



Modelo de Aplicación de Sesión Multimedia

Federico Montesino Pouzols

Tutores: Diego R. López y Manuel Valencia

Proyecto Fin de Carrera

Ingeniería Informática

Escuela Técnica Superior de Ingeniería Informática

Universidad de Sevilla



Objetivo



Desarrollo de una implementación *libre* de los protocolos *estándar* de nivel de transporte y aplicación definidos en la arquitectura de sistemas multimedia del IETF.

- Énfasis en el desarrollo de una implementación genérica y flexible del modelo de protocolos de transporte RTP.



Guión (1/2)



- Contexto
- Transporte en Tiempo Real
 - Sistemas Multimedia en Tiempo Real y Alternativas de Transporte
 - Protocolo/Modelo RTP
- Control de Sesiones Multimedia
 - Funciones
 - Arquitecturas de Control: H.323 e IETF
 - Arquitectura del IETF



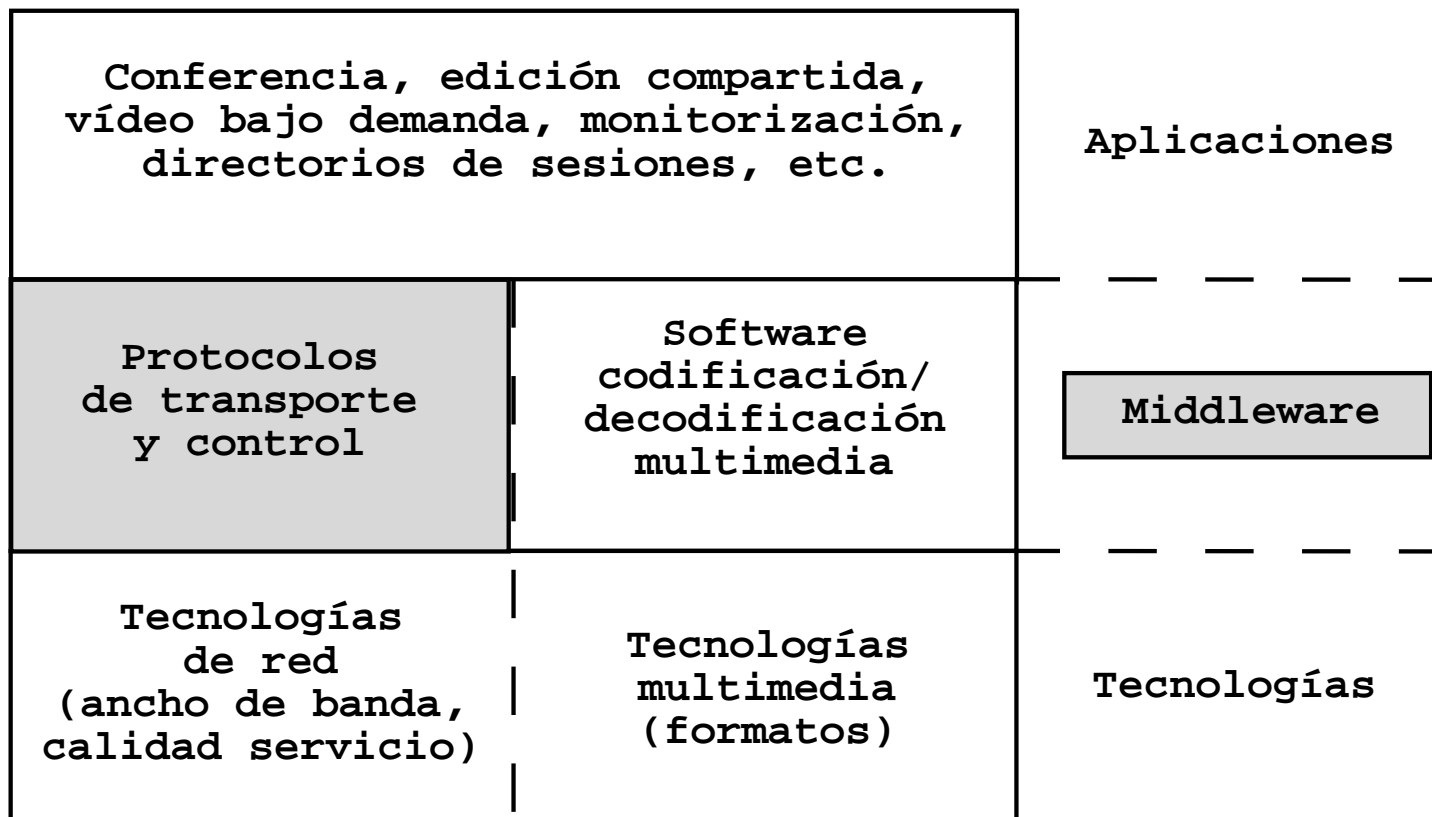
Guión (2/2)



- Diseño e Implementación del Sistema
 - Principios de Diseño
 - Arquitectura
 - GNU Common C++
 - GNU ccRTP
 - GNU oSIP
 - FreeSDP
- Conclusiones y Trabajo Futuro



Contexto





Transporte en Tiempo Real



Sistemas Multimedia en Tiempo Real



- No sólo corrección lógica, además *corrección temporal*.
- Son sistemas en tiempo real no estricto.
- Etapas de transporte de datos: codificación, empaquetado, transporte, desempaquetado y decodificación.
- Parámetros temporales: son relevantes tanto la duración como la variación de la duración de las etapas.



Alternativas de Transporte



- Transporte de datos y señalización (control).
- Protocolos “tradicionales”. TCP no es válido. UDP es válido pero insuficiente.
- Protocolo de transporte en tiempo real del IETF: RTP
- Protocolos “innovadores”: SCTP, UDP Lite.
- Problemas: control de congestión. Alternativas con control de congestión: DCCP



RTP



- Más que un protocolo, es un *modelo de protocolos*.
- Principios de diseño novedosos
 - Segmentación de nivel de aplicación
 - Procesamiento integrado de niveles
- Datos: RTP
- Control de la transferencia: RTCP
- Elementos intermedios: traductores y mezcladores de nivel RTP





Control de Sesiones Multimedia



Control de sesiones



- Funciones:
 - Establecer, modificar y finalizar llamadas/sesiones.
 - Registro y localización de participantes. Movilidad.
 - Gestión del conjunto de participantes.
 - Descripción de características de las sesiones y negociación de capacidades de los participantes.
- Arquitecturas estándar: H.323 y SIP (IETF).



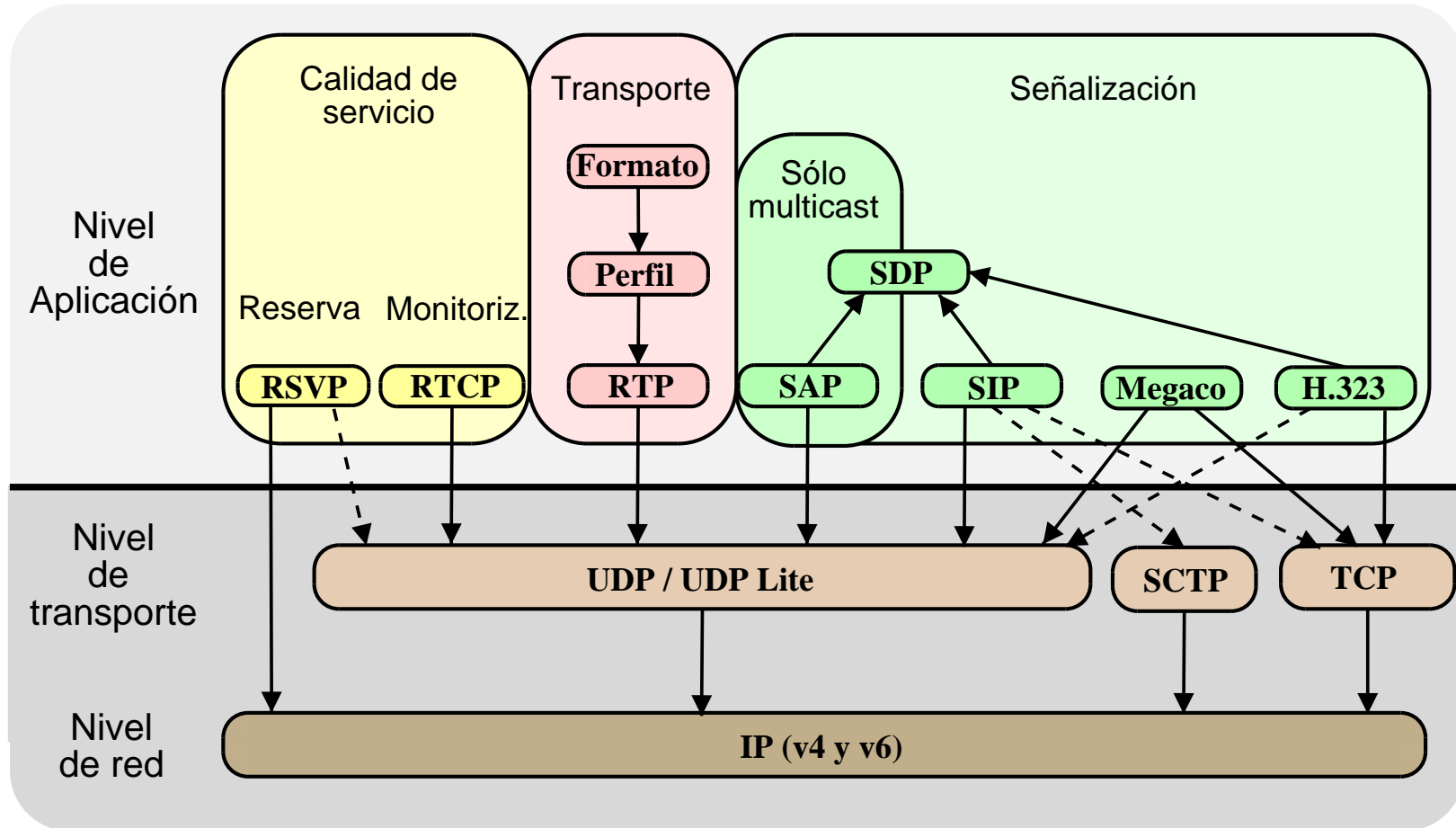
Arquitectura IETF (1/2)



- Arquitectura distribuida
- Basada en un conjunto de protocolos independientes.
- Combinable con H.323.
- Establecimiento, modificación y finalización de sesiones: protocolo SIP.



Arquitectura IETF (2/2)



Arquitectura de control basada en SIP



- Integrada en la infraestructura web.
- Elementos
 - Agentes de usuario (cliente y servidor)
 - Servidores: registro, redirección y proxys
- Protocolo y sintaxis asimilable a HTTP.
- Válido para unicast y multicast. Complementado con SAP.



Descripción de Sesiones



- Formato estándar de descripción de sesiones: SDP.
- Proporciona la información necesaria para participar en una sesión.
- Sintaxis simple y ampliable.
- No pensado para negociación de capacidades.
- Alternativa para el futuro: SDPng.





Diseño e Implementación del Sistema



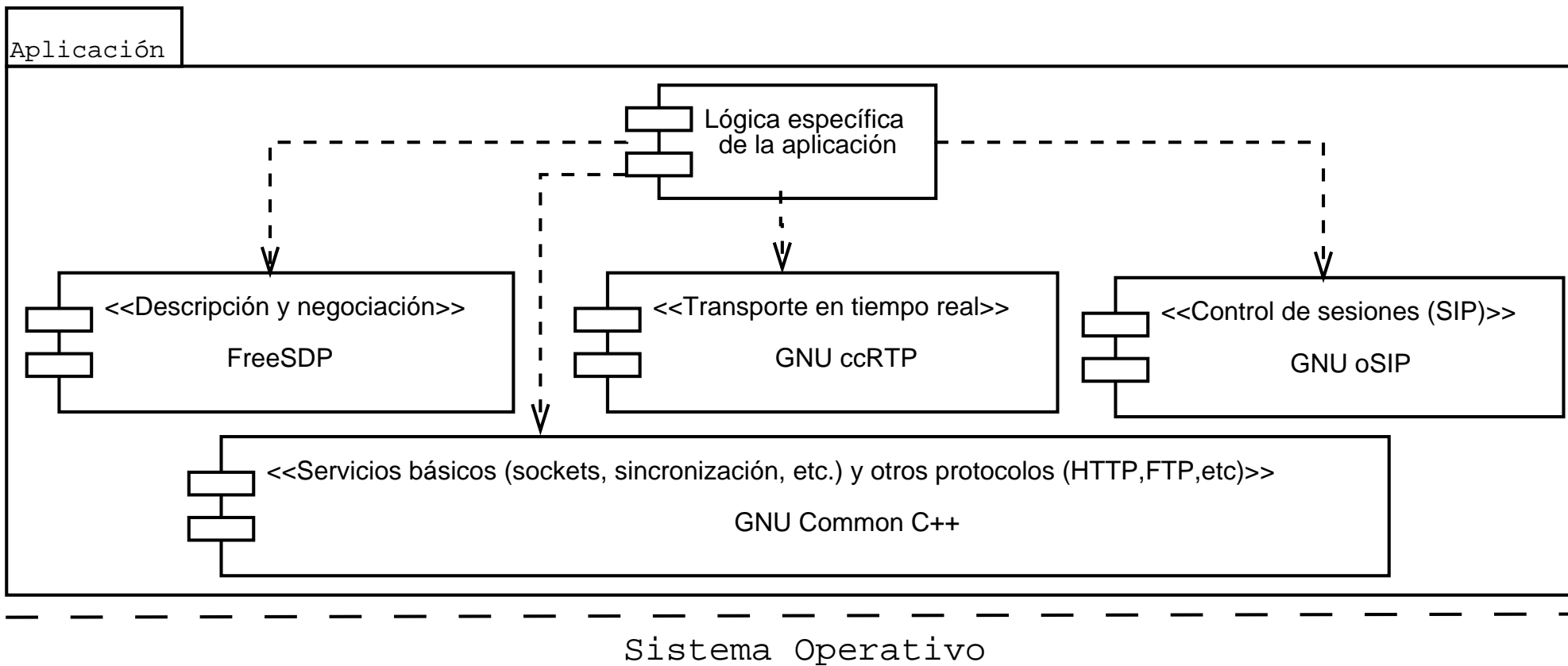
Requisitos



- Reflejar principios de diseño de los protocolos
 - Segmentación de nivel de aplicación y procesamiento integrado de niveles.
 - Componentes independientes y modularidad.
 - Flexible y genérico. Por tanto, portable.
 - Escalable.



Arquitectura del Sistema



Líneas Generales de Implementación



- Lenguajes base: C++ y C. Programación orientada a objetos, genérica y orientada a aspectos.
- Estructura fuentes y paquetes: estándares GNU.
- Múltiples y diversas plataformas.
- Documentación: manual y manual de referencia.



GNU Common C++



- Modelo genérico para aplicaciones de red.
- Abstracciones para:
 - Servicios del sistema (hilos, sockets, entrada/salida).
 - Patrones de diseño comunes.
 - Protocolos estándar: HTTP, FTP, SSL, XMLRPC.
- Integración con aplicaciones y servicios web.
Manipulación URLs y XML.
- Eficiente, multihilo, multiprocesador.



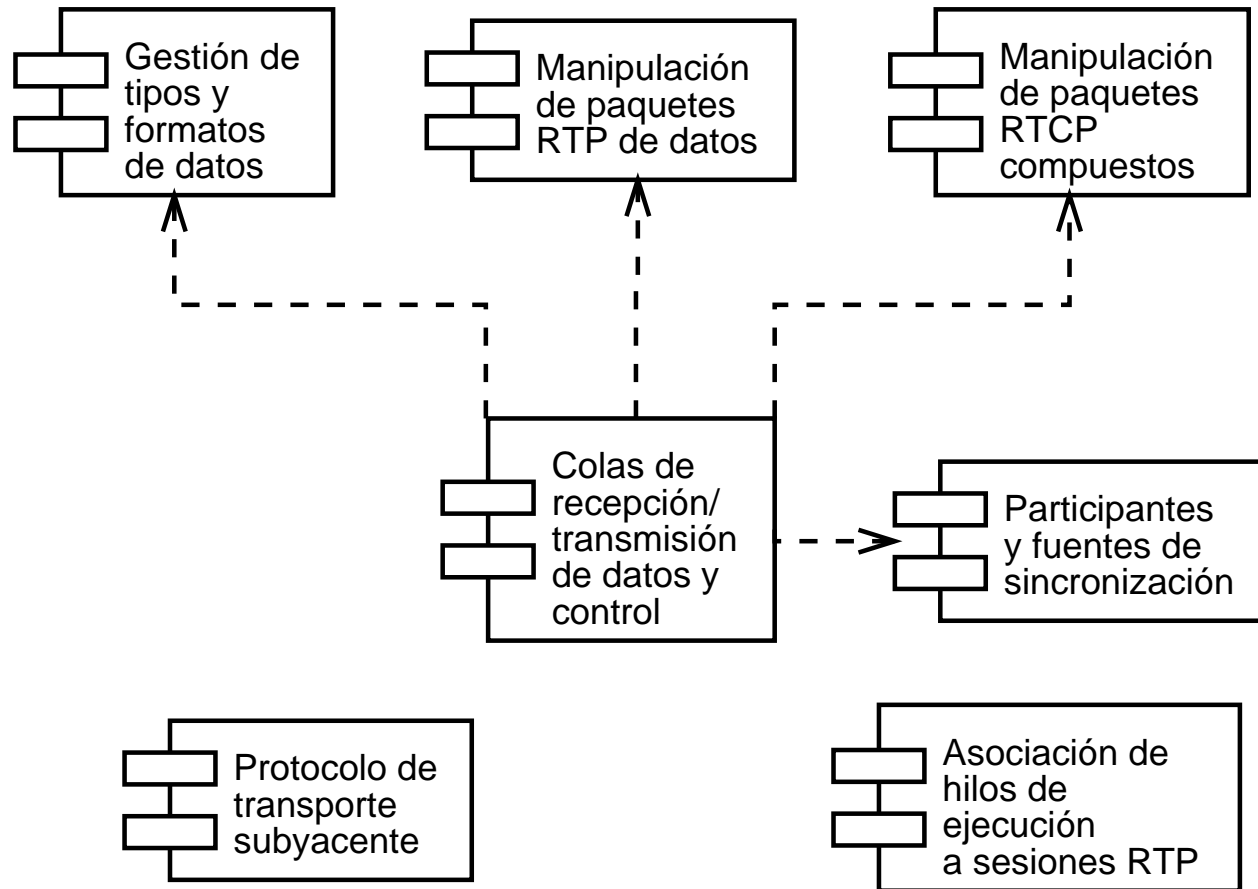
GNU ccRTP (1/2)



- Proporciona un modelo de programación para el modelo de protocolos RTP. Flexible y actualizada.
- Principios de diseño:
 - Varios niveles de interfaz.
 - Genericidad respecto a servicios del sistema.
 - Elementos independizables desacoplados.
- Solución a dos categorías de problemas:
 - Implementación del modelo RTP.
 - Implementación de aplicaciones multimedia de tiempo real.



GNU ccRTP (2/2): Componentes



GNU oSIP



- Dos componentes principales:
 - Manipulación de mensajes SIP.
 - Gestión de las máquinas de estados de las transacciones que componen las llamadas SIP.
- Analizador sintáctico de mensajes permisivo.
- Constructor de mensajes SIP estándar.
- Máquinas de estados redefinibles.
- Gestión de negociación de capacidades mediante SDP.



FreeSDP



- Solución integrada para la manipulación de descripciones SDP.
- Incorpora sintáxis base y ampliaciones.
- Dos componentes:
 - Analizador sintáctico permisivo.
 - Constructor de descripciones SDP estándar.





Conclusiones y Trabajo Futuro



Conclusiones



- Sistema completo y libre.
- Estándares, modular, portable, seguro, ya operativo
- ccRTP es flexible y válida como plataforma de experimentación de algoritmos específicos de los sistemas multimedia en tiempo real.



Trabajo Futuro



- Ampliaciones en las bibliotecas.
- Desarrollo de un conjunto base completo de aplicaciones.
- Integración con sistemas de convergencia: redes IP – redes conmutadas.

