ETSI TR 102 467 V1.1.1 (2007-01)

Technical Report

Satellite Earth Stations and Systems (SES);
Broadband Satellite Multimedia (BSM);
Transition to IPv06

Reference

DTR/SES-00107

Keywords

broadband, multimedia, interworking, IP, IPv6,
satellite

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Satellite Earth Stations and Systems (SES).

Introduction

This Technical Report is the output from Task 6 of SES BSM STF 283, "Transition to IPv6".

The terms of reference for task 6 state that this Technical report will address the following issues:

- A technical review of the transition to IPv6
- Provide an overview of the satellite specific issues relating to the introduction of IPv6 networks
- Produce recommendations for further work to support the introduction of IPv6 in satellite communications, and in particular into the BSM architectures and standards

The overall structure of this document follows the broad outlines of previous BSM Technical Reports, with Chapters on Services, Functional Requirements and Functional Architecture, followed by a Chapter on Recommendations for Future Work. Running through these topics will be reference to the various underlying Satellite-specific network aspects that have been defined beneath the SI-SAP layer in previous BSM work. In particular, the impact of transition to IPv6 may be different for star architecture, mesh architecture, transparent and regenerative satellite payloads.

1 Scope

The present document examines the transition to IPv6 is in the context of the forces motivating this transition and both the "end-game" services and architecture (when IPv6 is ubiquitous) and the transitional phase where interworking between IPv4 equipment and IPv6 equipment will become the norm. The purpose of this document is to survey the major technical and functional consequences of these developments on the BSM. It will consider the particular effects on the BSM Functional and Services Architecture as already developed in earlier Technical Reports and Technical Standards. By reference to these documents, especially references [24] to [30] and [36]. This Technical Report identifies areas of the BSM architecture where further Technical Specifications and Reports need to be developed to handle the issues arising from the transition to IPv6 (see clause 12). Aspects of address resolution, configuration management, performance and interworking will form the main focus of the work. Consideration will also be given to Quality of Service impacts, multicasting and security.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

[1] IETF RFC 3095: "Robust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed", Borman, C. et al.

[2] IETF draft-byun-ipdvb-ule-header-comp-00: "Header Compression over Unidirectional Lightweight Encryption (ULE)", Byun, D., Border, J. and R. Ragland.

[3] IETF RFC 2529: "Transmission of IPv6 over IPv4 Domains without Explicit Tunnels", Carpenter, B. and C. Jung.

[4] IETF RFC 3056: "Connection of IPv6 Domains via IPv4 Clouds", Carpenter, B. and K. Moore.

[5] IETF RFC 2508: "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links", Casner, S. and V. Jacobson.

[6] IETF RFC 2740: "OSPF for IPv6" Coltun, R., D. Ferguson, J. Moy.

[7] IETF RFC 2473: "Generic Packet Tunnelling in IPv6 Specification", Conta, A. and S. Deering.

[8] IETF RFC 4443: "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", Conta, A., Deering, S. and Gupta, M.

[9] IETF RFC 2464: "Transmission of IPv6 Packets over Ethernet Networks", Crawford, M.

[10] IETF RFC 2467: "Transmission of IPv6 Packets over FDDI Networks", Crawford, M.

[11] IETF RFC 2460: "Internet Protocol, Version 6 (IPv6) Specification" Deering, S. and R. Hinden.

[12] IETF RFC 1883: "Internet Protocol, Version 6 (IPv6) Specification", (Obsoleted by RFC 2460), Deering, S. and R. Hinden.

[13] IETF RFC 4007: "IPv6 Scoped Address Architecture" Deering, S., Haberman, B., Jinmei, T., Nordmark, E. and B. Zill.

[14] IETF RFC 2507: "IP Header Compression" Degermark, M., Nordgren, B. and S. Pink.

[15] IETF RFC 3484: "Default Address Selection for Internet Protocol version 6 (IPv6)" Draves, R.

[16] IETF RFC 3646: "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", Droms, R.

[17] IETF RFC 3736: "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6" Droms, R.

[18] IETF RFC 3315: "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney.

[19] IETF RFC 3053: "IPv6 Tunnel Broker", Durand, A., Fasano, P., Guardini, L and D. Lento.

[20] IETF RFC 3091: "DNS IPv6 Transport Operational Guidelines", Durand, A. and Ihren, J.

[21] IETF RFC 3077: "A Link-Layer Tunnelling Mechanism for Unidirectional Links", Duros, E., Dabbous, W., Izumiyama, H., Fujii, N., and Y. Zhang.

[22] IETF RFC 1924: "A Compact Representation of IPv6 Addresses" Elz, R.

[23] ETSI TR 101 984: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; Services and Architectures".

[24] ETSI TR 101 985: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; IP over Satellite".

[25] ETSI TR 102 155: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; IP interworking over satellite; Addressing and routing".

[26] ETSI TR 102 156: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; IP interworking over satellite; Multicasting".

[27] ETSI TR 102 157: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; IP Interworking over satellite; Performance, Availability and Quality of Service".

[28] ETSI TR 102 287: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); IP Interworking over satellite; Security aspects".

[29] ETSI TS 102 292: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM) services and architectures; Functional architecture for IP interworking with BSM networks".

[30] ETSI TS 102 293: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM) services and architectures; IP Interworking over satellite; Multicast group management; IGMP adaptation".

[31] ETSI TS 102 294: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM) services and architectures; IP interworking via satellite; Multicast functional architecture".

[32] ETSI TS 102 295: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM) services and architectures; BSM Traffic Classes".

[33] ETSI TR 102 353, "Satellite Earth Stations and Systems (SES);Broadband Satellite Multimedia (BSM); Guidelines for the Satellite Independent Service Access Point (SI-SAP)".

[34] ETSI TS 102 357: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Common Air interface specification; Satellite Independent Service Access Point SI-SAP".

[35] ETSI TS 102 460: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Address Management at the SI-SAP".

[36] ETSI TS 102 461: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Multicast Source Management".

[37] ETSI TS 102 462: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); QoS Functional Architecture".

[38] ETSI TS 102 463: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Interworking with IntServ QoS".

[39] ETSI TS 102 464: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Interworking with DiffServ Qos".

[40] ETSI TS 102 465: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); General Security Architecture".

[41] ETSI TS 102 466: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Multicast Security Architecture".

[42] European Commission document: "Next Generation Internet - priorities for action in migrating to the new Internet protocol IPv6", 21 February 2002.

NOTE: Available at http://businessmobile.fr/livres-blancs/0,39044475,60129278p-39000971q,00.htm

[43] Evans, K., "Statement of the Honorable Karen Evans, Administrator for Electronic Government and Information Technology, Office of Management and Budget, before the Committee on Government Reform, U.S. House of Representatives", June 29 2005.

NOTE: Available at http://www.whitehouse.gov/omb/legislative/testimony/evans/evans052905.html,.

[44] IETF RFC 4326: "Unidirectional Lightweight Encapsulation (ULE) for Transmission of IP Datagrams over an MPEG-2 Transport Stream (TS)", Fairhurst, G. and B. Collini-Nocker.

[45] IETF draft-ipdvb-ar-05: "Address Resolution Mechanisms for IP Datagrams over MPEG-2 Networks", Fairhurst, G. and Montpetit, M.-J.

[46] Fritsche, W., Gessler, G. and Mayer, K., "Preparation for IPv6 in Satellite Communications", Final Report for ESA contract 17629/03/NL/ND, July 2004.

[47] IETF RFC 1519: "Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy", Fuller, V., Li, T., Yu, J. and K. Varadhan.

[48] IETF RFC 2893: "Transition Mechanisms for IPv6 Hosts and Routers", Gilligan, R. and E. Nordmark (Obsoleted by RFC 4213).

[49] IETF RFC 3493: "Basic Socket Interface Extensions for IPv6" Gilligan, R., Thomson, S., Bound, J., McCann, J. and W. Stevens.

[50] IETF RFC 3307: "Allocation Guidelines for IPv6 Multicast Addresses", Haberman, B.

[51] IETF RFC 3306: "Unicast-Prefix-based IPv6 Multicast Addresses", Haberman, B. and D. Thaler.

[52] IETF RFC 3178: "IPv6 Multihoming Support at Site Exit Routers" Hagino, J. and H. Snyder.

[53] IETF RFC 2472: "IP Version 6 over PPP", Haskin, D. and E. Allen.

[54] IETF RFC 4193: "Unique Local IPv6 Unicast Addresses", Hinden, R. and Haberman, B.

[55] IETF RFC 4291: "Internet Protocol Version 6 (IPv6) Addressing Architecture", Hinden, R. and S. Deering.

[56] IETF RFC 2375: "IPv6 Multicast Address Assignments", Hinden, R. and S. Deering.

[57] IETF RFC 3768: "Virtual Router Redundancy Protocol (VRRP)", Hinden, R., Ed.

[58] IETF RFC 3879: "Deprecating Site Local Addresses" Huitema, C. and B. Carpenter.

[59] IETF RFC 3177: "IAB/IESG Recommendations on IPv6 Address Allocations to Sites".

[60] IEEE "Guidelines for 64 bit Global Identifier (EUI-64) Registration Authority".

NOTE: Available at http://standards.ieee.org/regauth/oui/tutorials/EUI64.html, (March 1997).

[61] IETF RFC 4057: "IPv6 Enterprise Network Scenarios" Bound, J., Ed.

[62] IETF RFC 3843: "RObust Header Compression (ROHC): A Compression Profile for IP", Jonsson, L-E. and G. Pelletier.

[63] IETF RFC 3545: "Enhanced Compressed RTP (CRTP) for Links with High Delay, Packet Loss and Reordering", Koren, T., Casner, S., Geevarghese, J., Thompson, B. and P. Ruddy.

[64] IETF RFC 2545: "Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing", Marques, P. and F. Dupont.

[65] IETF RFC 3041: "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", Narten, T. and R. Draves.

[66] IETF RFC 2461: "Neighbour Discovery for IP Version 6 (IPv6)", Narten, T., Nordmark, E. and W. Simpson.

[67] IETF RFC 4213: "Basic Transition Mechanisms for IPv6 Hosts and Routers" Nordmark, E. and Gilligan, R.

[68] IETF RFC 2765: "Stateless IP/ICMP Translation Algorithm (SIIT)", Nordmark, E.

[69] IETF RFC 2909: "The Multicast Address-Set Claim (MASC) Protocol" Radoslavov, P., Estrin, D., Govindan, R., Handley, M., Kumar, S. and D. Thaler.

[70] IETF RFC 3697: "IPv6 Flow Label Specification" Rajahalme, J., Conta, A., Carpenter, B., and S. Deering.

[71] IETF RFC 3956: "Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address", Savola, P. and B. Haberman.

[72] IETF RFC 4038: "Application Aspects of IPv6 Transition", Shin, M-K., Hong, Y-G., Hagino, J., Savola, P., and E. Castro.

[73] IETF RFC 4389: "Neighbor Discovery Proxies (ND Proxy)" Thaler, D., Talwar, M. and C. Patel.

[74] IETF RFC 2462: "IPv6 Stateless Address Autoconfiguration", Thomson, S. and T. Narten.

[75] IETF RFC 3596: "DNS Extensions to Support IP Version 6", Thomson, S., Huitema, C., Ksinant, V., and M. Souissi.

[76] IETF RFC 3633 "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", Troan, O. and R. Droms, Decembe 2003.

[77] IETF RFC 2766: "Network Address Translation - Protocol Translation (NAT-PT)", Tsirtsis, G. and P. Srisuresh.

[78] IETF RFC 3810: "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", Vida, R., Ed. and L. Costa, Ed.

[79] IETF RFC 4215: "Analysis on IPv6 Transition in Third Generation Partnership Project (3GPP) Networks" Wiljakka, J.

[80] Christian Huitema. "IPv6: The New Internet Protocol", (Second edition). Prentice Hall, November 1997, ISBN: 0138505055.

[81] IETF RFC 3775: "Mobility support in IPv6", D.Johnson, C.Perkins and J.Arkko.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

NOTE: Some of the definitions are copied from TR 101 984 [23].

broadcast: communication capability which denotes unidirectional distribution to an unspecified number of access points connected to the network

NOTE: The communication may reach any or all access points and each terminal may select which broadcast information to receive. In the context of IPv6, the term "broadcast address" is replaced by "multicast address" (see RFC 4291 [55], "Internet Protocol Version 6 (IPv6) Addressing Architecture").

channel: means of unidirectional transmission of signals between two points

NOTE: Several channels may share a common transport mechanism.

connection oriented: communication method in which communication proceeds through three well-defined phases: connection establishment, data transfer, connection release

connectionless: communication method which allows the transfer of information between users without the need for connection establishment procedures

control plane: plane which has a layered structure and performs the call control and connection control functions, and deals with the signalling necessary to set up, supervise and release calls and connections

layer management functions: management functions (e.g. meta-signalling) relating to resources and parameters residing in its protocol entities

NOTE: Layer Management handles the Operation And Maintenance (OAM) information flows specific to the layer concerned.

management plane: plane which provides two types of functions, namely layer management and plane management functions

NOTE: Plane management has no layered structure.

multicast: communication capability which denotes unidirectional distribution from a single source access point to a number of specified destination access points

multipoint: communication configuration attribute which denotes that the communication involves more than two access points

plane management functions: management functions related to a system as a whole and provides co-ordination between all the planes

service category or service class: se **service attribute:** specified characteristic of a telecommunication service

NOTE: The value(s) assigned to one or more service attributes may be used to distinguish that telecommunication service from others.

rvice offered to the users described by a set of performance parameters and their specified values, limits or ranges

NOTE: The set of parameters provides a comprehensive description of the service capability.

service component: single type of telecommunication service

NOTE: Service components are divided into speech, audio, video and data:
speech: voice telecommunication;
audio: telecommunication of sound in general;
video: telecommunication of full motion pictures, and of stills;
data: telecommunication of information-files (text, graphics, etc); and
MultiMedia (MM): combination of two or more of the above components (speech, audio, video, data), with a temporal relationship (e.g. synchronization) between at least two components.

telecommunication service: service offered by a network operator or service provider to its customers in order to satisfy a specific telecommunication requirement

NOTE: Telecommunication services are divided into two broad families: bearer services and teleservices:
bearer service: type of telecommunication service that provides the capability of transmission of signals between access points;
teleservice: type of telecommunication service that provides the complete capability, including terminal equipment functions, for communication between users according to standardized protocols and transmission capabilities established by agreement between operators.

user plane: plane which has a layered structure and provides for user information flow transfer, along with associated controls (e.g. flow control, recovery from errors, etc.)

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AAA	Authentication, Authorization and Accounting
AS	Autonomous System
BGP	Border Gateway Protocol
BGP4+	BGP-4 with multi-protocol extensions for IPv6 inter-domain routing
BSM	Broadband Satellite Multimedia
BSM_ID	BSM IDentifier
BSMS	BSM System
CBT	Core Based Tree
CERN	Centre Europeenn de Recherche Nucleaire
CIDR	Classless Inter-Domain Routing
DAD	Duplicate Address Detection
DiffServ	Differentiated Services
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Server
DSCP	Differentiated Services Code Point
ECN	Explicit Congestion Notification
ESA	European Space Agency
ESP	Encapsulating Security Payload
EUI	Extended Unique Identifier
FTP	File Transfer Protocol
GID	Group IDentity
HIP	Host Identity Protocol
HMIPv6	Hierarchical Mobile IPv6
IANA	Internet Assigned Number Association
ICMP	Internet Control Message Protocol
ICMPv4	Internet Control Message Protocol for IPv4
ICMPv6	Internet Control Message Protocol for IPv6
ID	IDentifier
IETF	Internet Engineering Task Force
IGP	Interior Gateway Protocol
IRTF	Internet Research Task Force
ISP	Internet Service Provider
IntServ	Integrated services
IP	Internet Protocol
IPv4	Internet Protocol version 4

IPv6	Internet Protocol version 6
ISP	Internet Service Provider
LAN	Local Area Network
MAC	Medium Access Control
MLDv1	Multicast Listener Discovery Version 1
MLDv2	Multicast Listener Discovery Version 2
MIPv6	Mobile IPv6
MM	MultiMedia
MOSPF	Multicast Open Shortest Path First
MPOE	Multi Protocol Over Ethernet
MSDP	Multicast Source Discovery Protocol
MTU	Maximum Transmission Unit
NAT	Network Address Translation
NAT-PT	Network Address Translation - Protocol Translation
NAPT	Network Address Port Translation
NCC	Network Control Centre
ND	Neighbour Discovery
NEMO	NEtwork MObility
NS	Neighbour Solicitation
NUD	Neighbour Unreachability Detection
OAM	Operation, Administration and Maintenance
OSPF	Open Shortest Path First
OPSFv3	Open Shortest Path First version 3
PIM-DM	Protocol Independent Multicast - Dense Mode
PIM-SM	Protocol Independent Multicast - Sparse Mode
PIM-SSM	Protocol Independent Multicast - Source Specific Multicast
PMTU	Path Maximum Transmission Unit
PPP	Point of Presence Protocol
PPPoE	PPP over Ethernet
QoS	Quality of Service
RFC	IETF Request For Comments
ROHC	RObust Header Compression
RSVP	Resource reSerVation Protocol
RTP	Real Time Protocol
RTT	Round Trip Time
SDAF	Satellite Dependent Adaptation Function
SIAF	Satellite Independent Adaptation Function
SIIT	Stateless IP/ICMP Translation
SI-SAP	Satellite Independent Service Access Point
SOHO	Small Office Home Office
ST	Satellite Terminal
TCP	Transmission Control Protocol
TOS	Type Of Service
UDLR	UniDirectional Link Routing
UDP	User Datagram Protocol
ULE	Unidirectional Lightweight Encapsulation
VRRP	Virtual Router Redundancy Protocol
VPN	Virtual Private Network
WG	Working Group

4 Introduction and background

4.1 History

The Internet grew from a US Department of Defense-sponsored research network, DARPA-net, during the 1970's and 1980's, although much of the original research that underpinned this work took place in Europe at CERN. The network layer addressing scheme in the Internet Protocol (that became codified as IPv4) used 32 bit addresses. This allows for a theoretical maximum number of four-and-a-half billion unique addresses. In practice, however, the address architecture of IPv4 restricts the available number of "public" addresses, due to reserved address ranges for multicast and for private address space. Added to this, the use of subnet addressing gives rise to inevitable inefficiencies in address usage, even though the original class-based address allocation scheme has been largely superseded by Classless Inter-Domain Routing (CIDR), as described in RFC 1519 [47], as subnet sizes have to be powers of 2, and many addresses are therefore wasted. Due to the Internet's US and subsequently European origins, address allocation is highly skewed geographically, and the Far East in particular is experiencing an acute shortage of public IPv4 addresses. As a result of the address shortages, a number of techniques have been developed in order to manage addresses in a local fashion, often through the use of Network Address Translation (NAT) and Network Address Port Translation (NAPT). There has been widespread deployment of NAT devices, and this is regarded as a serious challenge to the "end-to-end transparency" principle that many network designers believe should be upheld.

At the same time, the explosion in the number of networked and network-ready - therefore addressable - computing devices is placing additional pressure on the available address space. This is particularly important with the growth in mobile devices. As a result of the likelihood of this evolution, in the late 1980's, the IRTF and IETF began investigating a new Internet Protocol, which culminated in the specification of IPv6 in December 1995 in RFC 1883 [12]. The uptake of IPv6 has thus far been disappointingly slow, but it is generally felt that, should a strong necessity arise (such as appears likely with the growth of web-enabled mobile devices in the Asian market) there will be a major shift towards IPv6 systems. Recently, the US Government announced that it is instructing all federal agencies to deploy IPv6 by June 2008 [43]. The European Commission issued the following press release in February 2002 (see also "Next Generation Internet - priorities for action in migrating to the new Internet protocol IPv6" [42]:

- "The Communication which is entitled "IPv6 Priorities for Action" requests a European action plan to speed up the rollout of Internet Protocol version 6 (IPv6), a key technology for the "next generation" Internet. IPv6 will significantly increase the capacity for IP addresses and provide greater stability, security, privacy, power and efficiency. It is expected that space on the current Internet will be exhausted by 2005 as already 74% of the current IPv4 addresses have been allocated to organizations in North America. Also future Internet developments such as wireless machine-to-machine communications, mobile computing and third generation (3G) telephony will put an even greater strain on these limited resources. The Commission sees IPv6 alongside the European Broadband Strategy and many other initiatives launched since the Lisbon Council (which called for Europe "to become the most dynamic knowledge-based economy in the world by 2010") as a critical part of Europe's Next Generation Internet strategy. While some of the new addresses will be assigned to user's traditional PC's, most of the addresses are likely to be assigned to new types of Internet-capable devices such as mobile phones, car navigation systems, home appliances, industrial equipment and other electronic equipment."

As a result, we can anticipate the appearance of IPv6 "islands", whilst the bulk of the Internet remains IPv4-based. It is anticipated that interworking with legacy IPv4-based equipment will continue to be needed well into the first half of the present century, and perhaps beyond.

It is to be noted that IPv6 128 bit addressing and header structure are not backward compatible with IPv4 32 bit field addressing and header structure (see clause 4.3). This means that a dual IPv4/IPv6 protocol stack is required to provide backward compatibility, this is discussed further in clause 6.

4.2 IPv6 basics

The IPv6 Header structure is shown below:

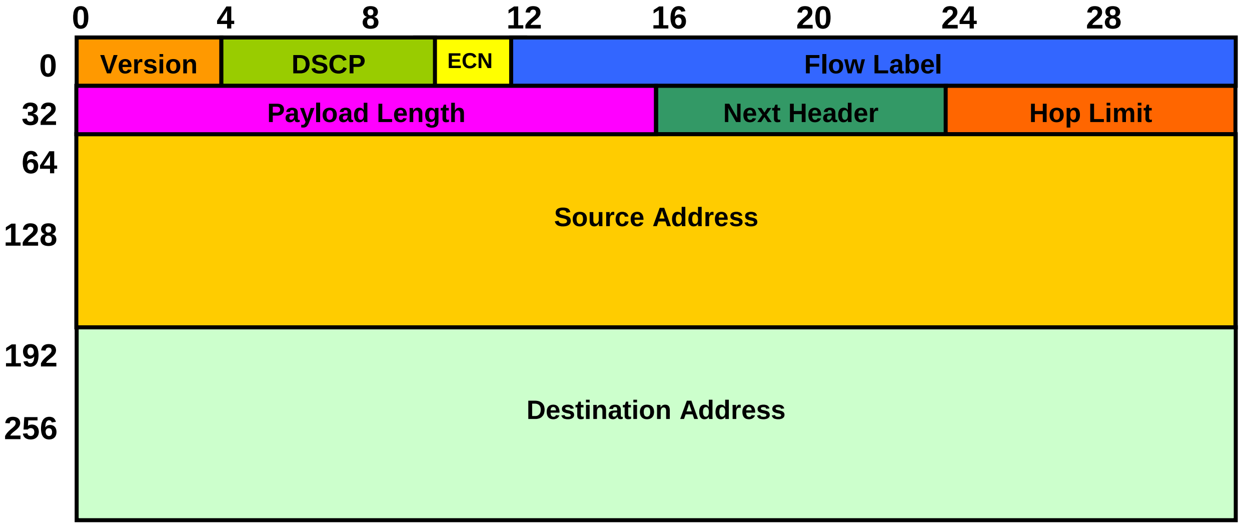


Figure 4.2.1: IPv6 Header Structure

The various fields in the IPv6 packet header are as follows:

Table 4.2.1: IPv6 Header Fields

Version	4 bit Internet Protocol version number = 6.
DSCP	6-Differentiated Services Code Point. See clause 7.2.
ECN	2 bit Explicit Congestion Notification field.
Flow Label	20 bit flow label. See clause 7.2.
Payload Length	16 bit unsigned integer. Length of the IPv6 payload, i.e., the rest of the packet following this IPv6 header, in octets. (Any extension headers present are considered part of the payload, i.e., included in the length count.)
Next Header	8 bit selector. Identifies the type of header immediately following the IPv6 header. Uses the same values as the IPv4 Protocol field.
Hop Limit	8 bit unsigned integer. Decremented by 1 by each node that forwards the packet. The packet is discarded if Hop Limit is decremented to zero.
Source Address	128 bit address of the originator of the packet.
Destination Address	128 bit address of the intended recipient of the packet (possibly not the ultimate recipient, if a Routing header is present).

4.2.1 IPv6 address representation

The 128 bit address field is usually represented in the following ways (for further details see RFC 4291 [55], for an alternative view see also [22]).

The preferred form is x:x:x:x:x:x:x:x, where the x's are one to four hexadecimal digits of the eight 16 bit fields of the address.

Examples are:

ABCD:EF01:2345:6789:ABCD:EF01:2345:6789

2001:AB8:0:0:8:8C00:2010:21EA

Note that it is not necessary to write the leading zeros in an individual field, but there must be at least one numeral in every field (except as described below).

Due to some methods of allocating IPv6 addresses, it will be common for addresses to contain long strings of zero bits. In order to make writing addresses containing zeroes easier, a special syntax is. The use of ":" indicates one or more groups of 16 bits of zeros. The "::" can only appear once in an address. It can also be used to compress leading or trailing zeros in an address.

For example, the following addresses may be represented as follows, using this method.

2001:AB8:0:0:8:8C00:2010:21EA (a unicast address) may be written 2001:AB8::8:8C00:2010:21EA

FF01:0:0:0:0:0:0:103 (a multicast address) gives FF01::103

0:0:0:0:0:0:0:1 (the loopback address) gives ::1

0:0:0:0:0:0:0:0 (the unspecified address) gives simply:

Where an address contains an embedded IPv4 address (see clause 4.2.3 and 6.1.1.1), an alternative representation is available. Here the format is

x:x:x:x:x:x:d.d.d.d,

Where the x's are the hexadecimal values of the six high-order 16 bit pieces of the address, and the d's are the decimal values of the four low-order 8 bit pieces of the address (standard IPv4 representation). Examples are:

0:0:0:0:0:0:13.3.76.3 (an IPv4-compatible IPv6 address. *NB these have recently been deprecated by the IETF*)

0:0:0:0:0:FFFF:129.142.48.18 (an IPv4-mapped IPv6 address, which is now preferred to the above - see RFC 4291 [55])

or in compressed form:

::13.3.76.3 and

::FFFF:129.142.48.18

IPv6 Address prefixes are represented in a manner similar to that used in IPv4 Classless Inter-Domain Routing (CIDR), as defined in RFC 1519 [47], in the form:

IPv6-address/prefix-length,

Where IPv6-address is an IPv6 address expressed using the notations described above, and prefix-length is a decimal value specifying the number of bits in the prefix (i.e. the leftmost contiguous prefix length).

For example, the hexadecimal 60 bit prefix 20010CC80000C93 can be represented as follows:

2001:0CC8:0000:C930:0000:0000:0000:0000/60

2001:0CC8::C930:0:0:0:0/60

2001:0CC8:0:C930::/60

4.2.2 IPv6 address scope

In addition to the expanded address space, IPv6 introduces the concept of address scope. There are three scopes now defined for IPv6 addresses (see RFC 4291 [55]), as follows:

Node-local scope: these addresses are valid only for interfaces belonging to a particular node.

Link-local scope: these addresses are valid only on the link to which the interface sending/receiving the packet is attached. Routers never forward such packets.

Global scope: these addresses are unique and addressable from anywhere in the Internet.

Note that there was a "site-local" address scope, but this has been deprecated by the IETF. This is probably good news for the satellite community, as many of the reasons for dropping this concept arose from ambiguities that would be particularly likely to occur in the kind of multi-customer, multi-national networks to which satellites are particularly well adapted because of their broad geographical coverage and lack of hierarchical structure. (See RFC 3879 [58]).

4.2.3 IPv6 address types

IPv6 introduces a number of different address types. In particular, the broadcast type of IPv4 is subsumed into a new multicast architecture. There is a new type, called anycast, which addresses any one of a number of interfaces. The other major type is unicast, which can exist as link-local or global (as mentioned above, an earlier type, site-local unicast, has recently been deprecated by the IETF). Certain prefixes have been defined as follows:

Address Type	Binary Prefix	IPv6 Notation
Unspecified	00...0 (128 bits)	::/128
Loopback	00...1 (128 bits)	::1/128
Multicast	11111111 (8 bits)	FF00::/8
Link-local unicast	1111111010 (10 bits)	FE80:/10
Global unicast	anything other than the above	

NOTE: Anycast addresses use the unicast address format.

4.2.3.1 Interface identifiers

Interface identifiers in unicast addresses identify interfaces on a link, and are required to be unique within a subnet prefix. In many cases, it will be derived directly from the interface's link layer address. It is worth noting that the uniqueness of interface identifiers is independent of the uniqueness of IPv6 addresses. For example, a Global Unicast address may be created with a local scope interface identifier and a Link-Local address may be created with a universal scope interface identifier.

For all unicast addresses, except those that start with the binary value 000, Interface IDs are required to be 64 bits long and to be constructed in Modified EUI-64 format. The details of this are described in RFC 4291 [55]. In clause 4.2.4, we describe how such interface identifiers can be generated from the BSM_ID.

The unspecified address, which consists of 128 bits set to zero (::) indicates the absence of an address, and is used, for example, in the source address field of IPv6 packets sent by a host that is in an initialization process prior to obtaining its address.

The address ::1 is known as the loopback address, and is used by a node to send a packet to itself, and is treated as having link-local scope.

The general format for Global Unicast addresses is as follows:

n bits	*m* bits	128 - *n* - *m* bits
Global Routing Prefix	Subnet ID	Interface ID

Where the global routing prefix is a value assigned to a site (typically this is hierarchically structured), the subnet ID identifies a link within the site, and the interface identifier will usually be a 64 bit, modified EUI-64 identifier (see [55]), although other types of interface identifiers are permissible. Given the broad scope of satellite coverage, these definitions may prove somewhat difficult to apply, as the concept of "site" may not be as meaningful as, for example, other node aggregations, such as "customer" in a scenario where a single satellite networks supports multiple corporate customers. Indeed, the concept of "site" has proved difficult for the IETF in general, with the result that the use of "site-local" identifiers has been deprecated
(see RFC 3879 [58]).

An address type of particular interest is the IPv6 address with embedded IPv4 address. Two such types have been defined. The first, the "IPv4-compatible IPv6 address" simply padded an IPv4 address to the left with zeroes, giving:

::d.d.d.d/96,

Where d.d.d.d is the embedded IPv4 address. This address type has now been deprecated by the IETF in favour of the following type of address.

The "IPv4-mapped IPv6 address" (see RFC 4291 [55]) prepends a 96 bit field consisting of 80 zeroes followed by 16 ones, to give:

::FFFF:d.d.d.d/96,

Where d.d.d.d is the embedded IPv4 address.

4.2.4 Generation of IPv6 address from the BSM_ID

The BSM SI-SAP Common Air Interface Specification, TS 102 357 [34] mandates IEEE 802 48 bit LAN MAC address format for BSM_ID, and an implementation may use them to create IPv6 interface identifiers due to their uniqueness properties. This process is specified in RFC 4291 [55], the IPv6 Addressing Architecture. In order to use the BSM_ID to generate a unique IPv6 address, it must first be extended to form an IEEE EUI-64 [56] identifier, which is 64 bits in length. This is then modified to form an IPv6 interface identifier. To generate the full 128 bit IPv6 address, the 64 bit interface identifier so created is prepended with a Global routing prefix and subnet ID, which together make up the most significant 64 bits of the address.

Since there are some particularities involved in creating the 64 bit IEEE EUI-64 identifier from the 48 bit LAN MAC address (the IETF does not follow the present IEEE standard exactly), we first look at how the 64 bit IEEE EUI-64 identifier is adapted to form the IPv6 interface identifier.

The only change needed to transform an IEEE EUI-64 identifier to an interface identifier is to invert the "u" (universal/local) bit. An example is a globally unique IEEE EUI-64 identifier of the form:

0 15	16 31	32 47	48 63
cccccc0gcccccccc	**ccccccccmmmmmmmm**	**mmmmmmmmmmmmmmmm**	**mmmmmmmmmmmmmmmm**

Where "c" represents a bit of the assigned company_id, "0" is the value of the universal/local bit to indicate universal scope, "g" is the individual/group bit, and "m" represents a bit of the manufacturer-selected extension identifier. The IPv6 interface identifier is be of the form:

0 15	16 31	32 47	48 63
cccccc1gcccccccc	**ccccccccmmmmmmmm**	**mmmmmmmmmmmmmmmm**	**mmmmmmmmmmmmmmmm**

Here the universal/local bit has been inverted.

We can now look at how the IEEE EUI-64 identifier is obtained from the BSM_ID. The IEEE guidelines on the EUI-64 identifier define a method to create an IEEE EUI-64 identifier from an IEEE 48 bit MAC identifier. This involves the insertion of two octets, with hexadecimal values of 0xFF in the middle of the 48 bit MAC (between the company id and vendor-supplied id). There is a similar approach to dealing with IEEE EUI-48 identifiers, but the octets 0xFF and 0xFE are employed. In earlier versions of the specification, this was incorrectly specified for the 48 bit MAC identifier as well, and so adopted as part of the IETF method for transforming 48 bit MAC identifiers into the interface identifier in an IPv6 address. Since the only requirement for IPv6 interface identifiers is that each one be unique on the link, it is immaterial whether FFFF or FFFE is used as the two octet insertion, and so the IETF has persisted with the use of FFFE for IEEE 48 bit MAC address conversion. Thus, as an example, take the 48 bit IEEE MAC address with Global scope.

0 15	16 31	32 47
cccccc0gcccccccc	**ccccccccmmmmmmmm**	**mmmmmmmmmmmmmmmm**

Where "c" represents a bit of the assigned company_id, "0" is the value of the universal/local bit indicating Global scope, "g" is the individual/group bit, and "m" represents a bit of the manufacturer-selected extension identifier. The interface identifier would be of the form:

0 15	16 31	32 47	48 63
cccccc1gcccccccc	**cccccccc11111111**	**11111110mmmmmmmm**	**mmmmmmmmmmmmmmmm**

As another example, consider the general 48 bit IEEE MAC address with Local scope:

0 15	16 31	32 47
cccccc1gcccccccc	**ccccccccmmmmmmmm**	**mmmmmmmmmmmmmmmm**

Where "c" represents a bit of the assigned company_id, "0" is the value of the universal/local bit indicating Global scope, "g" is the individual/group bit, and "m" represents a bit of the manufacturer-selected extension identifier. The interface identifier would be of the form:

0 15	16 31	32 47	48 63
cccccc0gcccccccc	cccccccc11111111	11111110mmmmmmmm	mmmmmmmmmmmmmmmm

Therefore, this approach should be taken with the BSM_ID, and allows the BSMS to generate unique interface identifiers within a subnet. Prepending the global and site identifiers (together making up the 64 MSBs of the address, yields a unique IPv6 unicast address).

The following figure shows the complete 128 bit address in the IPv6 header, showing how the interface identifier field derived above is embedded in the address.

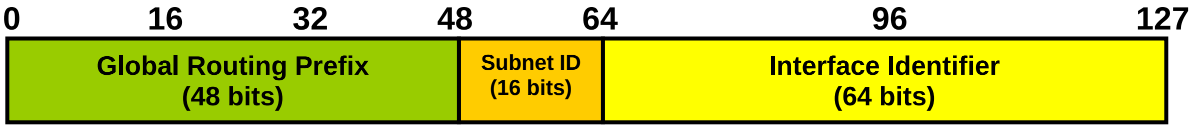


Figure 4.2.2: The IPv6 Address Field

4.2.5 Path MTU discovery

One of the significant differences between IPv6 and IPv4 is that IPv6 mandates a higher minimum value of 1280 for the Maximum Transmission Unit (MTU), which is the maximum link layer payload size that can be delivered by a network, such as the BSMS (see RFC 2460, the IPv6 Specification [11]). Thus, any link within the BSMS must offer a transmission service at the SI-SAP that transports IP packets of at least 1280 octets. (Clearly the Satellite Dependent implementation of this service may involve segmentation and reassembly of these packets into smaller units at lower layers, but this must be transparent at the SI-SAP.) RFC 2460 [11] adds a recommendation that the link MTU be at least 1500 bytes. IPv6 packets larger than the link's MTU have to be fragmented at the sending node and the frames have to be reassembled to the original IPv6 packets at the receiving node.

IPv6 tries to prevent fragmentation by intermediate nodes on the way from the IPv6 packet source to the destination by discovering the minimal MTU on the path from the sender to the receiver. This is known as the Path MTU (PMTU), and is set as the local MTU at the source node. In order to achieve this, the sending node must determine the PMTU before transmitting the IP packet.

PMTU discovery is done at the source, and may be achieved by sending ICMPv6 messages to the destination. The size of the ICMPv6 message is increased until a node replies with an ICMPv6 error message because the message does not fit in a single link layer packet. The ICMPv6 error message states that the ICMPv6 packet was too large and it includes the link's MTU. This is not a particularly efficient approach, and alternative methods are currently under consideration by the IETF.

4.2.6 IPv6 Neighbour Discovery (ND)

One of the key functionalities of IPv6 is Neighbour Discovery (ND) specified in RFC 2461 [66]. IPv6 network nodes apply Neighbour Discovery to discover their neighbouring nodes, for example, to determine which hosts and routers are available on-link, or to determine the link layer address of a specific neighbour node.

Neighbour discovery enables the following functionalities (see [66]):

- Router Discovery: Hosts detect routers attached to the link by applying Router discovery.
- Prefix Discovery: Nodes need to know the on-link prefixes to distinguish destinations that reside on-link from those only reachable through a router.
- Parameter discovery: This functionality enables nodes to learn parameters including the link MTU or the hop limit value.

- Address Autoconfiguration: used by nodes to automatically configure an IPv6 address for an interface.
- Address Resolution: process of obtaining information about the relation of link-layer address and IPv6 address of a neighbouring node.
- Next-hop determination: this algorithm is for mapping an IPv6 destination address into the IPv6 address of the neighbour to which traffic for the destination should be sent.
- Neighbour Unreachability Detection (NUD): enables nodes to determine that a neighbour is no longer reachable.
- Duplicate Address Detection (DAD): used by nodes to verify that an address it wishes to use is not already in use by another node.
- Redirect: a router sends a redirect message to inform a host of a better first-hop node to reach a particular destination.

In Neighbour Discovery five different message types are used:

- Router Solicitation: nodes send out Router Solicitations to query a router to reply with a Router Advertisement, including information for autoconfiguration.
- Router Advertisement: sent by routers periodically or as reply to a Router Solicitation message. Router Advertisements signal the presence of a router on-link and they include configuration parameters for the nodes attached to the link.
- Neighbour Solicitation: sent by nodes to discover other nodes on-link.
- Neighbour Advertisement: sent as response to Neighbour Solicitation messages.
- Redirect: messages sent by routers to inform a host of a better first hop for a specific destination address.

4.2.7 Duplicate Address Detection (DAD)

With IPv6 stateless address autoconfiguration (see below), nodes create IPv6 addresses for their interfaces automatically. Before assigning an address to an interface, the vIPv6 node evaluates its uniqueness by performing Duplicate Address Detection.

A node begins this process by multicasting a Neighbour Solicitation message on the link, with the tentative IPv6 address as target address, the unspecified address as source address, and the solicited-node multicast address as destination address. All IPv6 nodes on-link receive the message and determine whether they are already using the given tentative IPv6 address. If this is the case (DAD failure), the node replies with a Neighbour Advertisement message, with a destination address of the all-nodes multicast address. RFC 2462 [74] specifies that if a node performing DAD receives a Neighbour Advertisement as reply it abandons the automatic assignment of IPv6 addresses and manual configuration is required. RFC 3041 [65] specifies a more relaxed behaviour, in which a node creates an interface identifier and the corresponding IPv6 address randomly, repeating this process up to 5 times while the DAD process continues to fail before abandoning the attempt.

Issues regarding the use of DAD over links that are broadcast-capable are discussed in [46], and include the need for terminals to be able to differentiate & discard packets that they originate and have been subsequently re-broadcast over the L2 link.

4.2.8 Neighbour Unreachability Detection (NUD)

An IPv6 node actively tracks the reachability state of its neighbours. NUD is performed only for neighbours to which unicast packets are sent. For NUD, only the reachability on the forward path is of interest. Reachability can be obtained from upper layers, as, for instance, when a host receives a TCP acknowledgement it can be sure that the corresponding IPv6 packets have reached the next-hop. If reachability information is not provided by upper layers, a node actively probes its neighbours by sending them Neighbour Solicitation messages. After receiving a solicited Neighbour Advertisement message as reply, a node can consider the respective neighbour as reachable.

4.2.9 Router and prefix discovery

Router and Prefix Discovery is the process by which IPv6 nodes learn the address of the default router and the IPv6 prefixes that reside on-link.

Routers multicast Router Advertisement messages periodically to the all-nodes multicast address. The message contains a list of prefixes that reside on-link and the MTU of the link. Furthermore, a router signals via the Router Advertisement message that it is willing to be a default router by setting a Router Lifetime parameter greater than zero.

Having obtained a prefix, a node gets information about the range of IPv6 addresses that can be reached via the respective link without going beyond a router.

4.2.10 IPv6 stateless autoconfiguration

In IPv6 a stateful as well as a stateless autoconfiguration mechanism is specified. In the stateful method hosts request configuration parameters like IPv6 addresses explicitly from a server, which keeps track of all IPv6 addresses assigned to network nodes.

Stateless address autoconfiguration allows hosts to configure an interface without an additional server. Hosts apply autoconfiguration as specified in RFC 2461 [66]. Routers are expected to be configured in a different way (e.g. manually), but the automatic generation of link-local addresses and the performance of Duplicate Address Detection is assumed [66].

Stateless address autoconfiguration, see RFC 2462 [74] uses the mechanisms of Neighbour Discovery described above. A node creates a link-local address for one of its interfaces by prepending the default link-local prefix FE80::0 to the interface identifier. In most cases, the interface identifier is a 64 bit long modified EUI-64 format as described in RFC 4291 [55]. The EUI interface identifier can be derived from the MAC address, for example.

After receiving Router Advertisement messages containing prefix information, a node is able to create global IPv6 addresses for the interface the message is received from. To form a global IPv6 address the respective prefix is appended by an interface identifier. The prefix length and the interface identifier length must total 128 bits.

Before assigning these IPv6 addresses to an interface, DAD must be performed. If the DAD process fails, manual configuration is required. DAD expects full multicast-capable links.

Additionally, an IPv6 node gets default router information from a Router Advertisement. The IPv6 source address (link-local scope) of the router is included in the message which can be used as default router address. A router signals its willingness to be a default router by a *Router Lifetime* value greater than zero in the Router Advertisement message.

4.2.11 IPv6 stateful autoconfiguration (DHCPv6)

As an alternative to the stateless autoconfiguration specified in RFC 2461 [66], the Dynamic Host Configuration Protocol for IPv6 (DHCPv6) specified in RFC 3315 [18] provides a mechanism for stateful autoconfiguration. DHCPv6 enables configuration parameters such as IPv6 network addresses to be sent to IPv6 nodes. DHCPv6 can be used in combination with stateless autoconfiguration to complete the autoconfiguration process.

DHCPv6 is a client/server protocol. Clients and servers exchange DHCP messages using UDP. Clients listen for DHCP messages on UDP port 546, servers and relay agents listen for DHCP messages on UDP port 547. Clients use a link-local address obtained through IPv6 stateless autoconfiguration or addresses determined through other mechanisms as the source address in a DHCPv6 communication.

In most cases, a client multicasts DHCP messages to the All_DHCP_Relay_Agents_and_Servers address (FF02::1;2). This multicast address has link scope. It is used by a client to communicate with on-link relay agents and servers. All servers and relay agents are members of this multicast group and listen to messages sent to it. Issues arising from the use of DHCP (and DHCPv6) over broadcast media are considered further in draft-ietf-ipdvb-ar-05 [45]

If no DHCP server is available on-link, a DHCP relay agent on the client's link may relay messages between the client and server. The operation of the relay is transparent to the client.

4.3 Summary of main differences between IPv6 and IPv4

Here we attempt to summarize the most significant differences between IPv6 and IPv4. Whilst there are important differences, it is important to remember that IPv6 is still "just IP" so it can run alongside existing IPv4 and utilize the same protocols above and below IP.

- IPv6 has no header checksum.
- IPv6 has no broadcast traffic, instead using link layer multicast.
- The IPv4 ARP functionality is replaced by Neighbour Discovery in IPv6,which uses ICMPv6 and multicast.
- IPv6 includes Duplicate Address Detection by default.
- IPv6 nodes will generally be multi-addressed, at least with link-local and global unicast addresses.
- There is no IPv6 packet fragmentation done by routers; any fragmentation must be performed by the end nodes. Hence PMTU discovery is required for IPv6.
- It is not expected that NAT will be used for IPv6 (it would defeat one of the primary design objectives).
- IPv6 Privacy Addresses mean that IPv6 nodes may change their source addresses over time, even if deployed in a fixed, static network.
- IPv6's "infinitely large" subnets make network resizing exercises redundant, and deter classic port-scanning attacks.

5 Services and scenarios

5.1 Transitional service requirements - mixed IPv4/IPv6 environments

The following figure, taken from TR 102 155 [25], shows the various interworking configurations that we can consider in a mixed IPv4/IPv6 Internet.

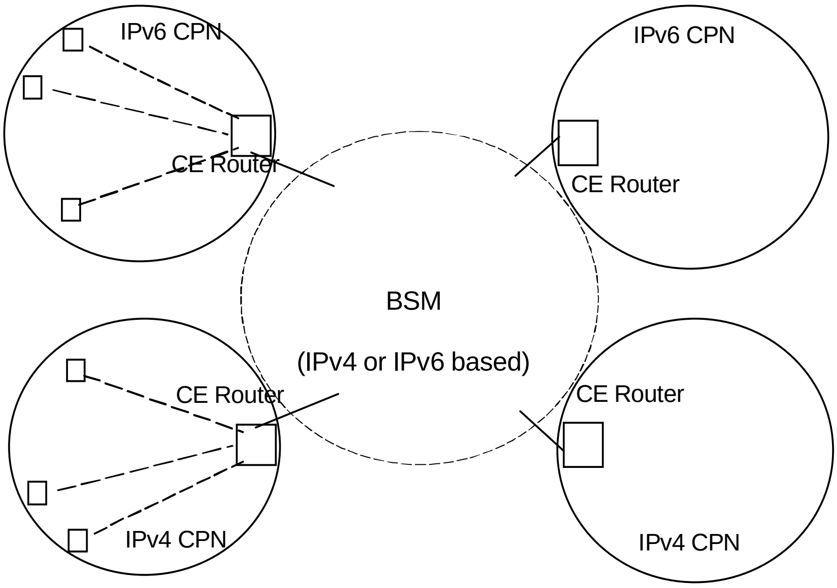


Figure 5.1.1: General IPv4 to IPv6 Interconnection Scenario across the BSM

Assumption 5, made in [25], was:

- "The range of situations that need to be considered for BSMS includes all permutations of IPv4 and IPv6 network interconnection through either an IPv4 or IPv6 based BSMS."

The BSMS itself can be IPv4-based or IPv6-based. Clearly, in the longer term, it is more sensible, both technically and commercially, to build IPv6-based systems. However, it is clear that today and for a number of years to come the Internet remains overwhelmingly IPv4-based, and so it may be simpler to persist with IPv4-based solutions in the near term. Against this, however, it must be noted that the development and deployment lead-times for satellite networks often extend to several years, and the operational lifetime of communications satellites is typically 10-15 years or more. Certainly if regenerative satellites are used, the choice of IPv4 over IPv6 may impact the payload, and condemn the BSMS in question to remain and IPv4-based system for up to 20 years beyond the design date. Even with the use of transparent satellites, designing a purely IPv4 baseband in the ground segment could be severely limiting. Should IPv6 roll-out occur relatively fast, complete replacement of a legacy IPv4 baseband would necessitate considerable time and additional cost.

Unless there are overriding technical or commercial considerations mandating the use of IPv4 in a particular case, it seems likely that, in the future, BSM networks will be designed to be as far as possible compatible with IPv6, as discussed below. This will lead us to examine the constraints placed upon such an IPv6-based BSMS by the need to maintain compatibility with IPv4. However, given that the vast majority of existing satellite infrastructure is IPv4-based, we will also look at how IPv4-based satellite networks can be used to support emerging IPv6-based services.

On the whole, the focus of most of the service- and applications-oriented work carried out in the IETF and elsewhere has been on this latter aspect of allowing IPv6-based services to operate in an essentially IPv4-based Internet. See, for example, RFC 3056 [4], RFC 4057 [61], RFC 4213 [67], RFC 4038 [72], RFC 3596 [75], RFC 2766 [77] and RFC 4215 [79].

5.2 BSM scenarios for IPv6

AN overview of BSM scenarios is given in the Technical Report TR 101 984 [23]: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; Services and Architectures". Various types of BSM IP networking scenarios with associated services are examined in TR 102 155 [25]: "Satellite Equipment and Systems (SES); Broadband Satellite Multimedia; IP Interworking over Satellite; Addressing and Routing" [25].

In what follows, we consider how IPv6/IPv4 interworking impacts these and related scenarios. In the first two cases, we envisage either an IPv4 or an IPv6 BSM network as the access network.

5.2.1 Satellite as access network IPv4 to IPv6

In this scenario, the BSMS connects IPv4 hosts to an IPv6 Distribution Network. The overall architecture is illustrated in figure 5.2.2, showing both the connection of Intranets via border routers and the direct connection of IPv4 hosts. In this example, some form of address translation or dual stack architecture is required in some or all of the BSM STs. The underlying BSM architecture could be either star or mesh. In the star case, it would be natural for the hub station to be the sole access from the BSM Access Network to the IPv6 Distribution Network (ST 4 in the figure). In that case, it would be logical to co-locate the IPv4/IPv6 interworking functions here, and leave the rest of the BSM Network as pure IPv4. These questions will be explored further in clause 6 on Routing and Addressing.

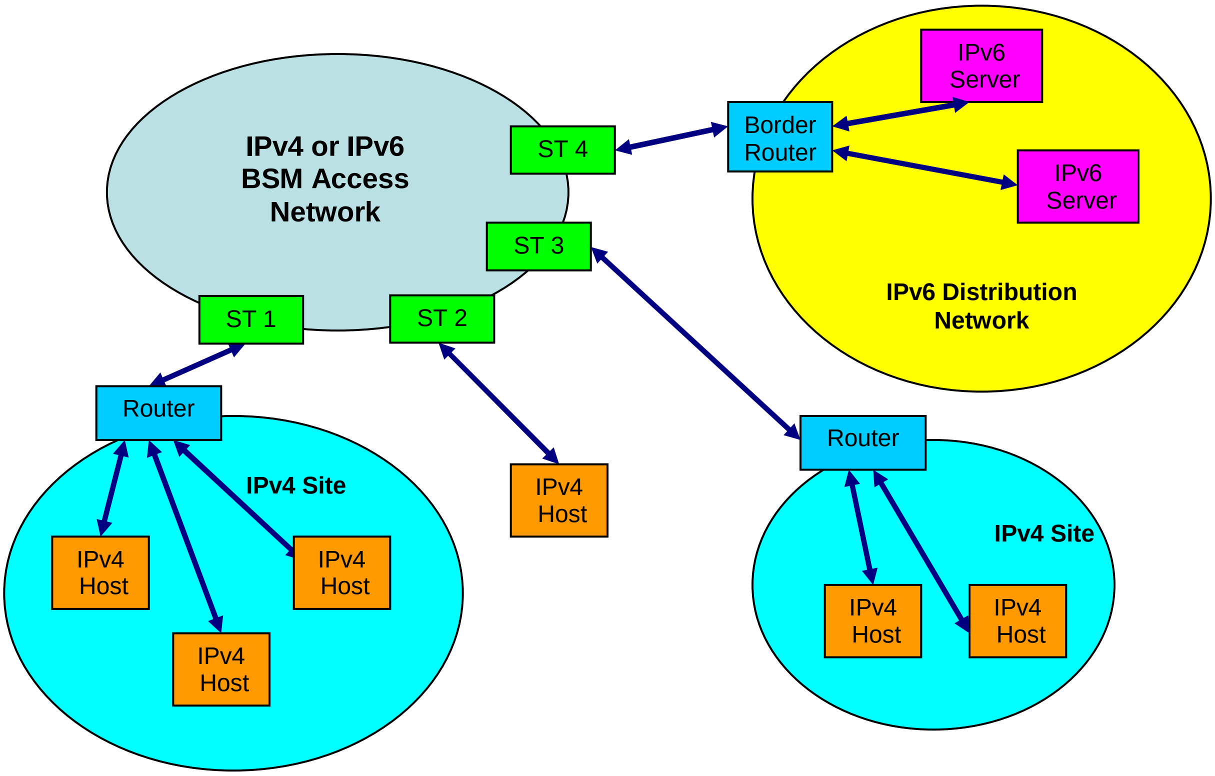


Figure 5.2.1: BSM as Access Network for IPv4 Hosts to IPv6 Distribution Network

5.2.2 Satellite as access network IPv6 and IPv4 to IPv4 (mixed case)

In the short term, it may be more likely for IPv6 islands to require access to legacy IPv4 infrastructure. In this case, an equivalent architecture to that in clause 5.2.1 might be appropriate, but it is more likely to be embedded in a mixed IPv4/IPv6 Access Scenario, as shown in figure5.2.3. Here there are a number of architectural choices to be made regarding location of dual stack or translation mechanisms. The simplest is for all STs to have equivalent capabilities, and for the dual stack/translation function to be performed at the edges of the BSM network.

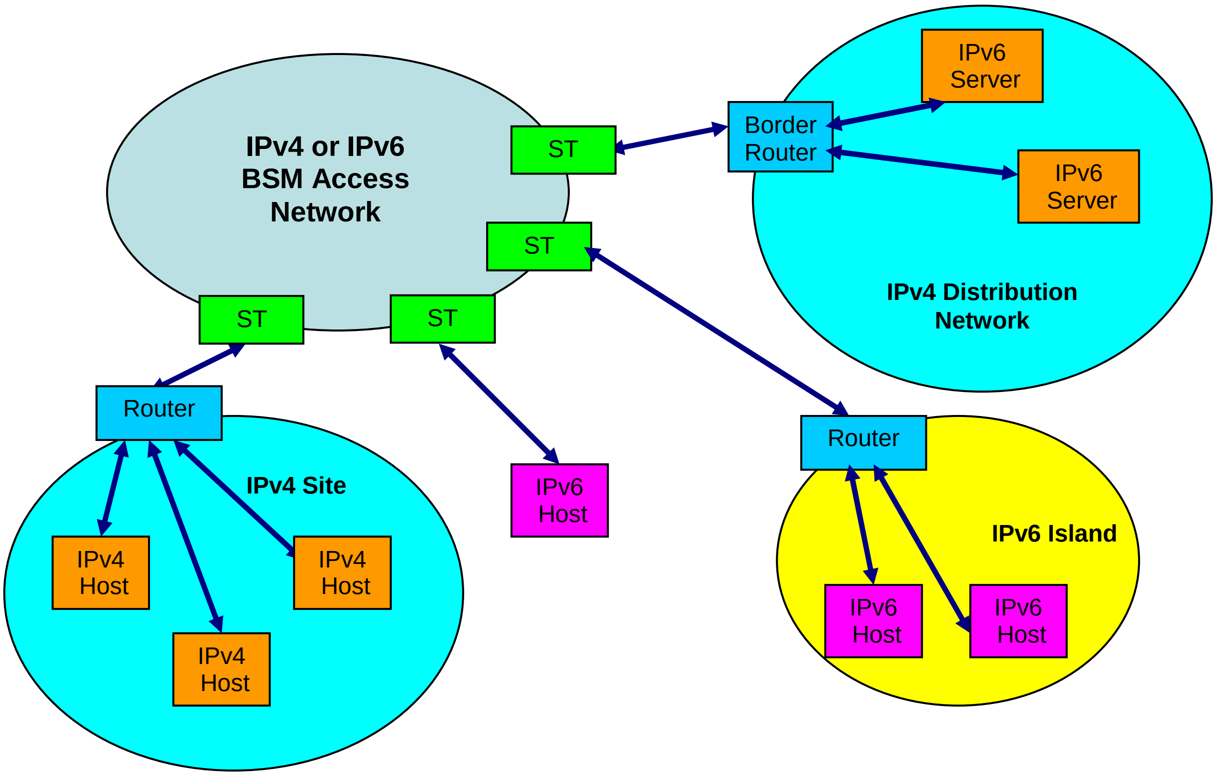


Figure 5.2.2: BSM as Access Network for IPv4 and IPv6 Hosts to IPv4 Distribution Network

5.2.3 Satellite as L2 forwarder or interconnect between IPv4 and IPv6 clouds

This scenario relies on providing a transparent (to the IP layer) layer 2 interconnection between IPv4 and IPv6 islands. Thus it relies upon suitably enabled routers at ingress and egress of the BSMS (above the SI-SAP) to perform the appropriate IPv4 <-> IPv6 interconnection (either through dual stack, tunnelling or translation). We do not consider it further in the present document.

5.2.4 SOHO Internet access and residential access using tunnels

In TR 102 155 [25], SOHO Internet Access is considered as a BSMS service, using a PPPoE tunnelling approach (clause 6.2.5.1 of [25]). In clause 6.2.5.2 of the same document, residential access using a similar approach is considered. The two following figures, taken from [25], illustrate the approach.

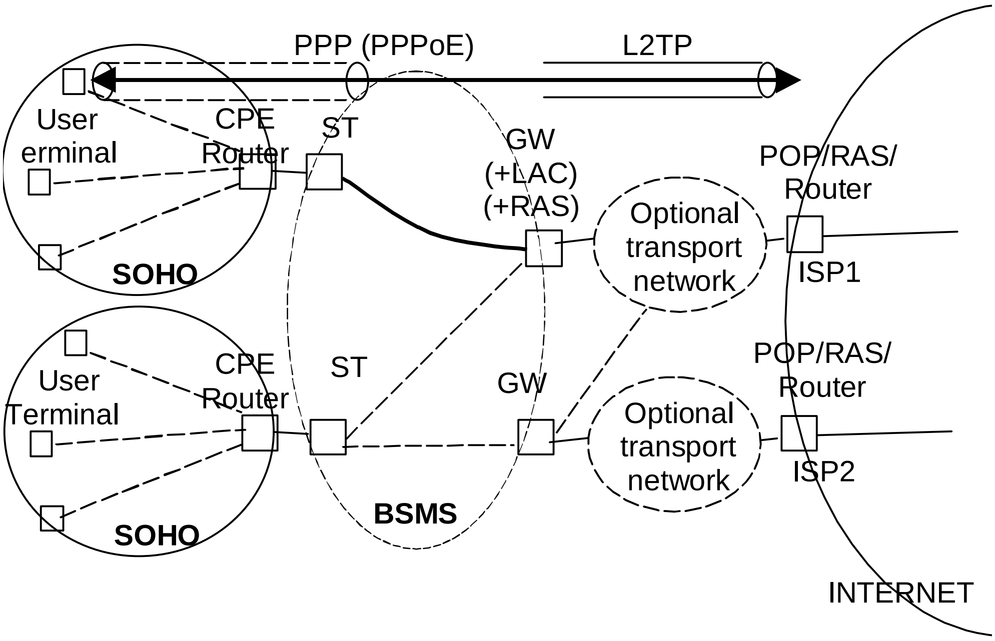
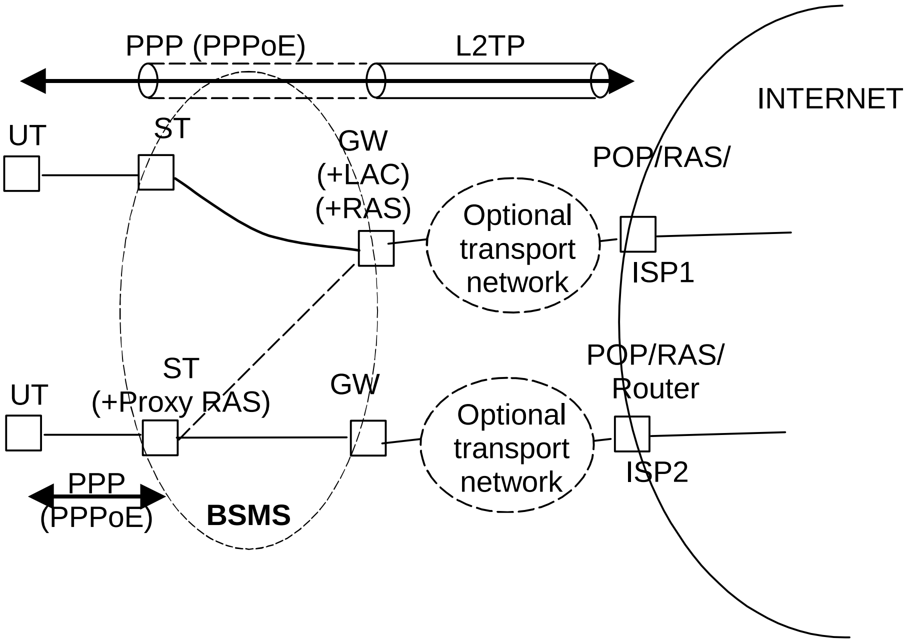


Figure 5.2.3: Configuration of BSMS-based SOHO access to the Internet



NOTE: Dashed lines in the figure represent optional items (e.g. an ST can connect to alternative Gateways).

Figure 5.2.4: Example of residential access scenario

These scenarios both envisage the use of PPP tunnels, which are affected by the transition to IPv6. RFC 2472 [53] describes how IPv6 packets can be tunnelled over PPP. Two aspects in particular need to be noted: the specification of the IPv6 header-compression protocol and the interface-identifier negotiation. Any ST that supports IPv6 PPP tunnelling must be compatible with these options. This will be further considered in clause 6.1.2 (Tunnelling).

6 Routing and address resolution issues

6.1 Introduction

A number of architectural approaches have been considered for the transition from IPv4 to IPv6. These fall into three basic classes: dual-stack configurations, IPv6/IPv4 tunnels and IPv6/IPv4 address translation. Any of these mechanisms could be used in a BSMS. In the context of the BSM SI-SAP architectural model, these would be part of the SIAF (see figure 6.1.1, which is taken from TR 101 984; ref. [23], "BSM Services and Architecture").

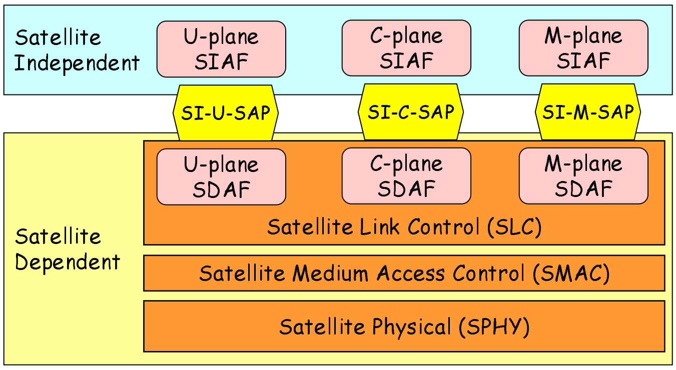


Figure 6.1.1: Satellite Independent Service Access Point (SI-SAP)

However, the precise architectural role of each one would be different. We consider each case separately in the following clauses.

6.1.1 Dual stack approach

In this method, each ingress/egress node to the BSM would contain two parallel protocol stacks, one for IPv4 communications and another for IPv6. In a BSM architecture in which there is no internal Layer 3 processing (a "transparent" satellite system, such as a DVB-based star using ULE to encapsulate IP), incoming IPv6 packets would be handled using the IPv6 stack at the ingress point, and upon reception at the egress ST, would be passed to the IPv6 stack for presentation at the SI-SAP. Similarly, IPv4 packets would remain within IPv4 stacks at both ingress and egress. Where regenerative architectures are concerned, the picture is more complicated: whilst it is conceivable that this architecture could be maintained into the core satellite network, with, for example dual IPv6/v4 routers on board the satellite, such an architecture would be technically and economically infeasible. Therefore, somewhere either above or below the SI-SAP, there must be convergence of the two stacks onto a common bearer service architecture.

6.1.1.1 Convergence below the SI-SAP

In the case that the convergence of the two protocol stacks occurs within the Satellite Dependent Adaptation Function (SDAF), it gives rise to the following generic structure, as shown in fig 6.1.2. In dual stack architecture, there are two possible approaches at a routing level. If routing from one IPv6 domain to another, or form one IPv4 domain to another, there can be effective separation of IPv4 and IPv6 traffic; that is to say that an IPv6 ingress packet will become an IPv6 packet at egress, with IPv4 packets treated similarly. This implies that there is a convergence function for IPv6 packets and another for IPv4 packets, together with a discrimination function to determine at the egress terminal to which of these functions a packet received over the air interface should be directed. However, these functions must also be able to perform an address translation function in the case of IPv6/IPv4 domain interworking. In this case, an incoming IPv4 packet will be given the IPv4-mapped IPv6 address described in clause 4.2.1 (::FFFF:d.d.d.d/96) upon egress into an IPv6 domain. An incoming IPv6 packet will require an IPv4 address resolution function to obtain an egress IPv4 address.

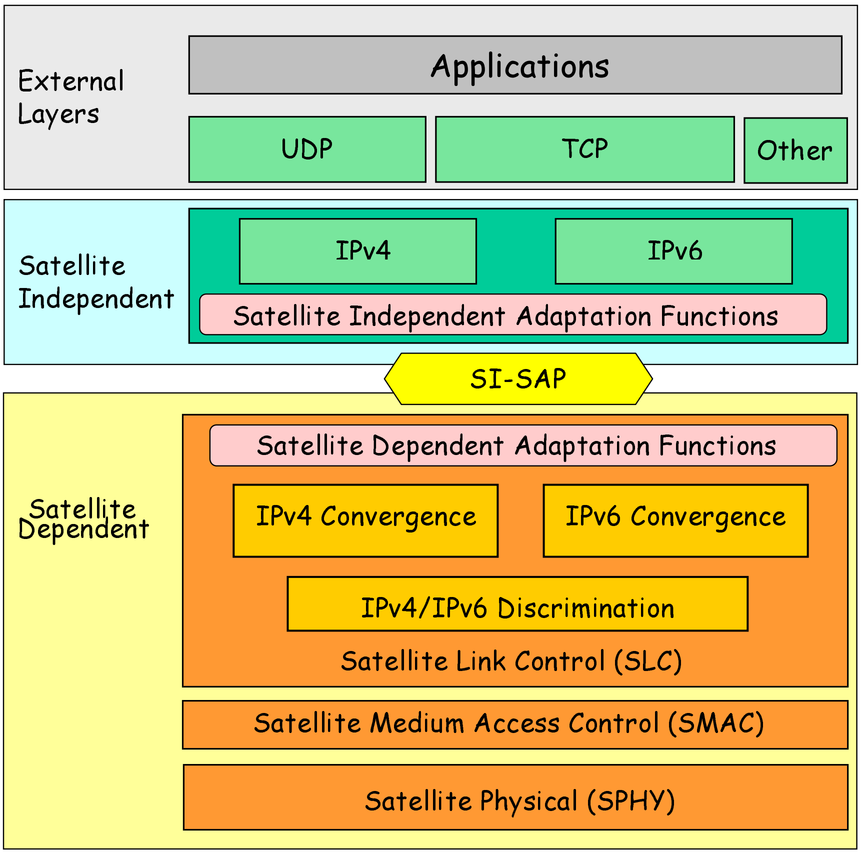

External Layers
Applications
UDP
TCP
Other
Satellite Independent
IPv4
IPv6
Satellite Independent Adaptation Functions
SI-SAP
Satellite Dependent
Satellite Dependent Adaptation Functions
IPv4 Convergence
IPv6 Convergence
IPv4/IPv6 Discrimination
Satellite Link Control (SLC)
Satellite Medium Access Control (SMAC)
Satellite Physical (SPHY)

Figure 6.1.2: Dual Stack Architectural Overview (convergence below SI-SAP)

To examine the U-plane and C-plane functionality associated with address resolution in greater detail, we refer to the BSMS protocol reference model as defined in TR 101 985 [24]. This model defines two components, as follows:

- The address resolution function in the C-plane. This function is used to determine the satellite link address when the address translation is unknown. The results of address translation are stored in the cache for future use.
- The satellite address mapping function in the U-plane. This function maps the IP address to the corresponding satellite link address (e.g. a Satellite MAC address). This function makes use of an address cache which stores the address pairs.

The following figure is taken from [24].

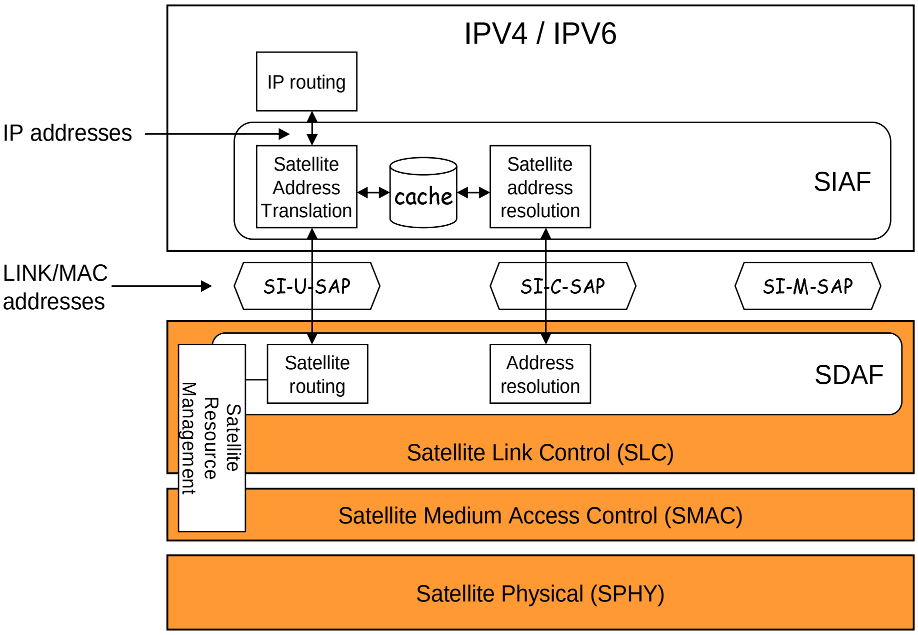

IPV4 / IPV6
IP routing
IP addresses
Satellite Address Translation
cache
Satellite address resolution
SIAF
LINK/MAC addresses
SI-U-SAP
SI-C-SAP
SI-M-SAP
Satellite routing
Address resolution
SDAF
Satellite Resource Management
Satellite Link Control (SLC)
Satellite Medium Access Control (SMAC)
Satellite Physical (SPHY)

Figure 6.1.3: Address and Routing reference model

In order to accommodate the dual stack architecture, this model must be modified as shown in figure 6.1.4 .

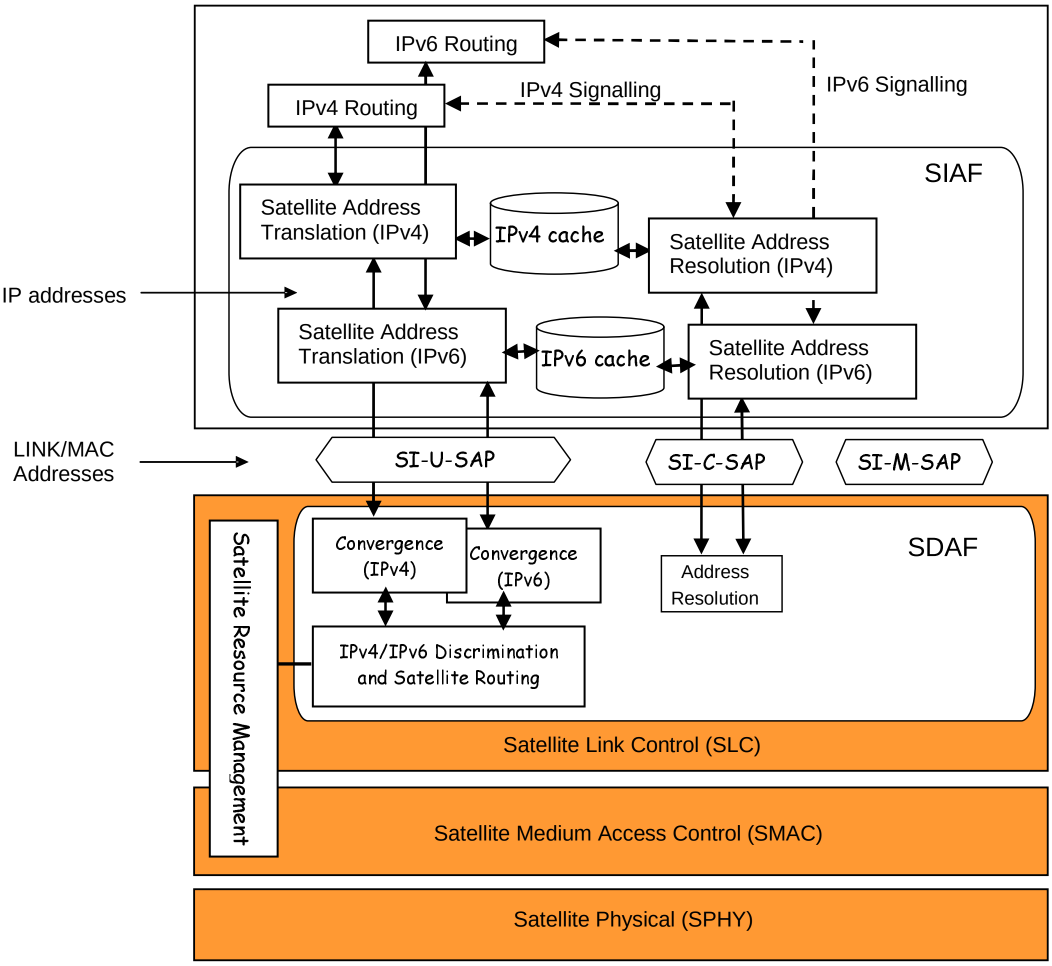

IPv6 Routing
IPv4 Routing
IPv4 Signalling
IPv6 Signalling
SIAF
Satellite Address Translation (IPv4)
IPv4 cache
Satellite Address Resolution (IPv4)
IP addresses
Satellite Address Translation (IPv6)
IPv6 cache
Satellite Address Resolution (IPv6)
LINK/MAC Addresses
SI-U-SAP
SI-C-SAP
SI-M-SAP
SDAF
Convergence (IPv4)
Convergence (IPv6)
Address Resolution
IPv4/IPv6 Discrimination and Satellite Routing
Satellite Resource Management
Satellite Link Control (SLC)
Satellite Medium Access Control (SMAC)
Satellite Physical (SPHY)

Figure 6.1.4: Dual Stack Address and Routing Reference Model

Here SIAF U-plane and C-plane functions have been replicated for IPv4 and IPv6 traffic and signalling. Note that the IPv4 and IPv6 address resolution C-plane functions must handle the IPv4 and IPv6 signalling protocols respectively. Thus, the IPv4 function will employ, for example, a protocol such as DHCP for address acquisition, while the IPv6 function might use stateless address autoconfiguration or DHCPv6. This is indicated by the dashed arrows in the figure. In practice, this architecture really corresponds to a dual-SI-SAP architecture, in which all SIAF functions are replicated between their IPv4 implementation and their IPv6 implementation. The convergence/discrimination between the two stacks is performed in the SDAF.

DNS resolution for a dual stack terminal does not necessarily require an IPv6-enabled DNS server. In other words, communication with the DNS server may be over IPv4, but the server must maintain "AAAA" records for IPv6 addresses, as well as "A" records for the IPv4 addresses it holds (see clause 6.3).

6.1.1.2 Convergence above the SI-SAP

In the case that the convergence of the two protocol stacks occurs within the Satellite Dependent Adaptation Function (SDAF), it gives rise to the following generic structure, as shown in fig 6.1.3. This has no impact below the SI-SAP.

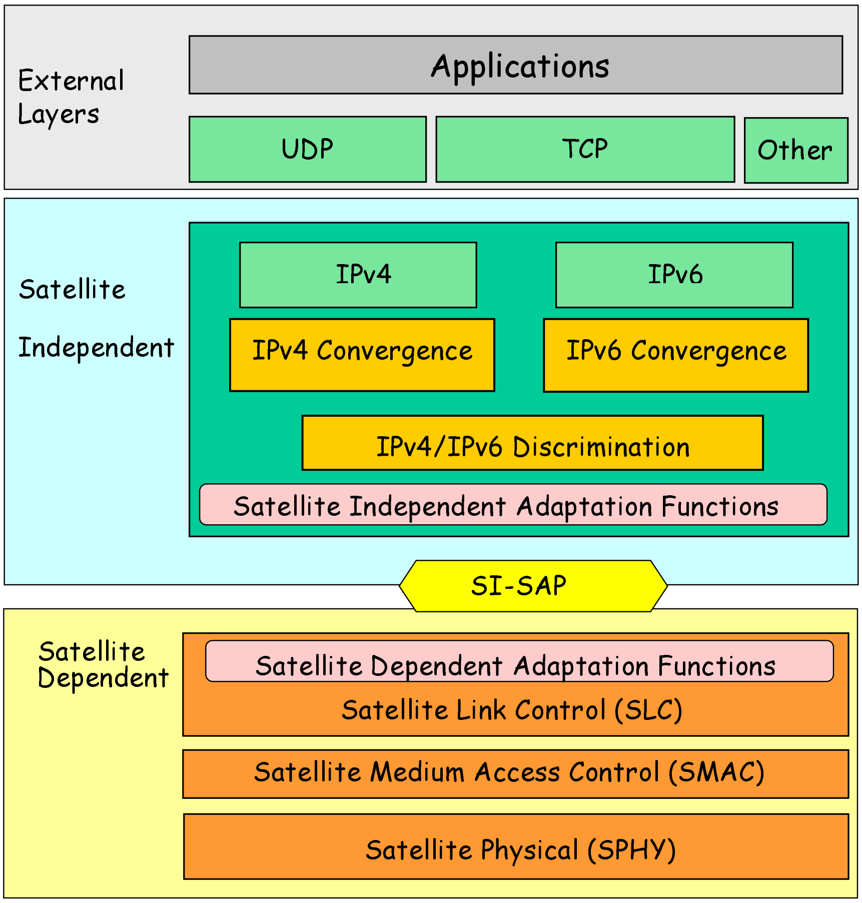

External Layers
Applications
UDP
TCP
Other
Satellite Independent
IPv4
IPv6
IPv4 Convergence
IPv6 Convergence
IPv4/IPv6 Discrimination
Satellite Independent Adaptation Functions
SI-SAP
Satellite Dependent
Satellite Dependent Adaptation Functions
Satellite Link Control (SLC)
Satellite Medium Access Control (SMAC)
Satellite Physical (SPHY)

Figure 6.1.5: Dual Stack Architectural Overview (convergence above SI-SAP)

6.1.2 IPv6/IPv4 tunnelling approach

In this approach, incoming IPv6 packets are encapsulated in IPv4 packets (after, if necessary being fragmented to respect the MTU of the IPv4 network segment), then routed over IPv4 infrastructure to another IPv6 node where reassembly (if necessary) and decapsulation is performed before the IPv6 packets are delivered to their destination. The generic architecture for this case is shown in figure 6.1.5. Although this figure indicates that there is no impact below the SI-SAP, this is, in fact, misleading. Whilst there are no additional IPv6-specific functional entities required in the SDAF, the fact that the satellite sub-network is transporting IPv6 packets may impose modifications or constraints on existing satellite dependent functions. The precise nature of this impact will vary according to the underlying satellite network architecture and may be quite sensitive to implementation details. For example, in a star architecture network with limited or no return channel, the unidirectionality of outbound links may impact both the User Plane and Control Plane functionality below the SI-SAP.

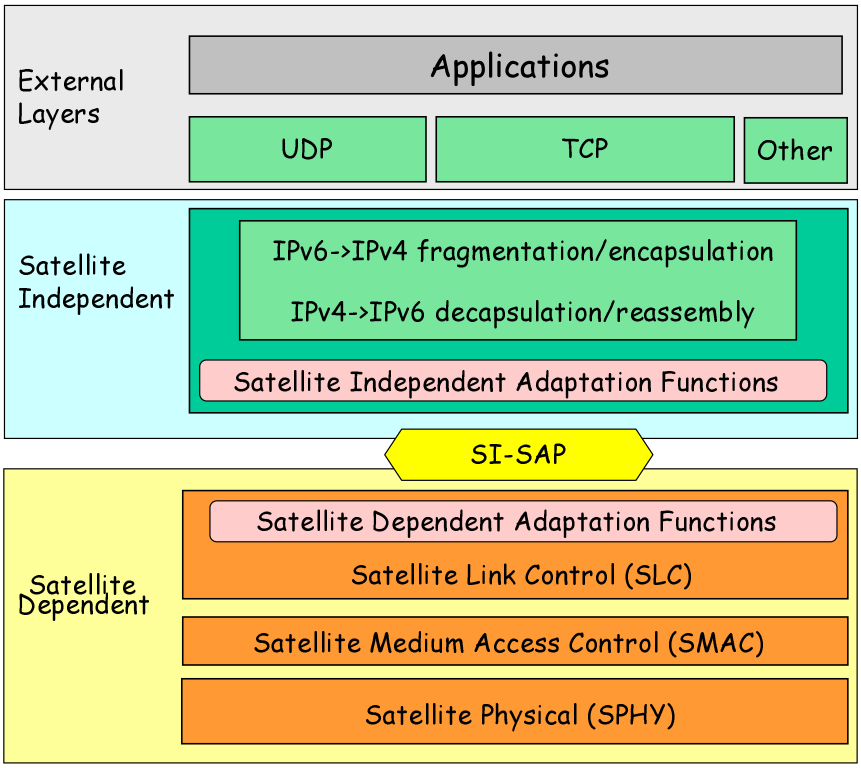

External Layers
Applications
UDP
TCP
Other
Satellite Independent
IPv6->IPv4 fragmentation/encapsulation
IPv4->IPv6 decapsulation/reassembly
Satellite Independent Adaptation Functions
SI-SAP
Satellite Dependent
Satellite Dependent Adaptation Functions
Satellite Link Control (SLC)
Satellite Medium Access Control (SMAC)
Satellite Physical (SPHY)

Figure 6.1.6: IPv6/IPv4 Tunnelling Architectural Overview

We are assuming in the above that the tunnel is restricted to the BSM subnetwork. It is also conceivable that the tunnels might extend out beyond the BSM subnetwork to other IPv4 nodes. This would considerably complicate the functionality required of the BSM. In particular, handling ICMPv4 error messages, hop counts, MTU considerations, etc. would require a more sophisticated C-plane functionality. Similarly, the tunnel address management function would need to interact with address management entities outside the BSMS.

In 2000, RFC 2893 [48], specified two methods of tunnelling IPv6 packets over and IPv4 infrastructure, "automatic tunnelling" and "configured tunnelling". In 2005, RFC4213 [67] obsoleted [48], and in the process deprecated automatic tunnelling. In particular, the IPv6-compatible address, ::d.d.d.d/96, where d.d.d.d is the IPv4 address of the tunnel endpoint, was abandoned. RFC 4213 [67] retains the configured tunnelling method, which is described in what follows.

6.1.2.1 Configured tunnelling

In contrast to automatic tunnelling, configured tunnelling as described in [67] does not require any specific relationship between the IPv4 and IPv6 addresses of a tunnelled IPv6 packet. This means that the tunnel end points in a configured tunnelling scenario can be located not only on the sending and receiving hosts, but also on intermediate gateways and routers located on the path between sender and receiver. Therefore, with configured tunnelling is possible to connect two isolated IPv6 island via a legacy IPv4 backbone by a single configured tunnel, which is to be established between two routers located at the respective border of their IPv6 island to the legacy IPv4 backbone.

While configured tunnelling is more interesting for scenarios with a shortage of global IPv4 addresses, the tunnels have to be pre-configured, which may involve significant management overheads.

One solution to ease the management effort involved in configured tunnelling is the use of a "tunnel broker", which is used to automate the establishment of configured tunnels. The concept of "IPv6 Tunnel Broker" is described in RFC 3053 [19]. The Tunnel Broker supports the automatic setup of tunnels between a requesting client and available tunnel servers served by the Tunnel Broker. The overall context for the use of the Tunnel Broker is shown in figure 6.1.7.

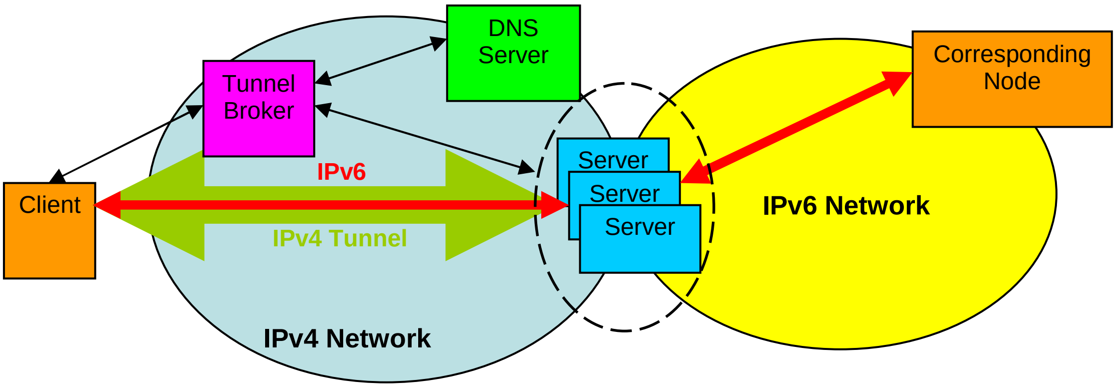


Figure 6.1.7: Concept of the Tunnel Broker

When the client requests the setup of a new tunnel from the Tunnel Broker, the Tunnel Broker first authenticates the client by means such as RADIUS, and establishes a secure connection between itself and the client. Over this secure connection, the client then provides the information required to establish a tunnel, including its IPv4 tunnel address, the DNS name for its new IPv6 tunnel address, whether it is a host or router, and if it is a router, how many IPv6 hosts it serves. Knowing this information, the tunnel broker will assign an IPv6 tunnel endpoint address to the requesting client host, and additionally an IPv6 prefix to be used by a requesting client router. It also informs the client about the peer tunnel address to be use, which is located on one of the tunnel servers served by the tunnel broker. The tunnel broker also initiates tunnel establishment on the selected tunnel server, and registers the client's DNS name. After finalizing these configuration steps, the client and possible nodes behind it can access the IPv6 network behind the tunnel server via a configured tunnel to the tunnel server.

This mechanism is well suited to the connection of isolated IPv6 hosts or small subnetworks. In the context of the BSM, it corresponds well to an IPv4-based satellite subnetwork offering an access service to IPv6 islands. In this case, it would be logical to place the Tunnel Broker in the NCC and use appropriate Satellite Dependent C-plane signalling to manage the tunnel allocation process.

6.1.2.2 6to4

"6to4" is another mechanism for tunnelling IPv6 packets over IPv4 networks. It is described in RFC 3056, "Connection of IPv6 Domains via IPv4 Clouds" [4]. As with automatic tunnelling, 6to4 requires a specific address format including the IPv4 address of a tunnel endpoint, but this time the IPv4 address is included in the prefix part of the IPv6 address (see clause 4.2 above). Figure 6.1.7 illustrates the 6 to 4 address format. All 6to4 addresses begin the prefix with hexadecimal 2002 in the first 16 bits, followed by the 32 bit IPv4 address. These 48 bits together represent the IPV6 6to4 prefix, reachable via a 6to4 tunnel with the respective IPv4 address, thus using 6to4 one global IPv4 address can be assigned a ::/48 IPv6 prefix.

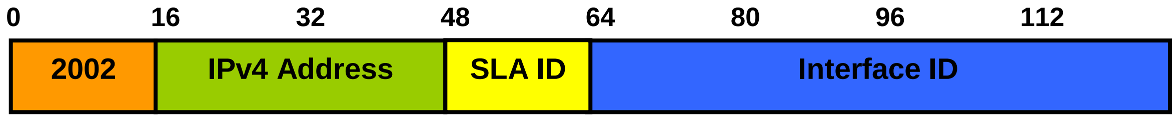


Figure 6.1.8: 6to4 Address Format

Figure 6.1.8 shows a typical 6to4 application scenario, with several IPv6 capable networks connected over a legacy IPv4 network. Each of these IPv6 networks has a 6to4 Border Router at the respective connection to the IPv4 network. A 6to4 prefix is derived, based on the global IPv4 address of this Border Router, and is assigned to the IPv6 nodes inside the 6to4 site. If now IPv6 nodes located in different 6to4 sites want to communicate with any of these nodes, they make use of their 6to4 addresses. Once a packet leaves a 6to4 site via its 6to4 Border Router, the Border Router is able to tunnel the packet to the IPv4 address of the remote 6to4 Border Router, as this IPv4 address is contained in the prefix part of the 6to4 based destination address of the tunnelled IPv6 packet.

When a packet has to be sent to the native IPv6 Internet, which is not addressed with 6to4 addresses, a 6to4 router placed on the border of the native IPv6 Internet may as the required relay. In this case, a tunnel from each of the 6to4 Border Router to the 6to4 relay has to be pre-configured.

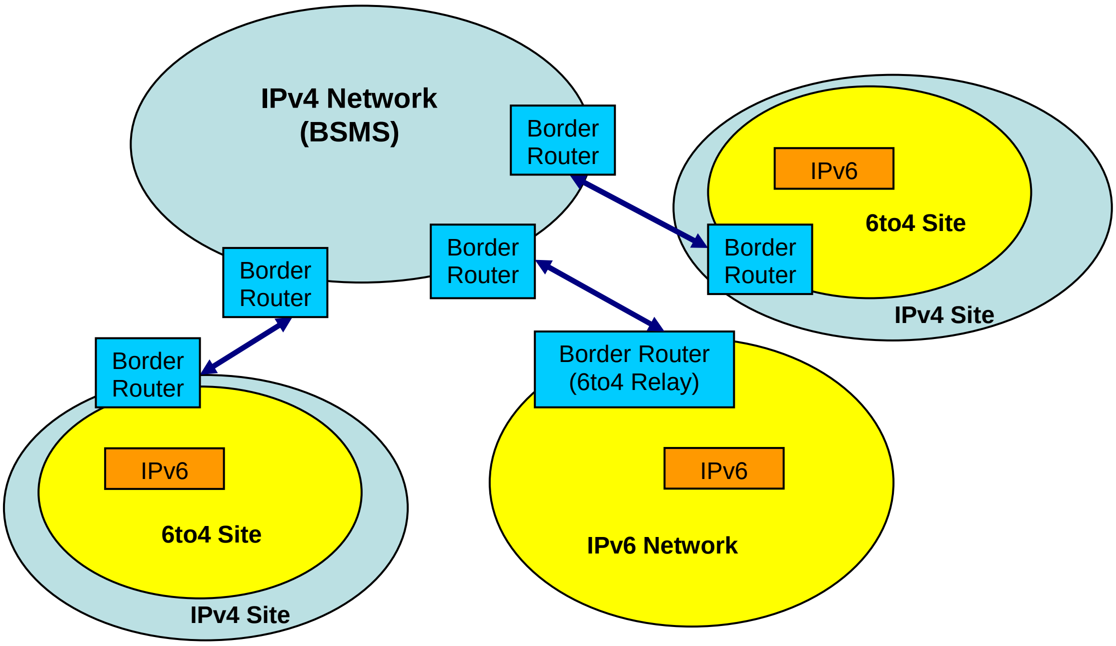


Figure 6.1.9: 6to4 tunnelling

While 6to4 can also be used for connecting single hosts or small sites, it has been designed to support larger sites. It has the important feature of enabling the interconnection of large IPv6 networks but requiring only a single global IPv4 address for each. Furthermore, the tunnelling between 6to4 routers is done automatically and requires no pre-configuration.

6.1.2.3 6over4

RFC 2529, "Transmission of IPv6 over IPv4 Domains without Explicit Tunnels" [3], specifies a mechanism allowing two IPv6 hosts isolated in a legacy, IP Multicast-capable IPv4 network automatically to configure tunnels to communicate with each other.

The idea behind this concept is to treat the IP Multicast IPv4 network from the perspective of the isolated IPv6 hosts as an IP-Multicast-capable virtual link layer. On top of this IP-Multicast-capable link layer, the IPv6 hosts make use of their standard Neighbour Discovery Mechanisms in order to resolve the address of the IPv6 peer on the virtual link layer, that is, the IPv6 peer's IPv4 address. Knowing the IPv6 peer's IPv4 address allows the automatic generation of a tunnel between the two isolated IPv6 hosts, over which they can send their IPv6 traffic.

While 6over4 provides effective abstract model for using the underlying IPv4 network as a virtual link layer for IPv6 communication, and allows the setup of tunnels without requiring specific address formats or pre-configuration, it still requires the IPv6 nodes using it to have their own IPv4 address, routable in the connecting IPv4 network. Furthermore, it requires the connecting IPv4 network to be IP Multicast- capable. Thus, this technique has limited applicability in large parts of the IPv4 Internet.

However, where there is the availability of an appropriate IPv4 address space, the method can be usefully applied. In particular, the BSM, being inherently multicast-capable provides a suitable underlying virtual link layer. One application scenario is corporate networks using private IPv4 addressing. In this case, a 6over4 router can be located at the border of the native IPv6 Internet. Isolated IPv6 hosts inside the corporate network make then use of 6over 4 in order to tunnel their IPv6 traffic through the Ip4 corporate network, supported over the BSMS, towards the 6over4 router, which will finally forward the IPv6 packets to the attached native IPv6 Internet, or destination corporate IPv6 node.

6.1.3 IPv6/IPv4 address translation approach

Here, the addresses of incoming IPv6 packets are translated into corresponding IPv4 addresses and the reformatted packets sent across the BSM subnetwork. Upon reception at the BSM egress node, the address is retranslated into the original IPv6 destination address and the appropriately reformatted packet forwarded to its destination. As far as the User Plane is concerned, the impact on the BSM functional architecture is limited to the SIAF, above the SI-SAP. There are variants on this method regarding selection or generation of IPv4 addresses that may impact the C-plane address resolution function of the BSM, however. In figure 6.1.4, we show the generic functional architecture for the IPv4/IPv6 address translation approach.

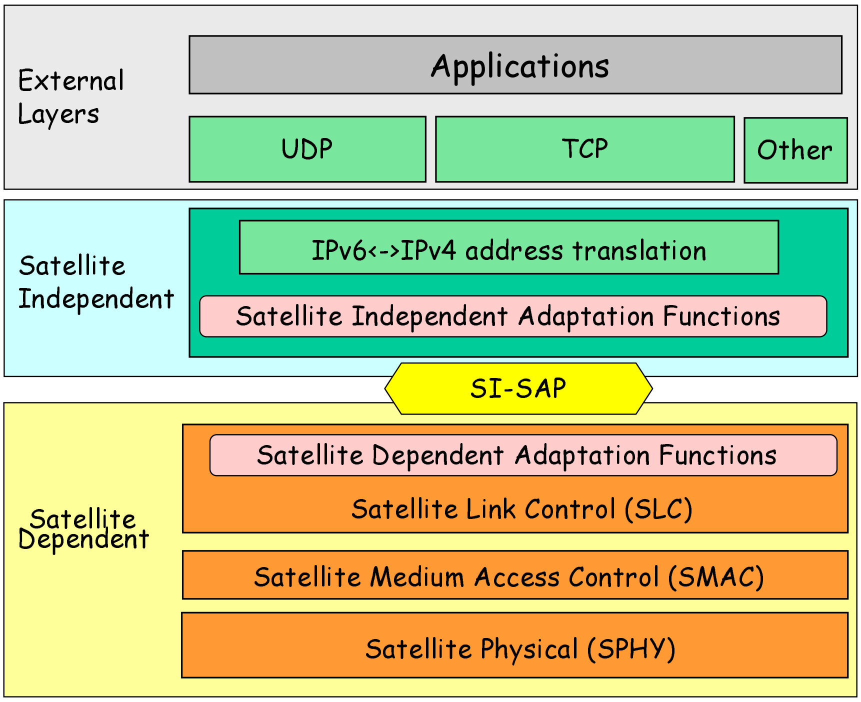

External Layers
Applications
UDP
TCP
Other
Satellite Independent
IPv6<->IPv4 address translation
Satellite Independent Adaptation Functions
SI-SAP
Satellite Dependent
Satellite Dependent Adaptation Functions
Satellite Link Control (SLC)
Satellite Medium Access Control (SMAC)
Satellite Physical (SPHY)

Figure 6.1.10: IPv6/IPv4 Address Translation Overview

Both the dual-stack approach and the tunnelling methods described above suffer from certain drawbacks; on the one hand, dual stack architectures require the nodes to have IPv4 addresses, and therefore offer no relief to the dynamic driving the transition to IPv6, namely pressure on the IPv4 address space. Tunnelling may involve considerable overheads in bandwidth, which is a precious commodity in satellite networks. Therefore, address translation is an option that must be studied carefully for the BSMS. A further advantage of this approach is that it offers a simple method of directly interconnecting IPv4-only nodes with IPv6-only nodes. (Although such interconnection begs the question of whether the applications are IP-protocol agnostic; many, such as FTP, rely on IP-layer information to interoperate, and so remain unsuitable for such heterogeneous interoperation.)

6.1.3.1 SIIT and NAP-PT

NAT-PT (Network Address Translation - Protocol Translation) is proposed as an IPv4/IPv6 translation mechanism in RFC 2766 [77]. As its name implies, NAT-PT translates not only between IPv4 and IPv6 addresses but also addresses issues of translation between protocol aspects of the two standards. RFC 2766 [77] also specifies how NAT-PT supports IPv6 nodes for resolving names in the IPv4 domain and vice versa. Although RFC 2766 [77] specifies the procedure of address translation, it refers for the procedure of protocol translation to RFC 2765, "Stateless IP/ICMP Translation Algorithm (SIIT)" [68]. In particular, SIIT defines mappings between ICMPv4 and ICMPv6 messages. Any BSMS supporting this form of address translation must implement a protocol translation engine at appropriate point in the network. Alternative architectures for accomplishing this will be considered below.

Whist the SIIT protocol translation is stateless, address translation as specified in NAT-PT is performed statefully, i.e. IP packets belong to a single flow crossing a NAT-PT box on the border between IPv4 and IPv6 networks need to return via the same NAT-PT box.

Furthermore, the NAT-PT translation functionality focuses on the translation of addressing information contained in the IP header of a packet. However, as mentioned above, applications may make use of IP information inside the payload of the packet, as in FTP and DNS. In order to permit interworking between IPv4 instantiations of these applications and IPv6 instantiation, it is necessary to implement Application Level Gateways (ALGs) collocated with NAT-PT boxes. These map between the IPv4 and IPv6 domains at an application level. As they are application specific, there is a danger of considerable additional software in a network offering network translation. Since it may also be necessary to process SIP, RSVP and other C-plane messages, this may become a significant burden. Nevertheless, the bulk of this additional development is not likely to be satellite-specific, and so its impact on BSM may not be too significant.

Another potential problem with protocol and address translation that it breaks the end-to-end transparency of the Internet. Reinforcement of this principle was one of the Left's main motivations in introducing IPv6.

6.1.3.2 Use of proxies

Another approach to providing an IPv4/IPv6 translation mechanism is the use of a proxy server. For example, if an IPv6 client wished to access an IPv4 server, at the border between the IPv6 and IPv4 network a proxy server is located that takes the client requests in the first instance. The proxy server thereby terminates the IPv6 connection from the client, and initiates a communication with the real server using IPv4. The real server replies via IPv4 to the proxy server, which terminates the IPv4 connection and forwards the reply via a separate IPv6 connection back to the client.

Again, such proxy server functionality has to be provided for each application. Of course, end-to-end transparency is again broken by his mechanism.

6.2 Comparisons of dual stack, tunnelling and translation approaches

There is a need for a detailed study of the various mechanisms whereby IPvIPv4 and IPvIPv6 can be supported by the BSM. This should form the object of a further TR, as recommended in clause 12. AT this stage, it is possible to list some general observations, as follows:

- Tunnelling is probably too onerous as regards transmission overheads in BSM context.
- Dual stack is the most widespread approach at present, and offers clear advantages.
- Translation requires availability of IPv4 addresses to translate into, which is unlikely to be the case in the long run.

- Dual stack requires compatibility with two sets of C-plane protocols.
- Dual stack implementations where convergence is performed above the SI-SAP are generally to be preferred.

6.2.1 Impact on BSM

The conclusion of the preceding clauses is that the SI-SAP should be designed to support both native IPv6 and dual stack IPv4/IPv6 services in addition to native IPv4 services. In the U-plane, this implies support for IPv6 SDUs, with the corresponding implications for MTU (see clause 4.2.5 on path MTU discovery). In the C-plane, the Address Resolution function must support both IPv4 and IPv6 addresses. This will require some minor modifications to existing BSM documents as outlined in clause 12.

6.3 DNS Issues

Domain Name Servers are used to resolve text-based identifiers referring to Internet nodes into IP addresses. DNS queries in the IPv4 world expect a 32 bit address to be returned. DNS servers that support IPv6 may return 128 bit addresses for an IPv6-addressable node. RFC 3596 [75] introduces DNS extensions to support IPv6. Specifically it defines a new resource record type that stores a single IPv6 address as well as a special domain in which to look up a given IPv6 address (so-called reverse look-up, which returns the text-based domain name from the 128 bit IPv6 address). It also redefines some existing query types to cater for IPv6.

In addition to the "A" record type that is used to store IPv4 addresses, [75] defines the "AAAA" record type, which encodes an IPv6 address in a 128 bit field. An application attempting to resolve a domain name into an IPv6 address must be enabled to receive an AAAA record type. Since some nodes may have both an IPv4 and an IPv6 address, it must generally be prepared to accept either format. The resolver libraries used by applications have to support reception of multiple records of both type, and may need to order their return to applications depending upon whether they are IPv6-enabled or not. These issues are explored at greater length in RFC 4213, "Basic IPv6 Transition Mechanisms" [67] and RFC 3493, "Basic Socket Interface Extensions for IPv6" [49], which defines primitives that allow a resolver to determine whether a node has an IPv4 address, and IPv6 address, or both.

A further issue concerning DNS is that of accessibility of DNS servers in a mixed IPv4/IPv6 Internet. RFC 3596 [75] states:

- "The IP protocol version used for querying resource records is independent of the protocol version of the resource records; e.g., IPv4 transport can be used to query IPv6 records and vice versa."

The DNS tree structure used in address resolution relies on accessibility to subsidiary DNS servers from the root servers. A resolver that tries to look up a name starts out at the root, and follows referrals until it is referred to a name server that is authoritative for the name it is attempting to resolve. If it is using IPv4 transport to achieve this, for example, then referral to a point in the chain of referrals that is only accessible using IPv6 will cause the lookup to fail. RFC 3091 [20] recommends that:

1) "every recursive name server SHOULD be either IPv4-only or dual stack".
2) "every DNS zone SHOULD be served by at least one IPv4-reachable authoritative name server".

Indeed, this need for preventing fragmentation of the domain name space may discourage domain name resolvers from adopting IPv6 transport, as IPv4 is the only method that guarantees complete resolution.

6.4 BSM architecture-specific routing considerations

TR 102 155 [25] envisaged 7 routing scenarios as listed in table 6.4.1.

Table 6.4.1: BSM Routing Scenarios

Connection service scenario	Interconnection w.r.t. Autonomous Systems	Satellite network topology avoiding double hops	Routing protocol across satellite
Internet Access - Single ISP	Single AS	Star	None (Static)
Internet Access - Multihomed	Single AS	Mesh	IGP
Head Office-based Intranet (VPN)	Single AS	Star	IGP (but static default gateway, etc.)
Intranet (VPN), Extranet	Single AS	Mesh	IGP
Independent IP "Island" Interconnect	Multi AS	Mesh	BGP
IP Core Network	Multi AS	Mesh	BGP
IP Edge Network Interconnect	Multi AS	Mesh	BGP

In mapping this into an IPv6-compatible context, it is necessary to specify an appropriate IGP. This should almost certainly be OSPFv3 [6]. The use of BGP with suitable IPv6 extensions is also feasible [65] in the last three entries in table 6.4.1.

6.4.1 Impact on BSM

In order to meet the routing schemes presented above, the BSM C-plane functions must be compliant with one or more of OPSFv3, BGP4+ or other IPv6-compatible gateway protocols. Other routing protocols for IPv6 are RIPng and IS-IS for IPv6. Definition of the impact of this is for further study (see clause 12).

7 Performance and Quality of Service issues

7.1 Performance implications

The most obvious immediate impact of transporting IPv6 packets is the additional bandwidth required due to the longer address field. However, within a BSM subnetwork, much of the address information may be identical, especially for link-local addresses. A general approach to saving bandwidth in these situations is provided by header compression, which is discussed below.

7.1.1 Header compression

Header compression is especially useful in limited bandwidth networks in order to compensate for the often redundant information contained in IP packet headers. This problem is exacerbated in IPv6 by the use of 128 bit addresses, meaning that in each IP packet there is at least an extra 24 bytes over and above the IPv4 header length. For many shorter packets this can represent a significant additional overhead. Currently all IETF specified header compression frameworks define a compressor on the sender side and a decompressor on the receiver side. The compression is achieved by generating context information for a specific packet stream on both sides of the link; once the context is established, the compressor starts to transmit only the differences between the created context and the current packet headers. This is done by classifying the fields of the headers and applying different kind of compression schemes to them.

An early IETF standard "IP Header Compression" (RFC 2507,[14]) proposed a protocol for compressing IPv4, IPv6, UDP and TCP headers. RFC 2508 [5] extended the framework of [14] to compress IP (IPv4 and IPv6)/UDP/RTP headers.
RFC 3545 [63] was published which proposes some protocol enhancements for [14] to perform better over links with high delay, packet loss and packet reordering. RFC 2507 [14] already considered the extension headers defined for IPv6 in
RFC 2460 [11] and proposed compression schemes, thus enabling the use of IPsec over satellite links using header compression.

In July 2001, the IETF ROHC (Robust Header Compression) working group published "Robust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed" (RFC 3095,[1]) which defines a general framework for using ROHC over unidirectional and bidirectional links and provides profiles for transmitting uncompressed IP packets as well as compressed IP/UDP, IP/UDP/RTP, or IP/ESP packets. The basic thesis of the ROHC Working Group is that efficient header compression is possible due to the fact that there is considerable redundancy between field values within the headers of a packet, and more especially between the headers of consecutive packets in a flow (see, for example RFC 3095 [1]). RFC 3095 [1] supports both IPv4 and IPv6 packets. As some of the ROHC parameters per channel are negotiated over the channel/link which connects two ROHC entities, it is necessary for unidirectional kinks either to exchange the parameters out-of-band or to configure them in advance. ROHC does not propose explicit compression schemes for IPv6 extension headers but defines a general table-based approach in which extension headers of a packet stream are recorded, linked to the stored context information and then used through an index. RFC 3843 [62] proposes a header compression profile for IP headers only, which offers a basis for applying compression in a link-independent manner above the SI-SAP. The handling of AH and ESP headers in ROHC is explicitly discussed and defined in [1].

7.1.2 IPv6 neighbour discovery issues

An overview of Neighbour Discovery has already been given in clause 4.2.6. In this clause we address issues that arise when performing Neighbour Discovery in satellite networks. In addition, we will also briefly consider the Virtual Router Redundancy Protocol (VRRP) used for automatic failover to a backup router in case the default router of a LAN ceases to be operational.

7.1.2.1 Parameter setting

There are a number of configuration parameters of IPv6 nodes that support the IPv6 Neighbour Discovery protocol. Some care may need to be taken when setting them in the BSMS. In particular, we need to consider the parameters *Retrans Timer* and *Router Lifetime*.

Retrans Timer specifies the time between retransmissions of Neighbour Solicitation messages to a neighbour when resolving the address or when probing the reachability of a neighbour.

When performing Duplicate Address Detection (DAD), *Retrans Timer* specifies the delay between consecutive Neighbour Solicitation transmissions as well as the time a node waits after sending the last Neighbour Solicitation before ending the Duplicate Address detection process. The default value for *Retrans Timer* is 1 000 millisecond [67], [76].

In satellite networks, the Round trip Time (RTT), the time until a node gets a reply to a Neighbour Solicitation message, may be relatively long. On bidirectional GEO satellite links, for instance, a RTT of about 550 ms can be observed. This value would be doubled if the satellite network includes double hop behaviour, e.g. when a meshed network is based on a star network technology like DVB-RCS, where traffic between STs is sent by a node to the hub and forwarded to its destination node by the hub gateway. If processing and queuing times are added to this, the 1000 millisecond limit may be exceeded. Hence, satellite network administrators should set the Retrans Timer parameter carefully in order to permit DAD processing to complete before the process times out.

Router Lifetime: This parameter is include in Router Advertisement messages and specifies the lifetime associated with the default router. A value of zero indicates that this router cannot be used as a default router [76].

On unidirectional links as used in satellite networks the *Router Lifetime* parameter in Router Advertisement messages has to be set to zero, because the link-local address of the advertising unidirectional interface cannot be used as default router address by nodes. An exception is the usage of the UDLR mechanism (RFC 3077 [21]). (cf draft-ietf-ipdvb-ar-05 for DVB support of bi-dir on unidir links.)

7.1.2.2 Performance issues of neighbour discovery

The neighbour Discovery mechanism involves the transmission by a node of various multicast ICMPv6 messages in order to allow the node to build up a picture of its local communications neighbourhood. Nodes also emit various other signalling messages in response to the reception of these messages emanating from their neighbours. Among these messages are the Neighbour Solicitation message, the Neighbour Advertisement message and the Router Advertisement message. See RFC 2461 [66].

7.1.2.2.1 Star architecture

In a star architecture BSMS, the only link over which a node other than the hub can transmit such messages is the inroute to the hub. In all cases, the next-hop IPv6 address of the packet will be that of the hub. In this case, it is comparatively straightforward for the hub to process these C-plane packets, and generate responses as appropriate on its outbound links. It is unlikely that in a practical implementation the hub would use the ND mechanism itself to explore its environment by emitting NS messages throughout the network. It would be more efficient for it to construct address tables based upon its Satellite Dependant knowledge of the BSMS MAC address space (the BSM_IDs of the STs in the network). Otherwise, the level of signalling traffic in the network could become problematic.

Issues relating to this have been raised in the IETF WG on IPDVB together with questions regarding message multiplication and possible circularity. See also RFC 4326 [44] and the Internet draft draft-ietf-ipdvb-ar-05 [45] (especially clause 6 therein) for more information on these issues.

7.1.2.2.2 Mesh architecture

In a mesh network, the problem of message flooding is compounded because it is possible for a node to have many hundreds or thousands of neighbours. If all nodes in a large system begin to broadcast NS messages, the traffic levels could become crippling. However, in all practical implementations of mesh networks, while the U-plane connectivity may be mesh, most or all of the C-plane connectivity is star.

In this case, there has to be some kind of interception of the ICMPv6 messages, with a mapping of their addresses onto some suitable destination in order to limit message exchanges to the node in question and the signalling hub of the BSM network. RFC 4389 [73] considers Neighbour Discovery Proxies in a related scenario, where L2 bridging is being offered. The same basic approach can be adopted in the BSM in order to manage the Neighbour Discovery process efficiently.

7.1.2.3 Neighbour address resolution

Neighbour Address resolution assumes bidirectional links, whereas in certain satellite networks unidirectional links are used. One mechanism for emulating bidirectional links with unidirectional links is UDLR which has been described in RFC 3077 [21](cf. RFC 4326 [44]).

Before a node is able to send IP traffic to a destination node, it has to obtain information about the link layer identifier of the next-hop node. When applying Neighbour Address resolution, it takes about 550ms until a Neighbour Advertisement is received as response to a Neighbour Solicitation message. During this time, the sending node has to buffer the traffic. On a 20Mbit/s satellite link this means buffering 11Mbit of data, i.e. about 1,4 Mbytes.

On less reliable satellite links with a high Bit Error Rate, packets may be lost. We therefore consider the possibility of losing a Neighbour Address resolution message. As described above, a node repeats Neighbour Address resolution when the process exceeds *Retrans Timer* milliseconds. If *Retrans Timer* is set to, say 2000 and if the first process fails, Neighbour Address resolution would take about 2 550 ms. This would require buffering of around 7 Mbytes of data on a 20Mbit/s satellite link.

Hub stations may provide a very large number of data links, so it is clear that co-location of the Address Resolution function with the hub station is probably required.

7.1.2.4 Duplicate address detection (DAD)

Duplicate Address detection requires full multicast capable links as already discussed above. This functionality is provided by satellite networks with fully meshed DVB-S duplex links (DVB-S on the forward and return link) or by DVB-RCS architectures with regenerative satellites, for instance.

The IETF WG on IPDVB has begun to investigate issues concerning ambiguity in determining the origin of duplicate message detection and other potential difficulties with DAD over satellite in a DVB architecture. In a mesh satellite architecture, there is the danger of a heavy overhead of xx messages being sent over the air; this may require enforcement of non-mesh architecture in the C-plane to force reference to a central address server rather than direct addressing of all neighbouring nodes (which might include all nodes in the BSMS).

7.1.2.5 Router and prefix discovery

Router Advertisement messages are multicasted periodically to the all-nodes multicast address. From the source address included in the Router Advertisement message, nodes attached to the link can determine the link-local address of the router. If the router signals its willingness to be used as default router, this address is added to the default router list of the nodes. Router Advertisements may also contain a list of IPv6 prefixes that are reachable on-link.

Router and Prefix discovery require that the underlying satellite link be visible as a bidirectional link at the IPv6 layer. Some satellite network architectures do not fulfil this requirement, for instance, architectures with unidirectional links (e.g. DVB-S) in which nodes have different IP addresses on the sending and receiving interface. When Router and Prefix Discovery are not supported, IPv6 Stateless Address Autoconfiguration can only be performed partially.

7.1.2.6 Virtual Redundancy Protocol (VRRP)

In this clause we will briefly discuss issues related to the deployment of the Virtual Router redundancy protocol (VRRP) [58] in an IPv6 satellite environment. Although it is not a mandated feature of IPv6 Neighbour Discovery, we will consider VRRP here because it uses the IPv6 Router Discovery process to enable automatic failover to a backup router in case the default router of a LAN becomes inactive. The principles of VRRP are based on the definition of a virtual router with a specific IP address assigned. This IP address is used by hosts as default router IP address. VRRP specifies a decision process whereby one of the routers attached to a LAN and configured for VRRP declares itself the Master. The Master advertises the virtual router IP address in Router Advertisements to the LAN and hosts configure this address as default router address. Afterwards, the Master is responsible for forwarding packets addressed to the virtual router's IP address. When the Master becomes inactive, the failover process starts again and a new Master is chosen, responsible for forwarding IP packets addressed to the virtual router IP address.

VRRP assumes that messages sent by hosts to the default router are received by each VRRP router, so VRRP routers and hosts have to be attached to a full link multicast capable link, e.g. an Ethernet LAN. Thus, the overall BSMS architecture must support this kind of topology in order to deliver VRRP.

7.1.3 Impact on BSM

Header compression is an important function in the BSM, since bandwidth is a scarce resource in all satellite networks. ROHC is generally viewed as the most appropriate header compression framework in widespread use, and is specifically targeted at use over error-prone and long-delay links. It offers a framework in which specifically designed compression protocols can be developed for obtaining the greatest efficiency in the BSM, and satellite network designers and operators should study particular approaches in order to assess the exact impact of the transition to IPv6 on bandwidth efficiency in satellite applications. A recent Internet draft on header compression for ULE [2] addresses some of these issues, but a more general approach within the context of BSM is likely to be of benefit.

Neighbour Discovery and Duplicate Address Detection raise potentially significant performance issues, which need to be studied further. Some work is being carried out in the IETF, mainly in the context of VB star architecture satellite networks, but there is a need for more standardization work, and a further TR to deal with these specific issues should be envisaged.

7.2 IPv6 Quality of Service

The overall approach to QoS in IPv6 is the same as for IPv4, the only difference between them being in the way packets are classified in terms of aggregates or flows.

Compared with IPv4 packets, where the header contains only the type of service (ToS) field for use of QoS, the header of IPv6 contains different fields for the use of QoS (see clause 4.1). IPv6 therefore includes *ab initio* standardised support for QoS; QoS implementation is defined so that routers can easily identify packets belonging to an individual QoS flow, which allows the routers to allocate the necessary Per Hop Behaviour (PHB) and amount of bandwidth to those packets. Furthermore, the QoS classification is included in the IPv6 packet header, which allows the packet body to be encrypted, whilst maintaining QoS functionality because the header is not encrypted. This makes it possible to send streaming audio and video over the Internet with IPSec encryption, but in a manner that guarantees adequate bandwidth for real-time playback.

Specifically the IPv6 header fields relevant to QoS are:

1) The traffic class field (8 bits) is similar to the ToS field in the IPv4 header, and can be used by originating hosts and intermediate routers to identify and distinguish between different classes or priorities of IPv6 packets. This field may be used to set specific precedence or differentiated services code point (DSCP) values. These values are used in exactly the same way as in IPv4.

2) IPv6 also has the flow label field (RFC 3697 [70]) of 20 bits, which enables per-flow processing at the IP layer. It is set by the source node and can be used for special s²ender requests. The flow label must not be modified by an intermediate node.

NOTE: The advantage of the flow label is that transit routers do not have to open the packet payload to identify the flow, which aids in identification of the flow when using encryption and in other scenarios. The flow label is currently specified as a random series of bits that are associated with a flow.

There is no standard way of processing IPv6 flow labels for IntServ or DiffServ. Whilst the flow label was intended for IntServ, there have been proposals for its use in DiffServ, for example to simplify the packet classification process at the ingress, though this might need some modification to the IPv6 standard in terms of the ways flow labels are assigned. At present, there is no intention of correlating the flow label between component flows of aggregate for the purposes of simplifying DiffServ classification, for example.

For Intserv, each flow should ideally be treated independently and the flow label allows easier classification of flows at the ingress, instead of inspecting the complete 5-tuple. Each IntServ flow has its own negotiated QoS, and in an ideal network the flows should be processed separately in transit to ensure their contracted QoS is maintained.

IntServ forIPv6 can be provided in the same way as for IPv4. This is possible since the L3 RSVP protocol which may be used for IntServ etc. is compatible with IPv6, and is used to reserve resources for IPv6 or IPv4 flows. RSVP operates as a Control Plane protocol in parallel to IPv4 or IPv6.

Note that as yet no standard signalling protocol exists to negotiate the flow label (which would be a change for IPv6 with respect to IPv4).

7.2.1 Impact on BSM

The processing of QoS for IPv6 packets in the network can be different than for IPv4 if the flow label is used. This requires a different procedure in L3 devices, including BSM ST's, for classifying packets and identifying flows. Nevertheless, the approach adopted for the BSM at the IP layer in transition to IPv6 QoS is identical to the general approach described in [6], namely the options of Dual Stacks, tunnelling or Protocol Translation (NAT-PT).

The general BSM QoS architecture for IPv4 shown in figure 7.2.1 (taken from [36]) will be identical for IPv6, the differences being in the implementation of the classifier which will add IPv6 traffic class and also IPv6 flow label classification capability.

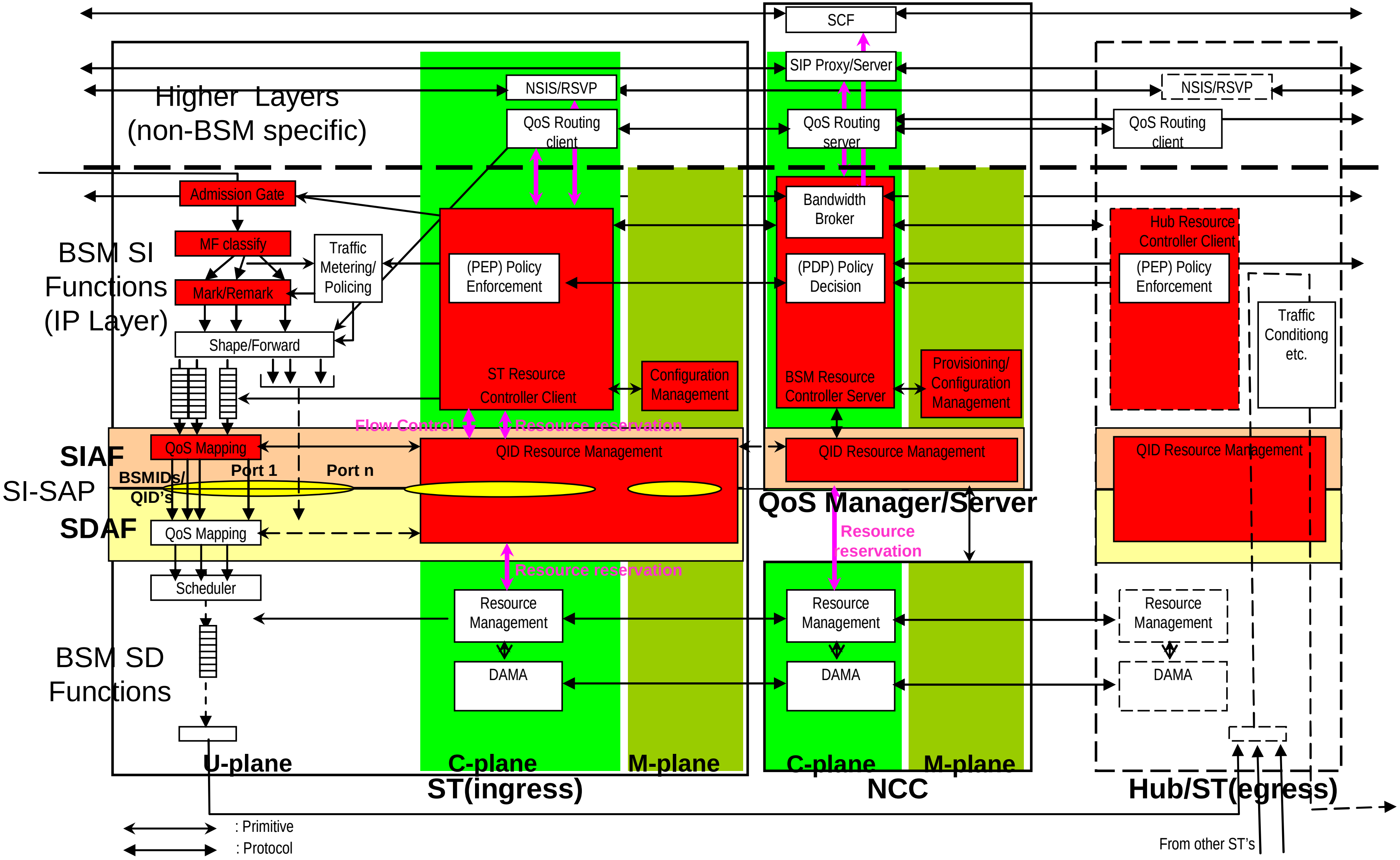


Figure 7.2.1: BSM QoS Architecture

At the SISAP, it can be assumed that the interface is compatible with IPv4 and IPv6 and no difference of approach is necessary, since the SD layers can and should offer transparency to high layers.

NOTE: should be qualified as per general comment about IPv6 support by the SI-SAP.

The QoS functionality defined in [36], specifically the use of QIDs is compatible with both IPv4 and IPv6 QoS.

For IntServ, as indicated in [38], ideally each flow should be allocated to a QID at the SISAP and hence a flow label will correspond to a QID. However, there are also alternatives to IntServ flow processing which approach DiffServ by aggregating flows at QID or SD level, in order to simplify processing and provide scalability for increasing numbers of flows. In this case, there would clearly be no one-to-one QID to flow label correspondence.

8 Multicast issues

IPv6 introduces not only expanded multicast addressing compared with IPv4, but also a new multicast group management protocol (MLD - Multicast Listener Discovery, RFC 3810 [78]) which is included as part of ICMPv6, instead of IGMP.

8.1 Addressing

An overview of IPv6 addressing is given in clause 4 (see also RFC 4291 [55]). Multicast addresses are identified by the first 8 bits of the IPv6 address field being set to all 1's, or FF in hex format. The subsequent two four bit fields represent flags and scope respectively. The Group ID (or address) is therefore contained in the remaining 112 bits. Although multicast Group IDs are not guaranteed to be unique even in the public domain, this extended range of addresses in IPv6 compared to IPv4 makes any address collisions many times less likely, and so increases service reliability. Although use of source-specific multicast (SSM - see 8.1.1) has increased potential collisions, these are avoided by filtering on source addresses.

The rules for assigning new IPv6 multicast addresses are defined in RFC 2375 [56] (see also RFC 3306 [51] and RFC 3307 [50]).

This general multicast address format is suitable for "permanent" multicast addresses (i.e. those allocated by IANA) and is as follows:

Prefix	Flags	Scope	Group ID
8 bits	*4 bits*	*4 bits*	*112 bits*
All'1s	00xx	xxxx	x…….x
NOTE: The above IPv6 address equates to FFxx::/8 in IPv6 notation.			

8.1.1 Dynamic multicast addressing

The multicast address format has been further refined by RFC 3306 [51] for "dynamic", and for source-specific (SSM), addressing. By this means, network operators (e.g. for the BSM domain) will be able to identify their multicast addresses without needing to run an inter-domain allocation protocol, such as a Multicast Address Allocation Protocol and the Multicast Address-Set Claim Protocol (RFC 2909 [69]).

The dynamic multicast address is achieved by introducing a unicast network prefix into the multicast address as follows:

Prefix	Flags	Scope	reserved	plen	network prefix	Group ID
8 bits	*4 bits*	*4 bits*	*8 bits*	*8 bits*	*64 bits*	*32 bits*
All 1's	0011	xxxx	x..x	x..x	x…x	x…x
NOTE: The above equates to FF3x::/32 in IPv6 notation.						

In this case, the least significant 32 bits are denominated the "Group ID", and the next 64 bits are the unicast prefix. The remaining 16 bits of the original Group ID are reserved for specific purposes. The network prefix is that which has been assigned to the unicast subnet or domain owning, or allocating, the multicast address.

RFC 3307 [50] further defines specific address ranges:

- FF3x::4000:0001 through FF3x::7FFF:FFFF are reserved in for allocation by IANA.
- Addresses in the range FF3x::8000:0000 through FF3x::FFFF:FFFF are allowed for dynamic allocation by a host.
- Addresses in the range FF3x::0000:0000 through FF3x::3FFF:FFFF are invalid IPv6 SSM addresses. (RFC 3307 [50] indicates that FF3x::0000:0001 to FF3x::3FFF:FFFF must set P=0 and T=0, but for SSM, RFC3306 mandates that P=1 and T=1, hence their designation as invalid. Hence, the treatment of a packet sent to such an invalid address is undefined: a router or host may choose to drop such a packet).

8.1.1.1 Use of Dynamic Addresses by Link Layers (or BSM SISAP)

Dynamic addresses are used by several link layers (e.g. RFC 2464 [9], RFC 2467 [10]) to create a corresponding local multicast layer 2 address, in which the 32 bit group ID of the IPv6 address is transferred to the 32 lsb's of the link-layer address.

In IPv4 there is a potential problem for multicast when mapping to Ethertype addresses at layer 2; namely there is an overlap of 32 to 1 in the address spaces since 5 bits of the IP address are made ambiguous by only allocating 23 bits of the 48 bit Ethernet address for multicast. This situation is the case for the BSM since the SISAP GID address is based on the Ethertype address [34]. Therefore filtering of groups has to be done at IP level to resolve any ambiguity.

This ambiguity is avoided at layer 2 in IPv6 since the whole of the 32 bits Group ID of the IP multicast address are directly transferred to the least significant 32 bits of the Ethernet address. IP group IDs of 32 bits or less will then generate unique link-layer addresses within a given multicast scope. There is still potential ambiguity due to the more significant IP address bits not being transferred, but this is resolved at the IP level since link layers are part of the same domain.

8.1.2 Source-Specific multicast addressing

A further variation of dynamic addressing is defined by RFC 3306 [51] for Source-Specific Multicast (SSM) addresses, by setting the network prefix to zero. Instead the "owner" of the address is the source address as indicated in the IPv6 header. This address type is indicated by setting bits as follows:

Prefix	**Flags**	**Scope**	**reserved**	**plen**	**network prefix**	**Group ID**
8 bits	*4 bits*	*4 bits*	*8 bits*	*8 bits*	*64 bits*	*32 bits*
All 1's	0011	xxxx	x..x	All 0's	0…0	x…x

Therefore all IPv6 SSM multicast addresses will have the format FF3x::/96.

8.1.3 Embedding the rendezvous point for PIM-SM

A method for embedding the Rendezvous Point (RP) for PIM-SM in the dynamic multicast address has been defined in RFC 3956 [71], see also clause 8.2.1.

Whilst, in general, embedding a 128 bit RP address in another similar address space is impossible, by making some assumptions about multicast addressing, a solution has been found. This also requires a PIM-SM implementation of the Embedded RP group-to-RP mapping mechanism which takes this encoding into account.

Prefix	**Flags**	**Scope**	**rsvd**	**RIID**	**plen**	**network prefix**	**Group ID**
8 bits	*4 bits*	*4 bits*	*4 bits*	*4 bits*	*8 bits*	*64 bits*	*32 bits*
All 1's	0111	xxxx		x..x	All 0's	0…0	x…x

The second bit of the "flags" is set to 1 to indicate that the RP is embedded. The RIID (RP Interface ID) nibble is also taken from the reserved byte as an additional mechanism.

8.1.4 Impact on BSM

As indicated above, for the BSM system, IPv6 has a potential benefit compared with IPv4, when BSM GID's are allocated to dynamic multicast addresses on the basis of the Ethertype model.

Therefore a suitable IPv6-to-GID mapping scheme should be defined for the BSM (see clause 12).

8.2 Routing and group management protocols

8.2.1 Routing

The PIM-SM protocol (including the PIM-SSM variant) is almost exclusively used in existing and proposed multicast routing applications today. Therefore other approaches (e.g. PIM-DM, CBT, MOSPF), which may nevertheless be compatible with IPv6, will not be considered further here.

The PIM-SM protocol is specified for IPv6 and could be used for routing in a similar way to IPv4. However for PIM-SM the need for Rendezvous Points (RP's) causes a problem with global, interdomain IPv6 multicast, since the RP's have no way of communicating information about (active) multicast sources to other multicast domains, as the Multicast Source Discovery Protocol (MSDP) has deliberately not been specified for IPv6.

A solution (RFC 3956 [71]) has been to embed the RP in the multicast address, see also clause 8.1.3 above.

Use of PIM-SSM (where the source address is also specified for each multicast group join) avoids this problem as it does not need RP's. However PIM-SSM requires MLDv2 (see below) implementation and is not suitable for many-to-many applications such as conferencing.

8.2.2 Group management

The MLDv2 protocol has been specified for compatibility with PIM-SSM, and is backwards compatible with MLDv1 which is designed for PIM-SM. MLD is in general the replacement for IGMP in IPv4.

8.2.3 Impact on BSM

MLD is the protocol which is likely to have to operate over a BSM system in which the ST's act as proxies or snoopers for MLD hosts located in attached terminals or networks. As MLD is a garrulous protocol and is not necessarily optimized for high numbers of attached hosts such as in a satellite system, there is a probable benefit in modifying it for the BSM case to reduce signalling traffic and improve response times.

Therefore, the trade-offs for implementing and modifying the MLDv1 and MLDv2 protocols over BSM should be investigated further (see clause 12). Clearly, this is an issue internal to the IP layer and has no impact on the SI-SAP.

9 Security issues

9.1 Overview of IPv6 security

The most important difference between IPv6 and IPv4 from a security perspective is that IPSec is included by definition in the protocol stack. However, this does not mandate its operational use. The same consequences of enabling IPSec that impact the BSM when transmitting IPv4 packets apply in the IPv6 context. These have been described in TS 102 465 (BSM Security Functional Architecture) [40].

Another significant difference arises from the inclusion of ESP and AH fields in all packets, which allows OSPFv3 to dispense with explicit security and authentication mechanism present in the IPv4 versions of OSPF. However, this does not have a satellite-specific consequence as such.

9.1.1 Impact on BSM

The transition to IPv6 has certain security implications, but overall these are unlikely to have a major impact on the BSM.

10 Mobility

Mobility can include the following main cases:

- User mobility - the ability of a user to obtain his service profile on different end-hosts.
- Terminal mobility - the ability of a user's End Host (EH) to roam on different networks (e.g. to a different ST and/or to a different BSM network,) in either of two main cases:
  - Dynamically, where the terminal moves during a telecommunication session and maintains the session via handovers etc.
  - Quasi-statically (for transportable terminals) where the terminal only moves between sessions.

In all these cases, the ST is assumed to be fixed, and not itself subject to mobility.

Of the above cases, IP is concerned primarily with terminal mobility, and specifically with mobility within the network layer, where the user's end host changes its network point of attachment. User mobility can be provided by layers above IP.

Although for some services, application or session layer mobility may either partially replace or complement IP mobility, for terminal mobility, an IP-based solution is likely to be preferable, as it applies to all IP-based applications, rather than only real-time applications like Internet telephony and conferencing.

For BSM hosts, quasi-static mobility is appropriate since it is not expected that a host moves dynamically between BSM's or between a BSM and other networks. Rather the case considered is where the host is transportable and can attach itself from time to time to other networks. Another case for mobility is where the BSM system acts as a backhaul for several local wireless networks (e.g. WiFi, WiMax), and the user terminal moves between cells of the wireless networks. However, this case is similar to the treatment of mobility for other backhaul networks and no special impacts on the BSM are foreseen.

For overall network mobility, IPv6 offers enhancements over IPv4 as defined in RFC 3775.

The Mobile IPv6 protocol is an enhancement to the standard IPv6 protocol that make it possible for users to be reachable on "foreign" links. Mobile IP allows a Mobile Node (MN) to change its point of attachment (dynamically) to the Internet with minimal service interruption. Mobile IPv6"s routing optimization is superior to that of Mobile IPv4.

Within IPv6 itself, the larger IPv6 address field is better suited to scalability with extremely large numbers of mobile devices, than is IPv4. Also, there are inherent features within the IPv6 architecture to support Mobile IPv6, such as stateless autoconfiguration of IPv6 as well as Neighbour Discovery used for node discovery on a network. In addition, the IPv6 architecture includes options with the Next Header and Destination Options format, which lends Mobile IPv6 the ability to provide a better design of Mobility with IPv6 over IPv4.

The key issue with network layer mobility is that it requires the network to separate identity and network location, so that as a device "moves" within the network its identity remains constant while its location is changing. IPv4 overloaded the semantics of an address to include both identity and locality within an address, and IPv6 has not altered this architecture. In this respect, IPv4 and IPv6 offer the same levels of support for mobility. Both protocols require an additional header field to support a decoupled network identity, commonly referred to as the "home address". They then concentrate on the way in which the home agent maintains a trustable and accurate copy of the mobile node or network's current location.

10.1 Current work

Current work in the IETF includes Hierarchical Mobile IPv6 (HMIPv6) as an enhancement to MIPv6, designed to reduce the amount of signalling required and to improve handoff speed for mobile connections. HMIPv6 separates local from global mobility, which is managed by the MIPv6 protocols, while local handoffs are managed locally.

The IETF Network Mobility (NEMO) WG is investigating route optimization for mobility, due to the sub-optimal routing when communications to and from Mobile Network Nodes go through the bi-directional tunnel established between the Mobile Router and Home Agent when the mobile network is away.

The Host Identity Protocol (HIP) defines procedures for mobility of a HIP host where a host dynamically changes the primary locator that it uses to receive packets.

10.2 Impact on BSM

Since it is assumed that ST's are assumed to be fixed, and not themselves subject to mobility, then the impact of mobility on BSM is limited. The main impacts are on the user's end host which is beyond the scope of the BSM. In principle, the BSM can implement standard IPv4 or IPv6 mobility protocols without modification.

For the case of quasi-static terminal mobility considered for the BSM, it is not necessary for the end host to maintain the same IP session, or even the IP address, between network attachments, and the full provisions of Mobile IP are not required. The mobility that is more appropriate to the BSM is a higher layer mobility that allows the end host to gain access on a foreign network to its home services, without the complication of full Mobile IP implementation. A better solution is then to use AAA mechanisms, for example, which provide the means of administering policy to ensure proper use and management of resources within a roaming network environment.

Therefore, it is considered that all aspects Mobile IPv6 are not needed for the BSM case of transportable-only hosts. However if IPv6 address mobility is considered necessary, firstly the impact of MIPv4 on BSM would need to be studied further (which has not yet been done), followed by the incremental differences for MIPv6.

Where the BSM system acts as a backhaul for several local mobile networks (e.g. WiFi, WiMax), and the user terminal moves between cells of the mobile networks, no special impacts on the BSM are foreseen. Otherwise, the implementation of other aspects of IPv6, such as addressing, will help towards increasing BSM mobility without needing special provisions.

11 Multi-Homing

IPv6 potentially supports multi-homing (the ability for the local network to be attached to more than one Internet Service Provider) and dynamic provider selection, see RFC 3178 [52].

The objective here is to support some form of multi-homing of local networks where any incremental routing load is strictly limited in its radius of propagation. This remains an active area of consideration for the IETF and clear answers, in IPv4 or IPv6, are not available at present.

However, multi-homing is relatively easy if it is allowed to globally announce the network's address prefix without recourse to any form of provider-based address aggregation. But this is a case of achieving a local objective at a common cost of the scalability of the entire global routing system, and this is not a supportable cost.

So any special IPv6 capability in this area is considered to be over-stated at present.

The impact on the BSM is therefore not considered to be significant.

12 Recommendations on future work

The present document is concerned with identifying the impact of the IPv6 transition on the BSM deliverables. The current and previous BSM reports and specifications have been primarily focused on IPv4 interworking and there is thus a need for both new deliverables and revisions (maintenance) of existing deliverables.

A number of specific technical areas call for the production of one or more new Technical Specifications; these include IPv6 support at the SI-SAP, header compression, mobility, management, unreachability detection and address management.

12.1 Support for IPv6 at the SI-SAP

We propose the development of a Technical Specification to give a detailed description of the support of "native" IPv6 packets at the SI-SAP and in the Satellite Dependent layers.

- Title: "Support for IPv6 at the SI-SAP"

This TS should describe in detail the mechanisms and primitives at the SI-SAP that are required to deliver such support. Satellite-dependent functions as required in the various BSM architectures, mesh and star, both regenerative and non-regenerative, should be discussed. The document should address the impact of the new C-plane protocols used by IPv6 (e.g. ICMPv6 and OSPFv3) on BSM. It will identify all the changes that are needed to the SI-SAP specification to support IPv6, leading to a revision of the SI-SAP, as discussed in clause 12.7.

We recommend that this Technical Specification should elaborate the use of the dual-stack architecture above the SI-SAP. It should contain a discussion of the motivation for adoption of dual stack architectures (this should be an elaboration of clause 6.1.1 above, together with a discussion of industry best practice). It should define in detail the BSM dual-stack architecture at the SI-SAP, and assess the impact of the adoption of this architecture on the BSM, including any modifications that may be required to other BSM Technical Specifications.

The present document will also set the context for the various other Technical Specifications that are recommended in the following clauses.

12.2 Header compression

We recommend the development of a Technical Specification on IPv6 Header Compression. This TS should take into account related work being performed in other bodies, such as the IETF.

- Title: "IPv6 Header Compression schemes and protocols for BSM".

The TS should address specific header compression approaches suitable for satellite links. It should include not only a discussion of the protocols and procedures for header compression, but also analysis of the suitability of specific header compression algorithms or families of algorithms for use in BSM.

This work may identify the need for a further Technical Specification defining one or more specific compression algorithms suitable for use over satellite.

12.3 IP mobility

There is a need for a Technical Report to consider in further detail the issue of IP address mobility, as defined in clause 10

- Title: "Impact of IPv4 and IPv6 address mobility on BSM".

The present document should first consider the IPv4 case, looking initially at MIPv4, before studying the implications of the IPv6 mobility features. It should also investigate any security implications arising from the IPv6 implementation of mobility.

12.4 Stateless autoconfiguration and related areas

A Technical Specification (or possibly a Technical Report) should be produced to cover the impact of the IPv6 stateless autoconfiguration mechanism on the BSM.

- Title: "IPv6 stateless address autoconfiguration implications for BSM".

This TS/TR should examine the issues raised in clause 7.1.2 of the present document and other aspects of IPv6 stateless address autoconfiguration. In particular, consequences of Unreachability Detection and Neighbour Discovery should be considered, and any performance or efficiency impacts evaluated (especially those that are satellite-specific). Appropriate counter-measures to such impacts should be specified and assessed.

12.5 Management aspects

Management aspects arising from support of IPv6 should be studied. In particular, the development of IPv6-aware Management Information Bases should be considered, and documented as appropriate. At this stage, no specific deliverable is recommended, as the IPv6 management aspects should be considered in the context of proposed future BSM work on satellite network management.

12.6 Modifications to recent BSM documents

A number of Technical Specifications that have been developed concurrently with the present Technical Report. These include TS 102 460 (Address Management at the SI-SAP) [35], TS 102 461 (Multicast Source Management) [36], TS 102 462 (QoS Functional Architecture [37], TS 102 463 (Interworking with IntServ QoS) [38], TS 102 464 (Interworking with DiffServ QoS) [39], TS 102 465 (General Security Architecture) [40] and TS 102 466 (Multicast Security Architecture) [41]. These documents have largely restricted their attention to IPv4 aspects, in line with the Terms of Reference of STF 283. It was felt that at this stage, a Technical Report (the present document) was the appropriate level at which to consider IPv6 impacts on BSM. In the light of the preceding clauses, we can conclude that these four documents will be impacted to varying degrees by the need to address IPv6 issues. We look at the four technical areas separately.

12.6.1 Address management

We propose a major revision of TS 102 460 (Address management at the SI-SAP) [35] to take account of IPv6 issues.

12.6.2 Multicast source management

We propose a major revision of TS 102 461 (Multicast Source Management) [36] to take account of IPv6 issues.

A suitable IPv6-to-GID mapping scheme should be defined for the BSM.

The trade-offs for implementing and modifying the MLD protocols (see clause 8.1.3 above) over BSM should be investigated further.

12.6.3 Quality of Service

It is felt that only minor modifications will be required to the BSM QoS Technical Specifications, TS 102 462 (QoS Functional Architecture [37]), TS 102 463 (Interworking with IntServ QoS [38]) and TS 102 464 (Interworking with DiffServ QoS [39]) to take account of IPv6 implications.

12.6.4 Security

It is felt that only minor modifications will be required to the BSM Security Technical Specifications, TS 102 465 (General Security Architecture [40]) and TS 102 466 (Multicast Security Architecture [41]) to take account of IPv6 implications.

12.7 Summary of impacts of IPv6 support on other BSM documents

The following table summarizes the extent to which it is anticipated that IPv6 issues require revision of existing BSM Technical Specifications and Technical Reports.

The only TS that is likely to need significant revision is TS 102 357 (SI-SAP Specification) [34], together with the associated guidelines TR 102 353 [33]. The scope of these modifications will be determined by the first recommended document above, namely "Support for IPv6 at the SI-SAP" (see clause 12.1).

The only other TR that is likely to need significant revision is TR 102 155 (IP interworking over satellite; Addressing and routing) [25]. In particular, clause 7.3.2 and related clauses need to be reconsidered in the context of the various IPv6 address assignment options (autoconfiguration, DHCPv6 assignment, etc.) Attention should be given to the generation of sequence charts reflecting architectural options for address generation/assignment in the BSM, and consideration should be given to comparisons of distributed or autonomous generation as against centralized assignment. In clause 7.4 specific BSM link layer architectures should be considered and their impact on addressing and routing assessed. In particular DVB-S/DVB-RCS with ULE and MPOE options should be examined, as well as BSM-A and BSM-B mesh architectures.

A summary of the level of impact on each of these documents is given in table 12.1.

Table 12.7.1 Existing BSM Document Revision Estimate

ETSI reference	Ref	Title (SES BSM series)	Revision
TR 102 287	**Ref**	IP Interworking over satellite; Security aspects	Minor
TS 102 292	[28]	Services and architectures; Functional architecture for IP interworking with BSM networks	Minor
TS 102 293	[29]	Services and architectures; IP Interworking over satellite; Multicast group management;	Minor
TS 102 294	[30]	Services and architectures; IP interworking via satellite; Multicast functional architecture	Minor
TS 102 295	[31]	Services and architectures; BSM Traffic Classes	Minor
TR 102 353	[32]	Guidelines for the Satellite Independent Service Access Point (SI-SAP);	Major
TS 102 357	[33]	Common Air interface specification; Satellite Independent Service Access Point SI-SAP	Major
TR 101 984	[34]	Services and Architectures	Minor
TR 101 985	[23]	IP over Satellite	Minor
TR 102 155	[24]	IP interworking over satellite; Addressing and routing	Major
TR 102 156	[25]	IP interworking over satellite; Multicasting	Minor
TR 102 157	[26]	IP Interworking over satellite; Performance, Availability and Quality of Service	Minor

History

Document history		
V1.1.1	January 2007	Publication