SRv6 for Mobile User-Plane

draft-ietf-dmm-srv6-mobile-uplane-**01**

IETF101

S.Matsushima, C.Filsfils, M.Kohno, D.Voyer, C.Perkins, P.Camarillo

Comments since IETF100

- **Many positive comments from both On/Off-List**
  - Adopted as a WG document.
  - But with some complains mainly on readability, i.e: packet flow description.

- **Concerns on SRH**
  - SRH insertion or IPv6 encaps with SRH
  - Impact to size of overhead

- **Applicability for 5G architecture**
  - Need to be clear on which generation is the target

- **Functions covered versus existing User Plane protocol**
  - GTP-U Messages (e.g, End-Marker)
  - GTP-U Extension headers (e.g, 5G container)
  - IPv4/Ethernet PDU support

Summary of Updates from v00 to v01

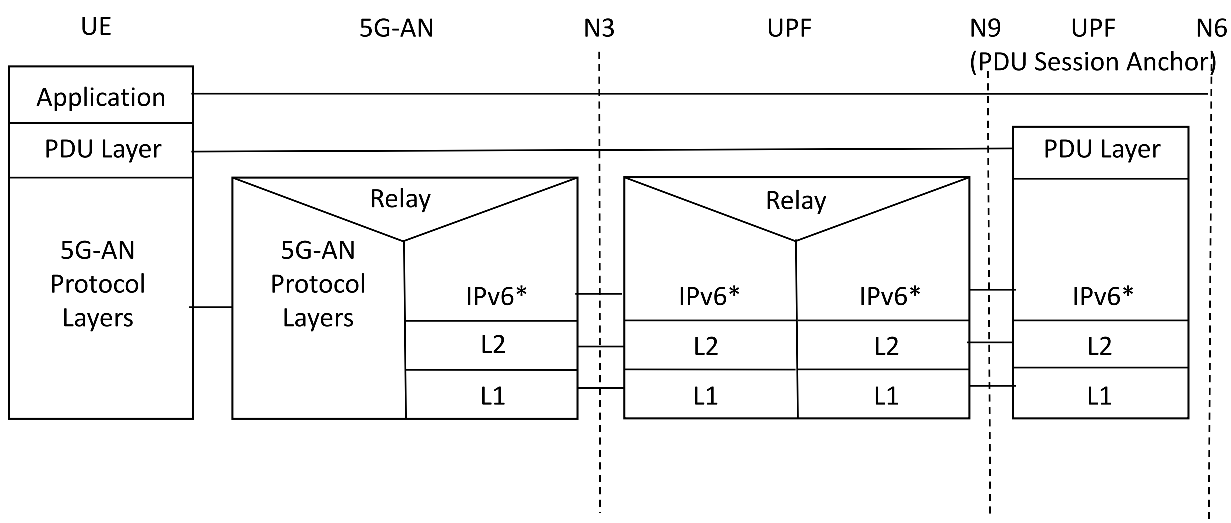
- **Reference architecture and terminology changes**
  - Refer to 5G arch (TS23.501), but not limited to only that.
  - Traditional mode (from Basic mode in v00)
  - Enhanced mode (from Aggregate mode in v00)
- **Uses IPv6 encapsulation with SRH by default**
  - … thereby supporting other types of PDU, e.g. IPv4, Ethernet, etc.,
- **Improved draft readability**
  - Added detailed packet flows, simplified use-cases.
- **Added interworking with IPv6/GTP**
  - New SR-GW functionality at UPF
  - No need for an uplink classifier

Next Steps

- **Extend function coverage**
  - To support GTP-U Messages and Extension headers, i.e, 5GS Container.
  - Ideally, GTP-U Extension header encodings are not modified. Needs further study.

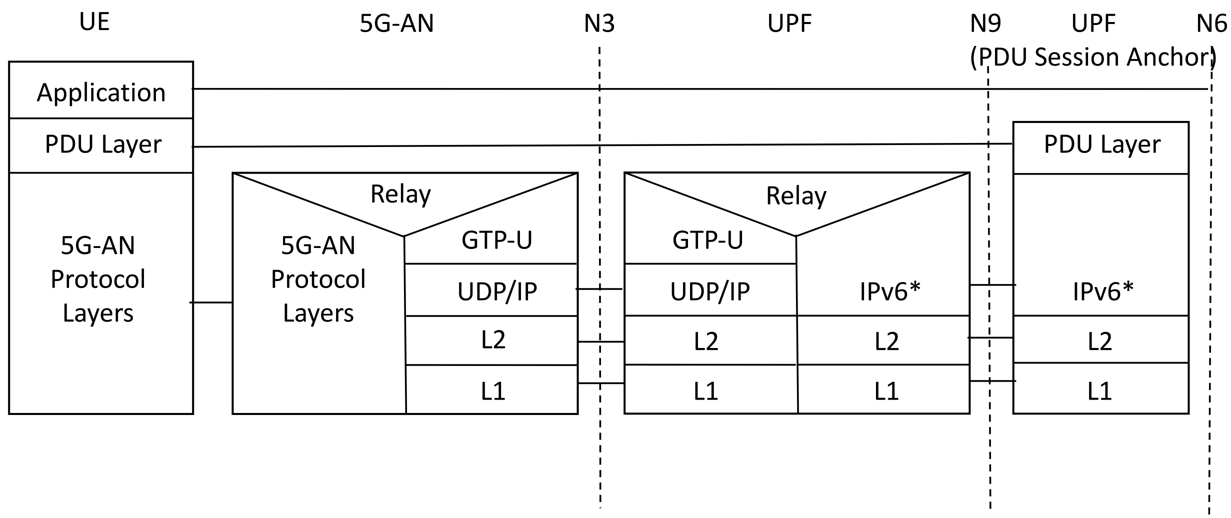
- **Examples with different UPFs**
  - Interaction with Uplink Classifier UPF and/or Branching Point UPF for use cases.
  - e.g, traffic measurement, lawful intercept, multi-homing, charging, etc.,

A View of 3GPP-friendly SRv6 UP Protocol Stack



*: IPv6 header + SRH (variable size: 1 SID = 16Bytes)
No SRH in traditional mode with just an IPv6 header (40Bytes)

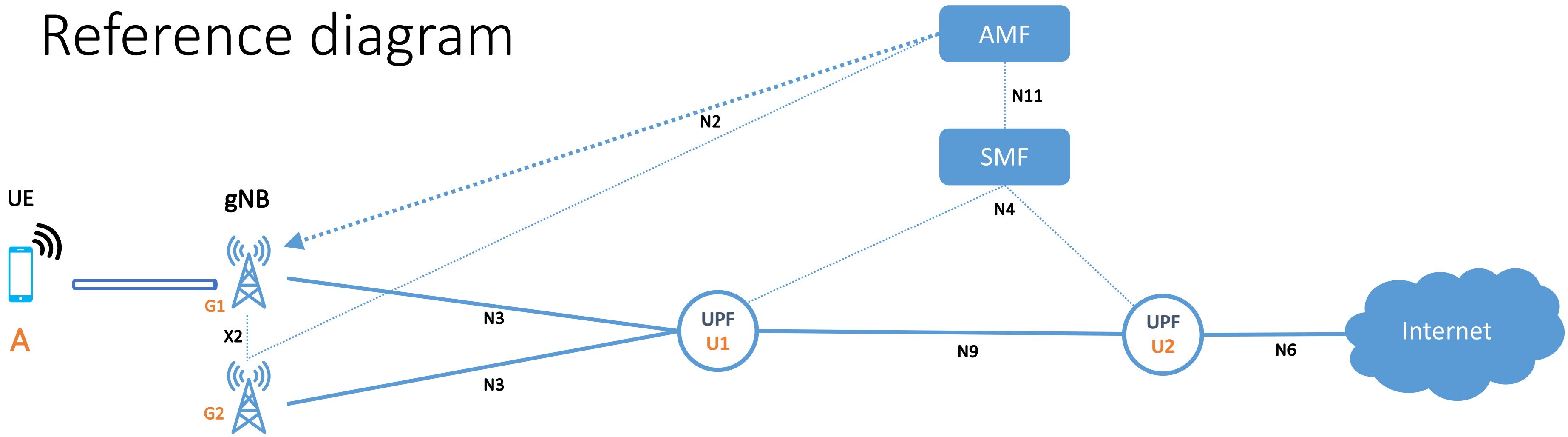
A View of 3GPP-friendly SRv6 UP Protocol Stack (gNB/N3 unchanged scenarios)



*: IPv6 header + SRH (variable size: 1 SID = 16Bytes)
No SRH in traditional mode with just an IPv6 header (40Bytes)

Traditional mode

Reference diagram

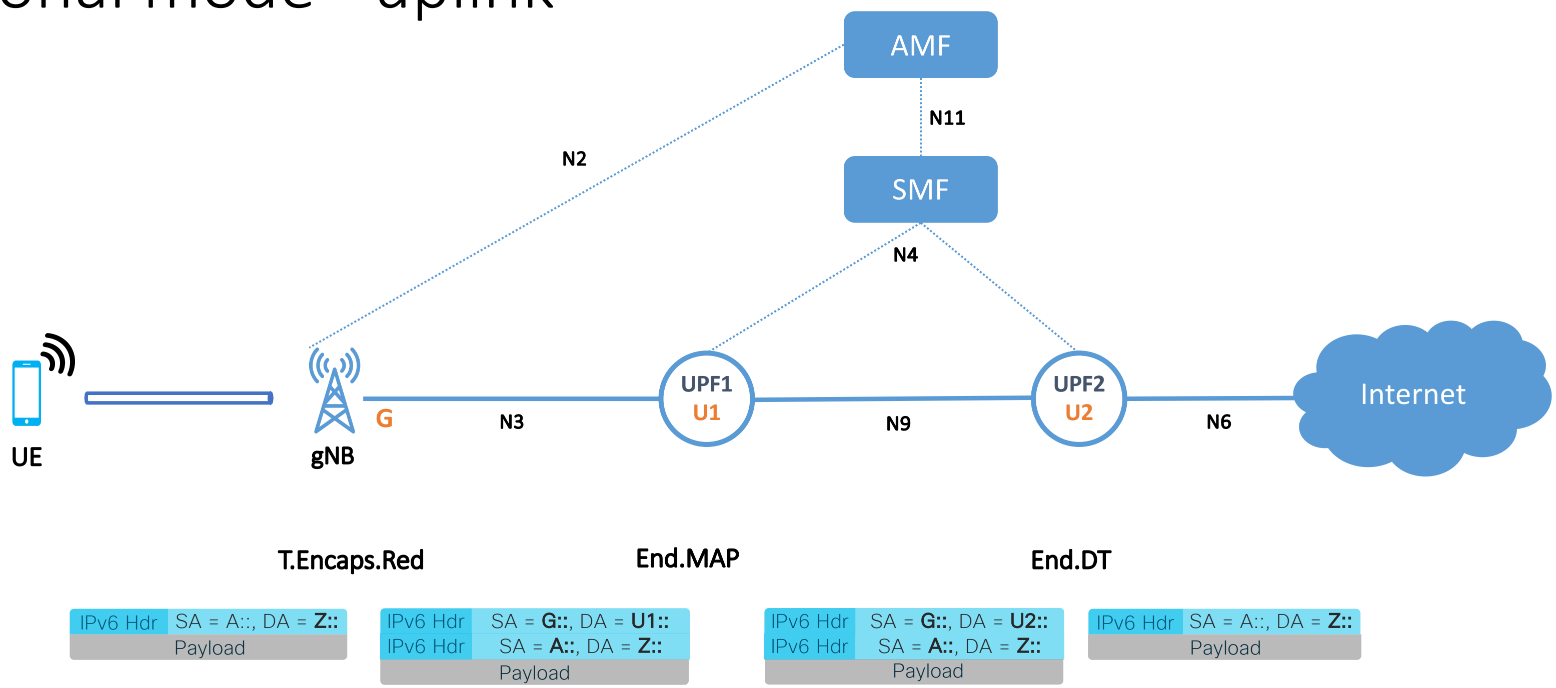


AMF: Access & Mobility Function
SMF: Session Management Function
gNB: 5G eNodeB (i.e., base station)
UPF: User Plane Function
N2, N3, N4, N6, N9, N11: 5G reference points (functional block interfaces)
X2: inter-base station reference point

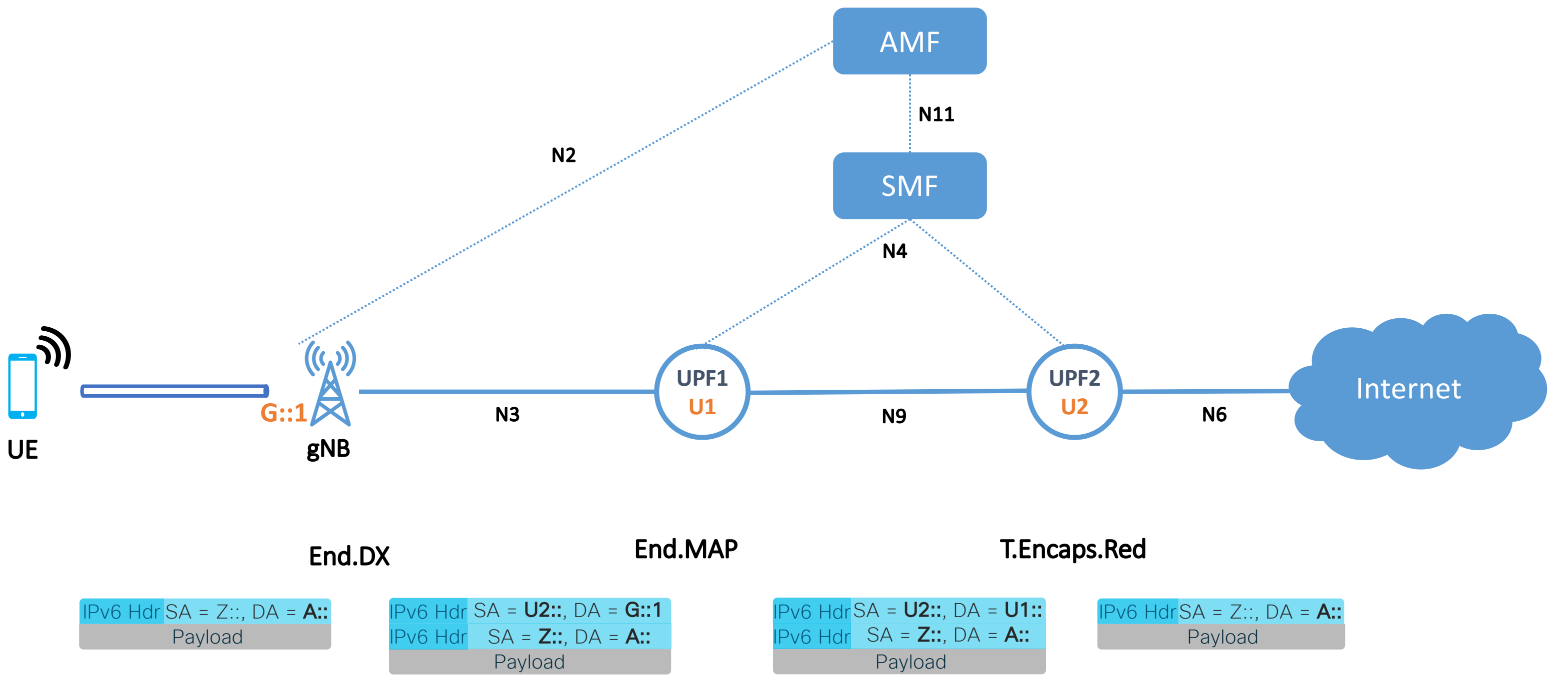
Traditional mode

- Mobile user-plane functions are the same ones as with GTP-U. It's just a data plane replacement.
- Equivalent with existing User Plane in terms of functionality.
- PDU sessions mapped 1-for-1 with a GTP-U tunnel. In this mode, mapped with SRv6 policy.
- gNB is SRv6 capable but from control plane viewpoint there's no change.
- Lower MTU overhead than GTP-U over IPv6/UDP!
  draft-dukes-spring-mtu-overhead-analysis-00

Traditional mode - uplink



Traditional mode - downlink



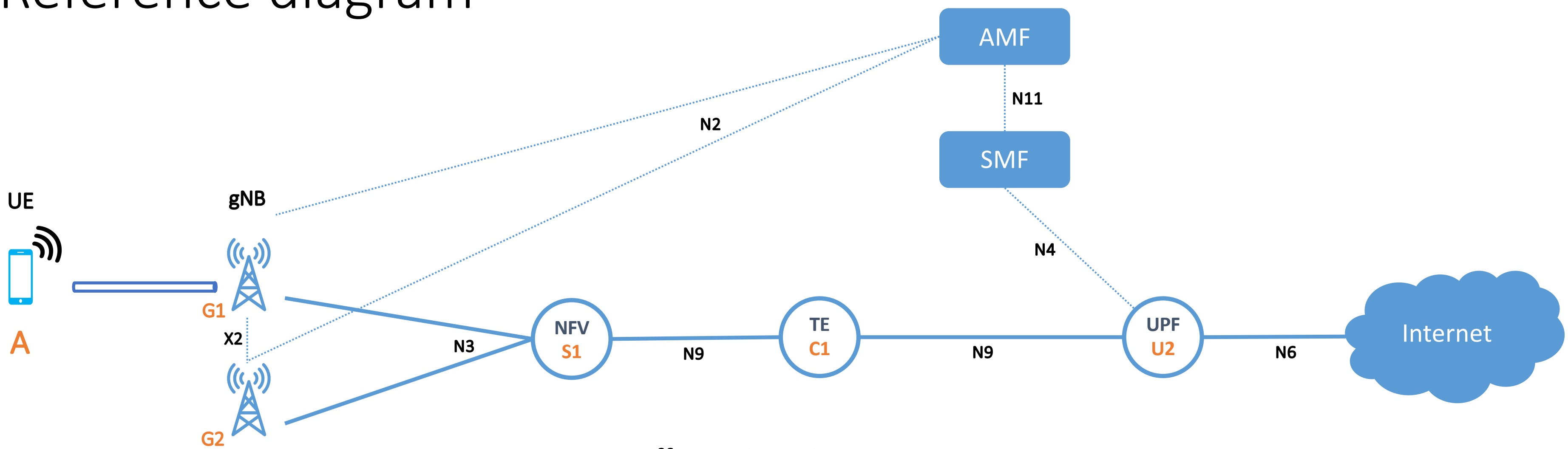
End.Map

- The “Endpoint function with SID mapping” (End.MAP) is used in several scenarios. Particulary in mobility, is is used in the UPF for
- When N receives a packet destined to S and S is a local End.Map SID, N does:
-
```
1.      look up the IPv6 DA in the mapping table
2.      update the IPv6 DA with the new mapped SID     ;; Ref1
3.      forward according to the new mapped SID
4.  ELSE  /* if S is NOT a local End.Map SID */
5.      Drop the packet

Ref1: SRH is NOT modified if it exists in the header.
```

Enhanced mode

Reference diagram



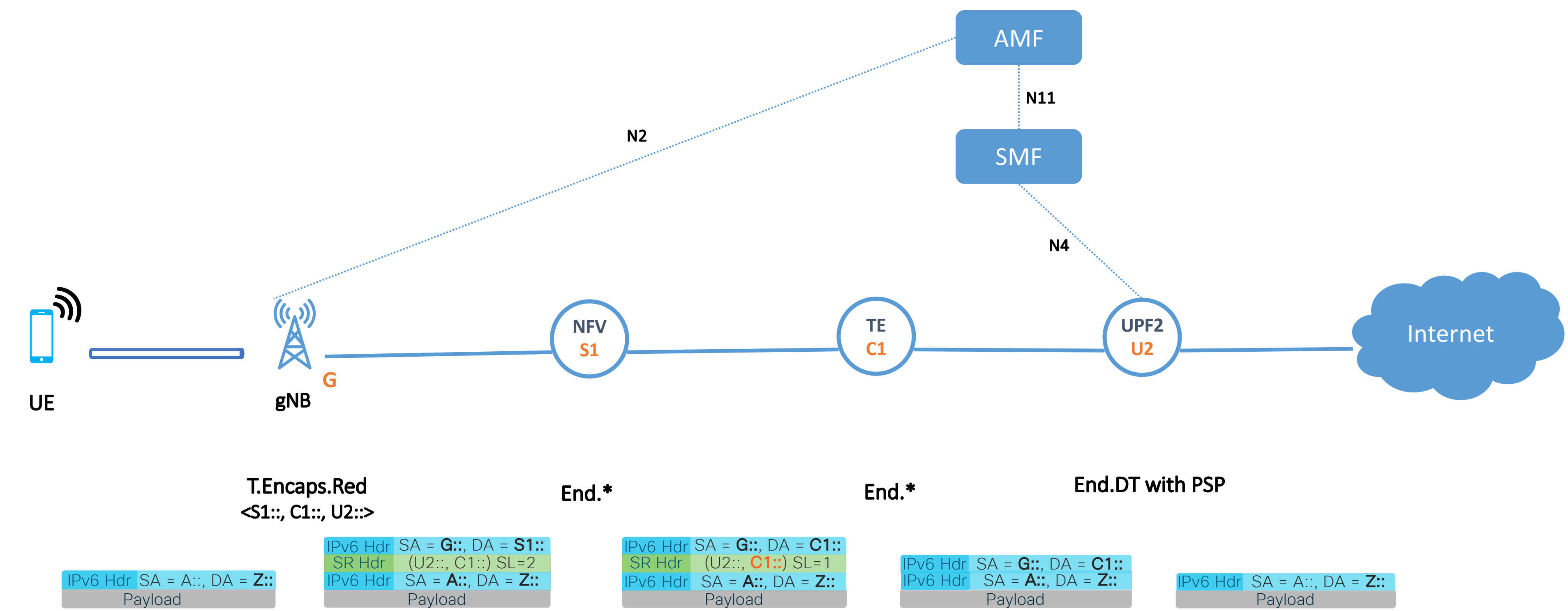
C1: Traffic Engineering
S1: Service function instance running on NFV platform

AMF: Access & Mobility Function
SMF: Session Management Function
gNB: 5G eNodeB (i.e., base station)
UPF: User Plane Function
N2, N3, N4, N6, N9, N11: 5G reference points (functional block interfaces)
X2: inter-base station reference point

Enhanced mode

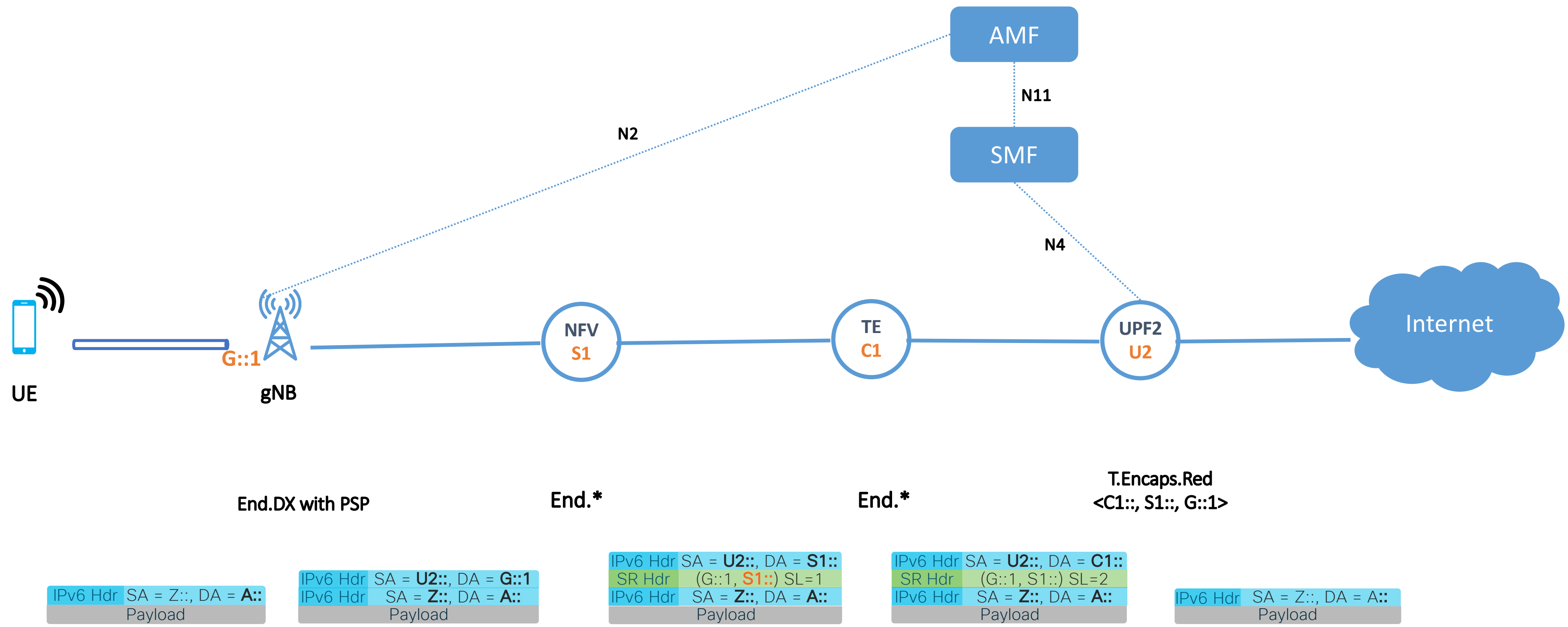
- Several UE share the same SR policy (and its SIDs)
- The SR policy includes Traffic Engineering(C1) and NFV(S1)
- The gNB control-plane (N2 interface) might, or might not be unchanged:
  - ➢ If unchanged, we signal a single IP address that the gNB resolves with PCEP, reverse DNS, LISP into a SID list
  - ➢ If changed, we signal a full SID list over the N2 interface

Enhanced mode - uplink



End.*: Appropriate SRv6 End function type for the purposes.

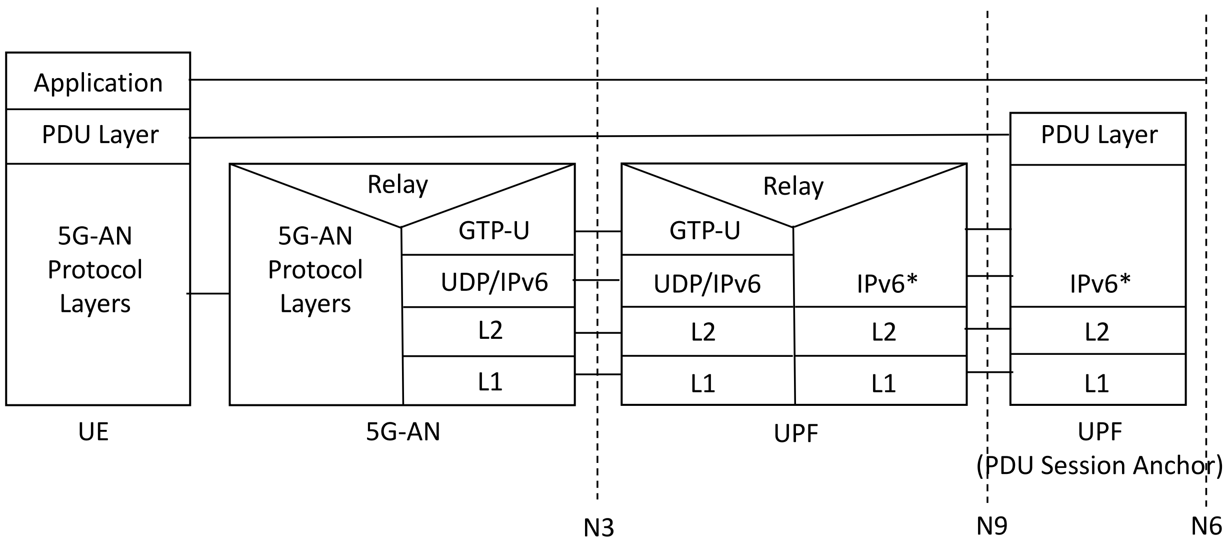
Enhanced mode - downlink



End.*: Appropriate SRv6 End function type for the purposes.

Enhanced mode with unchanged gNB IPv6 GTP behavior

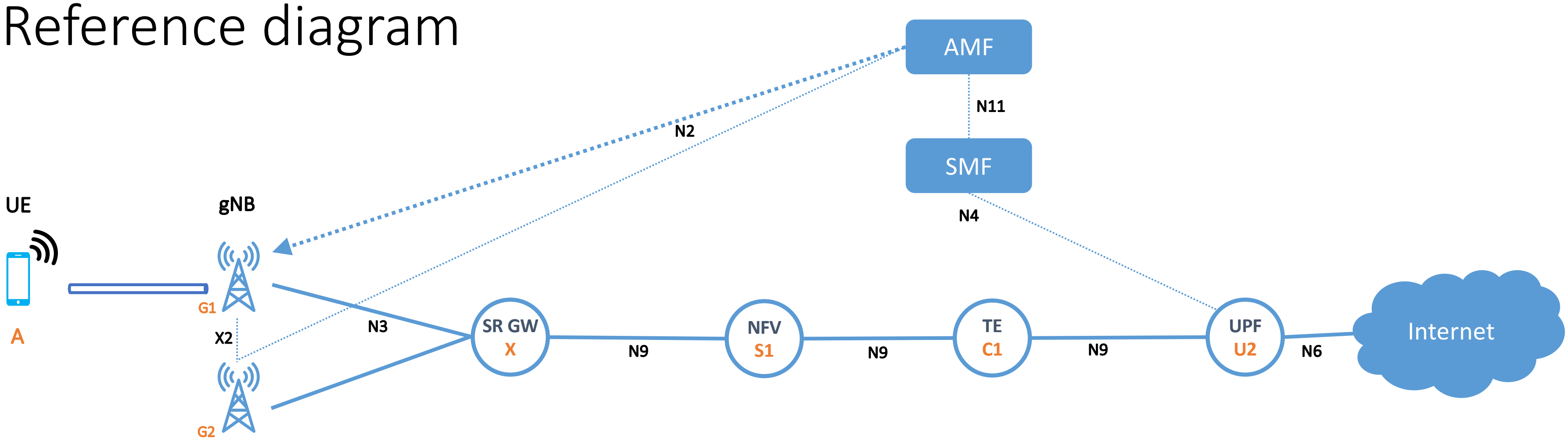
SRv6 (N3 unchanged – IPv6/GTP)



At the N3 the packet is IPv6/GTP, but IPv6 DA is an SRv6 segment. Routing is based on SRv6.

*: IPv6 header + SRH (variable size: 1 SID = 16Bytes)
No SRH in traditional mode with just an IPv6 header (40Bytes)

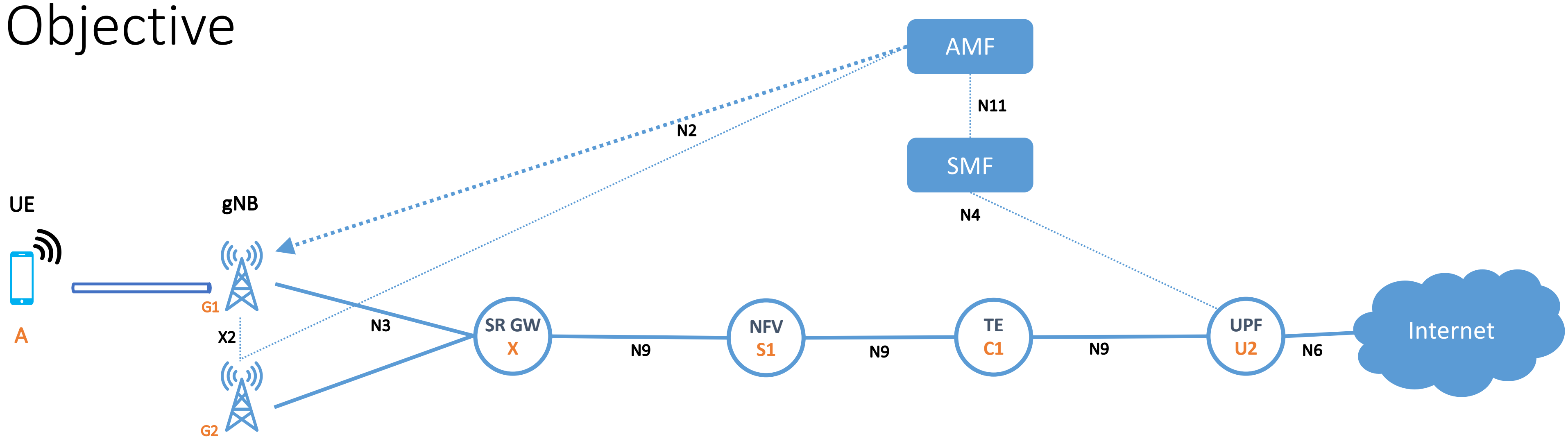
Reference diagram



C1: Traffic Engineering
S1: Service function instance running on NFV platform
X: GTP-U/SRv6 Interworking

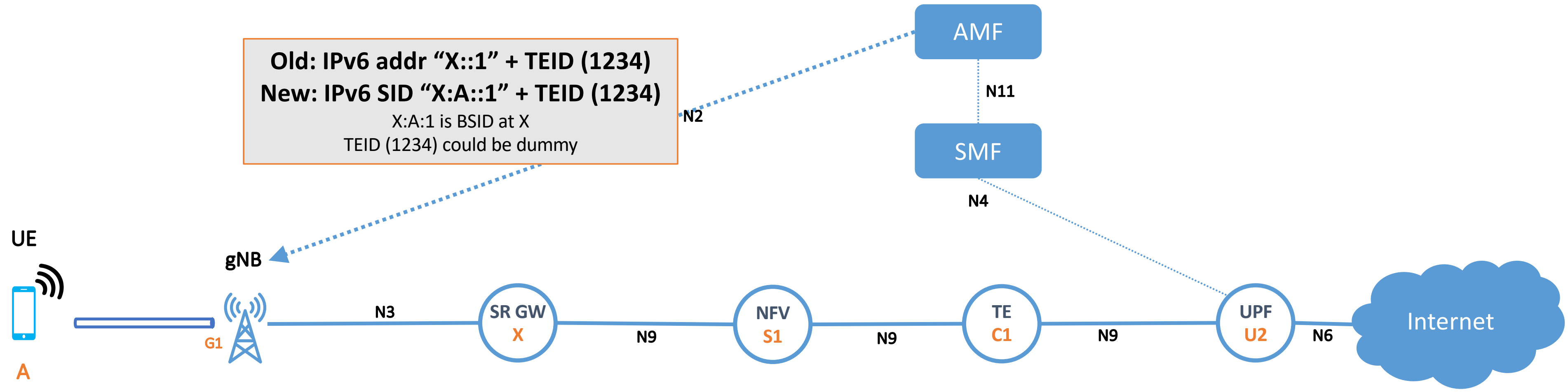
AMF: Access & Mobility Function
SMF: Session Management Function
gNB: 5G eNodeB (i.e., base station)
UPF: User Plane Function
N2, N3, N4, N6, N9, N11: 5G reference points (functional block interfaces)
X2: inter-base station reference point
SR GW: Segment Routing Gateway between GTP-U/IPv6 and SRv6

Objective

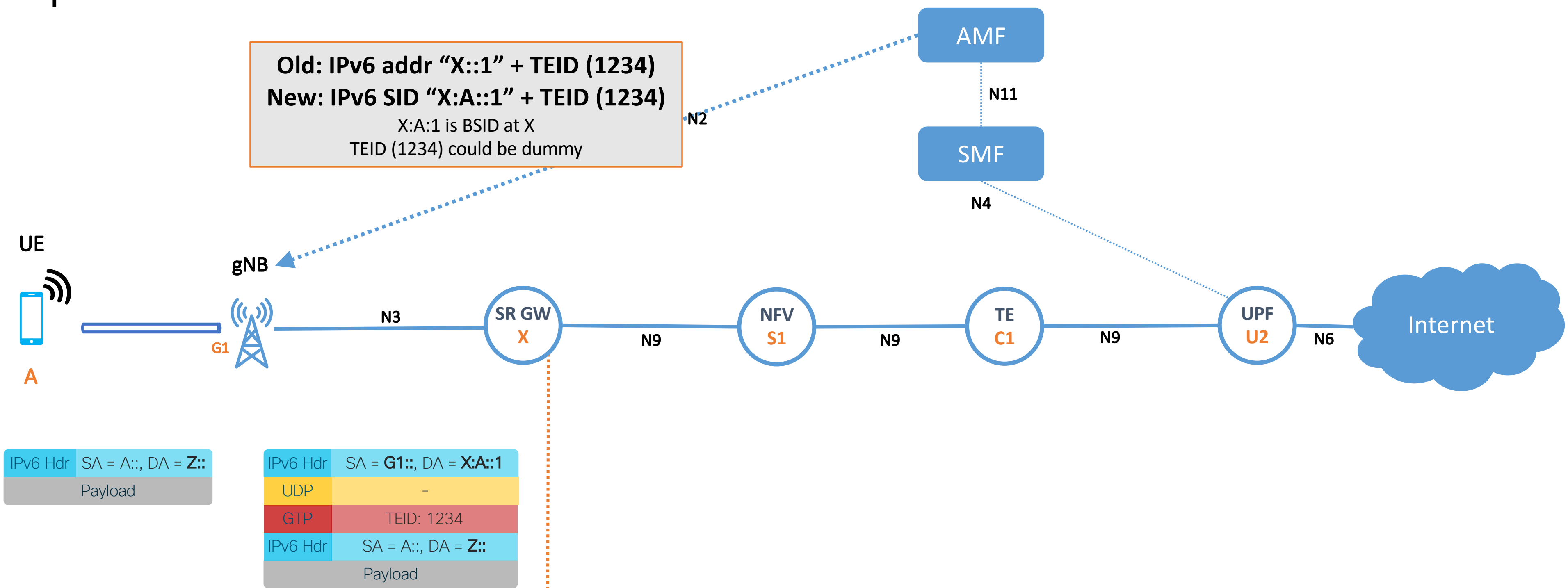


- SR GW to UPF U2 is SRv6 capable for the underlay, overlay and service chaining
- GTP-U endpoint IPv6 addresses of gNB and SR GW could be treated as SRv6 SID.
- No software changes in the gNB
- To achieve this we deploy an SR GW in between gNB and UPF (N3 interface)
  - Any SRv6 capable router on hardware or software.
- Applies to any kind of PDU session types

Uplink traffic

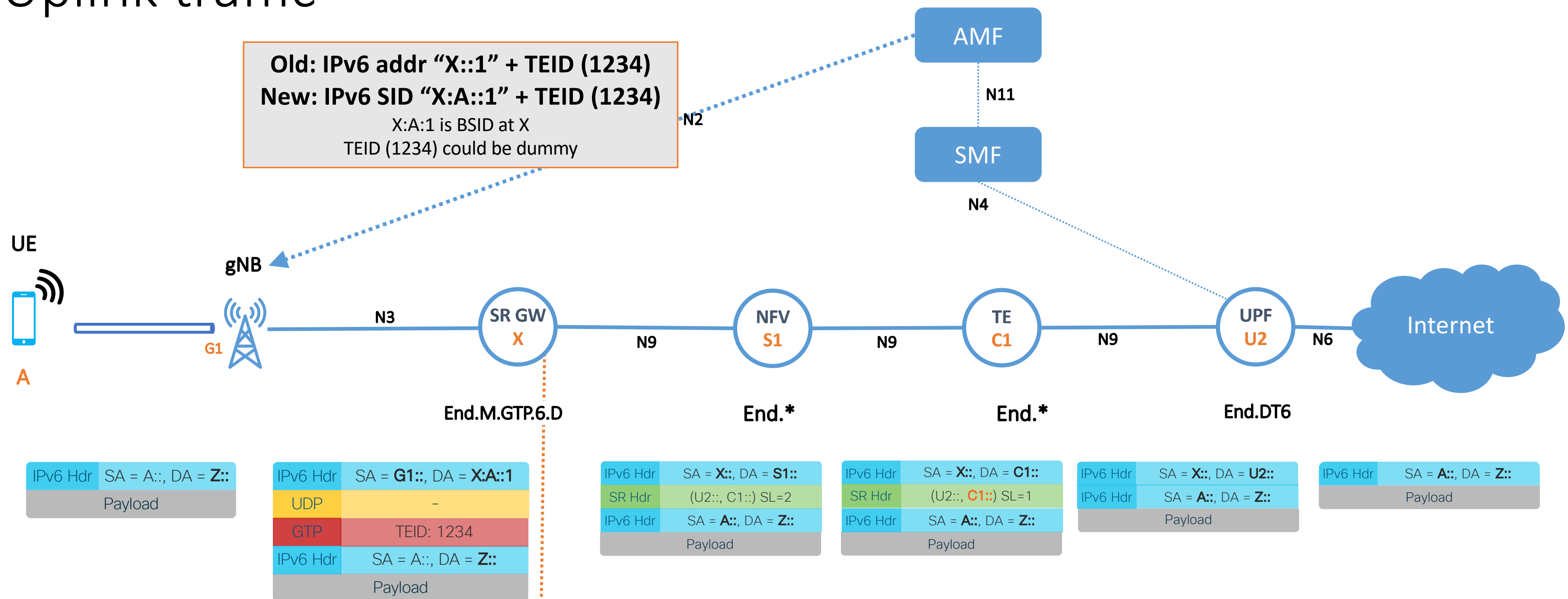


Uplink traffic



- SRGW advertise SID in LOC(64b of X) + Algorithm(8b of A and B) format :
  - X:A::/72 -> Low latency
  - X:B::/72 -> High bandwidth
- SRGW does End.M.GTP6.D function for X:A:: and X:B::.
- X:A::1 is a type of End.M.GTP6.D SID that pops the GTP header without lookup and pushes an outer header with SID list
- X:A:1 could be a flex-alg SID (end-to-end network slicing) for other SRv6 nodes.
- X:A::1 can be shared across UEs

Uplink traffic



- SRGW advertise SID in LOC(64b of X) + Algorithm(8b of A and B) format :
  - X:A::/72 -> Low latency
  - X:B::/72 -> High bandwidth
- SRGW does End.M.GTP6.D function for X:A:: and X:B::.
- X:A::1 is a type of End.M.GTP6.D SID that pops the GTP header without lookup and pushes an outer header with SID list
- X:A:1 could be a flex-alg SID (end-to-end network slicing) for other SRv6 nodes.
- X:A::1 can be shared across UEs

- Without modifying the N2 interface we are steering the UE packets to an SRGW along the designated algorithm path.

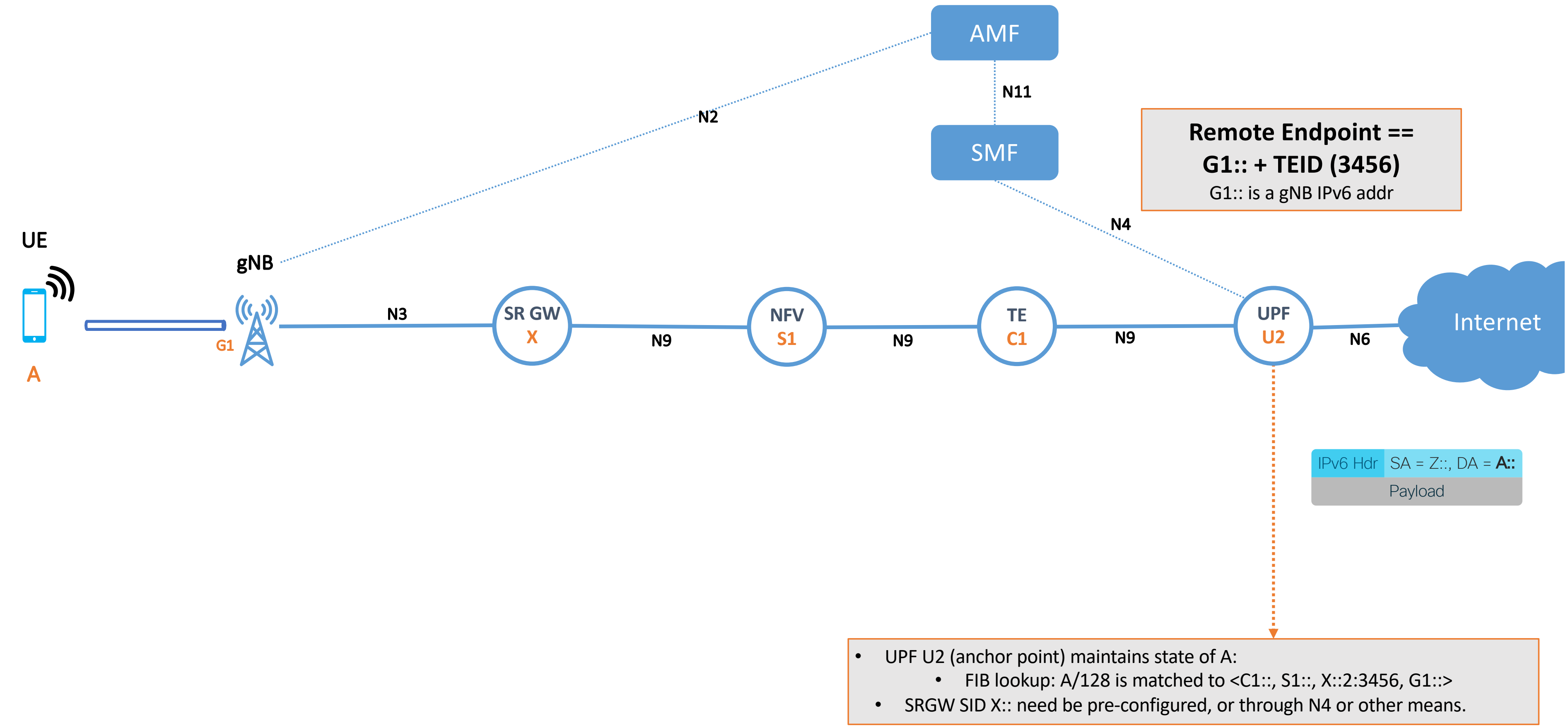
End.*: Appropriate SRv6 End function type for the purposes.

End.M.GTP6.D

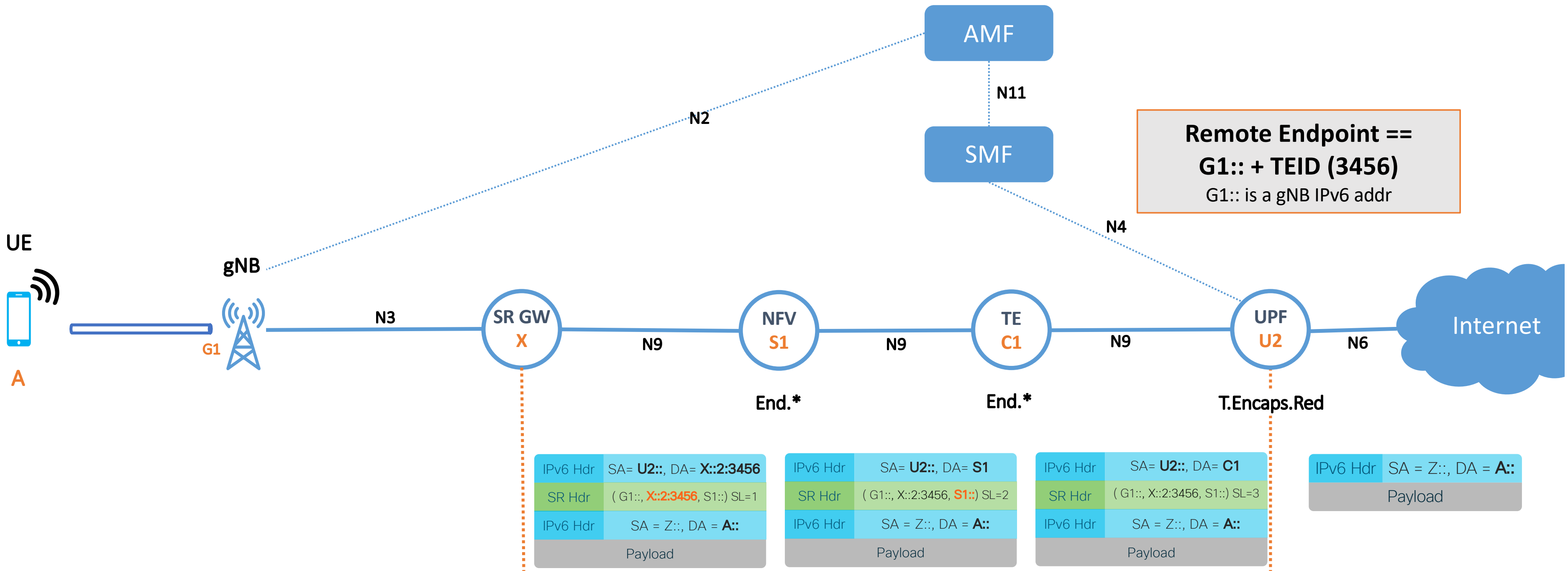
- The " Endpoint function with IPv6/GTP decapsulation into SR policy” (End.M.GTP6.D) is used to in interworking scenario the direction from legacy user-plane to SRv6 user- plane network.
- When N receives a packet destined to S and S is a local End.M.GTP6.D SID, N does:

```
•    1.   IF NH=UDP and UDP.DST_PORT=GTP
•    2.      pop IP, UDP and GTP headers
•    3.      push an outer IPv6 header with its own SRH
•    4.      set the outer IPv6 SA to A
•    5.      set the outer IPv6 DA to the first segment of the SRv6 Policy
•    6.      forward according to the first segment of the SRv6 Policy
•    7.   ELSE
•    8.      drop the packet
```

Downlink traffic



Downlink traffic

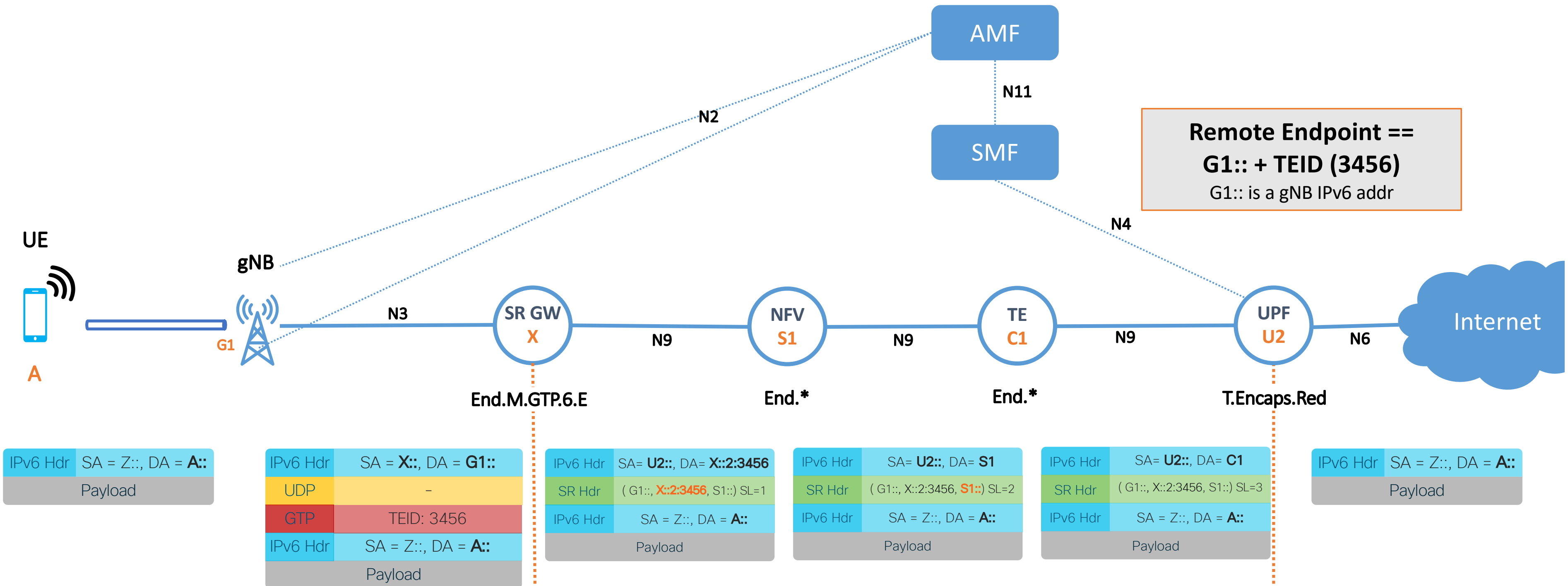


- SRGW does End.M.GTP6.E function.
- X::2 is an SRv6 End.M.GTP6.E SID.
- Removes SRH, adds UDP and GTP with TEID received in SID ARGs
- IPv6 DA is the last segment of the SRH
- Scales (no state per UE in SR gateway)
- gNB could configure an IPv6 addresses per network slice.
  - G1::1 -> Low latency
  - G1::2 -> High bandwidth

- UPF U2 (anchor point) maintains state of A:
  - FIB lookup: A/128 is matched to <C1::, S1::, X::2:3456, G1::>
- SRGW SID X:: need be pre-configured, or through N4 or other means.

End.*: Appropriate SRv6 End function type for the purposes.

Downlink traffic



- SRGW does End.M.GTP6.E function.
- X::2 is an SRv6 End.M.GTP6.E SID.
- Removes SRH, adds UDP and GTP with TEID received in SID ARGs
- IPv6 DA is the last segment of the SRH
- Scales (no state per UE in SR gateway)
- gNB could configure an IPv6 addresses per network slice.
    - G1::1 -> Low latency
    - G1::2 -> High bandwidth

- UPF U2 (anchor point) maintains state of A:
    - FIB lookup: A/128 is matched to <C1::, S1::, X::2:3456, G1::>
- SRGW SID X:: need be pre-configured, or through N4 or other means.

End.*: Appropriate SRv6 End function type for the purposes.

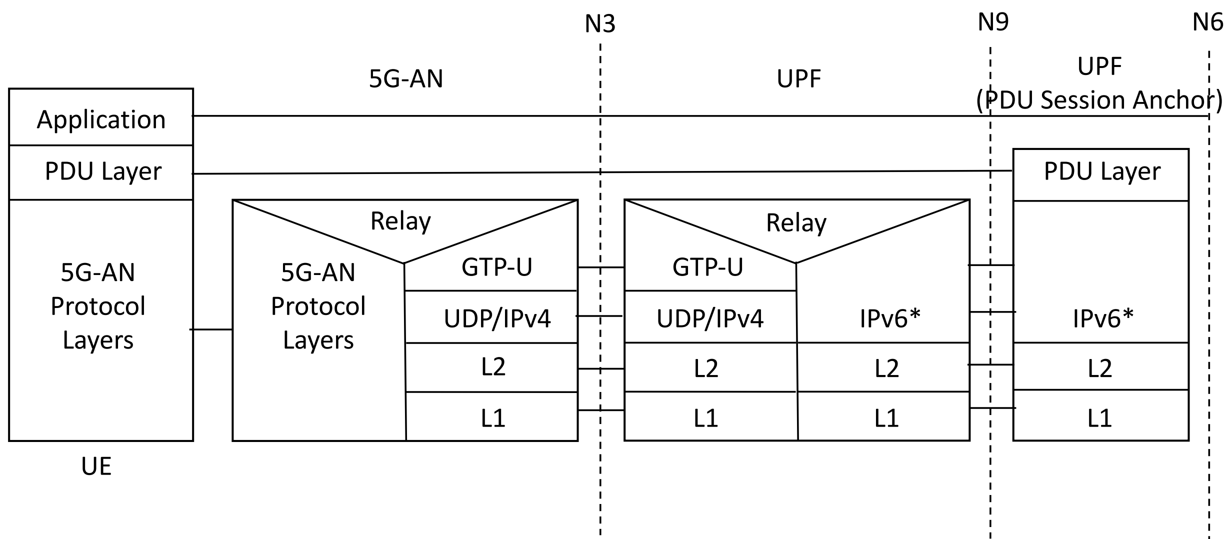
End.M.GTP6.E

- The " Endpoint function with encapsulation for IPv6/GTP tunnel” (End.M.GTP6.E) is used in interworking scenario for the direction from SRv6 user-plane to legacy user- plane network.
- When interworking node N receives a packet P destined to S and S is a local End.RAN SID, N does:

```
 1.   IF NH=SRH & SL = 1 THEN                  ;; Ref1
 2.      decrement SL
 3.      store SRH[SL] in variable new_DA
 4       store TEID in variable new_TEID      ;; Ref2
 5.      pop the (outer) IPv6 header and its extension headers
 6.      push an IPv6 header, a UDP header and a GTP-U header
 7.      set the IPv6 DA to new_DA
 8.      set the GTP_TEID to new_TEID
 9.      lookup the new_DA and forward the packet accordingly
10. ELSE
11.   drop the packet
Ref1: An End.M.GTP6.E SID must always be the penultimate SID.
Ref2: TEID is extracted from the argument space of the current SID.
```

Enhanced mode with unchanged gNB IPv4 GTP behavior

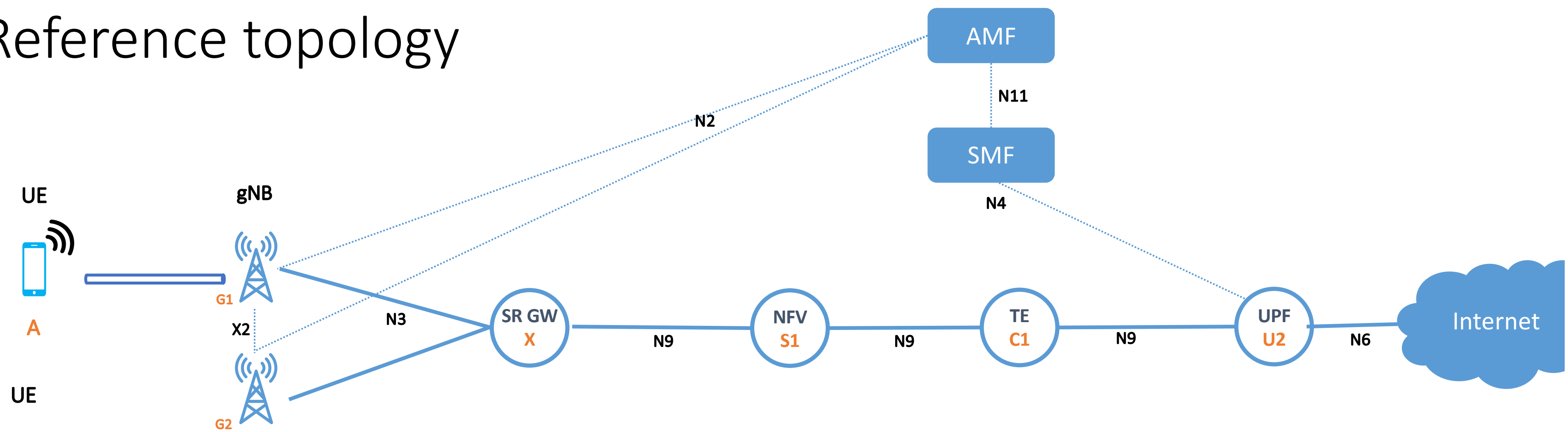
SRv6 (N3 unchanged – IPv4/GTP)



At the N3 the packet is IPv4/GTP.

*: IPv6 header + SRH (variable size: 1 SID = 16Byte)
No SRH in traditional mode with just an IPv6 header (40Byte)

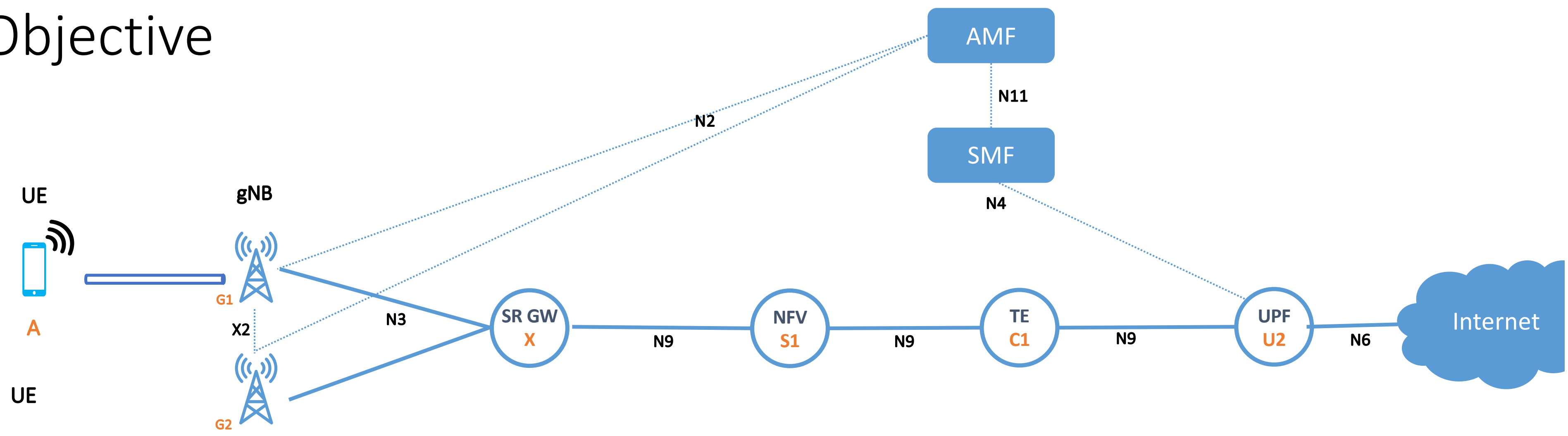
Reference topology



C1: Traffic Engineering
S1: Service function instance running on NFV platform
X: GTP-U/SRv6 Interworking

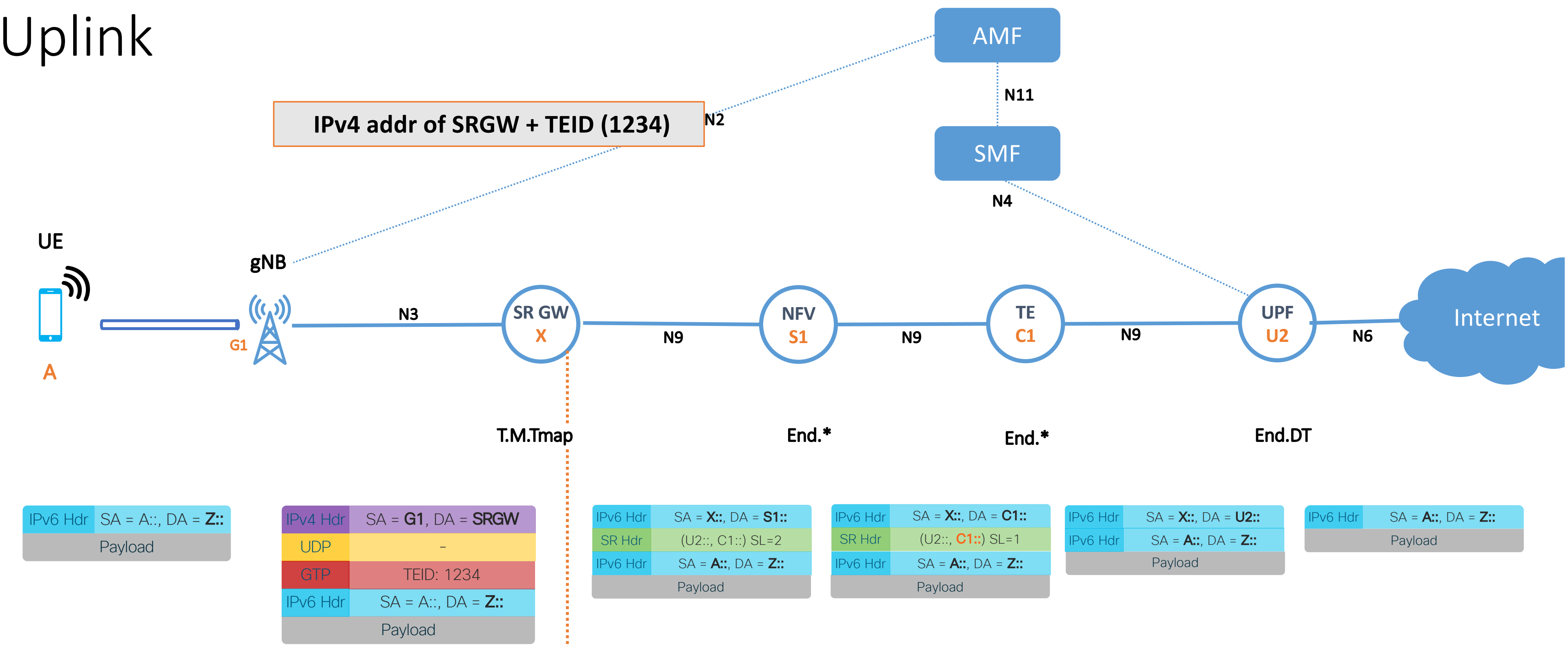
AMF: Access & Mobility Function
SMF: Session Management Function
gNB: 5G eNodeB (i.e., base station)
UPF: User Plane Function
N2, N3, N4, N6, N9, N11: 5G reference points (functional block interfaces)
X2: inter-base station reference point
SR GW: Segment Routing Gateway between GTP-U/IPv4 and SRv6

Objective



- SR GW to UPF U2 is SRv6 capable for the underlay, overlay and service chaining
- GTP-U endpoint of gNB and SR GW addresses on N3 is IPv4.
- No software changes in the gNB
- To achieve this we deploy an SR GW in between gNB and UPF (N3 interface)
  - Any SRv6 capable router on hardware or software.
- Applies to any kind of PDU session types

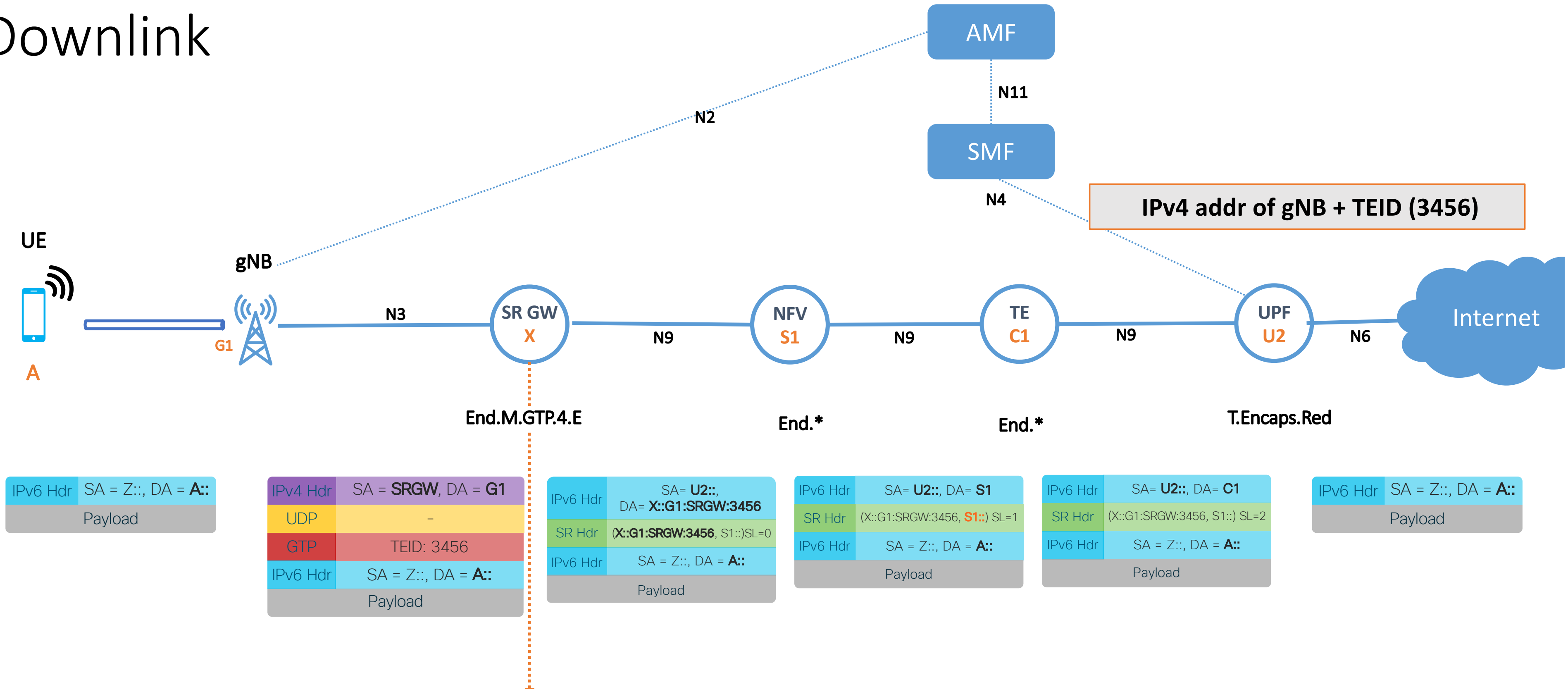
Uplink



- SRGW does T.M.Tmap function.
- SRGW is a UPF Uplink Classifier. Coexist with current UPFs.
- Steers traffic into SR policy.

End.*: Appropriate SRv6 End function type for the purposes.

Downlink



- X::G1:SRGW:3456is an SRv6 End.M.GTP4.E function
- Removes SRH, adds IPv4/UDP and GTP
- IPv4 SA, DA and GTP TEID are part of the SID's arguments
- Scales (no state per UE in SR gateway)

End.*: Appropriate SRv6 End function type for the purposes.

End.M.GTP4.E

- The “Endpoint function with encapsulation for IPv4/GTP tunnel” (End.M.GTP4.E) is used to the direction from SRv6 user-plane to legacy user-plane network.
- When interworking node N receives a packet destined to S and S is a local End.M.GTP4.E SID, N does:

```
+----------------------+-------+-------+-------+
|  SRGW-IPv6-LOC-FUNC  |IPv4DA |IPv4SA |TUN-ID |
+----------------------+-------+-------+-------+
       128-a-b-c           a       b       c

          End.M.GTP4.E SID Encoding
```

```
1. IF NH=SRH & SL > 0 THEN
2.    decrement SL
3.    update the IPv6 DA with SRH[SL]
4.    push header of TUN-PROTO with tunnel ID from S
5.    push outer IPv4 header with SA, DA from S
6. ELSE
7.    Drop the packet
Ref1: TUN-PROTO indicates target tunnel type.
```

T.M.Tmap

- The "Transit with mobile tunnel decapsulation and map to an SRv6 policy function (T.M.Tmap for short) is used to the direction from legacy user-plane to SRv6 user-plane network.
- When interworking node N receives a packet destined to a SRGW IPv4 address, N does:

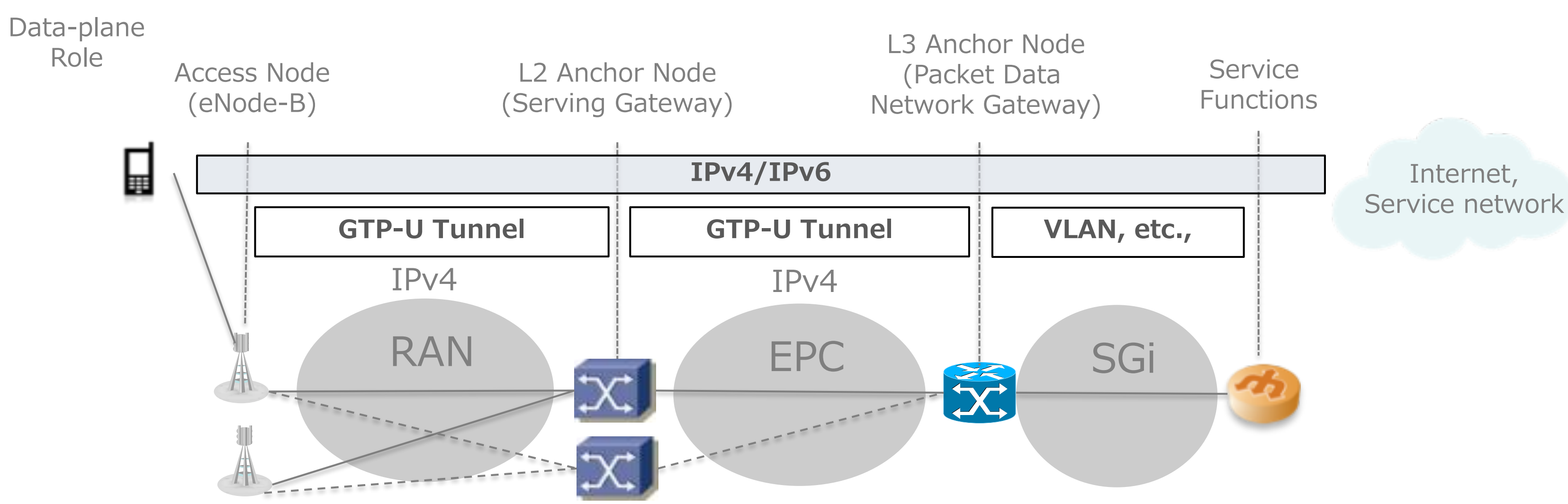
```
• 1.  IF P.PLOAD == TUN-PROTO THEN    ;; Ref1
• 2.     pop the outer IPv4 header and tunnel headers
• 3.     copy IPv4 DA, SA, TUN-ID to form SID B with SRGW-IPv6-Prefix ;; embedding IPv4 DA/SA/TEID in a SID could be an option.
• 4.     encapsulate the packet into a new IPv6 header ;; Ref2, Ref2bis
• 5.     set the IPv6 DA = B
• 6.     forward along the shortest path to B
• 7.  ELSE
• 8.     Drop the packet
```

- Ref1: P.PLOAD and T.PLOAD represent payload protocol of the receiving packet, and payload protocol of the tunnel respectively.
- Ref2: The received IPv6 DA is placed as last SID of the inserted SRH.
- Ref2bis: The SRH is inserted before any other IPv6 Routing Extension

Appendix

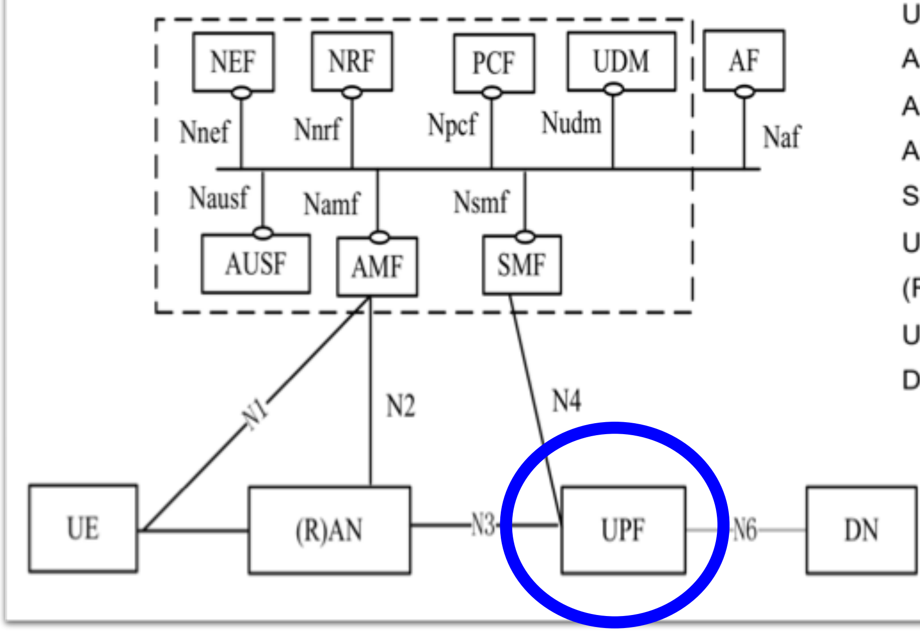
A Current Mobile Network Example

- **Well fragmented to RAN, EPC and SGi.** <- Redundancies lessen TCO
- **Per-session tunnel creation and handling.** <- Can be scaled up but costy
- **Non-optimum data-path.** <- Hard to meet Apps reqs



3GPP Rel-15 Architecture (5G Phase.1)

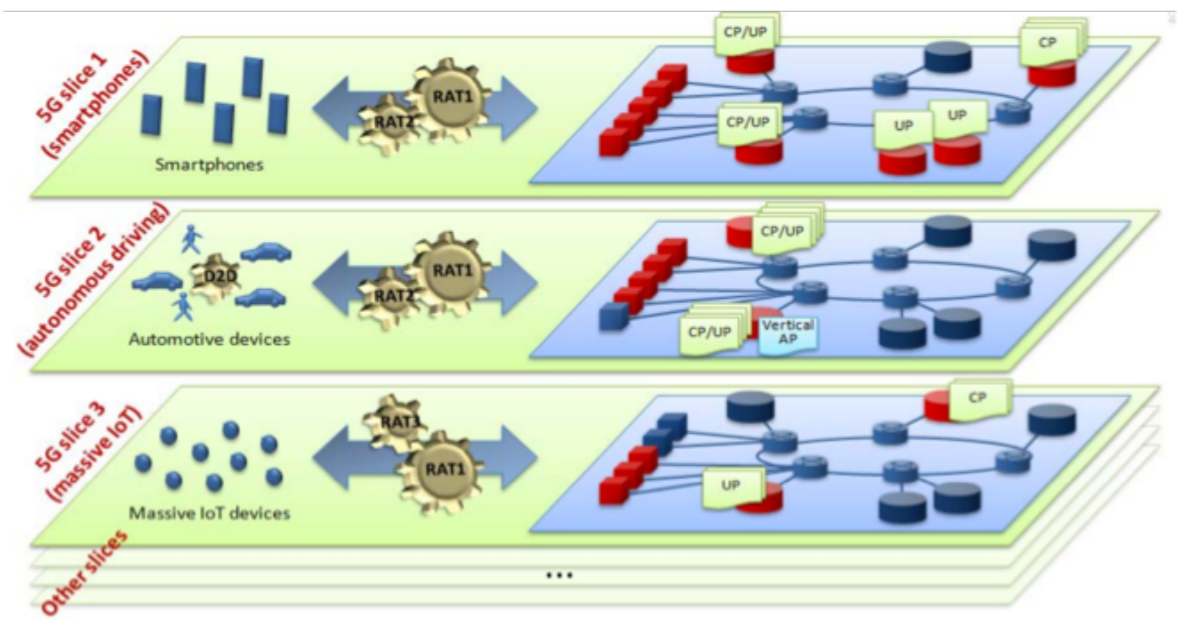
Dramatically Simplified, Why?



NEF	Network Exposure Function
NRF	Network Repository Function
PCF	Policy Control Function
UDM	Unified Data Management
AF	Application Function
AUSF	Authentication Server Function
AMF	Access &Mobility Management Function
SMF	Session Management Function
UE	User Equipment
(R)AN	(Radio) Access Network
UPF	User Plane Function
DN	Data Network

Generic Expectations for 5G Networks

U-Plane must be simplified because to meet Complicated Optimizations



eMBB

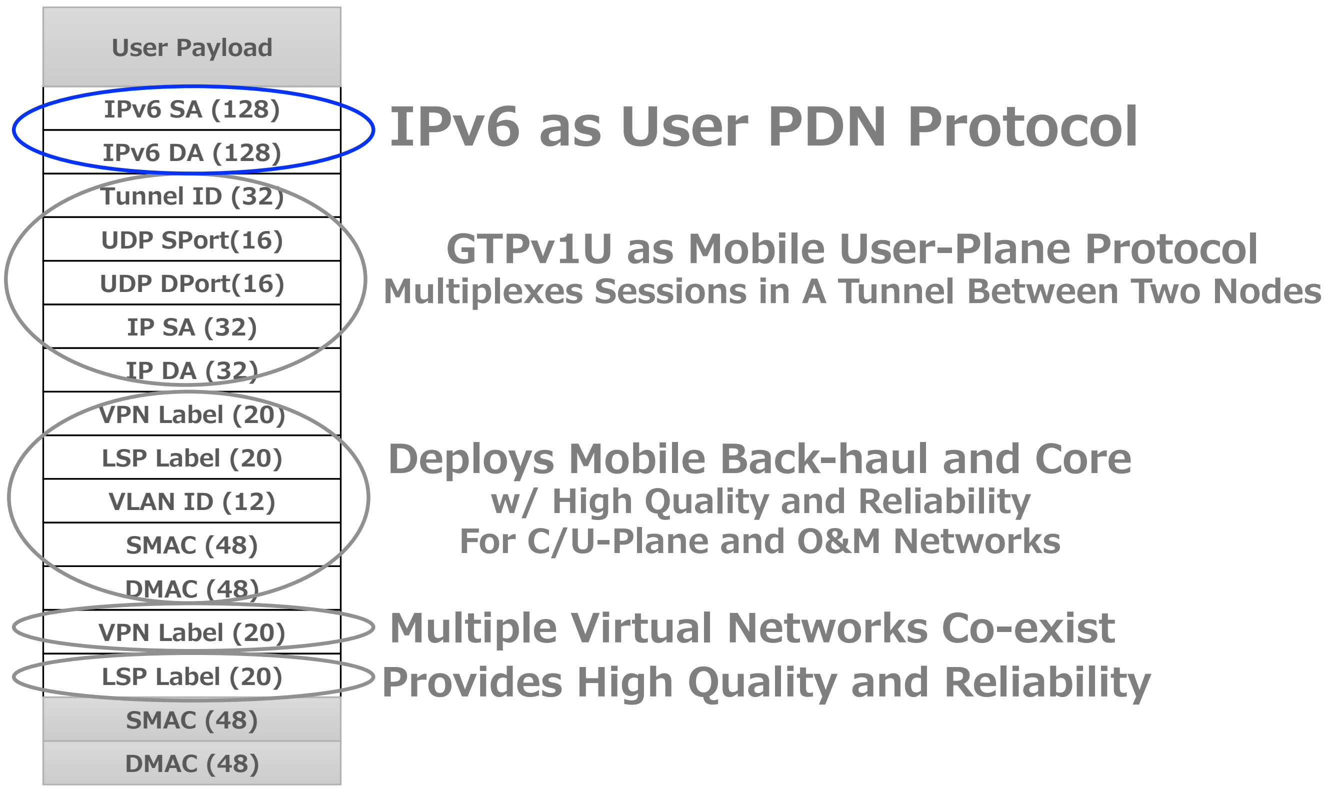
uRLLC

mMTC

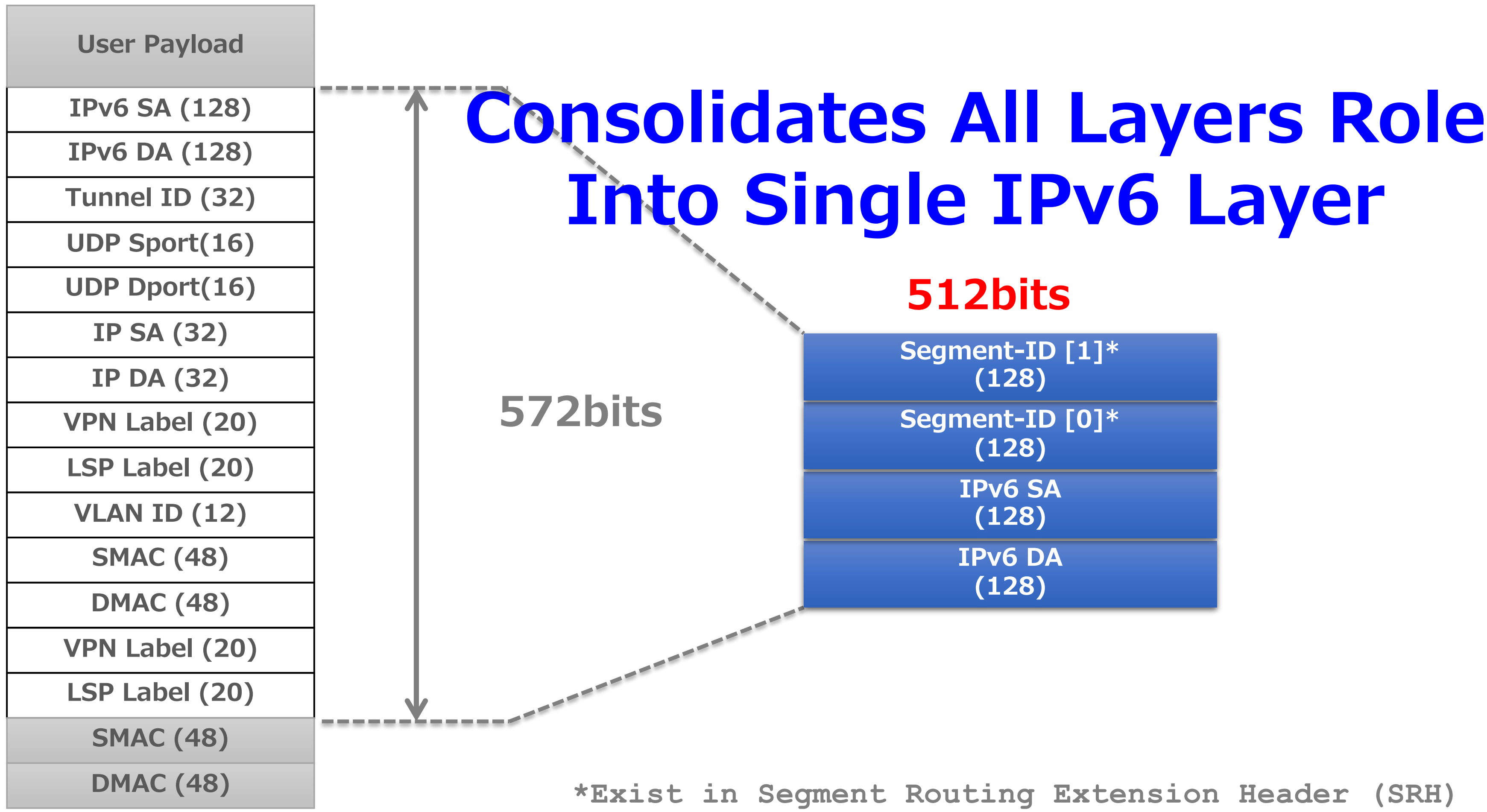
Source: NGMN white-paper

But Today's U-plane Transports Are Well Complicated Already, Why?

Stacking Multiple Small ID Space Networks to Fulfill Requirements of Reliability, VPNs, etc.,



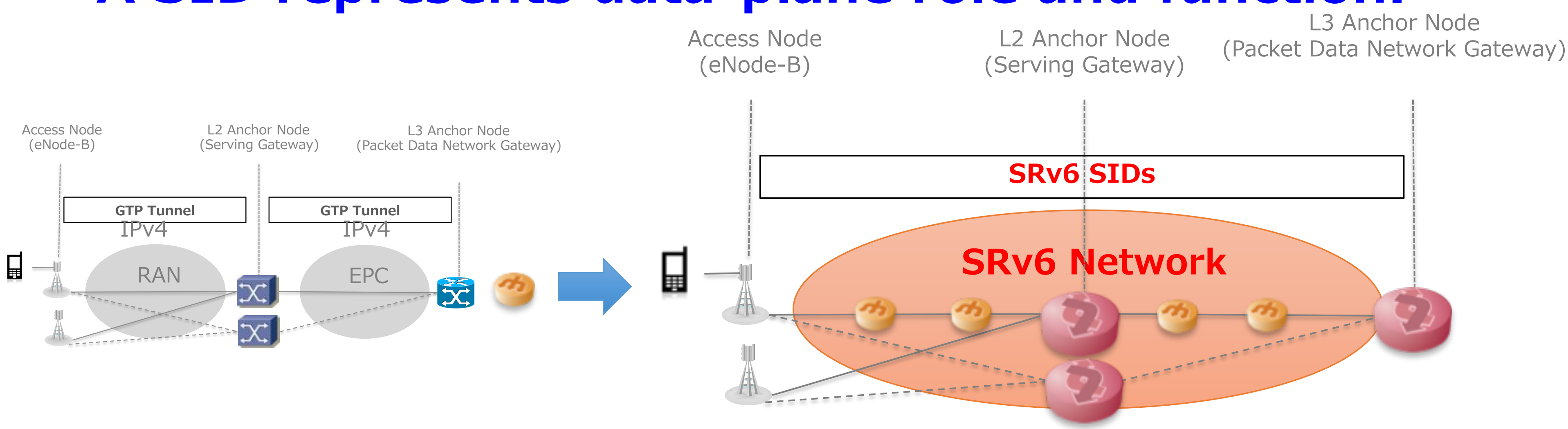
How to Simplify Such Complicating Stack?



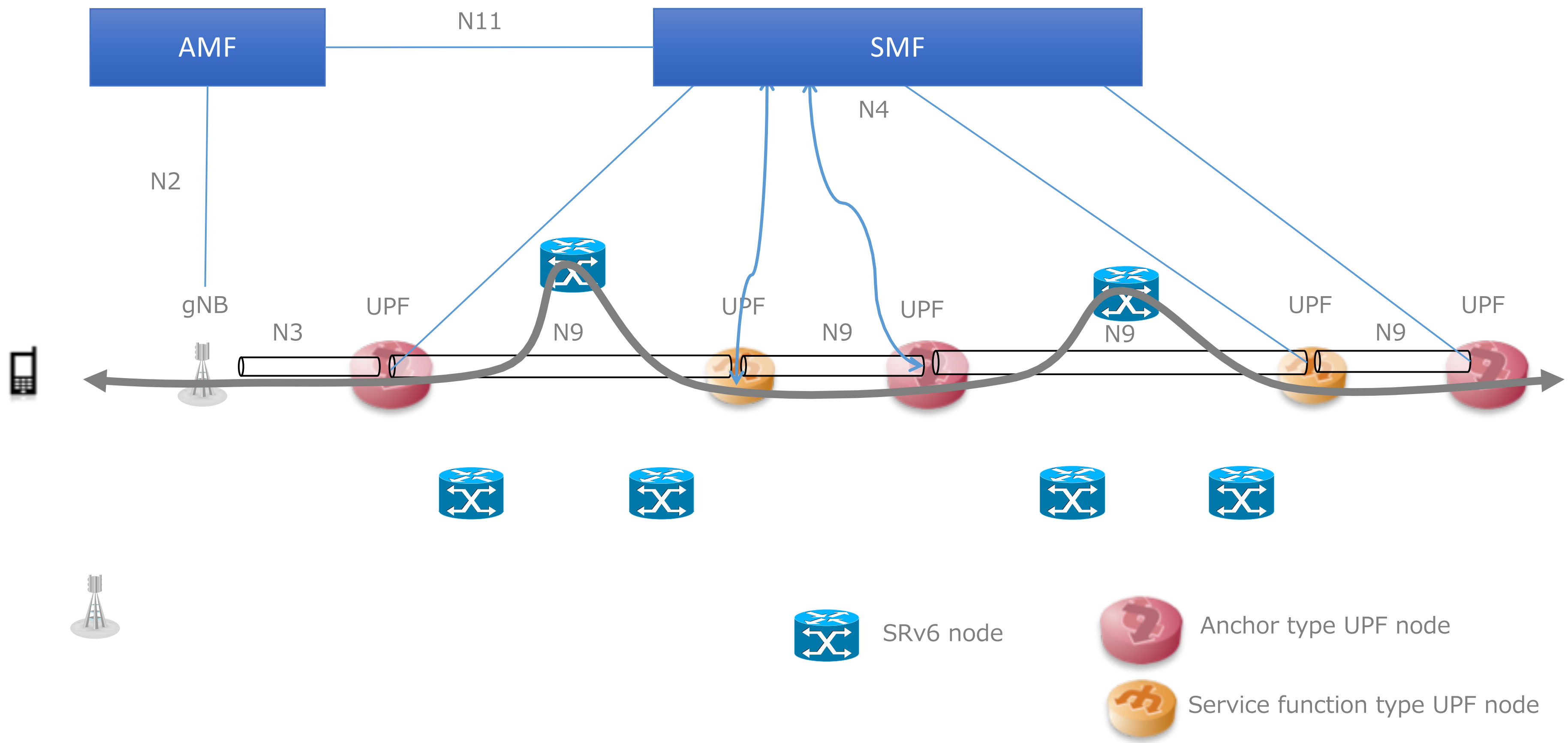
*Exist in Segment Routing Extension Header (SRH)

What if SRv6 Becomes An Alternative of GTP-U Tunnel?

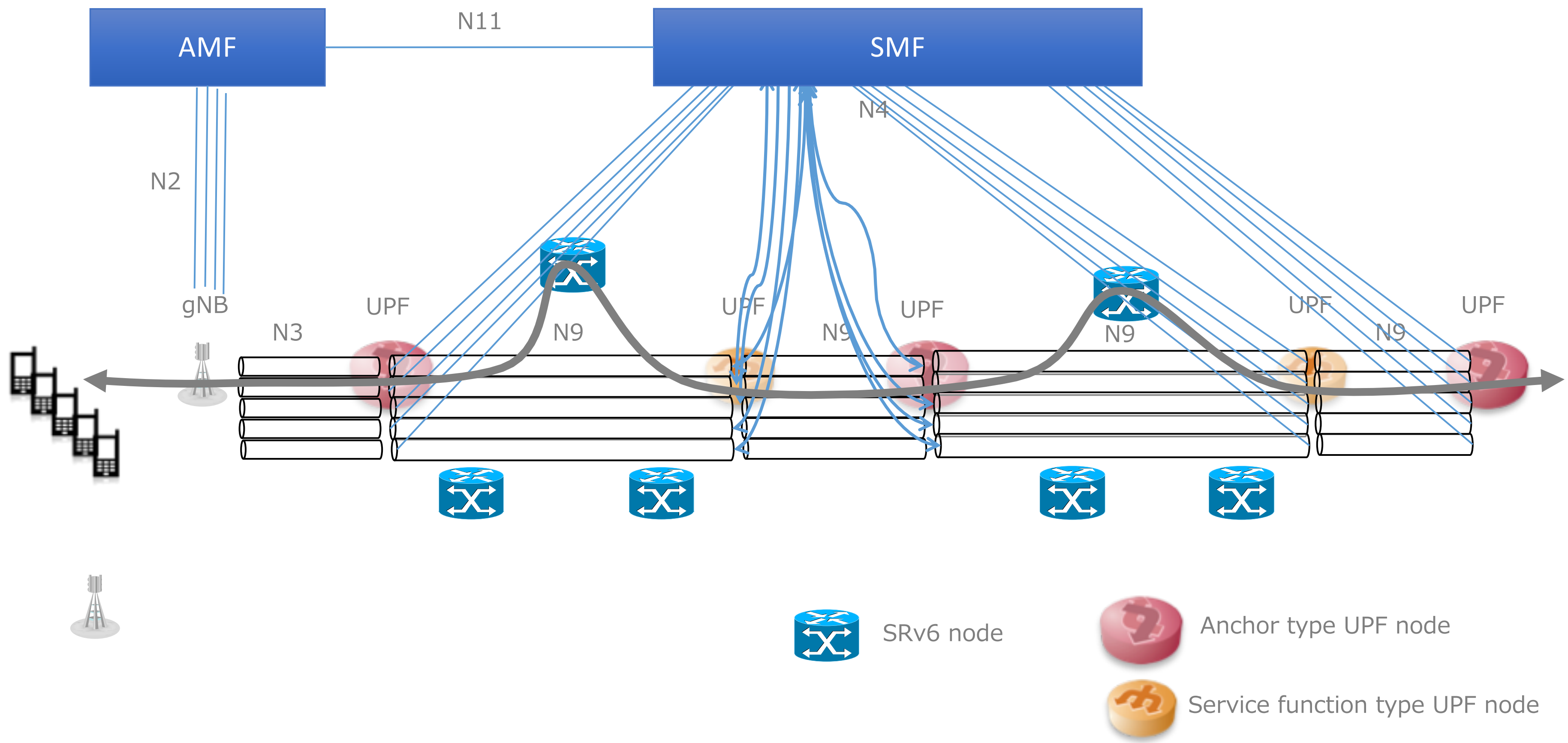
- ~~Well fragmented to RAN, EPC and SGi.~~
- ~~Per-session tunnel creation and handling.~~
- ~~Non-optimal data-path.~~
- IPv6 integrates networks of the mobile and others.
- A SID represents data-plane role and function.



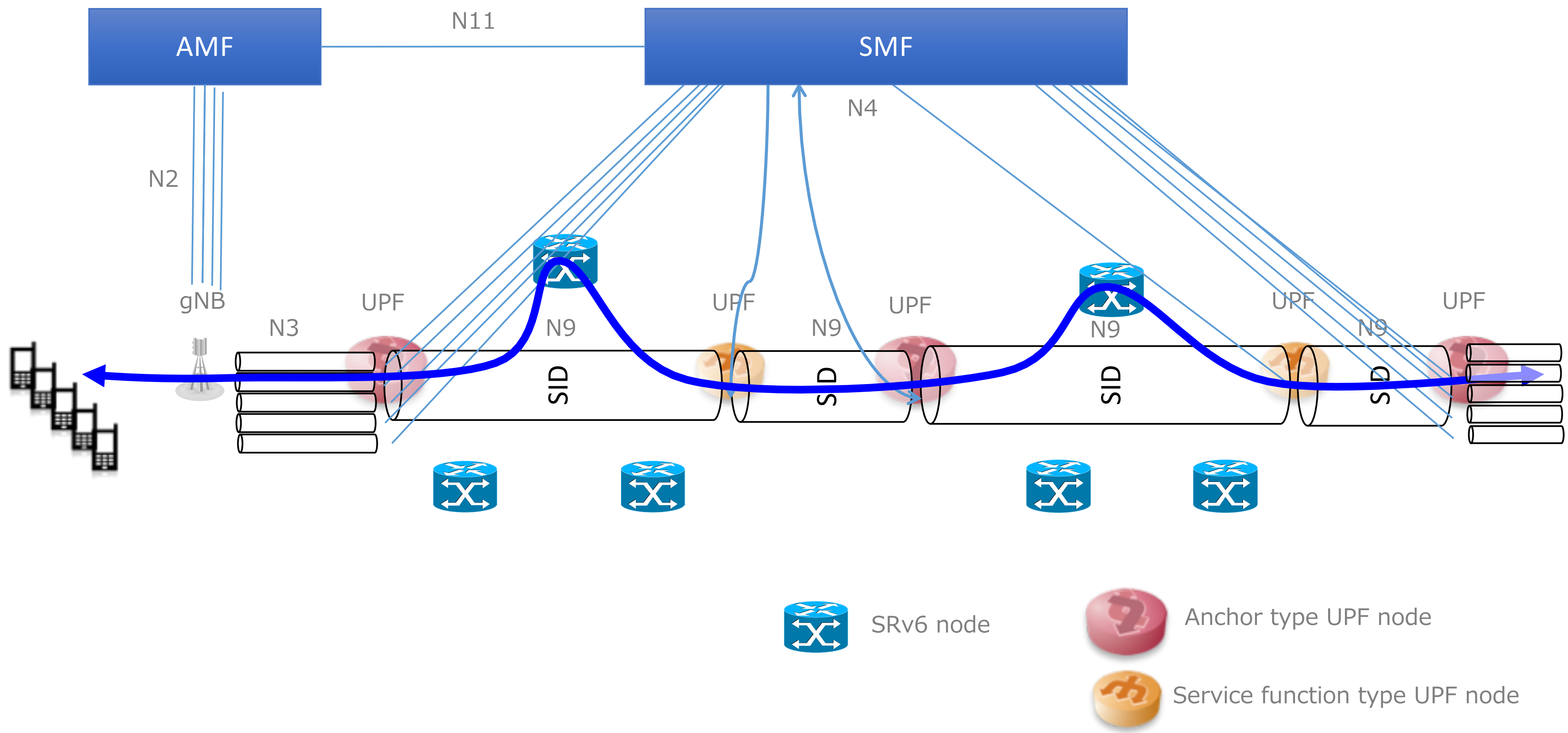
Multiple UPFs in GTP-U Case (1)



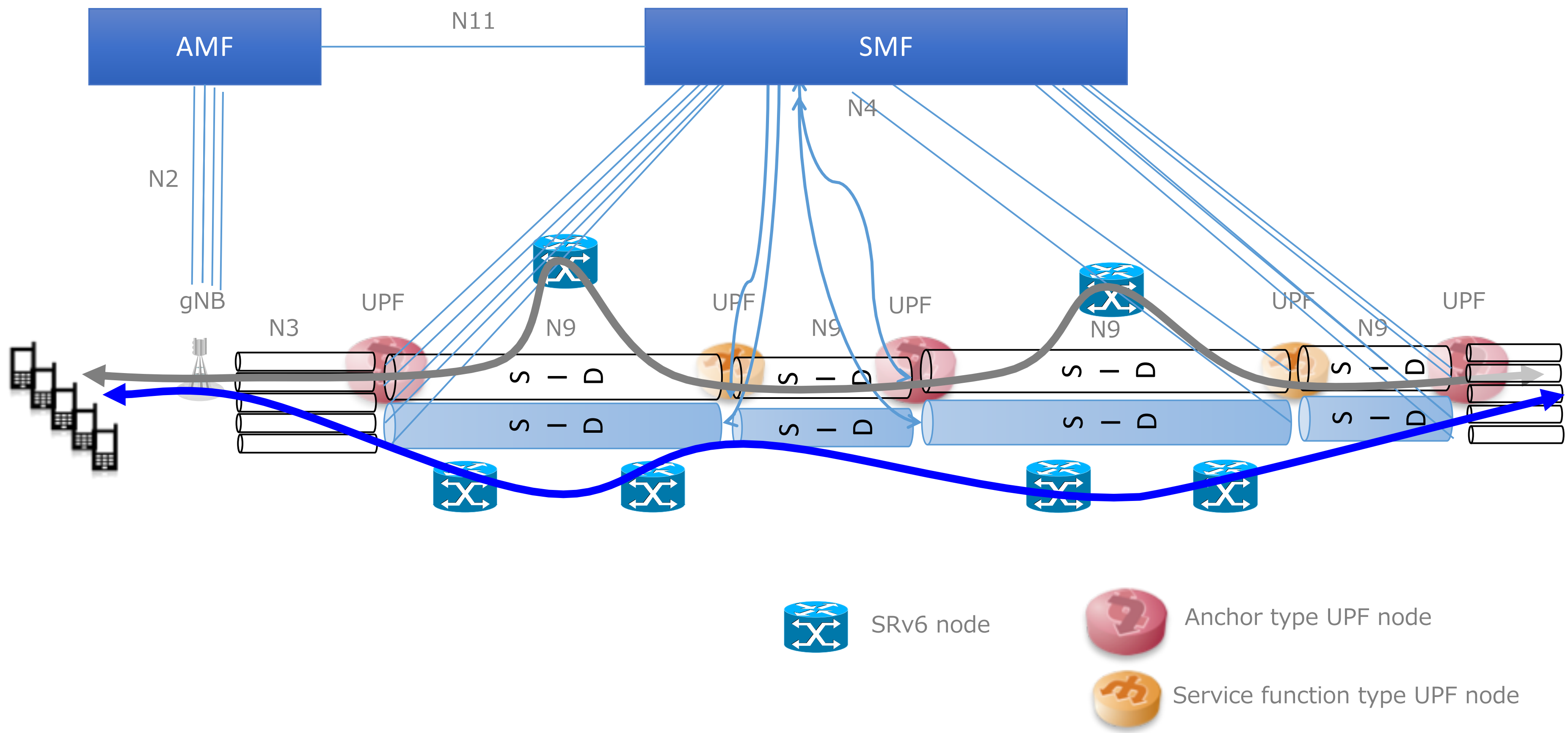
Multiple UPFs in GTP-U Case (2)



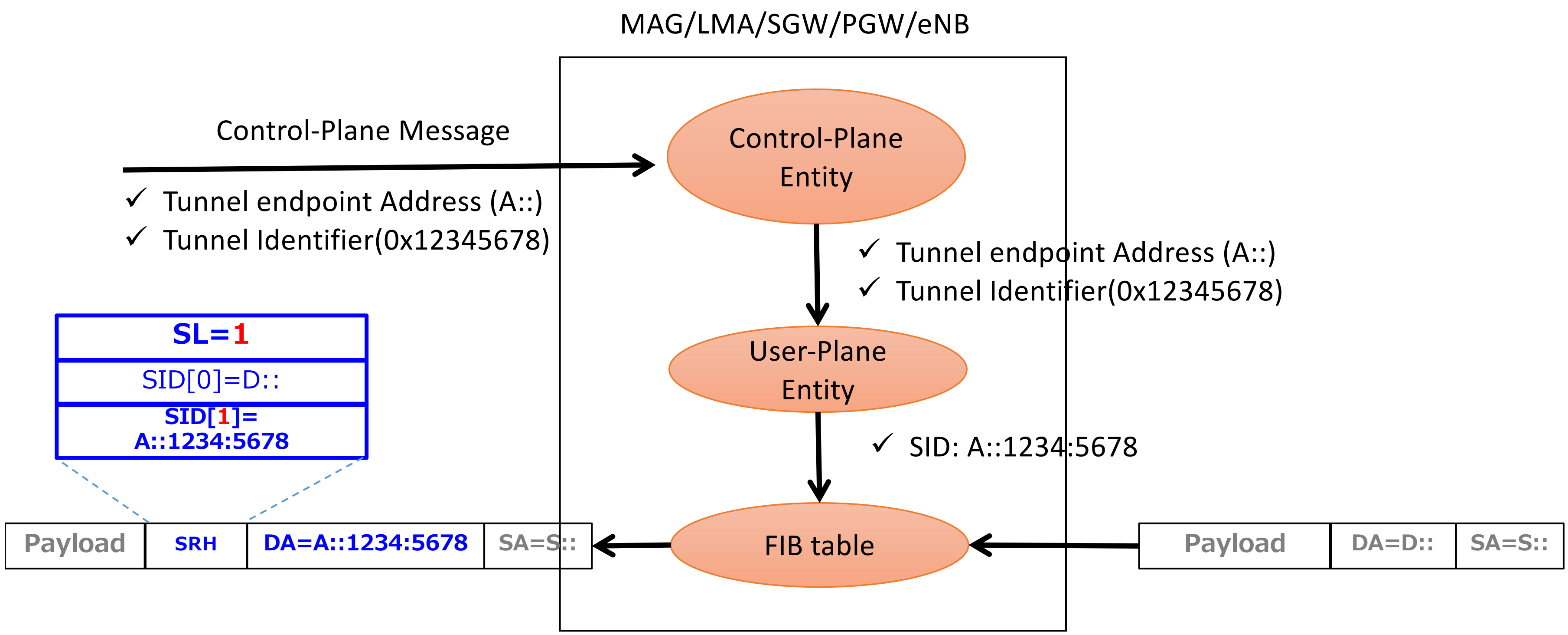
Multiple UPFs in A SRv6 Case (1)



Multiple UPFs in A SRv6 Case (2)

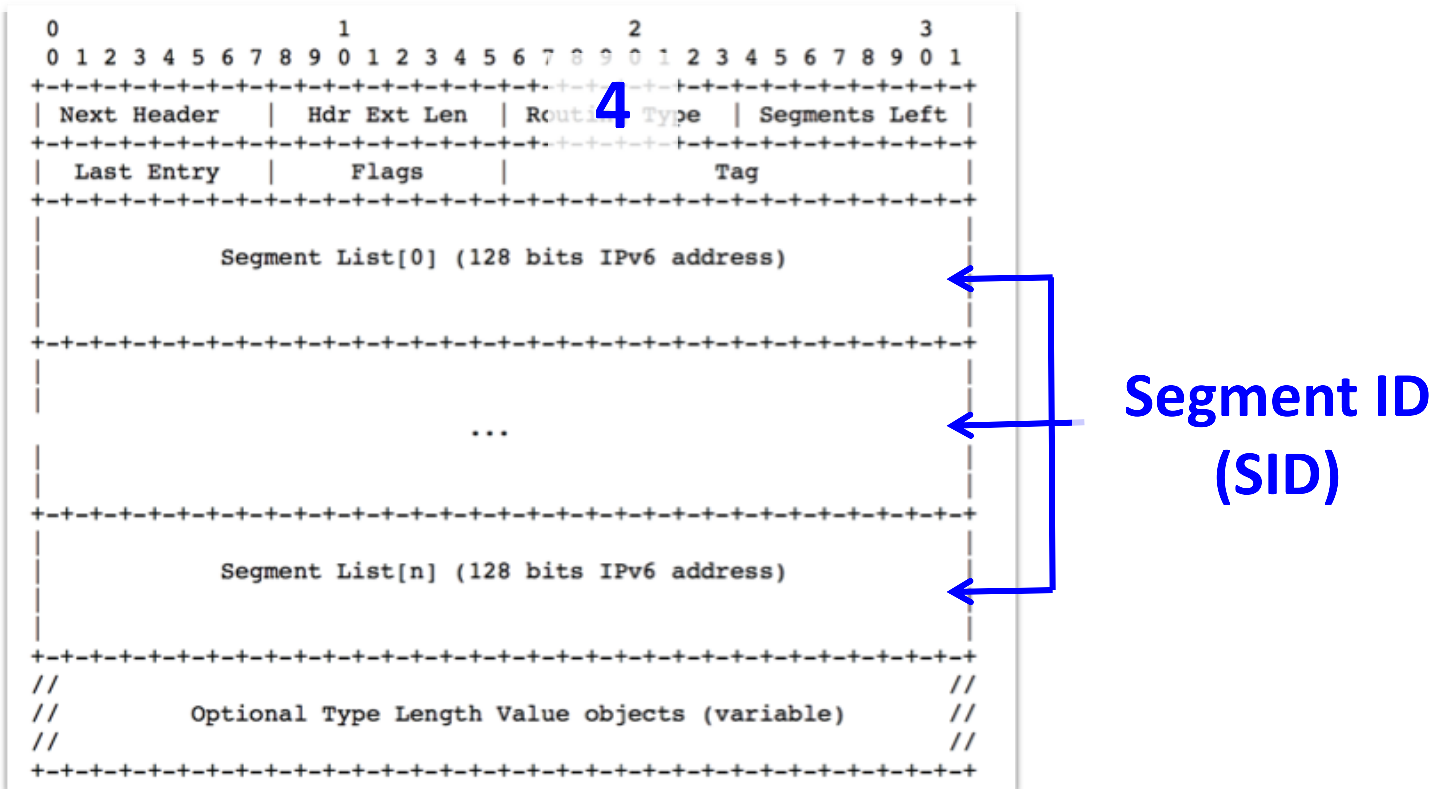


Leveraging Current Control-Plane



SRv6 in A Nutshell

SRH (Segment Routing Header)



SRv6 in A Nutshell (Cont'd)

SRv6 Function* Name	Forwarding
END	Lookup SRH
END.X	L3 cross-connect to next-hop
END.T	L3 lookup IPv6 table
END.DT6	Decap outer IPv6 hdr and lookup IPv6 table
END.DT4	Decap outer IPv6 hdr and lookup IPv4 table
END.DX6	Decap outer IPv6 hdr and IPv6 cross-connect
END.DX4	Decap outer IPv6 hdr and IPv4 cross-connect
END.B6	Bound to an SRv6 policy(SID list)

SRv6 Function* Name	Forwarding
T	Pure IPv6 transit
T.Insert	Insert an SRv6 policy (SID list)
T.Encaps	Encap SRv6 policy (SID list) by outer IPv6 hdr

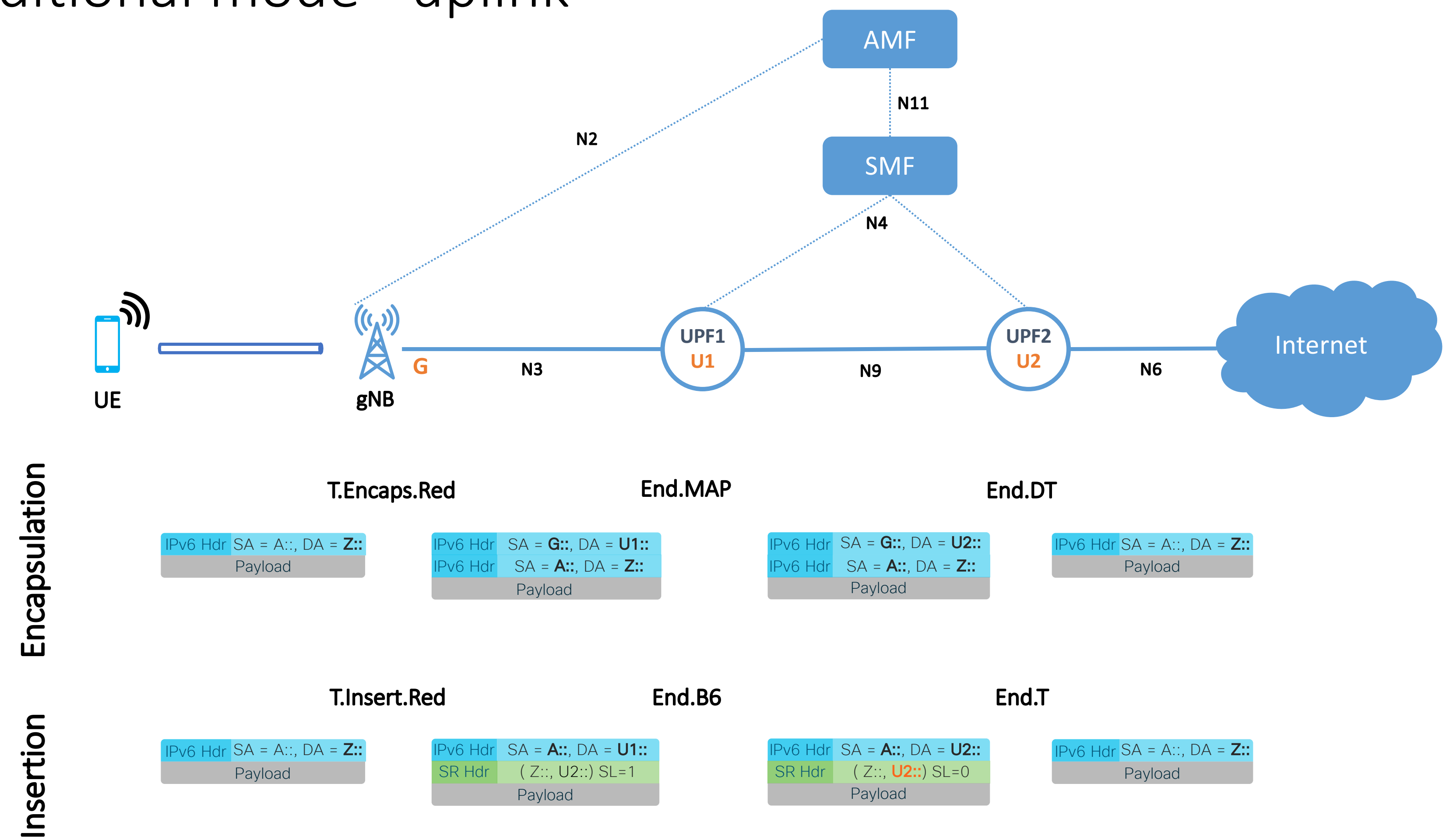
* Non exhaustive list of SRv6 Network Programming

References

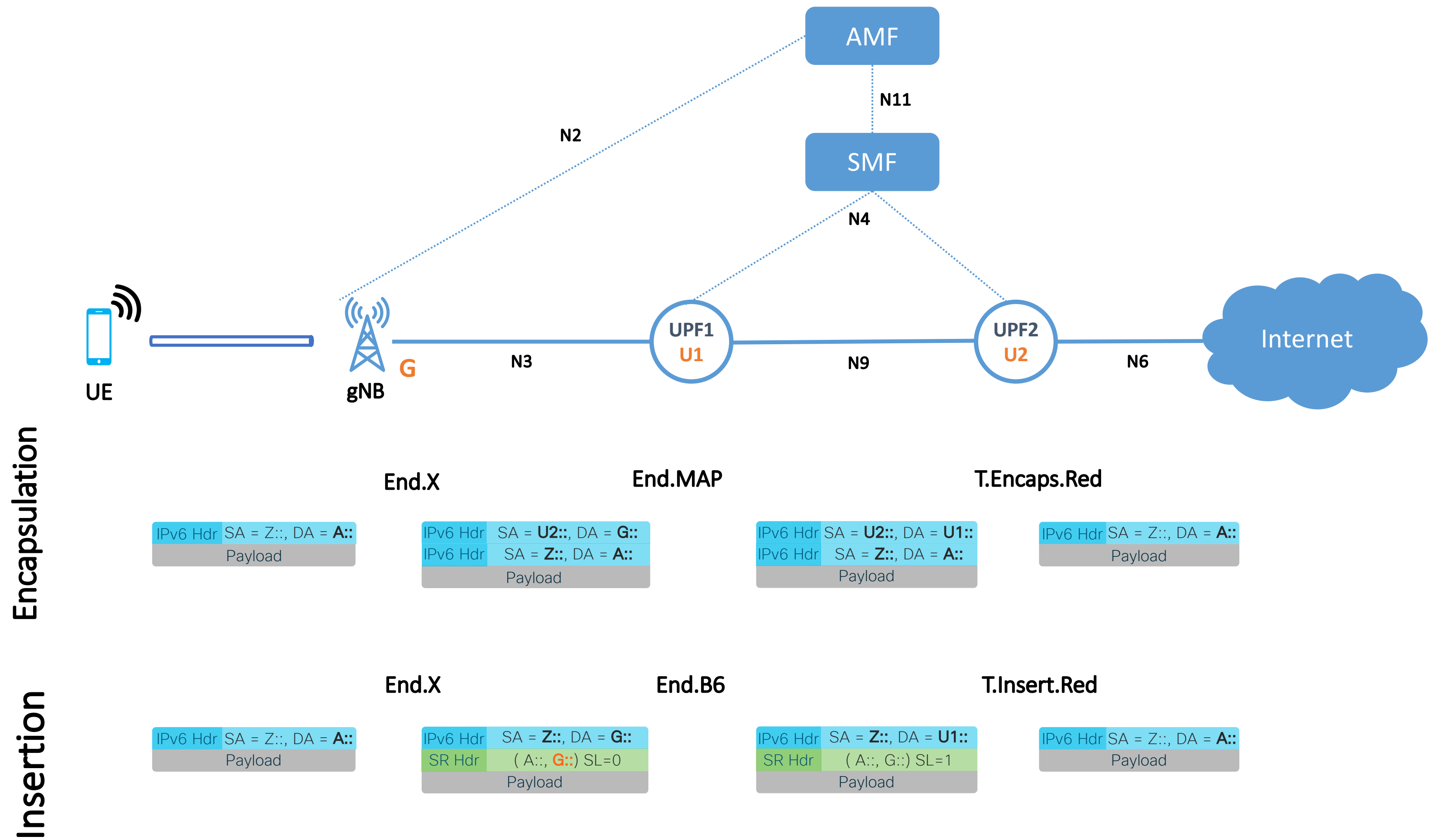
- IPv6 Segment Routing Header (SRH)
  - draft-ietf-6man-segment-routing-header

- SRv6 Network Programming
  - draft-filsfils-spring-srv6-network-programming

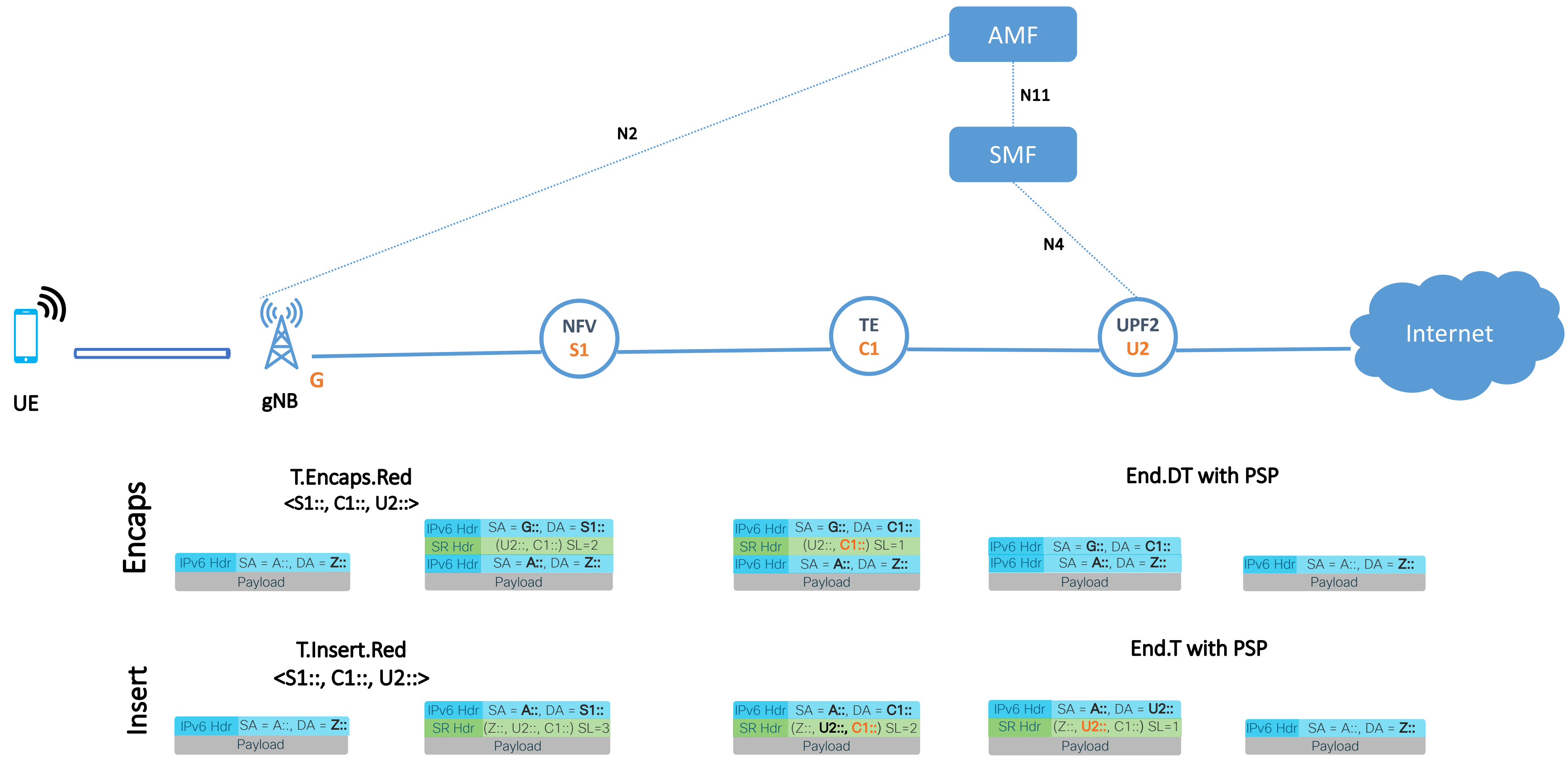
Traditional mode - uplink



Traditional mode - downlink



Enhanced mode - uplink



Enhanced mode - downlink