ETSI TS 101 658 V1.2.1 (2000-04)

Technical Specification

Terrestrial Trunked Radio (TETRA); Digital Advanced Wireless Service (DAWS); Logical Link Control (LLC) Service Description

Reference
RTS/TETRA-04046

Keywords
PDO, data, radio, TETRA

ETSI

Postal address
F-06921 Sophia Antipolis Cedex - FRANCE

Office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - 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

Internet
secretariat@etsi.fr
Individual copies of this ETSI deliverable
can be downloaded from
http://www.etsi.org
If you find errors in the present document, send your
comment to: editor@etsi.fr

Important notice

This ETSI deliverable 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.

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 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://www.etsi.org/ipr).

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 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 Specification (TS) has been produced by ETSI Project Terrestrial Trunked Radio (TETRA).

The present document is one of the three documents specifying the requirements for TETRA Digital Advanced Wireless Service (DAWS):

- **TS 101 658: Logical Link Control (LLC) Service Description;**
- TS 101 659: Medium Access Control (MAC) Service Description;
- TS 101 660: Physical Layer (PHY) Service Description.

An overview of the requirements for DAWS can be found in TR 101 156 [1].

Introduction

The DAWS protocol architecture is provided in [1]. The Logical Link Controller (LLC) provides services to the network layer (NWK) and requests services from the Medium Access Controller (MAC) [2]. The present document provides the requirements the LLC service shall satisfy to operate successfully within a Digital Advanced Wireless Service (DAWS) network. As described in [1], LLC functionality may be distributed across several DAWS nodes. The following prefixes will be used to specify the scope of a requirement:

LLC	the requirement applies to the LLC in general
GW_LLC	the requirement applies to Gateway functionality
BR_LLC	the requirement applies to Bridge functionality
BS_LLC	the requirement applies to Base Station functionality
MS_LLC	the requirement applies to Mobile Station functionality

Figure 1 shows the architecture of the LLC for the minimum complexity DAWS network described in [1]. The network layer (NWK) accesses LLC services via service access points (SAPs) A and B. LLC_SAP_A is for data transfer service primitives and LLC_SAP_B is for local control and status service primitives, including RSVP[3] operations.

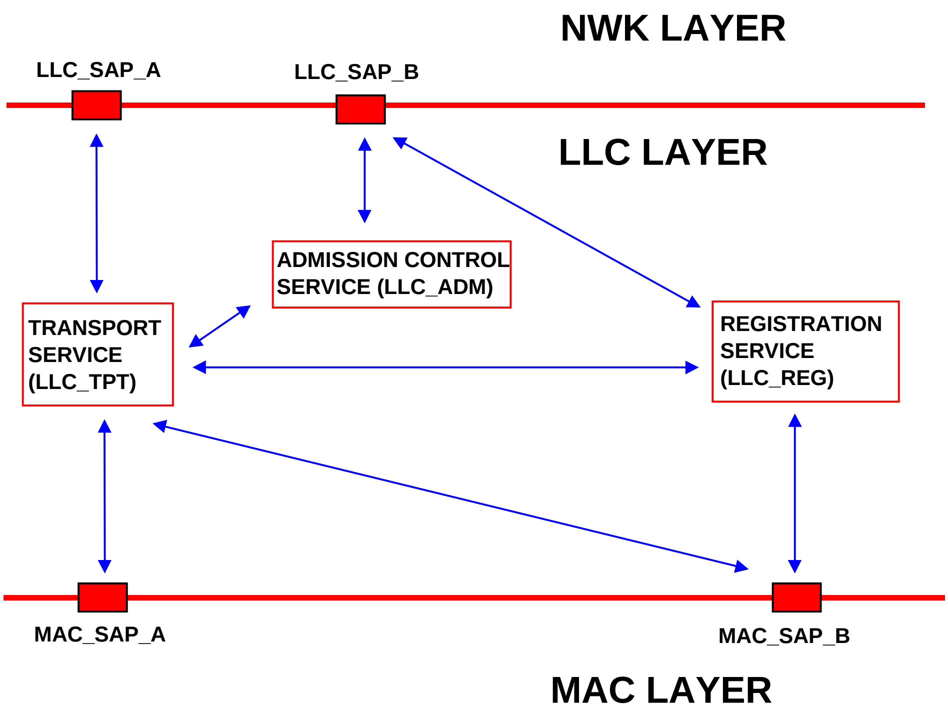

Figure 1: DAWS LLC Architecture

The LLC accesses MAC services via service access points A and B. MAC_SAP_A is for data transfer service primitives and MAC_SAP_B is for local control and status service primitives.

Requirements for the registration, admission control, and transport services are provided in the clauses 4, 5, and 6.Service primitives and associated service data units are provided in clause 7. Annexes A and B provide additional information on interworking Mobile IP and RSVP with the LLC. Annex C discusses the IPv4 to IPv6 transition.

1 Scope

The present document specifies the service requirements for a Digital Advanced Wireless Service (DAWS) Logical Link Control (LLC) layer compatible with IPv4 and IPv6 on the network layer. The LLC service requirements for other network layer protocols will be described in a separate document dedicated to each protocol.

The present document provides a conceptual architecture useful for specifying requirements but is not intended to imply a particular implementation.

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, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, subsequent revisions do apply.
- A non-specific reference to an ETS shall also be taken to refer to later versions published as an EN with the same number.

[1] ETSI TR 101 156: "Terrestrial Trunked Radio (TETRA); Technical requirements specification for Digital Advanced Wireless Service (DAWS)".

[2] ETSI TS 101 659: "Terrestrial Trunked Radio (TETRA); Digital Advanced Wireless Service (DAWS); Medium Access Control (MAC) service description".

[3] RFC 2205 (1997): "Resource Reservation Protocol (RSVP) - Version 1 Functional Specification".

[4] RFC 2215 (1997): "General Characterization Parameters for Integrated Service Network Elements".

[5] RFC 2211 (1997): "Specification of the Controlled-Load Network Element Service".

[6] RFC 1112 (1989): "Host Extensions for IP Multicasting".

[7] RFC 791 (1981): "Internet protocol darpa internet program protocol specification".

[8] RFC 2210 (1997): "The Use of RSVP with IETF Integrated Services".

[9] RFC 2185 (1997): "Routing Aspects Of IPv6 Transition".

3 Definitions and abbreviations

3.1 Definitions

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

base station: piece of equipment providing simultaneous, bi-directional network access to mobile stations

downlink: general term meaning "from the base station to the mobile station"

flow: sequence of data packets originating from a single source and addressed to the same destination for which special handling by intervening routers is desired

mobile station: piece of equipment able to create and consume data but only having network access via a base station

protocol data unit: set of parameters and/or data passed from peer to peer by a protocol primitive

protocol instance: two protocol processes which exchange messages in order to transfer data from one protocol process to the other

protocol primitive: request, response, or informative message sent from peer to peer

protocol process: entity created to manage one end of a peer-to-peer protocol. For unidirectional data flows, a protocol process can be further described as either a sender process or a receiver process

service data unit: set of parameters and/or data passed between adjacent layers by a service primitive

service primitive: request, response, or informative message sent between adjacent layers

sub-protocol: portion of a protocol performing a clearly identifiable operation

uplink: general term meaning "from the mobile station to the base station"

3.2 Abbreviations

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

ACK	Acknowledged
BEP	Best-Effort Plus
BS	Base Station
DAWS	Digital Advanced Wireless Services
DL	Downlink
GW	Gateway
IP	Internet Protocol
LLC	Logical Link Controller
LLC_ADM	LLC Admission Control Service
LLC_REG	LLC Registration Service
LLC_TPT	LLC Transport Service
LPDU	LLC Protocol Data Unit
MAC	Medium Access Controller
MS	Mobile Station
MSH	Mobile Station Handle
MSI	Mobile Station Identifier
NPDU	NWK Protocol Data Unit
NWK	Network
PDU	Protocol Data Unit
QoS	Quality Of Service
RSVP	Resource Reservation Protocol
SAP	Service Access Point
SDU	Service Data Unit
SW	Switch
UNACK	Unacknowledged
UL	Uplink

4 Registration Services

The LLC registration service (LLC_REG) supports cell selection, registration, de-registration, and hand-over operations.

Statements made in this clause regarding network level registration are for information only.

4.1 Base station registration

Every DAWS GW is assigned a globally unique identifier called the GWI. The GWI shall be assigned when the GW is manufactured and shall not be dynamically alterable. The GW address space shall be administered by an industry body to prevent GWI address duplication among manufacturers.

Every DAWS BS is assigned a globally unique identifier called the BSI. The BSI shall be assigned when the BS is manufactured and shall not be dynamically alterable. The BSI address space shall be administered by an industry body to prevent BSI address duplication among manufacturers.

Before a DAWS BS can provide service to MS, the BS shall register with a DAWS GW. During the registration process, the GW delivers its GWI to the BS. The BS broadcasts the GWI and its BSI in a system information message to all MS within the serving cell. A MS uses the broadcast GWI during hand-over to differentiate between intra-network and inter-network hand-overs.

4.2 Mobile station registration

4.2.1 Cell selection

When an MS is powered on, MS_LLC_REG shall issue a request to MS_MAC_REG to do a scan of available cells and report the results. MS_LLC_REG shall select the best cell and instruct MS_MAC_REG to camp on the cell. MS_LLC_TPT is then able to exchange PDUs with BS_LLC_TPT using the unacknowledged protocols.

4.2.2 Registration

After cell selection, MS_LLC automatically registers with BS_LLC and GW_LLC.

MS_LLC registration with a serving BS and GW involves the following steps:

1) MS_LLC sends a registration request to BS_LLC, providing its MSI;

2) BS_LLC forwards the MS registration request toward the GW via the next upstream bridge;

3) BR_LLC adds the binding (MSI, output_interface) to its routing table, and forwards the MS registration request toward the GW. For simplicity, this example will assume only one upstream bridge, so the next node will be the GW;

4) GW_LLC adds the binding (MSI, output_interface) to its routing table, and sends a MS registration response toward the serving BS;

5) BR_LLC forwards the MS registration response towards the serving BS;

6) BS_LLC generates a MSH for the MS and adds the binding (MSI, MSH) to its registration table. BS_LLC sends a registration response message containing the MSH to MS_LLC;

7) MS_LLC stores its assigned MSH for future uplink and downlink signalling with the BS;

8) MS_LLC sends a service indication message to MS_NWK indicating that LLC registration is complete, providing the GWI of the serving GW and BSI of the serving BS.

After LLC registration is complete, MS_NWK will usually perform network level registration with the GW in order to receive a care-of address for network level signalling. After obtaining a care-of address, downlink packet routing through the DAWS access network thus involves the following routing operations:

1) GW_LLC: MS care-of address to MSI;

2) GW_LLC: MSI to output interface;

3) BR_LLC: MSI to output interface (multiple intermediate bridges possible);

4) BS_LLC: MSI to MSH;

5) BS_LLC: MSH to Protocol Instance ID.

4.2.3 De-registration

A registration-related state created in a DAWS node has an associated de-registration timer. If the de-registration timer expires then the registration state is deleted. The following registration states have de-registration timers:

1) GW_LLC: the (care-of address, MSI) binding;

2) GW_LLC: the (MSI, output_interface) binding;

3) BR_LLC: the (MSI, output_interface) binding;

4) BS_LLC: the (MSI, MSH) binding;

5) MS_LLC: the MSH.

A de-registration timer is automatically restarted whenever traffic to or from the MS passes through the node.

A timer can also be restarted upon receipt by the managing entity of a "timer restart" message. MS_LLC will issue a "timer restart" message when necessary to maintain binding states between itself, the serving Base Station, and the Gateway.

4.2.4 Service interruption

When MS_LLC receives a service interruption indication from MS_MAC, MS_LLC sends a service indication interruption to MS_NWK. MS_NWK will discard uplink traffic until service is restored.

When service is restored, MS_LLC will send a service indication message indicating that LLC service is available. The GWI and BSI will not change. If the duration of the service interruption is sufficiently short, MS_NWK will not need to re-register with the GW_NWK but may continue to use the previously assigned care-of address.

4.2.5 Hand-over

The hand-over procedure allows a MS to move from one serving cell to another while possibly maintaining upper-level data flows. The hand-over procedures shall be designed to minimize PDU loss during the transition period. MS_LLC is entirely in control of the cell re-selection and hand-over process, but sends **LLC_service_indication** primitives to MS_NWK so that network level registration and resource management tasks can be performed.

When migrating between Base Stations within a single DAWS network, MS_LLC will send a message to MS_NWK indicating that service has been lost with the GW and BS. When MS_LLC registers with a new BS within the same DAWS network, the **LLC_service_indication** primitive will provide the new BSI along with the old GWI. If the duration of the service interruption is sufficiently short, MS_NWK will not need to re-register with the GW_NWK but may continue to use the previously assigned care-of address.

When migrating between Base Stations within different DAWS networks, MS_LLC will send a message to MS_NWK indicating that service has been lost with the GW and BS. When MS_LLC registers with a new BS within the new DAWS network, the **LLC_service_indication** primitive will provide the new GWI and new BSI. MS_NWK will need to re-register with GW_NWK to obtain a new care-of address.

5 Transport Services

The LLC transport service (LLC_TPT) provides link layer address resolution, packet classification, traffic measurement/policing/shaping, and packet segmentation/reassembly services.

5.1 Link layer address resolution

Link layer address resolution is the process of translating a network level address into a link layer address (the MSH). As described in clause 6, link layer address resolution is a two-step process. For a downlink packet, GW_LLC translates the network level address into a MSI. The MSI is then used for routing through the DAWS network to the BS.The BS then translates the MSI into the MSH for downlink transfer over the wireless link. For an uplink packet, MS_LLC uses the MSH assigned during BS registration for the uplink transfer over the wireless link.

5.2 Packet classification

At any given time, there will be multiple protocol instances in place between a BS and a particular MS. Packet classification is the process of selecting a particular protocol process to handle the transfer of a PDU to a peer entity. BS_LLC_TPT shall use the MSH, PDU destination address, and PDU destination port to obtain the protocol instance identifier for downlink IPv4 packets. BS_LLC_TPT shall use the MSH and PDU flow label to obtain the protocol instance identifier for downlink IPv6 packets. For uplink packets, MS_LLC_TPT shall use the same packet classification procedure as BS_LLC_TPT except that MS_LLC_TPT does not need to index based on the MSH because at any time the MS has at most one MSH.

The best-effort downlink or uplink protocol instance is the default method of transferring PDUs if no reserved protocol instance has been created for a particular flow.

LLC_TPT may also support best-effort plus (BEP) service. BEP analyses the flow of best-effort PDUs and detects when there is a sufficient flow directed to a single destination to justify a persistent bandwidth reservation. LLC_TPT then establishes a persistent bandwidth reservation to handle the best-effort traffic. In contrast to RSVP managed QoS, which has network-level scope, BEP has only link-level scope and is intended to optimize bandwidth utilization. The definition of requirements for BEP is for further study.

5.3 Traffic measurement, policing, and shaping

LLC_TPT shall monitor incoming flows for compliance with the associated TSPEC [4]. As defined in [5], the following requirements apply to MAC_TPT handling of nonconformant controlled-load flows:

1) MAC_TPT shall continue to provide the contracted QoS to conformant flows;

2) MAC_TPT should prevent nonconformant flows from unfairly impacting the handling of best-effort traffic;

3) MAC_TPT shall attempt to forward the excess traffic of nonconformant flows on a best-effort basis if resources are available and requirements 1) and 2) are satisfied.

5.4 Packet segmentation and reassembly

LLC_TPT shall divide a packet received from the NWK layer into a sequence of fixed-length blocks for submission to the MAC layer, and shall reassemble a packet from constituent blocks received from the MAC layer prior to the transfer of the packet to the NWK layer.

6 Admission Control Services

The LLC Admission Control Service (LLC_ADM) processes reservation requests from the NWK layer.

A DAWS network is based on a client-server architecture with the MS as client, so the BS RSVP entity will never send a reservation request for an uplink flow directly to the BS_LLC. Instead, the BS RSVP entity will forward an uplink flow reservation request message issued by a correspondent host to the MS RSVP entity, which will then make a reservation request to MS_LLC_ADM. MS_LLC_ADM will not reject a reservation request due to limited bandwidth but may reject a request for another reason, for example, lack of buffer space. If the reservation request is accepted by MS_LLC_ADM, then MS_LLC_ADM will perform signalling with BS_LLC_ADM to establish the reservation. BS_LLC_ADM shall consider available free bandwidth (and other factors to be specified) when arriving at an admission control decision. BS_LLC_ADM shall base admission control decisions on statistics calculated from traffic measurements (available from BS_LLC_TPT) as well as on calculations done with TSPEC parameters. Both MS_LLC_ADM and BS_LLC_ADM shall accept a reservation request before it will be established.

LLC_ADM shall exchange service primitives with MAC_BWM in order to create and delete protocol instances which satisfy service requests received from the NWK layer.

DAWS does not provide specific requirements for the handling of reservation requests by the DAWS GW and intermediate bridges.

7 Service Primitives

7.1 Primitive Definitions

7.1.1 LLC_transfer_request

LLC_transfer_request	
Usage	BS and MS
Source	NWK Layer
Destination	LLC Layer
Service Access Point	A
Multiple Outstanding	No
SDU Parameters	*NPDU*

This primitive is used by the BS NWK layer for the downlink transfer of a *NPDU* to the NWK layer of one or more MS. This primitive is used by the MS NWK layer for the uplink transfer of a *NPDU* to the BS NWK layer.

7.1.2 LLC_transfer_confirm

LLC_transfer_confirm	
Usage	BS and MS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	A
SDU Parameters	*Transfer_receipt_ack*

This primitive acknowledges the receipt of the *NPDU* associated with a LLC_transfer_request. It does not indicate that the *NPDU* has been transferred to one or more peer SAPs.

7.1.3 LLC_transfer_indication

LLC_transfer_indication	
Usage	BS and MS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	A
SDU Parameters	*NPDU*

This primitive passes a received *NPDU* to the NWK layer.

7.1.4 LLC_create_protocol_request

LLC_create_protocol_request	
Usage	BS
Source	NWK Layer
Destination	LLC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*protocol_type*
	protocol_parameters
	destination_IP_addr

This primitive instantiates an acknowledged protocol between the BS and MS. After instantiation, BS_LLC_TPT (for a downlink protocol) or MS_LLC_TPT (for an uplink protocol) does not route LPDUs to the new protocol instance, and BS_MAC_BWM does not allocate bandwidth for the protocol. Other primitives are used to enable PDU routing and bandwidth allocation.

7.1.5 LLC_create_protocol_confirm

LLC_create_protocol_confirm	
Usage	BS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	B
SDU Parameters	*create_protocol_result*
	protocol_instance_ID

This primitive confirms the creation of a protocol instance. All subsequent primitives referencing the protocol instance shall use the returned protocol instance ID.

7.1.6 LLC_delete_protocol_request

LLC_delete_protocol_request	
Usage	BS
Source	NWK Layer
Destination	LLC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*protocol_instance_ID*

This primitive deletes a protocol instance. Any PDUs awaiting transfer or in the process being transferred are discarded.

7.1.7 LLC_delete_protocol_confirm

LLC_delete_protocol_confirm	
Usage	BS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	B
SDU Parameters	*delete_protocol_result*

This primitive confirms the deletion of a protocol instance.

7.1.8 LLC_configure_scheduling_request

LLC_configure_scheduling_request	
Usage	BS
Source	NWK Layer
Destination	LLC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*protocol_instance_ID*
	new_scheduling_state

This primitive starts or stops the allocation of network bandwidth to a particular protocol instance.

7.1.9 LLC_configure_scheduling_confirm

LLC_configure_scheduling_confirm	
Usage	BS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	B
SDU Parameters	*configure_scheduling_result*

This primitive confirms a change in scheduling status for a protocol instance.

7.1.10 LLC_configure_routing_request

LLC_configure_routing_request	
Usage	BS
Source	NWK Layer
Destination	LLC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*protocol_instance_ID*
	new_routing_state

This primitive starts or stops the routing of packets to a particular protocol instance by the packet classifier.

7.1.11 LLC_configure_routing_confirm

LLC_configure_routing_confirm	
Usage	BS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	B
SDU Parameters	*configure_routing_result*

This primitive confirms a change in routing status for a protocol instance.

7.1.12 LLC_queue_empty_notify_request

LLC_queue_empty_notify_request	
Usage	BS
Source	NWK Layer
Destination	LLC Layer
Service Access Point	B
Multiple Outstanding	Yes
SDU Parameters	*protocol_instance_ID*

This primitive configures the MAC layer to send a primitive when the input queue for a protocol instance is empty.

7.1.13 LLC_queue_empty_notify_confirm

LLC_queue_empty_notify_confirm	
Usage	BS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	B
SDU Parameters	*protocol_instance_ID*
	queue_empty_result

This primitive notifies the NWK layer when the input queue for the specified protocol instance is empty.

7.1.14 LLC_redirect_routing_request

LLC_redirect_routing_request	
Usage	BS
Source	NWK Layer
Destination	LLC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*protocol_instance_ID_1*
	protocol_instance_ID_2

This primitive substitutes *protocol_instance_ID_*2 for *protocol_instance_ID_1* in the packet classifer routing table. PDUs formerly routed to *protocol_instance_ID_1* will now be routed to *protocol_instance_ID_2*.

7.1.15 LLC_redirect_routing_confirm

LLC_redirect_routing_confirm	
Usage	BS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	B
SDU Parameters	*redirect_routing_result*

This primitive confirms a packet classifier routing table redirection.

7.1.16 LLC_hunt_request

LLC_hunt_request	
Usage	MS
Source	LLC Layer
Destination	MAC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	---

This primitive tells the MAC to perform signal strength and signal quality measurements on the serving cell (if any) and all adjacent cells.

7.1.17 LLC_hunt_confirm

LLC_hunt_confirm	
Usage	MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*hunt_result*

This primitive returns cell measurement results.

7.1.18 LLC_service_request

LLC_service_request	
Usage	MS
Source	LLC Layer
Destination	MAC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*base_station_ID*

This primitive tells the MAC to camp on the BS specified by *base_station_ID*.

7.1.19 LLC_service_confirm

LLC_service_confirm	
Usage	MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*service_result*

This primitive confirms a service request.

7.1.20 LLC_service_indication

LLC_service_indication	
Usage	MS
Source	LLC Layer
Destination	NWK Layer
Service Access Point	B
SDU Parameters	*new_service_state*

This primitive is used by the LLC to indicate a change in service state to the NWK layer.

7.2 Parameter Definitions

7.2.1 *base_station_ID*

This parameter specifies a particular DAWS BS.

7.2.2 *configure_routing_result*

configure_routing_result	
0	Success: routing configured as requested
1	Failure: configure routing request outstanding
2	Failure: specified protocol instance does not exist

7.2.3 *configure_scheduling_result*

configure_scheduling_result	
0	Success: scheduler configured as requested
1	Failure: configure scheduling request outstanding
2	Failure: specified protocol instance does not exist

7.2.4 *create_protocol_result*

create_protocol_result	
0	Success: requested protocol instantiated
1	Failure: create protocol request already pending
2	Failure: requested resources unavailable

7.2.5 *delete_protocol_result*

delete_protocol_result	
0	success: requested protocol instance deleted
1	failure: delete protocol request already pending
2	failure: specified protocol instance does not exist

7.2.6 destination_IP_addr

This is the IPv6 address of the MS with which the BS will instantiate a protocol.

7.2.7 *hunt_result*

This parameter consists of a list of available cells with associated signal strength and signal quality measurements. The parameter will be defined when the DAWS PHY is defined.

7.2.8 *new_routing_state*

new_routing_state	
0	routing to the protocol instance disabled
1	routing to the protocol instance enabled

7.2.9 *new_scheduling_state*

new_scheduling_state	
0	scheduling disabled
1	scheduling enabled

7.2.10 *new_service_state*

new_service_state	
0	service now unavailable
1	service now available (new subnet)

7.2.11 *NPDU*

Definition of this parameter is beyond the scope of the present document. Most often, it will consist of a NWK layer header and an IPv6 datagram.

7.2.12 *protocol_instance_ID*

This parameter uniquely identifies a protocol instance.

7.2.13 *protocol_parameters*

protocol_parameters	
0	Source IP address
1	Flow label
2	Reservation specification

More information on the reservation specification can be found in [5] and [4].

7.2.14 *protocol_type*

protocol_type	
0	Reserved
1	Reserved
2	Controlled-load downlink
3	Controlled-load uplink

7.2.15 *redirect_routing_result*

redirect_routing_result	
0	success: routing has been redirected
1	failure: protocol instance does not exist (number 1 or 2)

7.2.16 *queue_empty_result*

queue_empty_result	
0	success: input queue of protocol instance is empty
1	failure: protocol instance does not exist

7.2.17 *service_result*

service_result	
0	success: requested service now available
1	failure: could not complete request

7.2.18 *transfer_receipt_ack*

transfer_receipt_ack	
0	Success: receipt acknowledged
1	Failure: transfer request already pending

Annex A (informative): Hand-over Using Mobile IP

This annex describes how the registration entity in the MS LLC (MS_LLC_REG) and the Mobile IP entity in the NWK layer (MS_NWK_MIP) interact to perform discontinuous hand-over between two cells. Two scenarios are provided: inter-network and intra-network.

A.1 Inter-network hand-over

1) DAWS service is available to the MS from DAWS network X. The MS has already registered with the network on the MAC, LLC, and NWK levels.

2) MS_NWK_MIP receives the **LLC_service_indication** primitive indicating that service with the current BS is failing but another BS in a different DAWS network is available.

3) MS_NWK_MIP performs signalling with GW_NWK to release the current care-of address.

4) MS_NWK receives the **LLC_service_indication** primitive indicating that DAWS service is no longer available.

5) MS_NWK receives the **LLC_service_indication** primitive indicating that DAWS service is available from DAWS network Y.

6) MS_NWK sends the **LLC_service_request** primitive to MS_LLC_REG requesting registration with the new BS and new GW. MS_NWK receives the **LLC_service_confirm** primitive when registration is complete.

7) MS_NWK_MIP does signalling with GW_NWK to obtain a care-of IP address, possibly using a protocol such as DHCP.

8) MS_NWK_MIP sends a PDU with a binding update destination option to its home agent. The home agent will respond with a binding update acknowledgement. If route optimization is in use, then correspondent hosts should also receive binding updates.

A.2 Intra-network hand-over

1) DAWS service is available from network X. The MS has already registered with the network on the MAC, LLC, and NWK levels.

2) MS_NWK_MIP receives the **LLC_service_indication** primitive indicating that an intra-network hand-over has begun.

3) MS_NWK_MIP receives the **LLC_service_indication** primitive indicating that DAWS service is no longer available.

4) MS_NWK_MIP receives the **LLC_service_indication** primitive indicating that DAWS service is again available from network X. MS_NWK and higher layers can continue to use the care-of address granted when the MS originally registered with DAWS network X.

On the NWK level, an intra-network hand-over and an interruption in service due to other causes appear to be identical.

Annex B (informative): Reservations Using RSVP

For the purposes of this annex, the RSVP entity will be assumed to exist within the network layer (although in reality it may be more appropriately placed in the transport layer). The BS and MS implementations of RSVP will be specified by BS_NWK_RSVP and MS_NWK_RSVP, respectively.

BS_NWK_RSVP passes RSVP PDUs to peer entities but does not directly access the BS LLC. MS_NWK_RSVP has direct access to MS LLC via LLC_SAP_B. Resource reservation related activity between a BS and MS is always initiated by the MS.

Three important procedures within MS_NWK_RSVP are create reservation, delete reservation, and modify reservation. This annex describes these procedures in terms of the service primitives and service data units defined in clause 8. This annex assumes that all service requests complete successfully.

B.1 Reservation creation procedure

The final result of the reservation creation procedure is a new protocol instance supplying the requested QOS to the specified flow.

1) MS_NWK_RSVP sends **LLC_create_protocol_request**(protocol_type, protocol_parameters,packet_classification_params)

2) MS_NWK_RSVP receives **LLC_create_protocol_confirm**(create_protocol_result, protocol_instance_ID)

The BS and MS now each have a new protocol process in the MAC layer to support the new protocol instance. The packet classifier in the BS (for a downlink flow) or MS (for an uplink flow) is routing PDUs to the new instance and the bandwidth manager in the BS is allocating bandwidth to the MS for the flow.

B.2 Reservation deletion procedure

The reservation deletion procedure assumes that a reservation was made as specified in clause B.1, and is referenced by *protocol_instance_ID*. The reservation deletion procedure releases all resources associated with an existing reservation.

1) MS_NWK_RSVP sends **LLC_delete_protocol_request**(*protocol_instance_ID*)

2) MS_NWK_RSVP receives **LLC_delete_protocol_confirm**(*delete_protocol_result*)

The protocol processes in the BS and MS which supported the protocol instance have been deleted.

B.3 Reservation modification procedure

The reservation modification procedure assumes that a reservation was made as specified in clause B.1, and is referenced by *protocol_instance_ID*.

1) MS_NWK_RSVP sends **LLC_modify_protocol_request**(*protocol_instance_ID*, *new_protocol_parameters*)

2) MS_NWK_RSVP receives **LLC_modify_protocol_confirm**(*modify_protocol_result*)

The following two clauses summarize how MS_LLC will respond to a reservation modification request from MS_NWK. Resource increase and decrease requests are handled differently by MS_LLC in order to prevent temporary over-utilization of resources during the reservation modification procedure.

B.3.1 Resource increase request

For an increase in resources allocated to a downlink or uplink flow, MS_LLC will perform a tentative resource increase for the flow before sending a request to BS_LLC to do the same. The tentative resource increase will become permanent after BS_LLC acknowledges the resource increase request.

B.3.2 Resource decrease request

For a reduction in resources allocated to a downlink flow, MS_LLC will wait for an acknowledgement from BS_LLC that the resource reduction has been performed for the flow before reducing resources itself. For a reduction in resources allocated to an uplink flow, MS_LLC will perform a tentative resource reduction for the flow before sending a request to BS_LLC to do the same. The tentative resource reduction will become permanent after BS_LLC acknowledges the resource reduction request.

Annex C (informative): The IPv4 to IPv6 Transition

The terms used in this appendix can be defined as follows:

IPv6-only address: an IPv6 address from which an IPv4 address cannot be derived;

IPv6 (IPv4-compatible) address: an IPv6 address from which an IPv4 address can be derived;

IPv4-only address: an address for routing via IPv4;

IPv6-capable BS: a BS running IPv6 routing protocols;

IPv4-capable BS: a BS running IPv4 routing protocols;

host-to-router tunneling: sending an IPv6 datagram encapsulated within an IPv4 datagram to an IPv6-capable router. The router decapsulates the datagram and forwards it using IPv6 routing protocols;

host-to-host tunneling: sending an IPv6 datagram encapsulated within an IPv4 datagram to a host. The host decapsulates the datagram and sends it to the transport layer.

During the IPv4 to IPv6 transition phase, a DAWS BS and a DAWS MS may have one of three configurations:

a) IPv4 only;

b) IPv4 and IPv6 simultaneously;

c) IPv6 only.

Table C.1 summarizes the possible interactions between a DAWS BS and MS. A "P" entry means that communication on the network level is possible between the BS and MS; a "NP" entry means that communication on the network level between the BS and MS is not possible. Table C.1 shows that dual-stack BS and MS configurations maximize compatibility.

Table C.1: Possible BS-MS Combinations

		Mobile Station		
		IPv4	IPv4/v6	IPv6
Base Station	IPv4	P	P	NP
	IPv4/v6	P	P	P
	IPv6	NP	P	P

A DAWS BS should not be required to function as the source or termination of an IPv6-over-IPv4 tunnel. A DAWS BS will simply forward whatever type of packet it receives without encapsulation or decapsulation. However, a dual-stack DAWS MS may be the source or termination of an IPv6-over-IPv4 tunnel.

For a dual-stack DAWS MS, there shall be an entity in the network layer which decides which IP protocol stack to use when sending an IP datagram. In this annex, we will call this entity NWK_VSEL (for Version SELect). The dual IP stack architecture for transmission can be visualized as shown in figure C.1.

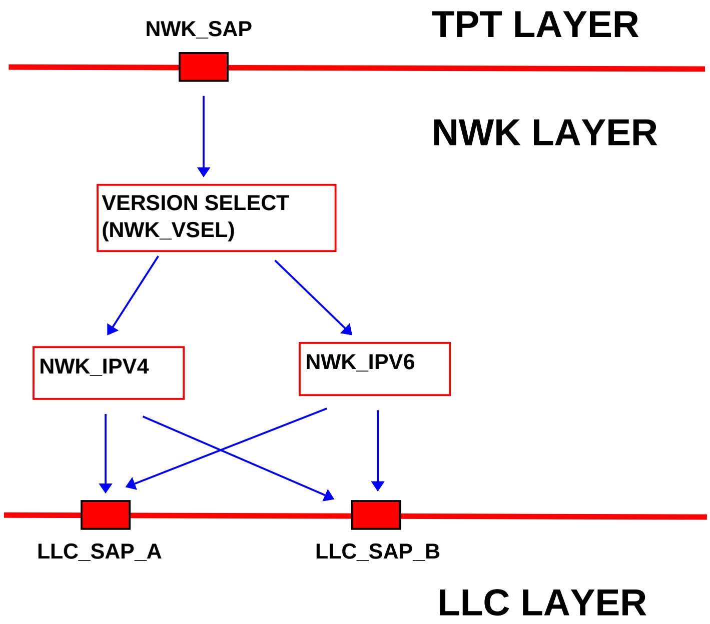

Figure C.1: Dual IP Protocol Stack Architecture - Transmission

The NWK_VSEL algorithm should favor the transmission of IPv6 packets. A possible NWK_VSEL algorithm is:

if destination address is IPv6-only

    if BS is IPv6-capable

        send IPv6 native packet

    else if BS is IPv4-capable

        if tunnel endpoint address is available

            do host-to-router tunneling

        else

            discard packet

    else

        discard packet

else if destination address is IPv6 (IPv4 compatible)

    if BS is IPv6-capable

        send IPv6 native packet

    else if BS is IPv4-capable

        do host-to-host tunneling

    else

        discard packet

else (destination address is IPv4-only)

    if BS is IPv4-capable

        send IPv4 native packet

else

discard packet

This algorithm assumes that any host which advertizes an IPv6 (IPv4-compatible) address is capable of functioning as the termination of a host-to-host tunnel.

Bibliography

The following material, though not specifically referenced in the body of the present document (or not publicly available), gives supporting information.

- RFC 2462 (1997): "IPv6 Stateless Address Autoconfiguration".
- RFC 2131 (1997): "Dynamic Host Configuration Protocol".

History

Document history		
V1.1.1	April 1999	Publication
V1.2.1	April 2000	Publication