

# Ingeniería Avanzada de Servicios sobre redes de Telecomunicación

Tomás Robles Valladares

[trobles@dit.upm.es](mailto:trobles@dit.upm.es)

Dpto. de Ingeniería de Sistemas Telemáticos

# INDICE

- ◆ Introducción
  - ▶ Servicios
  - ▶ Gestión de Servicios
  
- ◆ Arquitecturas Relacionadas
  - ▶ TINA
  - ▶ VHE
  - ▶ Web Services
  
- ◆ Arquitectura del 3GPP
  - ▶ Arquitectura General
  - ▶ OSA/Palay
  - ▶ IMS (IP Multimedia Subsystem)
    - SIP
    - QoS en UMTS

# INDICE

## ◆ Introducción

- ▶ Servicios
- ▶ Gestión de Servicios

## ◆ Antecedentes históricos

- ▶ TINA

## ◆ Arquitectura del 3GPP

- ▶ Arquitectura General
- ▶ OSA/Palay
- ▶ IMS (IP Multimedia Subsystem)

# El Mercado de las Telecomunicaciones

- ◆ Movimiento general hacia un mercado de telecomunicaciones liberado
- ◆ Los monopolios estatales se mueven hacia un mercado internacional abierto y competitivo
- ◆ Diversificación de participantes en el mercado
- ◆ Mayor interacción entre proveedores y clientes en la cadena de valor

# ¿Qué es un servicio?

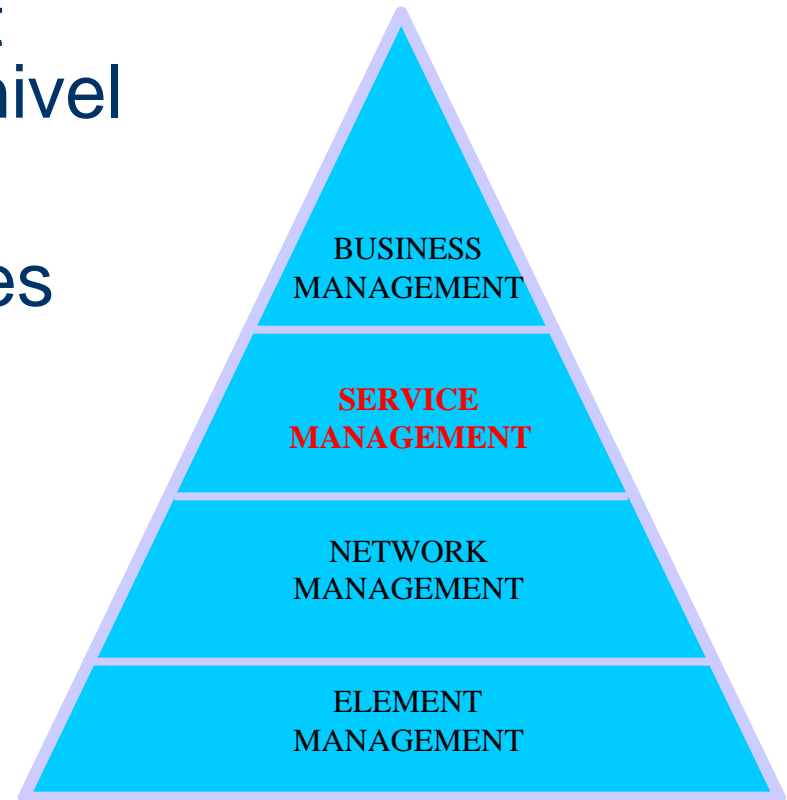
- ◆ Definición General: Un conjunto de cosas o funciones que aportan valor ofertadas por un proveedor de servicios a un cliente
- ◆ Servicio de Telecomunicaciones: un conjunto de capacidad empaquetado que se utiliza por un usuario final cuando este interacciona con un proveedor de servicios
- ◆ Tres tipos de servicios:
  - ▶ **Servicios de Telecomunicaciones:** responsables del establecimiento de las conexiones y procesar la información relacionada con las conexiones.
  - ▶ **Servicios de Gestión:** responsables de la gestión de los recursos.
  - ▶ **Servicios de Información:** servicios que son capaces de gestionar recursos de información como una película, sonidos o documentos.

# ¿Qué es un servicio? (II)

- ◆ Un servicio de valor añadido (o simplemente un servicio) es cualquier cosa más avanzado que una llama de voz entre dos participantes.
  - ▶ Servicios de telefonía (interacciones con control de llamadas)
    - Gestión de llamadas, facturas separadas,
    - Multiconferencias, juegos multijugador, ...
  - ▶ Servicios no telefónicos (no interacciona con el control de llamadas)
    - Acceso a web, e.g. acceso a cotizaciones o navegación desde un teléfono móvil
  - ▶ Combinados
    - E.G. Centros de Llamadas (call centers)

# Gestión de servicios (según TMN)

- ◆ El modelo de TMN (Telecom Management Network ) identifica el nivel de gestión de servicios
- ◆ Soporta las necesidades de gestión del negocio
- ◆ Impone requisitos en los niveles de gestión de la red



# Propósitos de la Gestión de Servicios

- ◆ Principalmente para:
  - ▶ Reducir el coste para los clientes
  - ▶ Proporcionar mejores servicios a los clientes
  - ▶ Reducir el tiempo de puesta en el mercado de nuevos servicios
- ◆ Logrado mediante sistemas de T.I. que proporcionan automatización extremos a extremo del proceso de gestión de los servicios
- ◆ Tecnologías de objetos distribuidos como CORBA son fundamentales para esta gestión



# Aspectos de la gestión de servicios

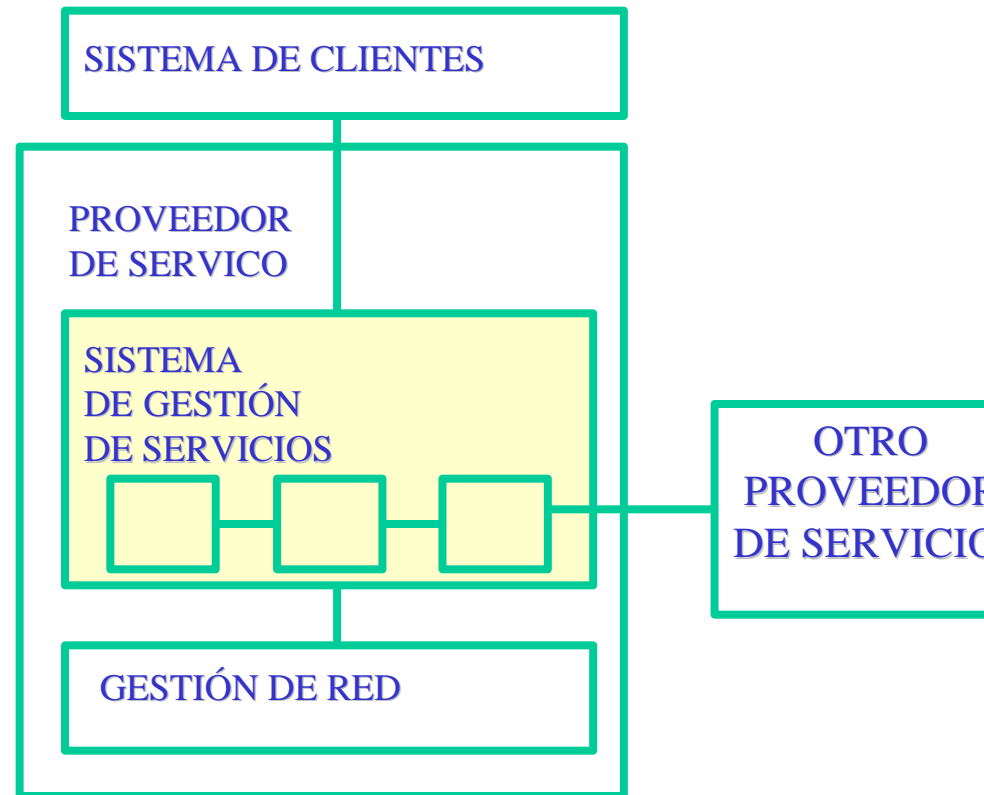
- ◆ Gestión de la Configuración
  - ▶ Ciclo de vida de los servicios
  - ▶ Ciclo de vida de los clientes
- ◆ Gestión de suscripciones
- ◆ Gestión de Accounting
- ◆ Gestión y personalización de perfiles
- ◆ Gestión de fallos
- ◆ Gestión de eficiencia de los servicios

# Ingeniería de Servicios

- ◆ Trata sobre los temas relacionados con la ingeniería de servicios avanzados
- ◆ Trata de los temas relacionados con el soporte de todas las fases del ciclo de vida
- ◆ Incluye aspectos de arquitectura
  - ▶ Conceptos, principios, roles
  - ▶ Entidades funcionales, interfaces y algoritmos
- ◆ ¿Por que es una disciplina importante?
  - ▶ Negocios: los servicios de valor añadido son necesarios para atraer suscriptores y generar beneficios
  - ▶ Ingeniería: trivial
    - Ejemplo: creación de servicios -> se necesita un acceso seguro y selectivo a los recursos de red

# La necesidad de los acuerdos industriales

- ◆ Áreas donde los acuerdos industriales son útiles:
  - ▶ Interfaz con el consumidor
  - ▶ Interfaz con otros proveedores de servicios
  - ▶ Interfaz con la red y elementos de la red
  - ▶ Interfaz entre procesos de gestión de servicios internos



# INDICE

- ◆ Introducción
  - ▶ Servicios
  - ▶ Gestión de Servicios
  
- ◆ Arquitecturas Relacionadas
  - ▶ TINA
  - ▶ VHE
  - ▶ Web Services
  
- ◆ Arquitectura del 3GPP
  - ▶ Arquitectura General
  - ▶ OSA/Palay
  - ▶ IMS (IP Multimedia Subsystem)

# TINA

# El consorcio TINA (TINA-C)

- ◆ Creado en 1992 por los operadores de telecomunicación, suministradores y vendedores de equipos
- ◆ El núcleo del equipo (~30 investigadores) se localizó en Bellcore, U.S.- siguiendo un modelo diferente de estandarización que ITU-T y el IETF
- ◆ Trabajo durante 5 años (1993-1997) del núcleo del equipo, y el trabajo continuó durante otros 3 años (1998-2000) mediante “grupos de trabajo”.
- ◆ Varios proyectos auxiliares de participación mundial ha completado, mejorado y validado la arquitectura y especificaciones de TINA mediante el diseño, implementación y demostración

# Servicios TINA

- ◆ Servicios basado en voz tradicionales
- ◆ Servicios multimedia, y multiconferencia
- ◆ Computer Supported Collaborative work (CSCW)
- ◆ Servicios de Operación y Mantenimiento
- ◆ Servicios de información y Vídeo bajo Demanda
- ◆ Soporte de Calidad de Servicios
- ◆ Soporte de Movilidad (personal, terminal, sesión)

# TINA= IN+TMN+ODP

Un intento de unir las telecomunicaciones y la IT

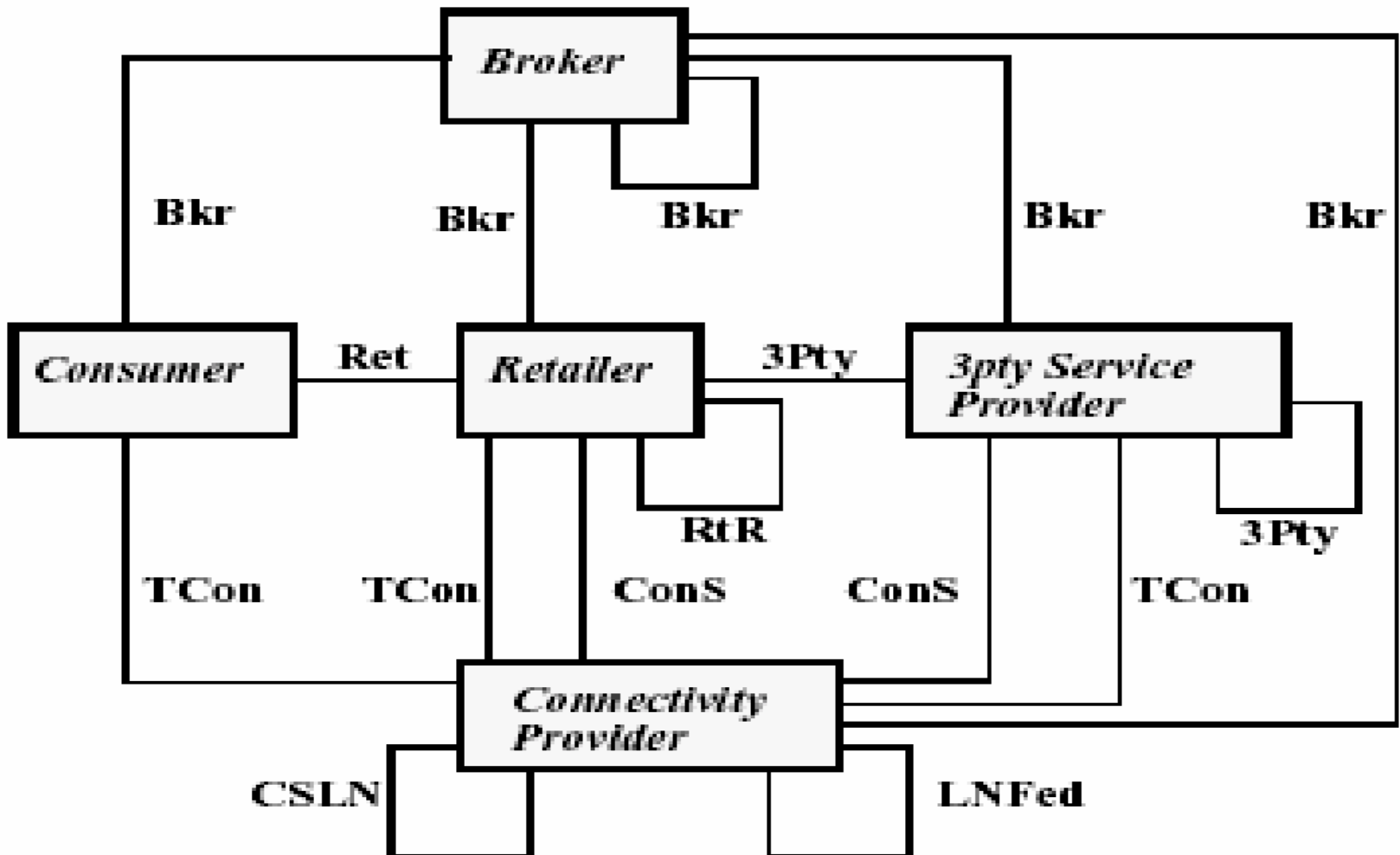
## ◆ Pilares de TINA:

- ▶ Tecnologías software avanzadas
  - Orientación a Objetos
  - Computación Distribuida
- ▶ Integración de normas de telecomunicación
  - IN
  - TMN
- ▶ Metodología ODP (Open Distributed Processing)
  - Negocios
  - Información
  - Computacional

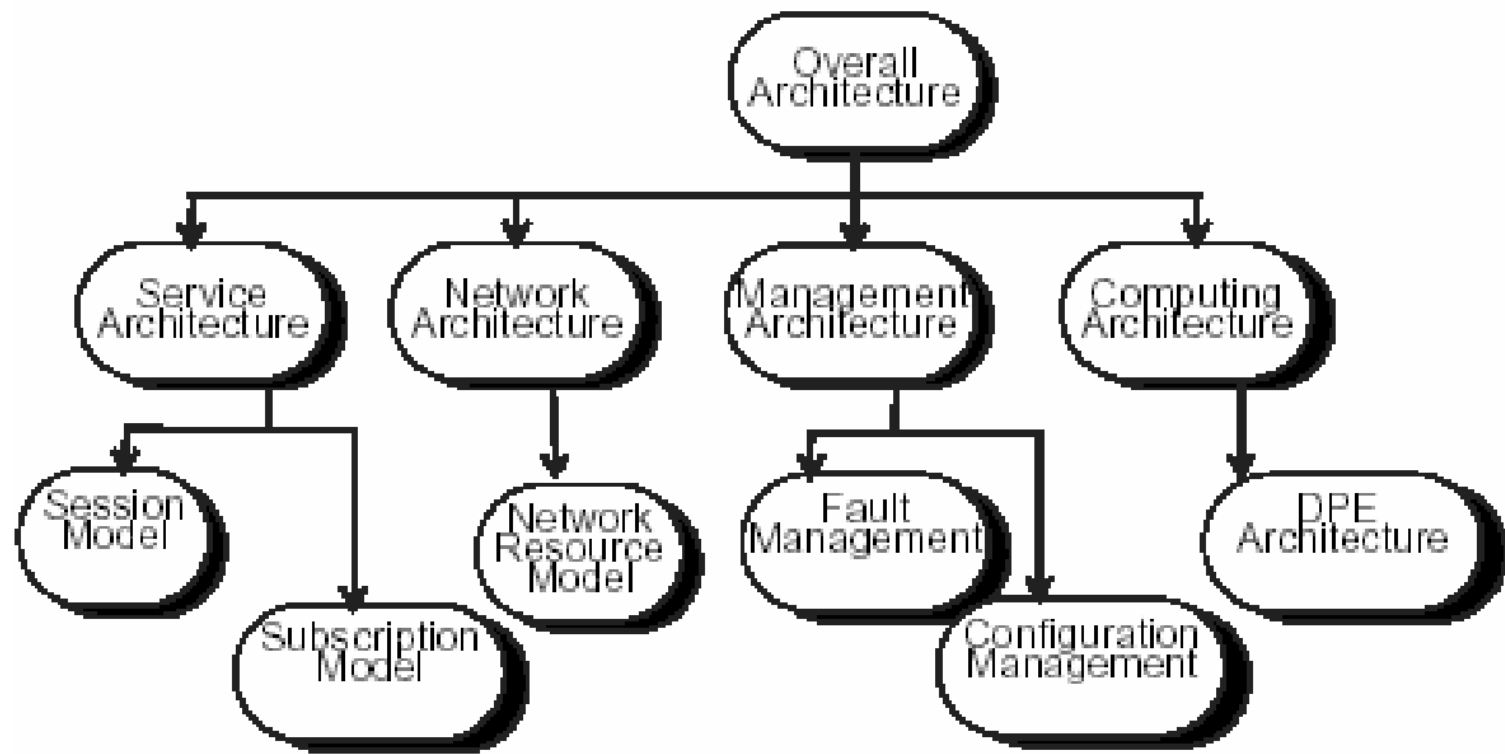
*TINA is a form of network intelligence that tries to integrate and take further ideas from IN and TMN, based on ODP distributed software processing principles -> uniform control and management plane with intelligence outside switching system*



# Modelo de Negocio de TINA



# Descomposición de la Arquitectura de TINA



Key:



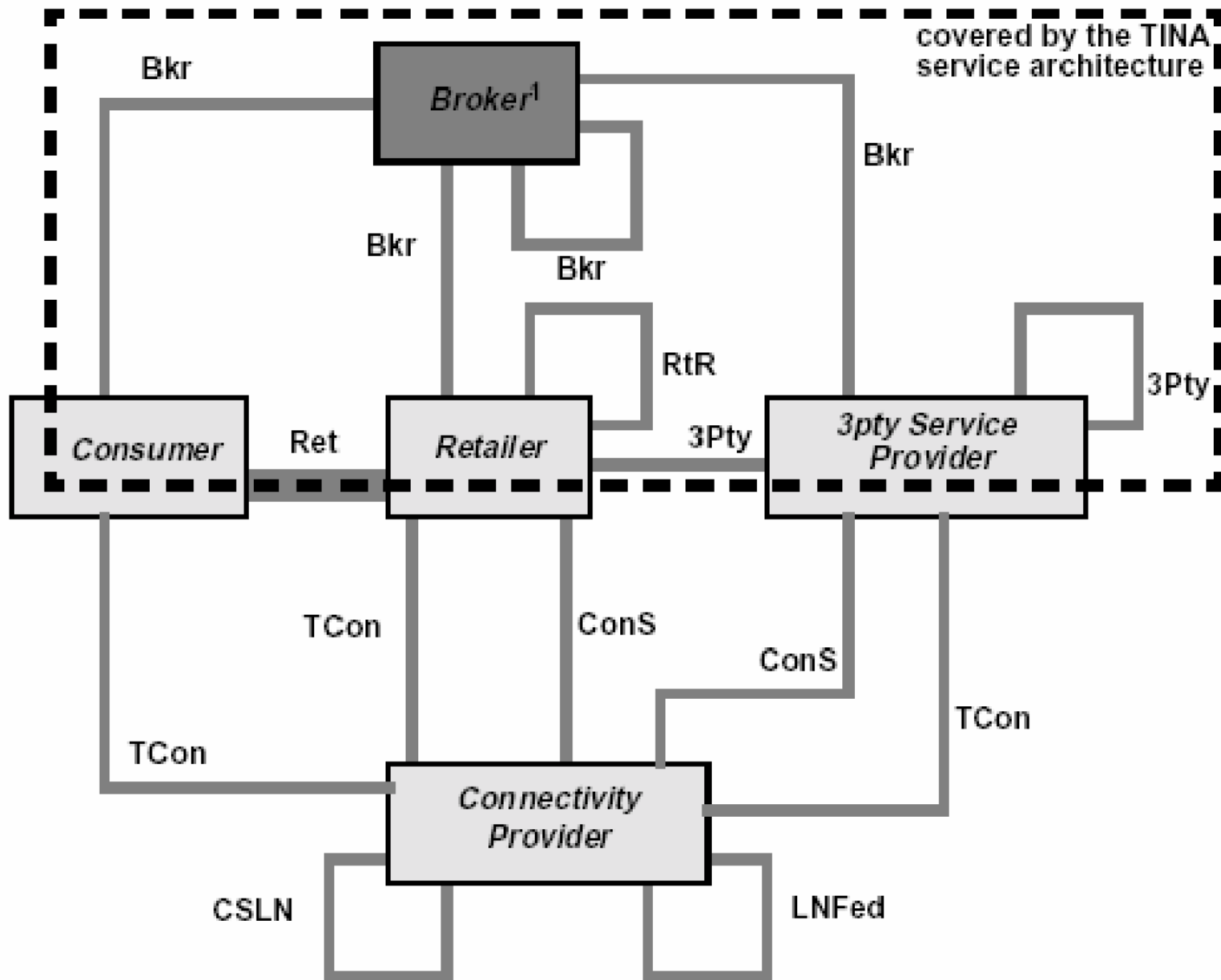
○ Subset of concepts and principles.

→ Concepts and principles are applied consistently.

# Arquitectura de Servicios de TINA

- ◆ Soportar una gran variedad de servicios
- ◆ Rápido desarrollo y provisión de servicios
- ◆ Adecuado a los servicios
- ◆ Evolución independiente de los servicios y la infraestructura de red
- ◆ Soporte par aun entorno abierto a múltiples jugadores
- ◆ Gestión de servicios
- ◆ Acceso universal a servicios

# Arquitectura de Servicios



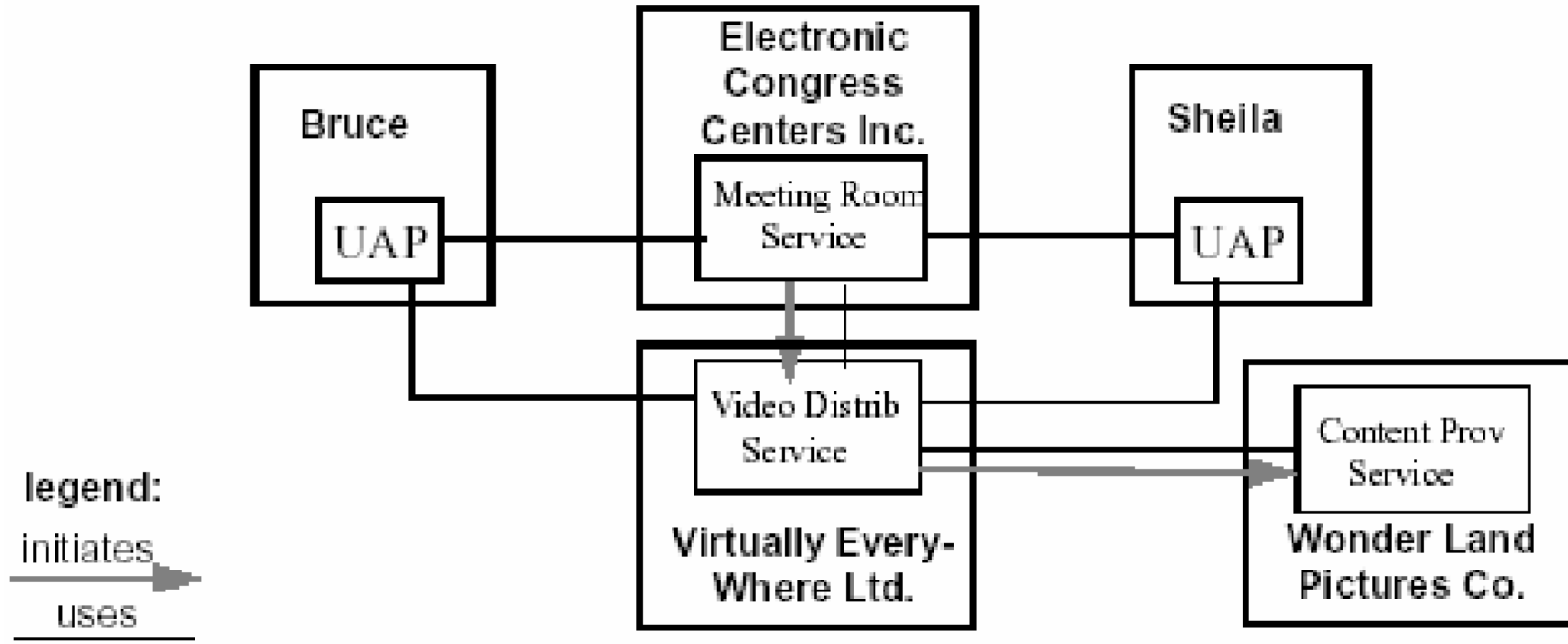
# El concepto de Sesión

- ◆ Sesión: el contexto para realizar actividades en un servicio, mediante una colección de objetos relacionados
- ◆ Proporciona separación de elementos en el tiempo, servicios recursos y roles. Se asocia con la reserva de recursos para ejecutar los servicios

# Composición de servicios

- ◆ Composición de servicios se refiera a la creación de nuevos servicios o instancias de servicios mediante la composición de servicios o servicios componentes
- ◆ El nivel de composición de servicios incluye la composición de sesiones de servicio, recursos de servicio o servicios componentes
- ◆ La composición de servicios en tiempo de ejecución incluye:
  - ▶ Composición estática o predefinida entre componentes del nivel de servicio
  - ▶ Composición dinámica entre componentes del nivel de servicio

# Ejemplo



# Conclusiones sobre TINA

- ◆ Arquitectura que plantó una Semilla inicial:
  - ▶ Muchos conceptos relacionados (e.g. ciclo de vida de servicios, sesión, composición de servicios, ...)
  - ▶ Muchos principios importantes (e.g. separación entre acceso & uso)
  - ▶ Model de negocios ampliamente reutilizado
  
- ◆ Pero un fracaso comercial
  - ▶ Varios prototipos y pruebas
  - ▶ Muy pocos despliegues comerciales debido a una gran diversidad de factores:
    - Complejidad
    - Demasiado ambicioso (abarca “todo” desde la red a la ingeniería de servicios)
    - Muy poca inclinación hacia IP
    - Muy poca consideración de los sistemas ya desplegados



VHE

# El concepto de VHE (Virtual Home Environment)

<b>3GPP</b>	“The VHE ensures that users are consistently presented with the same personalised User Interface features, User Interface customisation and services in whatever network and whatever terminal (within the capabilities of the terminal and network), where ever the user may be located”
<b>GSM</b>	“Virtual Home Environment (VHE) is a system concept for service portability in the Third Generation across network borders”
<b>ITU</b>	“VHE is a capability whereby a User is offered the same service experience in a visited network as in his Home system”
<b>UMTS</b>	“VHE means that the user will have the same interface and service environment regardless of location (personalised user interface independent from the current serving network).”
<b>Eurescom</b>	“The Virtual Home Environment is an environment, which presents the user with a common look and feel interface and service experience regardless of location, network and terminal type. The VHE is based on standardised service capabilities and personalised features that are consistently presented so that the user always "feels" that he is on his home network even when roaming across "feels" that he is on his home network even when roaming across network boundaries”
<b>VESPER</b>	“VHE main feature is that the customised environment will be following the user while he/she is roaming within different networks and using different terminals”

# Introducción a VHE

- ◆ Definido como: **“System Concept for Personalised Service Portability across Network Boundaries and Terminals”**
- ◆ ¿Qué significa? **“You can set up and take your familiar services and MMI with you on any supporting network and terminal, even when roaming.”**

# Orígenes

- ◆ Concepto que aparece en un brainstorming del GSM MoU en Enero de 1995
- ◆ Evolución de diferentes sistemas de la segunda generación a un único estándar mundial para el IMT-2000
- ◆ Permiten a usuarios y proveedores de servicios crear y ejecutar servicios de forma consistente y “look-and-feel” incluso en situaciones de romain en redes diferentes
- ◆ Permitir a los operadores y proveedores de Servicios diferenciar servicios
- ◆ Aproximarse a la flexibilidad de los servicios en Internet

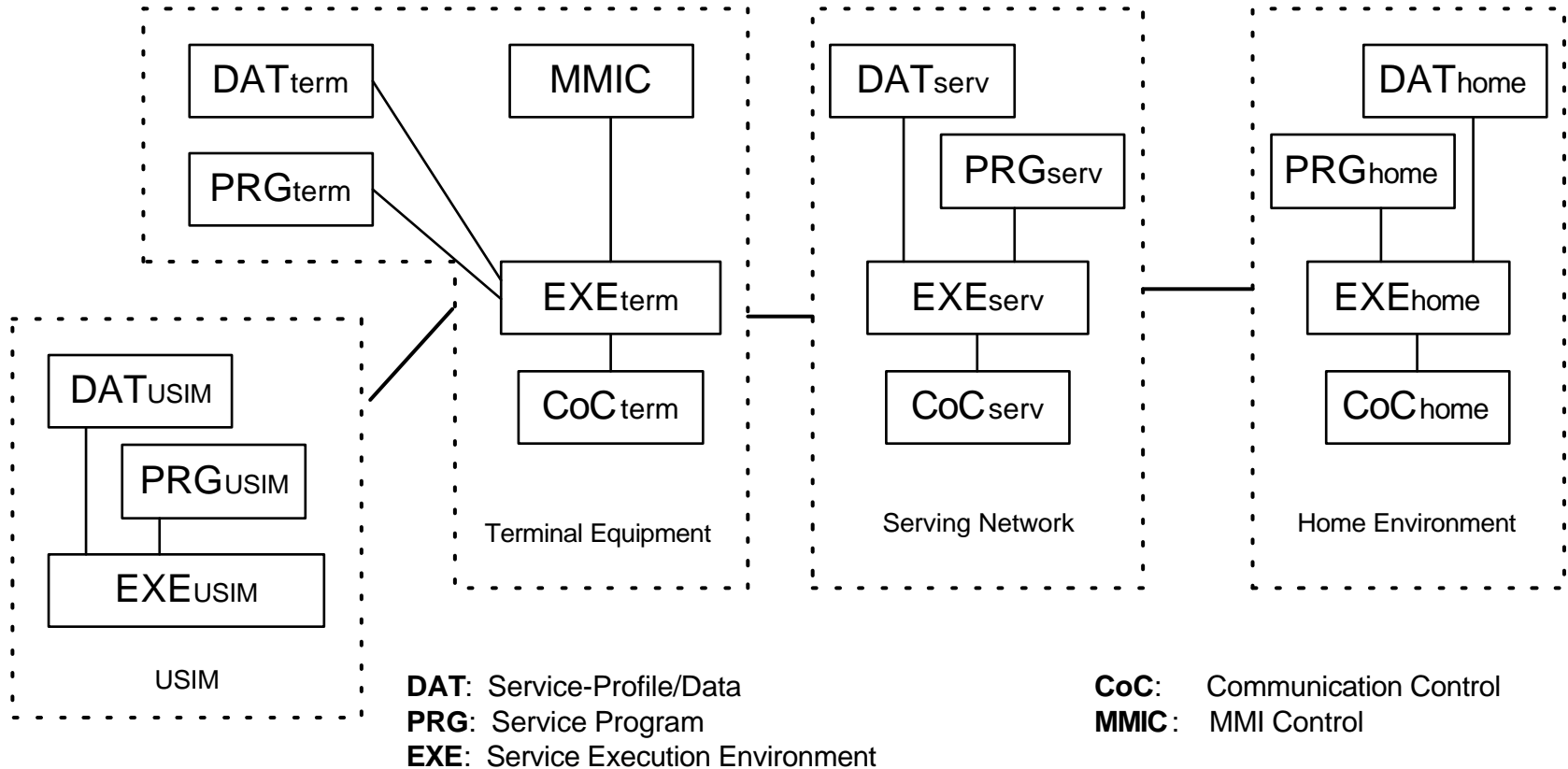
# Punto de vista del Usuario

- ◆ Los usuarios desean interfaces familiares (MMI)
- ◆ Desplegar sus propios servicios y perfiles
  - ▶ Como se hace en un PC
- ◆ Incluirá:
  - ▶ Servicios ya existentes de la propia red (home network)
  - ▶ Otros importados de otros sistemas como Internet
  - ▶ Creación de nuevos servicios
- ◆ Debe funcionar en situaciones de roaming en otras redes

# Requisitos

- ◆ Perfil de usuario pertenece al propio usuario, y debe estar almacenado en el USIM y/o la BD de la red Propia
- ◆ Estos datos deben controlar/ejecutar software en Estaciones Móviles/Terminales, Redes de Servicios o nodos remotos (e.g. Servidores de CAMEL)
- ◆ Este software puede existir con anterioridad (e.g. servicios normalizados) o puede ser creado y/o entregado a una localización específica
- ◆ Puede ser necesario que los programas y/o los datos estén presentes en más de un nodo
- ◆ Si se invoca el servicio por el usuario o el USIM y no está disponible, debe haber mecanismos para localizarlo y descargarlo
- ◆ Las normas deben definir interfaces y APIS
- ◆ Debe degradarse controladamente si no está completo

# Arquitectura Propuesta



# Componentes Funcionales

- ◆ Programas del servicio
- ◆ Entorno de ejecución de servicios
- ◆ Datos del Perfil del Servicio
- ◆ Control de las Comunicaciones
- ◆ Control MMI



# Web Services

# INTRODUCCIÓN

- ◆ Los Web Services son una arquitectura de computación distribuida en evolución que usan sus propias interfaces programa-programa, protocolos y servicios de registro de tal manera que posibilitan que aplicaciones de diferentes plataformas tecnológicas puedan utilizar “servicios” de otras aplicaciones.
- ◆ Un Web Service se aprovecha de la especificación de XML para definir tanto su descripción, como los mensajes que recibe y produce al igual que en los servicios de registro del servicio.

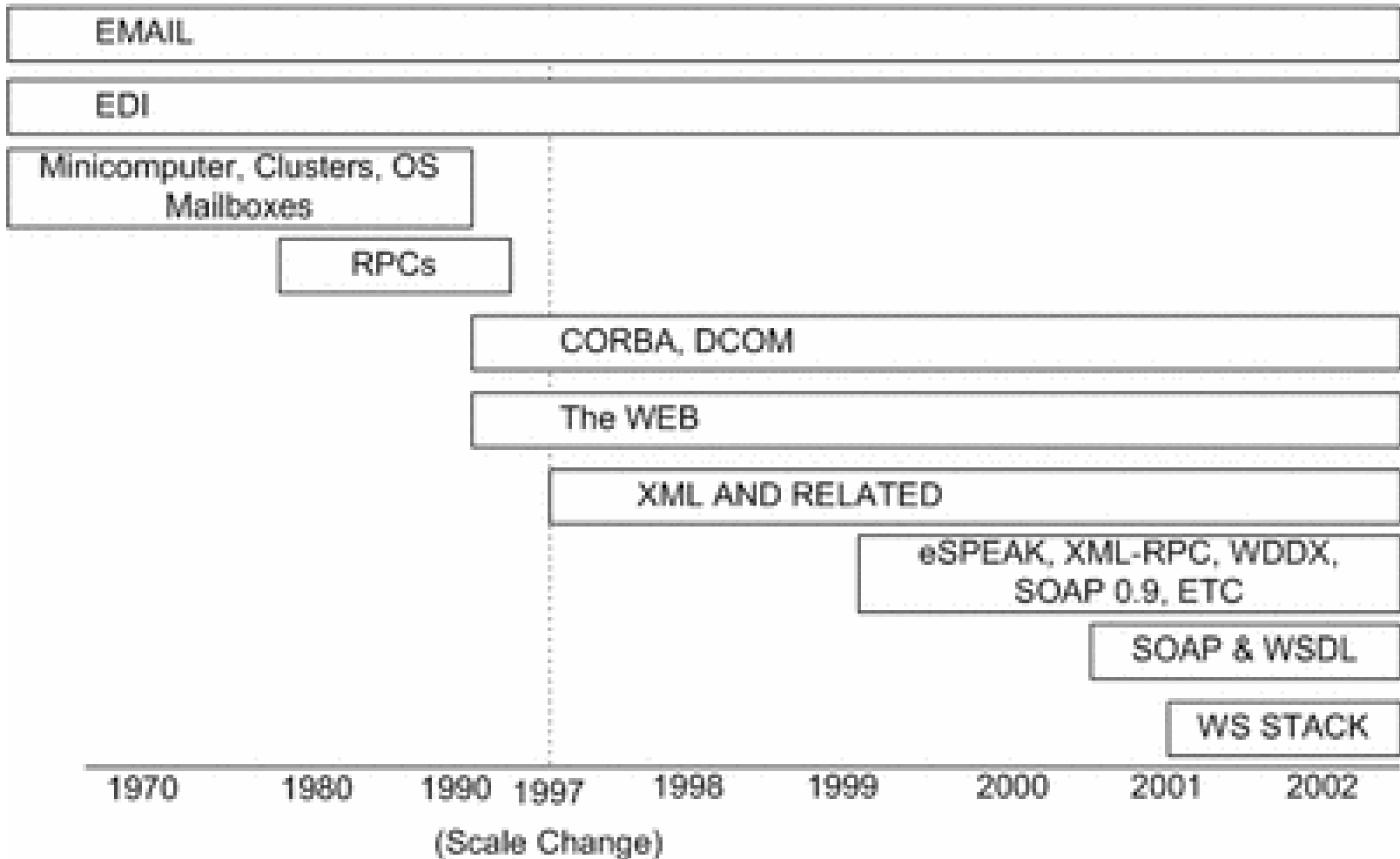
# Antecedentes

- ◆ La idea general alrededor de Web Services no es algo nuevo.
- ◆ Antecesores en el tema de computación distribuida han existido y funcionado:  
RPC, EDI, CORBA, COM, APPC
- ◆ La diferencia se encuentra en el alto nivel de encapsulación e independencia entre las aplicaciones.
- ◆ El gestor de la iniciativa es W3C, garante de evolución y futuro.
- ◆ “Programa a usuario” -> Semantic Web

# Antecedentes (II)

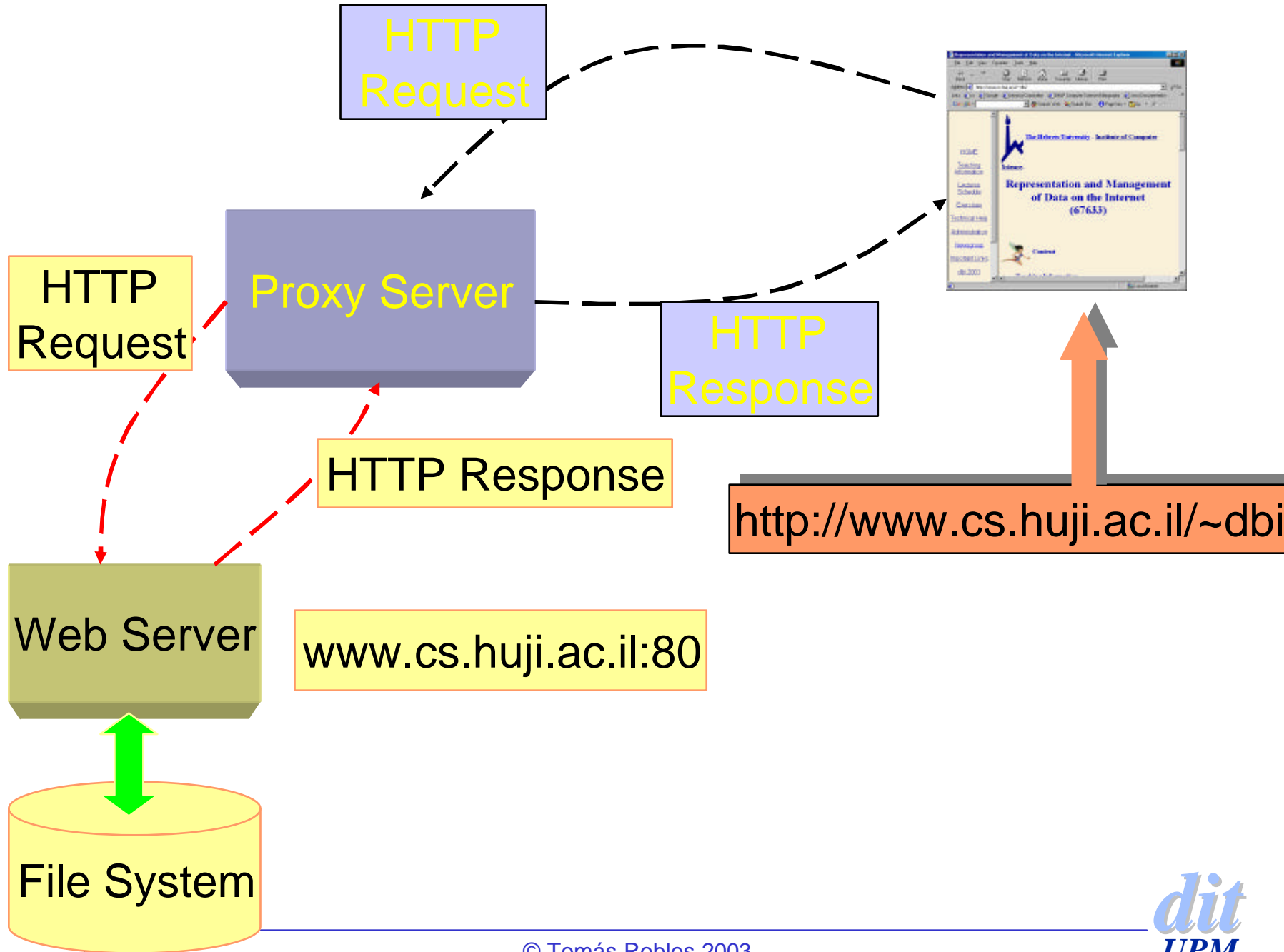
- ◆ Evolución tecnológica:
  - Código libre
  - Programación estructurada
  - Programación orientada a objetos
  - Objetos distribuidos - Componentes
- ◆ Evolución de integración de aplicaciones:
  - Computación por departamentos
  - Interfaces punto a punto (“spaguetti code“)
  - Sistemas ERP
  - Soluciones de “middleware”
- ◆ Necesidad de establecer conexiones tanto entre aplicaciones internas como B2B.

# HISTORY OF DISTRIBUTED COMPUTING



# World Wide Web

- ◆ Diseñado originalmente para acceder a recursos compartidos
- ◆ Protocolo de comunicación HTTP
  - ▶ Protocolo sin estado (stateless)
  - ▶ Browser – Servidor Web: GET recursos
  - ▶ Servidor Web – Browser: recurso estático (html, imágenes, ...)
- ◆ Formateo de datos
  - ▶ HTML, sintaxis y semántica de visualización
- ◆ Diferentes tipos de recursos
  - ▶ MIME (texto, html, imágenes, binarios, videos, etc)
- ◆ Navegación y localización NO lineal
  - ▶ Hipertexto
  - ▶ Hipermedia

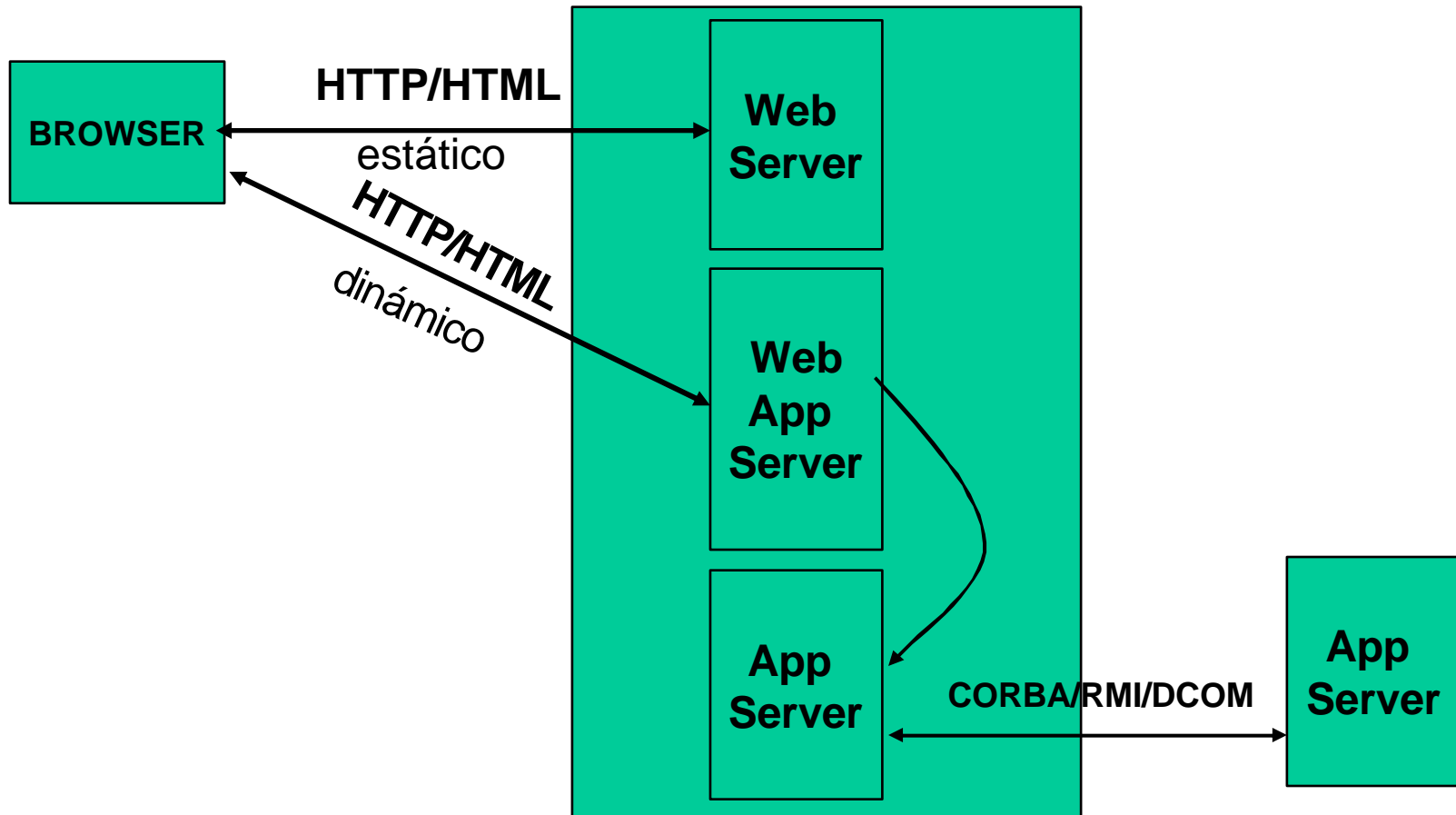


# Web como plataforma para Aplicaciones Distribuidas

- ◆ Aplicaciones Web (Web Applications)
- ◆ HTTP/HTML fue diseñado originalmente para acceder a recursos compartidos, NO para ejecución de aplicaciones remotas
  - ▶ Recursos estáticos
- ◆ ¿Porqué no permitir solicitar datos al usuario y ejecutar aplicaciones vía web?
  - ▶ Como capturar entrada del usuario?
    - FORMAS: `<FORM ...> <INPUT...> ... </INPUT> ... </FORM>`
  - ▶ El recurso es el nombre de una aplicación



# Integración de Aplicaciones Web con Objetos Distribuidos



# ¿Qué son los Web Services?

- ◆ Semánticamente encapsula funcionalidades discretas.
- ◆ Sistemas Débilmente-Acoplados, Componentes Reutilizables
- ◆ Accesibles programáticamente
- ◆ Distribuido Sobre Internet
- ◆ Dado que todas las comunicaciones se realizan en XML los Web Services no dependen de sistemas operativos específicos
- ◆ De acuerdo a esto, Java puede hablar con Perl, Windows con Unix.

# Cont.

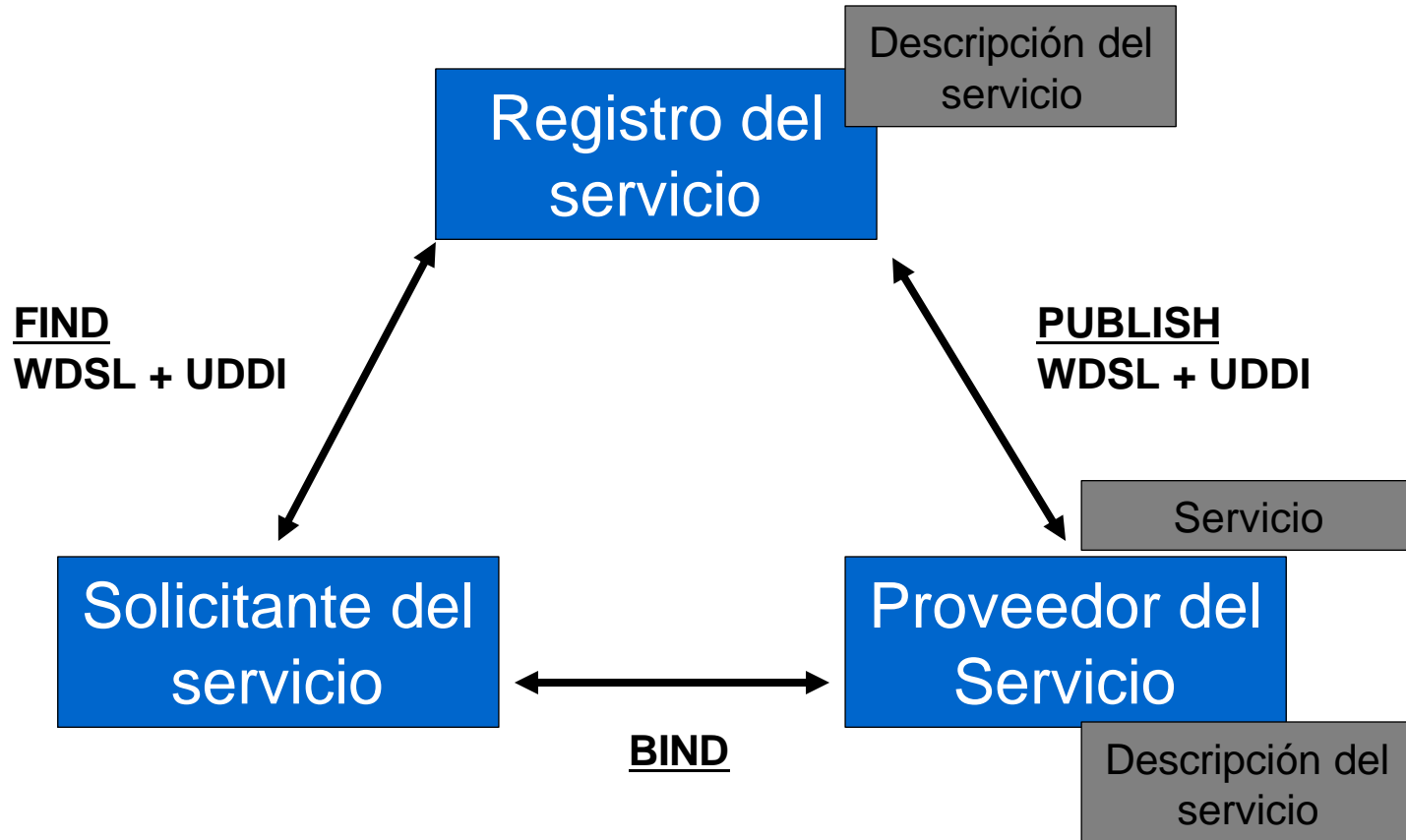
◆ De la definición anterior se desprende que los Web Services tienen dos propiedades adicionales:

1. Deben tener una interfaz pública definida en una gramática común en XML, la interfase describe todos los **METODOS** disponibles a los clientes y especifica la **firma** para cada método

La definición de esta interfaz se hace con el lenguaje **WSDL (Web Service Description Language)**

2. Si usted crea un servicio Web, debe tener una forma de publicarlo, debe existir una forma de localizar el servicio y localizar su interfase pública, esto se hace con UDDI (Universal Description, Discovery, and Integration)

# Actores, Objetos y Operaciones



# Aplicaciones de los Web Services

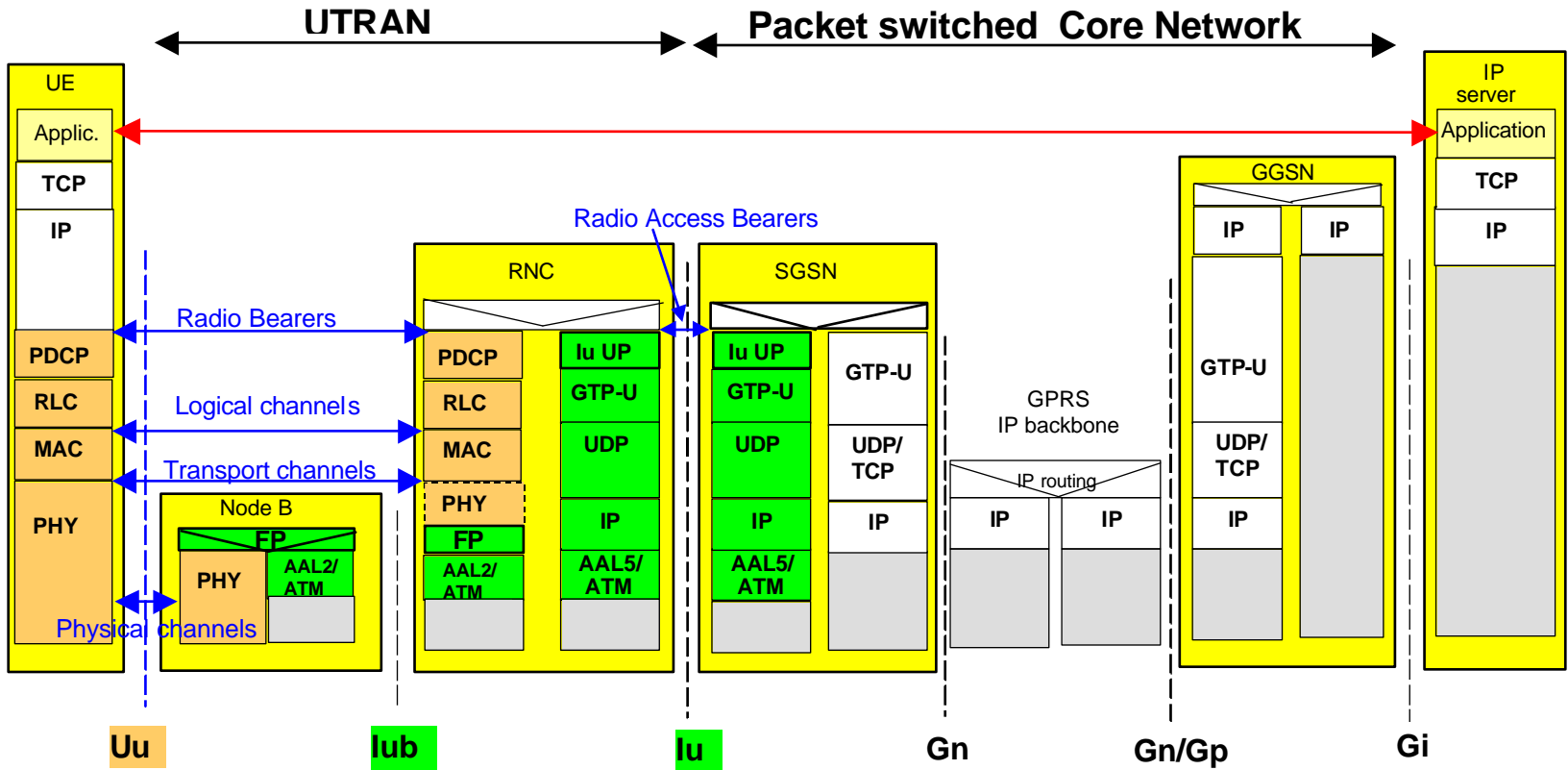
- ◆ Interoperabilidad entre ambientes de desarrollo
  - ▶ Tradicionalmente, antes de los Web Services una empresa debía decidir en que ambiente de desarrollo trabajar
    - “Me voy por JAVA”
    - “Me voy por Microsoft”
    - “Me voy por Oracle”
  - ▶ Una vez se decidía el Ambiente de Desarrollo, queda uno “Cazado” con dicha plataforma.
  - ▶ Con Web Services, desarrolle sus aplicaciones en lo que más le guste e intégrarlas con Web Services.
  - ▶ Una aplicación que parecía Irreal:
    - Llamar desde una aplicación hecha en Java y servicio hecho en ambiente Microsoft.
    - Esto es posible gracias a los Web Services.

# INDICE

- ◆ Introducción
  - ▶ Servicios
  - ▶ Gestión de Servicios
  
- ◆ Arquitecturas Relacionadas
  - ▶ TINA
  - ▶ VHE
  - ▶ Web Services
  
- ◆ Arquitectura del 3GPP
  - ▶ Arquitectura General
  - ▶ OSA/Palay
  - ▶ IMS (IP Multimedia Subsystem)
    - SIP
    - QoS en UMTS

# 3GPP General Architecture

# UMTS

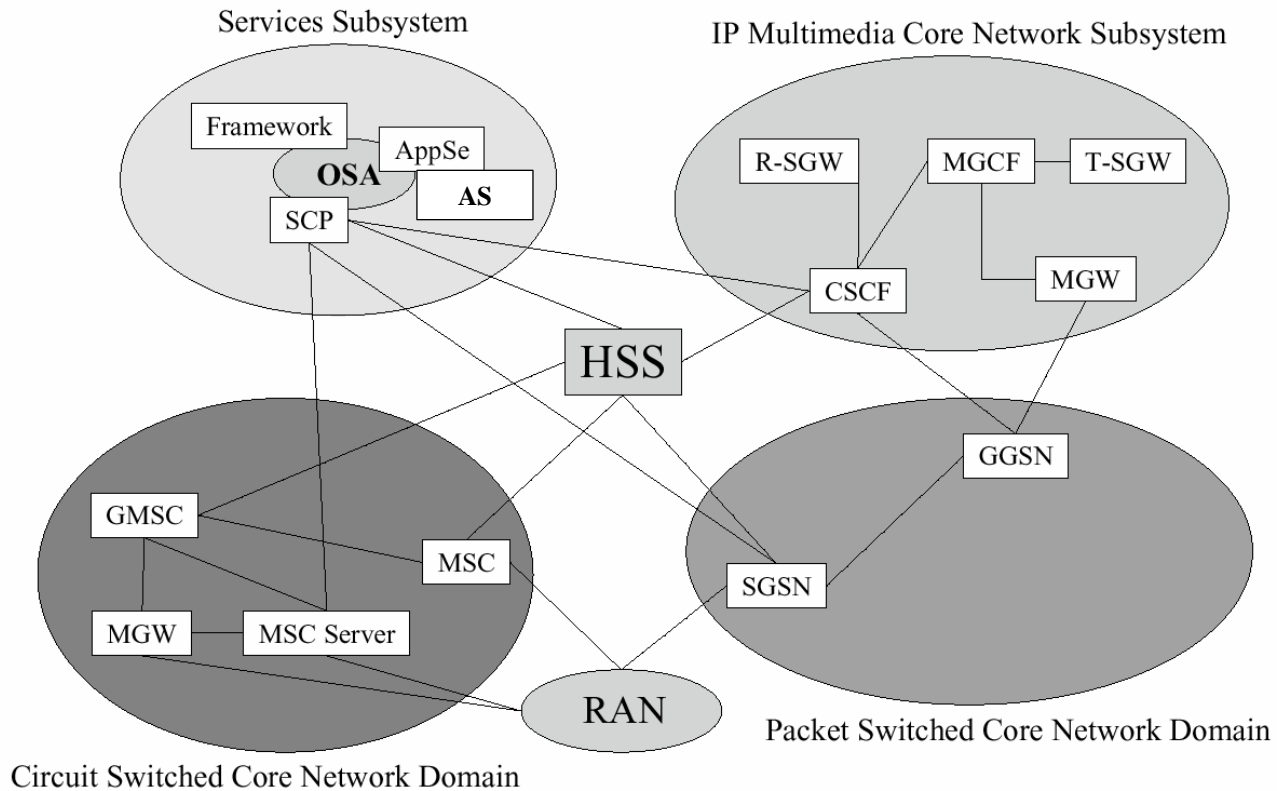
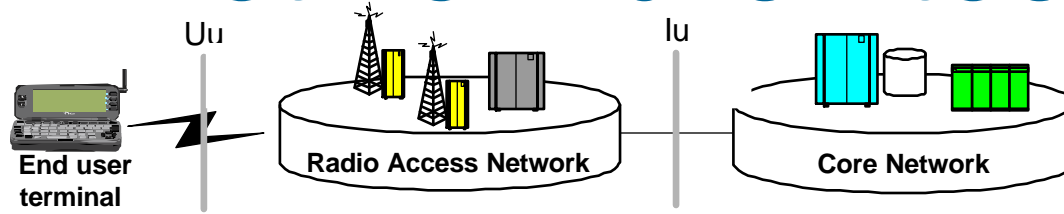


W. Granzow

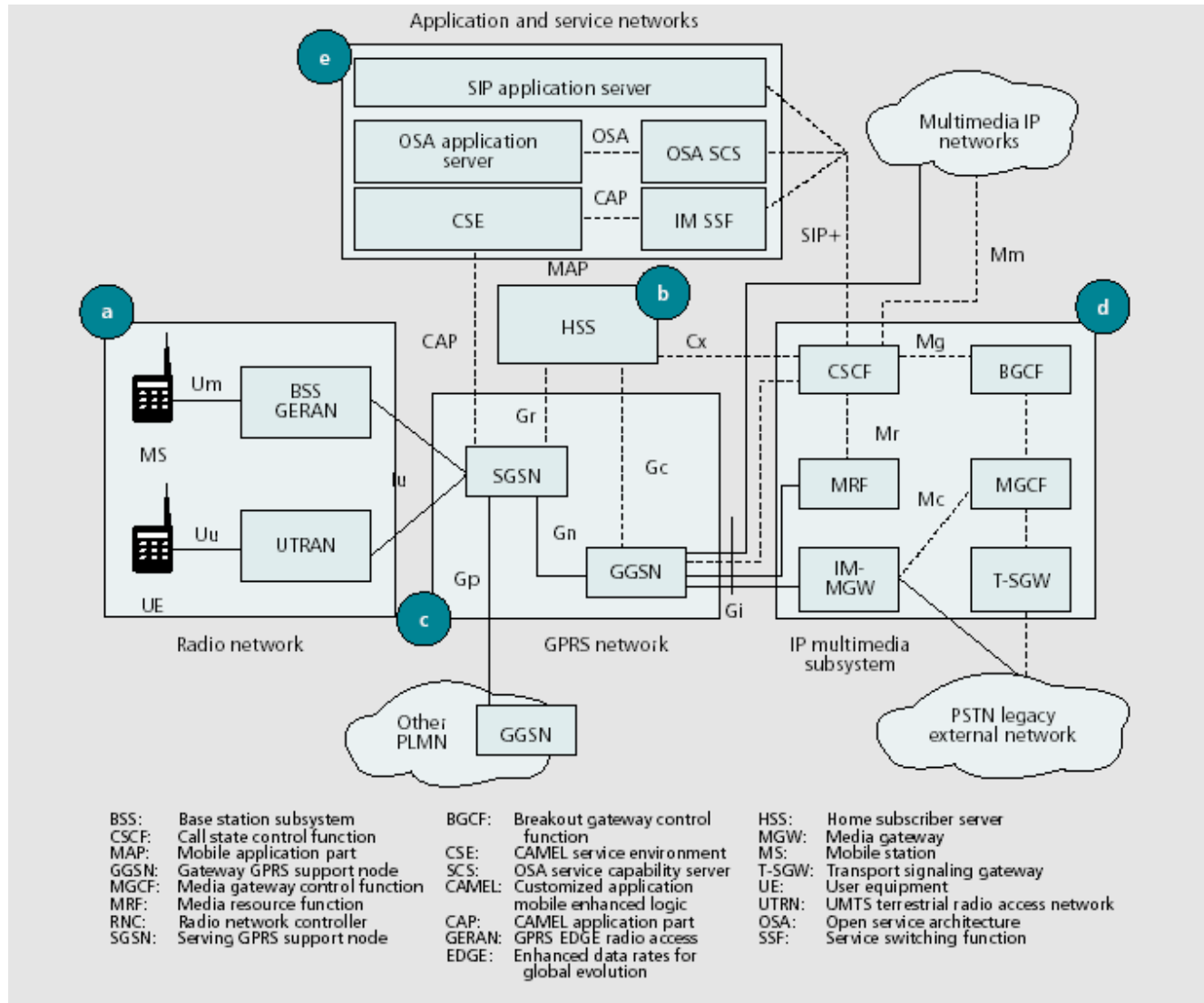




# 3GPP network architecture



# 3GPP network architecture (II)



# Arquitectura OSA/PARLAY

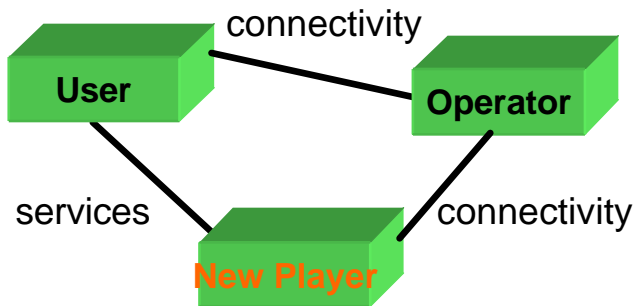
# OSA/Parlay

- ◆ Introduction: the Parlay/OSA API: why, where, what?
- ◆ A closer look at Parlay/OSA
  - ▶ Parlay/OSA Framework and Service Capability Features
  - ▶ Parlay/OSA current functionality
- ◆ The Parlay/OSA Framework
- ◆ Framework functionality

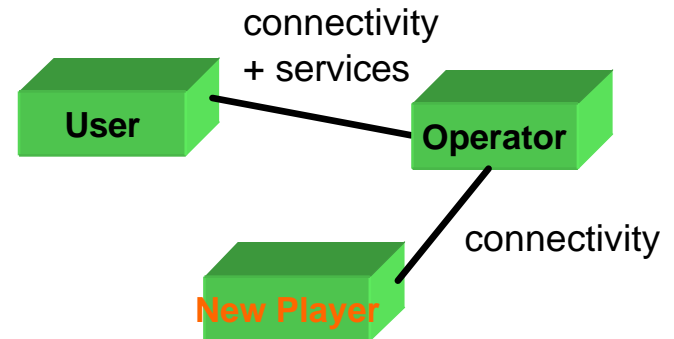
# Origins of Parlay/OSA API(1/2)

A change in business models has introduced new players in the telecom business

Some want to address users directly



Some prefer to do it via the Network Operator



But they have something in common:  
They compete in the services market...

and they have **no network!**

# Origins of Parlay/OSA API(2/2)

**This is a win-win situation!**

- It opens new sources of revenue for incumbent Network Operators
  - traffic in their networks increases
  - they may enrich their service offering
- It opens the telecom business to newcomers
  - VNOs, MVNOs, ASPs, ...

So we only need a technical enabler:



# Location of the Parlay/OSA API(1/2)

*Service network*

Services/application layer

Parlay / OSA

OSA/Parlay API's exposing network service capabilities

*Core network*

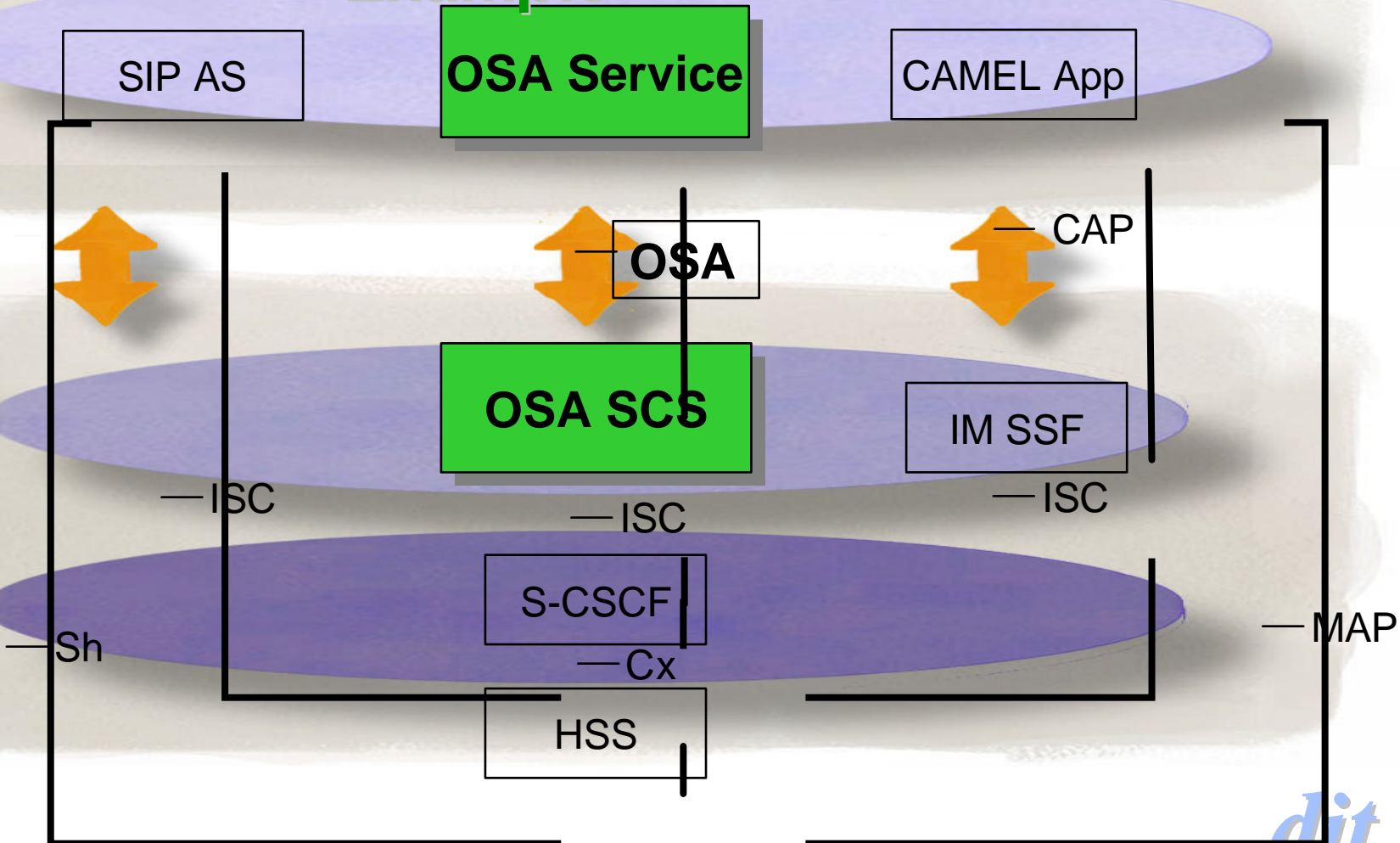
Control layer  
Service Capability Servers

Connectivity layer  
Core & Radio Networks  
2G 2.5G & 3G

Distribution via middleware

# Location of the Parlay/OSA API(2/2)

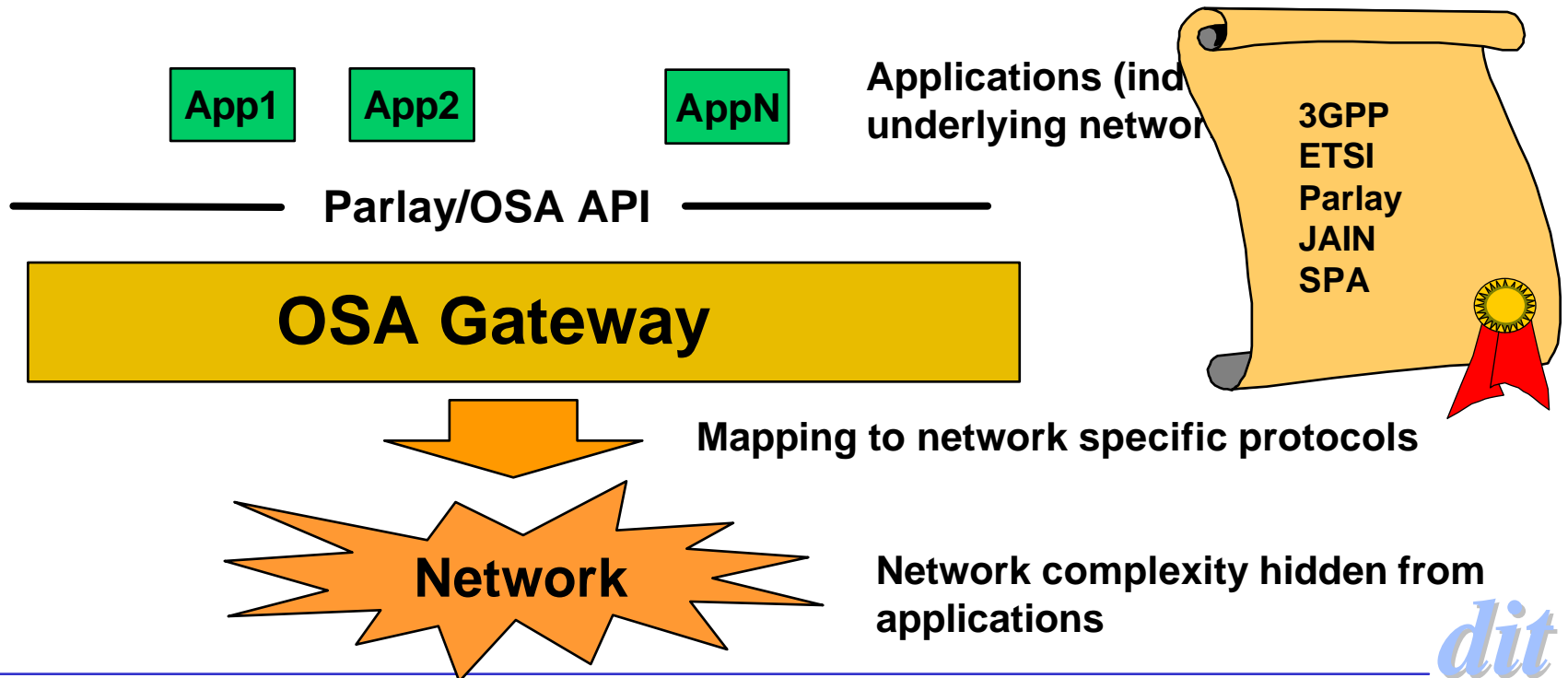
## The UMTS Example





# Overview (1/2)

**Parlay/OSA (Open Service Access)** is an API that enables operator and 3rd party applications to make use of network functionality through a set of open, standardised interfaces



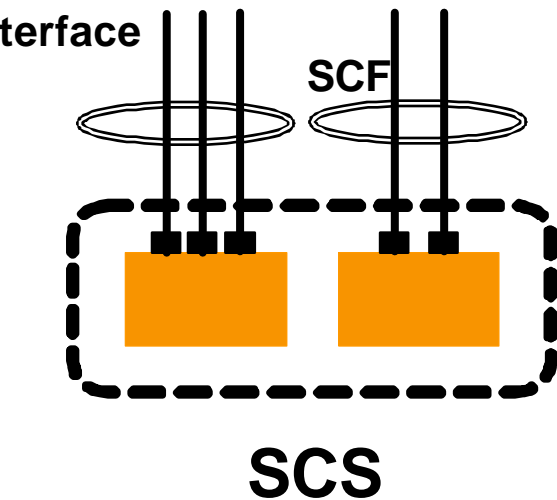
# Overview (2/2)

Opening up of network by means of standardized APIs based on open technology. This leads to :

- ▶ Shorter TTM for applications / services due to abstraction and open technology (developer community orders of magnitude larger than telco developer community)
- ▶ Applications can also be developed and deployed by 3rd parties (creative, new innovative services).
- ▶ Applications can be network independent (multi-access / multi-service)
- ▶ Applications can be combination of
  - different capabilities
  - enterprise data with network functionalityleading to new innovative services.

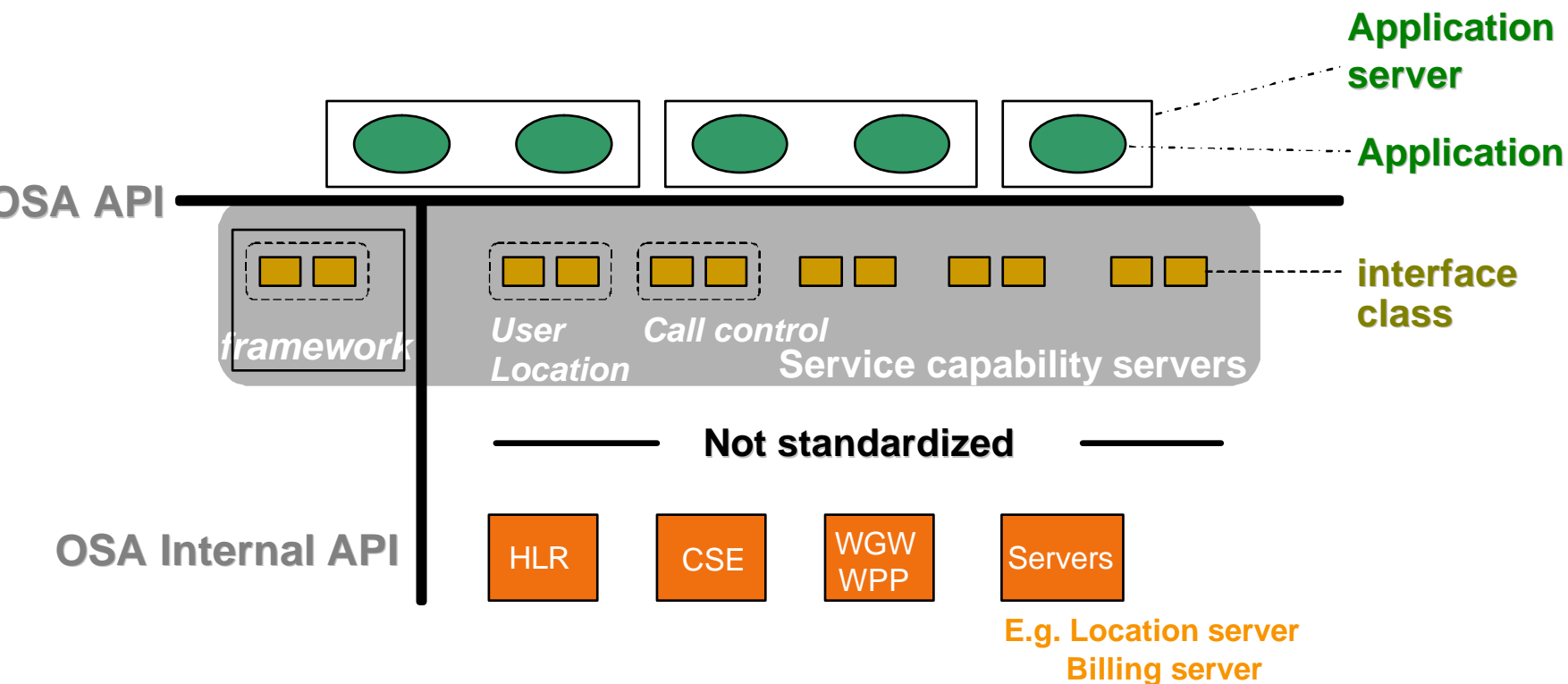
The objective is to combine telecom and data industries

# Terminology: SCSs and SCFs



- The Parlay/OSA Gateway consists of several **Service Capability Servers (SCS)**: functional entities that provide Parlay/OSA interfaces towards applications.
- Each SCS is seen by applications as one or more **Service Capability Features (SCF)**: abstractions of the functionality offered by the network, accessible via the Parlay/OSA API. Sometimes they are also called **services**
- The Parlay/OSA SCFs are specified in terms of interface classes and their methods

# Framework + A Set Of SCFs



One of the Parlay/OSA SCSs is called the Parlay/OSA **Framework**, and is always present, one per network

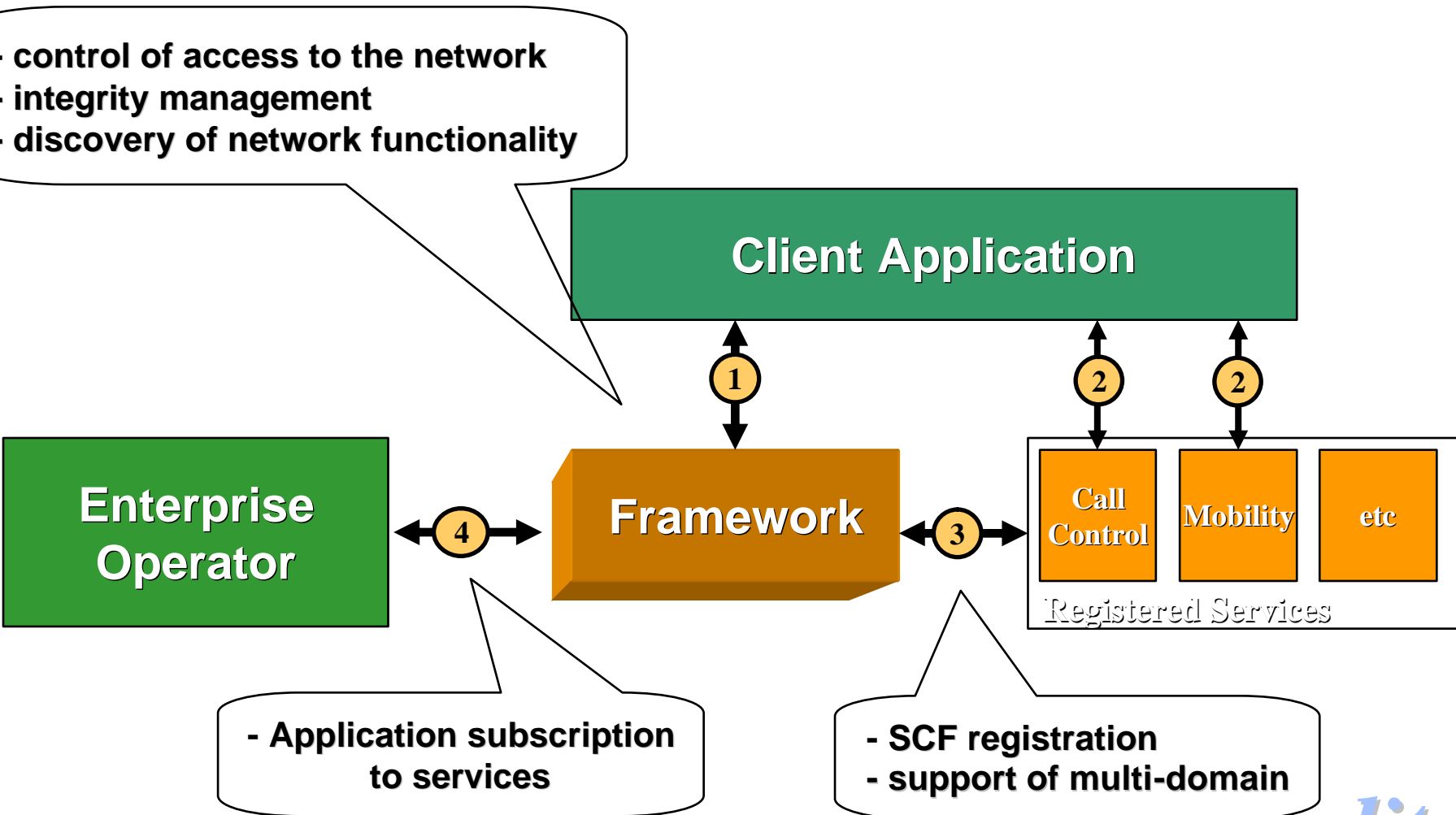
# Current Parlay/OSA SCFs

<b>Call Control</b>	<b>The Call Control family, with capabilities ranging from setting up basic calls to manipulating multimedia conference calls (see Note 1)</b>
<b>User Interaction</b>	<b>Obtain information from the end-user, play announcements, send short text messages, etc</b>
<b>User location / User status</b>	<b>Obtain location and status information</b>
<b>Terminal capabilities</b>	<b>Obtain the capabilities of an end-user terminal</b>
<b>Data session control</b>	<b>Control of data sessions</b>
<b>Generic Messaging</b>	<b>Access to mailboxes (see Note 2)</b>
<b>Connectivity Management</b>	<b>Provisioned QoS (see Note 2)</b>
<b>Account Management</b>	<b>Access end-user accounts</b>
<b>Content based Charging</b>	<b>Charge end-users for use of applications / data</b>

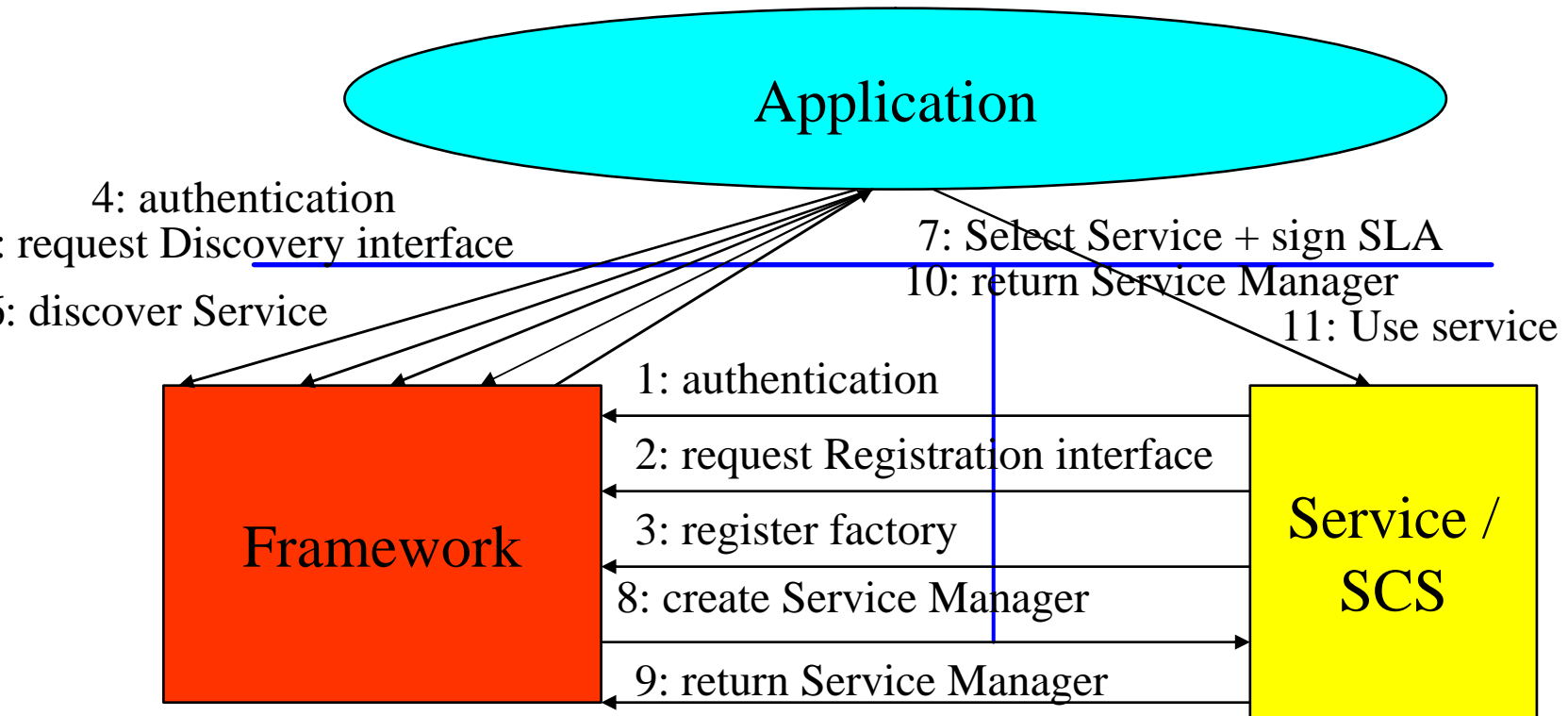
Note 1: Multimedia and Conference Call Control not part of 3GPP OSA Release 4

Note 2: Not part of 3GPP OSA Release 4

# The Parlay/OSA Framework



# How applications can use a Service / SCS



# Bodies Involved In Parlay/OSA Standardisation

OSA specification and standardisation is a **joint effort** by the following bodies:

- 3GPP CN5
- ETSI SPAN12
- ITU-T SG11 (only ref. document)
- The Parlay Group

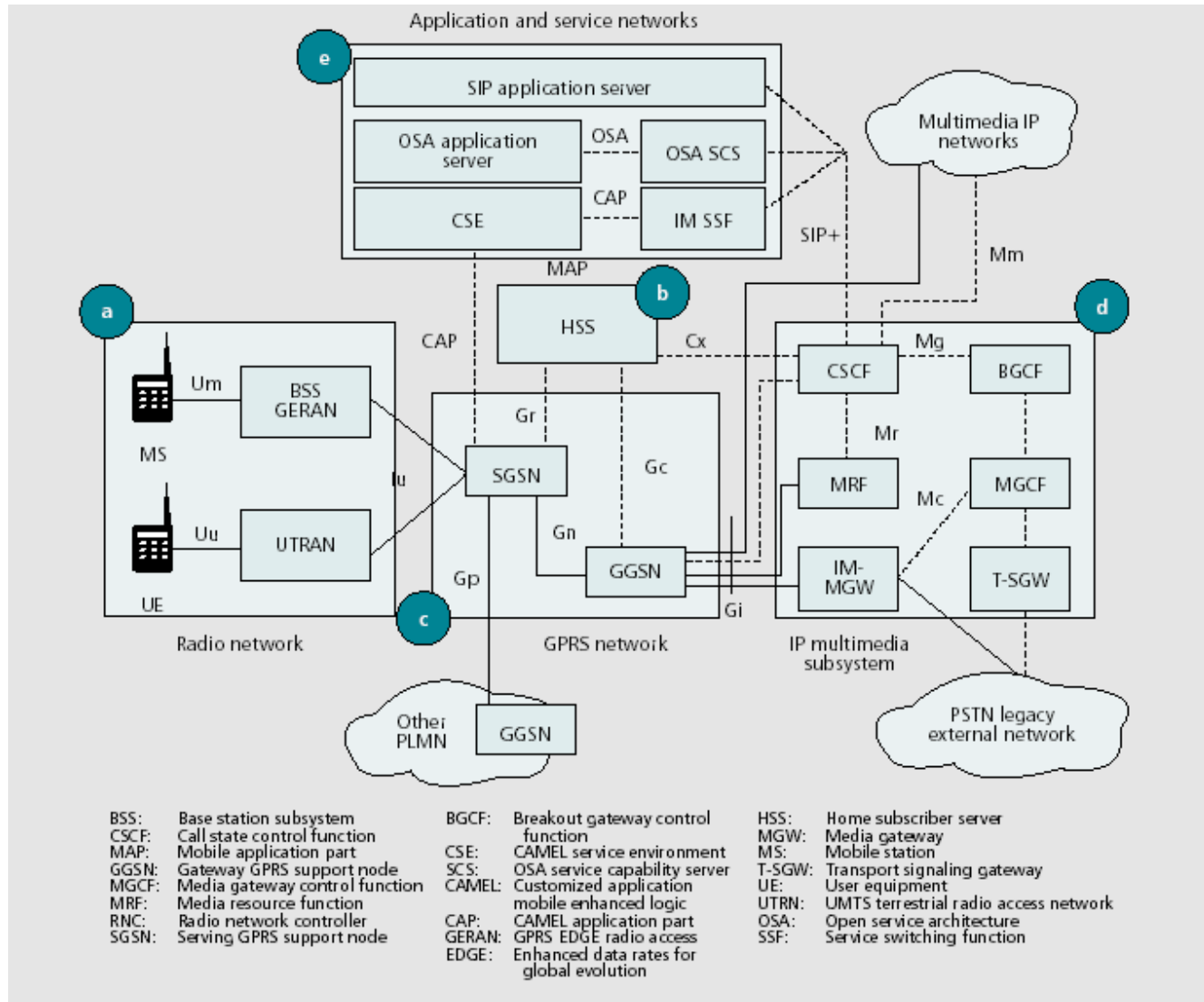


The objective is to have **a single API** for the **whole developer community**



# IMS

# 3GPP network architecture



# SIP: Session Initiation Protocol

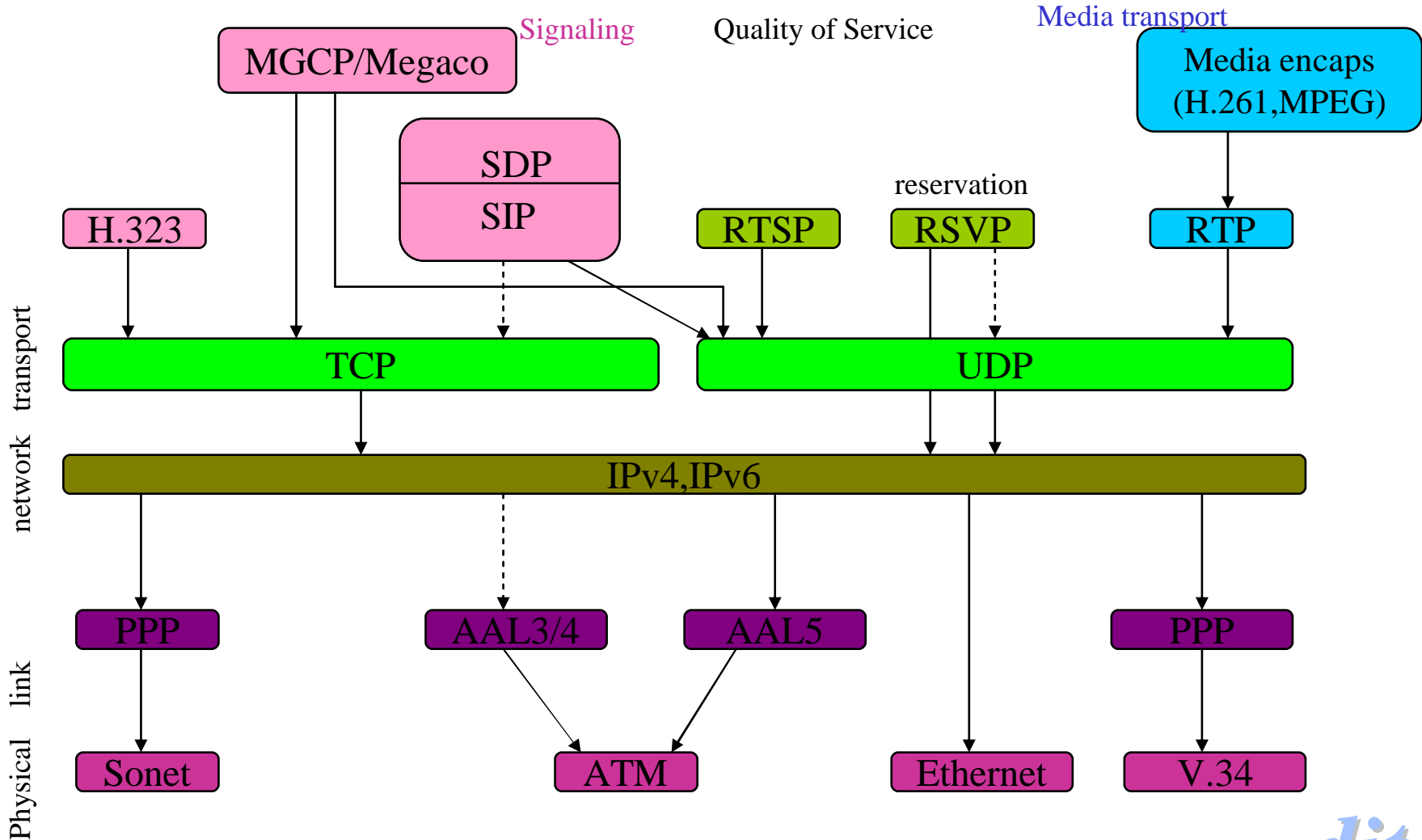
- ◆ Estándar del IETF (RFC 3261)
- ◆ Protocolo genérico de establecimiento de sesiones multimedia.
- ◆ Interoperabilidad con anteriores VoIP.
- ◆ Diseñado específicamente para IP e Internet: similar a HTTP.
- ◆ Permitirá la aparición de nuevos servicios y aplicaciones
- ◆ Escalable y flexible:
  - ▶ Bajo coste de establecimiento de llamada.

# Protocolos de tiempo real

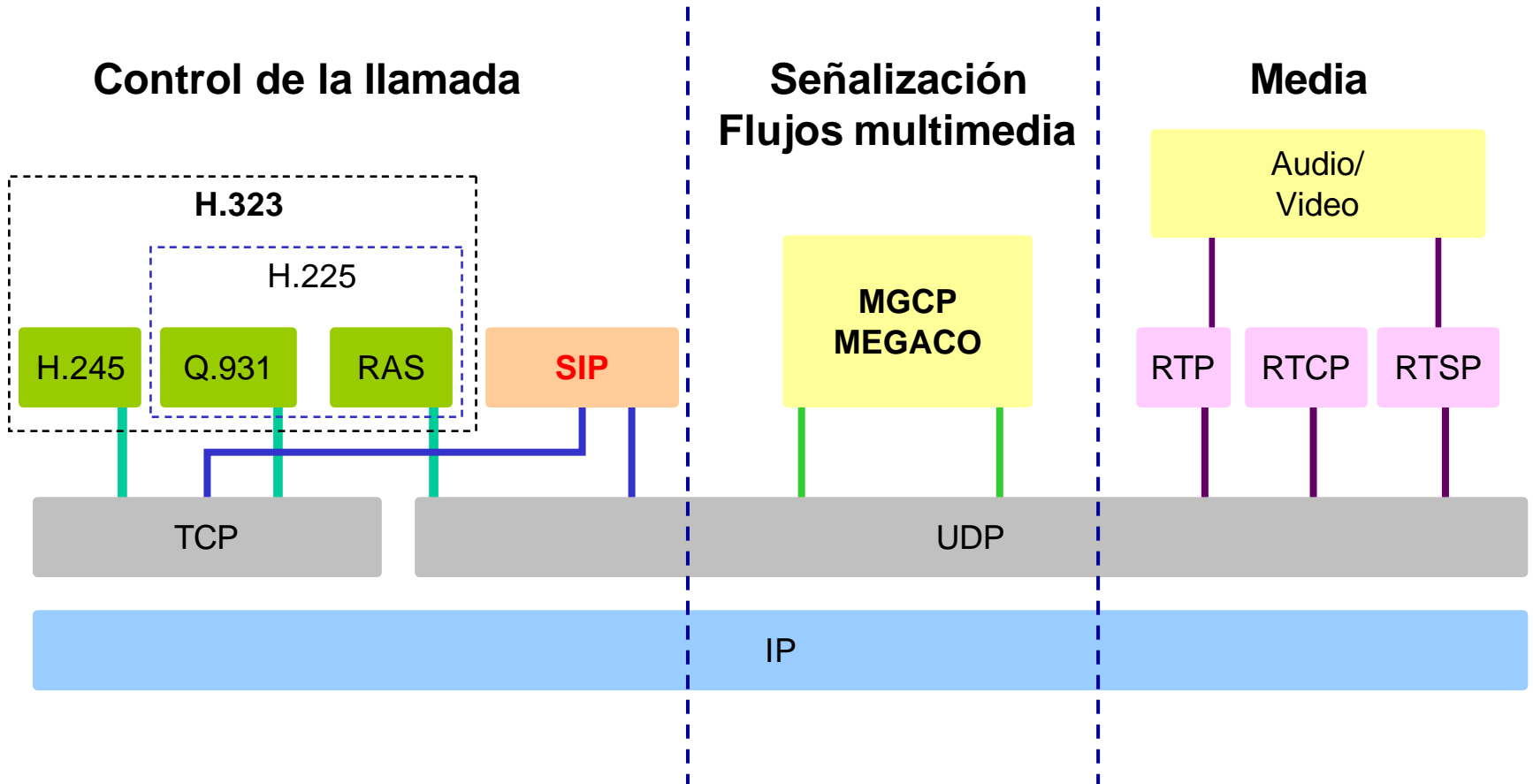
## ◆ Los protocolos definidos por el IETF

- ▶ Control de flujos y sesión: SIP, RTSP
- ▶ Descripción de los flujos SDP
- ▶ Transporte de datos t. Real: RTP/RTCP
- ▶ Reserva de recursos (QoS): RSVP, DiffSer

# Visión General



# SIP, H.323 y MGCP



# Direccionamientos @

◆ La dirección SIP es similar a la de correo. No tiene porque ser IP:

`user@host.`

◆ Por ejemplo URL son:

- ▶ `sip:hostname@vovida.org`
- ▶ `sip:hostname@192.168.10.1`
- ▶ `sip:14085551212@vovida.org`

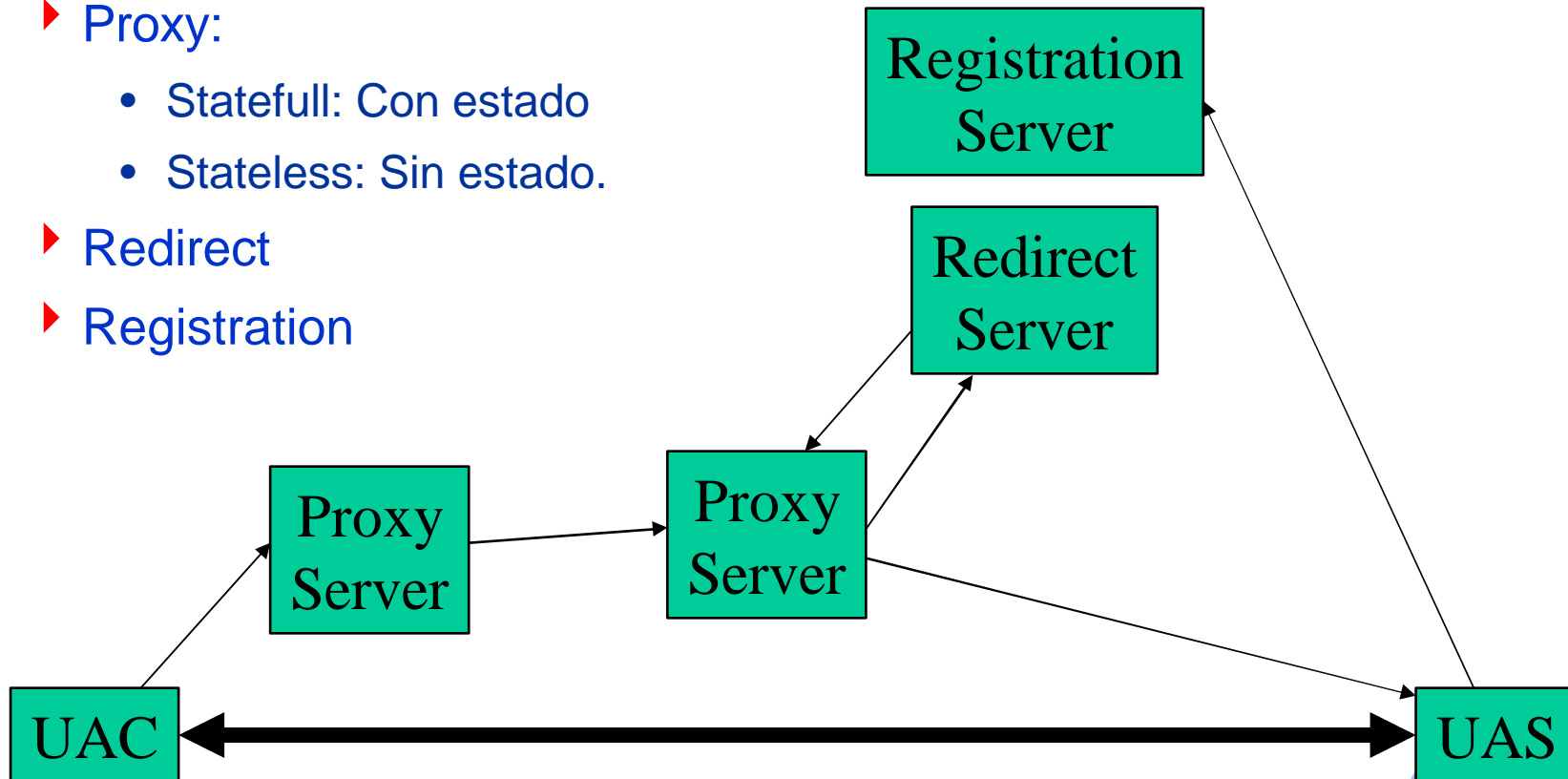
# Componentes SIP

## Agentes de Usuario:

- ▶ Cliente que inicial las llamadas
- ▶ Modo cliente y servidor.

## ◆ Servidores

- ▶ Proxy:
  - Statefull: Con estado
  - Stateless: Sin estado.
- ▶ Redirect
- ▶ Registration





# Similar a HTTP

Línea de  
comienzo

## Mensaje de petición

INVITE sip:ivan@etsit.es SIP/2.0

Call-ID: dfg56748hb4b4b2@yohoo.com  
From:<sip:mgonzalez@yohoo.com>  
To:Ivan<sip:idr@etsit.es>  
CSeq:1 INVITE  
Via:SIP/2.0/TCP 169.130.12.5  
Content-Type:application/sdp  
Content-Length:885

Línea en blanco (CRLF)

v=0  
o=bell 53655765 2353687637 IN IP4  
128.3.4.5  
c=IN IP4 135.180.144.94  
m=audio 3456 RTP/AVP 0 3 4 5

Cuerpo del  
mensaje

Campos o  
cabeceras

## Mensaje de respuesta

SIP/2.0 200 OK

Call-ID: dfg56748hb4b4b2@yohoo.com  
From:<sip:mgonzalez@yohoo.com>  
To:Ivan<sip:idr@etsit.es>  
CSeq:1 INVITE  
Via:SIP/2.0/TCP 169.130.12.5  
Content-Type:application/sdp  
Content-Length:885

Línea en blanco (CRLF)

v=0  
o=bell 53655765 2353687637 IN IP4  
128.3.4.5  
c=IN IP4 135.180.144.94  
m=audio 3456 RTP/AVP 0 3 4 5

# Mensajes SIP

## Todos los componentes usan mensajes SIP:

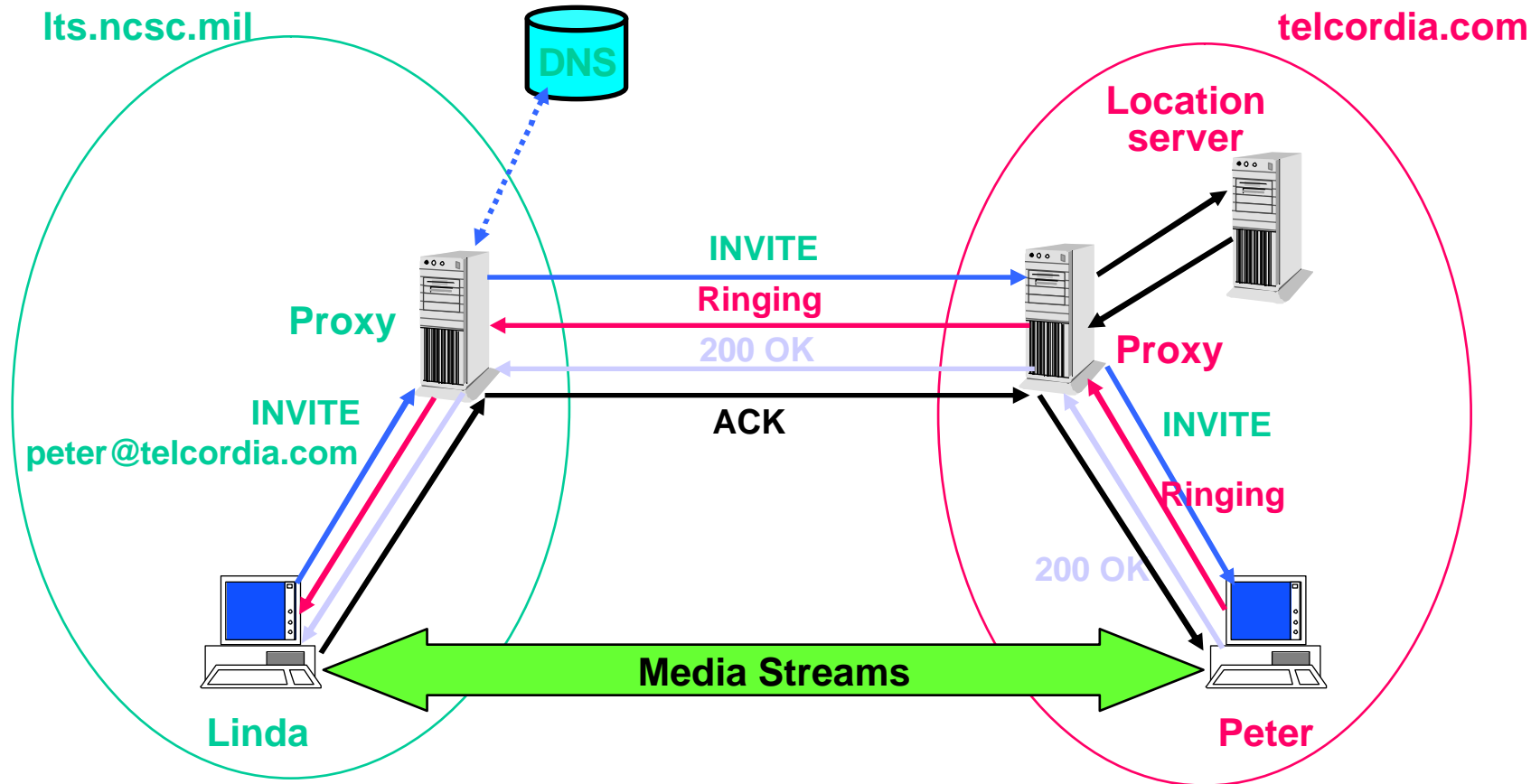
### ◆ SIP Requests:

- ▶ INVITE – Initiates a call by inviting user to participate in session.
- ▶ ACK - Confirms that the client has received a final response to an INVITE request.
- ▶ BYE - Indicates termination of the call.
- ▶ CANCEL - Cancels a pending request.
- ▶ REGISTER – Registers the user agent.
- ▶ OPTIONS – Used to query the capabilities of a server.
- ▶ INFO – Used to carry out-of-bound information, such as DTMF digits.

### ◆ SIP Responses:

- ▶ 1xx - Informational Messages.
- ▶ 2xx - Successful Responses.
- ▶ 3xx - Redirection Responses.
- ▶ 4xx - Request Failure Responses.
- ▶ 5xx - Server Failure Responses.
- ▶ 6xx - Global Failures Responses.

# Realización de una llamada



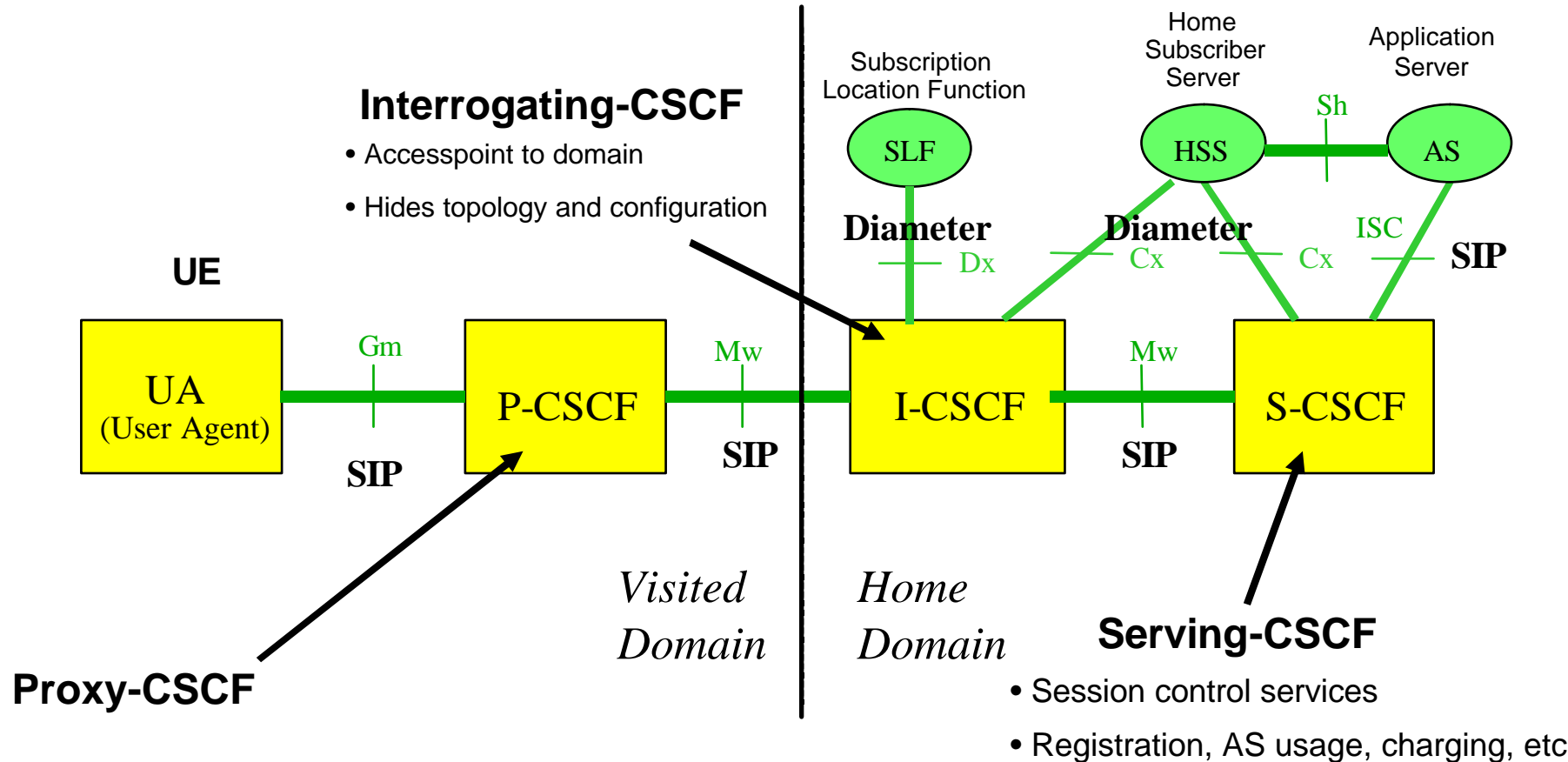
INVITE SDP proposes media type(s), IP & ports to send to  
200 OK SDP accepts/rejects media, gives IP & ports to send to

# 3GPP Internet Multimedia Subsystem

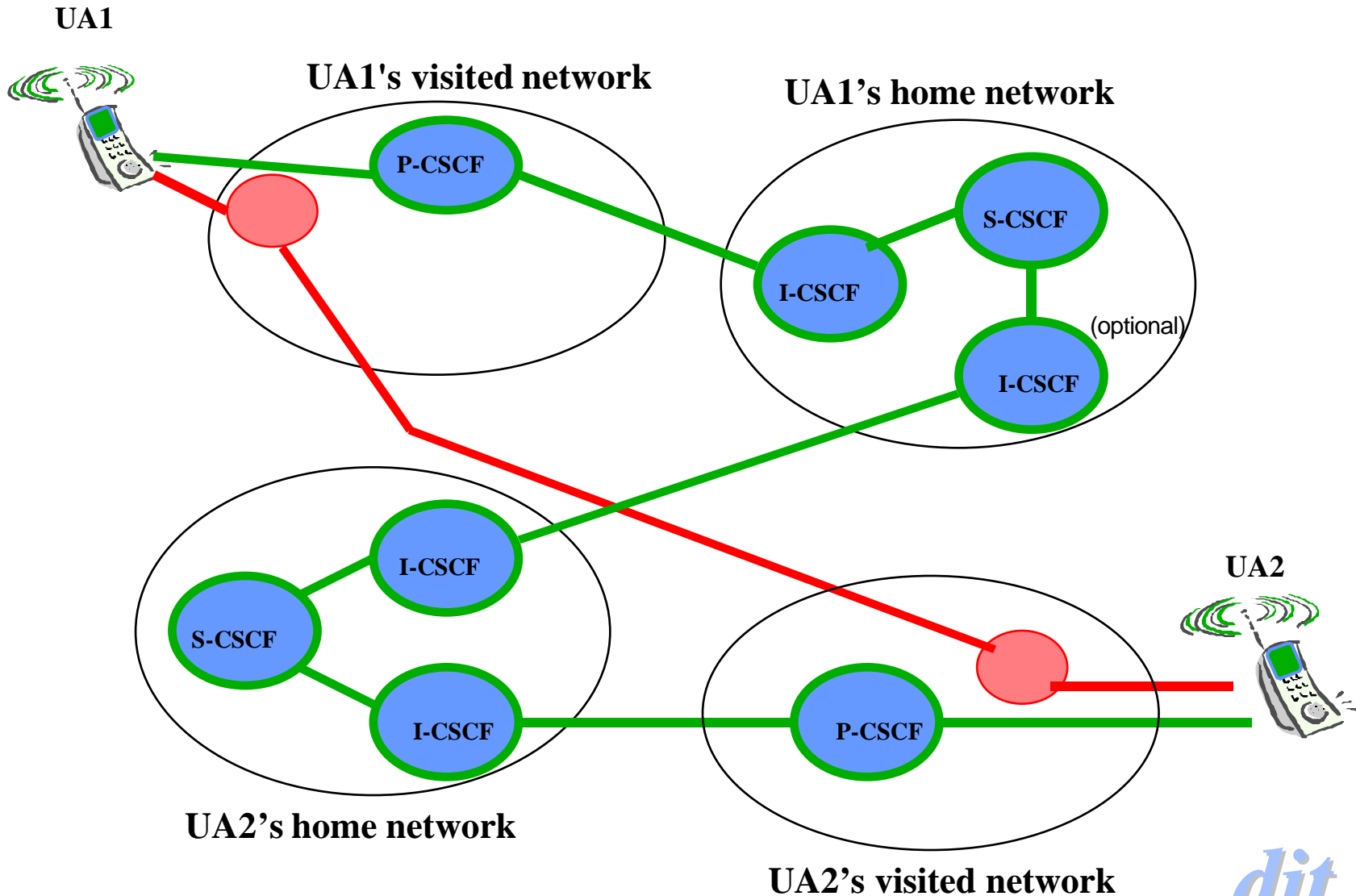
- ◆ services (call filtering, follow-me, ...) provided in home network, via Home Subscriber Server (HSS)
- ◆ *may* use CAMEL for providing services, but also
  - ▶ Call Processing Language (CPL)
  - ▶ SIP Common Gateway Interface (sip-cgi, RFC 3050)
  - ▶ SIP Servlets (JAIN)
  - ▶ VoiceXML for voice interaction (IVR)
- ◆ use ENUM (DNS) to map E.164 numbers to SIP URIs
  - ▶ +46-8-9761234 becomes 4.3.2.1.6.7.9.8.6.4.e164.arpa
- ◆ mechanisms and roles:
  - ▶ proxy servers → call routing, forking
  - ▶ user agents (UA) → voice mail, conferencing, IM
  - ▶ back-to-back UA (B2BUA) → 3<sup>rd</sup> party call control

# 3GPP Internet Multimedia Subsystem

## Call State Control Function (CSCF)



# IMS session overview



# QoS in UMTS

- ◆ Short term: signaling → tell network elements about QoS requirements
  - ▶ RSVP (IntServ)
  - ▶ DiffServ with DSCPs
  - ▶ PDP context
  
- ◆ Longer term: provisioning → allocate resources to QoS classes
  - ▶ low network utilization (overprovisioning)
  - ▶ DiffServ
  - ▶ IntServ (possibly for DiffServ classes, RFC xxxx)
  - ▶ MPLS
  
- ◆ Mechanisms can be heterogeneous
  - ▶ DSCP translation
  - ▶ localized RSVP

# QoS signaling in UMTS

- ◆ UMTS R5: two end-to-end QoS signaling scenarios
- ◆ QoS provisioning left vague
- ◆ RSVP currently not in standard
  - ▶ additional scenario featuring RSVP may be added to a later release of the standard
- ◆ QoS connected to application layer signaling (SIP)  
*SIP - Session Initiation Protocol*
  - ▶ necessary for IP telephony, not streaming or data
  - ▶ SIP allows applications to agree on address, port, codec, ...
  - ▶ standardized by IETF
  - ▶ but UMTS-specific SIP dialect
    - additional functionality compared to IETF SIP



# SIP for mobility

## ◆ Terminal mobility

- ▶ same *device*, different attachment point
  - nomadic/roaming user: change between sessions
  - mid-session mobility

## ◆ Personal mobility

- ▶ same *person*, multiple devices
- ▶ identified by SIP address-of-record

## ◆ Service mobility

- ▶ configuration information
- ▶ address book, speed dial, caller preferences, ...

## ◆ Session mobility

- ▶ hand-over active session to different device
  - e.g., cell phone to office PC

# SIP mobility vs. mobile IP

- ◆ Mobility at different layers:
  - ▶ permanent identifier
  - ▶ rendezvous point identified by that identifier
  - ▶ forwarding of messages

	mobile IP	SIP
permanent identifier	IP address	SIP AOR
temporary address	care-of-address	<b>Contact</b> header
rendezvous point	home agent (← permanent address)	registrar (← host part of AOR)
HA/FA discovery	ICMP	not needed (name)
binding update	UDP message	<b>REGISTER</b>
in visited network	foreign agent (FA)	none/outbound proxy

# 3GPP – IETF SIP differences

- ◆ SIP terminal + authentication = 3GPP terminal ~~X~~
- ◆ signaling as covert channel? → death of SMS?
- ◆ CSCFs are not quite proxies, not quite B2BUAs
  - ▶ modify or strip headers
  - ▶ initiate commands (de-registration, BYE)
  - ▶ edit SDP → violate end-to-end encryption
  - ▶ modify To/From headers

# UMTS/3G QoS classes

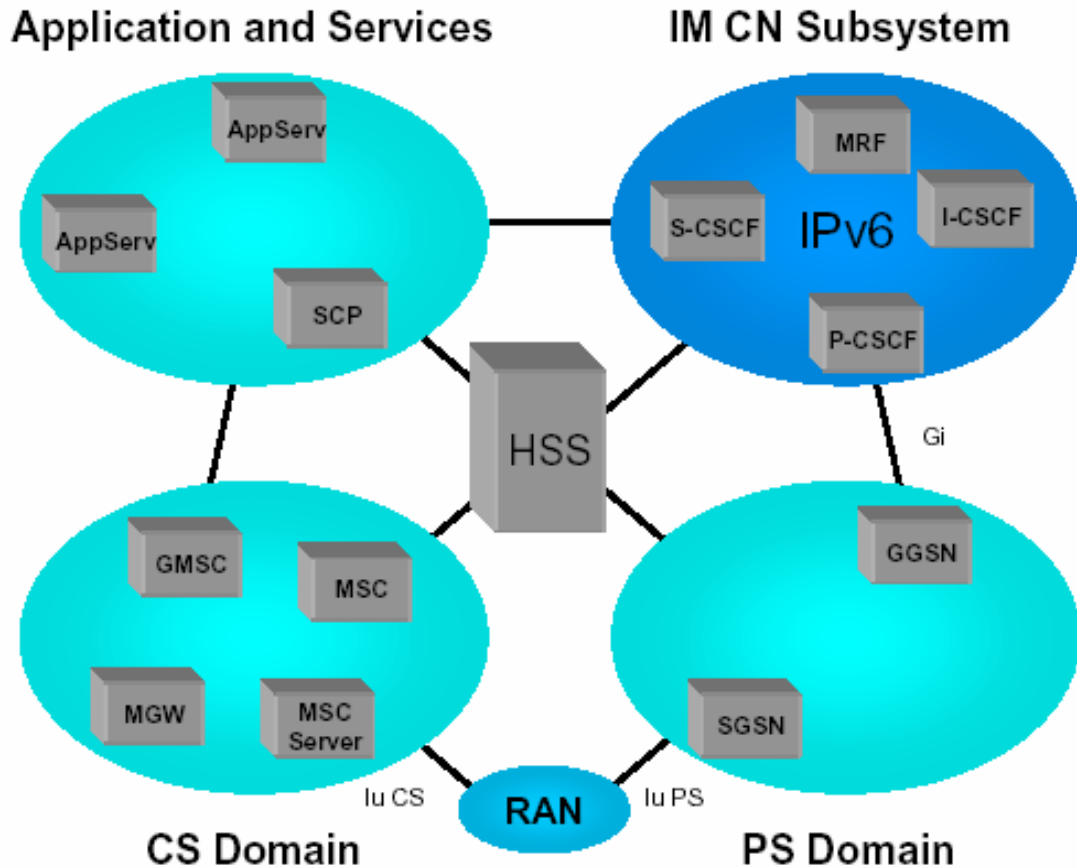
<b>conversational</b>	voice, video conferencing	low delay, strict ordering
<b>streaming</b>	video streaming	modest delay, strict ordering
<b>interactive</b>	web browsing, games	modest delay
<b>background</b>	email download	no delay guarantees

# CSCF Roles & Features

CSCF	Behaviour like	Features/Functions
<b>P-CSCF</b> (Proxy)	<ul style="list-style-type: none"> <li>◆ Proxy server</li> <li>◆ User Agent (abnormal cases)</li> </ul>	<p><b>Register</b></p> <ul style="list-style-type: none"> <li>◆ Forward SIP register to I-CSCF by home domain name</li> </ul> <p><b>Session Flow</b></p> <ul style="list-style-type: none"> <li>◆ Forward SIP messages from UE to SIP server (S_CSCF) as a result fo registration procedure</li> <li>◆ Detect an emergency session and select a S_CSCF to handle emergency session</li> <li>◆ Authorization of bearer resources &amp; QoS management</li> </ul>
<b>I-CSCF</b> (interrogating)	<ul style="list-style-type: none"> <li>◆ Proxy server</li> <li>◆ User Agent (register reject)</li> </ul>	<p><b>Register</b></p> <ul style="list-style-type: none"> <li>◆ Assignin a Service CSCF to the subscriber</li> </ul> <p><b>Session Flow</b></p> <ul style="list-style-type: none"> <li>◆ Routing a SIP request from another network towards the C_CSCF (Serving Terminating UE)</li> <li>◆ Obtaining the S_CSCF adres from HSS</li> </ul>
<b>S-CSCF</b> (Serving)	<ul style="list-style-type: none"> <li>◆ Registrar</li> <li>◆ Proxy Server</li> <li>◆ User Agent</li> </ul>	<p><b>Register</b></p> <ul style="list-style-type: none"> <li>◆ As A Registrar, store registration info to HSS</li> </ul> <p><b>Session-related/unrelated Flow</b></p> <ul style="list-style-type: none"> <li>◆ S-CSCF performs session control services for the UE</li> <li>◆ Interaction with Service Platforms</li> <li>◆ Maintaining a session state as needed by teh netowrk operator for support of the services</li> </ul>

# UMTS Release 4/5 Architecture

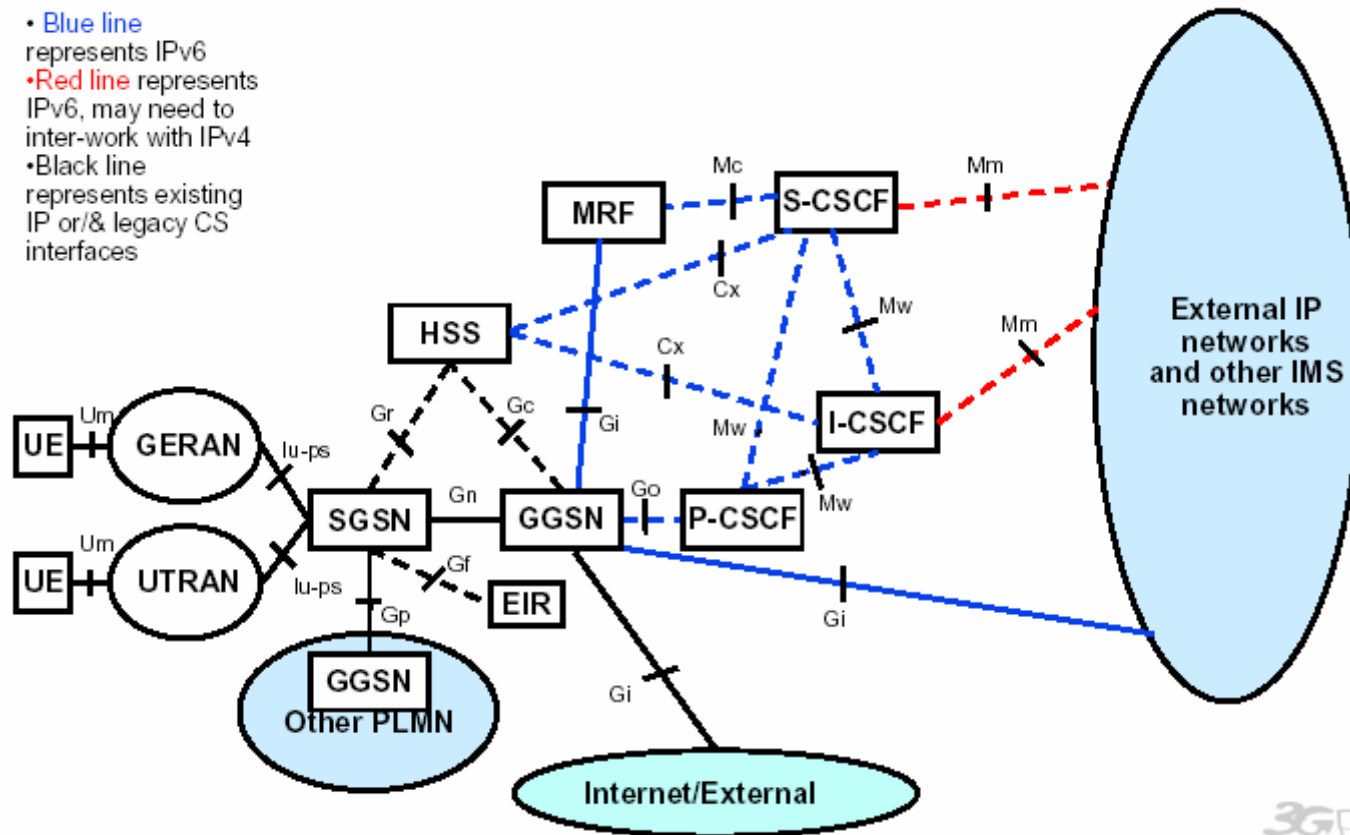
## Release 4/5 Architecture



# UMTS IP multimedia

## Simplified model for IP Multimedia

- Blue line represents IPv6
- Red line represents IPv6, may need to inter-work with IPv4
- Black line represents existing IP or/ & legacy CS interfaces



3GPP