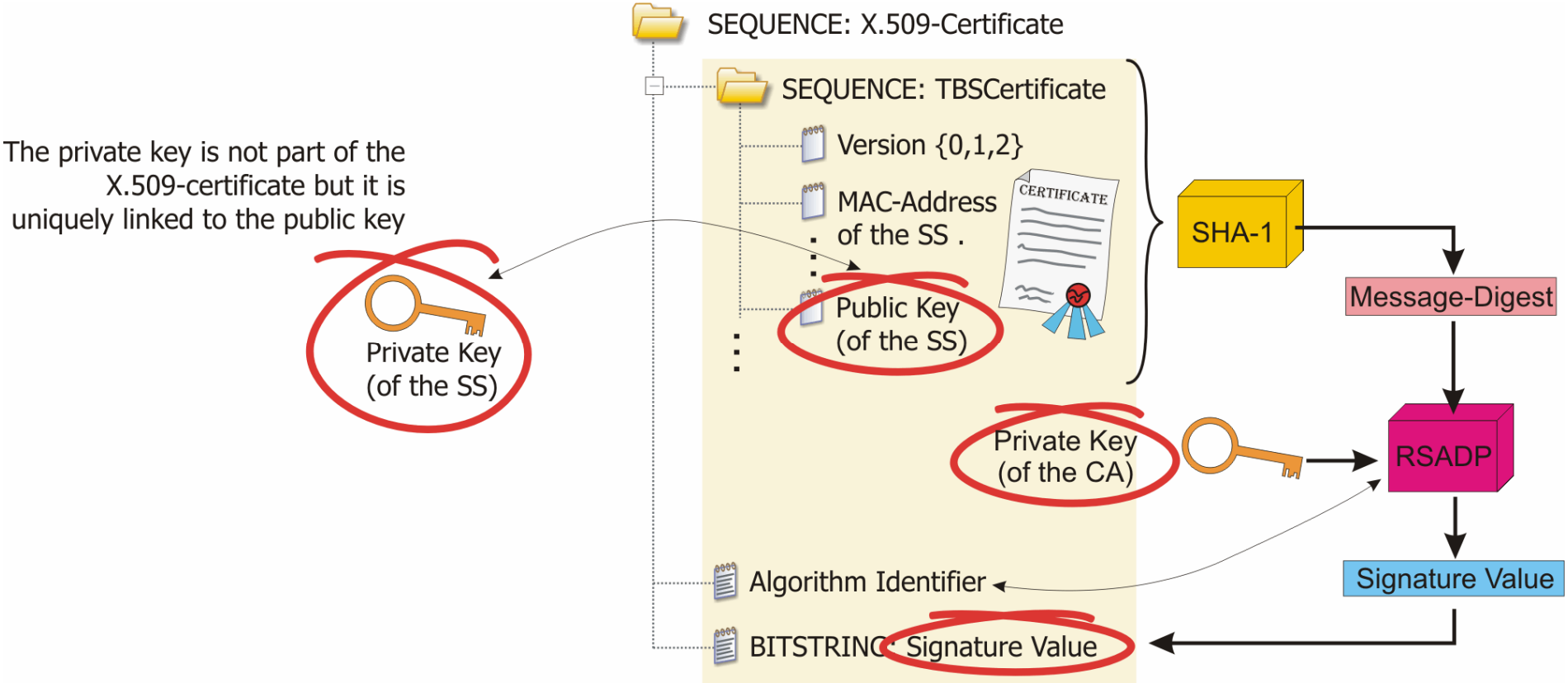
Diameter consists of the Diameter Base Protocol (DBP), a transport profile and a set of applications. The applications NASREQ and Mobile IPv4 provide for Authentication, Authorization and Accounting access, The Diameter Cryptographic Message Syntax (CMS) application provides end-to-end authentication, integrity, confidentiality at the AVP level.

DBP provides basic mechanisms for reliable transport, message delivery, and error handling and must be used in conjunction with a Diameter application. Each application relies on the services of the base protocol to support a specific type of network access.

DBP defines the basic Diameter message format. Data is carried within a Diameter message as a collection of AVPs.

Generating and Signing an X.509-Certificate

- **The X.509 certificate is generated by the CA and signed with the private key of the CA**
  Essentially, the CA ties the MAC-address of the SS and the allocated public key together.



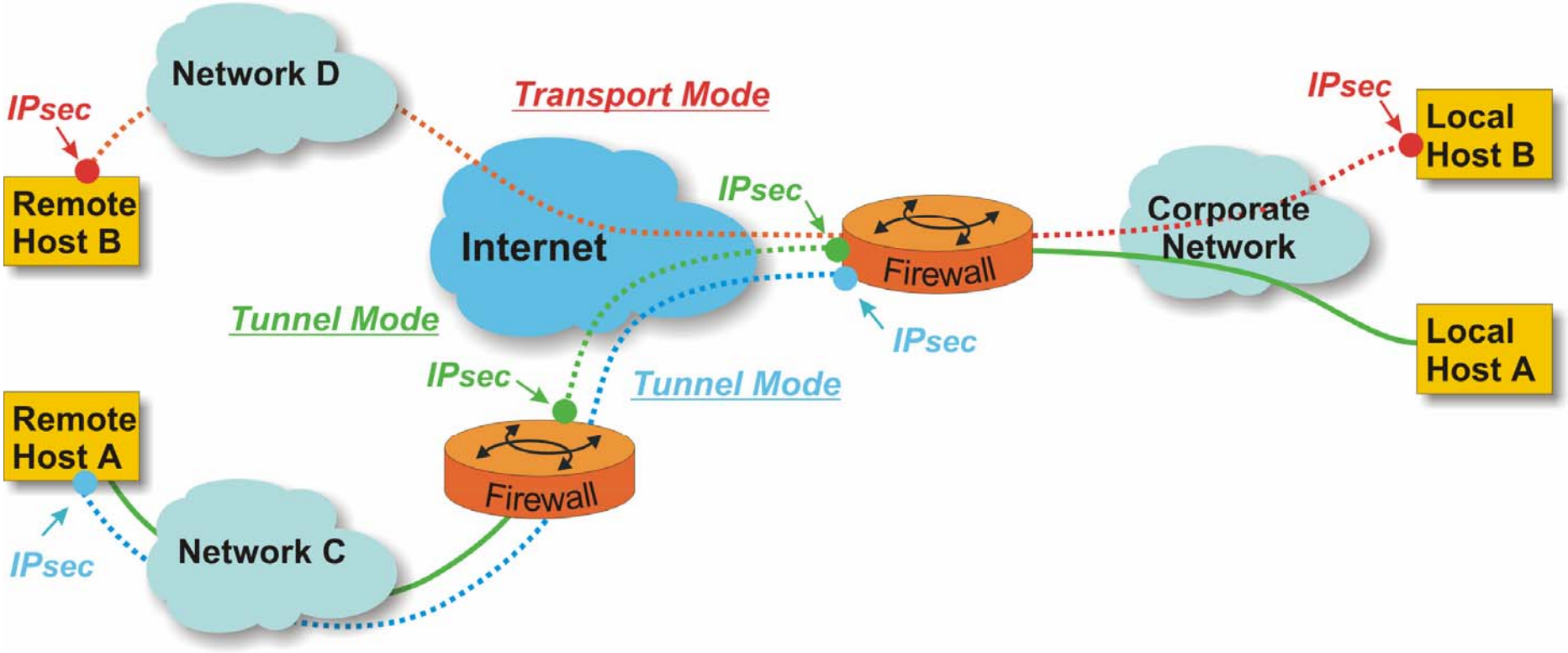
Generating and Signing an X.509-Certificate

Additional Information

⇒ There are two pairs of public key / private key: The first one belongs to the CA and the second one belongs to the SS.
⇒ Through its digital signature the CA approves that the SS with that MAC-address has been allocated the related SS public key and SS private key.
⇒ The X.509 certificate is usually sent unencrypted. This is no problem since any tampering with it will immediately be recognized when an altered certificate is sent to a second party.

[FIPS 186; FIPS = Federal Information Processing Standard]

VPN with IPsec in Tunnel Mode and Transport Mode



VPN with IPsec in Tunnel Mode and Transport Mode

If IPsec and VPN-technology is deployed, the two operation modes transport and tunnel mode need to be distinguished:

VPN with IPsec in Tunnel Mode

The standard mode of VPN-operation is the tunnel mode. In tunnel mode, two network operators have negotiated a service level agreement (SLA) and have exchanged relevant security information. Whenever needed or permanently, an IPsec tunnel is established between the two networks. The end users who communicate between the two networks remain unaware of the security mode and of any details related to security.
Another implementation of tunnel mode is indicated through the blue dotted line: Remote Host A has established an IPsec tunnel to the security gateway of the corporate network. This implementation is almost end-to-end as the communication through the blue link is secured also on its way through network C.

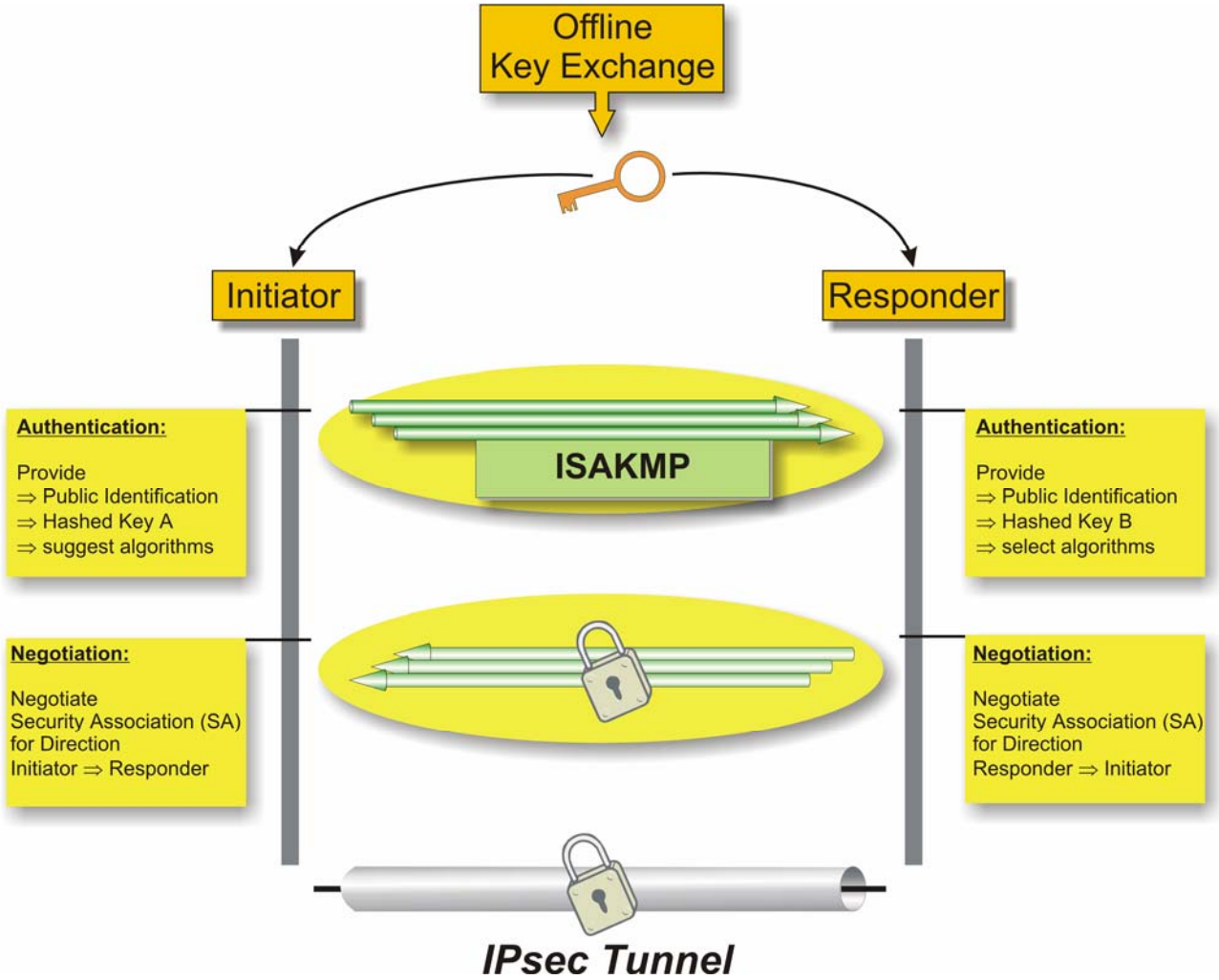
The tunnel mode is very appealing for PLMN operators offering GPRS. For certain subscribers (to be identified through their IMSI), the PLMN-operator offers an IPsec-tunnel to the corporate network of these subscribers. Obviously, there can be as many tunnels to as many corporate networks as necessary.

VPN with IPsec in Transport Mode

In transport mode we really have no VPN at all. As a matter of fact, in transport mode there needs to be an IPsec “tunnel” established between any two hosts on two different networks (in our example it is Remote Host B and Local Host B).

[RFC 2401 / RFC 2402 / RFC 2406]

Establishment of an IPsec-Relationship



Establishment of an IPsec-Relationship

ISAKMP (Internet Security Association and Key Management Protocol)

To establish an IPsec-connection, some communication over the insecure internet has to be performed. As the figure illustrates, the ISAKM-Protocol has been standardized (⇔ RFC 2409) to perform this communication and to allow for a secured handshaking procedure for IKE (Internet Key Exchange) and IPsec Security Association (SA) negotiation. In the first phase of a ISAKMP-handshaking procedure, information is transmitted in plain text which can eavesdropped. However, the IETF provides for three options to confirm the identity of an authorized peer:

Authentication through Signatures

The two peers use their digital signatures for ISAKMP-handshaking. These digital signatures can be obtained from a trusted source and provide a high level of security. Note that these signatures are never transmitted in plain text but in hashed format.

Authentication through Pre-Shared Key

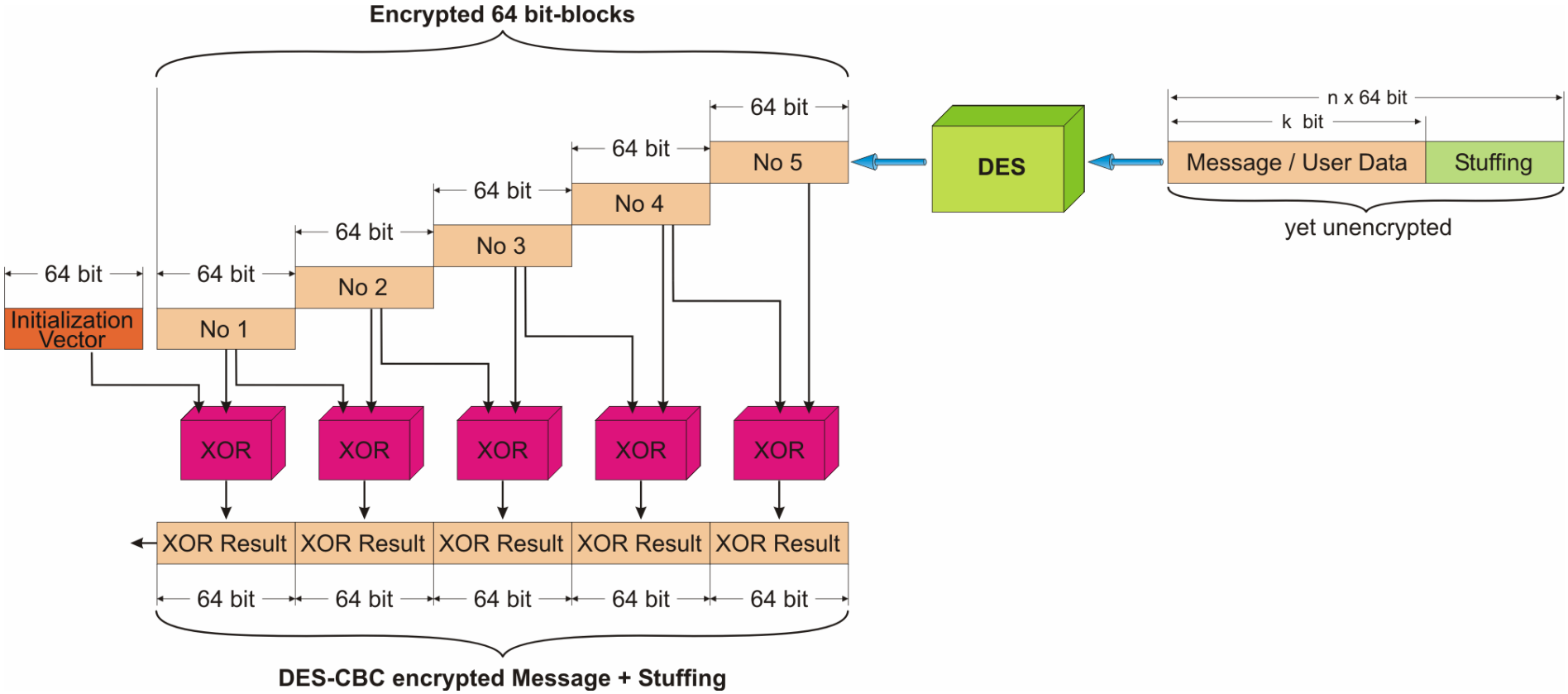
The two peers may -offline- exchange key values to identify themselves to each other. Like for the authentication through signatures, the pre-shared key is not transmitted in plain text but in hashed format.

Authentication through Public Key Encryption

Last but not least, public key encryption (as for PGP / Pretty Good Privacy) can be used for the initial handshaking process.

[RFC 2409]

DES-Operation in CBC-Mode (Cipher Block Chaining)



DES-Operation in CBC-Mode (Cipher Block Chaining)

- ⇒ In CBC-mode, the consecutive 64 bit-outputs of the DES-encoder are XORed with the previous 64 bit-output before they are finally concatenated into the encrypted MAC-PDU.
- ⇒ To be more precise, the first 64 bit-output of the DES-encoder uses the 64 bit of the Initialization Vector for XORing while the next uses the first 64 bit-output (and so on).

[FIPS 46, FIPS 81; FIPS = Federal Information Processing Standard]