

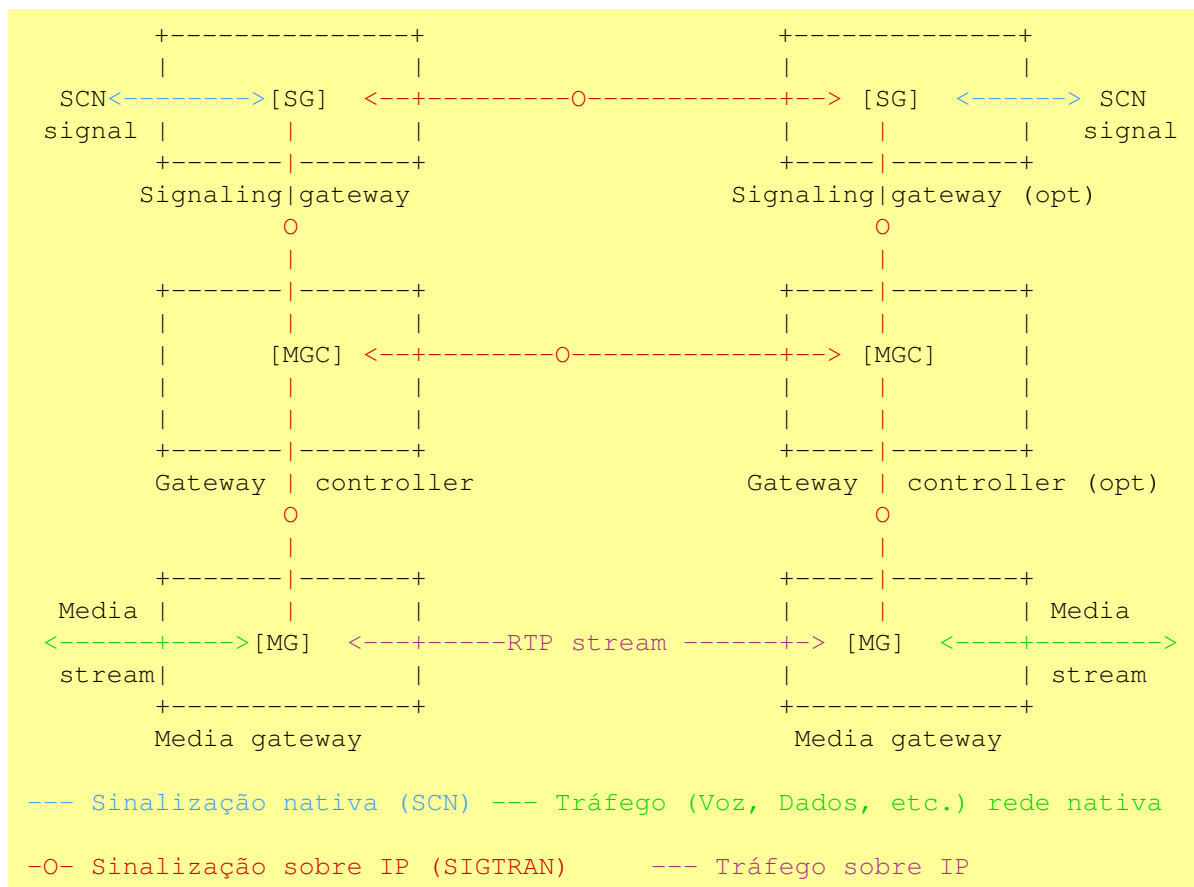
Transporte de Sinalização sobre IP (SIGTRAN)

- O SIGTRAN é:
 - Um grupo da IETF dedicado ao estudo do transporte de informação de sinalização (por mensagens) sobre redes IP.
 - <http://www.ietf.org/html.charters/sigtran-charter.html>
- Exemplos de aplicação:
 - Controlo de acesso à Internet por *dial-up*
 - Interligação de telefonia sobre IP com a PSTN (PLMN, ISDN)
 - Acesso a entidades do SCN a bases de dados (na Internet)
 - Desenvolvimento de novos serviços
- Conjunto de drafts e RFCs (alguns exemplos):
 - Architectural Framework for Signaling Transport (RFC 2719)
 - Stream Control Transmission Protocol (RFC 2960)
 - SCTProtocol Management Information Base (RFC 3873)
 - ISDN Q.921-User Adaptation Layer (RFC 3057)
 - Signalling Connection Control Part User Adaptation Layer (SUA) (RFC 3868)
 - SS7-MTP2: User Peer-to-Peer Adaptation Layer (M2PA) (RFC 4165)
 - SS7-MTP3: User Adaptation Layer (M3UA) (RFC 4666)

Arquitetura Genérica para Transporte de Sinalização sobre IP

- Âmbito de uma arquitectura para transporte de sinalização:
 - Definição de métodos de encapsulamento (formato)
 - Protocolos extremo-a-extremo
 - Suporte, utilizando as capacidades do IP, dos requisitos funcionais e não-funcionais do transporte de sinalização
- Arquitectura Genérica:
 - Define entidades funcionais e físicas
 - Definição de relações entre entidades funcionais (físicas)
 - Definição de requisitos para o transporte de sinalização
- Alguns termos importantes:
 - *SCN: Switched Circuit Network*
 - Rede que transporta o tráfego (e.g., PSTNs, PLMNs)
 - *SEP: Signaling End Point*
 - *STP: Signaling Transfer Point*
 - *Backhaul*
 - *Transporte de sinalização desde a interface onde é recebida a informação de sinalização até ao ponto onde essa informação é processada*

Entidades lógicas da arquitectura



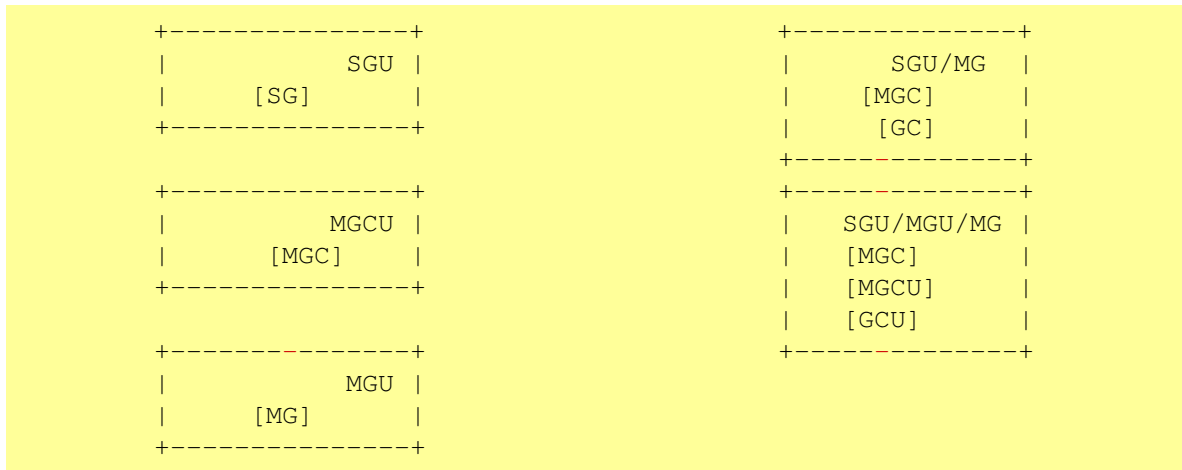
Componentes:

- **Media Gateway**
 - Recebe o tráfego da rede nativa e transporta-o sobre a rede IP
- **Media Gateway Controller**
 - Controla os recursos da MG
- **Signaling Gateway**
 - Recebe a sinalização da rede nativa (SCN) podendo trerminar, traduzir ou transferi-la através da rede IP,

• Nota:

- Esta arquitectura é genérica e independente da SCN

Entidades físicas da arquitectura



O que não faz parte da arquitectura:

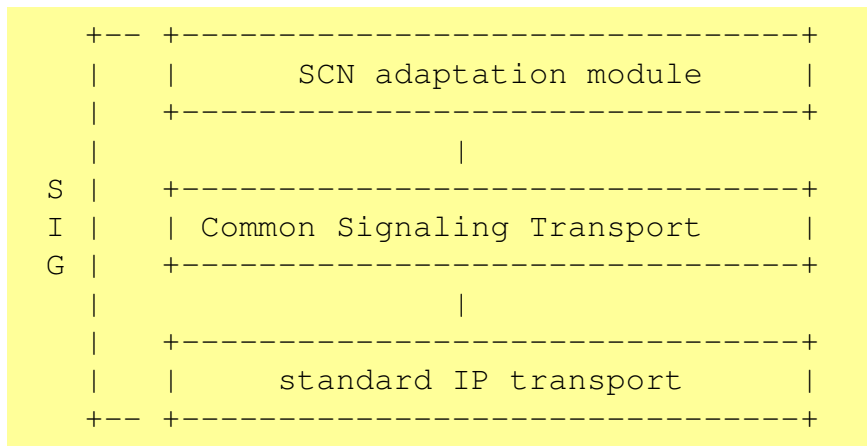
- *Definição do protocolo de sinalização*
- *Especificação das funções dentro do SGU e MGU*
- *Algumas das funções de gestão do SGU e MGU*
 - (e.g., determinação de IPs, teste das ligações)

Componentes:

- *Media Gateway Unit*
Unidade física contendo a funcionalidade MG
- *Media Gateway Controller Unit*
Unidade física contendo a funcionalidade MGC
- *Signaling Gateway Unit*
Unidade física contendo a funcionalidade SG
- (Entidades em que são implantadas várias funcionalidades)

Arquitetura protocolar *Signaling Transport(SIG)*

Componentes Arquitectura Protocolar SIG



SCN adaptation module

- Suporte das primitivas específicas do protocolo de sinalização

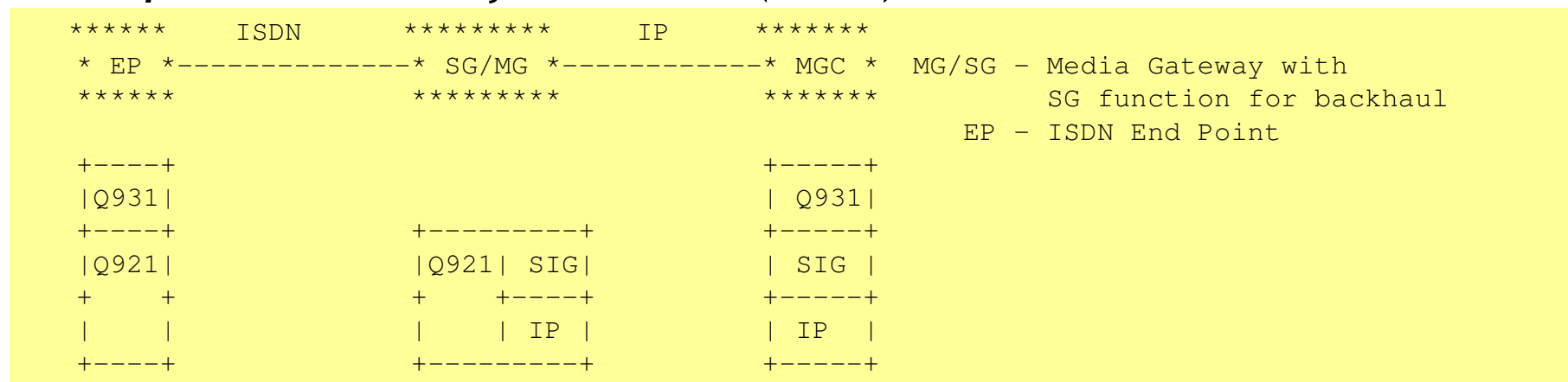
Common Signaling Transport

- Conjunto de funcionalidades para o suporte de protocolos de sinalização

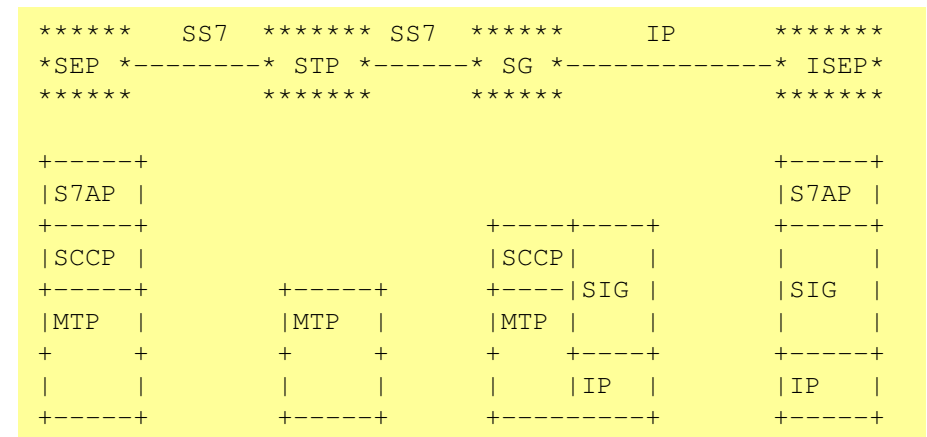
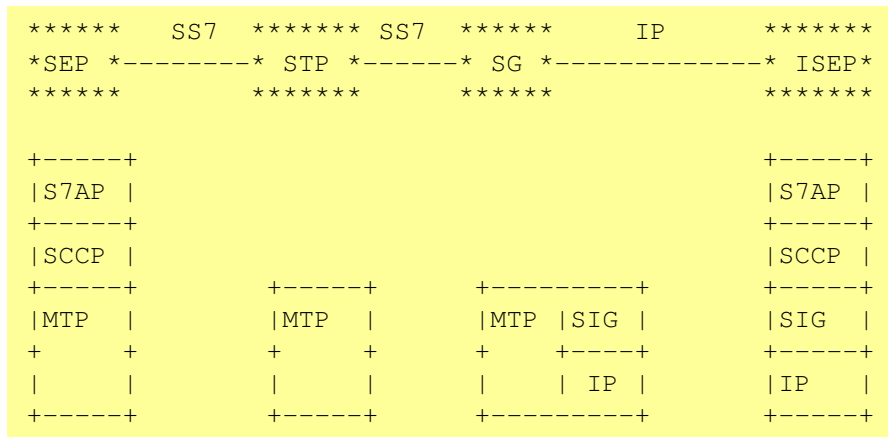
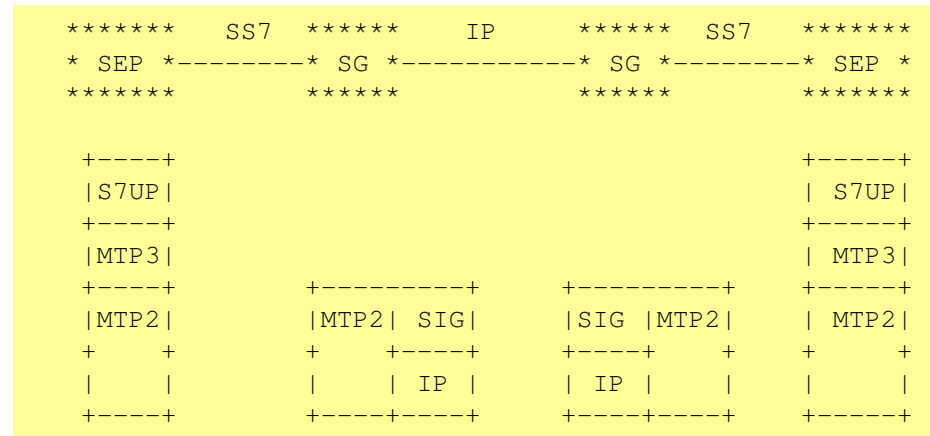
- *standard IP transport*

- Protocolo IP não-modificado

Exemplo de utilização do SIG (RDIS)



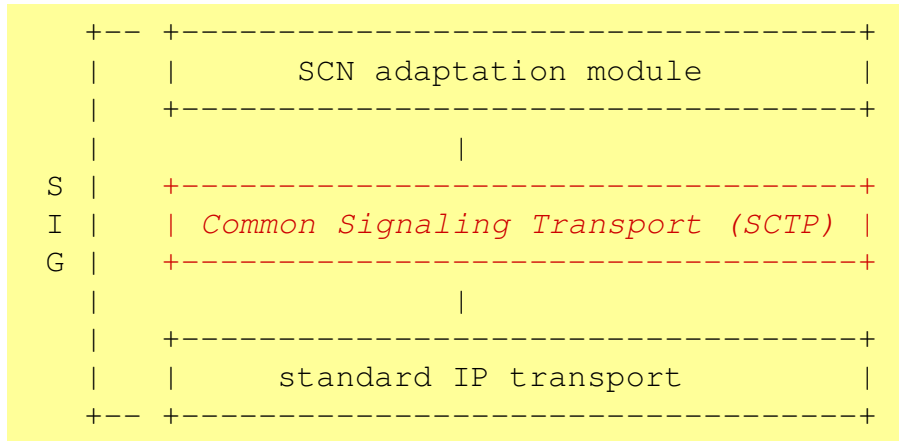
Signalling Transport (SIG) e SS7



Requisitos Signalização sobre IP

- Requisitos Funcionais Genéricos
 - Suporte e identificação para diferentes tipos de protocolos de sinalização
 - Definição de um protocolo base comum para o transporte de sinalização
 - Disponibilizar as funcionalidades definidas para os níveis inferiores do SCN:
 - Ex.: Controlo de fluxo, ordenamento de msgs, detecção de erros
 - Multiplexagem de várias sessões SCN na mesma sessão de transporte
 - Controlo de congestão (podendo utilizar funções do SCN)
- Requisitos de Desempenho para SS7 (exemplos):
 - Perda de mensagens: < 1 em $1E+7$
 - Mensagens fora de sequência: < 1 em $1E+10$
 - Mensagens com erro (não detectado pelo transporte): < 1 em $1E+9$
 - Disponibilidade: < 10 min por ano fora de serviço
 - Procedimentos extremo-a-extremo: atraso < 500 a 1200 ms

Implementação da arquitectura: SCTP (*Stream Control Transmission Protocol*)



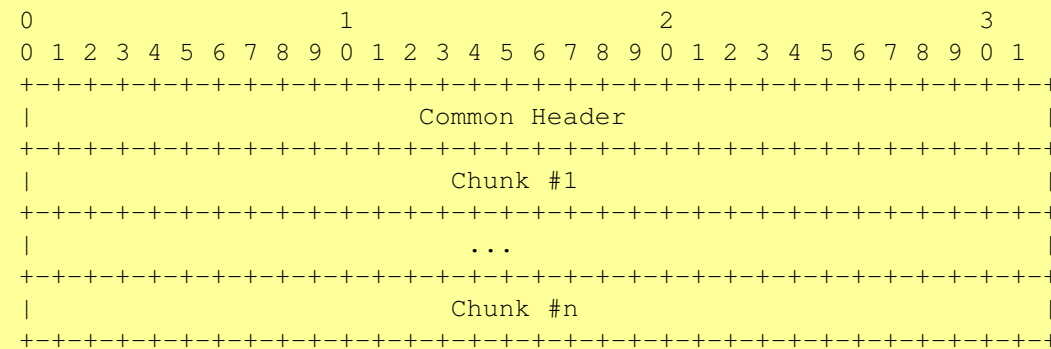
Interface para o nível superior:

- **Inicialização do SCTP**
 - `INITIALIZE ([local_port])`
- **Inicialização de uma associação**
 - `ASSOCIATE(local, remote, ...)`
- **Terminação de uma associação**
 - `SHUTDOWN(association_id)`
- **Envio de mensagem**
 - `SEND(assoc_id, buffer, count, ...)`
- **Recepção de mensagem**
 - `RECEIVE(assoc_id, buffer, size, ...)`
- **Pedido de heartbeat**
 - `REQUESTHEARTBEAT(assoc_id, dest_addr)`

- *O SCTP é um protocolo fiável de nível transporte sobre uma rede não fiável e sem ligação (IP)*
- *Oferece os seguintes serviços:*
 - *Trânsferencia de dados confirmada, ordenada e sem duplicados*
 - *Utilização de um TSN para as mensagens*
 - *Disponibilidade verificada com “heartbeats”*
 - *Multiplexagem de várias msg num mesmo pacote SCTP (ordenamento por fluxo de msgs)*
 - *Controlo de fluxo*
 - *Mecanismos de segurança (baseados em cookies)*

Implementação da arquitectura: SCTP

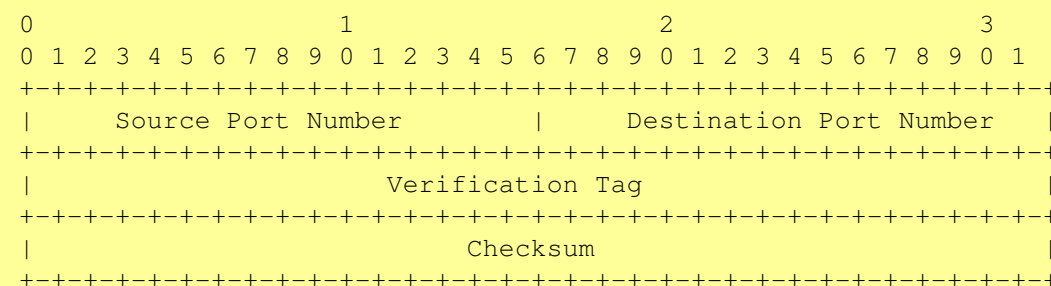
(Formato das mensagens)



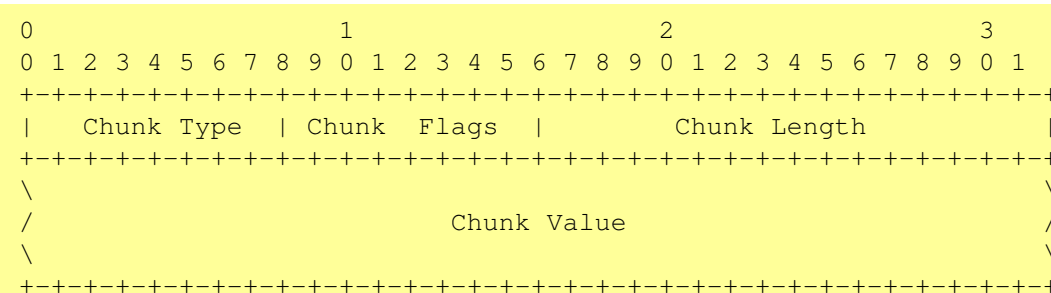
- *IP + Porto*
identifica cada extremo do SCTP
- *Verification Tag*
Utilizado para verificar a origem da mensagem

• *Chunk Type*

- 0 - *Payload Data (DATA)*
- 1 - *Initiation (INIT)*
- 2 - *Initiation Ack (INIT ACK)*
- 3 - *Selective Ack (SACK)*
- 4 - *Heartbeat Request (HEARTBEAT)*
- 5 - *Heartbeat Ack (HEARTBEAT ACK)*
- 6 - *Abort (ABORT)*
- 7 - *Shutdown (SHUTDOWN)*
- 8 - *Shutdown Ack (SHUTDOWN ACK)*
- 9 - *Operation Error (ERROR)*
- 10 - *State Cookie (COOKIE ECHO)*
- 11 - *Cookie Ack (COOKIE ACK)*
- 12 - ...



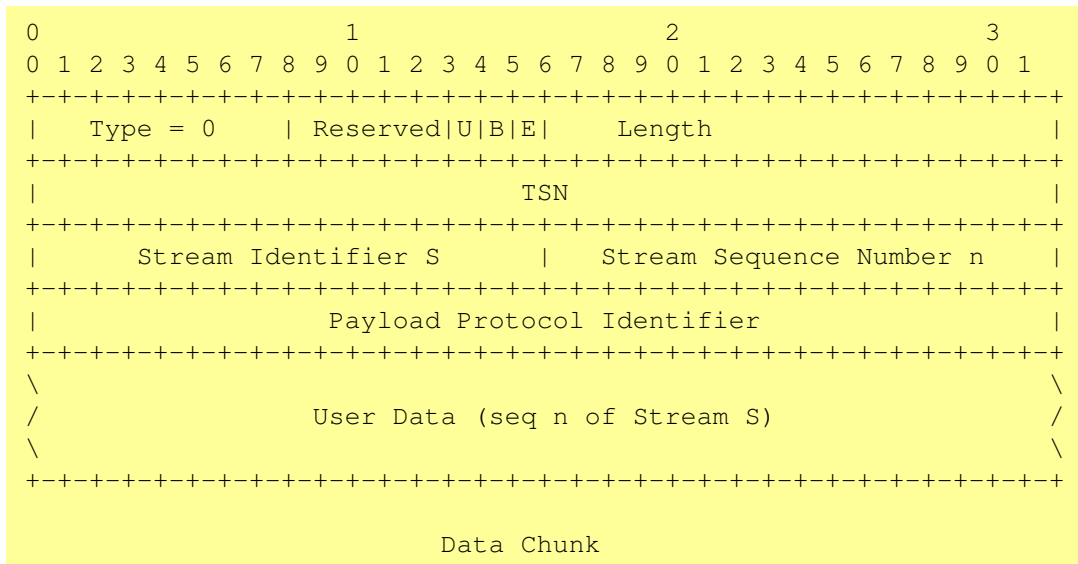
SCTP Common Header Format



SCTP Common Header Format

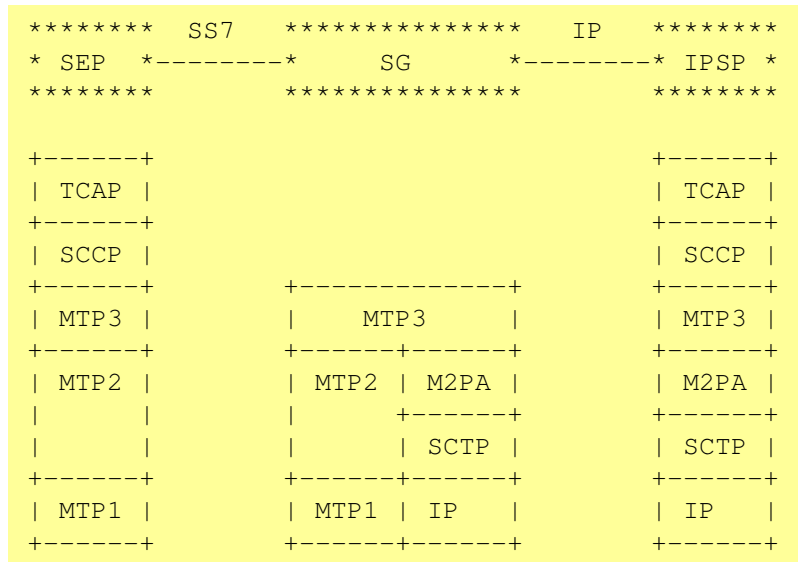
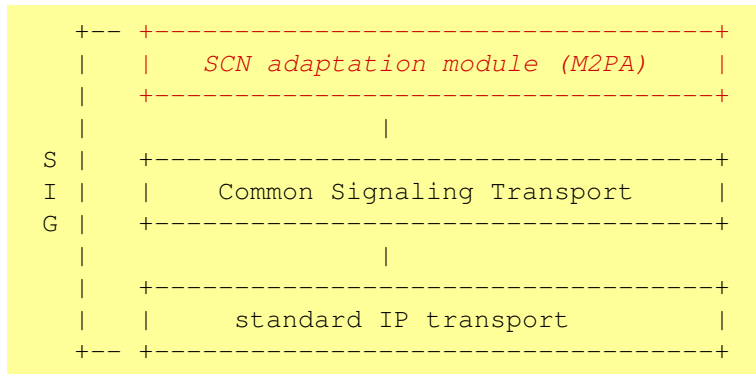
Implementação da arquitectura: SCTP

(Formato das mensagens)



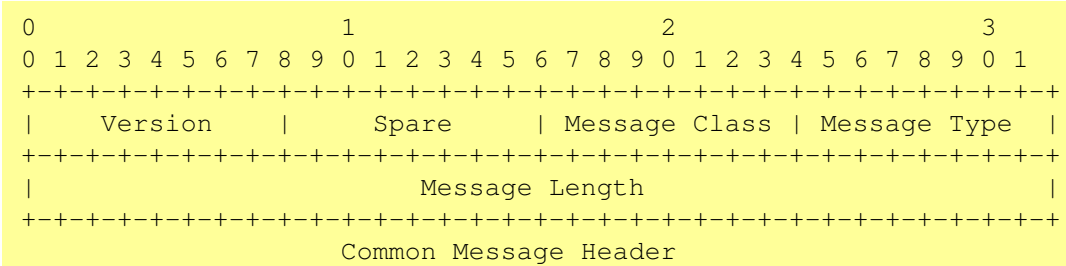
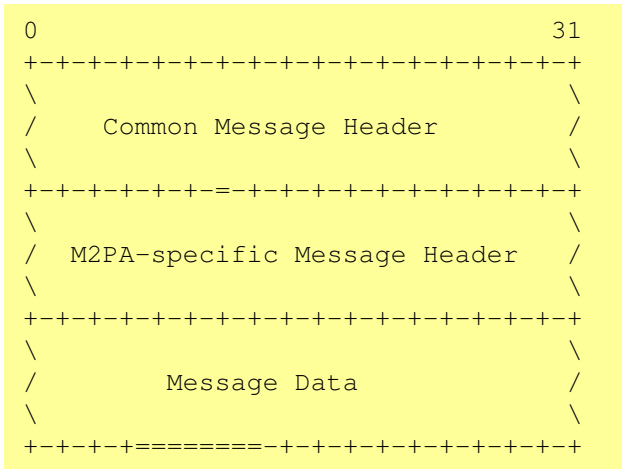
- *(U)nordered chunk*
- *(B)eginning fragment bit*
- *(E)nding fragment*
- *(T)ransmission (S)eq. (N)umber*
- *(S)tream Identifier*
- *Stream Sequence (N)umber*
 - *Numeração dentro do fluxo*
- *Payload Protocol Identifier*
 - *Identifica o utilizador (protocolo)*
 - *Não utilizado pelo SCTP*

Implementação da arquitectura: *M2PA* (*MTP2 - User Peer-to-Peer Adaptation Layer*)



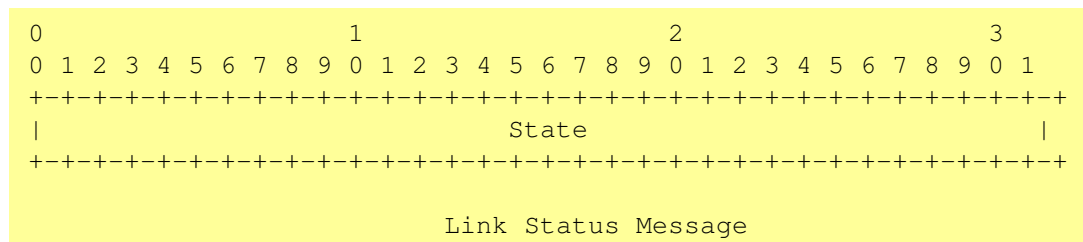
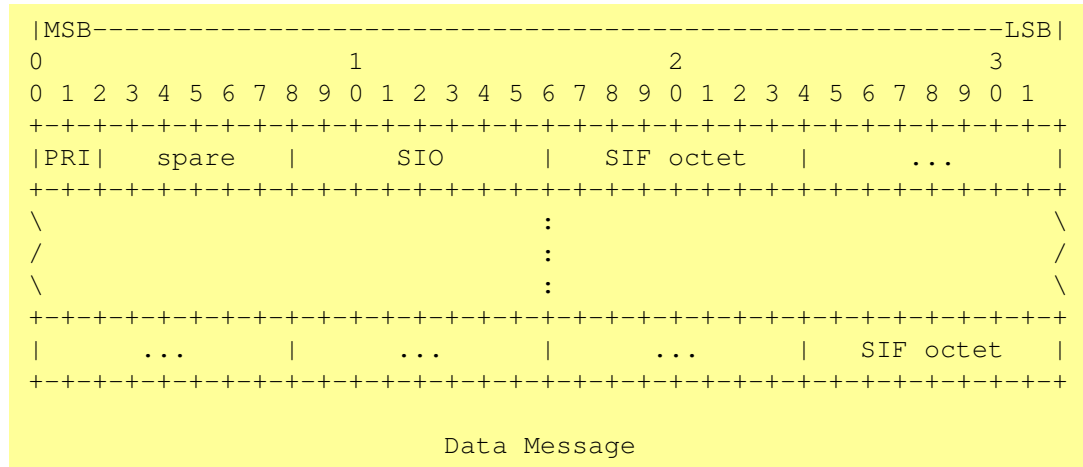
- O M2PA disponibiliza ao MPT3 uma interface e serviços idênticos aos do MTP2
 - Todos os serviços da interface MPT3/MPT2
- Utiliza os serviços do STCP.
 - Uma associação STCP representa um link SS7
- Comunicação peer-peer MPT2
 - MSU = User Data
 - LSSU = Link Status
 - FISU (não existem no M2PA)
 - MSUs são confirmadas através de mensagens User Data vazias
- Funcionalidades disponibilizadas
 - Recuperação de dados (MTP3 changeover)
 - Reporting Link Status ao MPT3
 - Procedimento de Processor Outage
 - Procedimento de Link Alignment

Protocolo M2PA: Formato das mensagens



- *Common Msg Header*
 - *Version: MP2A 1.0 (1)*
 - *Message Class: M2PA (11)*
 - *Message Type*
 - User Data (1)*
 - Link Status (2)*
- *M2PA-specific Header*
 - *FSN: Forward Seq.*
 - *BSN Backward Seq*

Protocolo M2PA: Formato das mensagens



- *Data Message*

- *PRI: Priority*
- *SO: Service Info Octet*
- *SIF: Signaling Info Field*

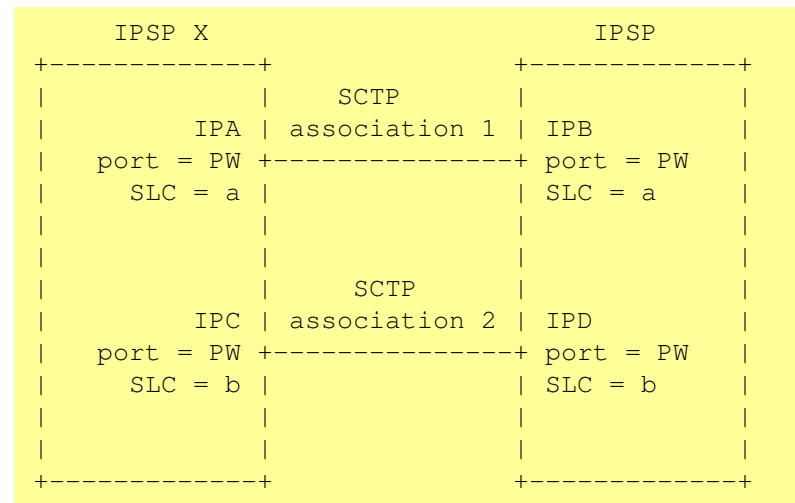
- ***Não contém:***

- Flag,
- BSN, FSN, BIB, FIB
- Length Indicator (LI)
- Check Bits (CK)

- *Link Status Message*

- 1 Alignment
- 2 Proving Normal
- 3 Proving Emergency
- 4 Ready
- 5 Processor Outage
- 6 Processor Recovered
- 7 Busy
- 8 Busy End
- 9 Out of Service (OOS)

Protocolo M2PA: Entidades MTP e STCP

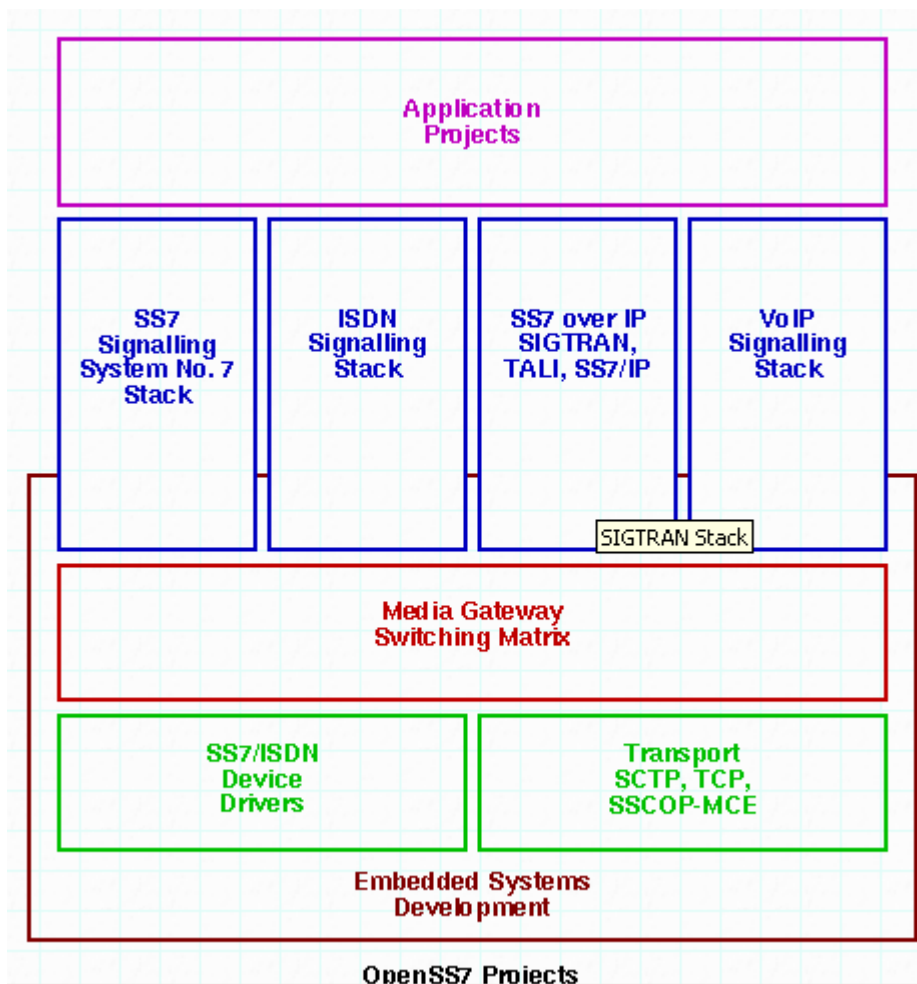


Association	IPSP X		IPSP Y		SLC
	IP address	Port	IP address	Port	
1	IPA	PW	IPB	PW	a
2	IPC	PW	IPD	PW	b

O projecto OpenSS7

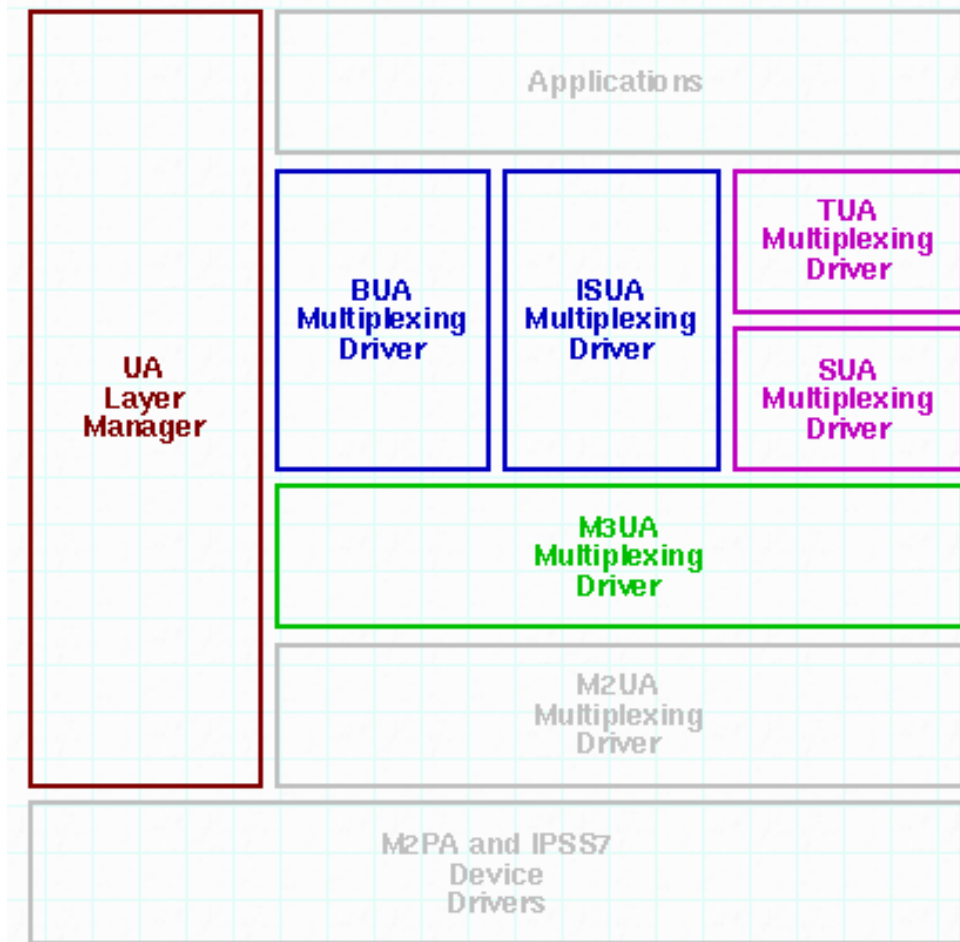
- O projecto OpenSS7 é:
 - Biblioteca de fonte aberta (*opensource*) implementando a pilha protocolar do SS7 e SIGTRAN para Linux.
 - Encontra-se disponível em <http://www.openss7.org>
- Motivação para o projecto OpenSS7:
 - Contribuir para uma utilização mais vasta do SS7, através de:
 - Uma solução de baixo custo (código disponibilizado, licença GPL).
 - Implementação simples da pilha SS7 e de fácil integração
 - Baseada em arquitecturas e interfaces abertas (e.g., STREAMS)
 - Promoção da colaboração no desenvolvimento/certificação Sw.

Arquitetura do OpenSS7



- Conjunto de módulos independentes do transporte
- Utilização de OSI STREAMS
 - Conjunto de primitivas comuns (utilizadas em todas as interfaces)
- Pilhas protocolares para SS7:
 - Pilha protocolar SS7
 - **SS7 sobre IP (SIGTRAN)**

OpenSS7 (*SS7 over IP e SIGTRANS*)



Implementação do SIGTRAN no OpenSS7:

- ISUA
SS7 ISUP-User Adaptation Layer
- TUA
SS7 TCAP-User Adaptation Layer
- SUA
SS7 SCCP-User Adaptation Layer
- M3UA
SS7 MTP3-User Adaptation Layer
- M2UA
SS7 MTP2-User Adaptation Layer
- M2PA
SS7 MTP2-User Peer-to-Peer Adaptation Layer
(Implementado como *device driver*)

Exemplos de aplicações com OpenSS7

- *Asterix PBX* (<http://www.asteriskpbx.org/>)
 - Implementação GPL de um PBX em Sw para plataformas Linux
 - Directory, Call Conferencing, Interactive Voice Response, Call Queuing
 - Suporte para VoIP sem qualquer Hw adicional
- *Kannel* (<http://www.kannel.org>)
 - Implementação de um gateway para SMS e WAP
 - Acesso à SMSC através de SS7

OpenSS7 no laboratório de IGRS

- *Experimentação com o OpenSS7*
 - *Observação da implementação SIGTRAN*
 - *Encapsulamento das mensagens*
 - *Protocolos de diferentes níveis (STCP, M2PA, TUA)*
- *Utilização do OpenSS7*
 - *Escrita de uma pequena aplicação fazendo uso do OpenSS7*
 - *Observação das mensagens trocadas entre componentes*