AEEC
AIRLINES ELECTRONIC ENGINEERING COMMITTEE

To	ADN Working Group	**Date**	June 17, 2002
From	Roy Courtney rcourtne@arinc.com tel +1 410-266-4689 fax +1 410-266-2047	**Reference**	02-080/SAI-800 ldh

Subject **Working Paper Circulation**
Aircraft Data Network Working Group
June 25-28, 2002, Munich, Germany

Attachments The attachments are identified by a two digit number: the first refers to the Part number of Project Paper 664, the second identifies the input for that Part.

2-1. Proposal for Project Paper 664, Part 2, Section 1	Boeing
2-2. Proposal for Project Paper 664, Part 2, Section 3	Boeing
2-3. Presentation Charts: Electrical and Optical Reference Plane Definition	Boeing
2-4. Input on ARINC 664 ad hoc Reference Plane Definition	Rockwell Collins
3-1. Proposals for Part 3: Purpose of This Supplement App. X – Aircraft Data Network Reference Model App. Y – Air-Ground Communications Mobility for the ADN App. Z – Communications Security for the ADN	Boeing
4-1. Proposals for Part 4: Purpose of This Supplement App. G – Example for Multiple Network Devices App. H – Internet Protocol Version 6	Boeing
4-2. Working Paper on Profiling of IPv6	Smiths Aerospace
4-3. Working Paper on Profiling IPv6 RFC 2460	Smiths Aerospace
5-1. Working Paper for Project Paper 664, Part 5 – Aircraft Data Network Reference Model	Boeing

5-2.	Input on Draft 2 for Project Paper 664, Part 5	Airbus - Deutschland
5-3	Working Paper on AND 664, Part 5 – Security and Quality of Service – Functions & Network Elements Analysis	CN&S
6-1.	Aircraft Network Management White Paper	Honeywell
6-2.	Assignment of Enterprise Number by IANA	ARINC
6-3.	Project Paper 664 Part 6 Review, Network Management Specification	Connexion by Boeing
6-4.	MIB Example: Strawman MCU Interface Control Document, page i, 9-13	Connexion by Boeing
7-1.	Strawman for Part 7	Airbus-France
7.2	Stawman for Part 7 – Transport Layer IP – Requirements Summary	Airbus-France
8-1.	Working Paper V5.0 – Project Paper 664 Part 8	Computer Networks & Software, Inc.
X-1.	CANaerospace – The AGATE data bus	Stock Flight Systems

Comments & Inquiries Please review the attached material prior to the meeting. Comments and additional inputs should be directed to Roy Courtney of the AEEC staff.

ATTACHMENT 2-1

1.0 INTRODUCTION

1.1 Purpose of Document

The purpose of this document is to define Ethernet *electrical and optical reference plane parameters* ~~physical~~ and data link layer specifications for use in a commercial aircraft environment. It provides general and specific guidelines for the use of IEEE 802.3 compliant Ethernet. Physical Layer and Medium Access Control sublayers are expected to comply with the Open System Interconnection (OSI) Reference Model.

It is the intention of this document to adapt the existing applicable IEEE and ISO standards as little as possible so as to enable maximal utilization of components, both hardware and/or software for aviation use. Part 1 of ARINC Specification 664 provides an overview of an Ethernet data network. This document, Part 2, is the second part of a multi-part standard.

1.2 Scope

The aeronautical industry has recognized the need to apply standard communications protocols and services using the OSI Reference Model. The aeronautical industry has chosen to implement standard protocols within their data communications systems, as appropriate, and to use products and services which have already been developed in accordance with appropriate existing, internationally-accepted, data-communications standards.

The aeronautical industry also recognizes that certain applications that utilize data communications do not have well defined nor widely used solutions within the strict context of the OSI Reference Model. Often they have no solution at all using the OSI standard protocols. The aeronautical industry has, therefore, decided not to limit the middle and upper layer protocols associated with this Specification to those conforming to the OSI protocol suite as defined by ISO.

The OSI Reference Model was developed to provide a common basis for the coordination of standards development for the purpose of integrating data communications systems. The OSI Reference Model itself does not achieve the goal of interoperability, however, it does serve as the catalyst to enable independent development of standard protocols that ultimately achieve the goal of interoperability within the selected protocol stack (for example, OSI, TCP/IP, or others). Part 2 is based on IEEE Standard 802.3, 2000 edition.

1.3 Document Organization

1.3.1 ARINC Specification 664, Aircraft Data Network

ARINC Specification 664 defines an Ethernet Data Network for aircraft installation. It is developed in multiple parts, listed as follows:

Part 1- Systems Concepts and Overview
Part 2- Ethernet ~~Physical~~ *Electrical and Optical Parameters* and Data Link Layer Specifications
Part 3- Internet-based Protocols and Services
Part 4- Internet-based Address Structures and Assigned Numbers
Part 5- Network Interconnection Devices
Part 6- Network Management Specification
Part 7- An Example Implementation of a Deterministic Network
Part 8- Upper Layer Services for Aeronautical Telecommunication Network (ATN) and Airline Operational Control (AOC)

1.3.2 Part 2, Ethernet Physical and Data Link Layer Specification

Part 2 is organized into sections that describe the set of provisions for the implementation of stations capable of operating in various Ethernet Local Area Network (LAN) environments.

Section 1.0, Introduction, provides background information on Ethernet, and describes the relationship of Part 2 to the other Parts of ARINC Specification 664. It refers to other relevant ARINC standards.

Section 2.0, General Description, describes the relationship of specific portions of the IEEE 802.3 Ethernet standard with respect to the OSI Reference Model. Section 2.0 introduces several media type options that may be implemented to support a particular aircraft installation. Finally Section 2.0 gives a perspective on the utilities and limitations associated with using middle and upper layer protocols other than those specified by ISO.

Section 3.0, Ethernet ~~Physical Layer~~*Reference Plane Parameters* Specification, describes the ~~physical~~ *electrical and optical* characteristics of an Ethernet *signal* network and how they may be applied to a commercial aircraft installation.

Section 4.0, Ethernet Data Link Layer Specification, describes the application of the Ethernet Data Link layer in a commercial aircraft environment.

1.4 Related Documents

1.4.1 Relationship of this Document to Other ARINC Standards

A list of other ARINC documents that are related to this Specification are listed below. When avionics systems and subsystems are designed to use the capabilities provided by this Specification, they should incorporate the provisions of this Specification by reference. References to this Specification should assume the application of the most recent version of this Specification.

ATTACHMENT 2-1

1.0 INTRODUCTION

ARINC Specification 600, "Air Transport Avionics Equipment Interfaces"

ARINC Specification 628, "Cabin Equipment Interfaces (CEI)", Part 4A, "Cabin Management and Entertainment System - Cabin Distribution System (CDS) – Daisy Chain"

ARINC Specification 628, "Cabin Equipment Interfaces (CEI)", Part 4B, "Cabin Management

1.4.1 Relationship of this Document to Other ARINC Standards (cont'd)

and Entertainment System - Cabin Distribution System (CDS) – Star Wiring"

ARINC Specification 646, Ethernet Local Area Network (ELAN)

ARINC Characteristic 763, "Network Server System"

1.4.2 Relationship to Industry Standards

IEEE Standard 802.3, 2000 Edition, is considered an integral part of this specification and is considered required reading. In this document, when referencing this standard, the title is shortened to simply "IEEE 802.3."

ANSI X3.263, 1995 Edition, is an integral part of IEEE 802.3; it specifies the 100BASE-TX Physical Medium Dependent and channel performance requirements. It should also be considered required reading. In this document, when referencing this standard, the title is shortened to simply "ANSI X3.263."

ISO/IEC 11801, Edition 1.2, forms the basis of the physical media specifications and media test methods for this document and is considered required reading as well. Additional specifications are discussed throughout this document. In this document, when referencing this standard, the title is abbreviated as "ISO 11801."

1.4.3 RTCA and EUROCAE Documents

RTCA and EUROCAE develop Minimum Operational Performance Standards (MOPS) that are applicable to avionics equipment, systems and processes. The latest revision of the following RTCA and EUROCAE documents pertain to this Specification:

RTCA DO-160/EUROCAE ED-14, Environmental Conditions and Test Procedures for Airborne Equipment. In this document, when referencing this standard, the title is shortened to simply "DO-160."

RTCA DO-254 Design Assurance Guidance for Airborne Electronic Hardware

COMMENTARY

Specific levels defined in the RTCA/EUROCAE documents are specified by the aircraft systems integrator according to application.

1.4.4 IEEE and ANSI Documents

IEEE develops standards for a number of commercial industries, including the computer and telecommunications industries. Some of these standards are shared with American National Standards Institute (ANSI). On occasion, these standards are endorsed as international standards under the ISO and International Electrotechnical Committee (IEC) standardization program. The following list of IEEE and ANSI documents pertain to this Specification:

IEEE Std 802.3, 2000 Edition, Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific Requirements - Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method And Physical Layer Specifications

IEEE Std 802.2, 1998 Release, also ANSI Std 802.2, Release 1998, Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 2: Logical Link Control

IEEE Std 802, Release 1990, IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture

ANSI X3.237, 1995, Rev2.1 (1 January 1995), FDDI Low-Cost Fiber Physical Layer Medium Dependent (LCF-PMD). Note: this document is duplicated as ISO/IEC CD 9314-9

ANSI X3.263, 1995, Rev 2.2 (1 March 1995), FDDI Twisted Pair Physical Layer Medium Dependent (TP-PMD)

TIA/EIA-568-A-5 2000, Commercial Building Telecommunications Cabling Standard

1.4.5 ISO and IEC Documents

ISO/IEC 11801, Edition 1.2 (January 2000), Information Technology – Generic Cabling for customer premises.

1.5 Document Precedence

This Specification is based on IEEE Std 802.3. The philosophy of this Specification is to define changes to the provisions of IEEE 802.3 only when the aeronautical environment or user desires conflict with the provisions of IEEE 802.3, or when it is necessary to

ATTACHMENT 2-1

remove ambiguity by restricting the options available to implementers. The contents of this Specification are limited to describing these changes and option restrictions. In case of a conflict between this Specification and the applicable ISO and IEEE standards, this Specification should have precedence.

1.6 Regulatory Approval

Implementation of this standard should meet all applicable regulatory requirements. Manufacturers are urged to obtain all necessary information for such regulatory approval. This information is not contained in this specification, nor is it available from ARINC.

ATTACHMENT 2-2

3.0 ETHERNET ~~PHYSICAL LAYER PARA~~ ~~METER~~ *ELECTRICAL AND OPTICAL REFERENCE PLANE* SPECIFICATION

3.1 Introduction

This document *defines the electrical and optical signal parameters that must be met to be ARINC 664 compliant. All signals are defined and measured at the associated Reference Plane (Electrical or Optical). This specification pertains to the following* ~~recommends~~ multiple types of Physical Layer implementations~~ that are summarized as follows:~~

10BASE-2 Coaxial, half-duplex only

10BASE-T 2 twisted pair or 1 star quad, half or full duplex

100BASE-TX 2 twisted pair or 1 star quad, half or full duplex

100BASE-FX Fiber Optic, full duplex only

1000BASE-FX Fiber Optic, full duplex only

~~Wire based links are discussed in Sections 3.2.1, 3.2.2, and 3.2.3. Fiber based links are discussed in Section 3.2.4. Considerations for the LRU implementation are discussed in Section 3.3. Link components are discussed in Section 3.4. ARINC Specification 664, Part 1 provides an introduction to Ethernet in an aircraft environment.~~

3.2 Reference Plane

The reference plane is a performance boundary location that specifies the various Ethernet Physical Layer implementations transmitter and receiver signals parameters values. The reference plane is located 1 meter outside the LRU. This includes the LRU connector (rack or circular or other connector style) and 1 meter length of cable.

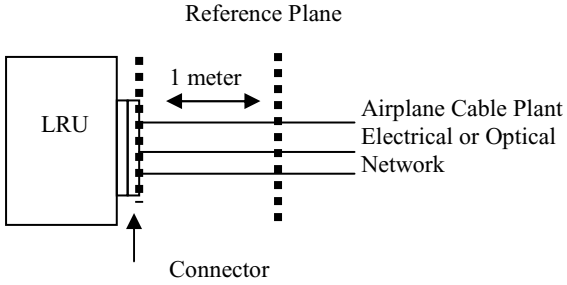


Figure 3.2 Reference Plane

The LRU portion of the link consists of everything contained within the LRU between the Physical Layer device and the LRU airplane connector and 1 meter of cable.

The aircraft portion of the link consists of the production breaks & interconnect wiring in the network path.

3.2.1 Electrical Reference Plane

LRU Waveform Compliance

Eye Pattern

3.2.2 Optical Reference Plane

Transmitter Parameters

Average Power	*-15.7 dBm Min*	*-9dBm Max*
Extinction Ratio	*5%*	
Wave Length	*1300nm*	

Receiver Parameters

Average Power	*-32 dBm Min*	*-9dBm Max*

3.2.2.1 Light Launch Conditions (move to the testing section)

ARINC 664 Light Launch Profile that all optical test equipment must meet, see Attachment 5. This is an important requirement because it assures accurate, repeatable results. The light launch specification "standardizes" a set of parameters for all pertinent launch conditions affecting accuracy and repeatability on performance measurements for fiber optic components.

3.2.2.2 Optical Fiber Geometry Specification (move to associated ARINC 700 series specification)

Core Diameter	*62.5*
Cladding Diameter	*125*
Numerical Aperture (NA)	*.275*
Core Non-Circularity	*1.0 % Max*
Core-Clad Concentricity	*1.0 % Max*
Core-Coat Concentricity	*2.0 % Max*
Coating Non-Circularity	*5.0 % Max*

3.3 Eye Pattern Mask Definition

Incorporate Rockwell Collins/Greg Sheffield Eye pattern Info

3.3.1 10 BASE-2

3.3.2 10 BASE-T

3.3.3 100 BASE-TX

3.3.4 100 BASE-FX

3.3.5 1000 BASE-FX

3.~~2~~*4* Link Details

This section provides details for several possible implementations of Ethernet on a commercial aircraft. The

mathematical means to calculate a specific link budget for an implementation are provided in each section, or in some cases, a dedicated attachment. A typical reference model is provided that shows a "typical" installation. This allows designers to model, characterize and prototype a specific budget. This diagram is provided in Attachment 1 as an example of the different installation types:

1. Traditional avionics installation. This general class of installation uses packaging and connector standards that comply with ARINC Specification 600 style.

2. Cabin (other) installations. This general class of installation should use best commercial connectors. This may include such systems as in-flight entertainment, file server, gatelink, as well as water and waste systems.

3. Severe environment installations as in engine and wing installations may alternately use circular style connectors rather than ARINC 600 connectors.

In specifying any component of the link, several factors, in addition to the electrical performance parameters, should be taken into consideration. These include, but are not limited to:

1. Variation of the electrical parameters (i.e., attenuation) with temperature

2. Variation of the electrical parameters (i.e., crosstalk) with manufacturing and installation tolerances

3. The effects of corrosion, vibration, HIRF, and lightning

Components should be specified with min/max values rather than a nominal value, with temperature range included. Components with a single-point (i.e., at 25C) nominal specification should be verified by the system integrator to the expected environmental range.
On any twisted wire link, all crossover should be implemented in the cable. For example, the pins labeled TX on any LRU should be connected to the transmitter within the LRU. By extension, the TX pins at one end of the link should be connected to the RX pins at the other end of the link. To avoid confusion, the "MDI-X" standard, as defined in IEEE 802.3, should not be implemented.

3.~~2~~*3*.1 10BASE-2 Ethernet

10BASE-2 can be described as a shared-bus link technology using coaxial cable. The sections below outline the special provisions of installing 10BASE-2 in an aircraft environment, beyond those specified by IEEE 802.3. The coaxial cable provisions are defined in Section 3.4.1.1.

3.~~2~~*3*.1.1 Special Interface Provisions

10BASE-2 should be connected in a daisy chain (or in-and-out) fashion physically, and have a passive bus topology electrically. The daisy chain connection limitation for 10BASE-2 stems from the high impedance coaxial cable tapping method used in the commercial implementation; BNC "T" taps are used so as to minimally disturb the propagating signal on the coaxial cable. The connection lengths from the "T" to the active receiver circuitry should be minimized due to the signal attenuation and jitter that may occur as a result of the relatively large coaxial cable capacitance.

In general, the interface design should be compatible with RTCA DO-160 environmental conditions for the specific aircraft category.

Specific recommendations are as follow:

1. Avoid multi-point ground 10BASE-2 systems below 200Hz.

2. Provide insulative coaxial contacts on LRU connectors and disconnects.

3. Provide capacitive coupling from coaxial shield to ground as required for RFI emissions limitations.

3.~~2~~*4*.1.2 Special Termination and Bonding Provisions

Grounding (earthing) of the 10BASE-2 coaxial cable shield may be provided at a maximum of one location per LAN segment. Multi-point grounding may lead to current loops that can interfere with the detection of collisions and collision signals on the network. Therefore, with the exception of exactly one LRU per LAN segment, all LRUs must provide dc isolation of the coax shield from ground. This is likely to require the use of special non-conductive coaxial cable inserts or a special coaxial cable connector.

All stations except the one for which a coax shield ground exists on a 10BASE-2 LAN segment should provide ac (capacitive) coupling of the coax shield to ground.

In general, the interface design should be compatible with RTCA DO-160 environmental conditions for the specific aircraft category.

3.2.1.3 Additional Connector Standards

Connectors should functionally conform to IEEE 802.3 specification. In addition, connectors should comply with the provisions of ARINC Specification 600 for a coaxial insert.

3.~~2~~*4*.1.4 Termination Standards

Coaxial cable termination specifications are per IEEE 802.3 with no added limitations.

COMMENTARY

ARINC Specification 646, Ethernet Local Area Network, provides general and specific design and implementation guidelines for use of IEEE 802.3 compliant Ethernet Local Area Networks (ELAN) in the commercial avionics environment that accommodate 10 Mbps networks.

3.0 ETHERNET PHYSICAL LAYER SPECIFICATION

3.~~2~~3.2 10BASE-T Ethernet

10BASE-T can be described as a point-to-point link technology that may be implemented using twisted wire. The use of repeaters and the resulting star system configuration has many advantages in some areas of distribution within airborne networks. This is the basic 10BASE-T topology.

COMMENTARY

The signal characteristics of 10BASE-T using only unshielded twisted wire (as specified in IEEE 802.3) may lead to electromagnetic incompatibility with respect to commercial airplane use. It is recommended that only shielded twisted wire cable be used in the aircraft applications.

3.~~2~~4.2.1 Link Budget

The consideration of the link budget is the same for both Profiled Networks and Complaint Networks defined in Part 1.

The link budget for a 10BASE-T Link, including all tolerance effects such as temperature, is as follows:

Parameter	Units	Limits
Insertion Loss	dB	less than 11.5 dB for 5 < f < 10 MHz
NEXT Loss	dB	less than 23-15log(f/10) for 5 < f < 10 MHz. This is the multiple-disturber value.
Differential Noise	mV	less than 264mV, crosstalk and external noise summed. Voltage is measured through a 3-pole Butterworth low-pass with a 15 MHz 3 dB cutoff
Propagation Delay	µs	not greater than 1000.
Jitter	ns	±5

10BASE-T equipment should be designed to operate in a 100BASE-TX network installation, albeit at low bit rates. It is recommended that a 100BASE-TX link be designed to support equipment from a 10BASE-T application. This should provide additional link margin and allow future growth to 100BASE-TX without a change in the installed cable plant.

It is further recommended that the link budget be divided between LRU and Aircraft Link in a manner similar to a 100BASE-TX system. Section 3.2.3 outlines the division of the link budget between LRU Link and Aircraft Link for 100BASE-TX systems.

3.~~2~~4.3 100BASE-TX Ethernet

100BASE-TX can be described as a point-to-point link technology that may be implemented using twisted wire. The basic topology is a star system using hubs or switches.

100BASE-TX PMD and medium are specified in IEEE 802.3 by incorporating the FDDI TP-PMD standard, ANSI X3.263-1995. TIA/EIA 568-A Category 5 unshielded twisted pair, and 150 ohm shielded twisted pair are FDDI TP-PMD options. The Aircraft Data Network should use 100 ohm shielded twisted wire cable because it is suited for on-aircraft use. The signal characteristics of 100BASE-TX using only unshielded twisted wire (as specified in IEEE 802.3) may lead to electromagnetic incompatibility with respect to commercial airplane use.

3.~~2~~4.3.1 Link Budget

The link budget analysis is the same for Profiled Networks and Compliant Networks defined in Part 1 of ARINC Specification 664.

A link reference model was developed for twisted wire Ethernet (10BASE-T or 100BASE-TX) analysis. This reference model represents typical installations that are found on board aircraft. It is recognized that individual installations may vary considerably from this reference. All connections are assumed to be symmetrical in construction. Attachment 1, Air Transport Category Reference Physical Layer, provides specific details of the model.

The link reference model has two distinct sections: the LRU portion and the aircraft portion. The reference plane is the aircraft rack to LRU interface boundary. The reference plane splits the LRU connector in half. One half is the LRU portion of the link and the other half is the aircraft portion of the link.

The LRU portion of the link consists of everything contained within the LRU between the Physical Layer device and the LRU side of the connector.

The aircraft portion of the link consists of everything in the link from the rack side connector of one tray to the rack side connector of a second tray, including production breaks.

In IEEE 802.3, it is assumed that the link endpoints have negligible loss and crosstalk. This is not the case in most avionics applications. Most LRU architectures do not

3.~~2~~4.3.1 Link Budget (cont'd)

allow the PHY to be close enough to the connector to be negligible. Therefore, some of the link budget allocated to aircraft should more accurately be allocated to the LRU.

COMMENTARY

A certain amount of compensation is used in the test fixtures for testing half a connector as the connector performance parameters can only be accurately measured in their mated state.

Attachment 4, 100BASE-TX Link Budget Calculations, gives the illustrations and equations from ANSI X3.263 to calculate the permissible link parameters for any complete link, over the entire range of temperature and

manufacturing variables. Division of the complete link into LRU and Aircraft Link Budgets is at the discretion of the system integrator. In no case should the link budget choices of the system integrator directly or indirectly mandate the use of Physical Layer devices with capabilities beyond those specified by IEEE 802.3 and this standard.

COMMENTARY

It is recommended that the values selected for the LRU and aircraft link budgets approach those used in Appendices C, D, and E. These values were chosen to accommodate most LRU architectures and cable plants, and are based on measurements of actual devices. Choosing values outside those practically feasible with a system may have a significant cost impact. Choosing values not technically feasible may result in a non-compliant system.

3.~~2~~4.3.2 Traditional Avionics Installation

In a traditional avionics installation, the link reference model assumes that ARINC 600 connectors are used to connect the LRU to the aircraft cable. However, in some cases (such as connecting to engine control units) the end unit may utilize other connectors appropriate to their environment. The reference model assumes circular style connectors for production/section breaks. For systems which may be at the transmit or receive end of an Ethernet cable plant and for which ARINC 600 connectors are not appropriate, the circular style connectors shown as the intermediate connectors can be substituted in the cable plant instead of ARINC 600 connectors.

An example of a typical avionics installation link is illustrated in Appendix C. The suggested values for the LRU Link and Aircraft Link portions of the budget are also calculated in Appendix C.

3.~~2~~4.3.3 Cabin (other) Installation

Aircraft Links for typical cabin equipment installations are defined in ARINC Specification 628, Parts 4A and 4B.

An example cabin installation link is illustrated in Appendix D. The suggested values for the LRU Link and Aircraft Link portions of the budget are also calculated in Appendix D.

3.~~2~~3.3.4 Severe Environment Installation

Aircraft Links in severe environments use circular style connectors at both ends, and typically have fewer than 8 production breaks.

An example severe environment link is illustrated in Appendix E. The suggested values for the LRU Link and Aircraft Link portions of the budget are also calculated in Appendix E.

3.~~2~~3.4 100BASE-FX Ethernet

Text for this section is planned for a future Supplement.

COMMENTARY

For data rates greater than 100 Mbps, fiber optic solutions are recommended.

3.~~3~~5 Equipment Physical Layer Design Considerations

3.~~3~~5.1 PMD and MDI Design Considerations

The LRU PMD and MDI design have the largest impact on the performance of the LRU Link. Component choices here can also impact the Aircraft Link performance.

COMMENTARY

It is possible to obtain or to fabricate transmitters and receivers that perform beyond the specifications listed in IEEE 802.3. However, unless the supplier can guarantee to the customer that this extra performance is not actually needed, the use of such "extended capability" parts is discouraged. This maximizes the probability of successful interoperability with mixed components.

The transformers should be compliant with the electrical parameters of IEEE 802.3 and the PHY device. The high and low temperature requirements of RTCA DO-160 need to be taken into consideration as commercial Ethernet transformers may not meet critical electrical parameters at the temperature extremes, and the link may cease to operate, or its bit error rate may increase dramatically. A resistor of greater than 300 kohms and less than 5 Mohms should be installed from the transmit transformer's center tap to ground on the cable plant side, to act as a static drain.

The assembly is assumed to be installed in an internal module that is connected via a supplier specified internal connector similar to that for a backplane. Prior to leaving (entering) the LRU, additional protection for High Intensity Radiated Field (HIRF) and lightning may be necessary. The inclusion of this protection depends on operating scenarios and installation locations.

COMMENTARY

The inclusion of HIRF/Lightning protection comes with the cost of decreased signal available for the cable. 100BASE-TX Ethernet has significant high frequency components at 62.5 MHz and above, thus requiring careful design of this protection. Typically,

it is desirable to maximize the length of the cable between transmitter and receiver, and so, designers may choose to forgo the protection offered by HIRF/Lightning protection. In addition, the loss of function resulting from HIRF/Lighting damage should also be taken into consideration when forgoing HIRF/Lightning protection for transmitters and receivers. If the consequences of permanent component damage, or transitory upset, are unacceptable both to the end user and from financial considerations in installations that are highly susceptible to lightning, then consideration should be given to the use of a fiber optic cable to minimize

3.0 ETHERNET PHYSICAL LAYER SPECIFICATION

these risks.

The backplane and connectors have an inherent degradation of the signal characteristics. This degradation needs to be considered as part of the LRU Link budget.

3.3.5.2 Upper Sublayers Design Considerations

The upper sublayers of the Physical Layer are implemented according to IEEE 802.3 exactly. This means that the ADN PMA and PCS for 100BASE-TX are implemented exactly as the IEEE 802.3 PMA and PCS. In addition, the ADN PLS and PMA for 10BASE-T are implemented exactly as the IEEE 802.3 PLS and PMA.

COMMENTARY

The AUI may need additional consideration for 10BASE-T LRU implementations that place the MAU outside the LRU.

3.4 Aircraft Network Components

The following sections describe the various components of the aircraft network. The specific components of the link include the cable and connectors.

3.4.1 Copper Cable

Copper cable may be used for 10BASE-2, 10BASE-T and 100BASE-TX installations where it is shown to meet long-term reliability and maintainability requirements for airlines use.

3.4.1.1 Coaxial Cable

Coaxial cable may be used for 10BASE-2 links. One cable should be installed for each link segment.

The signal characteristics of 10BASE-2 using single shielded coaxial cable that only conforms to the characteristics and parameter values specified as limitations in IEEE 802.3 could lead to electromagnetic incompatibility with respect to commercial airplane use. Cabling must meet RTCA DO-160 requirements described in: Section 21 - Emission of Radio Frequency Energy.

10BASE-2 coaxial cable parameters necessary to provide electromagnetic compatibility are as listed in Table 3-1.

Description	Coaxial Parameters
Continuous Working Voltage	300 V RMS
Characteristic Impedance	Zo = 50 ± 0.5 ohms
Attenuation	< 3.2 dB per 100 meters at 5 MHz
	< 4.5 dB per 100 meters at 10 MHz
	> 9 dB per 100 meters over 50 MHz
Velocity of Propagation	> 75% c
Conductor size	20 AWG
Capacitance	< 85 pf/m
Operating Frequency	0-40 MHz

Table 3-1 - 10BASE-2 Coaxial Cable Parameters

3.4.1.2 Shielded Twisted Wire Cable

Shielded twisted wire cable may be used for 10BASE-T and 100BASE-TX installations. There are two basic construction techniques: twisted pair and star quad. Two twisted pairs or one star quad are needed for a link.

This subsection describes the cable construction and various installation applications. Shield termination is discussed in the connector section.

Cable performance characteristics for 10BASE-T and 100BASE-TX installations are defined in Attachment 2. This material is derived from ISO 11801 Category 5 cable specifications, with additions and modifications recommended based on specific airplane environmental and performance needs. It is expected that several types of cable construction should satisfy these performance parameters. Twisted pair, star quad cable and other cable constructions may be applied to meet the specific aircraft installation requirement.

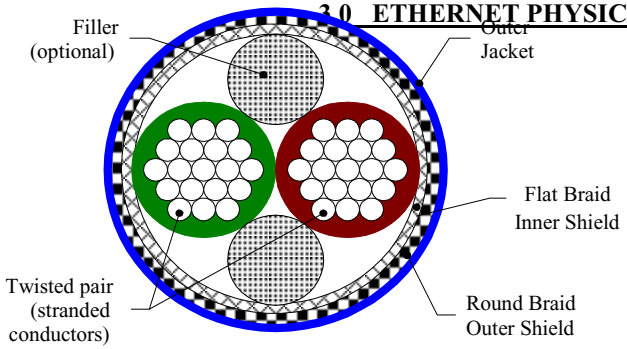
Appendix B summarizes characteristics for an avionics data network media applicable to 10BASE-T or 100BASE-TX operations that the Ethernet network designer may apply as guidance.

COMMENTARY

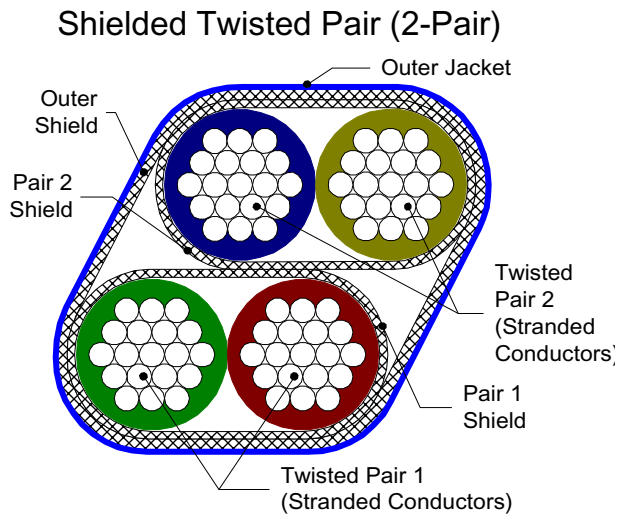
New installations should use cables that meet or exceed the performance parameters of Attachment 2. This should allow future expansion in 10BASE-T systems to 100BASE-TX communications without the need for aircraft modification.

3.4.1.2.1 Shielded Twisted Pair Cable

Shielded twisted pair cable, shown in Figure 3-1, is one cable construction method. Two pairs are used per link. It is constructed by twisting two wires together and enclosing them in an overall shield. Two versions exist which are considered here: 1-pair and 2-pair cable. The 1-pair cable comprises a single twisted pair, which is encased in one or two shields. It is recommended that the double-shielded configuration of the 1-pair cable be used with twinax contacts. The 2-pair cable comprises two shielded twisted pairs, with an overall outer shield. The 2-pair cable is typically used with standard contacts. Cable uniformity is key - this determines the crosstalk and impedance variability of the cable. Advantages include better compatibility with twin-axial connectors, and better high-frequency response as compared to an equivalent star quad cable.



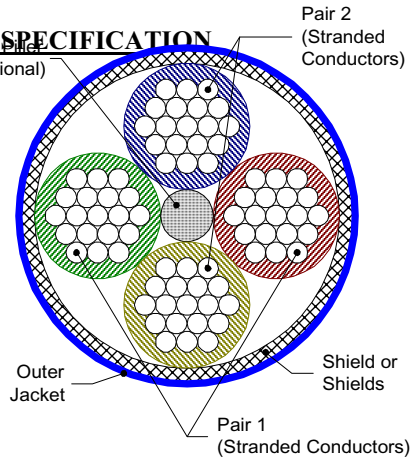
~~Figure 3-1 Shielded Twisted Pair Cable (1-Pair)~~



~~Figure 3-2 Shielded Twisted Pair Cable (2-Pair)~~

~~3.4.1.2.2 Star Quad Cable~~

~~Star quad (SQ) cable, shown in Figure 3-3, is also an acceptable cable construction technique. One quad is required per link. Also known as quad lay, quad axial, and twisted quad cable, it consists of four parallel wires uniformly twisted around a center filler. The four twisted wires are then encased in an overall shield. Cable uniformity is key as this is what determines the crosstalk and impedance variability of the cable. Advantages include: weight and space savings as compared to an equivalent twisted-pair cable, and better compatibility with quadrax connectors.~~



~~***Figure 3-3 Star Quad Cable***~~

~~3.4.1.2.3 Traditional Avionics Installations~~

~~The EE bay, where traditional avionics reside, offers forced cooling air, physical proximity to other flight critical avionics, shorter cable runs, fewer production breaks, and other unique environmental conditions. The cable described in Attachment 2 may be used in 10BASE-T and 100BASE-TX installations.~~

~~The system integrator should maintain installation flexibility as described by Section 3.2.3 over the full temperature range. It is understood that over the full length of the link, only a portion of the cable will be exposed to the extremes of the temperature range. It is also understood that a certain degradation of the signal characteristics needs to be accounted for. The cable selection should minimize installation limitations.~~

~~3.4.1.2.4 Cabin (other) Installations~~

~~Cabin installations typically have long cable runs, many production breaks, potentially hazardous cable routing in terms of temperature extremes, and sometimes accessibility to passengers.~~

~~The cable defined in Attachment 2 may also be used for cabin applications.~~

~~3.4.1.2.5 Severe Environment Installations~~

~~Wing installations, and other severe environment networks may use the cable defined in Attachment 2.~~

~~3.4.2 Fiber Cable~~

~~Text for this section is planned for a future Supplement.~~

~~COMMENTARY~~

~~Fiber optic media and connectors are specified in ARINC~~

ATTACHMENT 2-2

3.0 ETHERNET PHYSICAL LAYER SPECIFICATION

Specification 628 Part 6, "Cabin Equipment Interfaces (CEI), Part 6, Fiber Optic Cable Assembly General Specification".

3.4.3 Connectors

Different connectors, summarized in Table 3-2 – Connector Option Summary, are used in each of the three installation types. Connector technology should be based on which IEEE 802.3 technology is used (for example, 10BASE-T or 100BASE-TX.) This specification suggests three fundamentally different connector technology types to be used as described in the following sections. Within each connector technology type exists different options as defined in Section 3.4.3.2.

Table 3-2 Connector Option Summary

Connector Type	Option	Network	Contact Style
ARINC 600	1A	10BASE-T only	22 AWG pins
	1B	10BASE-T/ 100BASE-TX	22 AWG pins
	1C	10BASE-T/ 100BASE-TX	size 8 quadrax contact
	1D	10BASE-T/ 100BASE-TX	size 8 twinaxial contact
	1E	100BASE-FX only	TBD
ARINC 404	2A	10BASE-T/ 100BASE-TX	22 AWG pins
	2B	10BASE-T/ 100BASE-TX	size 8 quadrax contact
	2C	10BASE-T/ 100BASE-TX	size 8 twinaxial contact
	2D	100BASE-FX only	TBD
Circular Style	3A	10BASE-T/ 100BASE-TX	22 AWG Pins
	3B	10BASE-T/ 100BASE-TX	size 8 quadrax contact
	3C	10BASE-T/ 100BASE-TX	size 8 twinaxial contact
	3D	100BASE-FX only	TBD

3.4.3.1 Connector Technology

This section discusses the various connector options to use for typical installations.

3.4.3.1.1 Traditional Avionics Installation

Traditional avionics installations commonly comprise rack mounted LRUs, and one or more production breaks. ARINC Specification 600 connectors/contacts, ARINC 404 connectors/contacts and circular style connectors are recommended for this environment. Multiple solutions are expected to be developed to enable ARINC Report 664 Ethernet connections to be included in ARINC Specification 600 connector inserts. Several examples are described below. Some installations may connect to the flight deck or other non-avionics bay locations.

3.4.3.1.1.1 10BASE-T

For 10BASE-T connections, ARINC 600 option 1A is recommended for LRUs. Options 1B, 1C, or 1D also accommodate 10BASE-T systems while providing for growth to 100BASE-TX systems.

3.4.3.1.1.2 100BASE-TX

For 100BASE-TX connections, ARINC 600 options 1B, 1C, or 1D are recommended for LRUs.

3.4.3.1.1.3 100BASE-FX

For 100BASE-FX connections, ARINC 600 option 1E is recommended for LRUs.

3.4.3.1.2 Cabin (other) Installation

Many connector technologies are used in the cabin, including a combination of circular, ARINC Specification 404A, ARINC Specification 404B, D-subs, coax, and other rectangular connectors.

3.4.3.1.2.1 10BASE-T

For 10BASE-T connections, ARINC Specification 628 Parts 4A and 4B, and ARINC Specifications 404A and 404B provide guidance. Option 2A is included in this recommendation. Options 2B and 2C also accommodate 10BASE-T systems while providing growth to 100BASE-TX systems.

3.4.3.1.2.2 100BASE-TX

For 100BASE-TX connections, ARINC Specifications 628 Parts 4A and 4B, and ARINC Specifications 404A and 404B provide guidance. Options 2A, 2B, and 2C are included in this recommendation. The application of Option 2A should be similar to Option 1B.

3.4.3.1.2.3 100BASE-FX

For 100BASE-FX connections, ARINC Specification 628 Part 6, "Cabin Equipment Interfaces (CEI), Part 6, Fiber Optic Cable Assembly General Specification" provides guidance. Option 2D is included in this recommendation.

3.4.3.1.3 Severe Environment Installation

3.0 ETHERNET PHYSICAL LAYER SPECIFICATION

For the severe environment installation, circular style connectors are recommended.

3.4.3.1.3.1 10BASE T

For 10BASE T, circular style option 3A is recommended. Options 3B and 3C also accommodate 10BASE T systems while providing growth to 100BASE TX systems.

3.4.3.1.3.2 100BASE TX

For 100BASE TX, circular style options 3A, 3B and 3C are recommended. Application of Option 3A should be similar to Option 1B.

3.4.3.1.3.3 100BASE FX

For 100BASE FX, circular style option 3D is recommended.

COMMENTARY

Use of fiber optics for severe environments is for further study, but progress on media and connector technology is being advanced by ARINC Specification 628 Part 6, "Cabin Equipment Interfaces (CEI), Part 6, Fiber Optic Cable Assembly General Specification".

3.4.3.2 Connector Options

This section describes various connector technologies that may be used in an ARINC Report 664 Ethernet installation. This information should be considered in the design of new aircraft installations and for retrofit. The responsibility for the specific connector selection resides within the purview of aircraft system integrator and the specific ARINC 700 series Characteristic designer. Each connector example is called a "type". Choices within a connector type are named "options". The three types are:

ARINC Specification 600, Air Transport Equipment Interfaces

ARINC Specification 404A, Air Transport Equipment Cases and Racking, and ARINC 404B, Connectors, Electrical, Rack and Panel, Rectangular, Rear Release Crimp Contacts

Circular style connectors.

3.4.3.2.1 ARINC SPECIFICATION 600 Options

ARINC Specification 600 defines connector shells, inserts and contacts recommended for air transport equipment installations. Many technologies can be inserted into the ARINC Specification 600 connector shells. The following options should be considered for ARINC Report 664 Ethernet installations.

3.4.3.2.1.1 Option 1A

This option populates a standard ARINC Specification 600 insert with size 22 AWG signal pins. This option should demonstrate adequate performance and EMI protection to terminate 10BASE T signals. No additional backshells or other measures should be needed to ensure proper operation. Pigtails are adequate for shield termination for all but the most severe environments. However, special pin arrangements are needed. It is recommended that the pinout arrangements for option 1B be followed as appropriate for the cable type.

COMMENTARY

System integrators should take care to locate Ethernet signals away from power or analog signals.

Options 1B, 1C, and 1D meet or exceed the specifications of option 1A.

3.4.3.2.1.2 Option 1B

This option populates a standard ARINC Specification 600 insert with size 22 AWG signal pins. This option is intended for 100BASE TX signals. Signals on twisted pair cable and star quad cable should be located in different physical arrangements. The goal is to optimize cable hookup and to ensure that proper performance and EMI specification are met. Appendix G provides suggested pinouts and recommended practices for using size 22 AWG pins for 100BASE TX.

Many pin arrangements are suitable for use with STP cable and size 22 AWG pins. However, such arrangements should be analyzed carefully and tested to ensure that the signal specifications of IEEE 802.3 and the link budget with regard to loss and crosstalk are taken into consideration as well as the appropriate sections and categories from RTCA DO-160.

Any size 22 AWG pin termination scheme should take into account the physical parameters for two separate cables being terminated in close physical proximity. Access for insertion of the pins into, and extraction from, the connector insert need to be considered as well.

3.4.3.2.1.3 Option 1C

This option populates an ARINC Specification 600 insert with a size 8 multi-pin contact called "quadrax". This option is expected to provide a compact, ruggedized, high performance Ethernet connection solution. Four signal pins are included within each size 8 quadrax contact, which yields one Ethernet port per quadrax contact. Several quadrax contacts may populate an ARINC Specification 600 insert. For example, in an ARINC Specification 600 size 2 insert, up to 11 quadrax contacts may be included. In this option, no additional backshell should be required. The detail of the size 8 quadrax contact type is defined in Attachment 20 to ARINC Specification 600.

COMMENTARY

Quadrax inserts are typically used with star quad cable.

All equipment using the quadrax contact for Ethernet signals should use the standard pin allocation illustrated in Figure 3-4. This illustration depicts the equipment side interface, as viewed from the reference plane toward the

equipment (the mating end view).

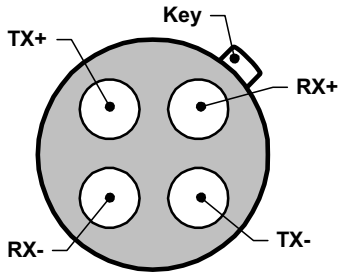


Figure 3-4 Quadrax Receptacle Signal Allocation on Equipment (mating end view)

3.4.3.2.1.4 Option 1D

This option populates an ARINC Specification 600 insert with a Size 8 concentric twinax contact, providing a compact, ruggedized, high performance solution. Two signal contacts are provided per each twinax, yielding one half of an Ethernet port (TX or RX). The twinax configuration incorporates a 360 degree shield around the signal contacts; no additional backshell is required. The twinax contact is compatible with ARINC Specification 600 connector inserts. The twinax contact is intermateable with contacts currently utilized in ARINC Specification 629 data bus applications. Several twinax contacts may populate an ARINC Spe ation 600 insert. For example, an ARINC 600 type 12 insert can accommodate up to 10 twinax contacts. See ARINC 600 for specific available insert arrangements.

COMMENTARY

Twinax contacts are typically used with 1 pair shielded twisted pair cable.

All equipment using the twinax contact for Ethernet signals should use the standard pin allocation illustrated in Figure 3-5.

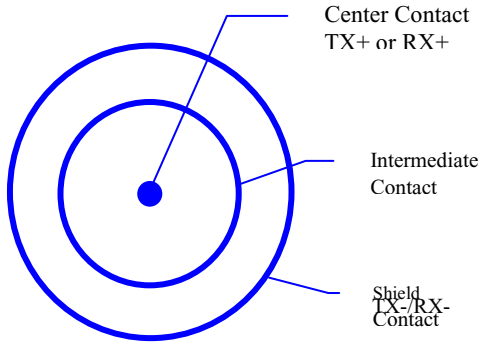


Figure 3-5 Twinax Contact Signal Allocation

The detail of the size 8 twinax contact type is defined in Attachment 21 to ARINC Specification 600.

3.4.3.2.1.5 Option 1E

Text for this section is planned for a future Supplement.

COMMENTARY

Fiber optic media and connectors are specified in ARINC Specification 628, "Cabin Equipment Interfaces (CEI), Part 6, Fiber Optic Cable Assembly General Specification".

Attachment 11 to ARINC Specification 600 specifies fiber optic contacts that are expected to be suitable for Ethernet installations.

3.4.3.2.2 ARINC 404A and 404B Options

ARINC Specification 404A and 404B define connector shells, inserts and contacts recommended for air transport installations. Many technologies can be inserted into the ARINC Specifications 404A and 404B defined inserts. The following options should be considered for ARINC Report 664 Ethernet installations.

3.4.3.2.2.1 Option 2A

Standard ARINC 404A or 404B inserts with size 22 AWG inserts into a standard connector insert provides a high density solution suitable for 10BASE T and 100BASE TX signals. Signals should be located in accordance with the guidelines in Option 1B. Appendix G also provides guidance on the usage of size 22 AWG pins. Similar recommended practices also apply.

3.4.3.2.2.2 Option 2B

Four signal pins are included within a size 8 quadrax contact, which yields one Ethernet port per quadrax contact. Guidelines for using the size 8 quadrax contacts are discussed as Option 1C.

3.4.3.2.2.3 Option 2C

Text for this section is planned for a future Supplement.

3.4.3.2.2.4 Option 2D

Text for this section is planned for a future Supplement.

COMMENTARY

Fiber optic media and connectors are specified in ARINC Specification 628 Part 6, "Cabin Equipment Interfaces (CEI), Part 6, Fiber Optic Cable Assembly General Specification".

3.4.3.2.3 Circular Style Options

Circular style connectors are widely used for production breaks, special environments, and the flight deck. The options recommended for Ethernet installations are described in the following sections.

3.0 ETHERNET PHYSICAL LAYER SPECIFICATION

~~3.4.3.2.3.1 Option 3A~~

~~The use of standard size 22 AWG signal pins with a circular style (any series) shell meets the electrical characteristics expected for 10BASE-T and 100BASE-TX.~~

~~3.4.3.2.3.1 Option 3A (cont'd)~~

~~The guidelines for use of size 22 AWG pins in an ARINC 600 connector, discussed previously as Option 1B, should be followed as well.~~

~~COMMENTARY~~

~~Circular style connectors may be used for production breaks as well as for connections to individual units, which do not use ARINC Specification 600 style connectors. If used in "exposed" locations (per RTCA DO-160) additional testing may need to be performed to confirm the ability of the connector to work within the specified environment. Additional shielding or 360 degree grounding of the shield may be required to meet performance standards for the RTCA DO-160 worst case environmental testing.~~

~~3.4.3.2.3.2 Option 3B~~

~~Four signal pins are included within a size 8 quadrax contact, which yields one Ethernet port per quadrax contact. Guidelines for using the size 8 quadrax contacts are discussed as Option 1C.~~

~~3.4.3.2.3.3 Option 3C~~

~~Two signal contacts are included in a size 8 twinax contact, which yields one half Ethernet port per cavity (TX or RX). Guidelines for using the size 8 twinax contacts are discussed as Option 1D.~~

~~3.4.3.2.3.4 Option 3D~~

~~Text for this section is planned for a future Supplement.~~

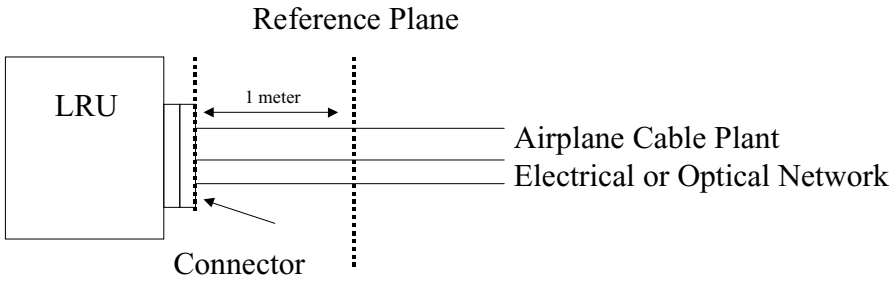
~~COMMENTARY~~

~~Fiber optic media and connectors are defined in ARINC Specification 628, "Cabin Equipment Interfaces (CEI), Part 6, Fiber Optic Cable Assembly General Specification."~~

Attachment 2-3

Electrical And Optical Reference Plane Definition

The interface that defines the boundary between the LRU and the Airplane cable plant.



At this reference plane, the LRU transmitter and receiver & the Airplane Cable Plant characteristics are independently specified.

D.E. Anderson, The Boeing Company, February 12, 2002

3.2 Reference Plane

The reference plane is performance boundary that specifies the Ethernet electrical and optical values. These values must be met to be an ARINC 664 compliant LRU.

The reference plane is located 1 meter passed the LRU. This will include the LRU connector (rack or circular) and 1 meter of cable.

The LRU portion of the link consists of everything contained within the LRU between the Physical Layer device and the LRU airplane connector and 1 meter of cable.

The aircraft portion of the link consists of the production breaks & interconnect wiring in the network path.

Commentary

This is the location that the LRU would typically be tested at during Acceptance Test Procedures and Qualification Testing.

D.E. Anderson, The Boeing Company, February 12, 2002

Attachment 2-3

3.2.1 Electrical Reference Plane

LRU manufactures & airframers – need to agree upon various Ethernet signal values

LRU Waveform Compliance

Eye Pattern

A working meeting will be held March 26th – 28th at Seattle. The goal will be to completely define the Electrical and Optical Parameters and Values at the Reference Plane.

Contact – Derek Anderson, derek.e.anderson@boeing.com 425.294.0953 if you would like to participate.

3.2.2 Optical Reference Plane

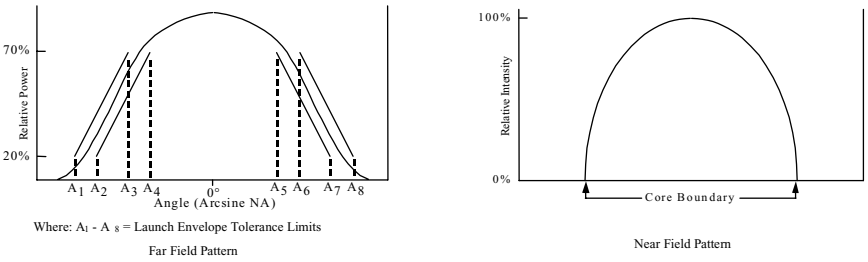
The output optical signal generated by a station as measured at the reference plane.

Transmitter Parameters		
Average Power	-15.7 dBm Min	-9dBmMax
Extinction Ratio	5%	
Wave Length	1300nm	

Receiver Parameters		
Average Power	-32 dBm Min	-9dBm Max

Attachment 2-3

Light Launch Condition Specification



Where: A_1 - A_8 = Launch Envelope Tolerance Limits

Far Field Pattern

Near Field Pattern

The Boeing 777 program developed a Light Launch Profile that all optical test equipment must meet. This is an important requirement because it assures accurate, repeatable results. The light launch specification "standardizes" a set of parameters for all pertinent launch conditions affecting accuracy and repeatability on performance measurements for fiber optic components. The bottom line is, if a measurement is done with test equipment that does not meet the light launch conditions it is meaningless.

D.E. Anderson, The Boeing Company, February 12, 2002

3.2.2.2 Optical Fiber Geometry Specification

Core Diameter	62.5
Cladding Diameter	125
Numerical Aperture (NA)	.275
Core Non-Circularity	1.0 % Max
Core-Clad Concentricity	1.0 % Max
Core-Coat Concentricity	2.0 % Max
Coating Non-Circularity	5.0 % Max

D.E. Anderson, The Boeing Company, February 12, 2002

Attachment 2-3

3.2.2.1 Light Launch Conditions

ARINC 664 Light Launch Profile that all optical test equipment must meet, see Attachment 5. This is an important requirement because it assures accurate, repeatable results. The light launch specification "standardizes" a set of parameters for all pertinent launch conditions affecting accuracy and repeatability on performance measurements for fiber optic components.

Reference Plane Specification Development Meeting

A working together meeting will be held this March 26th – 28th at Seattle. The goal will be to completely define the Electrical and Optical Parameters & Values at the Reference Plane.

Contact – Derek Anderson, derek.e.anderson@boeing.com
425.294.0953 if you would like to participate.

ATTACHMENT 2-4

Rockwell Collins Inc.

Greg Sheffield
400 Collins Road NE
Cedar Rapids, IA 52498
M/S 108-166

April 5, 2002

Roy Courtney
Airlines Electronic Engineering Committee
2551 Riva Road
Annapolis, Maryland 21401-7465

Mr Roy Courtney:

The following recommendations and comments from Rockwell Collins developed by Greg Sheffield and Nate Morrison are in response to ARINC Characteristic 664 Ad Hoc Reference Plane discussions.

100Base-TX Recommendations

After researching several 100Base-TX activities at Rockwell Collins, we have the following recommendations to provide.

- Link Budget

We recommend the following formula to define attenuation limits for all LRUs expecting to operate at 100Base-TX:

$$LRULinkAttenuation \leq 0.125 * \sqrt{f}$$

For $1 \leq f \leq 100\text{MHz}$

This formula is represented in the below limit table:

Frequency (Mhz)	Atten (dB)
1	.13
4	.25
10	.40
16	.50
20	.56
31.25	.70
62.5	.99
100	1.25

We recommend the following formula to define NEXT limits for all LRUs expecting to operate at 100Base-TX:

$$LRULink\_NEXT\_Limit = 47 - 15\log\left(\frac{f}{16}\right)$$

For $1 \leq f \leq 100\text{MHz}$

ATTACHMENT 2-4

The LRU's NEXT must be equal to or better than the limit. (Higher numbers represent lower crosstalk)

This formula is represented in the below limit table:

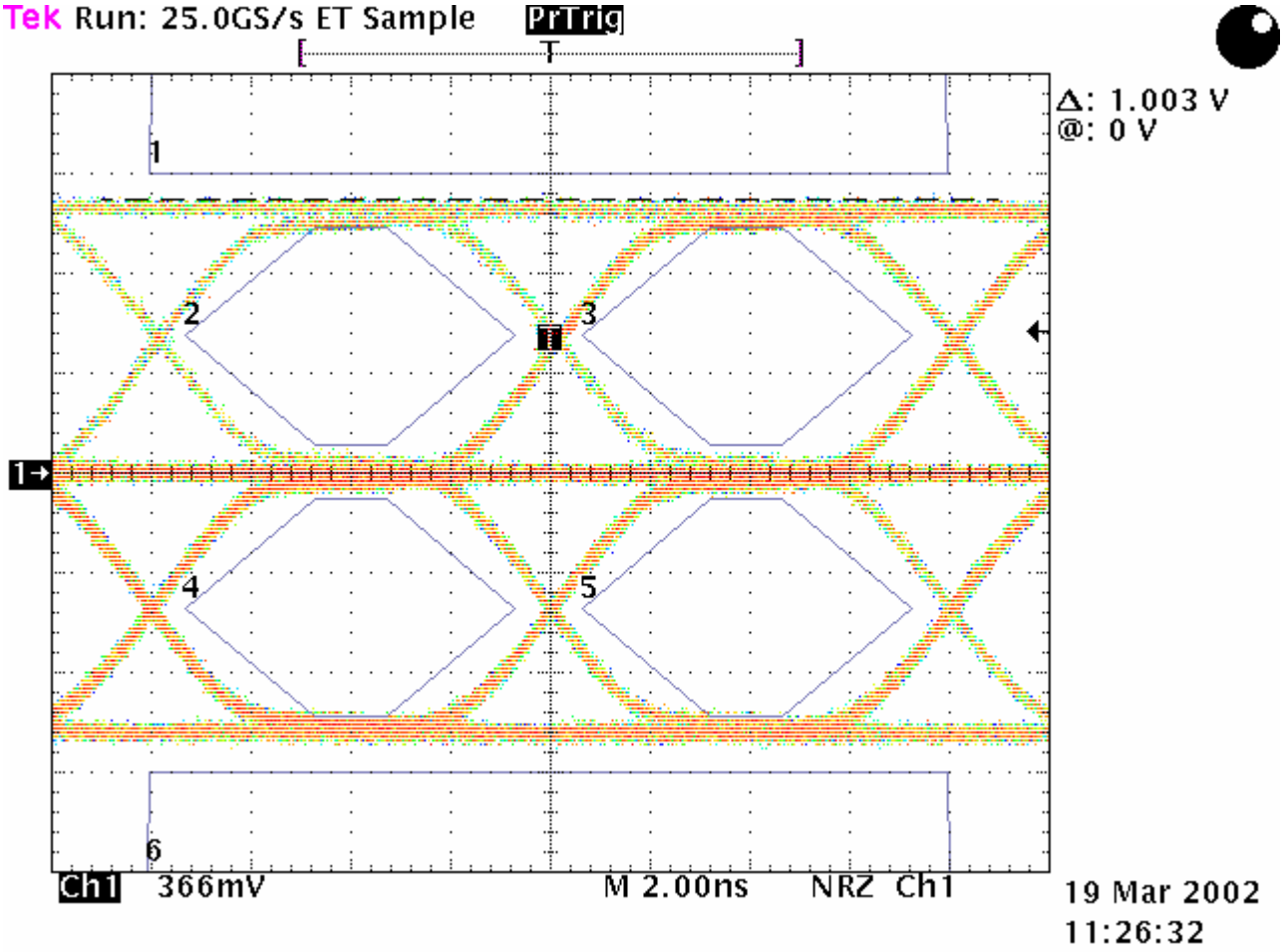
Frequency (Mhz)	NEXT (dB)
1	65
4	56
10	50
16	47
20	46
31.25	43
62.5	38
100	35

We recommend return loss also be part of this specification, but further study is needed on this topic.

ATTACHMENT 2-4

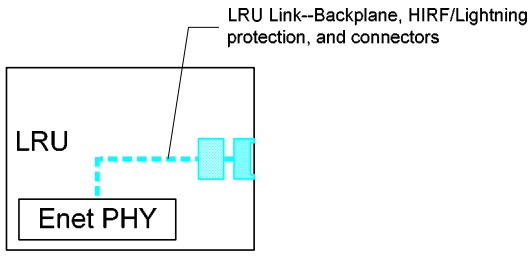
- Transmit Eye Pattern (100 Base-T Ethernet Mask)

The following is an average eye-pattern result for most LRUs that we have tested at Rockwell Collins.



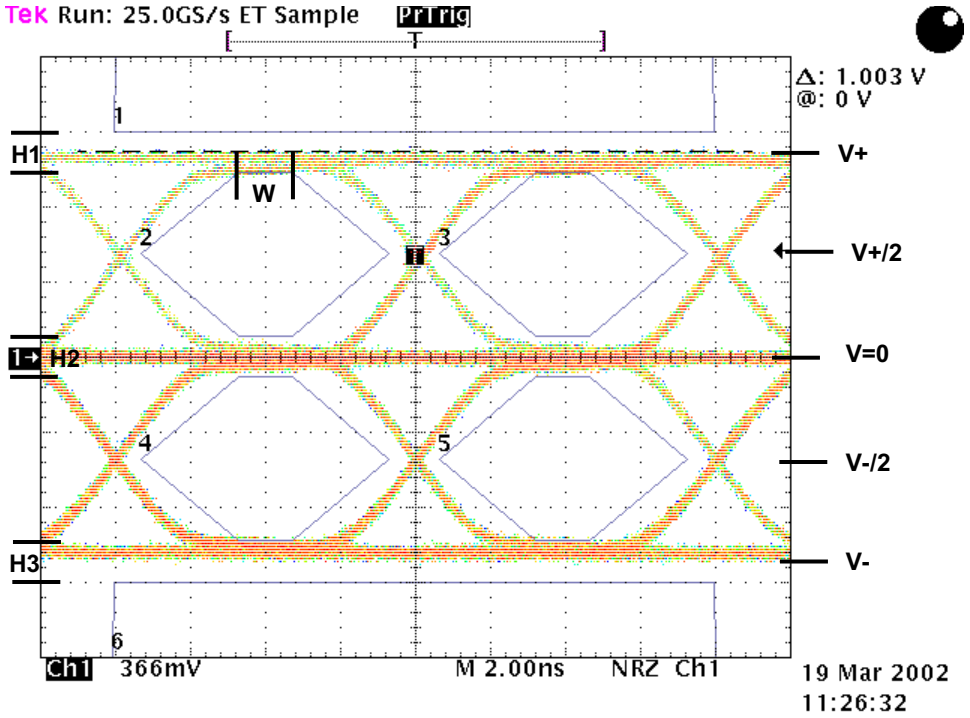
There are several factors that contribute to the above results failing. For example, the test point is not at the same point as defined in ANSI X3.263:1995, which is basically at the magnetic. Every LRU we looked at had at least one extra connector and several extra trace lengths than that expected in a commercial Ethernet solution.

The following illustrates the common Rockwell Collins LRU architecture supporting Ethernet:



ATTACHMENT 2-4

The following illustration shows the eye pattern key points to modify for the purpose of recommending a compliance eye pattern for Avionics LRUs.



Note:

ANSI X3.263:1995 Annex J has the following measurements for each key point in the above figure:

H1=H2=H3= V+/- 100mV
W= ~1.8ns
V+= .95 to 1.05V
V-= .95 to 1.05V

Rockwell Collins recommends changing each key point in the above figure to the following values:

H1=H2=H3= V+/- 100mV (same)
W= ~1ns
V+= .90 to 1.05V
V-= .90 to 1.05V

These changes need further study.

Using the combination of the eye pattern and dB levels we should get the following 100Base-TX network values at 16 MHz:

LRU#1------ 0.5dB ---- ---------- Cable Plant ---- 8.5dB ---------- ---- 0.5dB ----- LRU #2

Total = 9.5dB (leaving .5dB of margin)

- Receiver Sensitivity

We recommend that the receiver sensitivity be defined in the following manner:

An ANSI X3.263-compliant transmitter is connected to a 100-ohm balanced attenuator. The output of the attenuator is connected to the LRU at the reference plane. The LRU must maintain the nominal BER with this input configuration.

The attenuator value follows the formula:

$$Attenuator\_value = 2.25 * \sqrt{f}$$

For $1 \le f \le 100\text{MHz}$

The values of this attenuator are summarized in the following table:

Frequency (Mhz)	Atten (dB)
1	2.25
4	4.50
10	7.12
16	9.00
20	10.06
31.25	12.58
62.5	17.79
100	22.50

ATTACHMENT 2-4

Rockwell Collins

10Base-T Recommendations

At the moment, the only recommendations Rockwell Collins has are in regards to the overall link budget. We recommend the following for 10Base-T network values at 10 MHz.

LRU#1------ 0.4dB ---- ---------- Cable Plant ---- 10.2dB ---------- ---- 0.4dB ----- LRU #2

Total = 11dB (leaving .5dB of margin)

10/100Base-T Combination Comments

Please note if a network can use both 10Base-T and 100Base-TX, the LRU manufacturer and cable plant installer will have to consider both situations. This will likely mean designing the system to the 100BASE-TX limits, which the 10BASE-T limits are compatible with.

Summary

Rockwell Collins believes the above recommendations will make it possible for all manufacturers of LRUs to be able to develop compliant solutions.

Best Regards,

Greg Sheffield
Air Transport Systems Technical Director
Telephone #: 319-295-8575
Email: glsheffi@rockwellcollins.com

ATTACHMENT 3-1

PURPOSE OF THIS SUPPLEMENT

This supplement provides three additional appendices for ARINC Specification 664 Part 3 “Internet Based Protocols and Services”.

The first appendix “Appendix X – Aircraft Data Network Reference Model” provides shows the functional elements comprising the Aircraft Data Network. The function elements are abstract functions whose implementation may be in devices whose boundaries are not necessarily aligned with the boundaries of the functional elements.

The second appendix “Appendix Y –Air-Ground Communications Mobility for the Aircraft Data Network” provides information on implementing mobility in the communications network to allow aircraft applications to communicate with terrestrial applications even though the physical radio link changes while the aircraft travels.

The third appendix “Appendix Z – Communications Security for the Aircraft Data Network” addresses both the onboard network and it its offboard communications link to terrestrial (or other aircraft) systems. Issues of authentication, integrity and confidentiality (encryption) are covered.

ATTACHMENT 3-1

APPENDIX X
AIRCRAFT DATA NETWORK REFERENCE MODEL

Introduction

This appendix presents a Reference Model for the Aircraft Data Network. The Aircraft Data Network, as covered in this appendix, is the network and sub-networks onboard the aircraft including the connections offboard the aircraft (the radio links) that support the onboard networks. The domains of network users that are covered in this document are: Avionics, Crew, In Flight Entertainment (IFE), and Passenger Personal Electronic Devices (PEDs).

A Reference Model shows the functional elements and their interconnection. The functional elements represent a specific function or collection of functions and should not be taken to identify physical devices. Individual functional elements are shown even when multiple physical devices of the same function are required in the network; multiple functional elements are shown when necessary to present the interconnection scheme.

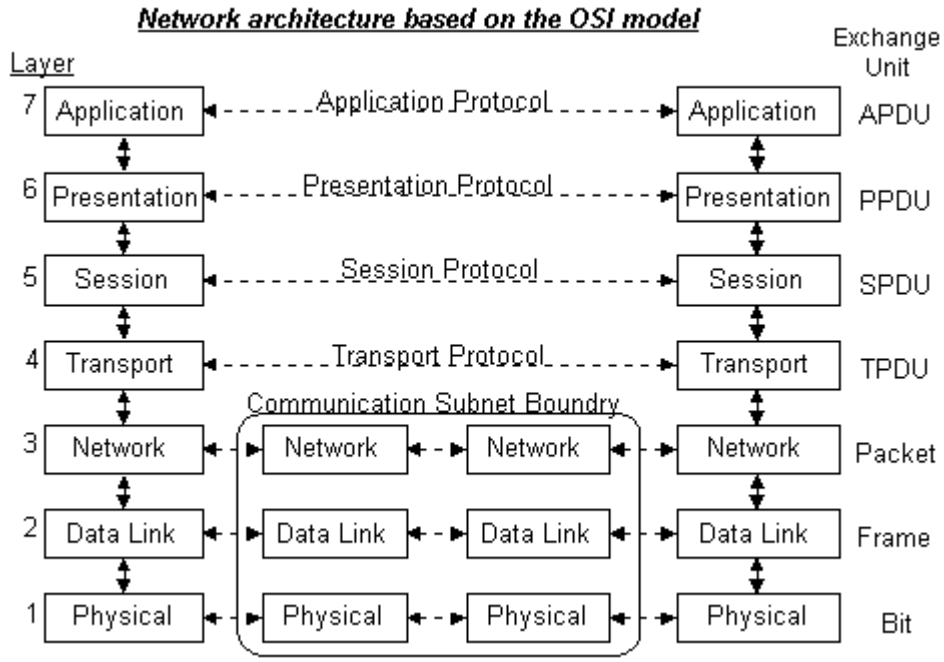
This document begins with a brief description of the OSI Reference Model followed by a table of the Functional Elements used in the Aircraft Data Network Reference Model. The Aircraft Data Network Reference Model is presented last.

ATTACHMENT 3-1

APPENDIX X
AIRCRAFT DATA NETWORK REFERENCE MODEL

OSI Reference Model

The seven layer OSI Reference Model is used in describing the Aircraft Data Network Reference Model functional elements.



The seven layers of the OSI Reference Model are defined as follows:
7) **Application** : Provides different services to the applications
6) **Presentation** : Converts the information (usually null in IP)
5) **Session** : Handles problems which are not communication issues (usually null in IP)
4) **Transport** : Provides end to end communication control (e.g. TCP or UDP)
3) **Network** : Routes the information in the network (e.g. IP)
2) **Data Link** : Provides error control between adjacent nodes (e.g. 802.2 or null)
1) **Physical** : Connects the entity to the transmission media (e.g. 802.3 Ethernet)

Note: In the above diagram TPDU/SPDU/SPDU/APDU means Transport, Session, Presentation or Application Protocol Data Unit. These are the units of transmission of data as viewed by the respective layer.

ATTACHMENT 3-1

APPENDIX X
AIRCRAFT DATA NETWORK REFERENCE MODEL

Functional Elements of the Aircraft Data Network
The following describe the functional elements used in the Aircraft Data Network Reference Model. Some elements are defined for future use and are not found in the current Reference Model.

Radio	The function of a Radio is to provide Physical and Link Layer transportation of frames on and off the aircraft.
Host	The function of a Host is to execute client and/or server applications.
Switch	The function of a Switch is to provide Link Layer interconnection of other networks or devices. All components that connect to a switch are of the same Link Layer technology (e.g. 802.3 Ethernet).
Bridge	The function of a Bridge is to provide Link Layer connectivity between networks or devices of the same of different Link Layer technology. Examples are an 802.3 (Ethernet)-to-802.5 (Token Ring) bridge and an 802.11 Access Point that connects to 802.3 (Ethernet).
VLAN Switch	The function of a VLAN Switch is to incorporate the functions of a Switch in addition to Virtual LAN technology. In one configuration a VLAN Switch will allow communications from all “client” ports through a single “trunk” port but will prevent communication between client ports.
Router	The function of Router is to provide communication between distinct subnets (A subnet is a group of contiguous IP addresses). The Router function also provides the ability to connect subnets of different Link Layer technologies (e.g. 802.3 Ethernet and Radio).
NAT Router	The function of a Network Address Translation (NAT) Router is to provide the function of Network Address Translation in addition to those of Router. The NAT Router function can translate IP addresses of packets as they traverse the Router. The NAT Router can also map multiple IP addresses from an interface to one or a few IP addresses on another interface by modifying and mapping the Transport Layer port numbers or another available protocol-specific index. This is sometimes referred to as a Network Port Address Translation (NPAT) Router.
Packet Filter Firewall	The function of a Packet Filter Firewall is to enforce network access based on protocol information in each IP packet. When an IP packet arrives at the firewall, the protocol information is compared to a collection of filtering rules. These rules specify the conditions under which packets should be passed through or denied access (discarded).
Session Filter Firewall	The function of a Session Filter Firewall is to incorporate all of the function of a Packet Filter Firewall and to determine and retain information on all active network sessions through it. This information is used to determine whether subsequent packets and packets flowing in the opposite direction belong to an approved connection. This function is sometimes referred to as a Smart Packet Filter or Stateful Packet Filter Firewall. By eliminating static configuration entries for returning packets security is enhanced. (The TCP protocol identifies the first and last packets of a session. For other protocols, such as UDP, which are not session oriented, the first packet may start a session and an elapsed time with no activity may be deemed to end the session.)

ATTACHMENT 3-1

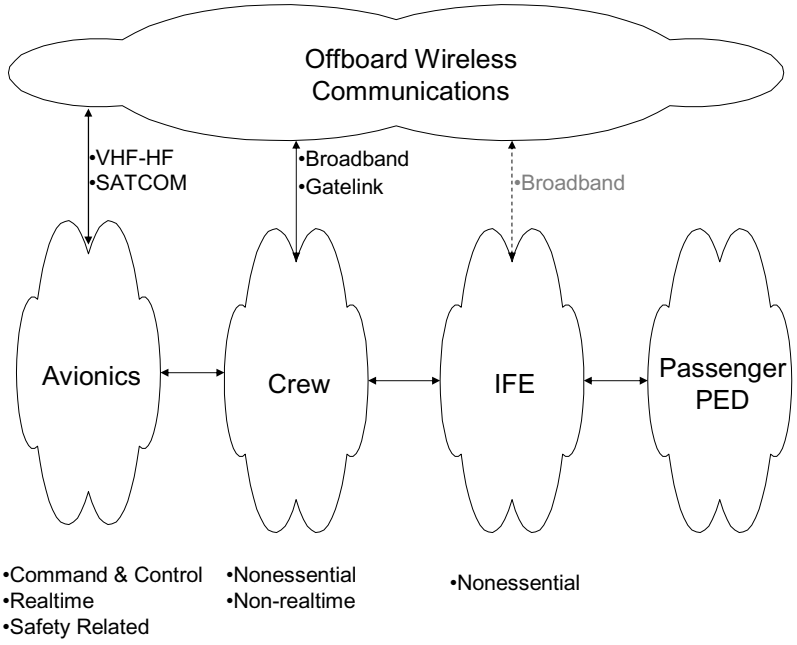
APPENDIX X
AIRCRAFT DATA NETWORK REFERENCE MODEL

Application Proxy Firewall	The function of an Application Proxy Firewall is to terminate all network connections and, if a requested access is authorized, create a separate connection to the desired destination. It then shuttles information between the original connection and the second connection. No IP packets are passed directly between the two networks. Proxy applications must be provided for each supported service.
Packet Filter Switch	The function of a Packet Filter Switch is to combine the functions of a Switch and a Packet Filter Firewall.
VPN Access Server	The function of a VPN Access Server is to allow IP packet-level connectivity to a firewall-protected or NAT-protected network from outside authenticated hosts. Typical implementations support PPTP and/or L2TP protocols. After authentication, the outside host is provided an IP address belonging to the protected subnet. The outside host creates IP packets using this address and securely tunnels the packets to the Remote Access Service which acts as the network attachment on its behalf. The secure tunnel provides packet integrity and optional confidentiality over an authenticated session.
Message Gateway Host	The function of a Message Gateway Host is to communicate with other onboard Hosts and provide Application Layer message exchanges with offboard Hosts. It may interface with Radios using only Link Layer protocols.
Airframe Host	The function of an Airframe Host is to execute crew applications outside of the avionics area that are provided by the airframe supplier.
Airline Host	The function of an Airline Host is to execute crew applications outside of the avionics that are provided by the airline or aircraft owner/operator.
Network Management Host	The function of the Network Management Host is to execute applications to support network fault management, configuration, accounting, performance monitoring and security.
Dynamic Address Assignment (DHCP)	The function of Dynamic Address Assignment (e.g. DHCP – Dynamic Host Configuration Protocol) is to dynamically assign IP address to Hosts, typically on the Passenger PED, IFE or Airline Host subnets.
Domain Name Server	The function of the Domain Name Server (DNS) is to provide a database of name to IP address mapping for other applications or network components.

ATTACHMENT 3-1

APPENDIX X
AIRCRAFT DATA NETWORK REFERENCE MODEL

Reference Model for the Aircraft Data Network
Domains of the Aircraft Data Network
The following diagram defines the domains for the Aircraft Data Network.



Avionics are the electronic and electromechanical subsystems and systems installed in an aircraft or attached to it. It does not include the power generation and distribution systems. Avionics includes the Command and Control functions of the aircraft. It is realtime in nature and safety related. Typical avionics applications include: Navigation and guidance, Communications, Surveillance, Flight Controls, Mission avionics (offensive and defensive), and Vehicle and utility management systems.

Crew applications are the nonessential functions supporting the flight deck and cabin crew. They can be subdivided into those applications provided by the airframe manufacturer and those applications provided by the airline.

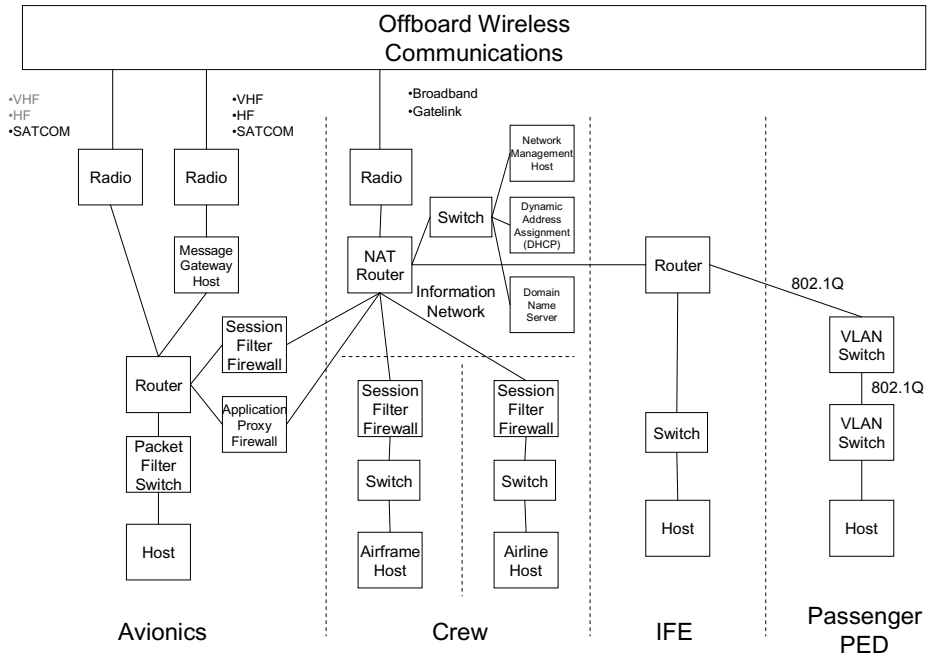
In Flight Entertainment (IFE) is the network supporting passenger entertainment. It also includes the Cabin Distribution System (CDS) used to connect Passenger Personal Electronic Devices (PEDs).

The Passenger Personal Electronic Devices (PEDs) is the collection of passenger provided electronic devices.

ATTACHMENT 3-1

APPENDIX X
AIRCRAFT DATA NETWORK REFERENCE MODEL

Reference Model Details
The following is the Aircraft Data Network Reference Model.



The Avionics network has connectivity offboard at the network Transport Layer (Radio to Router connection) and at the Application (Message) Layer (through the Message Gateway Host). Native IP applications will communicate using the Transport Layer. The Message facility supports existing ACARS messaging. The Avionics network is connected to Information Network through a Session Filter Firewall to support access for avionics applications to communicate through the offboard broadband or Gatelink connections. No inbound connections are allowed though this firewall. The Application Proxy Firewall allows access from non-avionics applications to restricted read-only functions on the Avionics network. The special Packet Filter Switch supports pre-defined communications among the components (hosts) on the Avionics network.

The Crew network is divided into three sections. The Airframer-provided hosts are protected by a firewall. The Airline-provided hosts are separately protected by a firewall. The Information Network provides the broadband and Gatelink connections offboard the aircraft. It also provides the internal support functions for Network Management, Dynamic Address Assignment and Domain Name Service. The Information Network requires no internal firewalls since its services are available to the other networks. The NAT router provides address translation (which is a form of firewall) offboard the aircraft. The router supports prioritization of transmissions for Quality of Service to the other networks.

The In Flight Entertainment network provides for passenger access to entertainment services. The Passenger PED network allows the attachment of Passenger Personal Electronic Devices. It is structured using VLAN Switches which are configured to disallow communications between the user ports, while allow traffic up through the router trunk. Passenger-to-passenger connections could be achieved using private VPN connections to a common corporate network.

ATTACHMENT 3-1

APPENDIX Y
AIR-GROUND COMMUNICATIONS MOBILITY FOR THE AIRCRAFT DATA NETWORK

Introduction

The focus of mobility in this appendix is Air-Ground Communications Mobility for ADN applications supporting Air Traffic Management, Airline Operations Centers and other information providers such as weather data. It is the ability to have these applications executing onboard an aircraft communicate with terrestrial applications (or applications on other aircraft) when the location of the aircraft is changing.

Types of Mobility

There are several kinds of Internet Mobility. Two kinds that may be familiar to the reader are not of interest for ADN applications. One example is the ability to have a portable computer connected to one network, then shut down that computer and restart it when connected to another network. This type of mobility is better described as Nomadicity. The mobile computer is reassigned an IP address for each network to which it is attached. Nomadicity can work for client-side mobility but for many applications appropriate for Air Traffic Management and Airline Operations, it is a requirement to be able to address the aircraft from the ground using a known IP address, and to have the communications constant and unbroken as the aircraft travels and changes phase of flight.

The other example of Internet Mobility that is not appropriate for ADN mobility is 802.11. The access point mobility designed into 802.11 is link layer-mobility. This mobility is invisible to the IP layer but depends on local link-layer connectivity. Link-layer mobility does not scale over a large geographic area.

The type of mobility needed for ADN applications is one that allows applications to execute without any disconnection of the underlying communications even when the aircraft is in flight. At the link layer the aircraft communications may be changing from one terrestrial radio link to another as the aircraft flies over ground or from one satellite link to another as the aircraft moves over a geographic area. The goal is that the aircraft applications retain the same IP address in an unbroken communications session with terrestrial or other applications. This model is not served by either Nomadicity or link-layer mobility solutions. The IETF standard that has been developed to meet these requirements is Mobile IP. Mobile IP is an optional part of IPv4 and is a standard part of IPv6 implementations.

Characteristics of Mobile IP

Some of the characteristics of Mobile IP are:

- IETF standards-based mechanism to deliver packets to nodes that change their location
- Operates exclusively at the Network Layer (Layer 3)
- Mobile nodes can communicate with other nodes that do not implement mobility functions
- Built-in support for mobile networks and mobile routers
- Administrative traffic is kept to a minimum
- Architecture is scalable, robust and secure

Mobile IPv4 was approved in 1996 and documented in RFCs 2002 through 2006.
IPv6 with mobility was approved in 1996.

How Mobile IP Works

Mobile IP operates by having an entity in the fixed (terrestrial) network having a know IP address for each network or computer that is mobile (in an aircraft). The entity with the fixed, statically located IP address is known as the Home Agent. Home Agents would usually reside in one or more terrestrial network routers. These routers are statically configured to function as Home Agents.

Mobile Nodes are computers - or routers supporting mobile networks - that can change their point of attachment (their link layer connections) to the network over time. For ADN, Mobile Nodes are computers or routers onboard aircraft. These computers or routers have the optional IPv4 mobility software or incorporate the standard IPv6 mobility function.

Mobile Nodes monitor the underlying link-layer connections and reacquire dynamic IP addresses based on their current link-layer connections as necessary. They register or re-register their current dynamic IP address with their Home Agent. This re-registration occurs in the protocol stack such that local computer applications, or in the case of mobile routers, other computers on the mobile network do not see the change of link-layer connections and Mobile IP re-registrations. They see a constant network connection.

APPENDIX Y
AIR-GROUND COMMUNICATIONS MOBILITY FOR THE AIRCRAFT DATA NETWORK

Other computers on the network, such as Air Traffic Management or Airline Operation computers that communicate with the Mobile Nodes do so by communicating with their respective Home Agents. The actual existence of Mobile Nodes and the underlying mobility mechanisms is transparent to them. They are able to always communicate in an unbroken session with their representative at a static location and address – the Home Agent.

The Home Agents always know the actual IP address of the Mobile Node even though the actual IP address of the Mobile Node changes over time. The Home Agents tunnel IP packets they receive to their corresponding Mobile Nodes using the currently registered actual IP address. Depending on how the mobility is configured, Mobile Nodes transmit IP packets using either their current network connection or a reverse tunnel back to the Home Agent. The reverse tunnel introduces some overhead but allows security mechanisms to see the correct source IP address of packets originating from the Home Agents that correspond to the Mobile Nodes.

Mobile IP and ADN

The Aircraft Data Network is not an island of connectivity onboard an aircraft but is part of a larger network supporting Air Traffic Management, Airline Operations Centers and other information providers such as weather data. The ADN is connected to a terrestrial network using air-to-ground radio, satellite or other links that change over time and phase of flight. Mobile IP – available in IPv4 and in IPv6 - with or without Mobile Routers onboard the aircraft -provides a standards-based solution that is independent of the actual link-layer connection technology. Mobile IP is transparent to applications, and is efficient, robust, secure and scalable to a global solution.

Mobile IP as defined in the IPv4 and IPv6 IETF standards and implemented in COTS products can be applied to the ADN without any special adaptation or consideration.

ATTACHMENT 3-1

APPENDIX Z
COMMUNICATIONS SECURITY FOR THE AIRCRAFT DATA NETWORK

Introduction

The Aircraft Data Network must provide secure communications between onboard applications residing in avionics and other systems as well as in offboard links to terrestrial or other applications.

Security can be provided by cryptographic and non-cryptographic mechanisms.

Cryptographic security

Cryptographic security provides mutual authentication of the identity of both ends of a communication session. Authentication relies on a Public Key Infrastructure to issue public key security certificates (signed binary strings) that bind the identity of one end of a session to a public key for which the matching private key is securely held by that end of the session. The entities that issue public key certificates are called (Security) Certification Authorities.

With authentication in place, a shared symmetric session key is usually generated dynamically and known only to both ends of the authenticated session. This session key enables integrity of message exchanges. Messages of the session are secured against modification while in transit. Optionally, the session key can be used to encrypt the messages to provide confidentiality.

The mutual authentication of both ends of a session and the messaging integrity and confidentiality bound to that authentication are frequently referred to as a Security Association (SA). Security Associations can be created at one or more levels of the OSI Reference Model. One or more of Link Layer, Network Layer and Application Layer Security Associations are frequently found in complex communication systems.

Onboard Network Security

Onboard network security is implemented by non-cryptographic mechanisms including Packet Filter Switches, NAT Routers, Session Filter Firewalls and Application Proxy Firewalls. These functions and a model for their interconnection are defined in Appendix I – Aircraft Data Network Reference Model.

Non-cryptographic functions are suitable for the onboard aircraft since the avionics and crew networks are physically protected and all communications outside these networks are through firewalls. The avionics network itself is protected internally by the packet filter switch that interconnects all components.

Offboard Communications Security

The remainder of this appendix focuses on offboard communications security.

The Aircraft Data Network and its offboard communications system is referred to as the ADN System or ADNS. Security for the ADNS relies on COTS products implementing network standards where appropriate and ADNS-specific mechanisms where considerations such as limited bandwidth require a non standards-based approach. Security is provided at two levels – the network/link level and the application level. Security mechanisms at both levels employ strong, standards-based cryptographic technologies. In addition, non-cryptographic technologies such as firewalls and packet filtering routers provide additional network layer isolation and protection from attack, including denial-of-service.

Network/Link Level Security

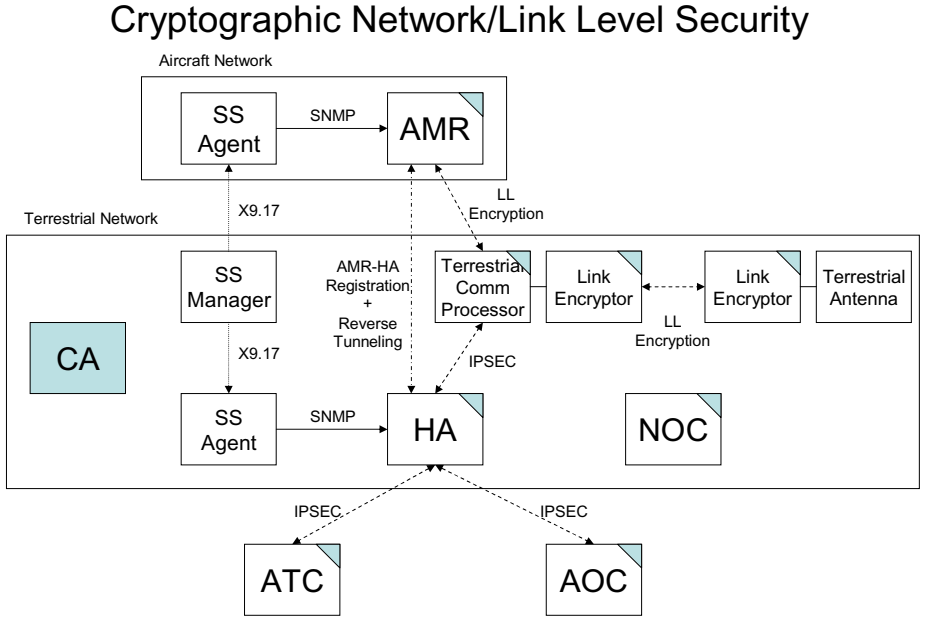
The Network/Link level security is achieved by both cryptographic and non-cryptographic mechanisms. They are described in separate sections below.

ATTACHMENT 3-1

APPENDIX Z
COMMUNICATIONS SECURITY FOR THE AIRCRAFT DATA NETWORK

Cryptographic Network/Link Level Security

The reference model for the ADNS cryptographic network/link level security is show below.



The AMR is an Airborne Mobile Router. It implements IP v4 or v6 mobility for the entire airborne network. HA is the Home Agent for the AMR. The NOC is Network Operations Center.

One or more (Security) Certification Authorities (CA) are used to implement a Public Key Infrastructure for the Cryptographic Network/Link Level Security. The CA's may be structured in a hierarchical arrangement that parallels the international organization of aeronautical administrative authorities. The triangle in the upper right corner of some network entities shown in the diagram represents the private key and corresponding public key certificate signed by the Certification Authority with administrative authority of the network entity.

Network Layer Security

Mobile IP requires that a Shared Secret (SS) be maintained between the Air Mobile Router (AMR) and the corresponding Home Agent (HA) in order to authenticate the Mobile IP registration action. A Shared Secret Manager uses a Shared Secret Agent local to the AMR and HA in order to set the Shared Secret. The X9.17 standard for Shared Secret management is used. Secure SNMP (V3) may be utilized instead of local SS Agents if the SNMP infrastructure is in place.

IETF Standards-based IPSEC security is maintained between the Airline Operational Control (AOC)/Air Traffic Control (ATC) and the HAs and between the HAs and the Terrestrial Communications Processor. The public key certificates are utilized for

authentication. A private session key is derived for integrity and confidentiality.

Reverse tunneling between the AMR and the HA in Mobile IP is used to ensure security.

The Network Operations Center (NOC) can communicate with other components using IPSEC. These security associations are not shown. Other entities can be certified by the CA, if necessary for management. SNMP V3 with operational security mechanisms is used throughout the network for management.

Link Layer Security

A Link Layer security mechanism that is proprietary to the Air-Ground Radio link is used between the AMR and the Terrestrial Communications Processor. A non-standard security implementation is used in order to accommodate message header and data compression and best utilize the limited radio-link bandwidth. The mechanisms and algorithms parallel the IPSEC mechanism and provide equivalent security.

Link Layer security may additionally enhanced by the use of a coding scheme such as CDMA, CRC and frame numbering within a finite state machine at the air-link (radio) layer.

Non-cryptographic Network/Link Level Security

Firewalls and packet filtering routers are used at points in the network to provide an additional level of security. These technologies are well suited to prevent intrusion attacks, including denial-of-service.

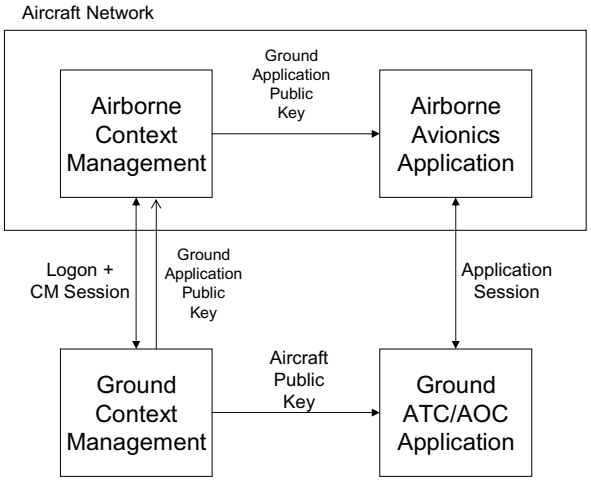
ATTACHMENT 3-1

APPENDIX Z
COMMUNICATIONS SECURITY FOR THE AIRCRAFT DATA NETWORK

Application Level Security

The ADNS uses the upper-level protocols of the Aeronautical Telecommunications Network (ATN) or similar application protocols. The ADNS supports the ATN application-level security mechanism for authentication, integrity and confidentiality (when confidentiality is incorporated into the ATN standards). The ATN upper-level protocols support both Air Traffic Services Communications (ATSC) and non-ATSC applications, including AOC applications. These applications communicate using a two-part operation, involving a support Context Management (CM) application.

Application Level Security



An aircraft authenticates to the CM application with its current flight ID. Ground applications query the CM with a flight ID to obtain the current ATN network address. Ground applications establish a session with applications in the avionics using the ATN network address.

The ATN applications on the ground and in the avionics authenticate each other, authenticate messages that are exchanged and ensure the integrity of the exchanged messages.

ATN security (authentication and integrity) is based on strong elliptic-curve cryptographic mechanisms utilizing public key certificates and message authentication codes.

ATTACHMENT 4-1

PURPOSE OF THIS SUPPLEMENT

This supplement provides two additional appendices for ARINC Specification 664 Part 4 “Internet Based Address Structures and Assigned Numbers”.

The first appendix “Appendix G - Example for Multiple Network Devices” provides information on network devices that must adapt their MAC, IP and multicast addresses depending on the aircraft configuration in which they are installed. They can also adapt to variability of the MAC, IP and multicast addresses of other devices with which they communicate in a particular aircraft configuration.

The second appendix “Appendix H - Internet Protocol Version 6” provides additional information for the IPv6 protocol implementation in Aircraft Data Networks. IPv6 does not replace the current ADN Internet Protocol Version 4 (IPv4) specification, but allows for the coexistence of IPv4 and IPv6.

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

This outline for this appendix follows:

Introduction
- Determining if a device should be considered Multiple Network
- ARINC 664 Compliant Addressed Networks
- Guidance to Managing Multiple Network Device Addressing
  - ARINC 429 Analogous to ARINC 664
  - Basic Multiple Network Device Expectations
  - Gaining Addressing Knowledge When Communicating with Other Devices
    - Multiple Network Device To Multiple Network Device Communication
    - Multiple Network Device To System Integrator Controlled Device Communication
      - Flow Three - from the Multiple Network Device to the System Integrator Controlled Device
      - Flow Four - from the System Integrator Controlled Device to the Multiple Network Device
    - Multiple Network Device to / from Data Loader Communication
  - Mapping Position Strapping to Address Id Values
    - Source MAC Interface Id
    - Source IP Id
    - Destination Multicast Id

Multiple Network Devices Data Flows
- Configuring LAN forwarding to support multiple network device flows
- Default flows in multiple network devices
- Multiple Network Device Flow Parameters
  - Multiple Network Device Flow Allocated Bandwidth and Frame Size
  - Multiple Network Device Flow Frame Contents Format Descriptions
  - Multiple Network Device Receiving Multicast Group Flows

Aircraft Example of Addressing for Multiple Network Devices
- Multiple Network Device X
- Multiple Network Device Y
- Multiple Network Device Z
- Aircraft Logical Topology Description
- ARINC Table Descriptions
- ARINC Hardware Specification Document Flow Addressing Tables
- ARINC 664 Multiple Network Device Addressing Id Allocation Tables

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Introduction

This appendix was created to deal with the issue of addressing for sub-networks that include Multiple Network Devices. The appendix starts with an introduction to what a Multiple Network Device is, and why ARINC 664 should deal with this issue. There is an explanation of the basic characteristics of a Multiple Network Device (as they are relevant to ARINC 664). The recommended ARINC 664 IP and MAC addressing formats are described. These are incorporated into ARINC 664 Part 4. Following is included a section that describes how a Multiple Network Device can determine its addressing when there is more than one instance of the same Multiple Network Device type in the aircraft.

To aid the reader in understanding, an example of an aircraft topology is included. This example allows the reader to follow how each of the Multiple Network Devices in the topology determine the IP and MAC source addressing used, as well as the destination multicast values they will receive. The example topology network also shows how tables will be created. These tables include Multiple Network Device allocation tables in ARINC 664, and address determination tables in ARINC hardware specification documents.

When an LRU participates on an Ethernet LAN it must use specific IP and MAC addressing values. These values are planned by the system integrator who is coordinating the logical LAN layout of an aircraft. In an avionics environment it is sometimes unacceptable for one LRU to 'learn' the addressing of another LRU. Instead, often avionics devices (or the system integrator) must be able to know prior to power up what the addressing used by other devices is.

When working with resident devices that the system integrator purchases specifically for the integration process, addressing determination is a straightforward process. Either the system integrator can drive the address determination and construction method by requirements to the vendors, or simply be aware of the approach taken by the vendor of the switches and end-systems selected. In this way the system integrator is fully aware of the address values that will be used by each switch and end-system of the LAN being integrated. These resident devices are designed specifically with a certain address construction approach in mind. Resident devices can not be freely moved from one aircraft to another. The pin-out definition may differ, and the addressing rules may be different. Consider that one aircraft may require addressing that accommodates one hundred end-systems, while another aircraft may require only a few end-systems. There may be valid reasons why each system integrator for these aircraft selects different addressing approaches.

When inserting Multiple Network Devices into an avionics LAN, the issue of address knowledge and uniqueness exists. The system integrator may wish to incorporate a Multiple Network Device into the LAN being integrated, but must insure that the addressing values used by the Multiple Network Devices are known and are unique within the LAN they are being inserted into. The system integrator will be unable to control the address determination / construction approach used by a Multiple Network Device. The system integrator will have to understand the approach taken by the Multiple Network Device and consider this in the integration of the aircraft. For example, a HF Data Radio (HFDR) will be designed by Collins to have a particular IP and MAC address construction approach. This approach will be designed independent of when system integrators design their addressing approaches.

What is critical to the system integrator is that the addresses of Multiple Network Devices, both MAC and IP, be unique with respect to those used by the resident devices and other Multiple Network Devices in the network. For example, the system integrator needs to be able to plug a HFDR into the LAN and know that the addressing used will not conflict with another resident device in the network.

Consider that different system integrators may want to make different choices about resident device addressing, but use common Multiple Network Device components. For example, perhaps an airline wants to use both Airbus and Boeing aircraft of different types each having unique addressing approaches, but wishes to use a Collins HFDR in all of them. This requires an ARINC 735 HFDR compliant radio to be able to operate in any of these aircraft.

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Determining if a device should be considered Multiple Network

When determining if a device should be considered Multiple Network the key question is:

Can the system integrator control the addressing used by the device?

Let us consider an aircraft engine. This device could be considered Multiple Network or resident.

Suppose a particular aircraft engine manufacture requires that an airline data load the address values used by its engine prior to installation on an aircraft. The system integrator for the aircraft can then decide what Ethernet address values should be. Therefore, this engine should be thought of as a resident device. There would be no need for ARINC to control the address values for this device.

Suppose another aircraft engine manufacture wants to insure that an engine can be installed on an aircraft, and no data load of addressing information is required. This minimizes the effort of engine installation. Notice that an airline can have spare engines of this kind in storage, and swap them on an aircraft, requiring no address information data load.

So it is important to observe that any device could be considered resident or Multiple Network. In the case of engines the decision could reasonably go either way. Considering the effort in installing an engine, an airline may not be concerned with the additional effort in data loading. Alternatively, an airline may reason that this is just one less issue to deal with. For aircraft engines the choice may not be so clear, but for smaller, more frequently swapped devices, such as HF Data Radio, almost certainly there would be an advantage in not having to data load them each time they are inserted into an aircraft.

ARINC 664 Compliant Addressed Networks

A typical aircraft integration may be composed of several Ethernet sub-networks. In any sub-network that incorporates a Multiple Network Device using ARINC administered addressing, the sub-network should follow the ARINC 664 guidelines for address construction. This will insure that the addresses used are unique for both Multiple Network and resident devices on the same sub-network. Each system integrator is responsible for insuring uniqueness of addressing of devices in the aircraft. When two sub-networks are connected, either the addressing used must be unique across both, or the connecting device must bridge the two address domains. The resolution of this is left to the system integrator. ARINC 664 address construction approach described herein is intended to support unique IP address values for the entire aircraft.

When a system integrator chooses to have multiple sub-networks in the aircraft, these sub-networks must be ‘appropriately’ separated from one another by isolating devices in order to insure address uniqueness in the system. Depending on the addressing choices made, this may include a switch, a Router, or an ‘IP Level Translator’.

ARINC 664 expects that these sub-networks are interconnected by functional IP Routers. They may require high integrity fault isolation in some cases.

Guidance to Managing Multiple Network Device Addressing

ARINC 429 Analogous to ARINC 664

ARINC 664 is taking on the role supported by ARINC 429 in regards to Multiple Network Devices. Consider a HF Data Radio. ARINC 429 describes the physical media to be used, allocates label numbers, describes L/R/C resolution, and calls out the basic structure of a label. Each appropriate ARINC hardware specification describes the data content format of each label, as well as the particular port-connector pins that will be used to transmit and receive 429 data. In the case of HFDR, ARINC 735 calls out the 429 pin definition for transmit and receive lines. A system integrator reads the ARINC 735 specification and wires the aircraft to accept an HFDR. This allows the HFDR to transmit and receive

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

data to / from other LRUs in the aircraft. When the HFDR transmits a 429 label it indicates which instance of an HFDR it is by indicating the SDI bits in the label. When a 429 label is received to a LRU the source of that data is determined by 1) the physical wire it was received on, and 2) the SDI value in the label. The type of label determines the type of data, and the contents are parsed accordingly.

Notice that 'addressing' is resolved in ARINC 429 through a combination of SDI bits and physical wire. Transmit 'bandwidth allocation' is described in the appropriate hardware specification by describing the Hz rate of each label. The logical topology is implemented by running separate physical wires for each 'connection' in the aircraft. If the transmission is a 'multicast' then there are multiple receivers to a ARINC 429 line.

This ARINC 429 approach has worked well for the airlines. It allows ARINC 429 to allocate label values and the appropriate ARINC Hardware specification describes the label format and rate at which the label is set. The drawback to the ARINC 429 approach is that it makes no attempt to allow more than one 'logical-flow' one a single physical connection. This requires a significant amount of wiring in an aircraft, and requires that any logical changes to the topology be made in the form of physical wiring.

The airlines are interested in an approach that simplifies wiring and makes changes to logical topology without making physical changes.

There are some subtle differences between the ARINC 429 approach and the ARINC 664 Ethernet based approach. In ARINC 429 each LRU was independently addressable. When a 429 label is received the consumer knows only the LRU type that produced it, and the instance of that type. In the ARINC 664 Ethernet approach it is possible for each LRU to have multiple IP addressable entities, and to have more than one physical Ethernet port each having a unique MAC address. When an Ethernet frame is received to an LRU the transmitting Ethernet physical port is identified by the source MAC address value, and the source IP addressable entity is identified by the source IP value. Therefore address resolution of logical connections used to be resolved by noting the physical wire data was received on, and is now resolved by looking at the IP and MAC addressing.

This change to logical based addressing requires that ARINC 664 manage address resolution values for Multiple Network Devices just as ARINC 429 now describes the SDI values, and states the use of physical connections.

Basic Multiple Network Device Expectations

Each Multiple Network Device will have one or more physical ports on the LAN. Each of these ports will require a source MAC address.

Each Multiple Network Device will have one or more IP addressable agents in its default configuration. These agents exist to support interoperation with portions of the LAN. For example, each Multiple Network Device will have one or more Data Load agents. These are uniquely IP addressable. A Multiple Network Device may also have maintenance agents. Central maintenance LRUs may communicate with these agents to collect status information.

Multiple Network Devices may communicate using unicast or multicast connections.

Communications to and from Multiple Network Devices may be with other Multiple Network Devices or system integrator specified devices. For example, HFDR might be specified to communicate with a CMU (another Multiple Network Device), and with a CMC application running as an application on a system integrator specified device. The ARINC 735 hardware specification for HFDR will describe the address construction used by the HFDR and the addressing expected from the devices it communicates with.

Each Multiple Network Device will transmit data on several IP multicast flows. On each transmit IP-flow the following need to be described: Payload Id, Payload format of data contents, Payload rate, maximum frame size, and the IP-flow each payload is to be transmitted on.

ATTACHMENT 4-1

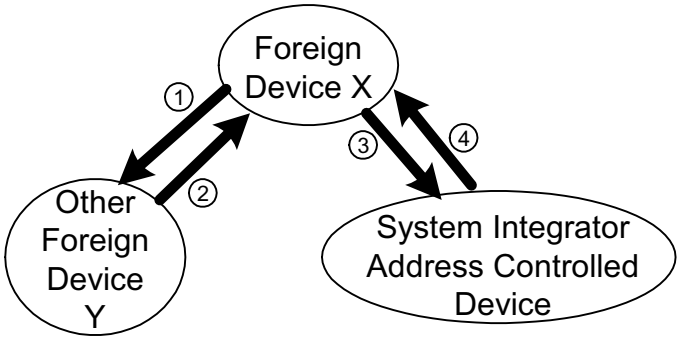
APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Each Multiple Network Device will receive data on several IP multicast flows. On each receive IP-flow the following need to be described: Payload Id, Payload format of data, payload rate, maximum frame size, and which IP-flows(s) the payload should be received on.

Gaining Addressing Knowledge When Communicating with Other Devices

There is a need in a closed avionics LAN for devices and the system integrator to be able to know the addressing choices used prior to powering up the system. This supports both integration and allows impersonation protection.

Multiple Network Devices will be planned to communicate with both other Multiple Network Devices and system integrator planned IP addressable agents. This section describes how those address values used will be made known to the Multiple Network Device and to the system integrator.



The concept of address definition can best be described through an example. The diagram shows a Multiple Network Device X which has four logical data flows. Two transmit flows, and two receive flows. Flows one and two are to / from Multiple Network Device Y. Flows three and four are to / from an IP addressable entity running on an application in a device whose addressing is system integrator controlled.

Multiple Network Device To Multiple Network Device Communication

The source addressing for flows one and two will be specified in each ARINC hardware specification for device types X and Y. Therefore the source addressing can be determined by simply reading ARINC hardware specification for X and for Y.

The destination addressing can be either unicast based or multicast based. Either approach will work fine.

When unicast based, the designers of each device (X and Y) will need to read what address values of the other device is by reading the ARINC hardware specification for that other device.

When multicast based the designers of each device (X and Y) will need to read what multicast address value should be expected to be used when a frame is received.

Either way, this will require the two ARINC committees to cooperate when establishing the addressing. This should be a simple, non-contentious, discussion simply agreeing on the values to use, and having them allocated in ARINC 664, and their usage specified in each of the hardware specifications for X and for Y.

Multiple Network Device To System Integrator Controlled Device Communication

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Flow Three - from the Multiple Network Device to the System Integrator Controlled Device

Flow number three flows from the Multiple Network Device to the system integrator controlled device.

The source IP / MAC addressing is straightforward. This will simply be as per the ARINC 664 construction approach using the field id values specified in the ARINC hardware specification for Multiple Network Device type X.

The destination addressing is a handled differently depending of whether unicast or multicast destination addressing is used.

If multicast addressing is used then the address specification is straightforward. This will simply be as per the ARINC 664 construction approach using the field id values specified in the ARINC hardware specification for Multiple Network Device type X.

If unicast addressing is used then there is no way for the Multiple Network Device X to know what the destination address values are until device type X is data loaded with them.

Therefore, when a Multiple Network Device is planned to communicate with a system integrator controlled device, either multicast addressing should be specified in the Multiple Network Device, or unicast addressing will be used and the Multiple Network Device will need to be data loaded with the addressing information. If the device must be data loaded with addressing information then it may represent a unique part number for each installation. ARINC has not defined a load set that contains addressing information that is device independent.

Flow Four - from the System Integrator Controlled Device to the Multiple Network Device

Flow number four flows from the system integrator controlled device to the Multiple Network Device.

Until the Multiple Network Device is data loaded with addressing information, there is no way for the Multiple Network Device to know the source IP / MAC address values of the system integrator controlled device. There are two ways to deal with this.

One, the Multiple Network Device could be data loaded prior to using this flow.

Two, the Multiple Network Device could be specified to accept data from any source MAC / IP address. This eliminates the possibility of allowing the Multiple Network Device to participate in impersonation protection.

Multiple Network Device to / from Data Loader Communication

There may be cases where the address values of devices communicating with a Multiple Network Device are not known at the time the Multiple Network Device is specified. This is the case for communication between a Multiple Network Device and a data loader.

An ARINC Multiple Network Device hardware specification should establish a receive socket connection that receives data to IP-Port <Data Load> from any IP / MAC source address. This will allow communication from the data loader to the Multiple Network Device if data load is needed.

An ARINC Multiple Network Device hardware specification should require each Multiple Network Device to support source IP learning upon receipt of a SNIP request. This allows a Multiple Network Device to transmit frames back to a data loader using the IP address of the data loader.

An ARINC Multiple Network Device hardware specification should require each Multiple Network Device to respond to an ARP request, or generate an ARP request if needed. When an ARP request is received to a Multiple Network Device the MAC address value should be updated in its ARP cache. These approaches allow the data loader MAC address to be known to the Multiple Network Device.

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Mapping Position Strapping to Address Id Values

In some cases the addressing used by a Multiple Network Device will depend on the position strapping. This section describes how that determination is specified.

Source MAC Interface Id

Each Multiple Network Device LRU type will have one or more 'Source MAC Interface Id' values allocated to it by ARINC 664. More than one value may often be needed when there is more than one instance of the same type of Multiple Network Device in the aircraft, or when a given Multiple Network Device is designed to use more than one Ethernet port connected to the LAN. For example, there may be multiple HF Data Radios in an aircraft each of which want to transmit the same data to the CMU. The means by which a Multiple Network Device LRU determines which of the id values to use is left to the relevant ARINC specification for that LRU type. For example, ARINC 735 specifies HF Data Radio hardware characteristics. Suppose that ARINC 664 allocated HF Data Radio Source MAC Interface Id values of 25, 26, and 27. The following table may be specified in the ARINC specification for HFDR to determine what Source MAC Interface Id value should be used to construct source MAC address values.

Strapping Pin Bit Value	Source MAC Interface Id value
00	25
01	26
10	27
11 – not applicable	N/A

The meaning of the strapping bits to the Multiple Network Device is left to the system integrator. For example one system may have side / rack / and slot, while another has only side. Each system integrator will use the hardware specification to determine how to set the strapping on the Multiple Network Device. By knowing how the strapping is set, and reading the ARINC hardware specification, the system integrator can determine what source MAC address value will be used by a given Multiple Network Device.

Source IP Id

Each Multiple Network Device LRU type will have one or more 'Source IP Id' values allocated to it by ARINC. More than one value may often be needed when there is 1) more than one instance of the same type of Multiple Network Device in the aircraft, and 2) more than one IP addressable entity in a given Multiple Network Device. For example, there may be multiple HF Data Radios in an aircraft each of which have three IP addressable entities. The means by which a Multiple Network Device LRU determines which of the id values to use is left to the relevant ARINC specification for that LRU type. For example, ARINC 735 specifies HF Data Radio hardware characteristics. Suppose that ARINC 664 allocated HF Data Radio Source IP Id values of 37-39, 45-47, and 51-53. The following table may be specified in the ARINC specification for HFDR to determine what Source MAC Interface Id value should be used to construct source MAC address values.

Strapping Pin Bit Value	IP Addressable Entity	Source IP Id
00	1	39
00	2	40
00	3	41
01	1	45
01	2	46
01	3	47
10	1	51
10	2	52
10	3	53
11 – not applicable	N/A	N/A

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

The meaning of the strapping bits to the Multiple Network Device is left to the system integrator. For example one system may have side / rack / and slot, while another has only side. Each system integrator will use the hardware specification to determine how to set the strapping on the Multiple Network Device. By knowing how the strapping is set, and reading the ARINC hardware specification, the system integrator can determine what source IP address value will be used by a given Multiple Network Device.

Destination Multicast Id

Each Multiple Network Device LRU type will have one or more 'Destination Multicast Id' values allocated to it by ARINC. More than one value may often be needed when there is 1) more than one instance of the same type of Multiple Network Device in the aircraft, and 2) more than one multicast-IP flow is received / transmitted to / from each instance in the aircraft. For example, there may be multiple HF Data Radios in an aircraft each of which transmit three multicast flows, and receive two multicast flows. The means by which a Multiple Network Device LRU determines which of the id values to use is left to the relevant ARINC specification for that LRU type. For example, ARINC 735 specifies HF Data Radio hardware characteristics. Suppose that ARINC 664 allocated HF Data Radio Destination Multicast Id values of 2057-2059, 4044, and 1111-1113. The following table may be specified in the ARINC specification for HFDR to determine what Destination Multicast Id value should be used to construct destination MAC and IP address values.

LRU Flow Id	Strapping Bit Value	Transmit / Receive	ARINC Destination Multicast Id
1	00	Transmit	2057
	01		2058
	10		2059
2	00	Transmit	4044
	01		4044
	10		4044
3	00	Transmit	1111
	01		1112
	10		1113
4	xx	Receive	3001
5	00	Receive	25
	01		26
	10		27

The meaning of the strapping bits to the Multiple Network Device is left to the system integrator. For example one system may have side / rack / and slot, while another has only side. Each system integrator will use the hardware specification to determine how to set the strapping on the Multiple Network Device. By knowing how the strapping is set, and reading the ARINC hardware specification, the system integrator can determine what Destination Multicast address value will be used by a given Multiple Network Device.

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

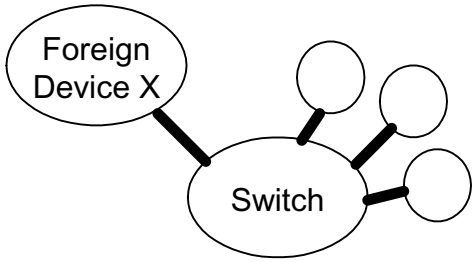
Multiple Network Devices Data Flows

There will be external devices connected to some avionics networks, which use a fixed set of transmit / receive data flows. For example, HFDR expects to transmit and receive a fixed set of logical data flows. The list of data flows are expressed on ARINC 429 lines, and are fixed. The functional behavior of the HFDR is designed, and this known behavior can be counted on when the radio is inserted into the aircraft. The burden is on the system integrator to insure that the connections are made available so that communication to / from the radio is provided in the aircraft as indicated by the radio ARINC hardware specification.

The system integrator will need to describe the communications to / from Multiple Network Devices, as well as external devices. The difference being that Multiple Network Devices use static fixed pre-known addressing, while external devices can learn their addressing values. Both types of devices however will have static flows that must be known to the system integrator and to the designer of the device.

For purposes of this section, the Multiple Network Devices will be referenced.

Configuring LAN Forwarding To Support Multiple Network Device Flows



Suppose a Multiple Network Devices exist in the aircraft called X, and is connected to a switch as shown. In order to allow the Multiple Network Device to communicate to the network the system integrator must establish the environment for this Multiple Network Device X.

First the physical location must be established to hold the Multiple Network Device X. For example if this were a HFDR, then a slide in tray with back panel connectors would be needed. The interconnect should support location pinning if needed (to describe physical position to the Multiple Network Device X), and will need to support the Ethernet connection to a switch (or other LAN connected Ethernet device).

The Multiple Network Device X expects to receive and transmit certain logical flows. These are described in the hardware specification for Multiple Network Device X. For example, the ARINC 735 HFDR hardware specification would describe the expected logical flows. The system integrator must insure that all of the expected logical flows are being provided to Multiple Network Device X by the appropriate device in the network, and that any device needing output from Multiple Network Device X can receive that information.

Finally, the system integrator must establish the appropriate forwarding tables in each switch to accommodate traffic to and from Multiple Network Device X. The addressing will be known by reading the ARINC Hardware specification for Multiple Network Device X. In this way all the addressing information is known to the components of the LAN without yet physically inserting Multiple Network Device X. When Multiple Network Device X is inserted, the communication can be supported.

It is important to understand that since the Multiple Network Device X was designed independent of the system integrator integration, that the only way for the system integrator to know what addressing values to expect is by reading the appropriate ARINC hardware specification as governed by ARINC Specification 664 for addressing.

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Default Flows In Multiple Network Devices

When a Multiple Network Device is manufactured it may need to be data loaded before it can be used operationally. In order to support data load several flows will need to exist. These logical flows will support the transfer of data to and from the data loader.

There should be a logical flow transmitted from the Multiple Network Device used to communicate to the data loader. This logical flow should use the source MAC and IP addressing, and destination MAC addressing as per the appropriate ARINC hardware specification for these Multiple Network Device types.

There should be a logical flow that the Multiple Network Device uses to receive data from the data loader. This flow will use a destination address value as per the appropriate ARINC hardware specification for these Multiple Network Device types. The source IP and MAC address used on this flow may either be locally administered, or be globally administered. This choice is left to the system integrator. The system integrator needs to insure that the logical flow can be supported by the intermediate systems used to forward the data.

There should be logical flows used to support the reception of a SNIP request to the Multiple Network Device. This may require broadcast to be used for the destination addressing.

There should be a transmit logical flow used to support the Multiple Network Device to issue broadcast data on. This could be used for example, to support ARP. Each Multiple Network Device should have this logical flow. The system integrator may however chose to not support broadcast in the system. That is fine. The switch will prevent broadcast data from entering the network. Provisions will need to be taken to insure that the Multiple Network Device can operate without use of ARP.

Multiple Network Device Flow Parameters

Multiple Network Device Flow Allocated Bandwidth and Frame Size

Each logical flow transmitted by a Multiple Network Device must declare a bandwidth utilization that it will limit its operations to, and a maximum frame size it will send. This is needed to allow a system integrator to include a Multiple Network Device into the network, and analyze the deterministic behavior of the system. Without understanding the bandwidth loading the Multiple Network Device will impose on the network there is no way to do this. The most straightforward way of approaching this is to specify a frame hertz rate for each logical flow.

Each logical flow transmitted by a Multiple Network Device must declare a maximum frame size. This is needed to aid the system integrator in doing deterministic analysis of the network with the Multiple Network Device attached.

The appropriate ARINC hardware specification must establish the frame size and rate constraint that the Multiple Network Device will adhere to. For example, HFDR will specify the flows it requires and the frame size and frame rate on each flow.

Below is a table showing an example of a table that would be found in an ARINC hardware specification for a Multiple Network Device. In the case of HFDR, for example, this would be ARINC Specification 735. In the example in this appendix, there is a Multiple Network Device of type X requiring three transmit flows, and two receive flows. The hardware specification for this example Multiple Network Device is ARINC hardware specification 73456.

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

ARINC Hardware Spec 73456
For Devices Of Type X
Multiple Network Device X
Flow Addressing Table

LRU Flow Id	Transmit / Receive	Frame Size (bytes)	Frame Rate (bytes / sec)	Frame Jitter (ms)
1	Transmit	300	10	20
2	Transmit	350	20	10
3	Transmit	200	50	5
4	Receive	300	30	30
5	Receive	340	25	20

Multiple Network Device Flow Frame Contents Format Descriptions

The format of the data payload of frames transmitted to / from Multiple Network Devices is left to the ARINC hardware specification for each device type. For example, the ARINC Hardware Spec 73456 may specify two payload formats for flow 1 as follows:

Payload Id = 34 (2 bytes)	Air Speed (4 bytes)	Frequency (2 bytes)

Payload Id = 23 (2 bytes)	Time of Day (4 bytes)	Attitude (2 bytes)	Fuel Remaining (2 bytes)

There is no need for Multiple Network Devices to agree on a common format, or to insure that payload ids are unique among Multiple Network Device types.

ARINC Specification 664 may consider recommending a format style for payloads for the convinience of the implementers in making the approach consistent.

Multiple Network Device Receiving Multicast Group Flows

Each Multiple Network Device will expect to receive certain data on multicast flows. This will be specified in the appropriate ARINC hardware specification. The system integrator will need to route those flows to the Multiple Network Device location in the LAN. The Multiple Network will simply receive the frames, not knowing the topology of the LAN in which it is existing.

Aircraft Example of Addressing for Multiple Network Devices

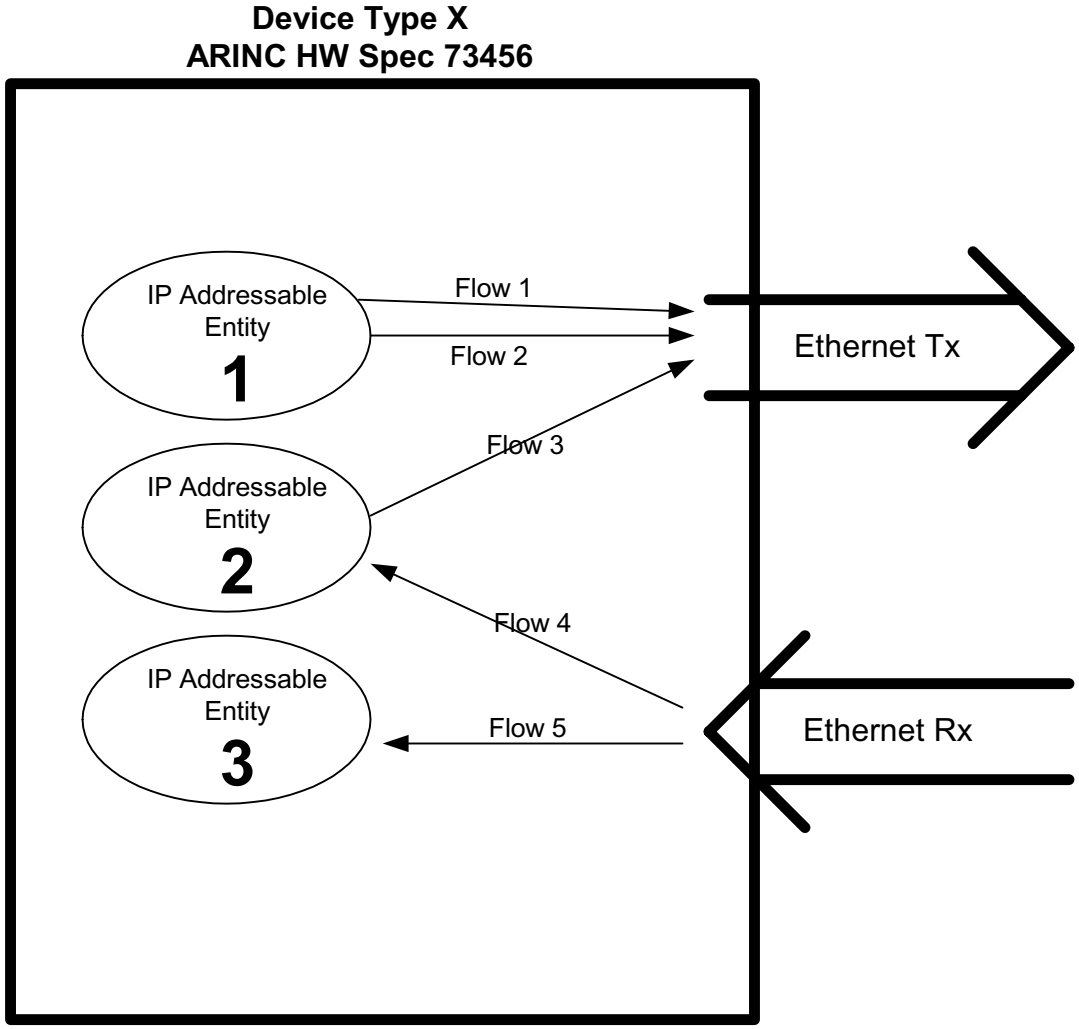
This section is included to show an example to help illustrate how an ARINC hardware specifications would describe the addressing of flows transmitted and received from a compliant device. The example describes three different Multiple Network Device types. These Multiple Network Device types are X, Y, and Z. To be clear, the Multiple Network Device could be a TCAS, HF Data Radio, Display, VHF, … , or any Multiple Network Device that is connected to a profiled network.

Multiple Network Device X is the main focus of the example. Other devices are included only so that they can originate data flows that X receives. The necessary specifications in the ARINC documents for Multiple Network Devices Y and Z are included to support these flows.

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Multiple Network Device X



The Multiple Network Device type X has three transmitting flows and two receive flows. Transmitting flows are flows that a device compliant to the specification transmits. Receiving flows are flows that a device compliant to the specification expects to receive from its environment.

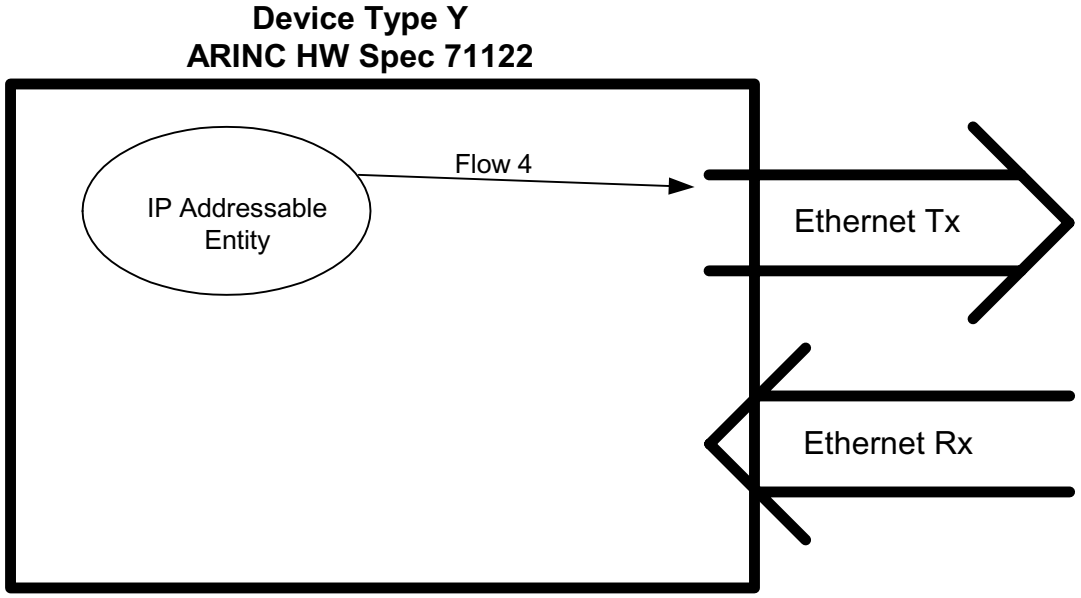
Multiple Network Device X has three IP addressable agents. Agent 1 transmits flows 1 and 2. Agent 2 transmits flow 3 and receives flow 4. Agent 3 receives flow 5.

The input and output definitions for Multiple Network Device type X are defined in ARINC 73456. The participants at the ARINC committee meetings for Specification 73456 have decided that there can be three installations of devices of type X in the aircraft. For example, three HFDRs are installed in some aircraft. In order to distinguish one device from another strapping bits have been specified to indicate which instance of the device is being referred to.

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Multiple Network Device Y



The Multiple Network Device type Y has one transmitting flow.

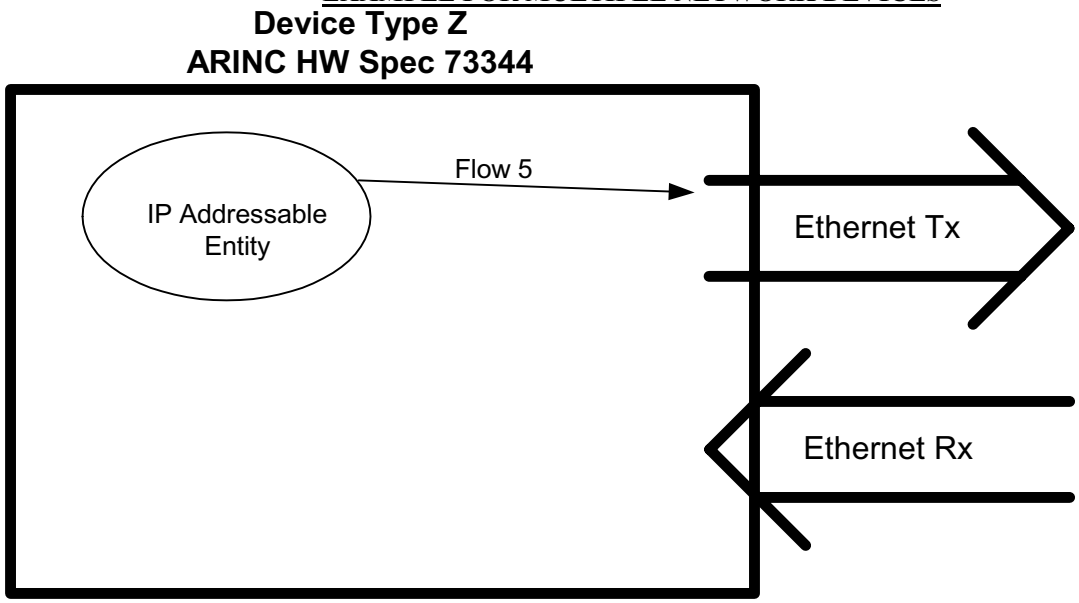
Multiple Network Device Y has one IP addressable agent. This agent transmits flow 4.

The input and output definitions for Multiple Network Device type Y are defined in ARINC 71122. The participants at the ARINC committee meetings for Specification 71122 have decided that there can be only one installations of devices of type Y in the aircraft.

Multiple Network Device Z

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES



The Multiple Network Device type Z has one transmit flow.

Multiple Network Device Z has one IP addressable agent. Agent 1 transmits flows 5.

The input and output definitions for Multiple Network Device type Z are defined in ARINC 73344. The participants at the ARINC committee meetings for Specification 73344 have decided that there can be three installations of devices of type Z in the aircraft. In order to distinguish one device from another strapping bits have been specified to indicate which instance of the device is being referred to.

ATTACHMENT 4-1

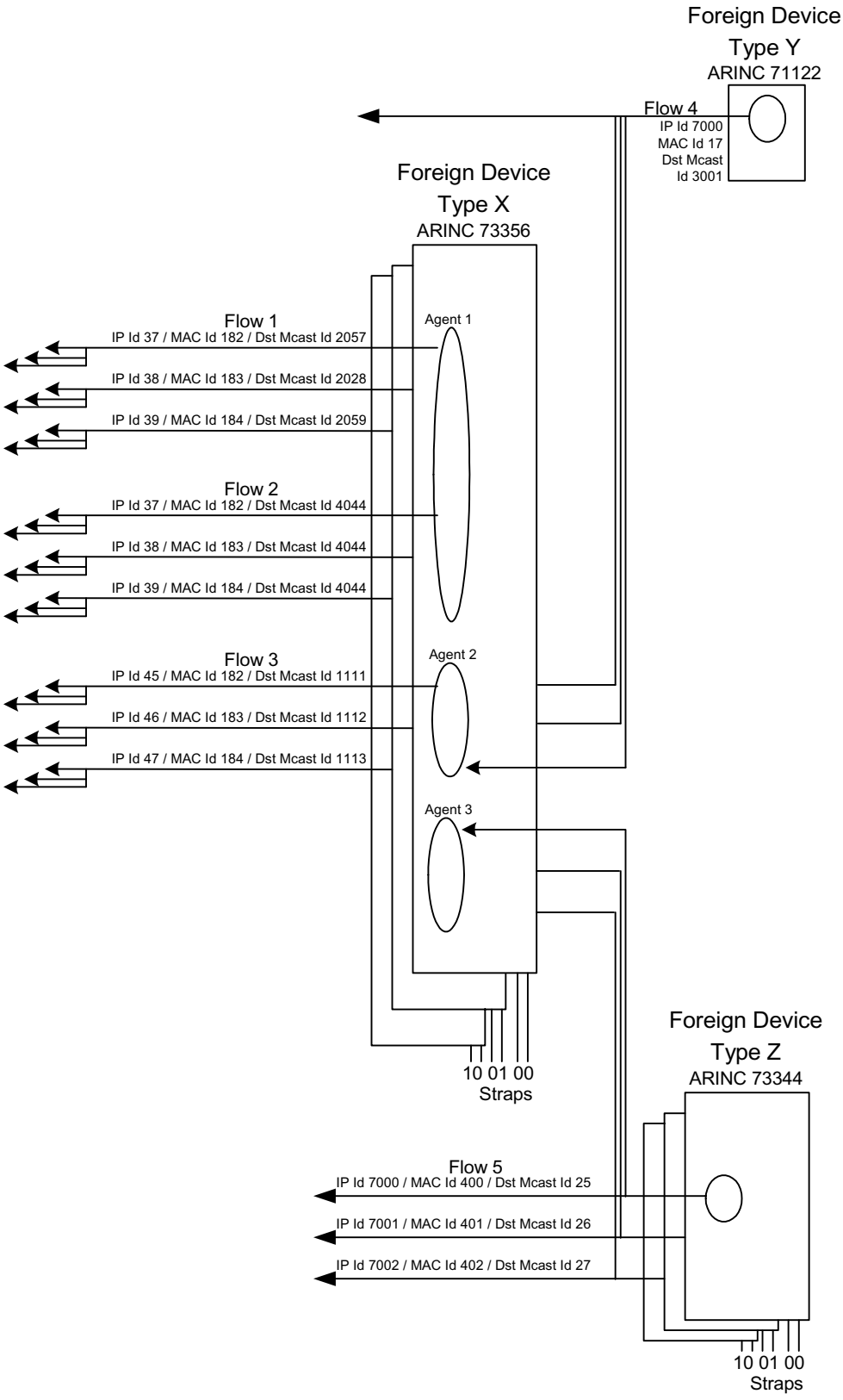
APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Aircraft Logical Topology Description

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

This drawing shows the logical topology of the aircraft. The topology is simple (in that there are few flows and few



ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

LRUs), but the example satisfies each type of usage. Therefore it is a complete example from the point of view of considering Multiple Network Device flow definition.

Notice that there are a total of 7 LRU Multiple Network Devices on the aircraft. This includes three instances of Multiple Network Device type X, one instance of Multiple Network Device type Y, and three instances of Multiple Network Device type Z.
The three instances of Multiple Network Device type X, and Multiple Network Device type Z each determine which instance they are based on strapping information. Two bits of position strapping is made available to each instance.

Flow 1 is transmitted by each of the three instances of device type X on the aircraft. Notice that they have unique Source IP, Source MAC, and Destination Multicast values.

Flow 2 and 3 are similar to flow 1.

Notice that Flow 1 and 2 are transmitted by the same IP addressable agent in device type X. Therefore, the two flows 1 and 2 transmitted by a common instance of Multiple Network Device type X share a common source IP address, while Flow 3 has a different source IP address.

Notice that the instances of Flows 1, 2, and 3 that are transmitted from a common instance of Multiple Network Device type X all have the same source MAC address. This is because the same physical port is being used to support these flows.

Notice that each instance of Multiple Network Device type X transmits Flow 1 on a different multicast address value. This allows a consumer to receive flow 1 from all three instances of Multiple Network Device type X, and determine which instance transmitted the data. This could be useful for left, right, center – for example.

Notice that all three instances of Multiple Network Device type X have been configured to transmit Flow 2 using a common destination multicast address. This allows for a circumstance where data availability is important, not the source. This is included for completeness.

Flow 4 is transmitted by the Multiple Network Device type Y. Flow 4 is consumed by each of the three instances of Multiple Network Device type X into IP addressable agent number 2. This situation is very analogous to when an LRU transmitted a 429 line that three receivers listened to. The same logical data stream is being received by three consumers.

Flow 5 is being transmitted by each of the three instances of Multiple Network Device type Z.

Notice that the source IP values of flow 5 are unique for each instance of Multiple Network Device type Z.

Notice that the source MAC values of flow 5 are unique for each instance of Multiple Network Device type Z.

Notice that each instance of Multiple Network Device type 5 uses a different multicast value when transmitting Flow 5. This situation is very analogous to when there are two control panels and two radios in an aircraft. One control panel talks to one radio, and the other talks to the other radio. In the 429 world this distinction was accomplished by having separate physical lines – one between each pair. In a switched Ethernet world this is accomplished by having each producer use a different transmit destination multicast address, and each consumer listens only to 'its' address values.

Multiple Network Device Table Descriptions

Notice, from the table, that each of the three different installations of devices of type X have a different source MAC value. The strapping value 00, 01, 10 produce ARINC Source MAC Id values of 182, 183, and 184 respectively. This insures that 1) each of the three devices will use a unique source MAC address in the aircraft, and 2) that the system integrator can know apreori what source MAC will be used by each device based on the strapping selected. This allows

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

the system integrator to fill out all needed configuration files for MAC addressing, thus preparing the aircraft topology to accept the Multiple Network Device.

In this example, Specification 73456 is assumed to define three IP addressable entities within a device of type X. The first entity has Source IP Address Id value of 37, 38, and 39 depending on the strapping selected of 00, 01, 10 respectively. This allows three devices of type X to be placed into an aircraft, and each of the common entities are uniquely IP addressable. Similarly the second entity has Source IP Address Id values of 45, 46, and 47. The third entity has Source Id Values of 51, 52, and 53. Therefore, while each device of type X has multiple IP addressable entities, all will have a unique IP address when multiple instances of device type X are installed in the same LAN network. Because the specification described the IP address values for the IP addressable entities, other applications can be written to communicate to these entities. Note that the third IP addressable entity does not transmit, therefore its IP values do not show up in the table.

The specification describes that devices of type X will have three multicast transmit flows. The Destination Multicast Id values are specified for each of these flows.

The first entity has two transmit flows. Flow number 1 has Destination Multicast Id values 2057, 2058, and 2059 depending on the strapping values 00, 01, and 10. Therefore, each instance of device X on the aircraft is transmitting flow number 1 into a different multicast group. Flow number 2 has Destination Multicast Id values 4044 regardless of the strapping values selected. Therefore, each of the three instances expected on an aircraft will transmit to the same multicast group.

The second entity has one transmit flow. Flow number 3 has Destination Multicast Id values 1111, 1112, and 1113 depending on the strapping values 00, 01, and 10. Therefore, each instance of device X on the aircraft is transmitting flow number 3 into a different multicast group.

ARINC Hardware Specification Document Flow Addressing Tables

ARINC Hardware Spec 73456
For Multiple Network Devices Of Type X
Flow Addressing Table

LRU Flow Id	Strapping Bit Value	Transmit / Receive	ARINC Source MAC Interface Id	ARINC Source IP Address Id	ARINC Destination Multicast Id
1	00	Transmit	182	37	2057
	01		183	38	2058
	10		184	39	2059
2	00	Transmit	182	37	4044
	01		183	38	4044
	10		184	39	4044
3	00	Transmit	182	45	1111
	01		183	46	1112
	10		184	47	1113
4	xx	Receive	17	6000	3001
5	**00**	Receive	400	7000	25
	01		401	7001	26
	10		402	7001	27

ATTACHMENT 4-1

APPENDIX G
EXAMPLE FOR MULTIPLE NETWORK DEVICES

Multiple Network Device Addressing Id Allocation Tables

The following tables allocate addressing ID numbers used by Multiple Network Devices.

Source MAC Interface Id
Allocation Table

Source MAC Interface Id	Device Type	Relevant Multiple Network Device Specification
17	Y	Device Y
182-184	X	Device X
400-402	Z	Device Z

Source IP Id
Allocation Table

Source IP Id	Device Type	Relevant Multiple Network Device Specification
37-39	X	Device X
45-47	X	Device X
51-53	X	Device X
6000	Y	Device Y
7000-7002	Z	Device Z

Destination Multicast Id
Allocation Table

Destination Multicast Id	Device Type	Relevant Multiple Network Device Specification
2057-2059	X	Device X
4044	X	Device X
1111-1113	X	Device X
3001	Y	Device Y
25-27	Z	Device Z

Note: Only devices transmitting on a multicast address have allocations in this table. When a device is expecting to receive traffic on a multicast address, the specification of the transmitting device must be referenced.

ATTACHMENT 4-1

APPENDIX H
INTERNET PROTOCOL VERSION 6

A. OVERVIEW

Production Internet Protocol (IP) version numbers have not been allocated in sequential order. IPv4 is the currently predominant protocol and version 5 was allocated to an experimental version. IPv6 is the designation for the follow on protocol to IPv4.

IPv4 is not considered to contain any major flaws and its good design has contributed to the success of the Internet over the last many years. However, IPv4 is based on 32-bit addresses, which might have been a good choice in 1978 but constrain the allocation of new sub-networks (subnets) and their attached computers today. IPv6 provides 128-bit addresses. It could have been sufficient to have simply increased the address size in IPv4 and have kept everything else unchanged. However, the 20+ years of experience brought additional insights. IPv6 was developed between 1992 and 1996 and incorporates the lessons learned without losing the successful characteristics of IPv4. IPv4 and IPv6 can coexist on the Internet such that there can be a migration to the new version over time.

When deploying new networks when assigned addressing is not constrained, using public Internet addresses rather than private addresses is a good choice. Firewall and other security mechanisms can be used to control the actual allowed communications. An Aircraft Data Network should support communications between onboard equipment as well as communications between Air Traffic Management, Airline Operations Centers and other legitimate parties such as weather information providers. Each aircraft should contain one (or perhaps more than one) subnet with devices that are addressable from these diverse parties. With security mechanisms in place, these aircraft devices can be assigned public Internet addresses. Many major corporate networks operate this way today with secure public addresses.

The expected number of aircraft that are candidates for improved communications from Air Traffic Management, Airline Operations Centers and other sources is expected to be over 100,000 in the next 25 years. This includes current air transport providers as well as business and general aviation aircraft that would be suitable for an advanced communication system.

Introducing 100,000 new subnets in IPv4 for these applications is not practical. IPv6 with its 128-bit addresses is positioned to provide the added addressing capability needed for ADN. Supporting IPv6 for ADN applications that require communications to terrestrial systems is a good choice. IPv6 is being deployed today because many countries such as Japan do not have sufficient subnet addresses. IPv6 is expected to be available world wide in the near future.

ATTACHMENT 4-1

APPENDIX H
INTERNET PROTOCOL VERSION 6

B. IPv6 ADDRESSING BASICS

IPv6 provides 128 bits of addressing compared to 32 bits in IPv4.

There are three classes of addresses in IPv6:

- *Unicast*: Associated with a single physical interface
- *Multicast*: Associated with a group of interfaces. Packets destined for a multicast address are delivered to all interfaces in the group.
- *Anycast*: Associated with a group of interfaces. Packets destined for an anycast address are delivered to one and only one interface in the group – the 'nearest' interface.

At this time, this supplement only covers Unicast addresses.

IPv4 represents 32-bit addresses as a series of four decimal numbers separated by periods. An example is 192.168.1.3. Each number ranges from 0 to 255.

IPv6 uses a string of eight 4-digit hexadecimal numbers separated by colons. Each 4-digit hexadecimal number represents 16 bits.

The human readable forms of IPv6 addresses are:

- **Preferred form**
  x:x:x:x:x:x:x:x where x is a four digit hexadecimal value
  Example – c0a:b2d3:0:0:0:0:8888:99
- **Alternate compressed form**
  x:x::x:x where the :: denotes several groups of zeros
  Example – c0a:b2d3::8888:99
- **Alternate IPv4 compatible form**
  ::x.x.x.x where x.x.x.x is an IPv4 address.
- **Classless Inter-Domain Routing (CIDR) convention is used for prefixes**
  x:x:x::/y where x:x:x:: is the address and y is the prefix length
  Example – c0a:b2d3::/48

IPv6 address types are defined in RFC 2373. The IPv6 address type is indicated by a variable number of leading bits in the address, called the Format Prefix (FP).

Currently defined Format Prefixes (in binary) are:

- '0000 0000' Reserved for 'unspecified', loopback and IPv4
- '0000 001' OSI NSAP Allocation
- '0000 010' IPX Allocation
- '001' Aggregatable Global Unicast Addresses (AGUA)
- '1111 1110 10' Link-local Unicast Addresses
- '1111 1110 11' Site-local Unicast Addresses
- '1111 1111' Multicast Addresses

All other prefixes are unassigned.

Note that OSI addresses can be translated to IPv6 so that an IPv6 network can serve as a transport for OSI packets. This offers the capability to support Aeronautical Telecommunication Network (ATN) applications.

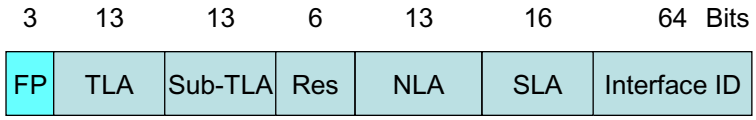
The Aggregatable Global Unicast Address (AGUA) format specified in RFC 2374 provides the means to allocate Internet addresses. The structure for Unicast addresses has evolved over the past few years based on real world experience. These changes have been documented as Regional Internet Registry (RIR) policies rather than RFCs. (Regional Internet Registries are the organizations that assign Internet addresses and domain names.) The document "The Provisional IPv6 Assignment and Allocation Policy" can be found at any RIR web site or by searching for the title on the Internet. These policies are continuously evolving and a new one was published on December 22, 2001. It is pending adoption.

The AGUA address structure from RFC 2374 as modified by current RIR policies is shown in the following diagram.

ATTACHMENT 4-1

APPENDIX H
INTERNET PROTOCOL VERSION 6

Aggregatable Global Unicast Address (AGUA) Format -
RFC 2374 as modified by RIR policies



FP	**Format prefix ('001' = AGUA)**
TLA ID	Top-Level Aggregation identifier ('0x001')
Sub-TLA ID	Sub-TLA identifier
Res	Reserved
NLA ID	Next-Level Aggregation identifier
SLA ID	Site-Level Aggregation identifier
Interface ID	Interface Identifier

Note: In the following descriptions, the notation 0x signifies the following attached characters are hexadecimal digits.

The TLA value 0x0001 indicates the Sub-TLA digits are assigned. The TLA value 0x0002 indicates IPv6 to IPv4 mapping. Other values are for testing or are reserved.

The Sub-TLA values used with TLA 0x0001 are assigned by the Internet Assigned Number Authority (IANA) to Regional Internet Registries (RIRs).

The Sub-TLA values are used by RIRs to allocate addresses to Tier 1 Internet Service Providers (ISPs). They are uniquely identified by the last 6 bits in the Sub-TLA field. These Tier 1 ISPs allocate up to 8191 (13 bits worth of) NLAs for each one of several Sub-TLAs they are assigned. As of 11/28/01 there are 21 TLA Registry Allocations. (Note that the term "TLA Registry" is used but the Sub-TLA is what is used to assign numbers to them.)

Each NLA value, assigned by a TLA Registry, represents a lower Tier ISP or an enterprise.

Each NLA can itself assign up to 65,535 sites using the SLA field.

Each IPv6 Site is analogous to an IPv4 subnet. The shortest prefix that can be assigned to a site is 64 bits. The remaining 64 bits are reserved for interface addresses.

The 64-bit Interface ID can accommodate an essentially unlimited number of interfaces per site. The Interface ID cannot be used to further partition a site. Typically, the Interface ID is derived from the Media Access Control (MAC) address. Standard encoding is defined for EUI-64 and Ethernet MACs. Support for a pseudo-random Interface ID is in progress.

ADN Unicast addresses need to be allocated under the current RIR policy. The AGUA addressing described in the remainder of this document is based on currently adopted RIR policies.

C. APPLICATION TO ADN

Since ADN applications could require more than 100,000 subnets (sites), at least two NLAs would be required. These (few) NLAs should be

contiguous in a binary block for optimal route aggregation.

If ARINC or some other organization were to become a TLA Registry for all ADNs, an entire block of NLA/SLAs would be available for allocation to aircraft.

Attachment 4-2

From: Jerry Van Baren [vanbaren_gerald@si.com]
Sent: Friday, June 14, 2002 7:44 PM
To: RCOURTNE@arinc.com
Cc: jean-paul.moreaux@airbus.com
Subject: Action Item: profiling of IPv6

Hi Roy,

Attached is an Excel spreadsheet with the IPv6 RFC (2460) slashed up into "must/should/may/should not/must not" tags. Pretty much all of the RFC text should be in the spreadsheet, but I cut out figures and examples: I didn't feel they were necessary for profiling purposes and they were not very readable in Excel without a lot of effort reformatting them so they ended up in the bit bucket.

The first question to be answered by the Working Group is whether IPv6 needs to be "profiled" at all. IPv4 is being used in moderately to highly constrained networks and I consider it to be more than adequate for that purpose. The question is, in a profiled network, does IPv6 bring any benefit that is unavailable using IPv4 and does the benefit outweigh the added complexity of IPv6?

The principle of profiling is that an industry standard (i.e. RFC) is selectively modified to meet more restrictive requirements necessary for certain applications. Full RFC-based implementations are encouraged in any and all applications that do not need additional restrictions. In the light of this, is there things in IPv6 that we need in a restrictive (profiled) network but, at the same time, cannot use a full IPv6 implementation?

Note that I have not done any profiling: the "Profiled" "must/optional/must not" simply reflect the respective RFC-2460 columns. This would be the job of the Working Group as individuals and companies. As you may deduce from the fact that I didn't do any profiling, I am not convinced at the moment that IPv6 needs to be profiled (but then, that may be because I don't envision using it any time soon for stuff we are working on).

gvb

The following list comes as an extra value at no additional cost...

Other RFCs that reference RFC-2460 (IPv6):

Network Working Group M. Daniele
Request for Comments: 2452 Compaq Computer Corporation
Category: Standards Track December 1998

IP Version 6 Management Information Base
for the Transmission Control Protocol

Network Working Group M. Daniele
Request for Comments: 2454 Compaq Computer Corporation
Category: Standards Track December 1998

Attachment 4-2

IP Version 6 Management Information Base
for the User Datagram Protocol

Network Working Group — T. Narten, IBM
Request for Comments: 2461
Obsoletes: 1970 — E. Nordmark, Sun Microsystems
Category: Standards Track — W. Simpson, Daydreamer
December 1998

Neighbor Discovery for IP Version 6 (IPv6)

Network Working Group — A. Conta, Lucent
Request for Comments: 2463
Obsoletes: 1885 — S. Deering, Cisco Systems
Category: Standards Track
December 1998

Internet Control Message Protocol (ICMPv6)
for the Internet Protocol Version 6 (IPv6)
Specification

Network Working Group — M. Crawford, Fermilab
Request for Comments: 2464
Obsoletes: 1972 — December 1998
Category: Standards Track

Transmission of IPv6 Packets over Ethernet Networks

Network Working Group — D. Haskin, S. Onishi, Bay Networks, Inc.
Request for Comments: 2465
Category: Standards Track
December 1998

Management Information Base for IP Version 6:
Textual Conventions and General Group

Network Working Group — D. Haskin, S. Onishi, Bay Networks, Inc.
Request for Comments: 2466
Category: Standards Track
December 1998

Management Information Base for IP Version 6:
ICMPv6 Group

Network Working Group — M. Crawford, Fermilab
Request for Comments: 2467
Obsoletes: 2019 — December 1998
Category: Standards Track

Attachment 4-2

Transmission of IPv6 Packets over FDDI Networks

Network Working Group — M. Crawford
Request for Comments: 2470 — Fermilab
Category: Standards Track — T. Narten
IBM
S. Thomas
TransNexus
December 1998

Transmission of IPv6 Packets over Token Ring Networks

Network Working Group — D. Haskin
Request for Comments: 2472 — E. Allen
Obsoletes: 2023 — Bay Networks, Inc.
Category: Standards Track — December 1998

IP Version 6 over PPP

Network Working Group — A. Conta
Request for Comments: 2473 — Lucent Technologies Inc.
Category: Standards Track — S. Deering
Cisco Systems
December 1998

Generic Packet Tunneling in IPv6
Specification

Network Working Group — K. Nichols
Request for Comments: 2474 — Cisco Systems
Obsoletes: 1455, 1349 — S. Blake
Category: Standards Track — Torrent Networking Technologies
F. Baker
Cisco Systems
D. Black
EMC Corporation
December 1998

Definition of the Differentiated Services Field (DS Field)
in the IPv4 and IPv6 Headers

Network Working Group — G. Armitage
Request for Comments: 2491 — Lucent Technologies
Category: Standards Track — P. Schulter
Bright Tiger Technologies
M. Jork
Digital Equipment GmbH
G. Harter
Compaq
January 1999

Attachment 4-2

IPv6 over Non-Broadcast Multiple Access (NBMA) networks

Network Working Group — I. Souvatzis
Request for Comments: 2497 — The NetBSD Project
See Also: 1201 — January 1999
Category: Standards Track

Transmission of IPv6 Packets over ARCnet Networks

Network Working Group — M. Degermark
Request for Comments: 2507 — Lulea University of Technology/SICS
Category: Standards Track — B. Nordgren
Lulea University of Technology/Telia Research AB
S. Pink
Lulea University of Technology/SICS
February 1999

IP Header Compression

Network Working Group — D. Johnson
Request for Comments: 2526 — Carnegie Mellon University
Category: Standards Track — S. Deering
Cisco Systems, Inc.
March 1999

Reserved IPv6 Subnet Anycast Addresses

Network Working Group — B. Carpenter
Request for Comments: 2529 — IBM
Category: Standards Track — C. Jung
3Com
March 1999

Transmission of IPv6 over IPv4 Domains without Explicit Tunnels

---

Network Working Group — P. Marques
Request for Comments: 2545 — cisco Systems, Inc.
Category: Standards Track — F. Dupont
Inria
March 1999

Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing

---

Network Working Group — R. Gilligan
Request for Comments: 2553 — FreeGate
Obsoletes: 2133 — S. Thomson
Category: Informational — Bellcore
J. Bound
Compaq
W. Stevens
Consultant
March 1999

Attachment 4-2

Basic Socket Interface Extensions for IPv6

Network Working Group — D. Borman
Request for Comments: 2675 — Berkeley Software Design
Obsoletes: 2147 — S. Deering
Category: Standards Track — Cisco
R. Hinden
Nokia
August 1999

IPv6 Jumbograms

Network Working Group — S. Deering
Request for Comments: 2710 — Cisco Systems
Category: Standards Track — W. Fenner
AT&T Research
B. Haberman
IBM
October 1999

Multicast Listener Discovery (MLD) for IPv6

Network Working Group — C. Partridge
Request for Comments: 2711 — BBN
Category: Standards Track — A. Jackson
BBN
October 1999

IPv6 Router Alert Option

Network Working Group — R. Coltun
Requests for Comments: 2740 — Siara Systems
Category: Standards Track — D. Ferguson
Juniper Networks
J. Moy
Sycamore Networks
December 1999

OSPF for IPv6

Network Working Group — E. Nordmark
Request for Comments: 2765 — Sun Microsystems
Category: Standards Track — February 2000

Stateless IP/ICMP Translation Algorithm (SIIT)

Network Working Group — K. Tsuchiya
Requests for Comments: 2767 — H. Higuchi
Category: Informational — Y. Atarashi
Hitachi
February 2000

Attachment 4-2

Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS)

Network Working Group — S. Bradner, Harvard University
Request for Comments: 2780
BCP: 37 — V. Paxson, ACIRI
Category: Best Current Practice
March 2000

IANA Allocation Guidelines For Values In the Internet Protocol and Related Headers

---

Network Working Group — R. Gilligan, FreeGate Corp.
Request for Comments: 2893
Obsoletes: 1933 — E. Nordmark, Sun Microsystems, Inc.
Category: Standards Track
August 2000

Transition Mechanisms for IPv6 Hosts and Routers

Network Working Group — M. Crawford, Fermilab
Request for Comments: 2894
Category: Standards Track — August 2000

Router Renumbering for IPv6

Network Working Group — R. Hinden, Nokia
Request for Comments: 2928
Category: Informational — S. Deering, Cisco; R. Fink, LBNL; T. Hain, Microsoft
September 2000

Initial IPv6 Sub-TLA ID Assignments

Network Working Group — B. Carpenter, K. Moore
Request for Comments: 3056
Category: Standards Track — February 2001

Connection of IPv6 Domains via IPv4 Clouds

Network Working Group — H. Kitamura, NEC Corporation
Request for Comments: 3089
Category: Informational — April 2001

Attachment 4-2

A SOCKS-based IPv6/IPv4 Gateway Mechanism

Network Working Group — C. Bormann, Editor, TZI/Uni Bremen
Request for Comments: 3095 — C. Burmeister, Matsushita
Category: Standards Track — M. Degermark, Univ. of Arizona
H. Fukushima, Matsushita
H. Hannu, Ericsson
L-E. Jonsson, Ericsson
R. Hakenberg, Matsushita
T. Koren, Cisco
K. Le, Nokia
Z. Liu, Nokia
A. Martensson, Ericsson
A. Miyazaki, Matsushita
K. Svanbro, Ericsson
T. Wiebke, Matsushita
T. Yoshimura, NTT DoCoMo
H. Zheng, Nokia
July 2001

RObust Header Compression (ROHC):
Framework and four profiles: RTP, UDP, ESP, and uncompressed

Network Working Group — A. Conta
Request for Comments: 3122 — Transwitch Corporation
Category: Standards Track — June 2001

Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification

---

Network Working Group — K. Fujisawa
Request for Comments: 3146 — A. Onoe
Category: Standards Track — Sony Corporation
October 2001

Transmission of IPv6 Packets over IEEE 1394 Networks

Network Working Group — B. Aboba
Request for Comments: 3162 — Microsoft
Category: Standards Track — G. Zorn
Cisco Systems
D. Mitton
Circular Logic UnLtd.
August 2001

RADIUS and IPv6

Network Working Group — A. Shacham
Request for Comments: 3173 — Juniper
Obsoletes: 2393 — B. Monsour
Category: Standards Track — Consultant
R. Pereira
Cisco
M. Thomas
Consultant
September 2001

ATTACHMENT 4-3

	RFC-2460	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
4 IPv6 Extension Headers										
In IPv6 optional internet-layer information is encoded in separate headers that may be placed between the IPv6 header and the upper-layer header in a packet. There are a small number of such extension headers each identified by a distinct Next Header value.	4			X				X		
As illustrated in these examples, an IPv6 packet may carry zero, one, or more extension headers, each identified by the Next Header field of the preceding header:	4			X				X		
With one exception, extension headers are not examined or processed by any node along a packet's delivery path, until the packet reaches the node (or each of the set of nodes, in the case of multicast) identified in the Destination Address field of the IPv6 header. There, normal demultiplexing on the Next Header field of the IPv6, header invokes the module to process the first extension header, or the upper-layer header if no extension header is present.	4					X			X	
The contents and semantics of each extension header determine whether or not to proceed to the next header, Therefore, extension headers must be processed strictly in the order they appear in the packet;…	4	X					X			
...a receiver must not, for example, scan through a packet looking for a particular kind of extension header and process that header prior to processing all preceding ones.						X			X	

The exception referred to in the preceding paragraph is the Hop-by-Hop Options header, which carries information that must be examined and processed by every node along a packet's delivery path, including the source and destination nodes.	4	X					X		
The Hop-by-Hop Options header, when present, must immediately follow the IPv6 header. Its presence is indicated by the value zero in the Next Header field of the IPv6 header.	4	X					X		
If, as a result of processing a header, a node is required to proceed to the next header but the Next Header value in the current header is unrecognized by the node, it should discard the packet and send an ICMP Parameter Problem message to the source of the packet, with an ICMP Code value of 1 (unrecognized Next Header type encountered) and the ICMP Pointer field containing the offset of the unrecognized value within the original packet.	4		X					X	Not all implementations fully implement ICMP
The same action should be taken if a node encounters a Next Header value of zero in any header other than an IPv6 header.	4		X					X	
Each extension header is an integer multiple of 8 octets long, in order to retain 8-octet alignment for subsequent headers, Multi-octet fields within each extension header are aligned on their natural boundaries, i.e., fields of width n octets are placed at an integer multiple of n octets from the start of the header, for n = 1, 2, 4, or 8.	4	X					X		
A full implementation of IPv6 includes implementation of the following extension headers:	4	X					X		Subject to debate :-)
Hop-by-Hop Options	4	X					X		
Routing (Type 0)	4	X					X		
Fragment	4	X					X		
Destination Options	4	X					X		
Authentication	4	X					X		
Encapsulating Security Payload	4	X					X		
The first four are specified in this document; the last two are specified in [RFC-2402] and [RFC-2406], respectively.									
4.1 Extension Header Order									

When more than one extension header is used in the same packet, it is recommended that those headers appear in the following order: IPv6 header Hop-by-Hop Options header Destination Options header (note 1) Routing header Fragment header Authentication header (note 2) Encapsulating Security Payload header (note 2) Destination Options header (note 3) upper-layer header note 1: for options to be processed by the first destination that appears in the IPv6 Destination Address field plus subsequent destinations listed in the Routing header note 2: additional recommendations regarding the relative order of the Authentication and Encapsulating Security Payload headers are given in [RFC-2406] note 3: for options to be processed only by the final destination of the packet	4.1		X					X		
Each extension header should occur at most once…	4.1		X					X		
...except for the Destination Options header which should occur at most twice (once before a Routing header and once before the upper-layer header)	4.1		X					X		
If the upper-layer header is another IPv6 header (in the case of IPv6 being tunneled over or encapsulated in IPv6), it may be followed by its own extension headers, which are separately subject to the same ordering recommendations	4.1			X				X		
If and when other extension headers are defined, their ordering constraints relative to the above listed headers must be specified.										Not an implementation requirement
IPv6 nodes must accept and attempt to process extension headers in any order and occurring any number of times in the same packet, except for the Hop-by-Hop Options header which is restricted to appear immediately after an IPv6 header only.	4.1	X					X			

Nonetheless, it is strongly advised that sources of IPv6 packets adhere to the above recommended order until and unless subsequent specifications revise that recommendation.	4.1		X				X		
4.2 Options									
Two of the currently-defined extension headers -- the Hop-by-Hop Options header and the Destination Options header -- carry a variable number of type-length-value (TLV) encoded options, of the following format:									
The sequence of options within a header must be processed strictly in the order they appear in the header; a receiver must not, for example, scan through the header looking for a particular kind of option and process that option prior to processing all preceding ones.	4.2	X				X			
The Option Type identifiers are internally encoded such that their highest-order two bits specify the action that must be taken if the processing IPv6 node does not recognize the Option Type:									
00 - skip over this option and continue processing the header	4.2	X				X			
01 - discard the packet	4.2	X				X			
10 - discard the packet and, regardless of whether or not the packet's Destination Address was a multicast address, send an ICMP Parameter Problem, Code 2, message to the packet's Source Address, pointing to the unrecognized Option Type.	4.2	X				X			Not all implementations fully implement ICMP
11 - discard the packet and, only if the packet's Destination Address was not a multicast address, send an ICMP Parameter Problem, Code 2, message to the packet's Source Address pointing to the unrecognized Option Type.	4.2	X				X			Not all implementations fully implement ICMP
The third-highest-order bit of the Option Type specifies whether or not the Option Data of that option can change en-route to the packet's final destination.									

When an Authentication header is present in the packet, for any option whose data may change en-route, its entire Option Data field must be treated as zero-valued octets when computing or verifying the packet's authenticating value. 0 - Option Data does not change en-route 1 - Option Data may change en-route The three high-order bits described above are to be treated as part of the Option Type, not independent of the Option Type. That is, a particular option is identified by a full 8-bit Option Type, not just the low-order 5 bits of an Option Type. The same Option Type numbering space is used for both the Hop-by-Hop Options header and the Destination Options header.	4.2	X				X				
However, the specification of a particular option may restrict its use to only one of those two headers										Not an implementation requirement
Individual options may have specific alignment requirements, to ensure that multi-octet values within Option Data fields fall on natural boundaries. The alignment requirement of an option is specified using the notation xn+y, meaning the Option Type must appear at an integer multiple of x octets from the start of the header, plus y octets. For example: 2n means any 2-octet offset from the start of the header. 8n+2 means any 8-octet offset from the start of the header, plus 2 octets.										Not an implementation requirement
There are two padding options which are used when necessary to align subsequent options and to pad out the containing header to a multiple of 8 octets in length.										
These padding options must be recognized by all IPv6 implementations:	4.2	X				X				
Pad1 option (alignment requirement: none) NOTE! the format of the Pad1 option is a special case -- it does not have length and value fields.	4.2	X				X				

The Pad1 option is used to insert one octet of padding into the Options area of a header. If more than one octet of padding is required, the PadN option, described next, should be used, rather than multiple Pad1 options.									
PadN option (alignment requirement: none)	4.2	X				X			
The PadN option is used to insert two or more octets of padding into the Options area of a header. For N octets of padding, the Opt Data Len field contains the value N-2, and the Option Data consists of N-2 zero-valued octets.									
Appendix B contains formatting guidelines for designing new options									
4.3 Hop-by-Hop Options Header									
The Hop-by-Hop Options header is used to carry optional information that must be examined by every node along a packet's delivery path. The Hop-by-Hop Options header is identified by a Next Header value of 0 in the IPv6 header, and has the following format:									
The only hop-by-hop options defined in this document are the Pad1 and PadN options specified in section 4.2.									
4.4 Routing Header									
The Routing header is used by an IPv6 source to list one or more intermediate nodes to be visited on the way to a packet's destination. This function is very similar to IPv4's Loose Source and Record Route option. The Routing header is identified by a Next Header value of 43 in the immediately preceding header, and has the following format:									Source routing is not generally allowed in "profiled" networks.
If, while processing a received packet, a node encounters a Routing header with an unrecognized Routing Type value, the required behavior of the node depends on the value of the Segments Left field, as follows:	4.4	X				X			

If Segments Left is zero, the node must ignore the Routing header and proceed to process the next header in the packet, whose type is identified by the Next Header field in the Routing header.	4.4	X					X			
If Segments Left is non-zero, the node must discard the packet and send an ICMP Parameter Problem, Code 0, message to the packet's Source Address, pointing to the unrecognized Routing Type.	4.4	X					X			
If, after processing a Routing header of a received packet, an intermediate node determines that the packet is to be forwarded onto a link whose link MTU is less than the size of the packet, the node must discard the packet and send an ICMP Packet Too Big message to the packet's Source Address.	4.4	X					X			
Multicast addresses must not appear in a Routing header of Type 0, or in the IPv6 Destination Address field of a packet carrying a Routing header of Type 0.						X			X	
A Routing header is not examined or processed until it reaches the node identified in the Destination Address field of the IPv6 header In that node, dispatching on the Next Header field of the immediately preceding header causes the Routing header module to be invoked which, in the case of Routing Type 0, performs the following algorithm:										
4.5 Fragment Header										
The Fragment header is used by an IPv6 source to send a packet larger than would fit in the path MTU to its destination. (Note: unlike IPv4, fragmentation in IPv6 is performed only by source nodes, not by routers along a packet's delivery path -- see section 5.) The Fragment header is identified by a Next Header value of 44 in the immediately preceding header, and has the following format:										

In order to send a packet that is too large to fit in the MTU of the path to its destination, a source node may divide the packet into fragments and send each fragment as a separate packet, to be reassembled at the receiver.	4.5			X			X		
For every packet that is to be fragmented, the source node generates an Identification value, The Identification must be different than that of any other fragmented packet sent recently* with the same Source Address and Destination Address. If a Routing header is present, the Destination Address of concern is that of the final destination. * recently means within the maximum likely lifetime of a packet, including transit time from source to destination and time spent awaiting reassembly with other fragments of the same packet. However, it is not required that a source node know the maximum, packet lifetime. Rather, it is assumed that the requirement can be met by maintaining the Identification value as a simple, 32-bit, wrap-around counter, incremented each time a packet must be fragmented, It is an implementation choice whether to maintain a single counter for the node or multiple counters, e.g., one for each of the node's possible source addresses, or one for each active (source address, destination address) combination.	4.5	X				X			
The initial, large, unfragmented packet is referred to as the original packet, and it is considered to consist of two parts, as illustrated:									

The Unfragmentable Part consists of the IPv6 header plus any extension headers that must be processed by nodes en route to the destination, that is, all headers up to and including the Routing header if present, else the Hop-by-Hop Options header if present, else no extension headers. The Fragmentable Part consists of the rest of the packet, that is, any extension headers that need be processed only by the final destination node(s), plus the upper-layer header and data.									Not an implementation requirement?
The Fragmentable Part of the original packet is divided into fragments, each, except possibly the last (rightmost) one, being an integer multiple of 8 octets long. The fragments are transmitted in separate fragment packets as illustrated: Each fragment packet is composed of: (1) The Unfragmentable Part of the original packet, with the Payload Length of the original IPv6 header changed to contain the length of this fragment packet only (excluding the length of the IPv6 header itself), and the Next Header field of the last header of the Unfragmentable Part changed to 44. (2) A Fragment header containing: The Next Header value that identifies the first header of the Fragmentable Part of the original packet. A Fragment Offset containing the offset of the fragment in 8-octet units, relative to the start of the Fragmentable Part of the original packet. The Fragment Offset of the first (leftmost) fragment is 0. An M flag value of 0 if the fragment is the last (rightmost) one, else an M flag value of 1. The Identification value generated for the original packet. (3) The fragment itself The lengths of the fragments must be chosen such that the resulting fragment packets fit within the MTU of the path to the packets' destination(s) At the destination, fragment packets are reassembled into their original, unfragmented form, as illustrated: The following rules govern reassembly:	4.5	X				X			Requirement: all but the last fragment must be an integer multiple of 8 octets long.

ATTACHMENT 4-3

An original packet is reassembled only from fragment packets that have the same Source Address, Destination Address, and Fragment Identification.	4.5	X					X			
The Unfragmentable Part of the reassembled packet consists of all headers up to, but not including, the Fragment header of the first fragment packet (that is, the packet whose Fragment Offset is zero), with the following two changes:	4.5	X					X			
The Next Header field of the last header of the Unfragmentable Part is obtained from the Next Header field of the first fragment's Fragment header	4.5	X					X			
The Payload Length of the reassembled packet is computed from the length of the Unfragmentable Part and the length and offset of the last fragment, For example, a formula for computing the Payload Length of the reassembled original packet is:	4.5	X					X			
The Fragmentable Part of the reassembled packet is constructed from the fragments following the Fragment headers in each of the fragment packets. The length of each fragment is computed by subtracting from the packet's Payload Length the length of the headers between the IPv6 header and fragment itself; its relative position in Fragmentable Part is computed from its Fragment Offset value.	4.5	X					X			
The Fragment header is not present in the final, reassembled packet.	4.5	X					X			
The following error conditions may arise when reassembling fragmented packets:										
If insufficient fragments are received to complete reassembly of a packet within 60 seconds of the reception of the first-arriving fragment of that packet, reassembly of that packet must be abandoned and all the fragments that have been received for that packet must be discarded...	4.5	X					X			
...If the first fragment (i.e., the one with a Fragment Offset of zero) has been received, an ICMP Time Exceeded -- Fragment Reassembly Time Exceeded message should be sent to the source of that fragment.	4.5		X					X		

If the length of a fragment, as derived from the fragment packet's Payload Length field, is not a multiple of 8 octets and the M flag of that fragment is 1, then that fragment must be discarded…	4.5	X					X		
...and an ICMP Parameter Problem, Code 0, message should be sent to the source of the fragment, pointing to the Payload Length field of the fragment packet.	4.5		X					X	
If the length and offset of a fragment are such that the Payload Length of the packet reassembled from that fragment would exceed 65,535 octets, then that fragment must be discarded…	4.5	X					X		
...and an ICMP Parameter Problem, Code 0, message should be sent to the source of the fragment, pointing to the Fragment Offset field of the fragment packet.	4.5		X					X	
The following conditions are not expected to occur, but are not considered errors if they do:									
The number and content of the headers preceding the Fragment header of different fragments of the same original packet may differ. Whatever headers are present, preceding the Fragment header in each fragment packet, are processed when the packets arrive, prior to queueing the fragments for reassembly. Only those headers in the Offset zero fragment packet are retained in the reassembled packet.									
The Next Header values in the Fragment headers of different fragments of the same original packet may differ. Only the value from the Offset zero fragment packet is used for reassembly.									
4.6 Destination Options Header									
The Destination Options header is used to carry optional information that need be examined only by a packet's destination node(s). The Destination Options header is identified by a Next Header value of 60 in the immediately preceding header, and has the following format:									
The only destination options defined in this document are the Pad1 and PadN options specified in section 4.2									

Note that there are two possible ways to encode optional destination information in an IPv6 packet: either as an option in the Destination Options header, or as a separate extension header. The Fragment header and the Authentication header are examples of the latter approach. Which approach can be used depends on what action is desired of a destination node that does not understand the optional information:									
If the desired action is for the destination node to discard the packet and, only if the packet's Destination Address is not a multicast address, send an ICMP Unrecognized Type message to the packet's Source Address, then the information may be encoded either as a separate header or as an option in the Destination Options header whose Option Type has the value 11 in its highest-order two bits. The choice may depend on such factors as which takes fewer octets, or which yields better alignment or more efficient parsing.				X			X		Not an implementation requirement?
If any other action is desired, the information must be encoded as an option in the Destination Options header whose Option Type has the value 00, 01, or 10 in its highest-order two bits, specifying the desired action (see section 4.2).				X			X		Not an implementation requirement?
4.7 No Next Header									
The value 59 in the Next Header field of an IPv6 header or any extension header indicates that there is nothing following that header. If the Payload Length field of the IPv6 header indicates the presence of octets past the end of a header whose Next Header field contains 59, those octets must be ignored, and passed on unchanged if the packet is forwarded.	4.7	X				X			
5. Packet Size Issues									

IPv6 requires that every link in the internet have an MTU of 1280 octets or greater.	5		X					X		This is substantially more than IPv4's recommended minimum MTU of 576.
On any link that cannot convey a 1280-octet packet in one piece, link-specific fragmentation and reassembly must be provided at a layer below IPv6.			X					X		IPv4 does not require special fragmentation and reassembly handling for links with MTUs less than the recommended values.
Links that have a configurable MTU (for example, PPP links [RFC-1661]) must be configured to have an MTU of at least 1280 octets;	5		X					X		
...it is recommended that they be configured with an MTU of 1500 octets or greater, to accommodate possible encapsulations (i.e., tunneling) without incurring IPv6-layer fragmentation.	5			X					X	
From each link to which a node is directly attached, the node must be able to accept packets as large as that link's MTU.	5		X					X		
It is strongly recommended that IPv6 nodes implement Path MTU Discovery [RFC-1981], in order to discover and take advantage of path MTUs greater than 1280 octets.	5			X					X	
However, a minimal IPv6 implementation (e,g,, in a boot ROM) may simply restrict itself to sending packets no larger than 1280 octets, and omit implementation of Path MTU Discovery.	5				X				X	
In order to send a packet larger than a path's MTU, a node may use the IPv6 Fragment header to fragment the packet at the source and have it reassembled at the destination(s). However, the use of such fragmentation is discouraged in any application that is able to adjust its packets to fit the measured path MTU (i.e., down to 1280 octets).	5				X				X	
A node must be able to accept a fragmented packet that, after reassembly, is as large as 1500 octets.	5		X					X		
A node is permitted to accept fragmented packets that reassemble to more than 1500 octets.	5				X				X	

An upper-layer protocol or application that depends on IPv6 fragmentation to send packets larger than the MTU of a path should not send packets larger than 1500 octets unless it has assurance that the destination is capable of reassembling packets of that larger size.	5				X			X	
In response to an IPv6 packet that is sent to an IPv4 destination (i.e., a packet that undergoes translation from IPv6 to IPv4), the originating IPv6 node may receive an ICMP Packet Too Big message reporting a Next-Hop MTU less than 1280. In that case, the IPv6 node is not required to reduce the size of subsequent packets to less than 1280, but must include a Fragment header in those packets so that the IPv6-to-IPv4 translating router can obtain a suitable Identification value to use in resulting IPv4 fragments. Note that this means the payload may have to be reduced to 1232 octets (1280 minus 40 for the IPv6 header and 8 for the Fragment header), and smaller still if additional extension headers are used.	5		X				X		Requirement: if the IPv6 packet goes through an IPv6->IPv4 translation and the packet is too large (receives an ICMP Packet Too Big), the IPv6 source must include a Fragment header.
6. Flow Labels									
The 20-bit Flow Label field in the IPv6 header may be used by a source to label sequences of packets for which it requests special handling by the IPv6 routers, such as non-default quality of service or real-time service. This aspect of IPv6 is, at the time of writing, still experimental and subject to change as the requirements for flow support in the Internet become clearer.	6			X			X		
Hosts or routers that do not support the functions of the Flow Label field are required to set the field to zero when originating a packet, pass the field on unchanged when forwarding a packet, and ignore the field when receiving a packet.	6		X			X			
Appendix A describes the current intended semantics and usage of the Flow Label field.									
7. Traffic Classes									

The 8-bit Traffic Class field in the IPv6 header is available for use by originating nodes and/or forwarding routers to identify and distinguish between different classes or priorities of IPv6 packets. At the point in time at which this specification is being written there are a number of experiments underway in the use of the IPv4 Type of Service and/or Precedence bits to provide various forms of differentiated service for IP packets, other than through the use of explicit flow set-up. The Traffic Class field in the IPv6 header is intended to allow similar functionality to be supported in IPv6. It is hoped that those experiments will eventually lead to agreement on what sorts of traffic classifications are most useful for IP packets. Detailed definitions of the syntax and semantics of all or some of the IPv6 Traffic Class bits, whether experimental or intended for eventual standardization, are to be provided in separate documents. The following general requirements apply to the Traffic Class field:										
The service interface to the IPv6 service within a node must provide a means for an upper-layer protocol to supply the value of the Traffic Class bits in packets originated by that upper-layer protocol. The default value must be zero for all 8 bits.	7		X					X			IPv4 equivalent profiling changes this to a "may".
Nodes that support a specific (experimental or eventual standard) use of some or all of the Traffic Class bits are permitted to change the value of those bits in packets that they originate, forward, or receive, as required for that specific use. Nodes should ignore and leave unchanged any bits of the Traffic Class field for which they do not support a specific use.	7			X					X		
An upper-layer protocol must not assume that the value of the Traffic Class bits in a received packet are the same as the value sent by the packet's source.	7					X				X	
8. Upper-Layer Protocol Issues											
8.1 Upper-Layer Checksums											

Any transport or other upper-layer protocol that includes the addresses from the IP header in its checksum computation must be modified for use over IPv6, to include the 128-bit IPv6 addresses instead of 32-bit IPv4 addresses. In particular, the following illustration shows the TCP and UDP pseudo-header for IPv6: If the IPv6 packet contains a Routing header, the Destination Address used in the pseudo-header is that of the final destination. At the originating node, that address will be in the last element of the Routing header; at the recipient(s) that address will be in the Destination Address field of the IPv6 header. The Next Header value in the pseudo-header identifies the upper-layer protocol (e,g,, 6 for TCP, or 17 for UDP). It will differ from the Next Header value in the IPv6 header if there are extension headers between the IPv6 header and the upper-layer header. The Upper-Layer Packet Length in the pseudo-header is the length of the upper-layer header and data (e.g., TCP header plus TCP data). Some upper-layer protocols carry their own length information (e.g., the Length field in the UDP header); for such protocols, that is the length used in the pseudo-header. Other protocols (such as TCP) do not carry their own length information, in which case the length used in the pseudo-header is the Payload Length from the IPv6 header, minus the length of any extension headers present between the IPv6 header and the upper-layer header.										Implementation advice.
Unlike IPv4, when UDP packets are originated by an IPv6 node the UDP checksum is not optional. That is, whenever originating a UDP packet, an IPv6 node must compute a UDP checksum over the packet and the pseudo-header, and, if that computation yields a result of zero, it must be changed to hex FFFF for placement in the UDP header. IPv6 receivers must discard UDP packets containing a zero checksum, and should log the error.	8.1	X				X				

The IPv6 version of ICMP [ICMPv6] includes the above pseudo-header in its checksum computation; this is a change from the IPv4 version of ICMP, which does not include a pseudo-header in its checksum. The reason for the change is to protect ICMP from misdelivery or corruption of those fields of the IPv6 header on which it depends which, unlike IPv4, are not covered by an internet-layer checksum. The Next Header field in the pseudo-header for ICMP contains the value 58, which identifies the IPv6 version of ICMP.

8.2 Maximum Packet Lifetime

Unlike IPv4, IPv6 nodes are not required to enforce maximum packet lifetime. That is the reason the IPv4 Time to Live field was renamed Hop Limit in IPv6. In practice, very few, if any, IPv4 implementations conform to the requirement that they limit packet lifetime, so this is not a change in practice. Any upper-layer protocol that relies on the internet layer (whether IPv4 or IPv6) to limit packet lifetime ought to be upgraded to provide its own mechanisms for detecting and discarding obsolete packets.

8.3 Maximum Upper-Layer Payload Size

When computing the maximum payload size available for upper-layer data, an upper-layer protocol must take into account the larger size of the IPv6 header relative to the IPv4 header. For example, in IPv4, TCP's MSS option is computed as the maximum packet size (a default value or a value learned through Path MTU Discovery) minus 40 octets (20 octets for the minimum-length IPv4 header and 20 octets for the minimum-length TCP header). When using TCP over IPv6, the MSS must be computed as the maximum packet size minus 60 octets because the minimum-length IPv6 header (i.e., an IPv6 header with no extension headers) is 20 octets longer than a minimum-length IPv4 header.										Implementation advice.
8.4 Responding to Packets Carrying Routing Headers										
When an upper-layer protocol sends one or more packets in response to a received packet that included a Routing header, the response packet(s) must not include a Routing header that was automatically derived by reversing the received Routing header UNLESS the integrity and authenticity of the received Source Address and Routing header have been verified (e.g., via the use of an Authentication header in the received packet). In other words, only the following kinds of packets are permitted in response to a received packet bearing a Routing header: Response packets that do not carry Routing headers. Response packets that carry Routing headers that were NOT derived by reversing the Routing header of the received packet (for example, a Routing header supplied by local configuration). Response packets that carry Routing headers that were derived by reversing the Routing header of the received packet IF AND ONLY IF the integrity and authenticity of the Source Address and Routing header from the received packet have been verified by the responder.	8.4				X		X		Note the "UNLESS" clause.
Appendix A. Semantics and Usage of the Flow Label Field										

ATTACHMENT 4-3

A flow is a sequence of packets sent from a particular source to a particular (unicast or multicast) destination for which the source desires special handling by the intervening routers. The nature of that special handling might be conveyed to the routers by a control protocol, such as a resource reservation protocol, or by information within the flow's packets themselves, e.g., in a hop-by-hop option. The details of such control protocols or options are beyond the scope of this document.

There may be multiple active flows from a source to a destination, as well as traffic that is not associated with any flow. A flow is uniquely identified by the combination of a source address and a non-zero flow label. Packets that do not belong to a flow carry a flow label of zero.

A flow label is assigned to a flow by the flow's source node. New flow labels must be chosen (pseudo-)randomly and uniformly from the range 1 to FFFFF hex. The purpose of the random allocation is to make any set of bits within the Flow Label field suitable for use as a hash key by routers, for looking up the state associated with the flow.

All packets belonging to the same flow must be sent with the same source address, destination address, and flow label. If any of those packets includes a Hop-by-Hop Options header, then they all must be originated with the same Hop-by-Hop Options header contents (excluding the Next Header field of the Hop-by-Hop Options header). If any of those packets includes a Routing header, then they all must be originated with the same contents in all extension headers up to and including the Routing header (excluding the Next Header field in the Routing header). The routers or destinations are permitted, but not required, to verify that these conditions are satisfied. If a violation is detected, it should be reported to the source by an ICMP Parameter Problem message, Code 0, pointing to the high-order octet of the Flow Label field (i,e,, offset 1 within the IPv6 packet).

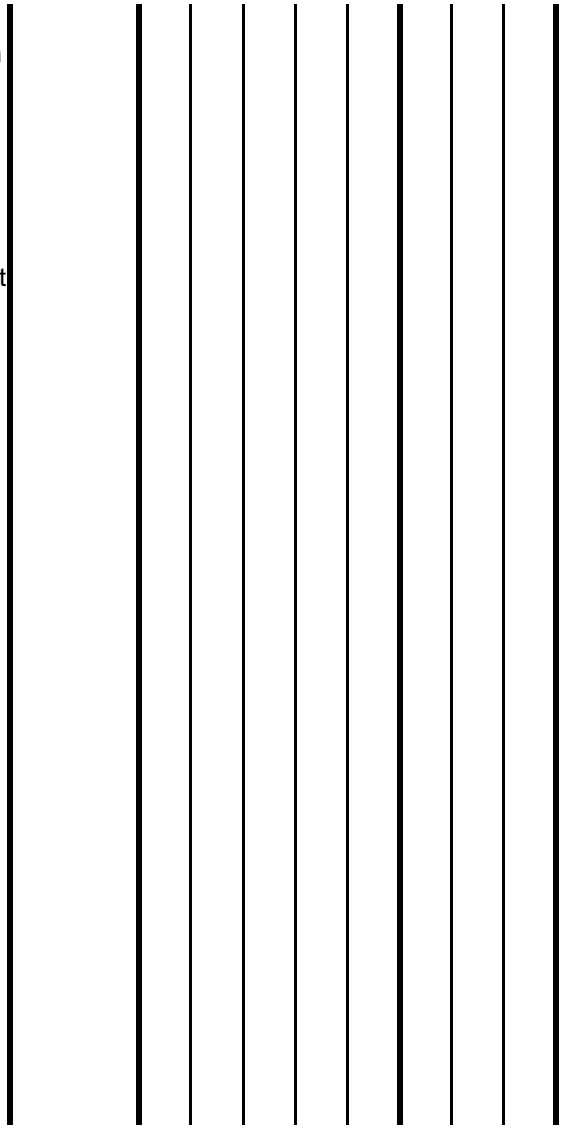
The maximum lifetime of any flow-handling state established along a flow's path must be specified as part of the description of the state-establishment mechanism, e.g., the resource reservation protocol or the flow-setup hop-by-hop option. A source must not re-use a flow label for a new flow within the maximum lifetime of any flow-handling state that might have been established for the prior use of that flow label.

When a node stops and restarts (e,g,, as a result of a crash), it must be careful not to use a flow label that it might have used for an earlier flow whose lifetime may not have expired yet. This may be accomplished by recording flow label usage on stable storage so that it can be remembered across crashes, or by refraining from using any flow labels until the maximum lifetime of any possible previously established flows has expired. If the minimum time for rebooting the node is known, that time can be deducted from the necessary waiting period before starting to allocate flow labels.

There is no requirement that all, or even most, packets belong to flows, i.e., carry non-zero flow labels. This observation is placed here to remind protocol designers and implementors not to assume otherwise. For example, it would be unwise to design a router whose performance would be adequate only if most packets belonged to flows or to design a header compression scheme that only worked on packets that belonged to flows.

Appendix B. Formatting Guidelines for Options

This appendix gives some advice on how to lay out the fields when designing new options to be used in the Hop-by-Hop Options header or the Destination Options header, as described in section 4.2. These guidelines are based on the following assumptions:



ATTACHMENT 4-3

ATTACHMENT 4-3

One desirable feature is that any multi-octet fields within the Option Data area of an option be aligned on their natural boundaries, i.e., fields of width n octets should be placed at an integer multiple of n octets from the start of the Hop-by-Hop or Destination Options header, for n = 1, 2, 4, or 8.

Another desirable feature is that the Hop-by-Hop or Destination Options header take up as little space as possible, subject to the requirement that the header be an integer multiple of 8 octets long.

It may be assumed that, when either of the option-bearing headers are present, they carry a very small number of options usually only one.

These assumptions suggest the following approach to laying out the fields of an option: order the fields from smallest to largest, with no interior padding, then derive the alignment requirement for the entire option based on the alignment requirement of the largest field (up to a maximum alignment of 8 octets). This approach is illustrated in the following examples:

ATTACHMENT 5-1

Subject: Working Paper for Project Paper 664, Part 5 - Aircraft Data Network Reference Model
Author: Bob Stephens – Boeing
Date: 12 June 2002 Revision 1

Introduction

This document presents a Reference Model for the Aircraft Data Network. The Aircraft Data Network, as covered in this document, is the network and sub-networks onboard the aircraft including the connections offboard the aircraft (the radio links) that support the onboard networks. The domains of network users that are covered in this document are:
Avionics, Crew, In Flight Entertainment (IFE), and Passenger Personal Electronic Devices (PEDs).

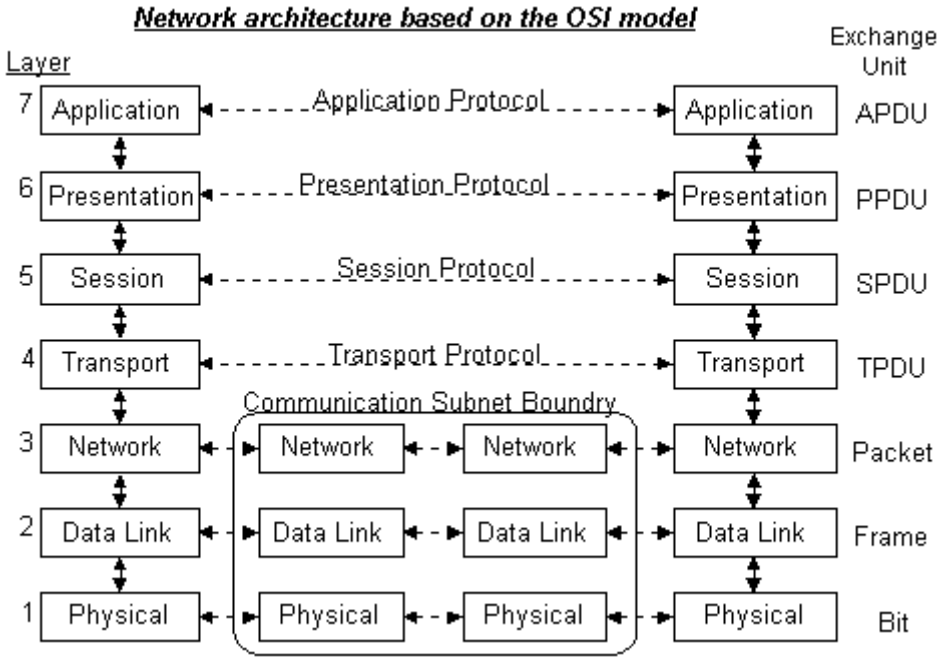
A Reference Model shows the functional elements and their interconnection. The functional elements represent a specific function or collection of functions and should not be taken to identify physical devices. Individual functional elements are shown even when multiple physical devices of the same function are required in the network; multiple functional elements are shown when necessary to present the interconnection scheme.

This document begins with a brief description of the OSI Reference Model followed by a table of the Functional Elements used in the Aircraft Data Network Reference Model. The Aircraft Data Network Reference Model is presented last.

ATTACHMENT 5-1

OSI Reference Model

The seven layer OSI Reference Model is used in describing the Aircraft Data Network Reference Model functional elements.



The seven layers of the OSI Reference Model are defined as follows:

7) **Application** : Provides different services to the applications
6) **Presentation** : Converts the information (usually null in IP)
5) **Session** : Handles problems which are not communication issues (usually null in IP)
4) **Transport** : Provides end to end communication control (e.g. TCP or UDP)
3) **Network** : Routes the information in the network (e.g. IP)
2) **Data Link** : Provides error control between adjacent nodes (e.g. 802.2 or null)
1) **Physical** : Connects the entity to the transmission media (e.g. 802.3 Ethernet)

Note: In the above diagram TPDU/SPDU/SPDU/APDU means Transport, Session, Presentation or Application Protocol Data Unit. These are the units of transmission of data as viewed by the respective layer.

ATTACHMENT 5-1

Functional Elements of the Aircraft Data Network

The following describe the functional elements used in the Aircraft Data Network Reference Model. Some elements are defined for future use and are not found in the current Reference Model.

Radio	The function of a Radio is to provide Physical and Link Layer transportation of frames on and off the aircraft.
Host	The function of a Host is to execute client and/or server applications.
Switch	The function of a Switch is to provide Link Layer interconnection of other networks or devices. All components that connect to a switch are of the same Link Layer technology (e.g. 802.3 Ethernet).
Bridge	The function of a Bridge is to provide Link Layer connectivity between networks or devices of the same of different Link Layer technology. Examples are an 802.3 (Ethernet)-to-802.5 (Token Ring) bridge and an 802.11 Access Point that connects to 802.3 (Ethernet).
VLAN Switch	The function of a VLAN Switch is to incorporate the functions of a Switch in addition to Virtual LAN technology. In one configuration a VLAN Switch will allow communications from all “client” ports through a single “trunk” port but will prevent communication between client ports.
Router	The function of Router is to provide communication between distinct subnets (A subnet is a group of contiguous IP addresses). The Router function also provides the ability to connect subnets of different Link Layer technologies (e.g. 802.3 Ethernet and Radio).
NAT Router	The function of a Network Address Translation (NAT) Router is to provide the function of Network Address Translation in addition to those of Router. The NAT Router function can translate IP addresses of packets as they traverse the Router. The NAT Router can also map multiple IP addresses from an interface to one or a few IP addresses on another interface by modifying and mapping the Transport Layer port numbers or another available protocol-specific index. This is sometimes referred to as a Network Port Address Translation (NPAT) Router.
Packet Filter Firewall	The function of a Packet Filter Firewall is to enforce network access based on protocol information in each IP packet. When an IP packet arrives at the firewall, the protocol information is compared to a collection of filtering rules. These rules specify the conditions under which packets should be passed through or denied access (discarded).

ATTACHMENT 5-1

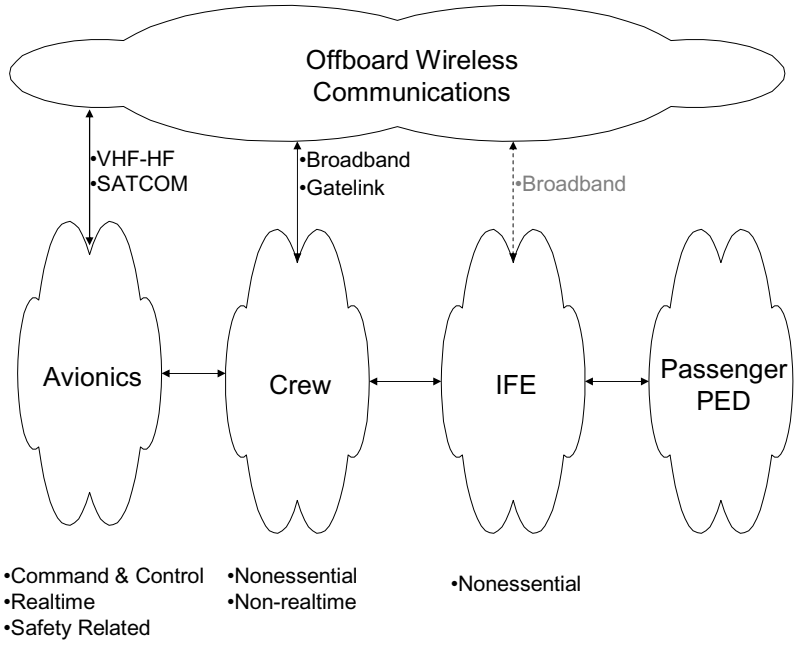
Session Filter Firewall	The function of a Session Filter Firewall is to incorporate all of the function of a Packet Filter Firewall and to determine and retain information on all active network sessions through it. This information is used to determine whether subsequent packets and packets flowing in the opposite direction belong to an approved connection. This function is sometimes referred to as a Smart Packet Filter or Stateful Packet Filter Firewall. By eliminating static configuration entries for returning packets security is enhanced. (The TCP protocol identifies the first and last packets of a session. For other protocols, such as UDP, which are not session oriented, the first packet may start a session and an elapsed time with no activity may be deemed to end the session.)
Application Proxy Firewall	The function of an Application Proxy Firewall is to terminate all network connections and, if a requested access is authorized, create a separate connection to the desired destination. It then shuttles information between the original connection and the second connection. No IP packets are passed directly between the two networks. Proxy applications must be provided for each supported service.
Packet Filter Switch	The function of a Packet Filter Switch is to combine the functions of a Switch and a Packet Filter Firewall.
VPN Access Server	The function of a VPN Access Server is to allow IP packet-level connectivity to a firewall-protected or NAT-protected network from outside authenticated hosts. Typical implementations support PPTP and/or L2TP protocols. After authentication, the outside host is provided an IP address belonging to the protected subnet. The outside host creates IP packets using this address and securely tunnels the packets to the Remote Access Service which acts as the network attachment on its behalf. The secure tunnel provides packet integrity and optional confidentiality over an authenticated session.
Message Gateway Host	The function of a Message Gateway Host is to communicate with other onboard Hosts and provide Application Layer message exchanges with offboard Hosts. It may interface with Radios using only Link Layer protocols.
Airframe Host	The function of an Airframe Host is to execute crew applications outside of the avionics area that are provided by the airframe supplier.
Airline Host	The function of an Airline Host is to execute crew applications outside of the avionics that are provided by the airline or aircraft owner/operator.
Network Management Host	The function of the Network Management Host is to execute applications to support network fault management, configuration, accounting, performance monitoring and security.
Dynamic Address Assignment (DHCP)	The function of Dynamic Address Assignment (e.g. DHCP – Dynamic Host Configuration Protocol) is to dynamically assign IP address to Hosts, typically on the Passenger PED, IFE or Airline Host subnets.
Domain Name Server	The function of the Domain Name Server (DNS) is to provide a database of name to IP address mapping for other applications or network components.

ATTACHMENT 5-1

Reference Model for the Aircraft Data Network

Domains of the Aircraft Data Network

The following diagram defines the domains for the Aircraft Data Network.



Avionics are the electronic and electromechanical subsystems and systems installed in an aircraft or attached to it. It does not include the power generation and distribution systems. Avionics includes the Command and Control functions of the aircraft. It is realtime in nature and safety related. Typical avionics applications include: Navigation and guidance, Communications, Surveillance, Flight Controls, Mission avionics (offensive and defensive), and Vehicle and utility management systems.

Crew applications are the nonessential functions supporting the flight deck and cabin crew. They can be subdivided into those applications provided by the airframe manufacturer and those applications provided by the airline.

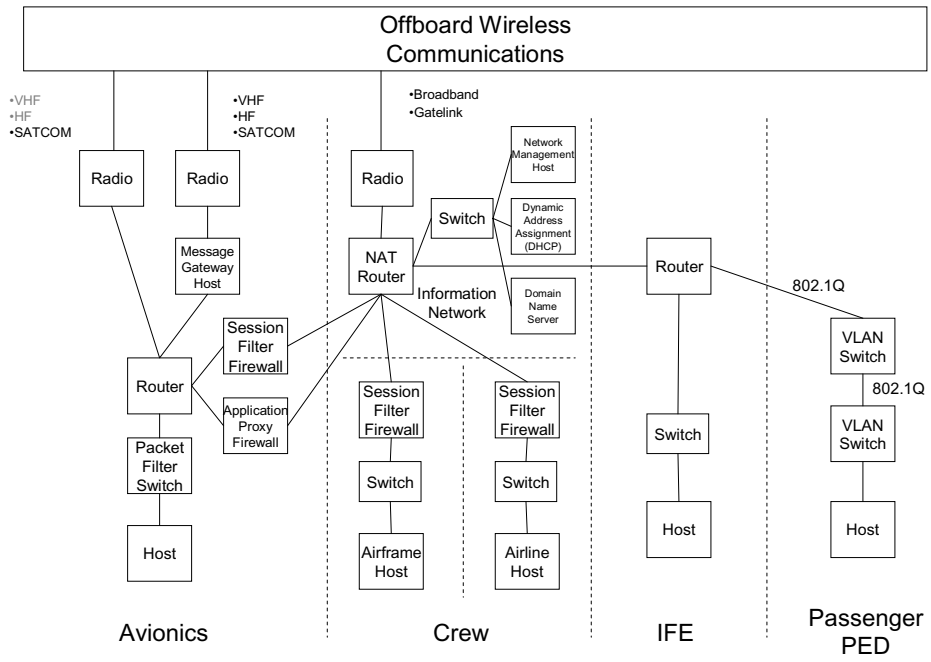
In Flight Entertainment (IFE) is the network supporting passenger entertainment. It also includes the Cabin Distribution System (CDS) used to connect Passenger Personal Electronic Devices (PEDs).

The Passenger Personal Electronic Devices (PEDs) is the collection of passenger provided electronic devices.

ATTACHMENT 5-1

Reference Model Details

The following is the Aircraft Data Network Reference Model.



The Avionics network has connectivity offboard at the network Transport Layer (Radio to Router connection) and at the Application (Message) Layer (through the Message Gateway Host). Native IP applications will communicate using the Transport Layer. The Message facility supports existing ACARS messaging. The Avionics network is connected to Information Network through a Session Filter Firewall to support access for avionics applications to communicate through the offboard broadband or Gatelink connections. No inbound connections are allowed though this firewall. The Application Proxy Firewall allows access from non-avionics applications to restricted read-only functions on the Avionics network. The special Packet Filter Switch supports pre-defined communications among the components (hosts) on the Avionics network.

The Crew network is divided into three sections. The Airframer-provided hosts are protected by a firewall. The Airline-provided hosts are separately protected by a firewall. The Information Network provides the broadband and Gatelink connections offboard the aircraft. It also provides the internal support functions for Network Management, Dynamic Address Assignment and Domain Name Service. The Information Network requires no internal firewalls since its services are available to the other networks. The NAT router provides address translation (which is a form of firewall) offboard the aircraft. The router supports prioritization of transmissions for Quality of Service to the other networks.

The In Flight Entertainment network provides for passenger access to entertainment services. The Passenger PED network allows the attachment of Passenger Personal Electronic Devices. It is structured using VLAN Switches which are configured to disallow communications between the user ports, while allow traffic up through the router trunk. Passenger-to-passenger connections could be achieved using private VPN connections to a common corporate network.

ATTACHMENT 5-2

AERONAUTICAL RADIO, INC.
2551 Riva Road
Annapolis, Maryland 24101-7465

PROPOSED TEXT

FOR GROUP DISCUSSION IN MUNICH

OF

ARINC PROJECT PAPER 664

AIRCRAFT DATA NETWORKS

PART 5

NETWORK INTERCONNECTION SERVICES AND FUNCTIONAL ELELMENTS

This text dated: June 2002

ATTACHMENT 5-2

1.0 INTRODUCTION

1.1 Purpose of Document

The purpose of ARINC Specification 664, Part 5 is to provide general and specific design and implementation guidelines for the interconnection of IEEE 802.3 devices and networks installed in aircraft

In ground based networks, the interconnection of IEEE 802.3 devices and networks is usually discussed and specified in terms of interconnection devices such as repeaters, switches/bridges, routers, gateways and firewalls which are further specified by IEEE standards. The IEEE standards define a set of characteristics and behaviors that, when implemented by a device and properly configured by a network administrator, provide a certain degree of connectivity and security. But the IEEE standards, by and large, do not specify the form, fit and functionality of the individual devices.

In contrast ARINC standards, especially ARINC 500/700 series standards, do specify the form, fit and minimum functionality of the individual "devices" allowing the airline to provision both the wiring and physical mounting/hold-down of the device without regard to the manufacturer.

Typically, it is the ARINC 600 series specifications that provide the framework and detail necessary for the interconnection and communication between ARINC devices and when required, with off-aircraft devices. The purpose of this document is to provide that framework and detail for airborne IEEE 802.3 ARINC devices and networks.

1.2 Scope

The scope of Part 5 is to provide general and specific design and implementation guidelines that when implemented by on-aircraft ARINC devices and configured by the system integrator, provide the required connectivity and security. The form, fit and function specification of ARINC "devices" is out-of-scope for this document and will be provided by other ARINC specifications.

Consequently, this document will not discuss bridges, switches, routers, gateways or firewalls except in a general sense. Instead this document will discuss guidelines and objectives for switching, routing, protocol conversion and network security.

As noted in ARINC 664, Parts 1 and 3, IEEE 802.3 networks that provide critical and essential services for the continued safe flight of the aircraft have tight regulatory oversight and control. These networks are typically "profiled" to provide the characteristics necessary for the avionics domain and depart from full compliance to IEEE specification.

So one question is: Is the interconnection of the individual devices <u>within</u> the profiled network IN or OUT of scope for this document???

1.3.1 Document Organization

1.3.1 ARINC Specification 664, Aircraft Data Network

ARINC Specification 664 specifies an Ethernet Data Network for aircraft installation. It is developed in multiple parts, listed as follows:

Part 1 - Systems Concepts and Overview
Part 2 - Ethernet Physical and Data Link Layer Specifications
Part 3 - Internet-based Protocols and Services
Part 4 - Internet-based Address Structures and Assigned Numbers
Part 5 - Network Interconnection Services and Functional Elements
Part 6 - Network Management Specification
Part 7 - An Example Implementation of a Deterministic Network
Part 8 – Upper Layer Services for Aeronautical Telecommunication Network (ATN) and Airline Operational Control (AOC)

1.3.2 Part 5, Network Interconnection Devices

Part 5 is organized into sections as follows:

Section 1.0, Introduction

Section 2.0, Overview

Section 3.0, Description of Services

Section 3.1, Introduction

Section 3.2, Security

Section 3.3, Quality

Section 3.4, Mobility

Section 3.5, Transport of Data

Section 3.6, Network Management

Section 3.7, Directory Services

Section 4.0, Functional Elements

Section 4.1, Description of Functional Elements

Section 4.2, Mapping of Services to Functional Elements

ATTACHMENT 5-2

1.0 INTRODUCTION

1.4 Related Documents

1.4.1 Relationship of this Document to Other Standards

The following documents are considered as an integral part of ARINC Specification 664 and constitute the core of the specification. These documents are required for a complete specification.

ANSI/IEEE Std 802.3, 1998 Edition:	Information Technology - telecommunications and information exchange between systems - Local and metropolitan area networks - Specific Requirements - Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method And Physical Layer Specifications
ANSI/IEEE Std 802.1D [ISO/IEC 15802-3]	Media Access Control (MAC) Bridges. Specifies an architecture and protocol for the interconnection of IEEE 802 LANs below the MAC service boundary.

COMMENTARY

ISO and ISO/IEC documents are available from ISO Central Secretariat, 1 rue de Varembe, Case Postale 56, CH-1211, Geneve 20, Switzerland/Suisse; and from the sales department, American National Standards Institute, 1430 Broadway, New York, NY 10018, USA.

IEEE documents are available from the Service Center, Institute of Electrical and Electronic Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331.

The following is a list of other documents that are related to ARINC Specification 664, although they are not considered an integral part of the specification:

ARINC Specification 600	Air Transport Avionics Equipment Interfaces
ARINC Specification 637A	xxx
ARINC Specification 638A	xxx
ISO/IEC 8802-2, 1998, ANSI/IEEE Std 802.2, 1998 edition	Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 2
Logical Link Control	
IEEE Std 802-1990	IEEE Standards for Local and Metropolitan Area Networks Overview and Architecture
ISO 7498	Information Processing Systems - Open Systems Interconnection - Basic Reference Model
ISO 7498-3	Information Processing Systems - Open Systems Interconnection - Basic Reference Model - Part 3, Naming and Addressing
ISO/IEC TR 9577	Information Technology - Telecommunications and Information Exchange Between Systems - Protocol Identification in the Network Layer
ISO/IEC TR 10178	1991, Information Processing Systems - Telecommunications and Information Exchange Between Systems - The Structure and Coding of Logical Link Control Addresses in Local Area Networks
ISO/DIS 10165-4	Structure of Management Information - Part 4 Guidelines for the Definition of Managed Objects
RTCA DO-160/EUROCAE ED-14	Environmental Conditions and Test Procedures for Airborne Equipment
RTCA DO-178/EUROCAE ED-12	Software Considerations in Airborne Systems and Equipment Certification
RTCA DO-205	Design Guidelines and Recommended Standards to Support Open Systems Interconnection for Aeronautical Digital Communications - Part 1, Internetworking
EIA/TIA 568 TBS 36	Category 5 twisted pair cable specification
ISO/IEC 9314	FDDI Physical Layers
ANSI X3T12	FDDI Physical Layers

ATTACHMENT 5-2

1.0 INTRODUCTION

COMMENTARY

RTCA Documents are available from RTCA, Inc., 1140 Connecticut Avenue, NW, Suite 1020, Washington, D.C. 20036

Environmental Factors for the network interconnection devices defined in RTCA DO-160/EUROCAE ED-14, "Environmental Conditions and Test Procedures for Airborne Equipment" and other appropriate airframe and regulatory agency documents.

ATTACHMENT 5-2

2.0 OVERVIEW

There has been an increasing demand for commercial aircraft to provide integrated network computing capabilities for passengers traveling, as well as for the people working aboard and around the aircraft. An airplane computing network is needed which can be inter-connected to ground-based computing networks such as the Internet, airline LANs/WANs, and other terrestrial networks.

The network reference model that will be used for the airplane computing network discussion is shown in the following figure. While an almost infinite number of network configurations can be created, they all should share the network interconnection services, properties and characteristics as the network reference model.

The network reference model is characterized by 4 domains, which were chosen to delineate the safety and security characteristics. Each of domains require connectivity with ground based computing networks however the physical connectivity (which domain contains the communication device or unit) is not defined in this specification. The Avionics domain uniquely connects to high-priority ATC and some AOC communication, but devices providing these air-ground links could still be accessed on a priority basis by the other domains.

The characteristics of each domain will be discussed individually and then compared with characteristics of the other domains in a matrix..

This document describes the objectives, characteristics, services and functions, that must be satisfied and provided by the airplane computing network as a whole. Their allocation to specific networks and "devices" is left to airplane network designers and other specifications.

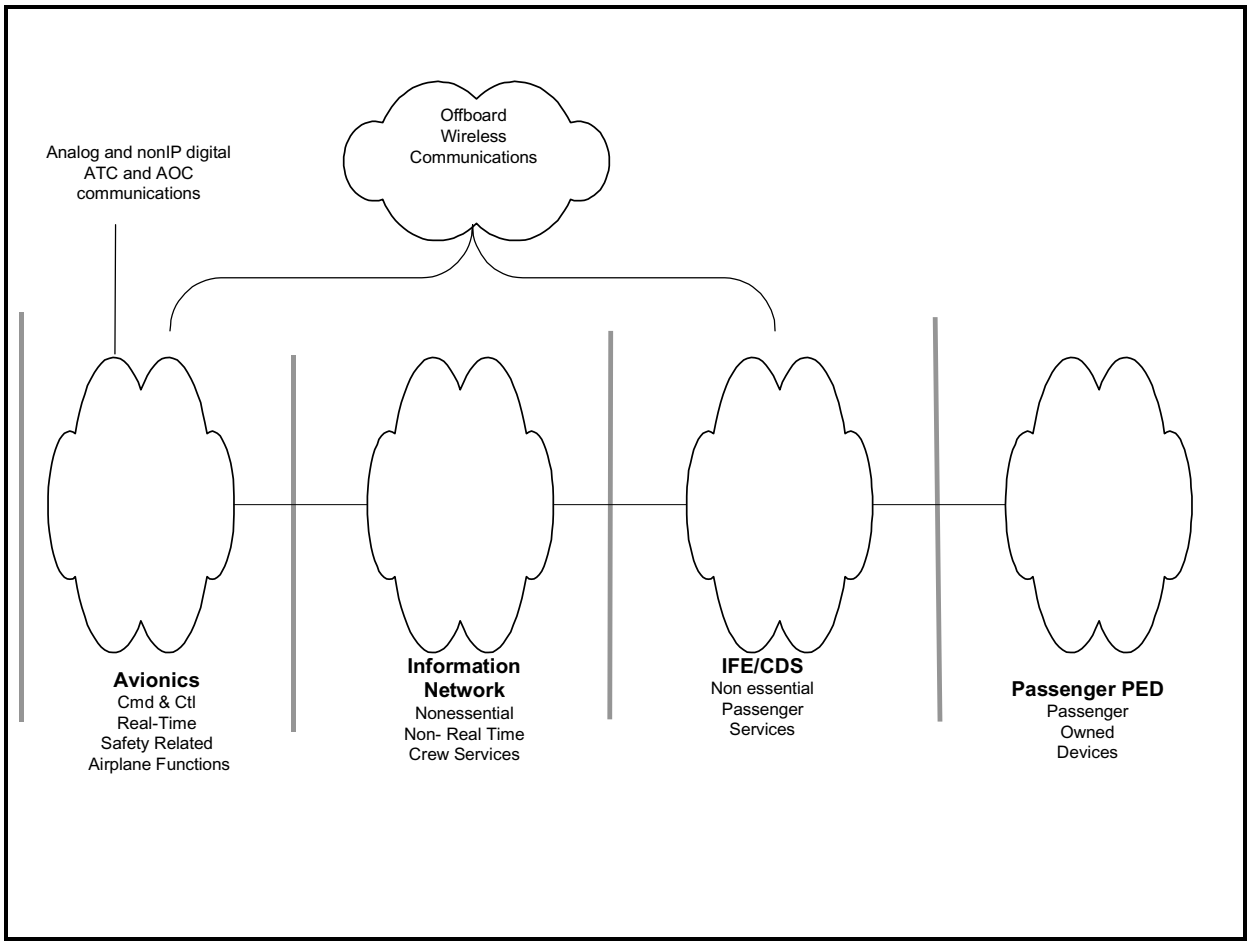


Figure 1 Airplane Network Reference Model

2.0 OVERVIEW

2.1 Avionics Domain

The Avionics domain[1] consists of systems and networks whose primary functions support the safe operation of the aircraft. The justification for most of these systems is traceable to safety of flight. When these systems perform non-safety related functions, it must generally be shown that no interference with safety related functions is possible.

Off-board communications for the Avionics domain aligns with the safety related characteristics of the domain in general. ATC & some AOC communication is considered high priority and other uses are based on non-interference. Currently Avionics off-board communication links are almost exclusively either analog or non-IP digital. However, an off-board IP link is a reasonable possibility in a future airborne network architecture.

A complicating factor for Avionics is that, while all air transport aircraft can be assumed to have an "Avionics domain", there is a tremendous variety of systems and network architectures used in Avionics. This means that characteristics internal to the domain can only be described in general terms. With appropriate assumptions (such as SATCOM with IP datachannels or the existence of an ARINC 763 Server Interface Unit (SIU)), characteristics of data flows in and out of the domain can be described in more detail but the specific implementation and network capacity will of necessity vary widely depending on the aircraft model and specific configuration.

While the Information Service domain is relatively new and has little fleet penetration, and IFE systems are typically updated and even replaced over time, Avionics systems designs change relatively slowly and wholesale replacement with a completely new system is extremely rare. This must be kept in mind when looking at fleetwide implementations of new functionality.

The fundamental principle for general IP interfaces with Avionics is that non-interference with safety related functions must be shown for any implementation. This includes safety-related communications functions. For the overwhelming majority of Avionics systems today, this means that Avionics systems will interface with IP networks only at the perimeter of the domain, and must either provide a robust partition which prevents interference in shared transport services (such as SATCOM and the Cabin Telephone Unit) or assure that dataflows are appropriately controlled (such as the SIU is designed to do).

Examples of systems in the Avionics domain include:

- Cockpit Displays
- Flight Controls
- Environmental Controls
- Electrical System
- Propulsion Systems
- Cabin Services System
- Flight Recorder System

Characteristics of such systems result in much higher cost compared to non-airborne systems of comparable complexity due in large part to requirements for:

- High reliability in a harsh environment
- Highly regulated business environment
- Certification level from critical to non-essential (software DO178B level A to E)
- Robust partitioning
- Rigorous process and documentation requirements
- Small production volume

Which result in:

- Custom design
- Low network bandwidth requirements
- "Closed" network
- Relatively slow rate of change
- Network "peers" are all known
- Connection oriented communication between peers

Because airplane networks are contained in mobile vehicles, access tends to be physically enforced with little or no computing security (at least in the information systems sense), and remote access is typically not available except for some relatively minor monitoring capabilities. The aircraft is generally flying or on the ground with limited physical access.

As embedded systems with an overall focus on real-time control, these systems are well characterized with respect to:

- Real-time behavior
- Latency / Jitter
- Functional availability
- Functional integrity

[1] This discussion is primarily focused on digital, and more specifically, IP data and networks. Analog communications and non-IP digital networks are out of scope except for discussions of protocol conversion.

2.0 OVERVIEW

2.2 Information Services Domain

The Information Services Domain (ISD) provides services and connectivity between independent aircraft domains such as avionics, in-flight entertainment, cabin distribution and any connected off-board networks. The ISD provides a multi-layered aircraft network security perimeter, incorporating network routing and security functions/services between ISD and less critical aircraft network domains as well as security functions/services between the more critical avionics domain and any connected external wireless networks.

The ISD provides general purpose switching, routing, computing, data storage and communications services for non-essential applications. The ISD may be comprised of one or more computing platforms for third party applications and content. ISD platforms may be used to support applications and content for either cabin or flight crew use.

The physical configuration of the ISD network on a given aircraft may vary based on network segregation, off-aircraft connectivity and airline functional requirements. Airline and airframe-defined operational requirements for functional availability will determine equipment and service redundancy requirements within the ISD.

Given that the ISD architecture may vary between aircraft types and airline operational requirements, the ISD must be defined based on open computing and commercial networking definitions to standardize its network environment. The ISD provides shared network services and resources for use by other sub-systems. Common network services and network management are required to enable use of common applications across mixed aircraft fleets.

ISD platforms may support applications that interface with avionics systems. Avionics systems may access mass storage devices in the ISD. ISD hosted applications may have limited communications with avionics systems. ISD platforms should support the distribution and storage of specified avionics data. Typical examples ISD avionics interface applications include ARINC 615-3 and 615A Data Loader, Virtual Quick Access Recorder (VQAR) and Central Maintenance Functions.

When a dedicated off-board network connection for passenger use is connected to and managed within the ISD, the ISD should provide central security and routing services to transparently support multiple aircraft-ground connections.

ISD managed switching and routing services work in cooperation with the avionics domain security functions to provide a Demilitarized Zone (DMZ) between the aircraft avionics and cabin networks and external networks as part of a layered network security model.

ISD external network connection requirements include network resources and services shared by connected sub-systems. ISD external network may be shared as a possible path for off-board passenger communications/data transfer (pass-through). As such, the ISD should be capable of prioritizing network traffic. ISD off-board network connectivity should provide a common application interface and transparent message routing via one or more wireless solutions.

Examples of ISD services include:

- Network Management
- Firewall/Router Management
- Remote Message services
- Remote Network Maintenance
- Directory Name Servers (DNS)
- Network File Services,
- print queue services.

ATTACHMENT 5-2

2.3 In-flight Entertainment and Passenger Device Domain

This domain is characterized by the need to provide passenger entertainment and network services. An analogy used many times is that the airline passenger should be able to enjoy the same services as being in a hotel room. The functionality of this domain is the most dynamic in that IFE systems are typically replaced every 5 years. Also, the technology available to the passenger changes regularly. The passenger can be expected to carry-on increasingly sophisticated devices which, in the passengers mind, should work as well on the aircraft as they would in the hotel room.

Passenger applications provided by the IFE system include:

- Streaming Video
- Streaming Audio
- PAX internet surfing
- PFIS (moving maps)
- VOIP
- Gaming
- SMS short message services

Applications that support the IFE applications include:

- FTP
- BITE
- Network Management
- DHCP
- Transporting of VPN Encrypted Packets
- Dataload Operational Software
- Dataload Content

Of course the applications carried on board the passenger devices are limitless. These applications should be considered both benign and malicious. The impact of these applications include the following:

- Impact from PEDs
- Viruses
- Worms
- Malicious Code
- Spoofing
- Splicing
- Directed Denial of Services (DDOS) attacks.

The characteristics of the IFE network include:

- High Bandwidth
- High QOS (must contain latency jitter and provide certain applications near real time performance)
- Multiple Protocols
- Availability
- Level E Certification
- ?????????? are these IFE Net characteristics?
- *Multilevel Security/ Authentication*
- *Encapsulated security containment*
- *Integrity*

2.4 Wireless links

In addition to the airplane network reference domains, there are the offboard wireless communication links. Today, these include:

- VHF ACARS (2400 bps)
- SATCOM (9600 bps)
- VDLM2 (33k bps)
- SATCOM HSD (64-384k bps)
- Gatelink (2-11M and 55M bps)
- Onboard WLAN (2-11M and 55M bps)
- Broadband (>5M bps)

This document discusses the guidelines and objectives for communications across these links without regard to which domain the links are physically connected.

All domains may share these links. However when these links are used by the Avionics Domain for ATC and other high priority traffic, it will have to be shown that other usage does not interfere with the high priority traffic.

2.5 Network Domain Comparison

The following matrix compares the network characteristics of each domain. There are considerable differences in the characteristics of the domain networks and the reason for the separation into domains should be apparent. Connecting two domain networks inherently poses interfacing challenges to preserve the characteristics and not allow any degradation.

ATTACHMENT 5-2

2.0 OVERVIEW

Table 1 Domain Network Characteristic Comparison

Network Characteristics	Avionics Domain Networks	Information Services Networks	IFE Networks	Passenger Carry-On-Device Networks
Real-Time (bounded jitter and latency)	High (critical in the extreme)	Medium, for audio	Medium, for video and audio	No
Throughput Required	Low, unless cockpit audio is included	Medium, assuming cabin audio	High, especially for video	Low
Availability Required	High (critical in the extreme)	Medium	Medium, IFE can be required for dispatch	Low
Integrity Required	High (critical in the extreme)	Medium	Low	Normal Internet
Data Flow Partitioning within network	High (critical in the extreme)	Some possible	Some possible	Normal Internet
Configuration Type	Static, all network nodes are known, private addressing determined by aircraft system integrator	Static and Dynamic, Hand-held crew devices could be dynamic	Static and Dynamic	Normal Internet
Configuration Change Frequency	Slow to change, requires regulatory agency approval	Medium, new devices added and subtracted over time	High, Entire IFE systems change over time	Dynamic
Security Requirements	High (critical in the extreme)	Medium	Medium	Normal Internet
Protocol	Limited, controlled by system integrator	Multiple, controlled by aircraft system integrator	Multiple, controlled by IFE manufacturer	Normal Internet
Software Standard	DO-178 B Level A in the extreme	Less than DO-178B Level A	DO-178B Level E	Normal Internet
Physical Access	Limited to cockpit and equipment bay, difficult for passengers to gain access	Limited to crew stations, passenger could get access to crew stations	Limited passenger access. Passengers could force unauthorized access at their seat	Normal passenger access
Regulatory Agency Secutiny	High, difficult for airlines to make modifications	Medium, airlines can make modifications	Low, airlines can make modifications	Low

ATTACHMENT 5-3

Accelerating CNS

ADN 664, Part 5

Action Item Update:
Security and Quality of Service -
Functions & Network Elements Analysis

April 11-12, 2002

Presented by: Chris Wargo

Accelerating CNS

Agenda

- **Methodology**
- **Security**
  - **Functions Summary**
  - **Mapping to Functional Elements**
  - **Requirements Definition**
  - **Next Steps**
    - » Specification by Data Flows - Example
- **Quality of Service (QoS)**
  - **Functions**
- **Reference: Network Functional Elements Definition**

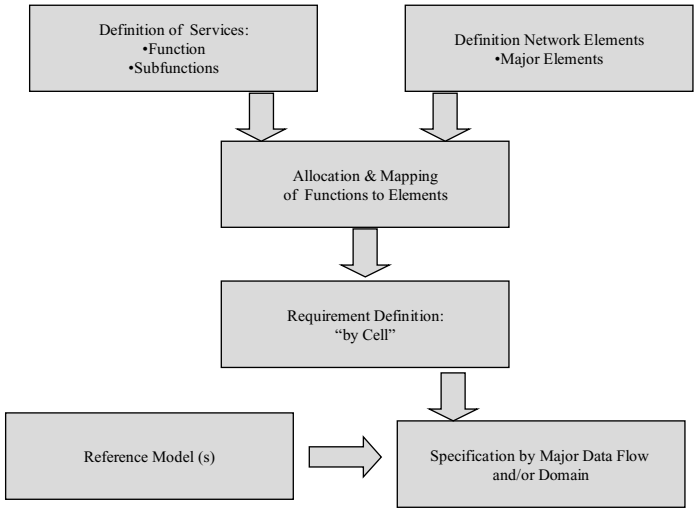
Prepared by: Crispin Netto and Bryan Myers, CNS as part NASA GRC Task Order

ATTACHMENT 5-3

Internetworking Architecture Analysis

CNS *Accelerating CNS*

Methodology



Network Security Functions

CNS *Accelerating CNS*

- F1: Authentication
- F2: Access
- F3: Data Confidentiality
- F4: Data Integrity
- F5: Non-Repudiation
- F7: Intrusion Protection Methods
- F8: Counter Measures
- F9: Recovery of System / Operation
- F10: Logging

ATTACHMENT 5-3

Network Security Sub-functions

Accelerating CNS

F1: Authentication

- §F1.1: Validity Checking
- §F1.2: Protection of Stored Validity Data
- §F1.3: Confidentiality of Data in Transit
- §F1.4: Additional Security Measures

F2: Access

- §F2.1: Access Control
- §F2.2: Access List Administration

F3: Data Confidentiality

- §F3.1: Encryption
- §F3.2: Key Distribution and Management
- §F3.3: Level of Security
- §F3.4: Layer of Encryption

Security Sub-functions (cont..)

Accelerating CNS

F4: Data Integrity

- §F4.1: Acceptable transmission error
- §F4.2: Anti-Spoofing / Message Digests

F5: Non-Repudiation

- §F5.1: Confirmation
- §F5.2: Retention of Confirmation

F7: Intrusion Protection Methods

- §F7.1: Bastion Host
- §F7.2: Filters
- §F7.3: Application Gateway (Proxy Server)
- §F7.4: Internal Domain Name Server (DNS)

ATTACHMENT 5-3

Security Sub-functions (cont..)

Accelerating CNS

F8: Counter Measures

- §F8.1: Protection
- §F8.2: Intrusion Detection
- §F8.3: Response

F9: Recovery of System / Operation

- § TBD
- § TBD

F10: Logging

- § TBD
- § TBD

Network Security Functional Elements

Accelerating CNS

	OSI Layer	Authentication	Access	Data Confidentiality	Data Integrity	Non-Repudiation	Intrusion Protection Methods	Counter Measures	Recovery of System / Operation	Logging
End System (or DTE)	7	l	l	l	l	l	l	l	l	l
Autoconfigure / Loader	7	-	-	-	-	-	-	-	-	-
Certification Authority	7	l	-	l	l	l	-	-	-	l
DHCP	7	-	-	-	l	-	-	-	-	¡
DNS	7	¡	-	-	l	-	l	-	-	¡
Network Management Station	7	l	l	l	l	l	-	-	-	l
Firewall	3	l	l	/	l	/	l	l	-	l
Gateway	3	/	l	/	l	/	l	l	-	¡
Router	3	/	l	/	l	/	l	l	-	¡
Access Point	2	/	/	l	/	/	/	-	-	-
Bridge (or Switch)	2	/	/	/	/	/	/	-	-	l
Backbone	1	¡	¡	/	/	/	/	-	-	-
Cable Plant	1	/	/	/	/	/	/	-	-	-
Repeater (or Hub)	1	/	/	/	/	/	/	-	-	-

Legend	Meaning
-	Not Applicable
¡	Optional
/	Present, but not required for a special task
l	Present, required for a special task

ATTACHMENT 5-3

Network Security Sub-functions - Authentication

CNS Accelerating CNS

F1: Authentication	*F1.1: Validity Checking*	*F1.2: Protection of Stored Validation Data*	*F1.3: Confidentiality of Data in Transit*	*F1.4: Additional Security Measures*
End System (or DTE)	l	l	l	i
Certification Authority	l	l	l	i
Network Management Station	l	l	l	i
Firewall	-	-	l	-

Legend	*Meaning*
-	Not Applicable
i	Optional
/	Present, but not required for a special task
l	Present, required for a special task

Network Security Sub-functions - Authentication

CNS Accelerating CNS

F1: Authentication	*F1.1: Validity Checking*	*F1.2: Protection of Stored Validation Data*	*F1.3: Confidentiality of Data in Transit*	*F1.4: Additional Security Measures*
End System (or DTE)	Shall require valid UserID/Password combination to access Network services.	May store passwords locally; if so, these passwords shall be stored in an encrypted format.	Shall encrypt sensitive information (e.g. passwords) before transmitting through the network.	May employ additional security measures (e.g. smart cards, single-use passwords).
Certification Authority	Shall validate credentials before performing services for a user.	May store passwords and private keys locally; if so, these shall be stored in an encrypted format.	Shall encrypt sensitive information (e.g. passwords, private keys) before transmitting through the network.	May employ additional security measures (e.g. smart cards, single use passwords).
Network Management Station	Shall require valid UserID/Password combination to access the system.	May store passwords locally; if so, these shall be stored in an encrypted format.	Shall encrypt sensitive information (e.g. passwords) before transmitting through the network.	May employ additional security measures (e.g. smart cards, single use passwords).
Firewall	-	-	Shall apply filters to prevent sensitive data from crossing into publicly accessible domains.	-

ATTACHMENT 5-3

Network Security Sub-functions - Access

Accelerating CNS

F2: Access	F2.1: Access Control	F2.2: Access List Administration
End System (or DTE)	l	l
Network Management Station	l	l
Firewall	l	-
Gateway	l	-
Router	l	-

Legend	Meaning
-	Not Applicable
i	Optional
/	Present, but not required for a special task
l	Present, required for a special task

Network Security Sub-functions - Access

Accelerating CNS

F2: Access	F2.1: Access Control	F2.2: Access List Administration
End System (or DTE)	Shall restrict access to system and network elements based on an access control list.	-
Network Management Station	Shall restrict access to system and network elements based on an access control list.	Shall maintain and distribute an access control list to connected network devices.
Firewall	Shall apply filters to limit access to network resources based on rules provided in an access control list.	-
Gateway	Shall apply filters to limit access to network resources based on rules provided in an access control list.	-
Router	Shall apply filters to limit access to network resources based on rules provided in an access control list.	-

ATTACHMENT 5-3

Network Security Sub-functions – Data Confidentiality

CNS Accelerating CNS

F3: Data Confidentiality	F3.1: Encryption	F3.2: Key Distribution and Management	F3.3: Level of Security
End System (or DTE)	l	-	l
Certification Authority	l	l	l
Network Management Station	l	-	l
Access Point	i	-	l

Legend	Meaning
-	Not Applicable
i	Optional
/	Present, but not required for a special task
l	Present, required for a special task

Network Security Sub-functions – Data Confidentiality

CNS Accelerating CNS

F3: Data Confidentiality	F3.1: Encryption	F3.2: Key Distribution and Management	F3.3: Level of Security
End System (or DTE)	When transmitted over the network, sensitive data shall be encrypted using either symmetric or asymmetric keys.	-	Algorithms chosen for encryption shall only be able to be broken by a brute-force method, and they shall require a significant amount of computing power to break.
Certification Authority	When transmitted over the network, sensitive data shall be encrypted using either symmetric or asymmetric keys.	A Certification Authority shall provide an infrastructure for the generation, distribution and maintenance of keys and digital certificates.	
Network Management Station	When transmitted over the network, sensitive data shall be encrypted using either symmetric or asymmetric keys.	-	
Access Point	Encryption may be provided through Wired Equivalent Privacy (WEP).	-	

ATTACHMENT 5-3

Network Security Sub-functions – Data Confidentiality

CNS Accelerating CNS

F3.4: Layer of Encryption	
Physical	For encryption performed at the physical layer, all information leaving the physical interface is encrypted. This layer of encryption may be used for point-to-point or point-to-multi-point links. In the latter case, keys shall be shared by multiple entities.
Network	For encryption performed at the network layer, all information flowing between two network endpoints is encrypted. This layer of encryption shall be provided by IPSec.
Higher Layers	Encryption algorithms may be used at higher layers to achieve better granularity. Secure Socket Layer (SSL) may be used for encryption in the transport layer. Application layer encryption (e.g. secure email) may also be used.
Encryption API	An encryption API provides API for use that is independent of the underlying encryption algorithms. RFC 2743 specifies Generic Security Service Application Program Interface (GSSAPI) which shall be used for security services.

Network Security Sub-functions – Data Integrity

CNS Accelerating CNS

F5: Data Integrity	F5.1: Acceptable Transmission Error	F5.2: Anti-Spoofing / Message Digests
End System (or DTE)	-	l
Certification Authority	-	l
Network Management Station	-	l
Firewall	l	-
Gateway	l	-
Router	l	-

Legend	Meaning
-	Not Applicable
j	Optional
/	Present, but not required for a special task
l	Present, required for a special task

ATTACHMENT 5-3

Network Security Sub-functions – Data Integrity

CNS Accelerating CNS

F5: Data Integrity	*F5.1: Acceptable Transmission Error*	*F5.2: Anti-Spoofing / Message Digests*
End System (or DTE)	-	A Message Digest shall be employed for critical communications to ensure data integrity. The method digest algorithm shall follow the specifications in RFCs 1319, 1320, or 1321.
Certification Authority	-	A Message Digest shall be employed for critical communications to ensure data integrity. The method digest algorithm shall follow the specifications in RFCs 1319, 1320, or 1321.
Network Management Station	-	A Message Digest shall be employed for critical communications to ensure data integrity. The method digest algorithm shall follow the specifications in RFCs 1319, 1320, or 1321.
Firewall	Checksums and CRC algorithms shall be used to achieve error free transmission.	-
Gateway	Checksums and CRC algorithms shall be used to achieve error free transmission.	-
Router	Checksums and CRC algorithms shall be used to achieve error free transmission.	-

Network Security Sub-functions – Non-Repudiation

CNS Accelerating CNS

F5: Non - Repudiation	*F5.1: Confirmation*	*F5.2: Retention of Confirmation*
End System (or DTE)	l	l
Certification Authority	l	l
Network Management System	l	l

Legend	*Meaning*
-	Not Applicable
i	Optional
/	Present, but not required for a special task
l	Present, required for a special task

ATTACHMENT 5-3

Network Security Sub-functions – Non-Repudiation

CNS
Accelerating CNS

F5: Non - Repudiation	*F5.1: Confirmation*	*F5.2: Retention of Confirmation*
End System (or DTE)	Shall provide the mechanism needed for non-repudiation by use of Digital Certificates.	Shall provide for retention of confirmation.
Certification Authority	Shall be the provider of Digital Certificates.	Shall provide for retention of confirmation.
Network Management System	Shall provide the mechanism needed for non-repudiation by use of Digital Certificates.	Shall provide for retention of confirmation.

Network Security Sub-functions – Intrusion Protection Methods

CNS
Accelerating CNS

F7: Intrusion Protection Methods	*F7.1: Bastion Host*	*F7.2: Filters*	*F7.3: Application Gateway (Proxy Server)*	*F7.4: Internal Domain Name Server (DNS)*
DNS	-	-	-	l
Firewall	l	l	l	l
Gateway	-	l	l	i
Router	-	l	-	-

Legend	*Meaning*
-	Not Applicable
i	Optional
/	Present, but not required for a special task
l	Present, required for a special task

ATTACHMENT 5-3

Network Security Sub-functions – Intrusion Protection Methods

Accelerating CNS

F7: Intrusion Protection Methods	*F7.1: Bastion Host*	*F7.2: Filters*	*F7.3: Application Gateway (Proxy Server)*	*F7.4: Internal Domain Name Server (DNS)*
DNS	-	-	-	Shall provide for resolving host names to IP address.
Firewall	Shall be located in the Bastion Host.	Shall make use of different traffic filters that intercept and inspect each packet passing through it.	Shall authenticate users at the application level crossing the firewall in either direction.	Internal DNS shall hide the IP addresses of private hosts from the outside world.
Gateway	-	Shall filter incoming traffic and route appropriate traffic on different on-board networks	Shall serve as the proxy server to authenticate users at the application level crossing the firewall in either direction.	Internal DNS may also serve as the gateway for private hosts to the external world
Router	-	Shall provide for filtering of incoming traffic by means of NAT.	-	-

Network Security Sub-functions – Counter Measures

Accelerating CNS

F8: Counter Measures	*F8.1: Protection*	*F8.2: Intrusion Detection*	*F8.3: Response*
End System (or DTE)	l	l	l
Firewall	l	l	-
Gateway	l	-	-
Router	l	-	-

Legend	*Meaning*
-	Not Applicable
i	Optional
/	Present, but not required for a special task
l	Present, required for a special task

ATTACHMENT 5-3

Network Security Sub-functions – Counter Measures

CNS
Accelerating CNS

F8: Counter Measures	*F8.1: Protection*	*F8.2: Intrusion Detection*	*F8.3: Response*
End System (or DTE)	Shall provide for protection mechanisms such as firewalls, filters, etc..	End System shall gather and analyze information from various areas within a computer or a network to identify possible security breaches.	End System shall provide for adequate responses on identification of security breaches.
Firewall	Shall provide for protection against intrusions.	Shall log occurrences of intrusions and report them.	-
Gateway	Shall have firewalls built into them to provide protection against intrusions.	-	-
Router	Shall have NAT built into them to provide protection to private hosts.	-	-

Network Security Sub-functions – Recovery of System / Operation

CNS
Accelerating CNS

F9: Recovery of System/Operation	*F9.1: TBD*	*F9.2: TBD*
End System (or DTE)	TBD	TBD

ATTACHMENT 5-3

Network Security Sub-functions – Logging

CNS
Accelerating CNS

F10: Logging	*F10.1: TBD*	*F10.2: TBD*
End System (or DTE)	TBD	TBD
Certification Authority	TBD	TBD
Network Management Station	TBD	TBD
Firewall	TBD	TBD
Bridge or (Switch)	TBD	TBD

Network QoS Functions

CNS
Accelerating CNS

- User Requirements
- Protocols
- SLAs
- Policies

ATTACHMENT 5-3

QoS Subfunctions

Accelerating CNS

User Requirements

- § Resource Availability
- § Error Performance
- § Response Time
- § Throughput
- § Automated Fault Detection
- § "Guaranteed Delivery" for critical traffic
- § Jitter / Delay
- § Classify Traffic Flow
- § QoS API

Service Level Agreements

QoS Policies

QoS Subfunctions (cont..)

Accelerating CNS

Protocol Options

- § Reservation Protocol (RSVP)
- § Differentiated Service (DiffServ)
- § Multi-Protocol Label Switching (MPLS)
- § Subnet Bandwidth Management (SBM)
- § Policy Routing

ATTACHMENT 5-3

Network QoS Functional Elements

Accelerating CNS

	OSI Layer	User Requirements	Protocol s Options	Service Level Agreements	Policies
End System (or DTE)	7	l	-	-	-
Autoconfigure / Loader	7	-	-	-	-
Certification Authority	7	l	-	-	-
DHCP	7	-	-	-	-
DNS	7	-	-	-	-
Network Management Station	7	-	-	-	-
Firewall	3	-	-	-	-
Gateway	3	/	l	-	-
Router	3	/	l	-	-
Access Point	2	/	l	-	-
Bridge (or Switch)	2	/	l	-	-
Backbone	1	¡	-	-	-
Cable Plant	1	/	-	-	-
Repeater (or Hub)	1	/	-	-	-

Legend	Meaning
-	Not Applicable
¡	Optional
/	Present, but not required for a special task
l	Present, required for a special task

Next Steps

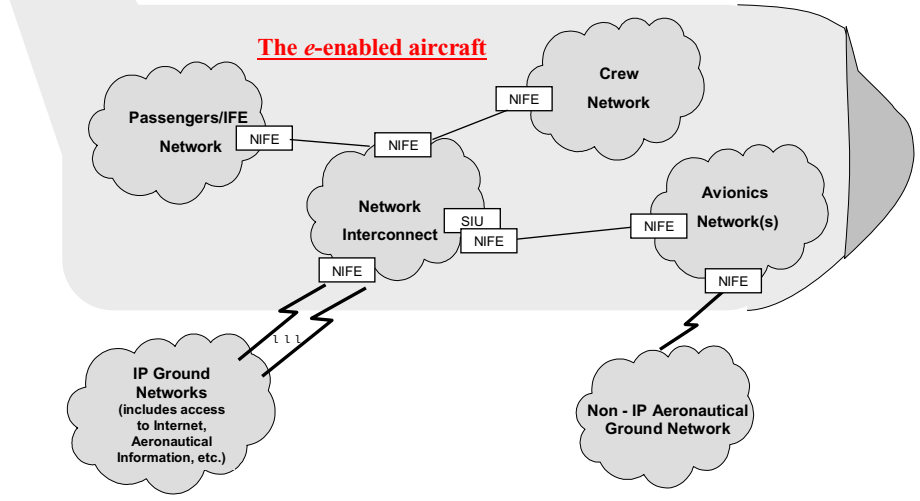
Accelerating CNS

- **Break down the End-to-End communications process by potential information flow and describe what services are required for each flow. (see reference model)**
- **Potential endpoints to consider include IP and Non-IP Ground systems, the Avionics and Pilot, the Crew, and the Passengers**
  - **Ground IP Avionics**
    - » Weather
  - **Ground Non-IP Avionics**
    - » CPDLC
  - **Avionics Crew**
  - **Ground IP Crew**
  - **Ground IP Passenger**

ATTACHMENT 5-3

Reference Model - Domains

CNS
Accelerating CNS



Source: Developed ADN 664 Meeting 02/13/02

Next Steps - Example

CNS
Accelerating CNS

	F1: Authentication	*F3: Data Confidentiality*	*F4: Data Integrity*
Ground IP Avionics	l	l	l
Ground Non-IP Avionics	l	l	l
Avionics Crew	l	l	l
Ground IP Crew	i	i	l
Ground IP Passengers	-	i	l

Legend	*Meaning*
-	Not Applicable
i	Optional
/	Present, but not required for a special task
l	Present, required for a special task

ATTACHMENT 5-3

Next Steps - Example

Accelerating CNS

F5: Data Integrity	F5.1: Acceptable Transmission Error	F5.2: Anti-Spoofing / Message Digests
Ground IP Avionics	l	l
Ground Non-IP Avionics	l	l
Avionics Crew	l	i
Ground IP Crew	l	i
Ground IP Passengers	l	i

Legend	Meaning
-	Not Applicable
i	Optional
/	Present, but not required for a special task
l	Present, required for a special task

Next Steps - Example

Accelerating CNS

F5: Data Integrity	F5.1: Acceptable Transmission Error	F5.2: Anti-Spoofing / Message Digests
Ground IP Avionics	Checksums and CRC algorithms shall be used to achieve error free transmission.	Essential IP communications shall be validated through message digests.
Ground Non-IP Avionics		Essential Non-IP communications shall be validated through message digests.
Avionics Crew		Communications between the avionics and the crew may be validated through message digests. If any commands are sent from the crew to the avionics, these shall be validated through message digests.
Ground IP Crew		Communications between the crew and Ground IP systems may be validated through message digests. Essential communications should not go through this channel.
Ground IP Passengers		Communications between the passengers and Ground IP systems may be validated through message digests. This is left to individual passengers to implement as required.

ATTACHMENT 5-3

Appendix: Glossary

Accelerating CNS

Access Point	A Layer 1 device. An Access Point converts data to and from a radio waves for wireless networking.
Autoconfigurer/ Loader	A Layer 7 System. An Autoconfigurer/Loader is used in network setup or after a network outage to automatically configure and restore network settings.
Backbone	A Layer 1 device. A Backbone provides high bandwidth connections among Intermediate Systems.
Bridge	A Layer 2 device. A bridge interconnects two sub-networks based on Layer 2 (MAC) addressing.
Cable Plant	A Layer 1 device. The physical transmission media in a network.
Certification Authority	A Layer 7 System. A Certification Authority creates and authenticates keys used in encryption algorithms.
Data Terminal Equipment (DTE)	A Layer 7 System. A DHCP Server provides Dynamic Host Configuration Protocol (DHCP) services on a TCP/IP network. A DHCP server assigns IP addresses to and releases IP addresses from devices as they join and leave the network.
DNS Server	A Layer 7 System. A DNS Server provides Domain Name System (DNS) services on a TCP/IP network. A DNS server translates hostnames into IP addresses.
End System	A Layer 7 System. An End System runs applications which depend on network services; End Systems create and process the data which is transferred through a network.
Firewall	A Layer 3 Intermediate System. A firewall applies rules on data flows between networks.
Gateway	A Layer 3 Intermediate System. A Gateway provides translation for traffic between incompatible networks.

Appendix: Glossary

Accelerating CNS

Hub	A Layer 1 or 2 Intermediate System. A Hub may be a Layer 1 Repeater Hub (used in Classic Ethernet to extend individual Ethernet segments) or a Layer 2 Bridge/Switch Hub (used in Full Duplex Ethernet to intelligently link multiple Full Duplex Ethernet Segments). In this Part, Hub is used strictly to mean a Repeater Hub, and as such is treated as a Layer 1 Intermediate system.
Network Management Station	A Layer 7 System. A Network Management Station is used to monitor and maintain the network.
Repeater	A Layer 1 device. A Repeater receives, amplifies, and retransmits signals.
Router	A Layer 3 Intermediate System. A Router is a node through which at least two networks are able to communicate with each other.
Switch	A Layer 2 Intermediate System. A Switch is a multi-port Bridge which is capable of running at or near theoretical maximum capacity. Because Switches and Bridges are functionally equivalent, they are treated together in this part.

ATTACHMENT 6-1

To: ARINC Aircraft Data Network Working Group

From: Brian Smith, Honeywell

Date: 6/18/2002

Re: Aircraft Network Management White Paper

Background

Data communications network technology is being incorporated into aircraft environments for a variety of purposes. Some aircraft Local Area Networks (LAN's) exist to provide passenger services. Other LAN's will support crew functions and aircraft flight operations. These networks may connect to ground based networks over a variety of Wide Area Network (WAN) air/ground network links. A failure in a LAN or WAN network component may hinder non-critical operation of the aircraft, perhaps preventing a gate departure from occurring on time. In ground based data networks, network management tools are used to monitor it's condition and respond to faults. These types of tools will need to be incorporated in aircraft networks. This paper provides a brief overview of some operational needs and a possible approach to address these needs.

Operational Needs

To be effective, network management must provide visibility into all network "domains" that affect aircraft network operations. This includes one or more airborne networks on a single aircraft, air/ground networks, and associated ground networks (such as Gatelink networks at airports). A network management system (including both air and ground NMS components) should be able to merge the states of each of these components. The scope of the problem is similar to large international corporate networks and calls for distributed network management. Some essential differences in the airborne environment include bandwidth is expensive and generally there will not be network expertise on the aircraft or at remote airports.

The most important operational need that aircraft NMS must address is to know the current state and health of the network. Typically, a NMS probes network components to determine if they are reachable and may gather state information from them. The NMS is able to provide a NMS user with an overview of the health of the network. When predefined events occur, an alert may be sent to the ground. Monitored network components include routers, switches, firewalls, gateways, servers and any other components that affect network operation. It is desirable that the airborne NMS integrate with existing fault reporting systems on the aircraft so status and fault information can be communicated through current methods. This approach provides for a "out of band" method of manages aircraft networks that can be available when normal air/ground network links fail.

The NMS system should provide tools to the NMS user to see network statistics and control network equipment to facilitate troubleshooting a problem and attempting to recover. It should be possible for the NMS user to be on the ground, assuming at least one air/ground data link and minimal network equipment is operational. Access to the control functions need to be protected by user authentication and encryption.

A NMS should be capable of logging important network statistics, which can be reviewed periodically, to solve problems and perform capacity planning. These functions are proactive in nature and enable the operator to correct problems before they are noticed the network users.

Network authentication, attack detection, and virus detection mechanisms will need to be incorporated into aircraft networks. The aircraft NMS should provide a means to access reports from these functions at a minimum.

ATTACHMENT 6-1

Predefined automatic responses may be built into the aircraft NMS to attempt to isolate the source of an attack, while in flight.

A simple user interface that provides basic network health information should be available to flight crew on board the aircraft. This might be a simple graphic with color-coded symbols, representing different parts of the network. This type of tool would assist a flight attendant in determining that there is a network fault at a passenger's seat that explains why a passenger service is not working (verses the passenger's laptop). The passenger might be moved to an available seat that is functional.

Approaches

Approaches to network management should be standards based, leverage commercial technology/experience, and be highly scalable/extensible. Most commercial IP network management systems are based on the Internet Standard, Simple Network Management Protocol (SNMP). Although after many years, commercial products fail to provide the comprehensive solution that was visualized, it is the best approach available and is used to manage large IP based computer networks. SNMP Version 3 provides for user authentication, and encryption so should be considered. Figure 1 shows one possible architecture. It assumes a global IP network exists between carriers, airports, WAN service providers and aircraft.

The aircraft NMS station monitors airborne network component and air/ground network links. This information is available in a simplified view to aircraft crew. It is also accessible to ground staff via air/ground data links. Airborne NMS stations report status periodically to the root NMS station. If alarm conditions occur, a SNMP trap may be sent to the ground to alert the root NMS station.

Air/ground data link service providers may periodically provide a carrier's root NMS station with status of their portion of the network. Likewise, airport networks, including LAN's such as Gatelink, may have NMS systems that report status to a carrier's root NMS station.

The root NMS provides the carrier with a comprehensive view of the networks associated with operation of it's fleet of aircraft. From this central location, staff is able to probe different components of the network to determine faults, reconfigure to isolate problems, and inform appropriate personnel of issues.

ATTACHMENT 6-1

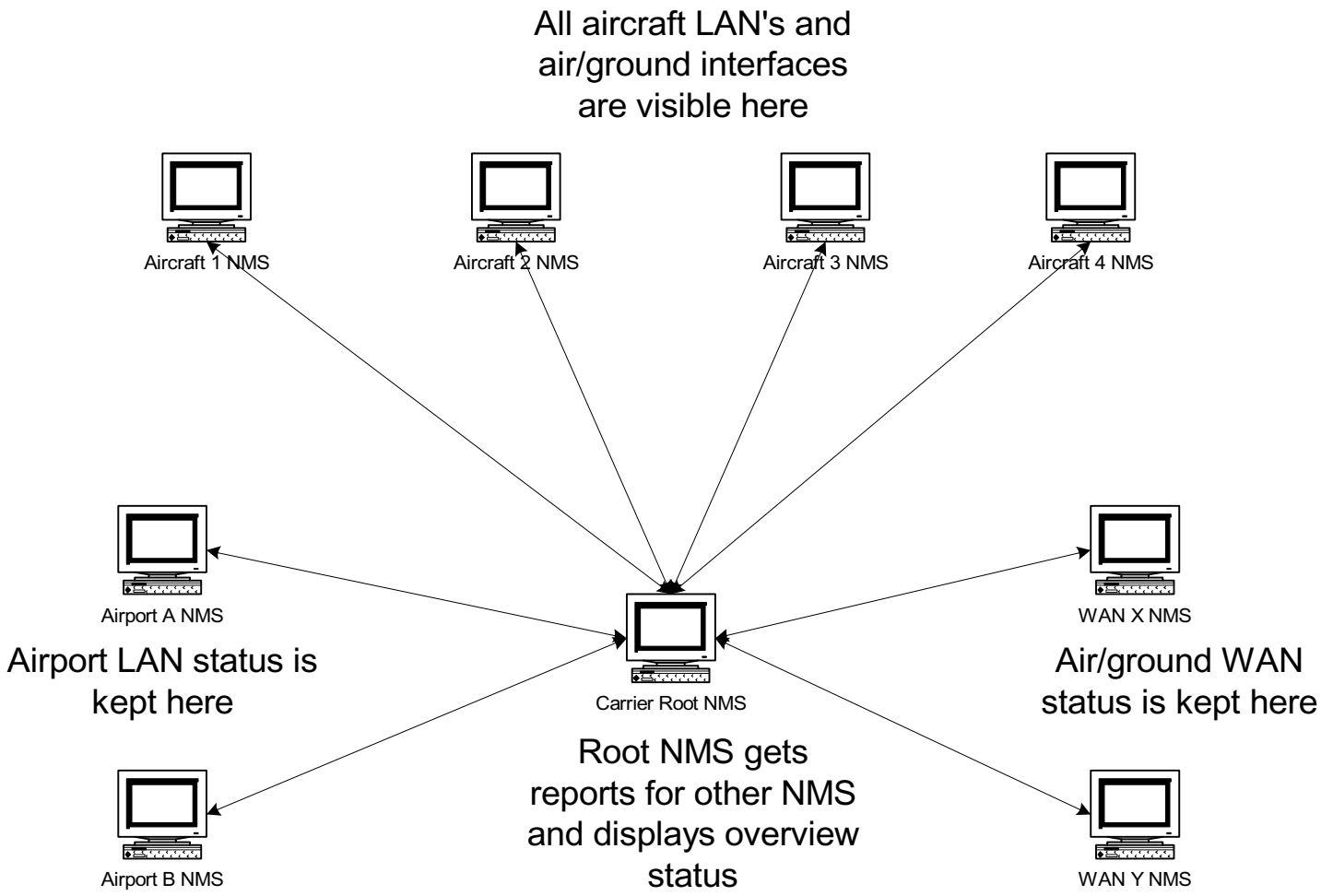


Figure 1 - Example Aircraft Network Management System Concept Architecture

ATTACHMENT 6-2

From: IANA Private Enterprise Number [iana-pen@icann.org]

Sent: Thursday, June 13, 2002 2:29 PM

To: Courtney, Roy L (RCOURTNE)

Subject: RE: Application for Enterprise-number (13712)

The IANA has assigned the following Private Enterprise Number 13712.

Please visit the following link to confirm all contact information associated with the Enterprise Number are correct:

<http://www.iana.org/assignments/enterprise-numbers>

Please notify the IANA if there is a change in your contact or company information.

Thank you,

Bill Huang
IANA - Private Enterprise Numbers

ATTACHMENT 6-3

ARINC 664 Part 6 Review
Network Management Specification

June 27, 2002

Taiboo Song
taiboo.song@boeing.com

Current Situation for Network Management Activities

- Part 6 covers definition of SNMP
  - 2.1 Network Management Solution
  - 2.2 SNMP description
  - 2.3 MIB description
- ARINC 763 activity: API
- Who is the collector of data?
- ARINC 628 activity: MCU
- Missing a integration plan
- Missing a private MIB definition plan

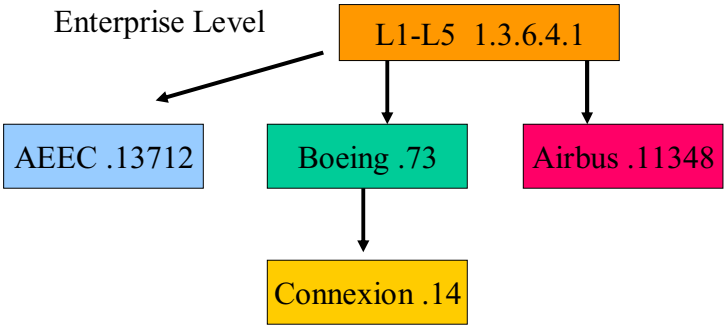
ATTACHMENT 6-3

Goal

- Simplify SNMP Interface - Manageability
- Support "Future-proof" - Scalability
- Using an existing SNMP standard - Interoperability
- "Future-creativity" or retain proprietary data - Creativity

SNMP Integration Plan

- Current OID tree structure?

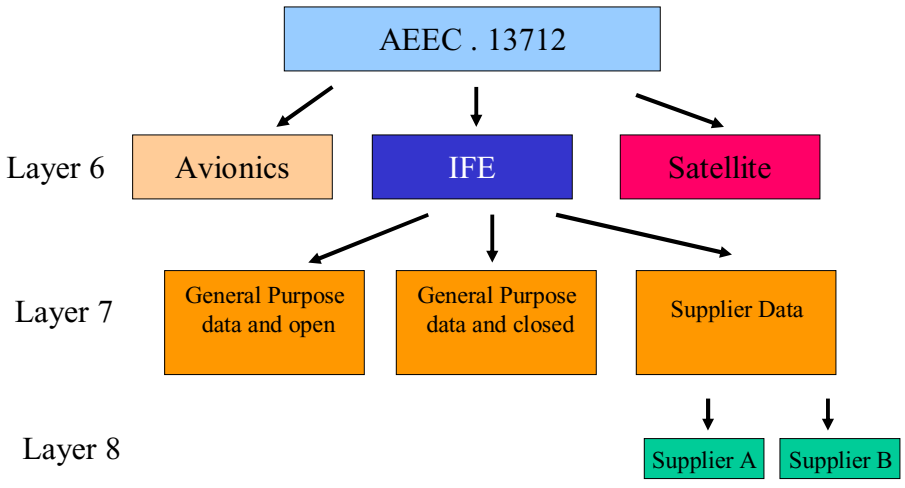


ATTACHMENT 6-3

Categories of the Model

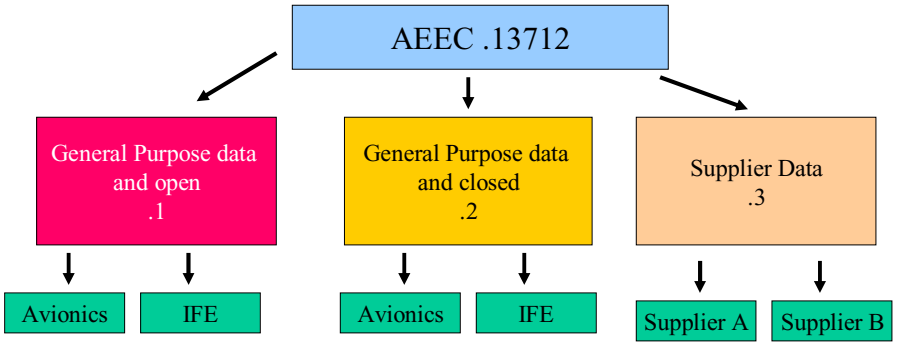
- Option A - Systems
  - Avionics
  - IFE
  - Satellite
- Option B - Attributes
  - General purpose data and open
  - General purpose data and closed
  - Supplier data

Models –option A



ATTACHMENT 6-3

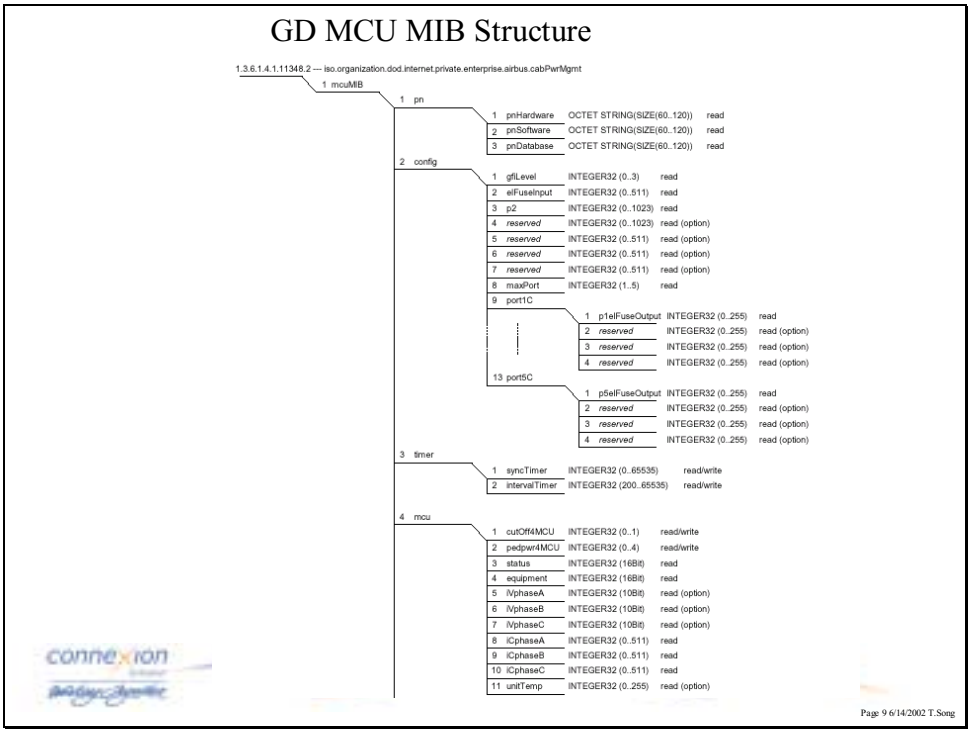
Models –option B



Proposal

- ADN (ARINC 664) is the owner of the SNMP definitions and global OID architecture
- ARINC 628 is the owner of the private MIB definitions for cabin entertainment equipments
- Future joint meeting, when appropriate, for final approval

ATTACHMENT 6-3



ATTACHMENT 6-4

Strawman

MCU Interface Control Document

Prepared By Doug Brown, General Dynamics Airborne Electronics Systems Date 02/28/2002

Prepared By Torsten Duering, KID Systeme GmbH Date 05/24/2002

Prepared By Katherine Garcia, Rockwell Collins Passenger Systems Date 03/19/2002

Prepared By Juergen Barg, Airbus Date 02/14/2002

ATTACHMENT 6-4

5.2 Subnet Address

Each MCU uses one standard default method for determining its IP address. The default base address for the MCU subnet is 172.17.254.0/24. Each unique MCU IP address is derived by using the binary number encoded on discrete input pins A4-A0 as the last octet in the base address, as shown in Table 1 MCU Pin Coded Addressing.

Table 1 MCU Pin Coded Addressing

Host	IP Address	A4	A3	A2	A1	A0
IFE SNMP Manager	172.17.254.1					
MCU 1	172.17.254.2 reserved for internal use 172.17.253.2	0	0	0	0	0
MCU 2	172.17.254.3 reserved for internal use 172. 17.253.3	0	0	0	0	1
MCU 3	172.17.254.4 reserved for internal use 172. 17.253.4	0	0	0	1	0
MCU 4	172. 17.254.5 reserved for internal use 172. 17.253.5	0	0	0	1	1
.	.	.	.	.	.	.
MCU 8	172. 17.254.9 reserved for internal use 172. 17.253.9	0	0	1	1	1
optional MCU 9	172. 17.254.10 reserved for internal use 172. 17.253.10	0	1	0	0	0
.	.	.	.	.	.	.
optional MCU 16	172. 17.254.17 reserved for internal use 172. 17.253.17	0	1	1	1	1
theoretical MCU 17	172. 17.254.18 reserved for internal use 172. 17.253.18	1	0	0	0	0
.	.	.	.	.	.	.
MCU 31	172. 17.254.32 reserved for internal use 172. 17.253.32	1	1	1	1	0
Service Address	Reserved for MCU service	1	1	1	1	1

In addition to supporting the standard default IP addresses, the MCUs may use the Configuration Module as an optional feature to set the base MCU subnet.
This option works like the default method, except instead of the default network address, the MCU uses a base network address defined in the Configuration Module. The discrete input pins are still used to derive the host address. The IFE SNMP Manger IP Address may also be determined by the Configuration Module. For this option the MCU address may conform to the block of addresses allocated to the IFE system by ARINC 628 Part 4: 172.16.0.0/16 - 172.19.0.0/16. The MCU will revert to using the default standard address if any of the following conditions occur[3]:

a. The Configuration Module is missing or corrupted,

b. The Configuration Module base MCU subnet address cannot be read.

[3] For using the option of changing the base network addresses by the Configuration Module, it has to be ensured that the system can operate on different base network addresses of each MCU and IFE head end in worst case!

ATTACHMENT 6-4

5.3 MIB Structure and Definition

The MCU MIB contains data accessible to the IFE headend via SNMP protocol. The Community String for the SNMP protocol is "ISPN" to encapsulate this communication from other SNMP communications.

5.3.1 MCU MIB Structure

Figure 4 MIB Structure, shows the MCU MIB structure. Some as option marked variables are not necessarily required. If theses variables are not implemented a request from the IFE to the MCU should be answered by a SNMPv1 standard error message of an unknown variable ("noSuchName") . If variables marked as option are not implemented a get-next request should deliver the next following variable back. Generally SNMPv1 standard error messages should be supported.
If the MCU supports less than 5 ports, the ports should be arranged in a secuence starting with port 1 and ending with the value given in the variable *maxPort*. The port related variables for unsupported ports should implement the value 0 as these variables are still included in the static variable binding list of the state trap. The only exceptions are the port status variables (*p1status* to *p5status*) that have to set Bit 15 to indicate that this port is not supported by this MCU. The writable variables are still writable and are initialized with a value 0.

ATTACHMENT 6-4

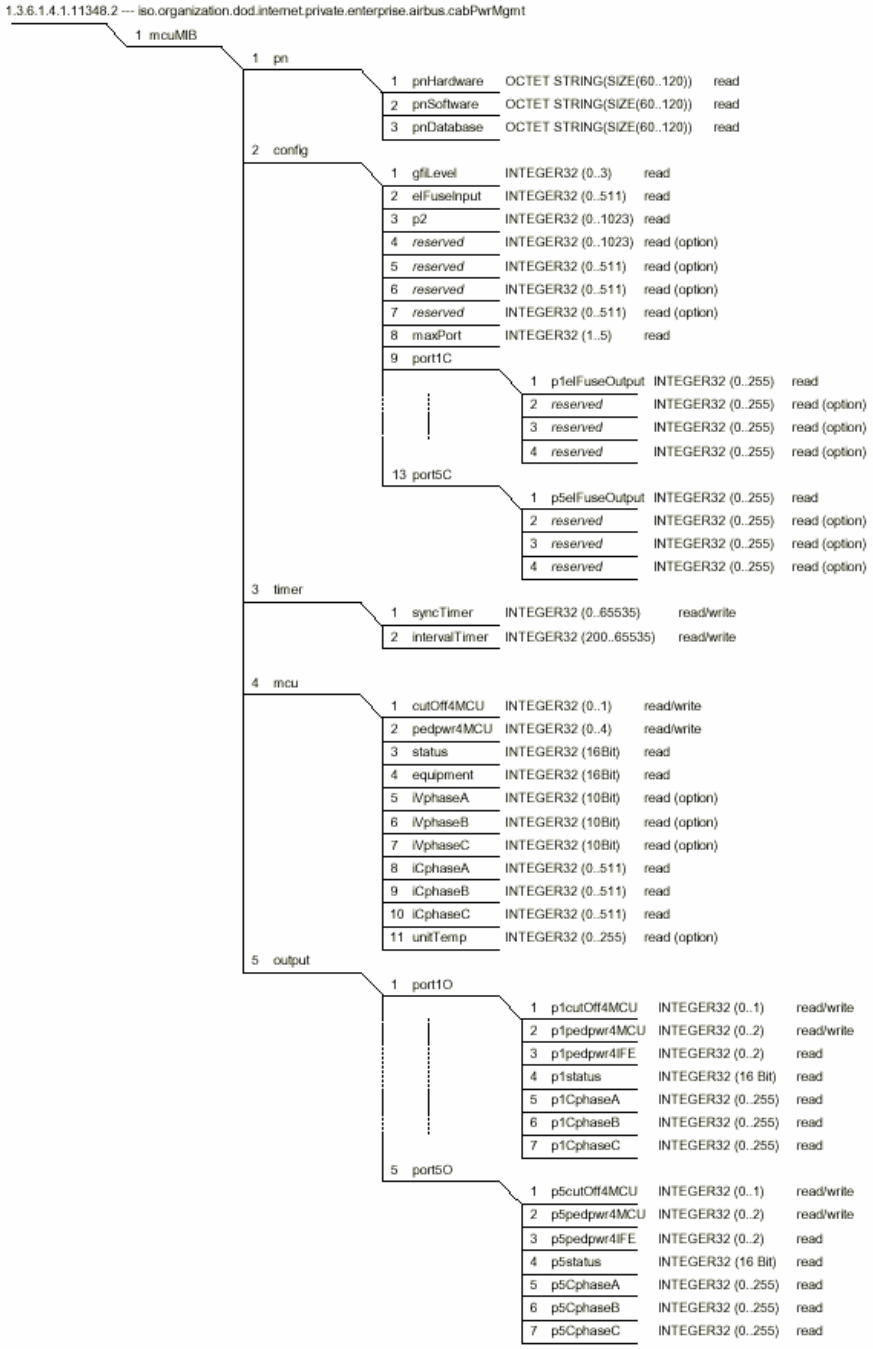


Figure 4 MIB Structure[4]

[4] All variables marked read/write have the information flow from IFE to the MCU but are readable for verification. The reserved variables are reserved for optional vendor specific power management level, resolution 0.1 Amps (for IFE information purposes only).

ATTACHMENT 6-4

5.3.2 MCU MIB Definition

The following specific MIB definition of the MCU hosted variables is based on SMI notation in compliance with the conventions of RFC1212.

```
MCU-MIB DEFINITIONS ::= BEGIN

IMPORTS OBJECT-TYPE     FROM RFC-1212
        TRAP-TYPE       FROM RFC-1215
        enterprises     FROM RFC-1155-SMI;


-- LAST-UPDATED "200205160000Z"
-- ORGANIZATION "MCU-IFE-Communication Workinggroup"
-- CONTACT-INFO "Workinggroup as listed in the ICD"
-- DESCRIPTION "MCU specific MIB."

airbus OBJECT IDENTIFIER ::= {enterprises 11348}

cabPwrMgmt OBJECT IDENTIFIER ::= {airbus 2}

mcuMIB OBJECT IDENTIFIER ::= {cabPwrMgmt 1}

pn OBJECT IDENTIFIER ::= {mcuMIB 1}

pnHardware OBJECT-TYPE
       SYNTAX OCTET STRING (SIZE (60..120))
       ACCESS read-only
       STATUS mandatory
       DESCRIPTION
              " pnHardware provides the hardware version description as a  min.
              60 up to 120 character long ASCII Text string. This string
              provides a framework of 6 possible P/N frames consisting of 20
              characters each. Unused octets should be filled with ASCII
              character 20h. 3 frame works should be supmitted as minimum, the
              remaining 3 could be used as an option for further P/N. "
       ::= {pn 1}

pnSoftware OBJECT-TYPE
       SYNTAX OCTET STRING (SIZE (60..120))
       ACCESS read-only
       STATUS mandatory
       DESCRIPTION
              " pnSoftware provides the software version description as a  min.
              60 up to 120 character long ASCII Text string.  This string
              provides a framework of 6 possible P/N frames consisting of 20
              characters each. Unused octets should be filled with ASCII
              character 20h. 3 frame works should be supmitted as minimum, the
              remaining 3 could be used as an option for further P/N. "
       ::= {pn 2}

pnDatabase OBJECT-TYPE
       SYNTAX OCTET STRING (SIZE (60..120))
       ACCESS read-only
       STATUS mandatory
       DESCRIPTION
              "pnDatabase provides the database version description as a  min. 60
              up to 120 character long ASCII Text string. This string provides a
              framework of 6 possible P/N frames consisting of 20 characters
```

ATTACHMENT 6-4

```
                each. Unused octets should be filled with ASCII character 20h.  3
                frame works should be supmitted as minimum, the remaining 3  could
                be used as an option for further P/N. "
        ::= {pn 3}

config OBJECT IDENTIFIER ::= {mcuMIB 2}

gfiLevel OBJECT-TYPE
        SYNTAX INTEGER (0..3)
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
                "Reports the configured GFI trip level as determined by input
                discretes GFI 2 and GFI 1 (00b = 30mA/30ms, 01b = 50mA/50ms, 10b =
                75mA/75ms, 11b = 100mA/100ms)."
        ::= {config 1}

elFuseInput OBJECT-TYPE
        SYNTAX INTEGER (0..511)
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
                "Provides the electronic Fuse-level per input phase information
                from the Configuration Module on current base, Range 0  - 511 (9
                bit, 0 - 51.1 Amps)"
        ::= {config 2}

p2 OBJECT-TYPE
        SYNTAX INTEGER (0..1023)
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
                "Provides the P2-level information (commentary: control data
                interface disable limit for the complete MCU, this definition
                differs from the definition of P2 in ARINC 6 28 Part 4A Supplement
                1) from the Configuration Module on current basis, Range 0  - 1023
                (10 bit, 0 - 102.3 Amps). This value is for informational purposes
                only and should not be used by the IFE for power management."
        ::= {config 3}

maxPort OBJECT-TYPE
        SYNTAX INTEGER (1..5)
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
                "maxPort provides the maximum Number of MCU ports. Valid numbers
                are 1 up to 5 ports per MCU"
        ::= {config 8}

port1C OBJECT IDENTIFIER ::= {config 9}

p1elFuseOutput OBJECT-TYPE
        SYNTAX INTEGER (0..255)
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
                "Provides the electronic Fuse-level for output port 1 per phase
                information from the Configuration Module on current base, Range 0
                - 255 (8 bit, 0 - 25.5 Amps)"
```

ATTACHMENT 7-1

PROJECT PAPER ARINC 664

PART 7

AN EXAMPLE
DETERMINISTIC NETWORK

STRAWMAN

April 2002

Default configuration for the ES

ATTACHMENT 7-1

CONTENTS

Table of Diagrams

1 Deterministic Network Overview

1.1 Purpose of the document

1.2 Structure

1.3 Referenced and Applicable Documents

1.4 System overview

The example deterministic network chosen for Part 7 of Project Paper ARINC 664 is based on the Aircraft Full Duplex Network (AFDX). References to AFDX in this document should be interpreted as references to the example deterministic network.

A part 7 ARINC 664 network is a profiled network with addressing assigned throughout by the System Design Authority. It is composed of a set of Ethernet 100/10 Mbits/s switches to which avionics computers are connected. The network interfaces of these computers are called the End Systems. There is no requirement for the End Systems and switches to be synchronised.

Ethernet switches provide a first answer to collision and flooding issues, however, enhancements are needed for use in avionics networks. This redundant network aims at providing services that guarantee :

- *latency in the network less than a defined maximum*
- *network partitioning with data flows through Virtual Links*
- *a bandwidth to each subscriber and to each identified flow of each subscriber*
- *a demonstrably ~~vanishingly~~ low bit error rate for each transmitted message*
- *3 application services (sampling, queuing and File Transfer)*
- *Impersonation protection ~~???~~*

These notions will be developed in the following sections.

Deterministic Ethernet is defined in this document as, per Virtual Link, a guaranteed bandwidth with a maximum latency, a maximum jitter, and defined probability of frame loss. Moreover, ordinal integrity is maintained within each Virtual Link.

The following is a brief description of the rationale for each element of AFDX determinism :

- Guaranteed bandwidth
  *Regulating the bandwidth of each Virtual Link ***
  Regulating the bandwidth of each physical link
- Maximum latency
  *Defined frame size per Virtual Link **
  Defined physical latency
  Defined maximum configuration latency
- Maximum jitter
  *Configuration limit for multiple Virtual Links (End System) **
  ~~Result of contention in the switches.~~
  Defined latency of switches
- Defined probability of frame loss
  No frame loss due to collision (full duplex switch)
  No frame loss due to contention (no buffer overflow)
  The probability of frame loss is linked to the Bit Error rate.
- The ordinal integrity is maintained within each Virtual Link.
  Packet sequencing for each Virtual Link is guaranteed by the End System

Packet sequencing for each Virtual Link is guaranteed by the switches

** defined by system integrator according to the guidelines defined in this document.*
*** System integrator defined and regulated by End System, and policed by the switches.*

The intent of part 7 is to rely as far as possible on standards protocols. Nevertheless, in order to enhance integrity for data transmission in an Avionics environment, some feature have been added.

The two main enhancements that will be described hereafter are :

- *Flow control (traffic shaping on the ES, and traffic policing on the switch)*
- *network redundancy.*

For interoperability purposes, ICMP, SNMP, TCP, IP and UDP have been chosen as communication protocols for AFDX.

To be added

Document breakdown:

Determinism related interoperability issues (ES and Switch)
Middle layer (above MAC) related interoperability issues (ES)
Application Layer related interoperability issues (ES). E.g. data packaging.
System issues (e.g. determinism algorithm, modeling, network management, configuration – dataloading, default configuration-, interconnection with compliant networks).

TBD
Reference to profile network (see above inrto.)

Systems issues (to be addressed later)
- Only one End System within the Avionics network should be the source of a Virtual Link
- Hardware switch over. Each redundant hardware device needs his own VL to transmit data. On the other hand, the same VL can be use to send data to the two redundant devices.
- Message construction rules in sampling (do not exceed UDP payload)

2 End System Functional characteristics

2.1 ~~I~~Determinism ~~related~~ interoperability and Determinism ~~issues~~

2.1.1 Introduction

The main function of the End System is to provide services, which guarantee a secure and reliable data exchange to the application software.

Quality of Service (QoS) provides a method for categorising traffic and for ensuring that particular categories of traffic will always flow across the network at the service level to which they are entitled, regardless of competing demands.

For the Aircraft Network, there is no need to differentiate between several categories or traffic classes. Each network transmission request must be serviced regardless of the data type and a maximum network transit delay (also called end-to-end latency) must be guaranteed. Therefore, the only service class needed in the Aircraft Network is a guaranteed service.

A guaranteed service provides a firm, mathematically provable, upper bound on end-to-end packet transit delay. As a result, to guarantee a bounded delay implies to guarantee a certain bandwidth at the link level.

Hence a guaranteed service provides both upper bounded delay and constant bandwidth leading to a logical open connection between one transmitting node and one or more receiving nodes. Packets belonging to the same connection define a flow.

To summarise, a guaranteed service leads to the following characteristics :

- bandwidth and bounded latency are guaranteed.
- particular delay jitter for a flow (end-to-end transit delay variation between any two packets of the same flow) is not fixed since it depends on the global network traffic at a given time. Nevertheless, a bound to the delay jitter can be mathematically computed.

A description of the End System communication stack is shown below :

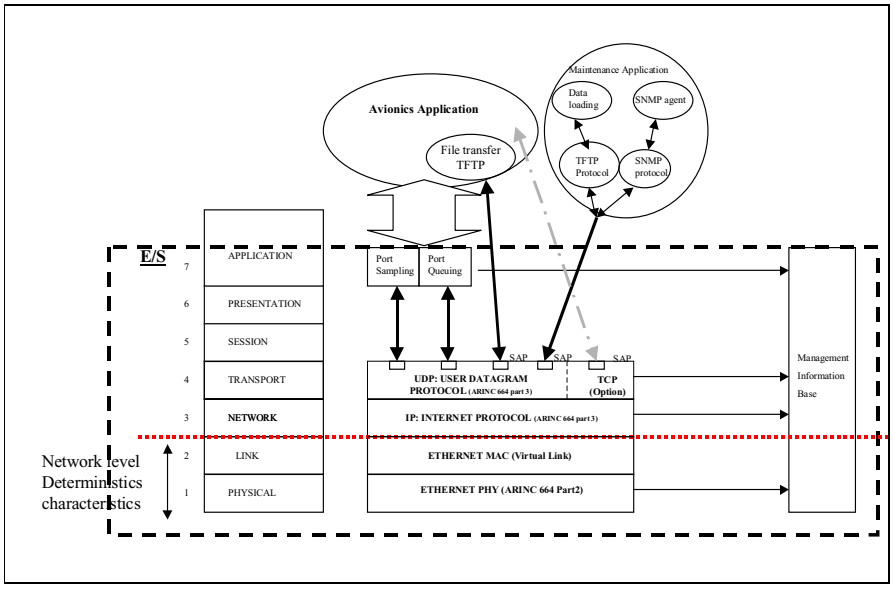


Figure 1 : End System Protocol Layers

2.1.2 Virtual Link

The description of the "Virtual Link" concept is presented here as it is widely used in the document.
One end-system is the origin of zero or more Virtual Links. End-systems exchange Ethernet frames through Virtual Links. Only one End System within the Avionics network should be the source of a Virtual Link.

A Virtual Link is a conceptual communication object, which has the following properties:

A Virtual Link defines a logical unidirectional connection from one source end-system to one or more destination end-systems.

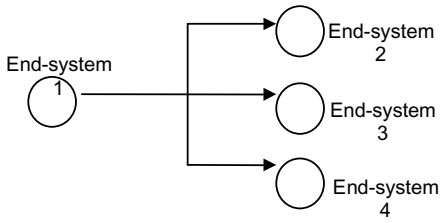


Figure 2 : A Virtual Link = a path

Each Virtual Link has a dedicated maximum bandwidth

The E/S should provide logical isolation with respect to available bandwidth among the Virtual Link(s) it supports. Regardless of the attempted utilisation of a VL by one application, the available Bandwidth on any other VL is unaffected.
For each Virtual Link, the End System should maintain the ordering of data as delivered by an application, for both transmission and reception.

COMMENTARY

The Virtual Link processing is achieved through a flow control mechanism which regulates the flows of data produced by the different sources belonging to this E/S~~VL~~, this mechanism provides partitioning at the network layer.
The End-system communication stack should guarantee in transmission the allocated bandwidth of each Virtual Link regardless of the attempted use of Bandwidth by other Virtual Links, in order to preserve segregation between applications (or partitions) at the network level. One Virtual Link should not be shared by two or more source applications (or source partitions).

2.1.3 Flow /traffic control

At the output of the End System, the flow of frames associated with a particular Virtual Link is characterised by 2 parameters : BAG (Bandwidth Allocation Gap) and JITTER.

The BAG represents the minimum space between the first bits of 2 consecutive frames as if the frames experienced no jitter from the scheduler.

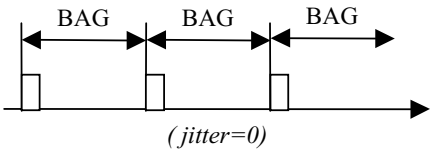


Figure 3 : The BAG in a VL for a maximum bandwidth data flow

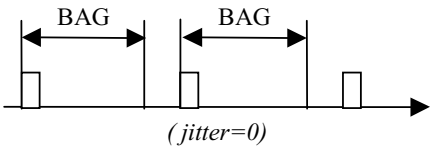


Figure 4 : The BAG in a VL for a non maximum bandwidth data flow

COMMENTARY

No frame will be transmitted while a VL is eligible but has no data for transmission.

The Scheduler multiplexes the different flows coming from the regulators

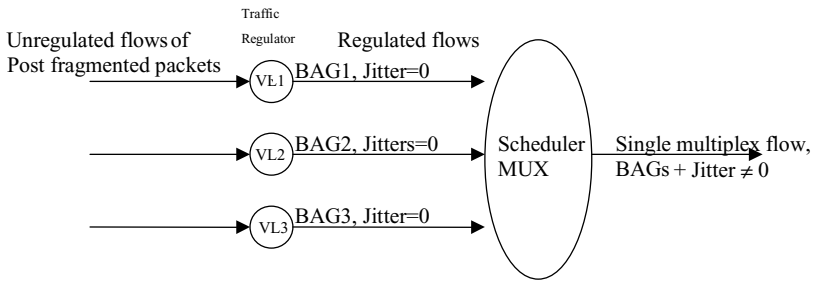


Figure 5 : Model of the flow control mechanism

At the output of the scheduler, for a given Virtual Link, frames can appear in a given time interval. This interval is defined as the maximum admissible jitter.
This jitter is introduced by the scheduler and not by the traffic flow itself.

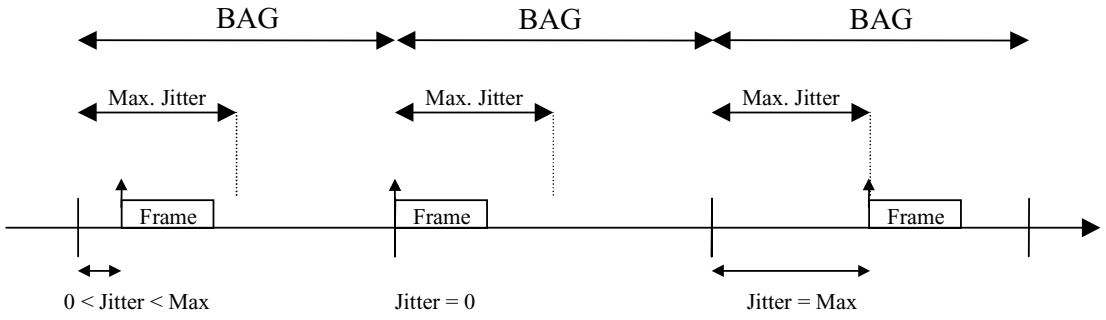


Figure 6 : The jitter effect for a maximum bandwidth data flow

The End System should regulate transmitted data on a per VL basis, since this Traffic Shaping Function (exact knowledge of flow characteristics) is the basis of the determinism analysis.

On a per VL basis the traffic regulator (traffic shaping function) should shape the flow to send no more than one packet in each interval of BAG milliseconds

COMMENTARY

The aim of the traffic shaping function is to limit the instantaneous packet rate of the Virtual Links by spacing the packets. The regulator is responsible for controlling the bandwidth given to a Virtual Link according to the BAG.

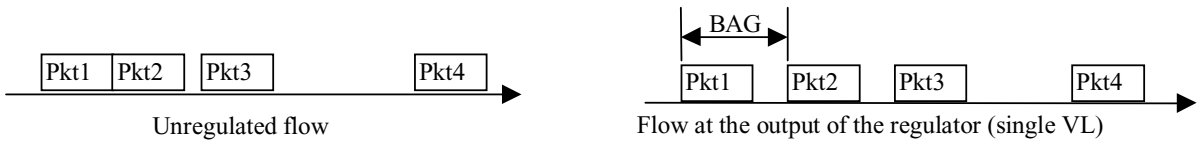


Figure 7 : Virtual Link flow regulation

The maximum usable bandwidth of each VL is characterised by its BAG and its authorized L^{max} (maximum VL frame size). The maximum usable bandwidth = L^{max} / BAG.

The End System should accommodate VL frames up to a size of 1518 bytes in both transmission and reception.

For each VL, the End System should have one BAG value given by the End System configuration table.

The Traffic shaping function of the E/S should be able to handle BAG values in range 1 ms to 128 ms. These values should satisfy the following formula: BAG=2^k [in ms], (k integer in range 0 to 7).

COMMENTARY

If an application needs to transmit data less often than 128ms, the BAG value 128 will be used. BAG values are limited to powers of 2 in order to simplify the E/S design.

2.1.4 The Sub-VL

A VL can be composed of a number of Sub-VLs, and in this case the VL is made up only from these Sub-VLs. Each Sub-VL has a dedicated FIFO, and the Sub-VL FIFO queues are read on a round robin basis by the (main) VL FIFO queue. This round robin function is done on a MAC frame basis, therefore IP fragmentation (if any) would have been done prior to loading the Sub-VL FIFOs.

COMMENTARY

Implementation of Sub-VL is an optional feature that has no impact on the determinism of the network. It can be used to optimise the bandwidth utilisation of a VL.

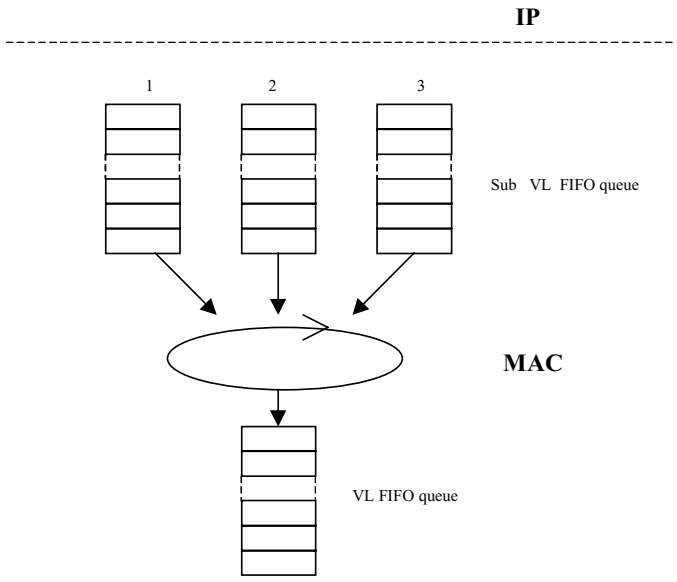


Figure 8 : The Sub-VL FIFO queue

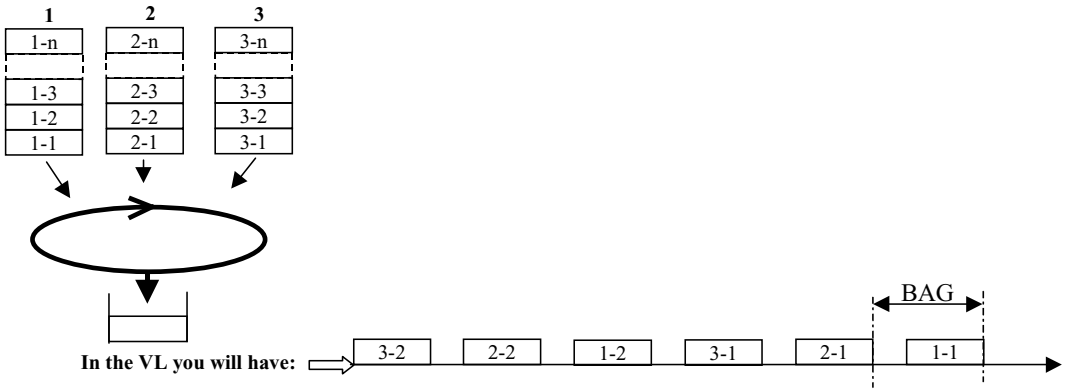


Figure 9 : 1st example of traffic on the VL

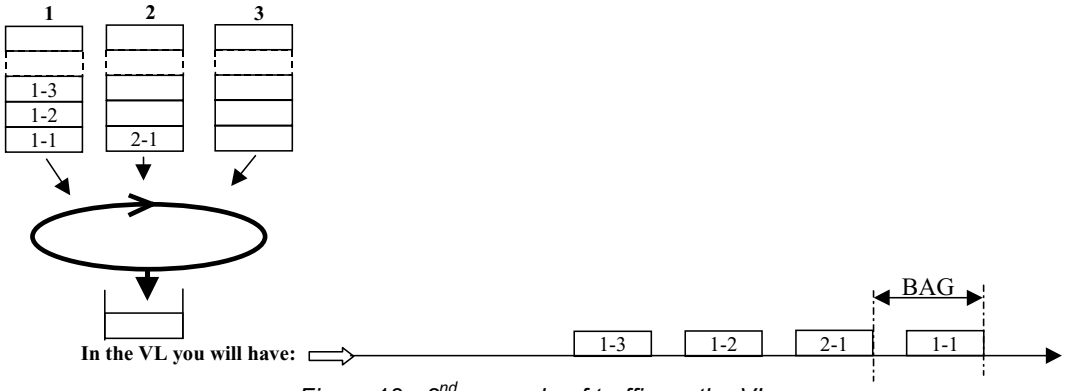


Figure 10 : 2nd example of traffic on the VL

A VL FIFO queue should be able to manage at least 4 Sub-VL FIFO queues.
Each Sub-VL FIFO queue should be read with an equal periodicity such that no more than one frame per BAG is sent to the main VL.

If a VL FIFO queue finds an empty Sub-VL FIFO queue, it should immediately read the next Sub-VL FIFO queue to use the maximum available bandwidth and not wait for frame availability on a given queue. This is defined as Work Conserving.

A Sub-VL FIFO queue should only be read by one VL FIFO queue. IP fragmentation should be performed at the Sub-VL FIFO level, if it is needed. This will avoid for example short sampling messages being delayed by long queuing messages. The round robin continues in the presence of IP fragmentation so that one fragment is fetched from one Sub-VL, and then a frame or fragment is taken from the following Sub-VL.
Sub-VLs are below the IP layer.

2.1.5 End System performance

The main goal of the system designer is to be able to use an AFDX End System in a deterministic way. By creating a measure of E/S performance, AFDX offers the system designer a reduced burden of certification, and a flexible solution with well defined constraints.

> ~~*What Roger replaced :*~~
> ~~*constraining as little as possible the system designers and to have a flexible system in order to reduce the certification process and proof of determinism.*~~

2.1.5.1 Latency

The latency in transmission is defined between the following points of measurement :

Start - last bit of an hosted application data is available to the communication services of the end-system;
End - last bit of the corresponding Ethernet frame is transmitted on the physical media.

Measurements of the technological latency are made with an empty mailbox with no conflicting resource access.

(See diagram below).

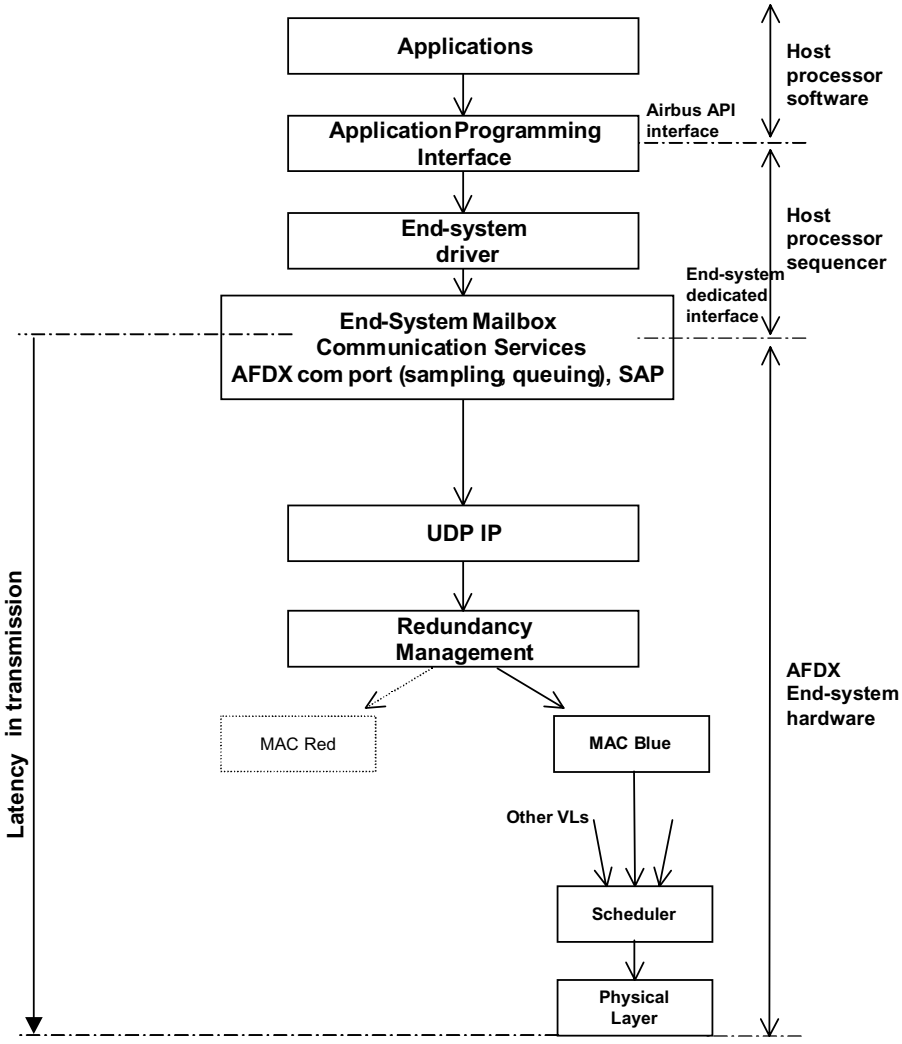


Figure 11 : Tx - Points of performance measurement

The technological latency of the end-system in transmission should be bounded and lower than 150 µs + frame delay.

COMMENTARY

It is assumed that the total latency of the E/S consists of technological latency (independent of traffic load) and configuration latency (depending on configuration and traffic load).

The "frame delay" is added to cover the time taken to deliver the frame to the physical layer.

The latency in reception is defined between the following points of measurement:

Start - last bit of an Ethernet frame is received on the physical media attachment.
End - last bit of the corresponding data is available to the end-system hosted application.

Measurements of the technological latency are made with an empty mailbox without any conflicting resource access.

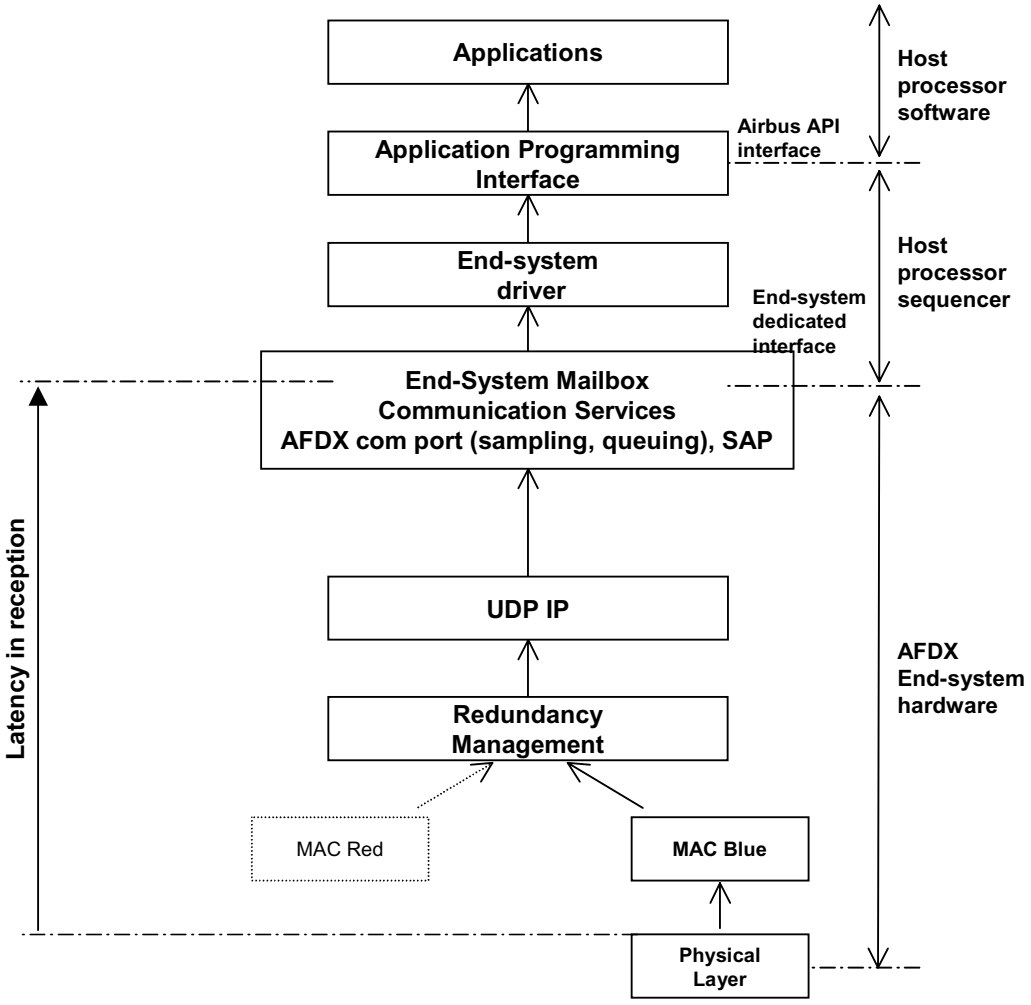


Figure 12 : Rx - Points of performance measurement

The technological latency of the end-system in reception should be bounded and lower than 150 µs.

2.1.5.2 MAC Constraints

In order not lose incoming frames during a burst and to fix the IFG in transmission, the MAC layer of the end-system should be able:

1- to process received frames back to back (frames can be received at full rate and appropriate (selected) frames are made available to the application.
2- To transmit frames back to back

For the shortest frame this corresponds to a maximum frame rate per attachment of :

64 bytes (frame) + 12 bytes (IFG) + 7 bytes (Preamble) + 1 byte (SFD) = 84 bytes to transmit at 100 Mbits/s. Equivalent to a duration of 6.72µs per frame (about 148800 frames per second).

COMMENTARY

This requirement could be relaxed for transmission. Nevertheless, the designer should very carefully consider the impact for compliance on maximum jitter in transmission.

This requirement is more stringent in terms of processing capabilities for the shortest frame (64 octets) with a minimum inter-frame gap (12 octets).

2.1.5.3 Jitter

In transmission, the maximum allowed jitter at the output of the end-system should comply with the following formulas:

$$\begin{cases} max\_jitter \le 40\mu s + \dfrac{\sum_{i \in \{\text{set of VLs}\}} \left(20 + L^{\max}\right) \times 8}{100} \\ max\_jitter \le 500\mu s \end{cases}$$

NOTE :
max_jitter is in micro-seconds ; L^{max} is in octets, 40µs is a typical minimum fixed technological jitter

According to the formula, the maximum allowed jitter will be lower for end-systems having few VLs and small frames sizes to process. In all cases, the jitter is bounded at 500µs to limit the impact on determinism for the whole network.

COMMENTARY

It is the system integrator's responsibility to determine that, for the chosen End System configuration and implementation, the 500 micro s limit is not exceeded.

These values are fundamental to the demonstration of determinism for AFDX, and can be used to evaluate the limitations of an end system. A non-optimised E/S will have bandwidth limitations resulting from limited processing capabilities.

~~Roger would remove this text in red subject to major surgery as below.~~
~~The two requirements below aim at defining a limit to the allowed latency for the end system in transmission.~~
~~The first requirement applies when the hosted application does not transmit bursts of data or long messages requiring fragmentation.~~
~~If the hosted application transmits bursts of data or, the transmission will be delayed according to the bandwidth limitation of the VL (traffic shaping). In this case, the second requirement is to be applied.~~

To treat mathematically the allowed latency in the End System in transmission, two limiting cases are defined.

For the first case it is assumed that the hosted applications transmit on a given VL evenly spaced data (no bursts) and that no data needs to be fragmented. Then, if the end-system has no other data to process on this virtual link, the total allowed latency for a given VL_i is :

$$MAX\_Latency_i \le BAG_i + \text{Max\_jitter} + \text{Technological\_Latency\_in\_transmission}$$

The second case applies when the hosted application transmits bursts of data or long messages requiring fragmentation. In this case, if the end-system has other data to process on this virtual link, the next data to transmit will be delayed. For the given VL_i in transmission, and if (p-1) frames are being processed, the maximum latency of the frame number p should be bounded according to the following formula:

$$MAX\_Latency_i(frame\_p) \le p \times BAG_i + \text{Max\_jitter} + \text{Technological\_Latency\_in\_transmission}$$

~~*What does the following convey in this context??? Roger*~~

the transmission will be delayed according to the bandwidth limitation of the VL (traffic shaping).

COMMENTARY

Some implementations could lead to an optimized solution regarding configuration latency. In all cases the values stated above should be adhered to.

2.1.6 MAC addressing

2.1.6.1 MAC destination address

In order to use the standard Ethernet frame, MAC group addresses should be used to send frames from End System to End System(s).

A Virtual Link should only be identified by the MAC destination address, and the MAC source address of AFDX frames should be the MAC unicast address used to identify the physical Ethernet interface.
A MAC destination address in the AFDX frame should be a Group and Locally Administered address and should be compliant with the following format .

48 bits	
Constant field *32 bits = "0000 0011 0000 0000 0000 0000 0000 0000"*	*Virtual Link Identifier* *16 bits*

Figure 13 : MAC Multicast Addressing Format

The Constant field value is always "0000 0011 0000 0000 0000 0000 0000 0000 " :

The first bit indicates the group address = 1
The second bit indicates the locally administered address = 1

2.1.6.2 MAC source address

The MAC unicast address should be an Individual and Locally Administered address compliant with IEEE 802.3. The structure of the address has to be defined by the network designer.

COMMENTARY

No specific source MAC address construction algorithm is recommended. Therefore, it may be necessary for AFDX End Systems to have a means to determine the address construction algorithm being used in the network they are placed in. For example, pin programming might be used as a means to indicate which address construction rule is used.

An example of MAC address structure is given in Appendix 2

2.1.7 Redundancy concept

End Systems communicate over multiple independent and redundant networks such that data flows are protected against the failure of any network component such as a link or a switch. The effect of this is to protect communication between End Systems against the loss of one complete network.

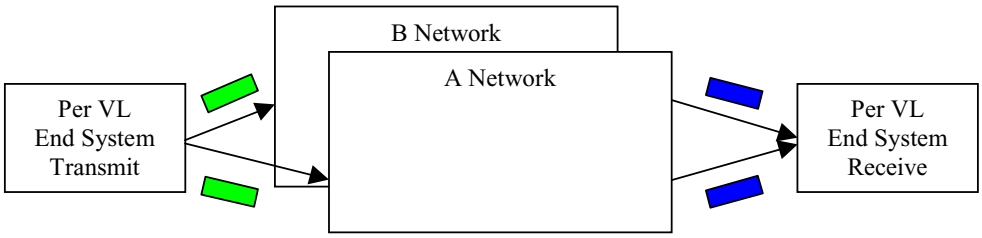


Figure 14 : Network Redundancy Concept

The Figure shows the basic concept for network redundancy. The redundancy scheme is operated on a per Virtual Link basis. A transmitting End-system and a receiving End-system communicate via a specific Virtual Link in the following manner :

An application in the transmitting End System prepares some data and passes it to the communications protocol stack. Here a sequence number field is added to each Ethernet frame, and the sequence numbers are incremented on each successive frame. The sequence number is added to enable the receive function to reconstruct a single ordered stream of packets without duplication before delivery to the receiving application. In this way the application is unaware of the underlying network redundancy, and a simple interface can be built between the communications stack and applications that utilize the network service.

In default mode each frame is sent across both of two networks. Upon reception, an algorithm in the communications stack (below IP layer) uses a "First Valid wins" policy. This means that the first frame to be received from either network with the next valid sequence number is accepted and passed up the stack to the receiving application. When the second frame is received with this sequence number, it is simply discarded.

As the flow of frames given in figure below indicates, RM (Redundancy Management) is placed after IC (Integrity Checking). Under fault-free network operation, the IC just passes on the frames that it has received from a network on to the RM. Again, the concern of the AFDX redundancy management is merely to eliminate frames that are redundant copies of frames that it has already passed on to the application.

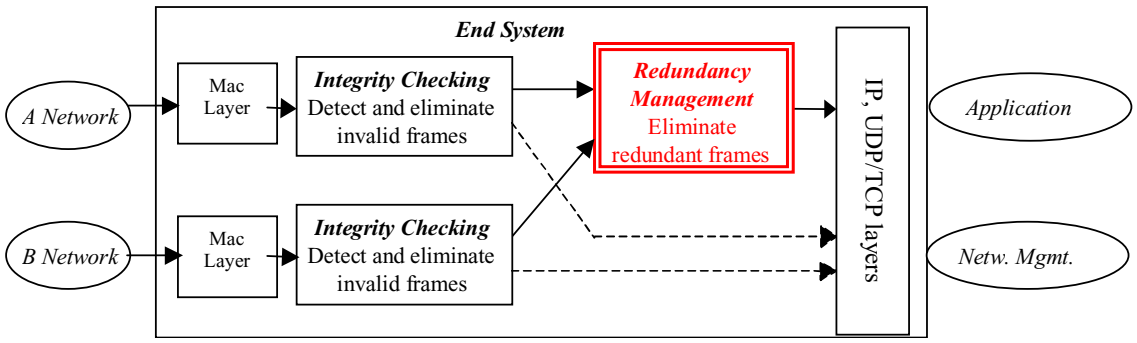


Figure 15 : Integrity Checking and Redundancy Management in the End System

Expected behaviour is illustrated in the following examples ("RMA" line refers to the frames transmitted to the application by the Redundancy Management Algorithm).

Example 1 : abnormal transmitted frame

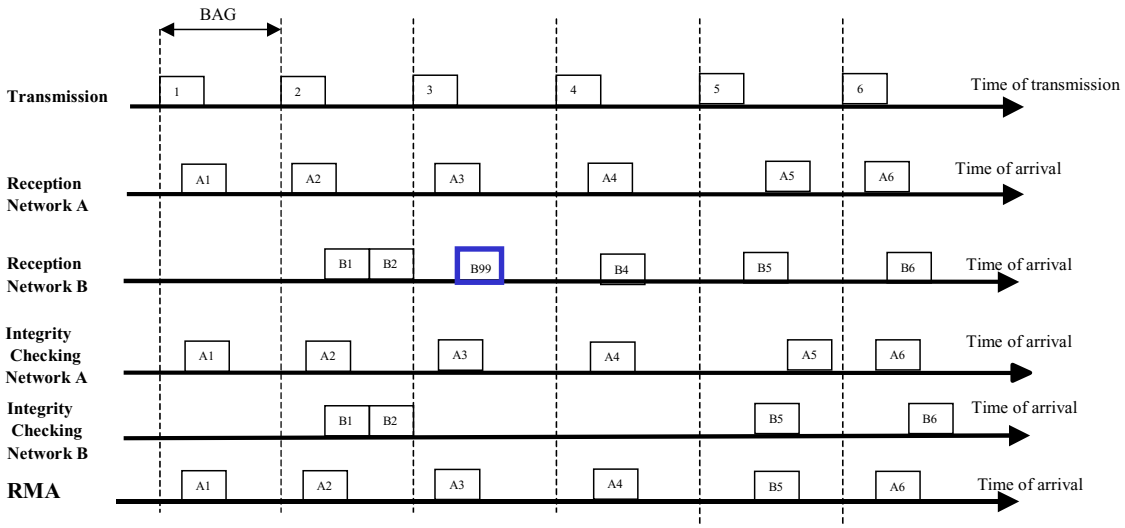


Figure 16 : Network B transmits an abnormal frame.

Redundancy management result : abnormal frame is not forwarded to application.

Example 2 : loss of a frame

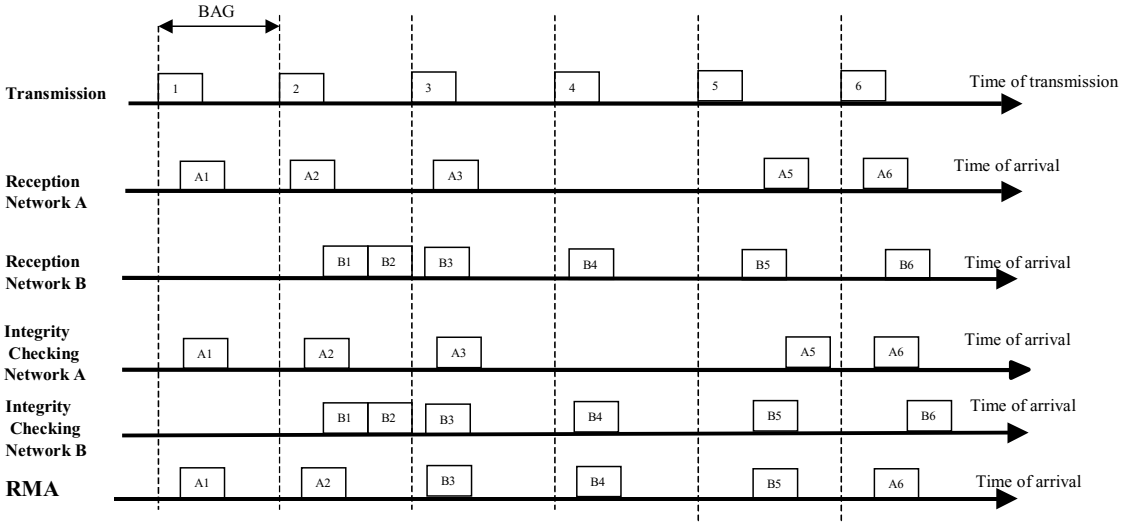
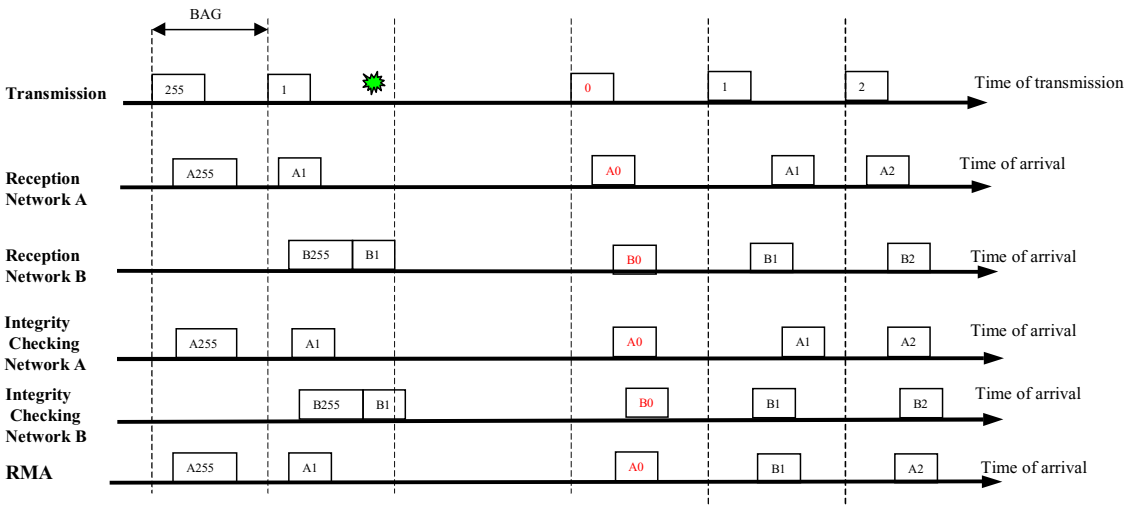


Figure 17 : A frame is lost on network A

Due to a Bit Error, frame "A4" is lost
Redundancy management result : The frame arriving on the B Network is accepted.

Example 3 : reset of the transmitter



: reset of the transmitting equipment

Figure 18 : Reset of the transmitting end system.

No frame is lost

Example 4 : babbling switch (stuck frame)

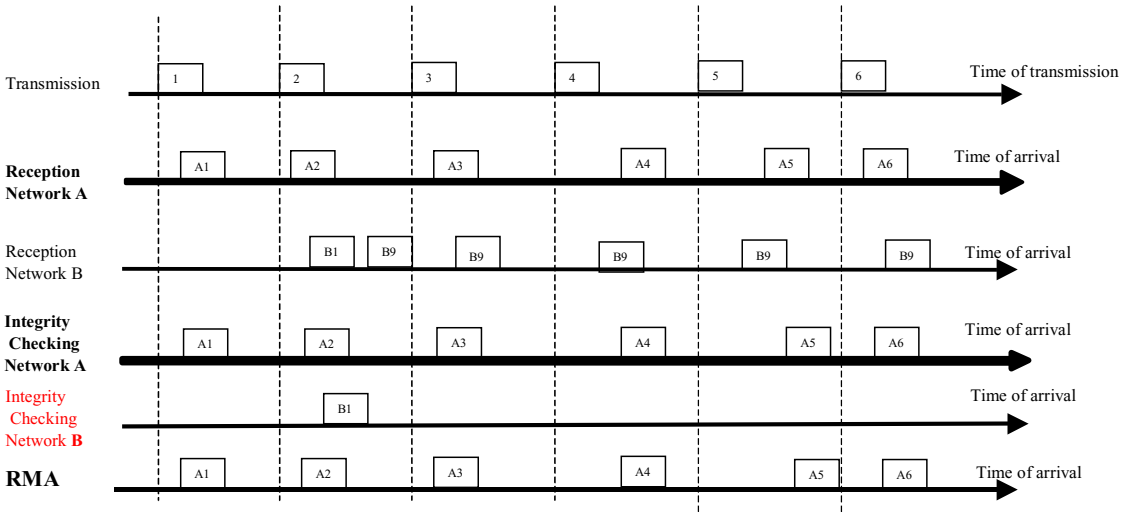


Figure 19 : Babbling on network B.

Frames are not transmitted due to Integrity Checking

2.1.7.1 Sequence numbers and the Sending End System

Per VL, the End System should add a sequence number for each transmitted frame on the AFDX network. The frame sequence number should be one octet long with a range of 0 to 255.

COMMENTARY

This is sufficiently big to detect redundant frames under normal operation, but still compact. For example, in the worst case with BAG = 1 ms, SkewMax = 5 ms, the maximum SN offset between two received frames is :

$Int[\frac{SkewMax}{BAG}]+2$ *= 7, which is well below 128 to take into account wrap around.*

For each VL, the sequence number should initially be set to 0. The sequence number is always assigned this initial value following a reset of the transmitting E/S.

The frame sequence number should be incremented by one for each consecutive frame of the same VL and wrap-around to ***1*** *following the value 255.*

COMMENTARY

Increment by one makes it possible to detect missing frames. Wrap-around to 1 allows the maximum range of sequence numbers, while reserving sn=0 for a reset condition. This improves integrity checking (see next section).

The frame sequence number should be located just before the MAC CRC field, as part of the MAC payload.

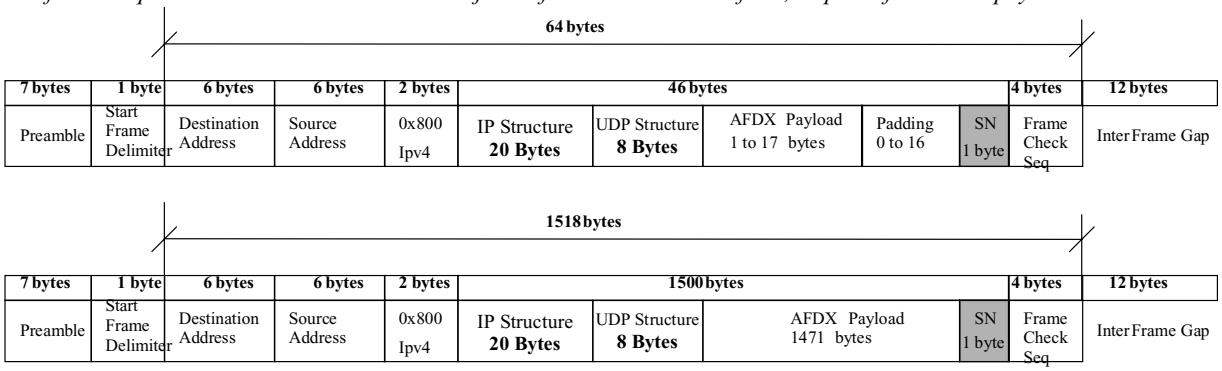


Figure 20 : Sequence number location

In order to simplify the algorithm on the receiving End System, redundant copies of a frame should be sent within a maximum time difference of 0.5 ms.

On a per VL basis, the ES should be able to send messages on either or both networks. This property should be configurable.

COMMENTARY

The system integrator is free to configure each Virtual Link with or without redundancy. If redundancy is turned off for any Virtual Link, there will clearly be impact on safety and availability which should be carefully assessed.

2.1.7.2 Sequence numbers and the Receiving End System

2.1.7.2.1 Integrity Checking

Under fault-free network operation, the Integrity Checking simply passes the frames that it has received on to the Redundancy Management, independently for each network. If there are faults (based on sequence number), the Integrity Checking has the task of eliminating invalid frames, and informs the network management accordingly. (Refer to section 2.1.7 *Redundancy concept for sequence number usage).*

For each network the Integrity Checking tests each frame for a sequence number in the interval :

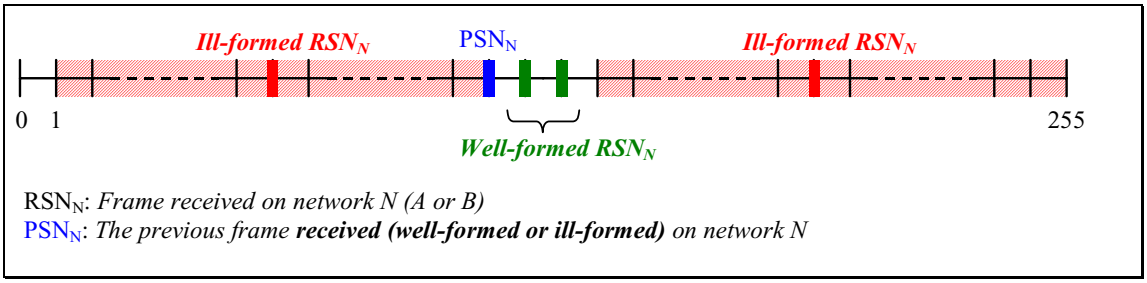
$[PSN_N"+"1, PSN_N"+"2]$

Where PSN_N is the sequence number of the previous frame received (but not necessarily forwarded) on this VL. The operator "+" takes the wrap-around of Sequence Numbers into account. So, for example if $PSN_N = 254$ then $PSN_N"+"1 = 255$ and $PSN_N"+"2 = 1$.

Frames, which do not meet these criteria, are discarded.

COMMENTARY

This function increases integrity robustness. E.g.: by eliminating stuck frames or single abnormal frames and reducing the impact of a babbling switch. Loss of one single frame is considered as a normal event due to a non-zero Bit Error Rate.



The Integrity Checking should accept the frame as valid in the following special cases :
1 - the Received Sequence Number (RSN_N) is equal to 0,
2 - the frame is the first frame received after any reset of the receiving ES.

COMMENTARY

These special cases improve the integrity of aperiodic data in particular. There would otherwise be systematic loss of a frame following a reset of either the transmitting E/S or the receiving E/S.

The sequence number 0 is transmitted only after a reset of the transmitting equipment.
It should be possible on a per VL basis to disable the integrity checking on both networks simultaneously through the configuration table. Disabling Integrity Checking allows the receiver to accept all packets from both Networks, A and B.

2.1.7.2.2 Redundancy management

The Redundancy Management (RM) assumes that the network is working properly and, in particular, the deterministic properties are verified.

Logic of RM including SkewMAX has to be defined.
RM configuration is generally based on the SkewMAX parameter: i.e. the maximum time between two copies received. This value depends on the network topology (number of switches crossed by the frames) and should be provided by the network manager. The SkewMax value (expressed in µs) is given by configuration per VL

On a per VL basis, the ES should be able to receive:

- a redundant VL and deliver to the application one of the redundant data (RM active);

- a redundant VL and deliver to the application both redundant data (RM not active);

- a non redundant VL on either attachment and submit data from it to the application (in this case, RM can be active or not).

This RM function should be configurable.

Redundant VL means that the same data is sent in frames through both network, A and B.
Non-redundant VL means that (possibly different) data are sent in frames through either network, A or B.

When the redundancy management is active, it should deliver the first of the two copies received.
Reset of any equipment involved in the communication (transmitting End system, receiving End system or the AFDX switch) should not affect this property. This "First Win" philosophy enables availability of the network in the case of one AFDX switch loss.

COMMENTARY

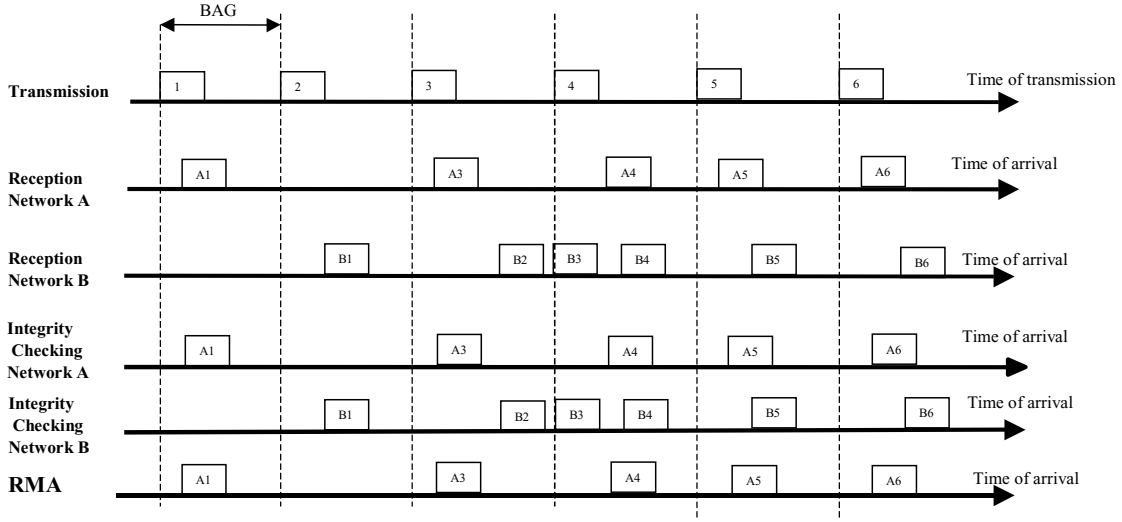
The hardware reset time is assumed to be larger than≠ SkewMax.

The Redundancy Management Algorithm should only use the redundancy of a frame as the criterion for rejection or acceptance. Integrity checking is a separate task, which has to be performed even if no redundancy is used.

For each VL at the receiver, the Redundancy Management function should ensure that frames are forwarded in an increasing sequence number order. This will still apply in case of resets and occasional lost frames.

The Redundancy Management function must forward only in-sequence frames, but reordering is not required. As a consequence, in some cases, the loss of one frame on one network could also lead to the loss of its copy.

For example (see figure below) the frame "A2" is lost on the A network, and the frame "A3" arrives on this network before the copy "B2" of the lost frame arrives on the B network. In this case the copy B2 will not be forwarded to the application despite being the first #2 packet received.



A frame is lost on the faster network (frame "A2" of network A).

Figure 21 : Loss of a frame

2.2 ~~Middle layer (above MAC) related~~ Interoperability and Communications Services ~~issues~~

2.2.1 Avionics services

The E/S provides different modes of transfer from an Avionics Application point of view with two types of ports:

1 - Communication port: Sampling or queuing modes (cf. ARINC 653)

2 - TFTP and communication with compliant networks via SAP (Service Access Points) ports

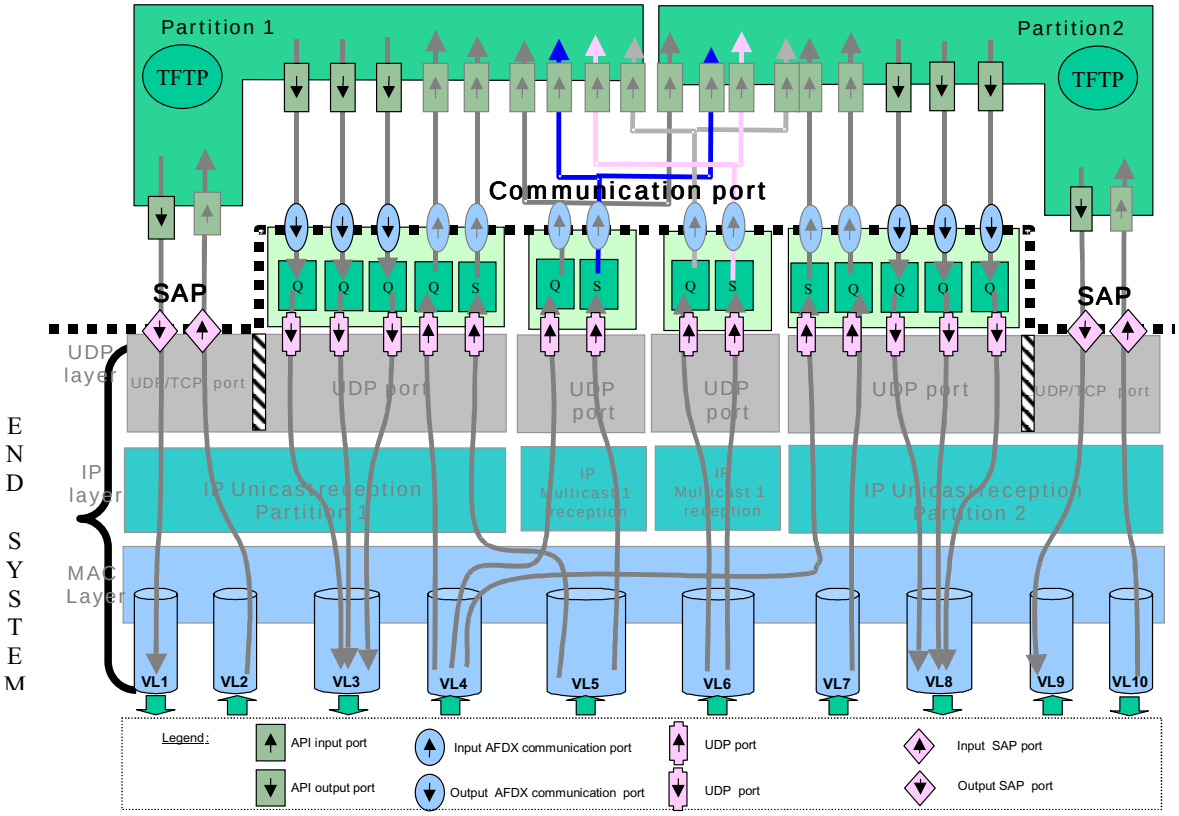


Figure 22 : Interface between Application and End System

The figure above describes an equipment which has two partitions (cf. ARINC 653 for a definition of a partition) and an End System. Each partition has an IP address. To communicate with partition, the End System uses two port types: Communication Port and SAP.

2.2.1.1 Communication Ports

Two types of services are provided by the End System via the Communication Ports: Sampling and queuing. UDP has been chosen for both services due to its relative efficiency.

2.2.1.1.1 Avionics Sampling Services

The End System should provide sampling services as defined in ARINC 653 (§2.3.5.6.1).

- Transmission

In order to avoid IP fragmentation and to reduce the delay introduced by the End System, the size of each sampling message should be less than or equal to the PAYLOAD size of the associated Virtual Link.

Sampling service should be based on multicast and unidirectional communication to send the message from one to one or several receptors.

COMMENTARY

The ES designer does not need to implement IP fragmentation for Sampling communication ports.

The sampling service is simple, connectionless without acknowledge. It does not add error control on packets for transmission and does not require flow control on transmitted messages in addition to the VL flow control. This transmission type is similar to services provided by classical ARINC 429 links.

- Reception
The last message stored in a particular sampling port should be able to be read by several applications (i.e. several application may subscribe to this sampling port).

A freshness indication is associated to each sampling message provided to the application. If the sampling message is shared by several host applications, a freshness indication should be given to each application.

2.2.1.1.2 Avionics Queuing service

The End System should provide queuing services to avionics application, as defined in ARINC 653 (§2.3.5.6.2)
The queuing service is simple, connectionless without acknowledge
The queuing service should be able to manage messages of different sizes for the same queuing Communication Port.
To guarantee the sequence of the messages, the queuing service should manage the messages with FIFO discipline in transmission and reception.
Each instance of Queuing Service should be able to manage up to 8 k octets of application data (IP fragmentation is thus needed).

COMMENTARY

A connectionless without acknowledge queuing service is acceptable for a large number of communications thanks to the low probability of frame loss in redundant AFDX

- Transmission
If a buffer overflow occurs in transmission, an error indication should be sent to the transmitting application and the frame should be discarded.

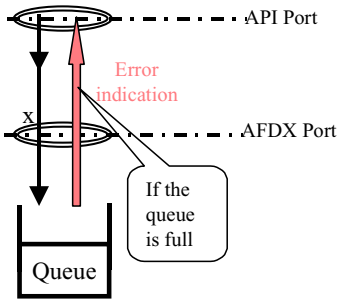


Figure 23 : Error indication for Tx buffer overflow

- Reception
In case of fragmentation, no data should be presented to an application in the queuing services FIFO until that data has been reassembled.
If a buffer overflow occurs in reception, an error message should be sent to the receiving application and the frame should be discarded.

2.2.1.2 SAP port

2.2.1.2.1 Services to compliant network

An E/S can be a Service Access Point with the following characteristics :
The SAP ports could be used for communications within the AFDX network.
Access to the compliant network can be through either a gateway or a router, as part of the E/S design.
The E/S should provide UDP services to communicate with Compliant Network.
Each instance of the UDP Service Access Point should handle up to 8 k octets.
As an option, TCP could be used below SAP ports.

To communicate with a Compliant Network, a transmitting E/S has the ability to specify the destination address : IP address and port number. For this purpose, these addresses are made available when the E/S receives a request from the compliant network.

2.2.1.2.2 SAP port error management

Degradation of the Quality of Service in a SAP port is monitored at the receiver. If receiver buffer overflow occurs in, an error message is sent to the receiving application, and the frame should be discarded.

2.2.1.2.3 File Transfer services

The Trivial File Transfer Protocol "TFTP" should be used to transfer files.
The specification of TFTP is defined in the following RFCs:

RFC	*Title*	*Category*
783	*The TFTP protocol (Revision 2)*	*Standard, Updated by RFC 1350*
1123	*Requirements for internet Hosts Application and support*	*Standard*
1350	*TFTP protocol (Revision 2)*	*Standard*
2347	*TFTP option extension*	*Standards track, Updates 1350*
2348	*TFTP Blocksize Option*	*Standards track, Updates 1350*
2349	*TFTP Timeout Interval and Transfer Size Option*	*Standards track, Updates 1350*
1785	*TFTP option negotiation analysis*	*Informational, Updates 1350*

Each instance of File Transfer service should be able to manage up to 8 k octets blocks.

2.2.1.3 Interface E/S and API or APEX

There is no standard mapping between API/APEX and the ports of the E/S. These requirements will be written into the specification of a particular equipment.

One communication or SAP port will be able to receive data from one and only one external, transmitting End System port (API port or APEX port).

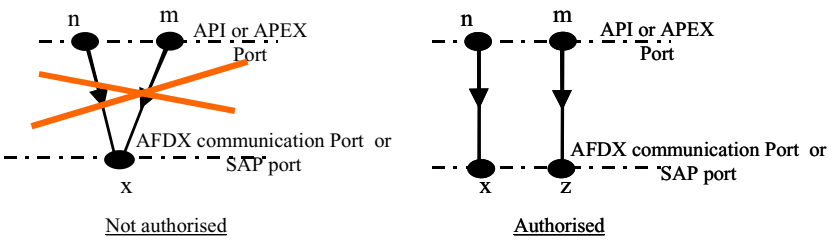


Figure 24 : External Transmission, ports not shared

One communication or SAP port will be able to transmit data to one or several external receiving port(s), (API ports or APEX ports).

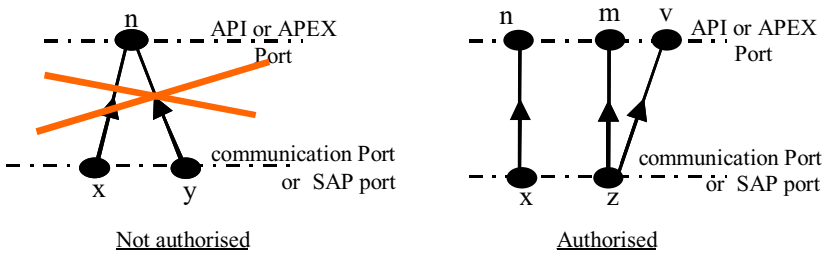


Figure 25 : External Reception, ports are shared

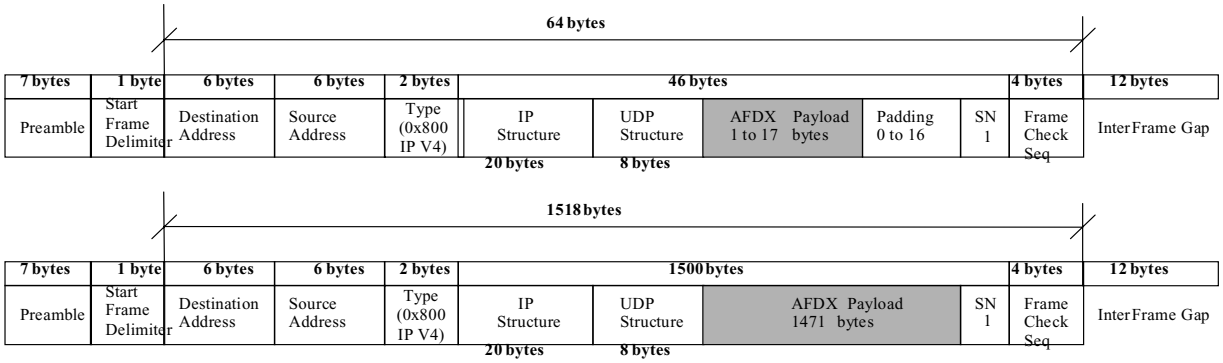
2.2.2 Addressing

2.2.2.1 Introduction

A data flow is uniquely identified within AFDX network by the set of UDP/TCP destination port, IP destination address, MAC destination address and the physical Ethernet connection(s) of the receiving E/S.

Frame based filtering is done by analysis of destination (UDP/TCP, IP, MAC) addresses and physical Ethernet connections.

2.2.2.2 Structure of an AFDX frame without fragmentation



Minimum and maximum frames are illustrated

Figure 26 : Structure of an AFDX Frame

2.2.2.2.1 Example of Addressing principle

In the following example, End System 1 has three Virtual Links: VL1, VL2 and VL3.

Partition 1 of End System 1 has access to one Virtual Link: VL1
Partition 2 of End System 1 has access to two Virtual Links: VL2 and VL3.

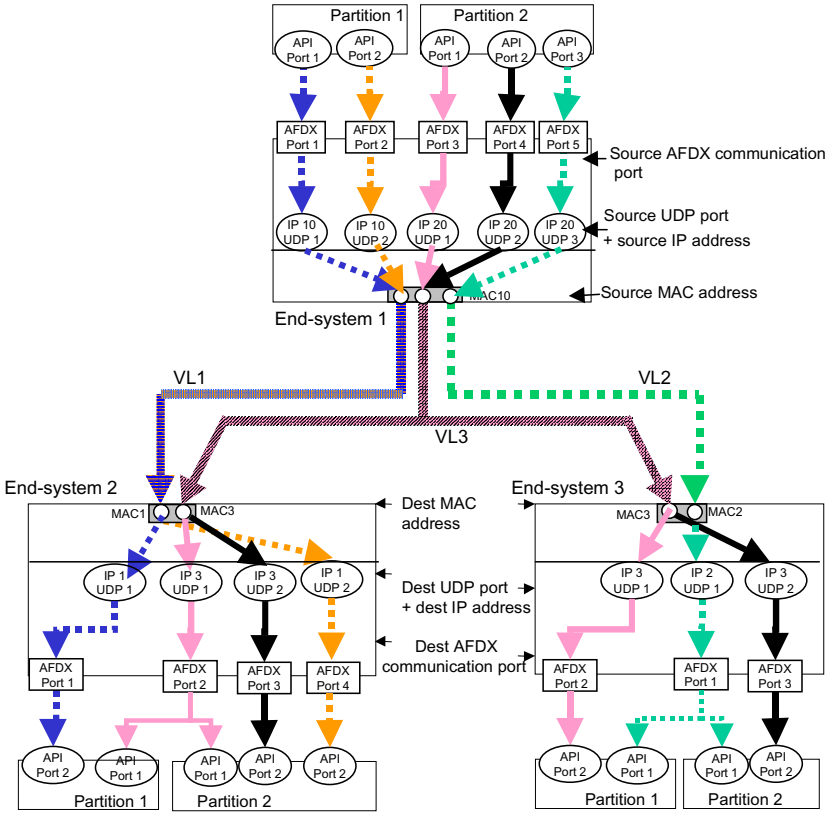


Figure 27 : Example of Addressing

Below are the address tables for each E/S:

Transmit table of End-system 1 :

Src AFDX communication Port	*Src partition*	*Src UDP*	*Src IP*	*Src MAC*	*Dest UDP*	*Dest IP*	*Dest MAC*
AFDX Port 1	*Partition 1*	*UDP1*	*IP10*	*MAC10*	*UDP1*	*IP1*	*MAC1 (VL1)*
AFDX Port 2	*Partition 1*	*UDP2*	*IP10*	*MAC10*	*UDP2*	*IP1*	*MAC1 (VL1)*
AFDX Port 3	*Partition 2*	*UDP1*	*IP20*	*MAC10*	*UDP1*	*IP3*	*MAC3 (VL3)*
AFDX Port 4	*Partition 2*	*UDP2*	*IP20*	*MAC10*	*UDP2*	*IP3*	*MAC3 (VL3)*
AFDX Port 5	*Partition 2*	*UDP3*	*IP20*	*MAC10*	*UDP1*	*IP2*	*MAC2 (VL2)*

Receive table of End-system 2 :

Dest AFXD Communication port(s)	*Dest partition(s)*	*Src UDP*	*Src IP*	*Src MAC*	*Dest UDP*	*Dest IP*	*Dest MAC*
AFDX Port 1	*Partition 1*	*UDP1*	*IP10*	*MAC10*	*UDP1*	*IP1*	*MAC1 (VL1)*
AFDX Port 4	*Partition 2*	*UDP2*	*IP10*	*MAC10*	*UDP2*	*IP1*	*MAC1 (VL1)*
AFDX Port 2	*Partition 1 and 2*	*UDP1*	*IP20*	*MAC10*	*UDP1*	*IP3*	*MAC3 (VL3)*
AFDX Port 3	*Partition 2*	*UDP2*	*IP20*	*MAC10*	*UDP2*	*IP3*	*MAC3 (VL3)*

Receive table of End-system 3 :

Dest AFDX Communication port(s)	*Dest partition(s)*	*Src UDP*	*Src IP*	*Src MAC*	*Dest UDP*	*Dest IP*	*Dest MAC*
AFDX Port 2	*Partition 1*	*UDP1*	*IP20*	*MAC10*	*UDP1*	*IP3*	*MAC3 (VL3)*
AFDX Port 3	*Partition 2*	*UDP2*	*IP20*	*MAC10*	*UDP2*	*IP3*	*MAC3 (VL3)*
AFDX Port 1	*Partition 1 and 2*	*UDP3*	*IP20*	*MAC10*	*UDP1*	*IP2*	*MAC2 (VL2)*

The next drawing presents the physical topology for this example :

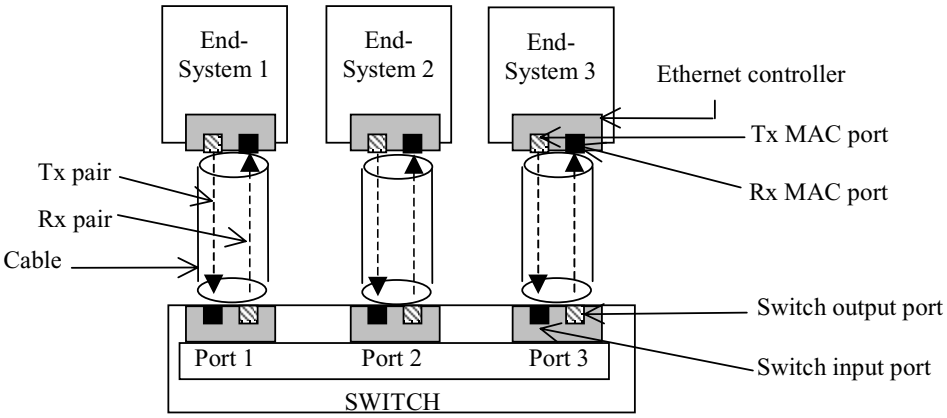


Figure 28 : Example Physical topology

Switch forwarding table :

Input port	*MAC destination field of the received frames*	*Output ports*
1	*MAC1 (VL1)*	*2*
1	*MAC2 (VL2)*	*3*
1	*MAC3 (VL3)*	*2 and 3*

COMMENTARY

MAC address should be understood as potentially unicast or multicast Ethernet address.

2.2.2.3 Identification for end to end communication

Peer to peer communications are identified in each frame by UDP Source Port + Source IP + Destination MAC (VL identification) + Destination IP + UDP Destination Port.
In the AFDX network, this quintuplet provides a unique identification for each message.

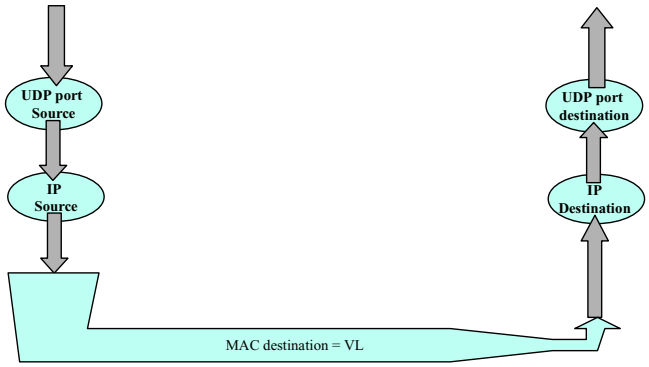


Figure 29 : Message Identification Concept

For a source IP, there will be several source UDP/TCP ports. For a destination IP there will be several destination UDP/TCP ports.

In the below example, we have 3 messages identified by 3 quintuplets.

Message 1 => UDP Source Port x + Source IP + destination Mac + destination IP + UDP destination port n
Message 2 => UDP Source Port y + Source IP + destination Mac + destination IP + UDP destination port m
Message 3 => UDP Source Port z + Source IP + destination Mac + destination IP + UDP destination port v

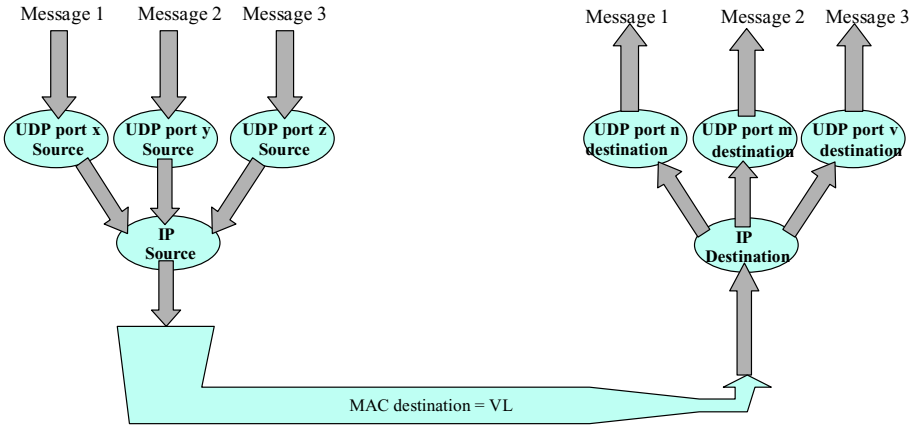


Figure 30 : Unique message identification

2.2.2.3.1 Intra-AFDX Communication

2.2.2.3.2 End to end communications, which remain within the AFDX network, can be regarded as Intra-AFDX.

The principal characteristic of Intra-AFDX communication is the fact that for each message, the addressing is statically defined.

For unidirectional communications:
AFDX communication ports are defined through UDP ports. Such ports can be either transmitters or receivers.

AFDX communication ports are characterised by the Sampling and the Queuing services.

For bi-directional communications:
Use may be made of the TFTP and TCP protocols.

They are two possibilities:

A) *Utilisation of the SAP (Service Access Point) ports. These are linked with UDP or TCP ports, and each SAP can be a transmitter or a receiver. To obtain a bi-directional communication, two SAPs should be used (e.g.: SAP 30 000 Tx and SAP 30 000 Rx).*

In this case, two quintuplets are defined, one for each direction of communication.
It is also recommended that SAP ports be used for full compliance to Internet protocols: e.g. port 69 is used for TFTP.

B) *Utilisation of the conventional AFDX communication ports. Bi-directional communication, at a single E/S will require two AFDX communication ports : one transmitter and one receiver, (e.g.: AFDX Com port 15 000 Tx and AFDX Com port 15 000 Rx).*

For a bi-directional communication, the ports will be use in queuing mode.

2.2.2.3.3 Extra-AFDX Communications

This describes communications between the AFDX network and a compliant network.
Two modes of communication are defined :

Unidirectional communications:

This will always be from a transmitting E/S to a compliant network, and use can be made of the conventional AFDX communication port with a UDP port link. The implication is that the E/S configuration table will contain the destination IP and port number. Hence there will be a statically defined quintuplet for addressing.

Bi-directional communications:

Can utilise TCP, TFTP, SNMP, 615A protocols.
SAP ports are used, and as for Intra-AFDX communications, each SAP is either a transmitter or a receiver. Two SAPs would be used to obtain a bi-directional communication.

The receiving SAP can pass to the application the IP address and the UDP/TCP port identification of the sources in the compliant network.

The transmitting SAP can pass the IP address and the UDP/TCP port identification of the destination in the compliant network.

2.2.2.4 IP addressing format

2.2.2.4.1 IP Source address

The IP source address should be used to identify the transmitting partition associated with the End System.

The IP destination address should be used by an End System to forward IP packets to one ore more destination End System(s).

The IP address should be Class A and private Internet unicast address (First 8 bits should be '0000 1010').
An example of IP Unicast Addressing Format is given in Appendix 3.

The IP source address in the IP header of the AFDX frame should be an IP unicast address used to identify the transmitter .

2.2.2.4.2 IP Destination address

The IP destination address in the IP header of the AFDX frame should be :
EITHER the IP Unicast address to identify the target subscriber
OR an IP Multicast address compliant to the following format :

IP Addressing Format *32 bits*		
4 bits	*28 bits*	
Class D *"1110"*	*IP Multicast Identifier*	
	Constant field *12 bits = "0000 1110 0000"*	*Virtual Link Identifier* *16 bits*

Figure 31 : IP Multicast Addressing Format

2.2.2.4.3 AFDX communication port, SAP and UDP/TCP addressing format

For Intra and Extra AFDX communications, there are two interfaces between the End System and the applications : AFDX communication ports and SAP ports

AFDX Communication port

It is characterised by:

Unidirectional access: Transmission (Tx) or reception (Rx).

Sampling or queuing mode: sampling and queuing have signification only in reception.

In transmission, there is only one link between "AFDX Communication port" and the quintuplet (UDP Source Port, Source IP, destination Mac, destination IP, UDP destination port). The "AFDX Communication port" belongs to an unique application.

In reception, there is only one link between "AFDX Communication port" and the quintuplet (UDP Source Port, Source IP, destination Mac, destination IP, UDP destination port). The "AFDX Communication port" may be accessed by different applications.

The transmission and the reception path is frozen by configuration, it can be represented by the following figure.

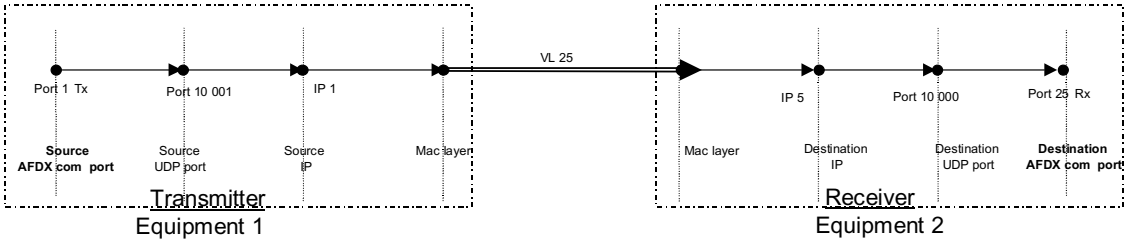


Figure 32 : AFDX communication port

SAP port.

It is characterized by :

A SAP port is a UDP or TCP port, this term is used to differentiate them from the "AFDX communication ports"

Unidirectional access: Transmission or reception

The possible coupling of two SAP ports to identify bi-directional communication e.g Port 500 TX and 500 Rx.

In transmission, the SAP port uses by configuration a frozen quadruplet (UDP Port source address, , IP source address, Mac Source address, MAC destination address (VL identification)), the destination IP and destination UDP (or TCP) port are given by the application.

In reception the SAP is linked only to one destination Port + destination IP + destination Mac (VL identification) + source MAC, the source IP and port are delivered by the End System to the application.

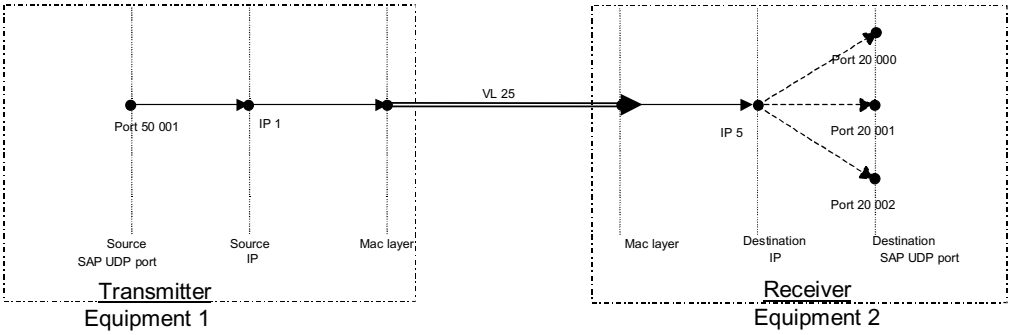


Figure 33 : SAP port using UDP

2.2.2.5 AFDX communication ports.

In transmission, an AFDX Communication Port should only be linked to a single set UDP port source, IP source, VL (MAC destination), IP destination and UDP destination.

In reception, an AFDX Communication Port should only be linked to a single set (unique UDP port destination, IP destination, VL (MAC destination)) and additionally to the Ethernet physical interface if the redundancy management is disable.

2.2.2.6 SAP ports.

In transmission, a SAP Port should be only linked to a single set (UDP port source, IP source and VL (MAC destination)).
In reception, a SAP Port should be only linked to a single set (UDP/TCP port destination, IP destination and VL (MAC destination))

The UDP/TCP port number should identify the Service Access Point.
In reception, the Service Access Point should make available the IP source and the UDP/TCP Source to the receiving application.

In transmission, the Service Access Point should permit the application to specify the IP address and the UDP/TCP port of the destination application.

2.2.2.7 <u>Allocation of the SAP and AFDX communication port numbers</u>

ARINC has defined port number allocation in the document "ARINC 664 part 4 Internet based address structures and assigned numbers" (draft August 20. 12001), the following table gives this allocation and the choice for the AFDX :

0 – 1023	*Administered by ICANN "Well-known" port number*	*Administered by ICANN "Well-known" port number*
1 024 – 16 383	*Registered by ICANN A664 assigned*	*Assigned by network manager*
16 384 – 32 767	*Registered by ICANN System integrator Or User defined*	
32 768 – 65 535	*Registered by ICANN Recommended for temporary port assignment*	

For each IP unicast or multicast IP, the repartition of the port allocation range is the following

AFDX Communication port	*AFDX AFDX Compliant AFDX network*	*1 024 – 65 535*	*Used for sampling and queuing communications*
SAP	*AFDX AFDX Compliant AFDX network*	*0 – 1023*	*Used for standard communications e.g Port 69 to open a TFTP, Data loading (ARINC 615A), SNMP, etc..*
	AFDX AFDX Compliant AFDX network	*1 024 – 65 535*	*Used for bi-directional communication: specific TFTP etc..,*

2.2.3 TFTP Example

This example describes the utilisation of TFTP to send a file from LRU 1 to LRU 2. For this, two VLs are defined: VL1 and VL2.
VL1: LRU1 to LRU2, VL2: LRU2 to LRU1
In the initialisation phase.

① The transfer is initiated by LRU 1, which sends a request from source port 45 000 on destination port 69, dedicated to TFTP in the LRU2.

② LRU 2 actives a TFTP session, which responds to the request by sending a frame to port 45000 of LRU 1. It indicates the port chosen to receive the transfer (Port 47 000).

③ At this moment the connection is established. The LRU 1 can send the file in data packets.
The transfer of the data packet from LRU 1 to LRU 2 uses source port 45 000 and destination port 47 000.
④ The acknowledgement of each data packet is sent by LRU2 which uses source port 47 000 and for destination port 45 000.

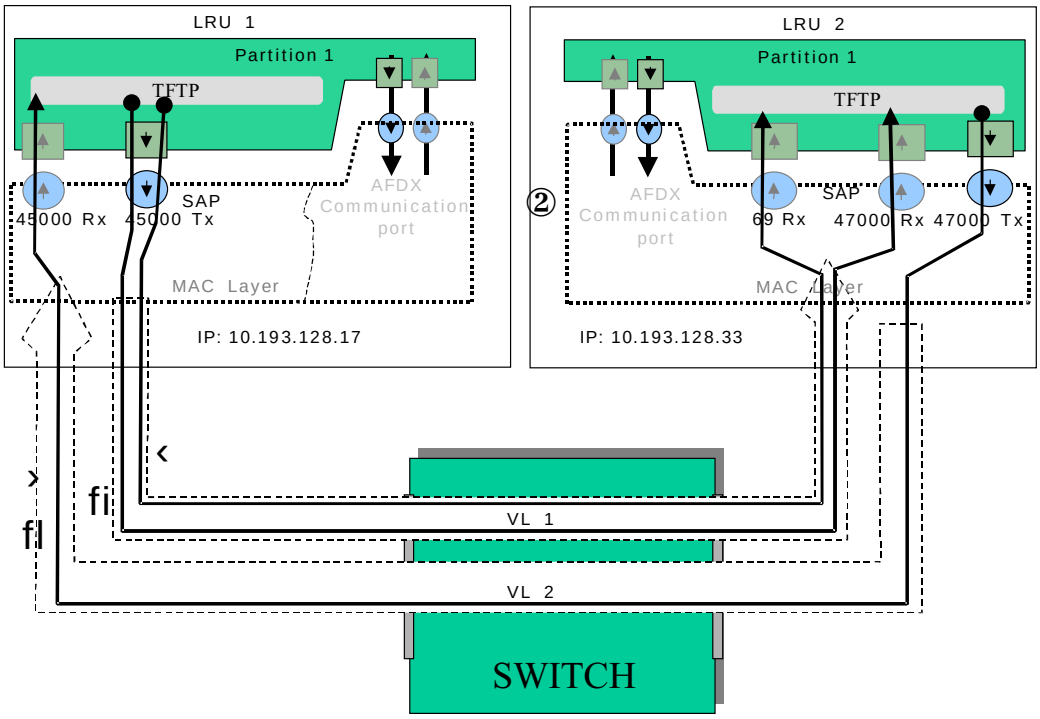


Figure 34 : Example of TFTP communication in the AFDX network

2.2.4 E/S Communication stack

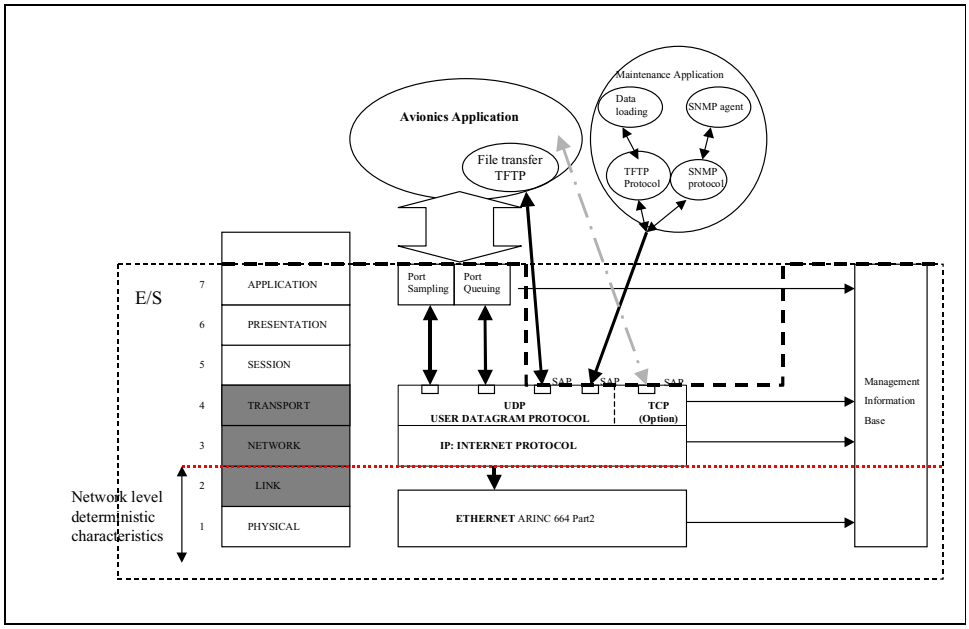


Figure 35 : ES Stack

2.2.4.1 E/S MAC Profile

The Datalink layer of the E/S should be based on the use of Full-duplex Ethernet links as defined in IEEE 802.3 standard.
Any Ethernet Frame generated by the End System should be compliant with IEEE 802.3 standard.
All output interfaces will continue to transmit, even in the case of a physical layer link failure.

COMMENTARY

This avoids the sending of old buffered frames after a long link loss (following a switch reset or intermittent physical layer failures for instance). It may also help in avoiding propagation of a failure from the switch to the End System as well as between switches.

A maximum AFDX frame length should be defined on a per VL basis.

In reception, if the AFDX frame format and FCS (CRC) are valid (without preamble and Start Frame Delimiter fields) the frame should be forwarded to the upper layer)

2.2.4.2 E/S IP profile

The packet structure version should be IP V4.

COMMENTARY

IPv4 packet structure should be compliant with:

4 bits	*4 bits*	*8 bits*	*16 bits*	*16 bits*	*3 bits*	*13 bits*	*8 bits*	*8 bits*	*16 bits*	*32 bits*	*32 bits*	*1-1479 bytes*
Version	*IHL*	*Type of service*	*Total length*	*Fragment identification*	*Control flag*	*Fragment offset*	*Time to live*	*Protocol*	*Header checksum*	*IP Source address*	*IP Destination address*	*IP payload*

Ordinarily, in the IPv4 packet structure, the Total Length field should range from 21 to 1500. bytes. In AFDX, this range is from 21 to 1499 due to Sequence Number (see redundancy management function).

COMMENTARY

The Total length field does not take into account the Sequence Number.

Next section is to be deleted if redundant with the table.

In the IPv4 packet structure, Internet Header Length(IHL) must be equal to 5 in decimal notation
In the IPv4 packet structure, Type Of Service should be 0.
In the IPv4 packet structure, the Total Length field should range from 21 to 1499 bytes.
In the reassembly phase, if the fragments of IP datagram are received out of order, it should be considered as a data link error, and the IP datagram should be silently discarded (The application in transmission and reception should not know this error).

The IP fragmentation/Reassembly should not use time-out condition.

Reassembly attempt should be given up if an unexpected datagram arrives in the same UDP/TCP port.

For a given VL IP destination, the End System should be capable of reassembling at least 4 datagrams concurrently.

In the IPv4 packet structure, The Fragment identification should be used to identify datagram fragments for reassembly.

In the IPv4 packet structure, the first bit of the control flag field should be 0.

In the IPv4 packet structure, the More Fragments flag bit should be set if the datagram is not the last fragment.

In the IPv4 packet structure, the Don't Fragments flag bit should be set if fragmentation is forbidden in parameter table.

In the IPv4 packet structure, the fragment offset field should be used to indicate where, in the datagram, this fragment begins.

In the IPv4 packet structure, the Time to live field should be equal to 1 (decimal notation) in transmission, and should be not checked in reception.

In the IPv4 packet structure, the protocol field should be 1, 6 or 17.

The E/S should be able to process an ICMP echo request.

ICMP structure should be compliant with:

8 bits	*8 bits*	*16 bits*	*16 bits*	*16 bits*	*1 - 64 bytes*
Type	*Code*	*Checksum*	*Identifier*	*Sequence Number*	*Data*

In the ICMP structure, the Type field should be:
8 for an echo message
0 for an echo reply message
In the ICMP structure, the Code field should be 0.
In the ICMP structure, the checksum should be the 16 bit's one's complement of the one's complement sum of the ICMP message starting with the ICMP type field.
In the ICMP structure, the Identifier field received in the echo message should be returned in the echo reply message.
In the ICMP structure, the data received in the echo message should be returned in the echo reply message.
In the ICMP structure, the data field should be in the range 1 to 64 bytes.
In the IPv4 packet structure, the header checksum field should be the 16 bit's one's complement of the one's complement sum of all 16 bit words in the header.
The time required to respond to an ICMP request (ping) should be less than 100 milliseconds. This time will be measured from ICMP layer.

~~*If echo client service is implemented, the answer to an ICMP echo request should be received in less than 1 second to avoid a Time out and a "destination unreachable" message.*~~

~~***Table was inserted here, now move to Appendix ???***~~

3 Switch Specification

4 System Issues

Fault effect isolation
determinism algorithm,
Contribution of ES specs (first part) to determinism
Principle of determinism proof (some examples…).
modeling,
network management,
configuration –dataloading- ,
default configuration-,
interconnection with compliant networks

APPENDIX 1: Performance characteristics

Performance is defined as a percentage of the maximum that the ES is able to handle.
The maximum throughput ("wire speed") corresponds to back-to-back frames.

The actual performance is measured by the time necessary to process all frames received during a 1 ms burst of back-to-back frames (see figure below).

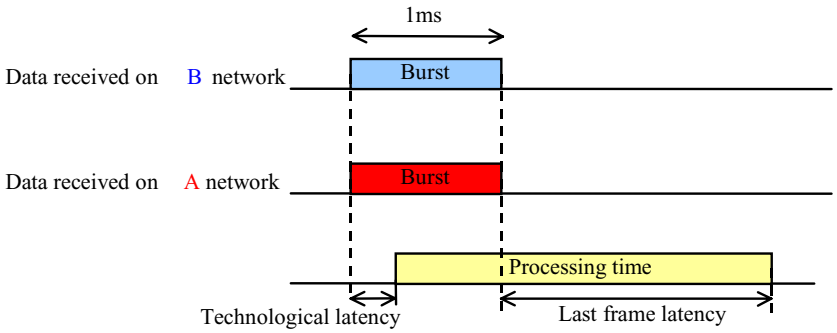


Figure 36 : 1 ms burst of back-to-back frames

The redundancy management algorithm is not disabled. It is assumed that data of B and A networks belong to different VLs and thus are both to be provided to the relevant applications.

The performance (as a percentage of wire-speed capability) is given by the formula:

$$Performance = 100 \cdot \frac{1}{\text{last frame latency} - \text{technological latency} + 1} = 100. \frac{1}{processing\_time}$$

The "last frame latency" and the "technological latency" are expressed in ms.

The "last frame latency" corresponds to the latency of the last frame of the 1 ms bursts received, whether this frame corresponds to the B or the A network.
The points of measurement are the same as those defined for the latency in reception (see above).

This performance may depend on several parameters such as frame size, queuing or sampling data, end-system activity in transmission, etc.

The End System designer should provide information on the processing capabilities in transmission and reception of the end-system. As a guide, the following invariant parameters should be provided:

Volume in transmission and reception given by:
number of ports
number of VLs
frame size
size of IP multicast group per VL
Speed in transmission given by :
Latency
Frame rate
Speed in reception given by :
Latency

ATTACHMENT 7-1

Traffic profile

A way to measure the performance in reception is proposed at the beginning of this section.

Services given by :
IP fragmentation and re-assembly
ICMP

Depending on the design of the end-system, the performance characteristics may vary with the actual configuration (e.g. number of VLs, use of fragmentation or not, …).

The End System designer should provide information regarding the data storage capacity in reception of the E/S.

APPENDIX 2: An example of MAC address structure

Ethernet MAC controller Identification *48 bits*						
Constant field: *24 bits* *"0000 0010 0000 0000 0000 0000 "*	*Network_ID 8 bits*		*Equipment_ID 8 bits*		*Interface _ID* *3 bits*	*Constant field: 5 bits* *"00000"*
	Constant field : 4 bits *"0000"*	*Domain ID* *4 bits*	*Side ID* *3 bits*	*Location ID* *5 bits*		

Figure 37 : MAC Unicast Addressing Format

*The **Constant field:***

*The **Constant** field is set to "0010 0000 0000 0000 0000 0000 " :*

The first bit indicates the Individual Address = 0
The second bit indicates the locally administered address = 1

*The **Network_ID** consists of 2 fields:*

*The **Constant** field has no signification, it is set to "0000" :*

Constant field	*Meaning*
0 0 0 0	*No signification*

*The **Equipment_ID** comprises two fields:*

*The **Side ID** field and the*

*The **Location ID** field: requirement AS-515-SM1-103.*

*The **Interface_ID** indicates on which redundant AFDX network(s) the Ethernet MAC controller is connected to.*

Interface_ID	*Meaning*
0 0 0	*Not used*
0 0 1	*The Ethernet MAC controller is connected to the network A*
0 1 0	*The Ethernet MAC controller is connected to the network B*
0 1 1	*Not used*
1 0 0	*Not used*
1 0 1	*Not used*
1 1 0	*Nor used*
1 1 1	*Not used*

APPENDIX 3: An example of IP Unicast Addressing Format

IP Unicast Addressing Format (source or unicast destination) *32 bits*							
Class A *1 bit*	*Private IP address* *7 bits*	*Network ID* *8 bits*		*Equipment ID* *8 bits*		*Partition ID* *8 bits*	
"0"	*"0001010"*	*Constant field* *"0000"*	*Domain ID* *4 bits*	*Side ID* *3 bits*	*Location ID* *5 bits*	*Constant field* *3 bits*	*Partition ID* *5 bits*

Figure 38 : IP Unicast Addressing Format

*The **Network ID** comprises 2 fields:*
*The **Constant** field has not signification, the value is set to "0000" :*

Constant filed	*Meaning*
0 00 0	*No signification*

*The **Domain ID** field:*

*The **Equipment_ID** comprises two fields:*
*The **Side ID** field*
*The **Location ID** field:*

*The **Partition ID** comprises two fields :*
*The **Constant** field has no signification, it is set to "000" :*

Constant filed	*Meaning*
0 0 0	*No signification*

*The **Partition_ID** indicates the source partition relatively to the equipment.*

Partition_ID	*Meaning*
0 0 0 0 0	*Basic software of the equipment which provides partitioning capability* ***or*** *Equipment without partitioning knowledge*
1 1 1 1 1	*Reserved*
All other values will be used to identify the partition relatively to the equipment	

APPENDIX 4: End System Identification

TBC whether this should be in the document.

The equipment which hosts one or several End System(s) is identified in the network by :
- *Domain ID (= Sub Network ID)*
- *Side ID*
- *Location ID*

The Domain ID indicates the domain (sub-network) to which the equipment belongs.
The Side ID indicates the side of the equipment within the Domain.
The Location ID indicates the position of the equipment relatively to the side in the domain.

Domain ID, Side ID and the Location ID are used to build the IP and MAC addresses.

The Domain ID shall be coded with 4 bits.
The Domain ID will use the following values :

Domain ID	*Meaning*
0 0 0 1	*Domain 1*
0 0 1 0	*Domain 2*
0 0 1 1	*Domain 3*
0 1 0 0	*Domain 4*
0 1 0 1	*Domain 5*
0 1 1 0	*Domain 6*

The 0000 and 1111 are forbidden values.

The Side ID format shall be coded with 3 bits.
The Side ID will use the following values :

Side ID	*Meaning*
0 0 1	*Side 1*
0 1 0	*Side 2*
0 1 1	*Side 3*
1 0 0	*Side 4*

The 000 and 111 are forbidden values.

The Location ID format shall be coded with 5 bits.
The 00000 and 11111 are forbidden values.

The Domain ID, Side ID and Location ID will be specified for each hosting equipment.

APPENDIX 5: IP Profile

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
Implement IP *The Internet layer of host software MUST implement IP. See Section 3.3.7 for the requirements on support of IGMP*	3.1	X					X			X			
Implement ICMP *The Internet layer of host software MUST implement ICMP.*	3.1	X						E		X			
Handle remote multihoming in application layer *At present, remote multihoming MUST be handled at the application Layer*	3.1	X						E				X	
Support local multihoming *A host MAY support local multihoming,*	3.1			X				X		X			
Meet gateway specifications if capable of forwarding datagrams *Any host that forwards datagrams generated by another host is acting as a gateway and MUST also meet the specifications laid out in the gateway requirements RFC [INTRO:2]*	3.1	X						E				X	*1
Configuration switch for embedded gateway *An Internet host that includes embedded gateway code MUST have a configuration switch to disable the gateway function…*	3.1	X						E				X	*1
Configuration default is non-gateway *...and this switch MUST default to the non-gateway mode.*	3.1	X						E				X	*1
Autoconfiguration based on number of interfaces *The host software MUST NOT automatically move into gateway mode if the host has more than one interface.*	3.1					X		E					
Able to log discarded datagrams *However, for diagnosis of problems a host SHOULD provide the capability of logging the error (see Section 1.2.3), including the contents of the silently-discarded datagram…*	3.1		X					X				X	*2
Record in counter *...and SHOULD record the event in a statistics counter.*	3.1		X					X				X	*2

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
Silently discard if IP version is not equal to 4 *A datagram whose version number is not 4 MUST be silently discarded.*	3.2.1.1	X					X			X			
Verify IP Checksum, silently discard bad datagram *A host MUST verify the IP header checksum on every received datagram and silently discard every datagram that has a bad checksum.*	3.2.1.2	X					X			X			
Addressing:													
Subnet addressing (RFC-950) *A host MUST support the subnet extensions to IP [IP:3].*	3.2.1.3	X					X			X			
Source address must be host's own IP address *When a host sends any datagram, the IP source address MUST be one of its own IP addresses (but not a broadcast or multicast address).*	3.2.1.3	X					X			X			
Silently discard datagram with bad destination address *A host MUST silently discard an incoming datagram that is not destined for the host.*	3.2.1.3	X					X			X			
Silently discard datagram with bad source address *A host MUST silently discard an incoming datagram containing an IP source address that is invalid by the rules of this section.*	3.2.1.3	X					X			X			
Support reassembly *The Internet model requires that every host support Reassembly.*	3.2.1.4	X						E		X			
Retain same ID field in identical datagrams *When sending an identical copy of an earlier datagram, a host MAY optionally retain the same Identification field in the copy.*	3.2.1.5			X				X		X			?

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
TOS:													
Allow transport layer to set TOS *The IP layer MUST provide a means for the transport layer to set the TOS field of every datagram that is sent; the default is all zero bits.*	3.2.1.6	X						E				X	
Pass received TOS up to transport layer *The IP layer SHOULD pass received TOS values up to the transport layer.*	3.2.1.6		X					X				X	
Use RFC-795 link-layer mappings for TOS *The particular link-layer mappings of TOS contained in RFC-795 SHOULD NOT be implemented.*	3.2.1.6				X			X				X	
TTL:													*3
Send packet with TTL of 0 *A host MUST NOT send a datagram with a Time-to-Live (TTL) value of zero .*	3.2.1.7					X			X			X	*4,5
Discard received packets with TTL < 2 *A host MUST NOT discard a datagram just because it was received with TTL less than 2.*	3.2.1.7					X		E				X	
Allow transport layer to set TTL *The IP layer MUST provide a means for the transport layer to set the TTL field of every datagram that is sent.*	3.2.1.7	X						E				X	
Fixed TTL is configurable *When a fixed TTL value is used, it MUST be configurable .*	3.2.1.7	X						E				X	
IP Options:													
Allow transport layer to send IP options *There MUST be a means for the transport layer to specify IP options to be included in transmitted IP datagrams.*	3.2.1.8	X						E				X	
Pass all IP options received to higher layer	3.2.1.8	X						E				X	

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
All IP options (except NOP or END-OF-LIST) received in datagrams MUST be passed to the transport layer (or to ICMP processing when the datagram is an ICMP message).													
IP layer silently ignore unknown options *The IP and transport layer MUST each interpret those IP options that they understand and silently ignore the others.*	3.2.1.8	X					X			X			
Security option *Some environments require the Security option in every datagram.*	3.2.1.8a			X				X				X	
Send Stream Identifier option *This option is obsolete; it SHOULD NOT be sent…*	3.2.1.8a				X			X				X	
Silently ignore Stream Identifier option *...and it MUST be silently ignored if received.*	3.2.1.8b	X					X			X			
Record Route option *Implementation of originating and processing the Record Route option is OPTIONAL.*	3.2.1.8d			X				X		?	?	X?	*6
Timestamp option *Implementation of originating and processing the Timestamp option is OPTIONAL.*	3.2.1.8e			X				X		?	?	X?	*6
Source Route Option:													
Originate and terminate Source Route Options *A host MUST support originating a source route and MUST be able to act as the final destination of a source route.*	3.2.1.8c	X						E				X	
Datagram with completed Source Route patted up to Transport Layer	3.2.1.8c	X						E				X	

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
If host receives a datagram containing a completed source route (i.e., the pointer points beyond the last field), the datagram has reached its final destination; the option as received (the recorded route) MUST be passed up to the transport layer (or to ICMP message processing).													
Build correct (non-redundant) return route *When a Return source route is built, it MUST be correctly formed even if the recorded route included the source host.*	3.2.1.8c	X						E				X	
Send multiple Source Route options in one header *An IP header containing more than one Source Route option MUST NOT be sent.*	3.2.1.8c					X			X			X	
Internet Control Message Protocol													
Silently discard ICMP messages with unknown type *If an ICMP message of unknown type is received, it MUST be silently discarded.*	3.2.2	X					X			X			
Include more that 8 octets of original datagram	3.2.2			X				X				X	
Include octets same as received *Every ICMP error message includes the Internet header and at least the first 8 data octets of the datagram that triggered the error; more than 8 octets MAY be sent.*	3.2.2	X						E				X	
Demultiplex ICMP Errors to transport protocol *In those cases where the Internet layer is required to pass an ICMP error message to the transport layer, the IP protocol number MUST be extracted from the original header and used to select the appropriate transport protocol entity to handle the error.*	3.2.2	X						E				X	
Send ICMPerror message with TOS=0	3.2.2		X					X				X	*7

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
An ICMP error message SHOULD be sent with normal (i.e., zero) TOS bits.													
Send ICMP error message for:													*8
- ICMP error message	3.2.2					X			X			X	*8
- IP broadcast or IP multicast	3.2.2					X			X			X	*8
- Link layer broadcast	3.2.2					X			X			X	*8
- Datagram with non-unique source address	3.2.2					X			X			X	*8
An ICMP error message MUST NOT be sent as the result of receiving: ** an ICMP error message, or* ** a datagram destined to an IP broadcast or IP multicast address, or* ** a datagram sent as a link-layer broadcast, or* ** a non-initial fragment, or* ** a datagram whose source address does not define a single host -- e.g., a zero address, a loopback address, a broadcast address, a multicast address, or a Class address.*													
Return ICMP error messages (when not prohibited) *Wherever practical, hosts MUST return ICMP error datagrams on detection of an error, except in those cases where returning an ICMP error message is specifically prohibited.*	3.2.2	X						E				X	*7
Destination unreachable:													
Generate /destination Unreachable (code 2/3) *A host SHOULD generate Destination Unreachable messages with Code 2 and 3.*	3.2.2.1		X					X				X	*7
Pass ICMP Destination Unreachable to higher layer *A Destination Unreachable message that is received MUST be reported to the transport layer.*	3.2.2.1	X						E				X	*7

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
Higher layer acts on Destination Unreachable *The transport layer SHOULD use the information appropriately.*	3.2.2.1		X					X				X	
Interpret Destination Unreachable as only a hint *A Destination Unreachable message that is received with code 0 (Net), 1 (Host), or 5 (Bad Source Route) may result from a routing transient and MUST therefore be interpreted as only a hint.*	3.2.2.1	X						E				X	*7
Redirect:													*10
Host send Redirect *A host SHOULD NOT send an ICMP Redirect message; Redirects are to be sent only by gateways.*	3.2.2.2				X			X		(X)		X	*9
Update route cache when receiving Redirect *A host receiving a Redirect message MUST update its routing information accordingly.*	3.2.2.2	X						E				X	*10
Handle both Host and Net Redirects *Every host MUST be prepared to accept both Host and Network Redirects and to process them as described in Section 3.3.1.2 below.*	3.2.2.2	X						E				X	*10
Discard illegal Redirect *A Redirect message SHOULD be silently discarded if the new gateway address it specifies is not on the same connected (sub-) net through which the Redirect arrived, or if the source of the Redirect is not the current first-hop gateway for the specified destination.*	3.2.2.2		X					X				X	*10
Source Quench:													
Send Source Quench if buffering exceeded	3.2.2.3			X				X				X	*10

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
A host MAY send a Source Quench message if it is approaching, or has reached, the point at which it is forced to discard incoming datagrams due to a shortage of reassembly buffers or other resources.													
Pass Source Quench to higher layer *If a Source Quench message is received, the IP layer MUST report it to the transport layer (or ICMP processing).*	3.2.2.3	X						E				X	*10
Higher layer act on Source Quench *The transport or application layer SHOULD implement a mechanism to respond to Source Quench for any protocol that can send a sequence of datagrams to the same destination and which can reasonably be expected to maintain enough state information to make this feasible.*	3.2.2.3		X					X				X	*10
Time Exceeded; pass to higher layer *An incoming Time Exceeded message MUST be passed to the transport layer.*	3.2.2.4	X						E				X	*10
Parameter Problem: *A host SHOULD generate Parameter Problem messages. An incoming Parameter Problem message MUST be passed to the transport layer, and it MAY be reported to the user.*													*10
Send parameter Problem messages	3.2.2.5		X					X				X	*10
Pass parameter Problem to higher layer	3.2.2.5	X						E				X	*10
Report Parameter Problem to user	3.2.2.5			X				X				X	*10
ICMP Echo Request or Reply Echo server and Echo client	3.2.2.6	X						E		X			

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
Every host MUST implement an ICMP Echo server function that receives Echo Requests and sends corresponding Echo Replies.													
Echo client *A host SHOULD also implement an application-layer interface for sending an Echo Request and receiving an Echo Reply, for diagnostic purposes.*	3.2.2.6		X					X		X			
Discard Echo Request to broadcast address *An ICMP Echo Request destined to an IP broadcast or IP multicast address MAY be silently discarded.*	3.2.2.6			X				X		X			
Discard Echo Request to multicast address *An ICMP Echo Request destined to an IP broadcast or IP multicast address MAY be silently discarded.*	3.2.2.6			X				X		X			
Use specific destination address as Echo Reply source *The IP source address in an ICMP Echo Reply MUST be the same as the specific-destination address (defined in Section 3.2.1.3) of the corresponding ICMP Echo Request message.*	3.2.2.6	X						E		X			
Send same data to Echo Reply *Data received in an ICMP Echo Request MUST be entirely included in the resulting Echo Reply.*	3.2.2.6	X						E		X			
Pass Echo Reply to higher layer *Echo Reply messages MUST be passed to the ICMP user interface, unless the corresponding Echo Request originated in the IP layer.*	3.2.2.6	X						E		X			
Reflect record Rout, Time Stamp options	3.2.2.6		X					X				X	*10

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
If a Record Route and/or Time Stamp option is received in an ICMP Echo Request, this option (these options) SHOULD be updated to include the current host and included in the IP header of the Echo Reply message, without "truncation" Thus, the recorded route will be for the entire round trip.													
Reverse and reflect Source Route options *If a Source Route option is received in an ICMP Echo Request, the return route MUST be reversed and used as a Source Route option for the Echo Reply message.*	3.2.2.6	X						E				X	*10
ICMP Information Request or Reply *A host SHOULD NOT implement these messages.*	3.2.2.7				X			X		(X)		X	*10
ICMP Timestamp and Timestamp Reply: *A host MAY implement Timestamp and Timestamp Reply.*	3.2.2.8			X				X				X	*10
Minimize delay variability *If this function is implemented, it SHOULD be designed for minimum variability in delay.*	3.2.2.8		X					X				X	*10
Silently discard multicast Timestamp *An ICMP Timestamp Request message to an IP broadcast or IP multicast address MAY be silently discarded.*	3.2.2.8			X				X				X	*10
Silently discard multicast Timestamp *An ICMP Timestamp Request message to an IP broadcast or IP multicast address MAY be silently discarded.*	3.2.2.8			X				X				X	*10
Use specific destination as Timestamp Reply source *The IP source address in an ICMP Timestamp Reply MUST be the same as the specific-destination address of the corresponding Timestamp Request message.*	3.2.2.8	X						E				X	*10
Reflect Record Route, Time Stamp options	3.2.2.6		X					X				X	*10

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
If a Source-route option is received in an ICMP Echo Request, the return route MUST be reversed and used as a Source Route option for the Timestamp Reply message.													
Reverse and reflect Source Route option *If a Record Route and/or Timestamp option is received in a Timestamp Request, this (these) option(s) SHOULD be updated to include the current host and included in the IP header of the Timestamp Reply message.*	3.2.2.8	X						E				X	*10
Pass Timestamp Reply to higher layer *Incoming Timestamp Reply messages MUST be passed up to the ICMP user interface.*	3.2.2.8	X						E				X	*10
Obey rules for "standard value" *The preferred form for a timestamp value (the "standard value") is in units of milliseconds since midnight Universal Time.*	3.2.2.8	X						E				X	*10
ICMP Address Mask Request and Reply													
Address Mask source configurable *The choice of method to be used in a particular host MUST be configurable…*	3.2.2.9	X						E				X	*10
Support static configuration of address mask *(1) static configuration information…*	3.2.2.9	X						E				X	*10
Get address mask dynamically during booting *(2) obtaining the address mask(s) dynamically as a side-effect of the system initialisation process…*	3.2.2.9			X				X				X	*10
Get address via ICMP Address Mast Request/Reply *(3) sending ICMP Address Mask Request(s) and receiving ICMP Address Mask Reply(s).*	3.2.2.9			X				X				X	*10
Retransmit Address Mask Request if no Reply	3.2.2.9	X						E				X	*10

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
When method 3, it MUST retransmit this message a small number of times if it does not receive an immediate Address Mask Reply.													
Assume default mask if no Reply *When method 3, until it has received an Address Mask Reply, the host SHOULD assume a mask appropriate for the address class of the IP address.*	3.2.2.9		X					X				X	*10
Update Address Mask from first Reply only *When method 3, the first Address Mask Reply message received MUST be used to set the address mask corresponding to the particular local IP address.*	3.2.2.9	X						E				X	*10
Reasonableness check on Address Mask *A host SHOULD make some reasonableness check on any address mask it installs.*	3.2.2.9		X					X				X	*10
Send unauthroized Address Mask Reply message *A system MUST NOT send an Address Mask Reply unless it is an authoritative agent for address masks.*	3.2.2.9					X			X			X	*10
Explicitly configured to be agent *An authoritative agent may be a host or a gateway, but it MUST be explicitly configured as a address mask agent.*	3.2.2.9	X						E				X	*10
Static configuration-> Address Mask Authorization flag *With a statically configured address mask, there SHOULD be an additional configuration flag that determines whether the host is to act as an authoritative agent for this mask.*	3.2.2.9		X					X				X	*10
Broadcast Address Mask Reply when initiated *If it is configured as an agent, the host MUST broadcast an Address Mask Reply for the mask on the appropriate interface when it initializes.*	3.2.2.9	X						E				X	*10

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
ROUTING OUTBOUND DATAGRAMS													
Use address mask in local/remote decision *To decide if the destination is on a connected network, the following algorithm MUST be used.*	3.3.1.1.	X						E				X	
Operate with no gateways on connected network *The host IP layer MUST operate correctly in a minimal network environment, and in particular, when there are no gateways.*	3.3.1.1	X						E		X		(X)	*11
Maintain "route cache" of next-hop gateway *To efficiently route a series of datagrams to the same destination, the source host MUST keep a "route cache" of mappings to next-hop gateways.*	3.3.1.2	X						E				X	*12
Treat Host and Net Redirect the same *Since the subnet mask appropriate to the destination address is generally not known, a Network Redirect message SHOULD be treated identically to a Host Redirect message.*	3.3.1.2		X					X				X	*10
If no cache entry, use default gateway *The IP layer MUST pick a gateway from its list of "default" gateways.*	3.3.1.2	X						E				X	*12
Support multiple default gateways *The IP layer MUST support multiple default gateways.*	3.3.1.2	X						E				X	*12
Provide table of static routes *As an extra feature, a host IP layer MAY implement a table of "static routes".*	3.3.1.2			X				X				X	
Flag: route over-rideable by Redirect *Each such static route MAY include a flag specifying whether it may be overridden by ICMP Redirects.*	3.3.1.2			X				X				X	*10
Key route case on host, not network address	3.3.1.3			X				X				X	*12

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
Each route cache entry needs to include the following fields: *(1) Local IP address (for a multihomed host)* *(2) Destination IP address* *(3) Type(s)-of-Service* *(4) Next-hop gateway IP address*													
Include TOS in route cache *The TOS SHOULD be included.*	3.3.1.3		X					X				X	*12
Able to detect failure of next-hop gateway *The IP layer MUST be able to detect the failure of a "next-hop" gateway that is listed in its route cache and to choose an alternate gateway.*	3.3.1.4	X						E				X	*12
Assume route is good forever *A particular gateway SHOULD NOT be used indefinitely in the absence of positive indications that it is functioning.*	3.3.1.4				X			X				X	*12
Ping gateways continuously *In particular, hosts MUST NOT actively check the status of a first-hop gateway by simply pinging the gateway continuously.*	3.3.1.4					X			X			X	*12
Ping only when traffic being sent *Pinging MUST be used only when traffic is being sent to the gateway…*	3.3.1.4	X						E				X	*12
Ping only when no positive indication *...and when there is no other positive indication to suggest that the gateway is functioning.*	3.3.1.4	X						E				X	*12
Higher and lower layers give advice *To avoid pinging, the layers above and/or below the Internet layer SHOULD be able to give "advice" on the status of route cache entries when either positive (gateway OK) or negative (gateway dead) information is available.*	3.3.1.4		X					X				X	*12

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
Switch from failed default gateway to another *If the failed gateway is not the current default, the IP layer can immediately switch to a default gateway. If it is the current default that failed, the IP layer MUST select a different default gateway.*	3.3.1.5	X						E				X	*12
Manual method of entering configuration information *A manual method of entering this configuration data MUST be provided.*	3.3.1.6	X						E				X	*13
REASSEMBLY AND FRAGMENTATION:													
Able to reassemble incoming datagrams *The IP layer MUST implement reassembly of IP datagrams.*	3.3.2	X						E		X			
At least 576 bytes datagrams	3.3.2	X						E				X	
EMTU_R configurable or indefinite (Note 7) *We designate the largest datagram size that can be reassembled by EMTU_R ("Effective MTU to receive"); this is sometimes called the "reassembly buffer size". EMTU_R MUST be greater than or equal to 576, SHOULD be either configurable or indefinite, and SHOULD be greater than or equal to the MTU of the connected network(s).*	3.3.2		X					X				X	
Transport layer able to learn MMS_R *There MUST be a mechanism by which the transport layer can learn MMS_R.*	3.3.2	X						E				X	
Send ICMP Time Exceeded on reassembly timeout *If this timeout expires, the partially-reassembled datagram MUST be discarded and an ICMP Time Exceeded message sent to the source host (if fragment zero has been received.*	3.3.2	X						E				X	*10
Fixed reassembly timeout value	3.3.2		X					X		X		(X)	*14

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
The TCP specification [TCP:1] arbitrarily assumes a value of 2 minutes for MSL. This sets an upper limit on a reasonable reassembly timeout value.													
Pass MMS_S to higher layers (Note 8) *A host MUST implement a mechanism to allow the transport layer to learn MMS_S.*	3.3.3	X						E				X	
Local fragmentation of outgoing packets *Optionally, the IP layer MAY implement a mechanism to fragment outgoing datagrams intentionally.*	3.3.3			X				X		X			
Else don't send bigger than MMS_S *A host that does not implement local fragmentation MUST ensure that the transport layer (for TCP) or the application layer (for UDP) obtains MMS_S from the IP layer and does not send a datagram exceeding MMS_S in size.*	3.3.3	X						E				X	
Send max 576 to off-net destination *In the absence of actual knowledge of the minimum MTU along the path, the IP layer SHOULD use EMTU_S <= 576 whenever the destination address is not on a connected network, and otherwise use the connected network's.*	3.3.3		X					X				X	
All-Subnets-MTU configuration flag *A host IP layer implementation MAY have a configuration flag "All-Subnets-MTU", indicating that the MTU of the connected network is to be used for destinations on different subnets within the same network, but not for other networks.*	3.3.3			X				X				X	
MULTIHOMING:													
Reply with same address as specified destination address	3.3.4.2		X					X		X			

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
(1) If the datagram is sent in response to a received datagram, the source address for the response SHOULD be the specific-destination address of the request.													
Allow application to choose local IP address *(2) An application MUST be able to explicitly specify the source address for initiating a connection or a request.*	3.3.4.2	X					X			X			
Silently discard datagram in "wrong" interface *(A) A host MAY silently discard an incoming datagram whose destination address does not correspond to the physical interface through which it is received.*	3.3.4.2			X				X		X			
Only send datagram through "right" interface *(B) A host MAY restrict itself to sending (non-source-routed) IP datagrams only through the physical interface that corresponds to the IP source address of the datagrams.*	3.3.4.2			X				X		X			
SOURCE-ROUTE FORWARDING:													
Forward datagram with Source Route Option *Subject to restrictions given below, a host MAY be able to act as an intermediate hop in a source route, forwarding a source-routed datagram to the next specified hop.*	3.3.5			X				X				X	*15
Obey corresponding gateway rules *However, in performing this gateway-like function, the host MUST obey all the relevant rules for a gateway forwarding source-routed datagrams.*	3.3.5	X						E				X	
Update TTL by gateway rules *The TTL field MUST be decremented and the datagram.*	3.3.5	X						E				X	
Able to generate ICMP error codes 4 and 5	3.3.5	X						E				X	*10

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
A host MUST be able to generate Destination Unreachable messages with the following codes: *4 (Fragmentation Required but DF Set) when a source-routed datagram cannot be fragmented to fit into the target network* *5 (Source Route Failed) when a source-routed datagram cannot be forwarded, e.g., because of a routing problem or because the next hop of a strict source route is not on a connected network.*													
IP source address not local host *A source-routed datagram being forwarded MAY (and normally will) have a source address that is not one of the IP addresses of the forwarding host.*	3.3.5			X				X				X	
Update Timestamp, Record Route options *A host that is forwarding a source-routed datagram containing a Record Route option MUST update that option, if it has room.* *A host that is forwarding a source-routed datagram containing a Timestamp Option MUST add the current timestamp to that option, according to the rules for this option.*	3.3.5	X						E				X	*10
Configurable switch for non-local Source Routing *A host that supports non-local source-routing MUST have a configurable switch to disable forwarding…*	3.3.5	X						E				X	
Default to OFF *...and this switch MUST default to disabled.*	3.3.5	X						E				X	
Satisfy gateway access rules for non-local Source Routing *The host MUST satisfy all gateway requirements for configurable policy filters restricting non-local forwarding.*	3.3.5	X						E				X	
If not forward, send Destination Unreachable (code 5)	3.3.5		X					X				X	*10

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
If a host receives a datagram with an incomplete source route but does not forward it for some reason, the host SHOULD return an ICMP Destination Unreachable (code 5, Source Route Failed) message, unless the datagram was itself an ICMP error message.													
BROADCAST:													
Broadcast address as IP source address *Section 3.2.1.3 defined the four standard IP broadcast address forms:* ** Limited Broadcast: {-1, -1}* ** Directed Broadcast: {<Network-number>,-1}* ** Subnet Directed Broadcast: {<Network-number>,<Subnet-number>,-1}* ** All-Subnets Directed Broadcast: {<Network-number>,-1,-1}* *A host MUST recognize any of these forms in the destination address of an incoming datagram.*	3.2.1.3					X			X			X	
Receive 0 or -1 broadcast formats OK *There is a class of hosts that use non-standard broadcast address forms, substituting 0 for -1. All hosts SHOULD recognize and accept any of these non-standard broadcast addresses as the destination address of an incoming datagram.*	3.3.6		X					X				X	
Configurable option to send 0 or -1 broadcasts *A host MAY optionally have a configuration option to choose the 0 or the -1 form of broadcast address, for each physical interface…*	3.3.6			X				X				X	
Default to -1 broadcast	3.3.6		X					X				X	

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
...but this option SHOULD default to the standard (-1) form.													
Recognize all broadcast address formats *A host MUST recognize any of these forms in the destination address of an incoming datagram.*	3.3.6	X						E				X	
Use IP broadcast/multicast address in link-layer broadcast *When a host sends a datagram to a link-layer broadcast address, the IP destination address MUST be a legal IP broadcast or IP multicast address.*	3.3.6	X						E				X	*16
Silently discard link-layer-only broadcast datagrams *A host SHOULD silently discard a datagram that is received via a link-layer broadcast (see Section 2.4) but does not specify an IP multicast or broadcast destination address.*	3.3.6		X					X				X	*16
Use Limited Broadcast address for connected network *Hosts SHOULD use the Limited Broadcast address to broadcast to a connected network.*	3.3.6		X					X				X	
MULTICAST:													
Support local IP multicasting (RFC-1112) *A host SHOULD support local IP multicasting on all connected networks for which a mapping from Class D IP addresses to link-layer addresses has been specified.*	3.3.7		X					X		X			
Support IGMP (RFC-1112) *Support for local IP multicasting includes sending multicast datagrams, joining multicast groups and receiving multicast datagrams, and leaving multicast groups. This implies support for all of except the IGMP protocol itself, which is OPTIONAL.*	3.3.7			X				X				X	
Join all-hosts group at startup	3.3.7		X					X				X	

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			NOTES
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	
If IGMP is not implemented, a host SHOULD still join the "all-hosts" group (224.0.0.1) when the IP layer is initialized and remain a member for as long as the IP layer is active.													
Higher layers learn interface multicast capability *A host SHOULD provide a way for higher-layer protocols or applications to determine which of the host's connected network(s) support IP multicast addressing.*	3.3.7		X					X				X	
INTERFACES:													
Allow transport layer to use all IP mechanisms *The interface between the IP layer and the transport layer MUST provide full access to all the mechanisms of the IP layer, including options, Type-of-Service, and Time-to-Live.*	3.4	X						E				X	
Pass interface identification up to transport layer *The transport layer MUST either have mechanisms to set these interface parameters, or provide a path to pass them through from an application, or both.*	3.4	X						E				X	
Pass all IP options up to transport layer *The parameter opt contains all the IP options received in the datagram; these MUST also be passed to the transport layer.*	3.4	X						E				X	
Transport layer can send certain ICMP messages *The transport layer MUST be able to send certain ICMP Messages.*	3.4	X						E				X	
Pass specified ICMP messages up to transport layer *The IP layer MUST pass certain ICMP messages up to the appropriate transport-layer routine.*	3.4	X						E				X	
Include IP header + 8-octets or more from original	3.4	X						E				X	

Transport Layer IP - Requirements Summary (RFC-1122 Paragraph 3.5)

	RFC1122	INTERNET					PROFILED			AFDX			
		MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
For an ICMP error message, the data that is passed up MUST include the original Internet header plus all the octets of the original message that are included in the ICMP message.													
Able to leap tall buildings in a single bound	3.4		X					X		X?	?	?	*17

7 EMTU_R - Effective Maximum Transfer Unit to Receive. RFC-1122 paragraph 3.3.2
8 MMS_R - Maximum Message Size that can be received and reassembled. RFC-1122 paragraph 3.3.2

Comment: The TCP profile uses "Not Applicable" which is less confusing than "NO" [gvb]

*1 AFDX doesn't allow embedded gateways therefore these don't apply?
*2 AFDX doesn't allow discarded packet logging or counting???
*3 TTL ignored by the receiver
*4 TTL must be set to 1
*5 AFDX says "YES, it must not be set to zero" which means "NO"
*6 Missing line in the AFDX specification
*7 AFDX doesn't allow ICMP errors therefore this doesn't apply?
*8 AFDX negated the standard wording, causing the selections to be "YES" instead of "NO" (this is following the RFC-1122 wording)
*9 AFDX says "YES", but should say "NO" (the question is positive, the answer is negative)
*10 Disallowed: only simple ICMP "Echo Request" and "Echo Reply" ICMP messages with no options are allowed by AFDX
*11 AFDX says "NO", but should say "YES" (it must operate properly with no gateways on the network)
*12 AFDX allows only static routing tables and no default gateways.
*13 AFDX says "NO", but should say "YES" (manual configuration, i.e. downloadable tables, are the only way to configure AFDX routing).
*14 AFDX says "NO", but should say "YES" (there has to be some reassembly timeout and it should be a fixed value).
*15 Source routing is not allowed.
*16 AFDX allows sending a unicast IP address using a multicast link layer datagram.
*17 Missing line in the AFDX specification, always a good feature for an aircraft.

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
PUSH FLAG													
Aggregate or queue un-pushed data *When an application issues a series of SEND calls without setting the PUSH flag, the TCP MAY aggregate the data internally without sending it. Similarly, when a series of Segments is received without the PSH bit, a TCP MAY queue the data internally without passing it to the receiving application.*	4.2.2.2			X				X		X			
Sender collapse successive PUSH flags *The transmitter SHOULD collapse Successive PSH bits when it packetizes data, to send the largest possible segment.*	4.2.2.2		X					X				X	
SEND call can specify PUSH *A TCP MAY implement PUSH flags on SEND calls…*	4.2.2.2			X				X		X			
If cannot, sender buffer indefinitely *...then the sending TCP: (1) must not buffer data indefinitely…*	4.2.2.2					X		X		X			*1
If cannot, PUSH last segment *...and (2) MUST set the PSH bit in the last buffered segment (i.e., when there is no more queued data to be sent).*	4.2.2.2	X					X			X			
Notify receiving application of PUSH *Passing a received PSH flag to the application layer is now OPTIONAL.*	4.2.2.2			X				X		X			
Send maximum size segment when possible *However, a TCP SHOULD send a maximum-sized segment whenever possible, to improve performance*	4.2.2.2		X					X		X			
WINDOW													

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
Treat as unsigned number *The window size MUST be treated as an unsigned number.*	4.2.2.3	X					X			X			
Handle as 32-bit number *In anticipation of the adoption of such an extension, TCP implementors should treat windows as 32 bits.*	4.2.2.3		X				X			X			
Shrink window from right *A TCP receiver SHOULD NOT shrink the window, i.e., move the right window edge to the left.*	4.2.2.16				X				X	X			*1
Robust against shrinking window *However, a sending TCP MUST be robust against window shrinking, which may cause "useable window" (see Section 4.2.3.4) to become negative*	4.2.2.16	X					X			X			
Receiver window closed indefinitely *A TCP MAY keep its offered receive window closed indefinitely.*	4.2.2.17			X				X				X	
Sender probe zero window *Probing of zero (offered) windows MUST be supported.*	4.2.2.17	X						X		X			
First probe after RTO (Note 1) *The transmitting host SHOULD send the first zero-window probe when a zero window has existed for the retransmission timeout period.*	4.2.2.17		X					X		X			
Exponential backoff *The transmitting host SHOULD increase exponentially the interval between successive probes.*	4.2.2.17		X					X		X			
Allow window stay zero indefinitely *The sending TCP MUST allow the connection to stay open.*	4.2.2.17	X						X		X			
Sender timeout OK connection with zero window *The sending TCP MUST allow the connection to stay open.*	4.2.2.17					X		X				X	

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
URGENT DATA													
Pointer points to last octet *The urgent pointer points to the sequence number of the LAST octet (not LAST+1)*	4.2.2.4	X					X						N/A
Arbitrary length urgent data sequence *A TCP MUST support a sequence of urgent data of any length.*	4.2.2.4	X					X						N/A
Inform application asynchronously of urgent data *A TCP MUST inform the application layer asynchronously whenever it receives an Urgent pointer and there was previously no pending urgent data.*	4.2.2.4	X						X					N/A
Application can learn if/how much urgent data is queued *There MUST be a way for the application to learn how much urgent data remains to be read from the connection, or at least to determine whether or not more urgent data remains to be read.*		X						X					N/A
TCP OPTIONS													
Receive TCP Option in any segment *A TCP MUST be able to receive a TCP option in any segment.*	4.2.2.5	X					X			X			
Ignore unsupported options *A TCP MUST ignore without error any TCP option it does not implement, assuming that the option has a length field.*	4.2.2.5	X					X			X			
Cope with illegal option length *TCP MUST be prepared to handle an illegal option length.*	4.2.2.5	X					X			X			
Implement sending and receiving MSS option (Note 2)	4.2.2.6	X						X		X			

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
TCP MUST implement both sending and receiving the Maximum Segment Size option.													
Send MSS option unless 536 *TCP SHOULD send an MSS (Maximum Segment Size) option in Every SYN segment when its receive MSS differs from the default 536,…*	4.2.2.6		X					X		X			
Send MSS option always *...and MAY send it always.*	4.2.2.6			X				X				X	
Send MSS default is 536 *If an MSS option is not received at connection setup, TCP MUST assume a default send MSS of 536.*	4.2.2.6	X					X			X			
Calculate effective send segment size *The maximum size of a segment that TCP really sends, the "effective send MSS," MUST be the smaller of the send MSS (which reflects the available reassembly buffer size at the remote host) and the largest size permitted by the IP layer.*	4.2.2.6	X					X			X			
TCP CHECKSUMS													
Sender compute checksum *The sender MUST generate.*	4.2.2.7	X					X			X			
Receiver check checksum *The receiver MUST check it.*	4.2.2.7	X					X			X			
Use clock-driven ISN selection (Note 3) *A TCP MUST use the specified clock-driven selection of initial sequence numbers.*	4.2.2.9	X						E		X			
OPENING CONNECTIONS													

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
Support simultaneous open attempts *A TCP MUST support simultaneous open attempts.*	4.2.2.10	X					X			X			
SYN-RCVD remember last state *TCP implementation MUST keep track of whether a connection has reached SYN_RCVD state as the result of a passive OPEN or an active OPEN.*	4.2.2.11	X					X			X			
Passive open call interfere with others *Every passive OPEN call it MUST NOT affect any previously created connection record.*	4.2.2.18					X			X	X			*1
Function simulate LISTENs for same port *A TCP that supports multiple concurrent users MUST provide an OPEN call that will functionally allow an application to LISTEN on a port while a connection block with the same local port is in SYN-SENT or SYN-RECEIVED state.*	4.2.2.18	X					X			X			
Ask IP for source address for SYN if necessary *The TCP MUST ask the IP layer to select a local IP.*	4.2.3.7	X					X			X			
Otherwise, use local address of connection *At all other times, a previous segment has either been sent Or received on this connection, and TCP MUST use the same local address is used that was used in those previous segments.*	4.2.3.7	X					X			X			
OPEN to broadcast/multicast IP address *A TCP implementation MUST reject as an error a local OPEN call for an invalid remote IP address (e.g., a broadcast or multicast address).*	4.2.3.10					X			X	X			*1
Silently discard segment to broadcast/multicast address	4.2.3.10	X					X			X			

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
A TCP implementation MUST silently discard an incoming SYN segment that is addressed to a broadcast or multicast address.													
CLOSING CONNECTIONS													
RST can contain data *A TCP SHOULD allow a received RST segment to include data.*	4.2.2.12		X					X		X			
Inform applications of aborted connection *If a TCP connection is closed by the remote site, the local application MUST be informed whether it closed normally or was aborted.*	4.2.2.13	X					X			X			
Half-duplex close connections *A host MAY implement a "half-duplex" TCP close sequence, so that an application that has called CLOSE cannot continue to read data from the connection.*	4.2.2.13			X				X				X	
Send RST to indicate data lost *If such a host issues a CLOSE call while received data is still pending in TCP, or if new data is received after CLOSE is called, its TCP SHOULD send a RST to show that data was lost.*	4.2.2.13		X					X		X			
In TIME-WAIT state for 2xMSL seconds *When a connection is closed actively, it MUST linger in TIME-WAIT state for a time 2xMSL.*	4.2.2.13	X					X			X			
Accept SYN from TIME-WAIT state *It MAY accept a new SYN from the remote TCP to reopen the connection directly from TIME-WAIT state.*	4.2.2.13			X				X				X	

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
RETRANSMISSIONS													
Jacobson Slow Start algorithm *Recent work by Jacobson [TCP:7] on Internet congestion and TCP retransmission stability has produced a transmission algorithm combining "slow start" with "congestion avoidance". A TCP MUST implement this algorithm.*	4.2.2.15	X						E				X	
Jacobson Congestion-Avoidance algorithm *Recent work by Jacobson [TCP:7] on Internet congestion and TCP retransmission stability has produced a transmission algorithm combining "slow start" with "congestion avoidance". A TCP MUST implement this algorithm.*	4.2.2.15	X						E				X	
Retransmit with same IP ident *If a retransmitted packet is identical to the original packet (which implies not only that the data boundaries have not changed, but also that the window and acknowledgment fields of the header have not changed), then the same IP Identification field MAY be used.*	4.2.2.15			X				X				X	
Karn's algorithm *A host TCP MUST implement Karn's algorithm and Jacobson's algorithm for computing the retransmission timeout ("RTO").*	4.2.3.1	X						E				X	
Jacobson Retransmission Timeout estimation algorithm *A host TCP MUST implement Karn's algorithm and Jacobson's algorithm for computing the retransmission timeout ("RTO").*	4.2.3.1	X						E				X	
Exponential backoff	4.2.3.1	X						E				X	

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
This implementation also MUST include "exponential backoff" for successive RTO values for the same segment.													
SYN Retransmission Timeout calculation same as data *Retransmission of SYN segments SHOULD use the same algorithm as data segments.*	4.2.3.1		X					X		X			
Recommended initial values and bounds *The following values SHOULD be used to initialize the estimation parameters for a new connection:* *(a) RTT = 0 seconds.* *(b) RTO = 3 seconds.* *The recommended upper and lower bounds on the RTO are known to be inadequate on large internets. The lower bound SHOULD be measured in fractions of a second and the upper bound should be 2*MSL, i.e., 240 seconds.*	4.2.3.1		X					X		X			
GENERATING ACKNOWLEDGMENTS													
Queue out-of-order segments *A TCP SHOULD be capable of queueing out-of-order TCP segments.*	4.2.2.20		X					X				X	
Process all queued data before sending ACK *In general, the processing of received segments MUST be implemented to aggregate ACK segments whenever possible.*	4.2.2.20	X					X			X			
Send ACK for out-of-order segment *A TCP MAY send an ACK segment acknowledging RCV.NXT when a valid segment arrives that is in the window but not at the left window edge.*	4.2.2.21			X				X				X	
Delayed ACKs	4.2.3.2		X					X		X			

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
A TCP SHOULD implement a delayed ACK, but an ACK should not be excessively delayed.													
Delay less than 0.5 seconds *The delay MUST be less than 0.5 seconds.*	4.2.3.2		X				X			X			
Every second full-size segment ACK'd *In a stream of full-sized segments there SHOULD be an ACK for at least every second segment.*	4.2.3.2		X				X			X			
Receiver SWS-Avoidance algorithm (Note 4) *A TCP MUST include a SWS avoidance algorithm in the receiver.*	4.2.3.3		X					E		X			
SENDING DATA													
Configurable Time to Live *The TTL value used to send TCP segments MUST be configurable.*	4.2.2.19		X				X					X	*2
Sender SWS-Avoidance algorithm *A TCP MUST include a SWS avoidance algorithm in the sender.*	4.2.3.6		X					E				X	
Nagle algorithm *A TCP SHOULD implement the Nagle Algorithm [TCP:9] to coalesce short segments.*	4.2.3.4			X				X				X	
Application can disable Nagle algorithm *There MUST be a way for an application to disable the Nagle algorithm on an individual connection.*	4.2.3.4		X					X					N/A
CONNECTION FAILURE													
Negative advice to IP on R1 retransmissions (Note 5)	4.2.3.5		X				X			X			

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
When the number of transmissions of the same segment reaches or exceeds threshold R1, pass negative advice to the IP layer, to trigger dead-gateway diagnosis.													
Close connection on R2 retransmissions (Note 6) *When the number of transmissions of the same segment reaches a threshold R2 greater than R1, close the connection.*	4.2.3.5	X					X			X			
Application can set R2 *An application MUST be able to set the value for R2 for a particular connection.*	4.2.3.5	X						E		X			
Inform application of R1<=retransmissions<R2 *TCP SHOULD inform the application of the delivery Problem (unless such information has been disabled by the application; see Section 4.2.4.1), when R1 is reached and before R2.*	4.2.3.5		X					X				X	
Recommended values for R1 and R2 *The value of R1 SHOULD correspond to at least 3 retransmissions, at the current RTO. The value of R2 SHOULD correspond to at least 100 seconds.*	4.2.3.5		X					X		X			
Same mechanism for SYNs *SYN Retransmissions MUST be handled in the general way just described for data retransmissions, including notification of the application layer.*	4.2.3.5	X					X			X			
R2 at least 3 minutes for SYN *R2 for a SYN segment MUST be set large enough to provide retransmission of the segment for at least 3 minutes.*	4.2.3.5		X					X				X	

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
SEND KEEP-ALIVE PACKETS	4.2.3.6			X				X				X	*3
Application can request *Implementors MAY include "Keep live" in their TCP implementation.*	4.2.3.6	X						E					N/A
Default is "off" *If keep-alives are included, they MUST default to off.*	4.2.3.6	X					X						N/A
Only send if idle for interval *Keep-alive packets MUST only be sent when no data or Acknowledgement packets have been received for the connection within an interval.*	4.2.3.6	X					X						N/A
Interval configurable *This interval MUST be configurable.*	4.2.3.6	X					X						N/A
Default is at least 2 hours *If a keep-alive mechanism is implemented it MUST NOT interpret failure to respond to any specific probe as a dead connection.*	4.2.3.6	X						E					N/A
IP OPTIONS													
Ignore options TCP doesn't understand *When received options are passed up to TCP from the IP layer, TCP MUST ignore options that it does not understand.*	4.2.3.8	X					X			X			
Time Stamp support *A TCP MAY support the Time Stamp.*	4.2.3.8			X				X				X	
Record Route support *A TCP MAY support the Record Route options.*	4.2.3.8			X				X				X	
Source Route:													
Application can specify Source Route	4.2.3.8	X						E				X	

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
An application MUST be able to specify a source route when it actively opens a TCP connection.													
Override Source Route in datagram *This MUST take precedence over a source route received in a datagram.*	4.2.3.8	X						E				X	
Build return route from Source Route *When a TCP connection is OPENed passively and a packet arrives with a completed IP Source Route option (containing a return route), TCP MUST save the return route and use it for all segments sent on this connection.*	4.2.3.8	X						E				X	
Later Source Route override *If a different source route arrives in a later segment, the later definition SHOULD override the earlier one.*	4.2.3.8		X					X				X	
RECEIVING ICMP MESSAGES FROM IP	4.2.3.9	X						E				X	
Destination Unreachable (0,1,5) sent to application *TCP MUST react to a Source Quench by slowing transmission on the connection. The RECOMMENDED procedure is for a Source Quench to trigger a "slow start," as if a retransmission timeout had occurred. Destination Unreachable -- codes 0, 1, 5.*	4.2.3.9		X					X					N/A
Destination Unreachable (0,1,5) aborts connection *Since these Unreachable messages indicate soft error conditions, TCP MUST NOT abort the connection.*	4.2.3.9					X		E					N/A
Destination Unreachable (2,4) aborts connection *Destination Unreachable -- codes 2-4 These are hard error conditions, so TCP SHOULD abort the connection.*	4.2.3.9		X					X					N/A

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET				PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT / MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
Source Quench => slow star *TCP MUST react to a Source Quench by slowing transmission on the connection. The RECOMMENDED procedure is for a Source Quench to trigger a "slow start," as if a retransmission timeout had occurred.*	4.2.3.9		X				X					N/A
Time Exceeded => tell application, don't abort *This should be handled the same way as Destination Unreachable codes 0, 1, 5.*	4.2.3.9		X				X					N/A
Parameter Problem => tell application, don't abort *This should be handled the same way as Destination Unreachable codes 0, 1, 5.*	4.2.3.9		X				X					N/A
ADDRESS VALIDATION												
Reject OPEN call to invalid IP address *A TCP implementation MUST reject as an error a local OPEN call for an invalid remote IP address.*	4.2.3.10	X				X			X			
Reject SYN from invalid IP address *An incoming SYN with an invalid source address must be ignored either by TCP or by the IP layer.*	4.2.3.10	X				X			X			
Silently discard SYN to broadcast/multicast address *A TCP implementation MUST silently discard an incoming SYN segment that is addressed to a broadcast or multicast address.*	4.2.3.10	X				X			X			
TCP/APPLICATION INTERFACE SERVICES												
Error Report mechanisms	4.2.4.1	X				X					X	*2

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
There MUST be a mechanism for reporting soft TCP error conditions to the application.													
Application can disable Error Report Routine *An application program that does not want to receive such ERROR_REPORT calls SHOULD be able to effectively disable these calls.*	4.2.4.1		X					X					N/A
Application can specify TOS for sending *The application layer MUST be able to specify the Type-of-Service (TOS) for segments that are sent on a connection.*	4.2.4.2	X						E				X	
Passed unchanged to IP *TCP SHOULD pass the current TOS value without change to the IP layer, when it sends segments on the connection.*	4.2.4.2		X					X					N/A
Application can change TOS during connection *The application SHOULD be able to change the TOS during the connection lifetime.*	4.2.4.2		X					X				X	
Pass received TOS up to application *TCP MAY pass the most recently received TOS up to the Application.*	4.2.4.2			X				X					N/A
FLUSH call *Some TCP implementations have included a FLUSH call.*	4.2.4.2			X				X				X	
Optional local IP address parameter in OPEN *The OPEN call MUST have an optional parameter.*	4.2.4.4			X				X				X	

Transport Layer TCP - Requirements Summary (RFC-1122 Paragraph 4.2)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES

NOTES
1 RTO: Retransmit timeout
2 MSS: Maximum Segment Size
3 ISN: Initial sequence number.
4 SWS: Silly Window Syndrome
5 R1: First retransmission threshold
6 R2: Second retransmission threshold

*1 AFDX says "YES", probably should be "NO" (conflict between AFDX and RFC-1122, ARINC-664)
*2 AFDX says "NO", probably should be "YES" (conflict between AFDX and RFC-1122, ARINC-664)
*3 [advice from gvb] "Keep alive" packets are very important for detecting if a connection that is used to only send data is closed improperly.

Transport Layer UDP - Requirements Summary (RFC-1122 Paragraph 4.1)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
UDP send Port Unreachable *If a datagram arrives addressed to a UDP port for which there is no pending LISTEN call, UDP SHOULD send an ICMP Port Unreachable message.*	4.1.3.1		X					X				X	
IP Options in UDP													
Pass received IP options to application layer *UDP MUST pass any IP option that it receives from the IP layer transparently to the application layer.*	4.1.3.2	X						E				X	*1
Application layer can specify IP options in Send *An application MUST be able to specify IP options to be sent in its UDP datagrams.*	4.1.3.2	X						E				X	*1
UDP passes IP options down to IP layer *UDP MUST pass these options to the IP layer.*	4.1.3.2	X						E				X	*1
Pass ICMP messages up to application layer *UDP MUST pass to the application layer all ICMP error messages that it receives from the IP layer.*	4.1.3.3	X						E		X			*2
UDP Checksums:													
Able to generate/check checksum *The host MUST implement the facility to generate and validate UDP checksums.*	4.1.3.4	X						E				X	*3
Silently discard bad checksum *If a UDP datagram is received with a checksum that is non-zero and invalid, UDP MUST silently discard the datagram.*	4.2.3.4	X						E				X	*3
Sender option to not generate checksum	4.1.3.4			X				X				X	*3

Transport Layer UDP - Requirements Summary (RFC-1122 Paragraph 4.1)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
An application MAY optionally be able to control whether a UDP checksum will be generated…													
Default is to checksum *…but it MUST default to checksumming on.*	4.1.3.4	X						E				X	*3
Receiver option is to require checksum *An application MAY optionally be able to control whether UDP datagrams without checksums should be discarded or passed to the application.*	4.1.3.4			X				X				X	*3
UDP Multihoming													
Pass spec-destination address to application *When a UDP datagram is received, its specific-destination address MUST be passed up to the application layer.*	4.1.3.5	X						E				X	
Application layer can specify local IP address *An application program MUST be able to specify the IP source address to be used for sending a UDP datagram…*	4.1.3.5	X						E				X	
Application layer specify wildcard local IP address *...or to leave it unspecified (in which case the networking software will choose an appropriate source address).*	4.1.3.5	X						E				X	
Application layer notified of local IP address used *There SHOULD be a way to communicate the chosen source address up to the application layer (e.g, so that the application can later receive a reply datagram only from the corresponding interface).*	4.1.3.5		X					X				X	
Bad IP source address silently discarded by UDP/IP	4.1.3.6	X						E				X	*4

Transport Layer UDP - Requirements Summary (RFC-1122 Paragraph 4.1)

		INTERNET					PROFILED			AFDX			
	RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES
A UDP datagram received with an invalid IP source address (e.g., a broadcast or multicast address) must be discarded by UDP or by the IP layer.													
Only send valid IP source address *When a host sends a UDP datagram, the source address MUST be (one of) the IP address(es) of the host.*	4.1.3.6	X					X					X	
UDP Application Interface Services													
Full IP interface of 3.4 for application *The application interface to UDP MUST provide the full services of the IP/transport interface described in Section 3.4 of this document. Thus, an application using UDP needs the functions of the GET_SRCADDR(), GET_MAXSIZES(), ADVISE_DELIVPROB(), and RECV_ICMP() calls described in Section 3.4. For example, GET_MAXSIZES() can be used to learn the effective maximum UDP maximum datagram size for a particular {interface,remote host,TOS} triplet.*	4.1.4	X						E				X	
Able to specify TTL, TOS and IP options when sending datagrams *An application-layer program MUST be able to set the TTL and TOS values as well as IP options for sending a UDP datagram, and these values must be passed transparently to the IP layer.*	4.1.4	X						E				X	
Pass received TOS up to application layer *UDP MAY pass the received TOS up to the application layer.*	4.1.4			X				X				X	

Transport Layer UDP - Requirements Summary (RFC-1122 Paragraph 4.1)

	INTERNET					PROFILED			AFDX			
RFC1122	MUST	SHOULD	MAY	SHOULD NOT	MUST NOT	MUST	OPTIONAL	MUST NOT	MUST	OPTIONAL	MUST NOT	NOTES

Comment: The TCP profile uses "Not Applicable" which is less confusing than "NO" [gvb]

*1 The options in IP layer are not used in the AFDX network.
*2 Only for ICMP echo reply not received. [TBC: Needs clarification gvb]
*3 UDP checksums are not used in the AFDX network.
*4 AFDX specification says "No for UDP, Yes for IP", intent apparently is that IP must discard invalid IP source addresses

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

AERONAUTICAL RADIO, Inc.
2551 Riva Road
Annapolis Maryland 21401-7465 USA

PROJECT PAPER 664 Part 8
AIRCRAFT DATA NETWORK
UPPER LAYER and USER SERVICES
Working Paper V5.0
This working draft dated June 14, 2002

Chris Wargo, Computer Networks & Software, Inc. is the industry editor for this Project Paper. He can be reached at: 1-410-280-2763, or chris.wargo@cnsw.com
Electronic copies of the project paper are available on request from the Industry Editor.

This document was authored by Chris Wargo, Computer Networks & Software, Inc., Manu Khanna, Comptel, Inc., and Crispin Netto, Computer Networks & Software Inc., Chris Wargo can be reached at: 1-410-280-2763, or chris.wargo@cnsw.com. Manu Khanna can be reached at mkhanna@comptelinc.com. Crispin Netto can be reached at crispin.netto@cnsw.com.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

This is a working paper prepared by the AEEC. It does not constitute air transport or ARINC approved policy, nor is it endorsed by the U.S. Government, any of its agencies or others who may have participated in its preparation

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

ARINC-664 Aircraft Data Network, Part 8 Upper Layer Services

Authors' Note:

The current document is an attempt to collect the issues and concepts that could be included as the contents of this Part. The expectation is to use this working paper as the means to develop a document that will present the user's, system level requirements that are imposed upon the ADN. Part 8 is intended to provide the guidance for PN or CN usage to support user applications that range from passenger related data to air traffic control communications. The guidance is focused at ensuring interoperability between applications hosted on nodes within an ADN as well as those that are connected to the ADN by internetworking paths that are external to the aircraft.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

TABLE OF CONTENTS

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

LIST OF FIGURES

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

LIST OF TABLES

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

1. INTRODUCTION

1.1. Purpose and Scope of this Document

This part is intended as the guidance specification for the implementation of "application" level services (or upper layer services as viewed within the context of the OSI reference model). The purpose of this Part is the mapping of Upper Layer requirements for the Aeronautical Telecommunications Network (ATN), Airline Operational Control (AOC), and the Data Loader 615A. These services will be supported by either the Compliant Network (CN) or the Profiled Network (PN) using Internet Engineering Task Force (IETF) standard protocols and services. It is the intent of this specification to define the guidance required to ensure the interoperability between user applications that are hosted on the onboard data network and those applications that are hosted on offboard networks. The interconnection of the onboard to the offboard networks may be performed by wireless subnetworks or by cable connection (e.g. gatelink).

1.2. Organization of this Document

1.2.1. Part 8, Upper Layer Services Internet-based Protocols and Services

Part 8 of ARINC Specification 664, "Upper Layer and User Services" is organized as follows:

- Section 1 provides an introduction to this Part
- Section 2 provides a description of the general requirements for user services and their application to the AND
- Section 3 provides a description of the various scenarios to support ATN Upper Layers over TCP/IP.
- Section 4 provides a description of XPORT Switch, a logical entity required to interface ATN Upper Layers with TCP/IP.
- Section 5 provides an approach for interworking for Arinc Specification 637 Distributed ATS Application Architecture.

This Part assumes the following view as shown in figure 1.2.1.

Services are used to provide information and control between different local entities, and Protocols are used to exchange information and control between different remote peer entities. From another perspective the Services view is useful from a Systems Engineering perspective where the behavior and interactions are considered. The Protocol view is much more precise and is intended to be used to establish Software Requirements.

ATTACHMENT 8-1

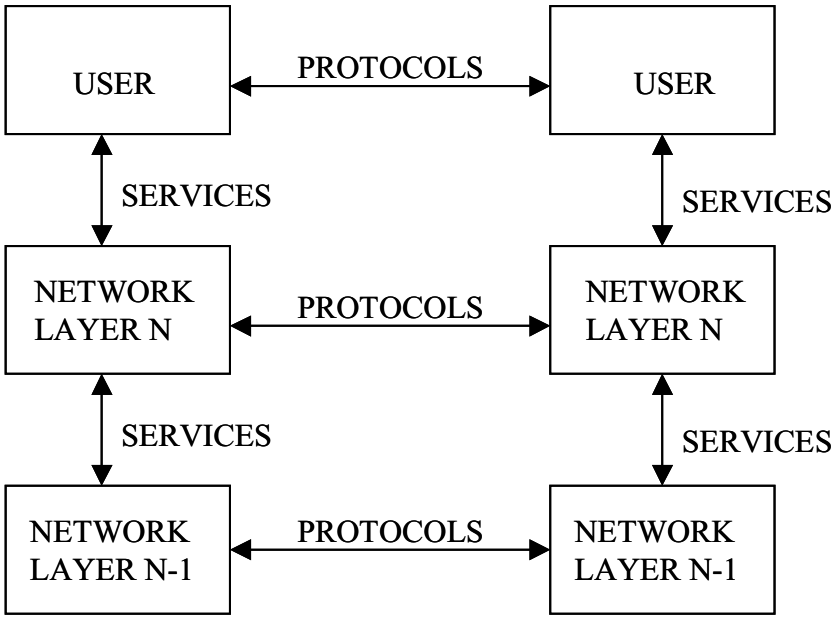


Figure 1.1. Relationship between Protocols and Services

1.3. Relationship to Other Documents

1.3.1. Relationship to ARINC Documents

This section provides a relationship with various ARINC documents to this specification. References to ARINC documents should assume the application of the most recent version of the same.

1.3.1.1. ARINC 664 Specification 664, Aircraft Data Network

The ARINC Specification 664 for "Aircraft Data Network" is structured into multiple parts, each specifying different layers of the OSI Reference Model or other interface to the ADN. System considerations and other communication network topics are specified in other Parts of the ADN specification. Each Part is independent and will evolve on its own time line.

The multiple Parts of ARINC 664 (at the time the release of this Part) are listed as follows:

ARINC-664 Aircraft Data Network Part 1, System Concepts
ARINC-664 Aircraft Data Network Part 2, Ethernet Physical and Datalink Layers, Specification
ARINC-664 Aircraft Data Network Part 3, Internet Based Protocols and Services
ARINC-664 Aircraft Data Network Part 4, Internet Based Address Structures and Assigned Numbers
ARINC-664 Aircraft Data Network Part 5, Network Interconnection Devices
ARINC-664 Aircraft Data Network Part 6, Network Management Specification

ARINC-664 Aircraft Data Network Part 7, An Implementation of a Deterministic Network
ARINC-664 Aircraft Data Network Part 8, Upper Layer and User Services

1.3.1.2. Relationship to ARINC 758, 656, 637A

*** TBD

1.3.1.3. Other ARINC Documents

A list of other ARINC documents that are related to this specification are listed below. When avionics systems and subsystems are designed to use the capabilities provided by this Specification, they should incorporate the provisions of this Specification by reference. References to this Specification should assume the application of the most recent version of this Specification.
List ***TBD

1.3.2. Relationship to Internet Documents

- IETF RFC 2002 - IP Mobility Support.
- IETF RFC 1825 - Security Architecture for the Internet Protocol
- IETF RFC 1826 - IP Authentication Header
- IETF RFC 1827 - IP Encapsulating Security Payload
- IETF RFC 1888 - OSI NSAPs and IPv6
- IETF RFC 2373 - IPv6 Addressing Architecture

1.3.3. Relationship to ICAO Documents.

*** Describe the ICAO Manual 9705

1.3.4. Other references

- IPv6 - The New Internet Protocol, Christian Huitema
- Mobile IP - Design Principles and Practices, Charles E. Perkins
- Mobile IP - The Internet Unplugged, James D. Solomon
- IP Quality of Service, Srinivas Vegesna

1.4. OSI Reference Model (OSI RM)

This Part of the Specification focuses on the Protocols and Services provided by Layers 5 and higher. Additional information on the OSI RM is provided in Part 1 of this specification.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

2. USER BASED SERVICES

2.1. Introduction

This section describes a protocol reference model and typical interoperability issues.

2.2. Reference Model

In addition to the onboard user applications, the ADN must be able to support connections to various user applications that are located in off board domains or Autonomous Networks (AN). For the AN, the ADN must be able to transport the data while supporting end-to-end interoperability. Part of this interoperability involves fulfilling network performance parameters that vary from user application to user application. The designer of the ADN must provide for the integrated use of the ADN in order to meet economic considerations. This is becoming increasingly important as the future free flight concepts begin to move towards a more tightly coupled set of user services. These user services are necessary to perform the integrated functional processes across the aircraft's flight management systems, the airline's operational control systems, and air traffic authorities' systems. The emerging set of tightly coupled services for FMS, AOC, and ATM requires an increased use of data communications in place of today's voice communications. However, the existing applications for these services that are data link enabled are not IP-based. The purpose of this document is to provide guidance for the transitioning of these existing non-IP-based applications to the IP-based ADN environment.

2.3. Typical Interoperability issues

Ideally, the achievement of interoperability between user applications would be accomplished with the minimal use of specialty gateway functions. The predominant set of interoperability issues for aviation users has been created by the implementation of the ICAO ATN Standards (a vertical set of standards useful only in the aviation community, and thus, creating in essence a closed network despite the use of open system ideas). Meanwhile, the rest of the world's industries, including those of the aviation's trading partners, have adopted IETF set of standards as the method to support data transport for all means of commerce, entertainment, etc. Thus, the direct interoperability between the trading partners of the aviation community will require the implementation of gateway functionality to interconnect one another's networks. The secondary set of issues in interoperability are related to legacy system compatibility. The issues are both protocol and system architecture related. The framework, or context, for understanding both the protocol and architecture interoperability issues is shown in Figure 2.1.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

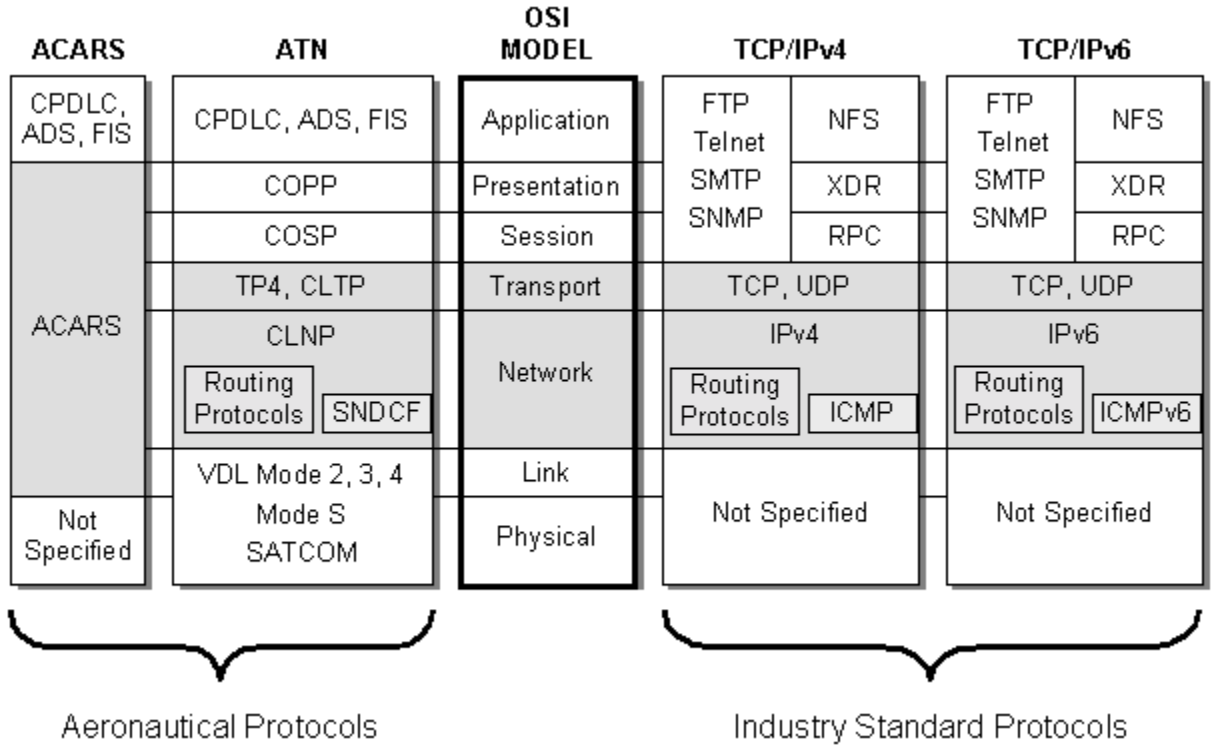


Figure 2.1 Framework for Interoperability

The focus of this document is to standardize the approaches to interoperability in order to support ATN applications over TCP/IP.

2.4. ATN Overview

For information summarizing ATN protocol architecture see Appendix A or refer to ARINC 637.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

3. ATN OVER TCP/IP

3.1. Introduction

This section focuses on issues related to supporting ATN services on top of TCP/IP stack. Five operational scenarios are discussed to define interoperability requirements.

1. All ATN Universe
2. TP4/CLNP to TCP/IP (Airborne Gateway)
3. TP4/CLNP to TCP/IP (Ground based Gateway)
4. ATN on TCP/IP
5. ATN on TCP/IPv6

It is expected that these interoperability architectures will be reviewed and feasible model(s) be studied further.

3.2. Interoperability Scenario - All ATN Universe (ATN to ATN)

There are no interoperability issues in this scenario and it is presented here only for reference. In this scenario, both communicating ends are using full ATN stack and are attached to an ATN compliant network. Figure 3.1 depicts this scenario.

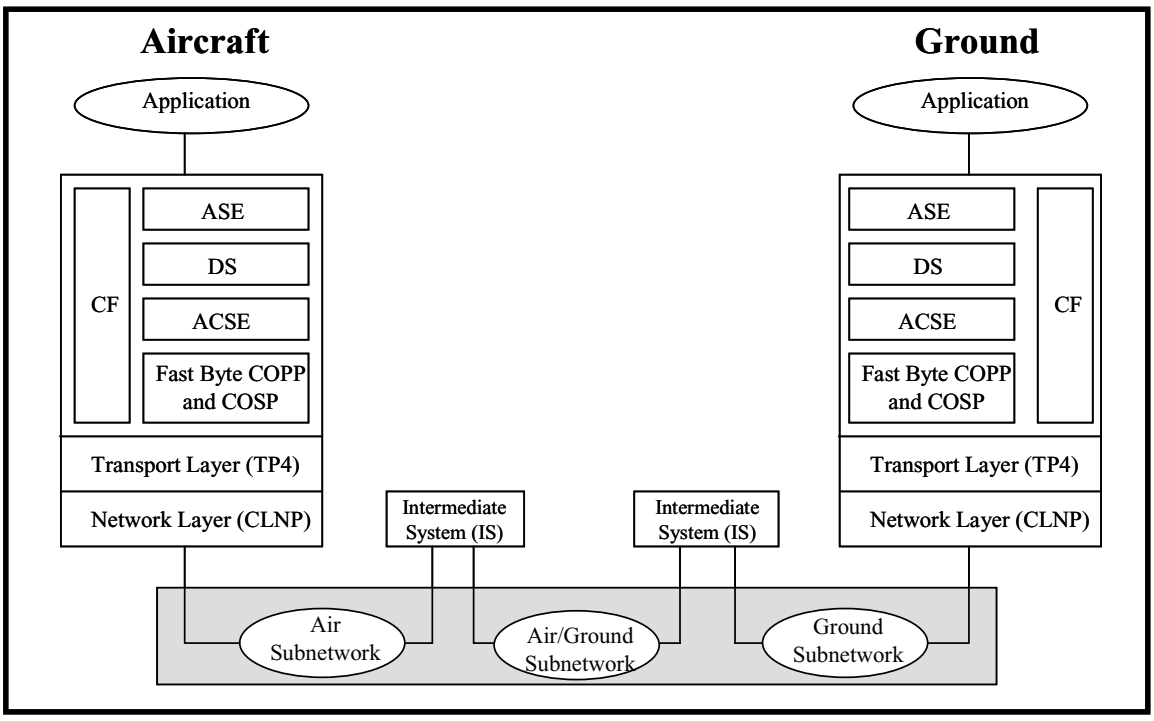


Figure 3.1 All ATN Network

3.2.1. Mobility

Mobility under this scenario is provided as per ATN SARPs.

3.2.2. Quality of Service

Quality of Service under this scenario is provided as per ATN SARPs.

3.2.3. Security

Security under this scenario is provided as per ATN SARPs.

3.3. Interoperability Scenario – TP4/CLNP to TCP/IP (Airborne Gateway)

It is quite likely that there are ground based applications which are ATN compliant. To make these applications available to ADN, we could run ATN applications on top of TCP/IP on the aircraft side while the ground end is still running on top of TP4/CLNP. It becomes immediately obvious that in this scenario a gateway function must exist. Figure 3.2 depicts this scenario. It is assumed that the gateway would reside onboard the aircraft.

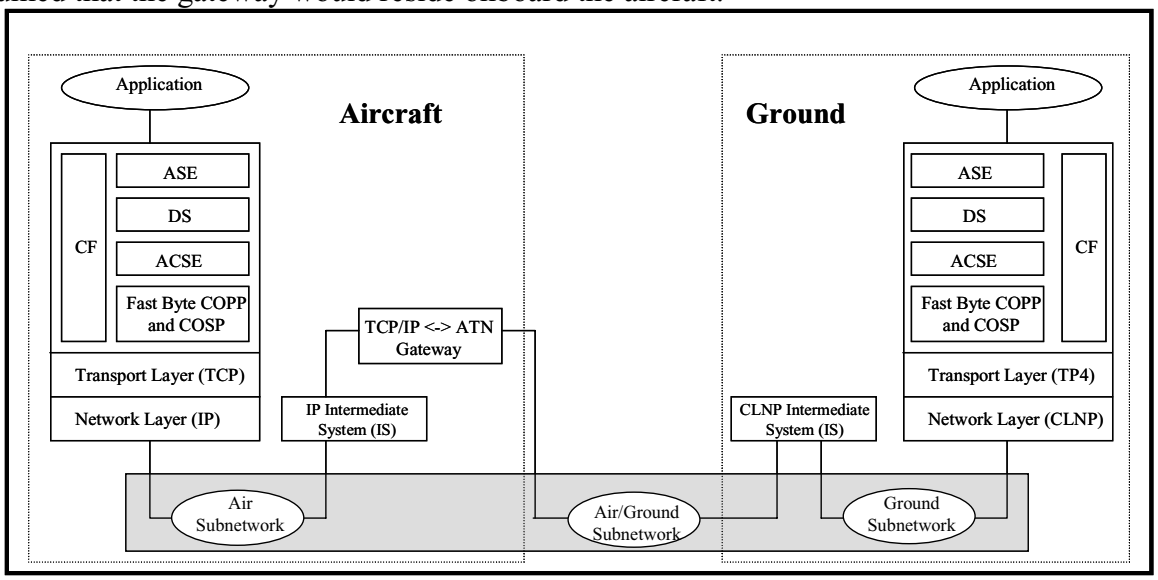


Figure 3.2 TCP/IP<->ATN (Airborne Gateway)

3.3.1. Gateway Function

The gateway function is responsible for:

- Terminating TP4/CLNP on one side and translating that connection into TCP/IP on the other side, and vice versa.
- As such gateway will be required to manage both TCP and TP4 termination points.
- In addition to performing translation from one protocol to another, gateway will also be required to perform address translation between IP domain and ATN domain. It is expected that the translation table will be resident on the gateway itself and that the gateway will not depend on any external service to perform the address translation.

Gateway function can be implemented at the Application layer or at the Transport layer. Following sections describe these two options.

3.3.1.1. Application Layer Gateway

Application layer gateway acts as a proxy server for both Air and Ground endpoints. In addition to performing address translation at the TCP/IP layer, it also performs translation for addresses contained within the application data. Hence, Aircraft side deals only with IP addresses and Ground side deals only with ATN addresses. Later sections explain how IP addresses can be carried in NSAP fields to maintain compatibility with the upper layers.

Figure 3.3 shows the gateway function at the application layer. Application layer gateway, however, has its shortcomings. A new gateway function is needed every time a new application is added. In addition, the gateway can easily become a bottle-neck since it will be required to maintain a context at the application level of all application end-points. Hence, this scenario will not be studied further.

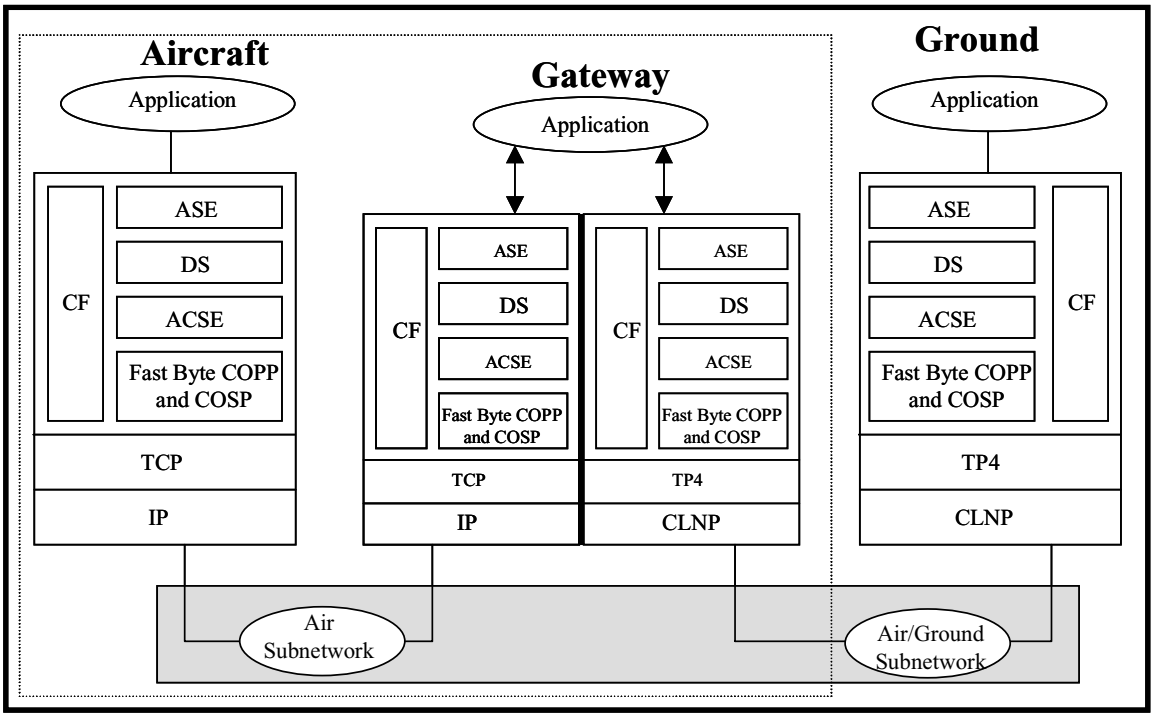


Figure 3.3 Application Layer Gateway (Airborne)

3.3.1.2. Transport Layer Gateway

The gateway at the transport layer terminates TP4/CLNP connections on one side, puts the upper layer payload in TCP/IP connections on the other side and transmits the data on the TCP/IP network. A similar process takes place for the data traversing in the other direction. The gateway

is responsible for carrying out the translation between ATN and IP addresses at the network and transport layers.

Since, the gateway does not have any visibility into the application data, any addresses contained in the application data are translated by the Aircraft side i.e. ADN. Application data always contains ATN NSAPs.

No address translation is required at the Ground end points. Figure 3.4 shows the gateway function at the transport layer.

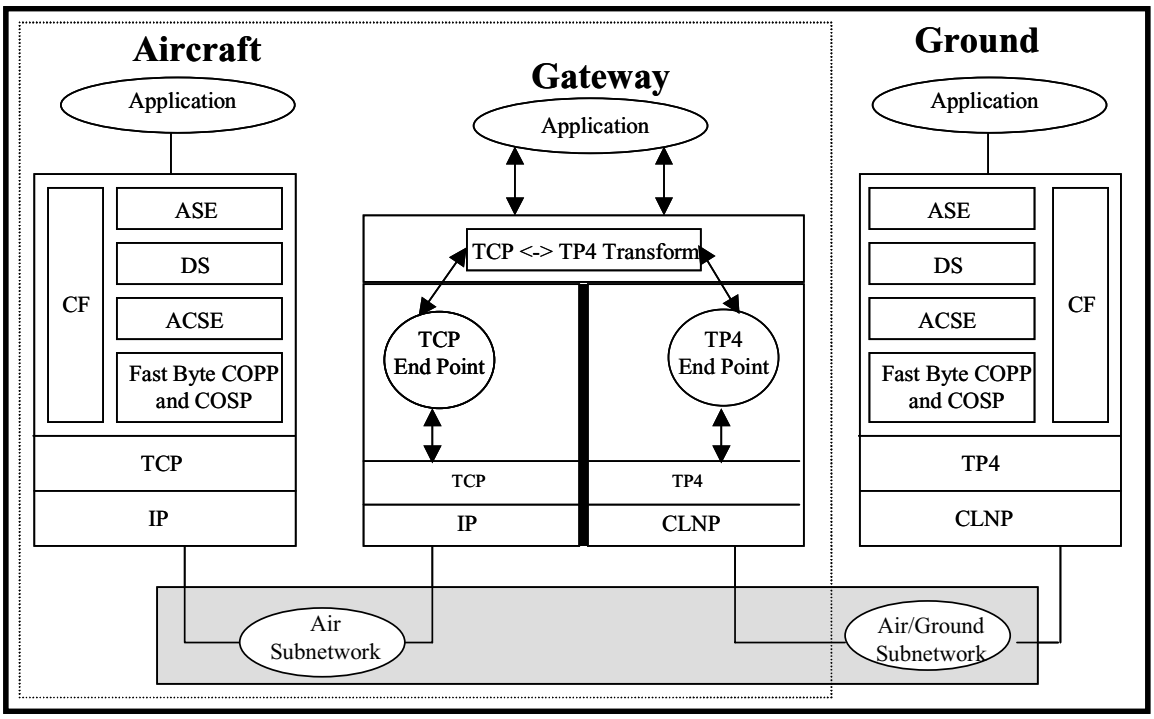


Figure 3.4 Transport Layer Gateway (Airborne)

3.3.2. Mobility

Since it is assumed that the gateway resides within the aircraft's onboard network, the TCP/IP stacks will be independent of mobility aspects. The "outer" side of the gateway which uses TP4/CLNP will provide for mobility support that is being used today as per the ATN SARPs.

3.3.3. Quality of Service

Quality of Service under this scenario is provided as per ATN SARPs.

3.3.4. Security

Security under this scenario is provided as per ATN SARPs.

3.4. Interoperability Scenario – TP4/CLNP to TCP/IP (Ground based Gateway)

This case looks at having ground based applications which are ATN compliant. To make these applications available to ADN, we could run ATN applications on top of TCP/IP on the aircraft side while the ground end is still running on top of TP4/CLNP. It becomes immediately obvious that in this scenario a gateway function must exist. Figure 3.5 depicts this scenario. Here, it is assumed that the gateway would be part of the ground network.

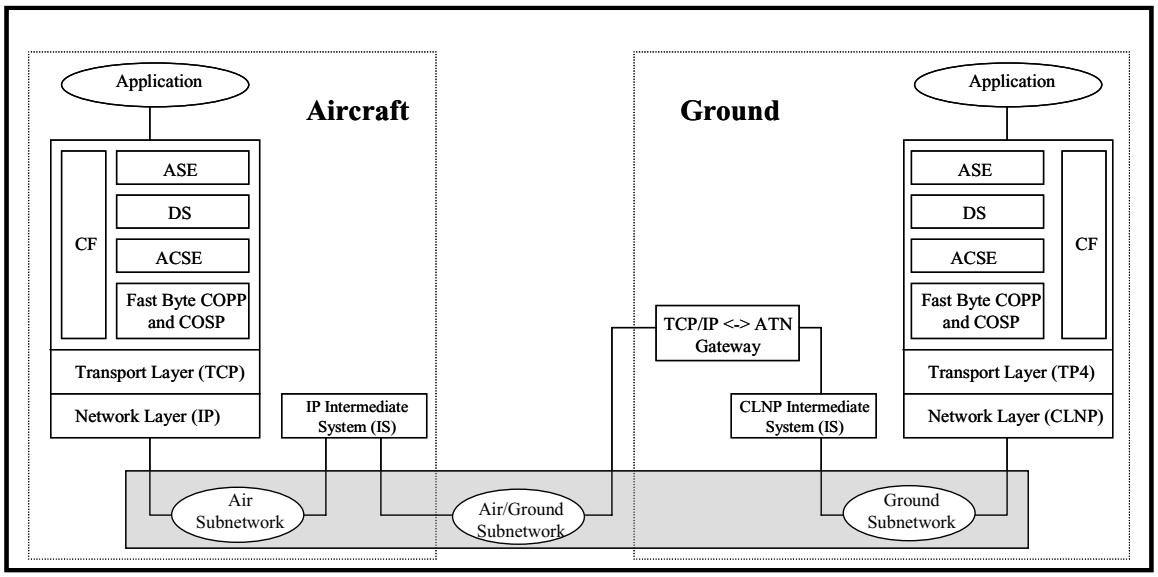


Figure 3.5 TCP/IP <-> ATN (Ground based Gateway)

3.4.1. Gateway Function

The gateway function is responsible for:

- Terminating TP4/CLNP on one side and translating that connection into TCP/IP on the other side, and vice versa.
- As such gateway will be required to manage both TCP and TP4 termination points.
- In addition to performing translation from one protocol to another, gateway will also be required to perform address translation between IP domain and ATN domain. It is expected that the translation table will be resident on the gateway itself and that the gateway will not depend on any external service to perform the address translation.

Gateway function can be implemented at the Application layer or at the Transport layer. However, the Application layer Gateway will not be studied, due to the shortcomings noted in Section 3.3.1.1. The following section describe the transport layer gateway.

3.4.1.1. Transport Layer Gateway

The gateway at the transport layer terminates TP4/CLNP connections on one side, puts the upper layer payload in TCP/IP connections on the other side and transmits the data on the TCP/IP network. A similar process takes place for the data traversing in the other direction. The gateway

is responsible for carrying out the translation between ATN and IP addresses at the network and transport layers.

Since, the gateway does not have any visibility into the application data, any addresses contained in the application data are translated by the Aircraft side i.e. ADN. Application data always contains ATN NSAPs.

No address translation is required at the Ground end points. Figure 3.6 shows the gateway function at the transport layer.

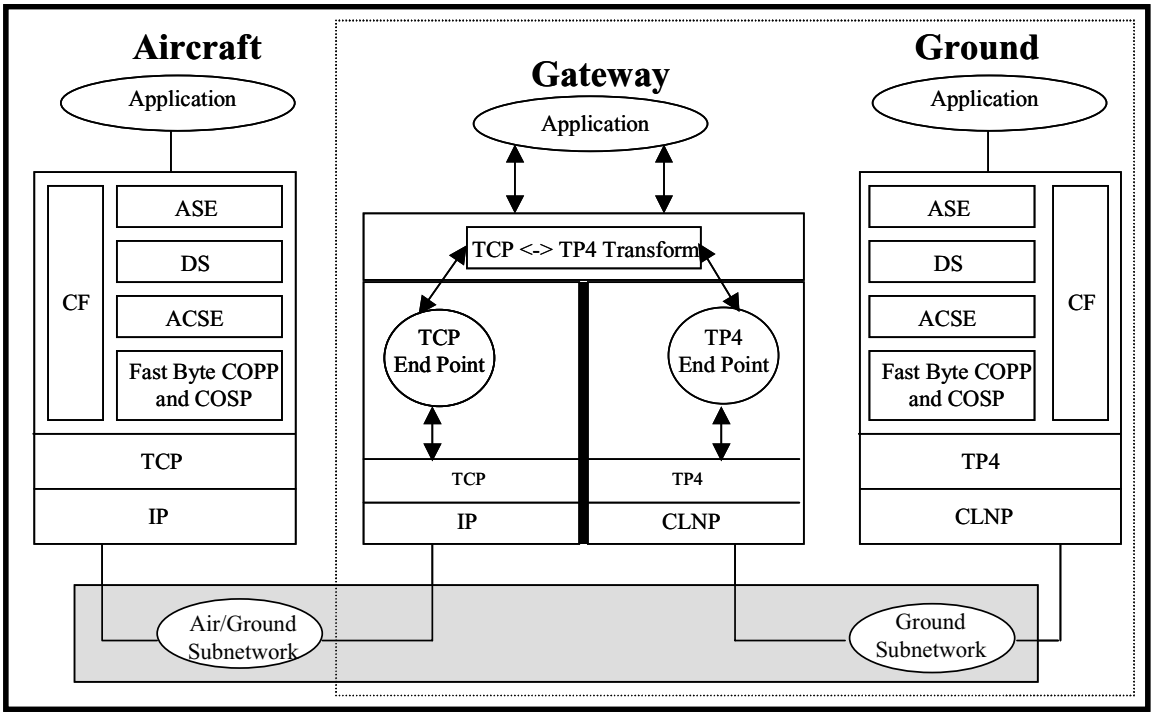


Figure 3.6 Transport Layer Gateway (Ground based)

3.4.2. Mobility

Since it is assumed that the gateway resides as part of the ground network, support for Mobility will be provided by Mobile IP. A summary on Mobile IP technology is presented in Appendix-C. Further details can be referenced from the IETF standard RFC 2002.

3.4.3. Quality of Service

Support for Quality of Service at IP Layer 3 is best understood by studying the concepts of Flows, DiffServ, RSVP and other real time protocols. Appendix-E contains a description of these technologies that form the core of IP QoS.

3.4.4. Security

Security under this scenario is provided by IPSec. A description of the Authentication Header, and, the Encapsulating Payload Header is presented in Appendix-D.

3.5. Interoperability Scenario – ATN on TCP/IP

In this scenario, all ATN end points will be operating on top of TCP/IP. Figure 3.7 depicts this scenario.

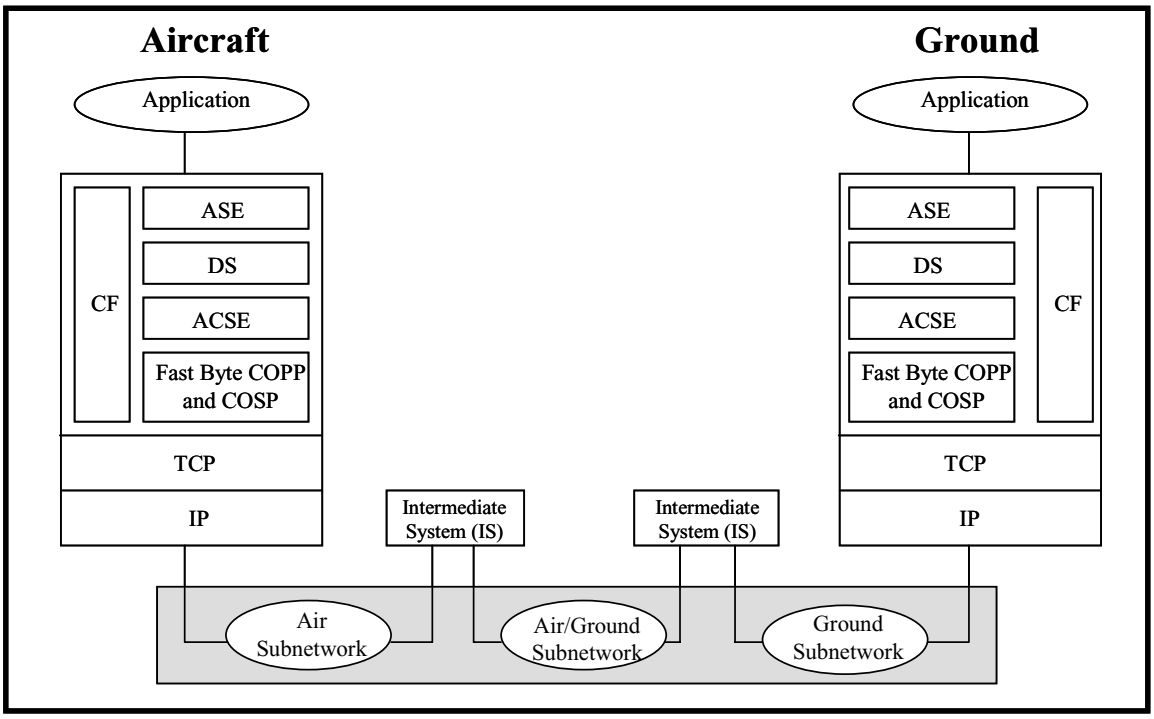


Figure 3.7 All ATN application on TCP/IP

In this scenario the main issue is that of address translation. There are basically, two options to manage it:

- *Addressing Option I:* All addresses are IP addresses. This implies that ATN applications that currently exchange NSAP/TSAP addresses will exchange IP addresses instead. Existing address format will have to be refined to indicate whether an IP or ATN address is contained in a protocol field. e.g. Air-CM sends CPDLC-ATN address to the ground-CM for initiating CPDLC service. This address will need to be IP based.

- *Addressing Option II:* ATN applications use ATN addresses as currently defined. This will require a Address Translation function. Address translation function can either be implemented at the end points or at a central server. For ADN applications, the translation table is expected to be relatively small and should be implemented at the end points.

In Section 4., we give additional details for address translation for both options in the context of XPORT Switch.

3.5.1. Mobility

Support for Mobility will be provided by Mobile IP. A summary on Mobile IP technology is presented in Appendix-C. Further details can be referenced from the IETF standard RFC 2002.

3.5.2. Quality of Service

Support for Quality of Service at IP Layer 3 is best understood by studying the concepts of Flows, DiffServ, RSVP and other real time protocols. Appendix-E contains a description of these technologies that form the core of IP QoS.

3.5.3. Security

There are numerous security protocols and cryptographic algorithms in use throughout portions of the global Internet. A complete treatment of these protocols and algorithms is beyond scope of this document. Only those protocols that provide solutions to the security problems introduced by mobility in general and Mobile IP in particular are discussed. Specifically, the discussion is confined to the IP-layer security and key management protocols.

The Security Architecture for the Internet Protocol [RFC 1825] defines a framework for security at the IP layer. There are two specific headers that are used to provide security in IPv4 and IPv6. These are the IP Authentication Header [RFC 1826], and IP Encapsulating Payload [RFC 1827] header.

A description of the Authentication Header, and, the Encapsulating Payload Header is presented in Appendix-D.

3.6. Interoperability Scenario – ATN on TCP/IPv6

Under this scenario it is assumed that the network has migrated to IPv6. All addresses used will be IPv6. ATN applications will carry IPv6 addresses within NSAP structures.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

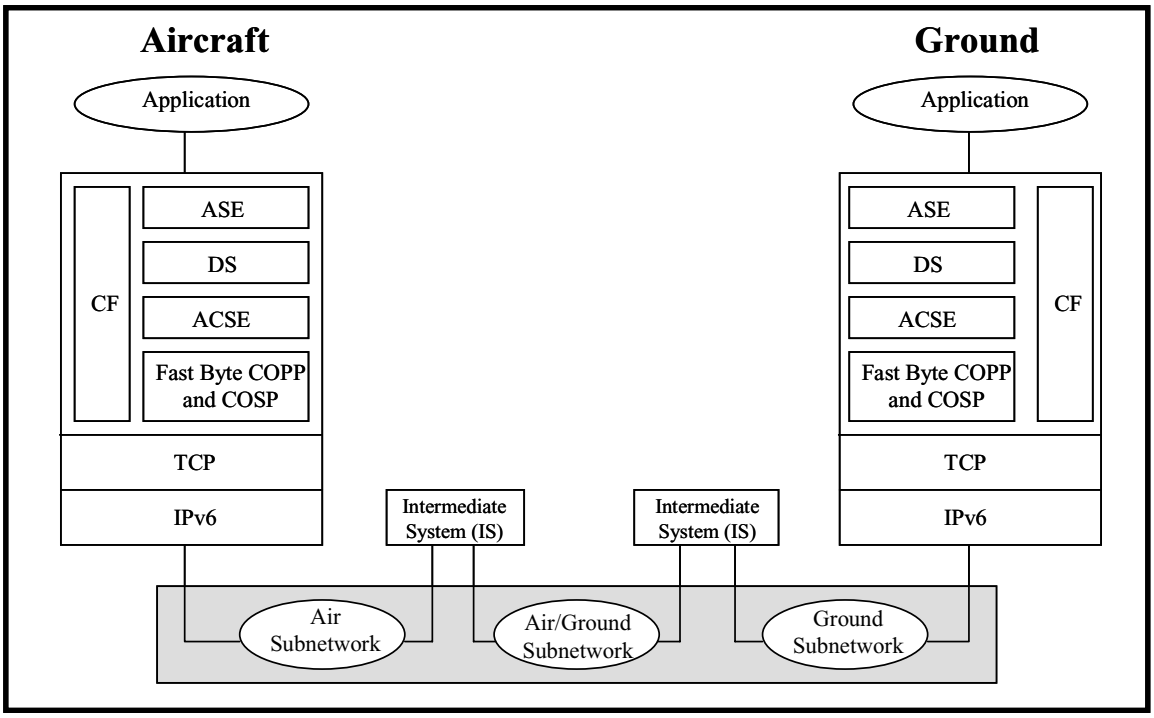


Figure 3.8 ATN on TCP/IPv6

Figure 3.8 shows the ATN on TCP/IPv6 scenario.

3.6.1. Mobility

Support for Mobility will be provided by Mobile IPv6. A summary on Mobile IPv6 is presented in Appendix-C.

3.6.2. Quality of Service

Appendix-E contains a description of technologies that form the core of IP QoS.

3.6.3. Security

The discussion on Security is presented in Appendix-D

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

4. XPORT SWITCH – ATN TO TCP INTERFACE

For ATN upper layers to use the services of TCP/IP some glue logic or interface will be needed. XPORT Switch is a logical entity that provides ATN to TCP interface. The following discussion is applicable to any end point where ATN layers are operating over TCP/IP. There are various options as to how and where in the stack this interface is implemented:

- *Option I:* A separate function/thin layer translates the TP4 primitives into TCP/IP API.
- *Option II:* Upper layers directly interface with TCP using TCP/IP API like Sockets.
- *Option III:* Upper layers directly interface with TCP using TCP primitives (theoretically possible but not likely). A designer is not likely to choose this option since it may require code modifications to off-the-shelf products.

Figure 4.1 shows XPORT Switch in an ATN Upper Layers over TCP/IP stack.

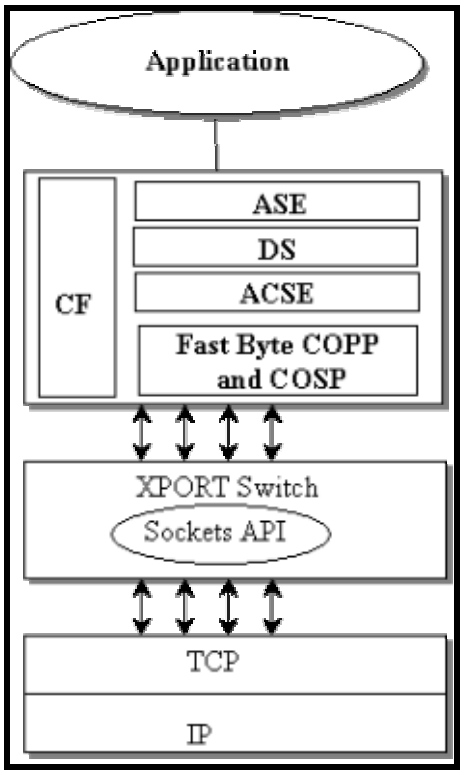


Figure 4.1 XPORT Switch

Regardless of the option selected to implement it, XPORT Switch must do the following tasks at the minimum:

- *Interface to upper layers:* XPORT provides primitives for the layers above it to simulate services provided by TP4.
- *Address Translation:* All interaction with TCP/IP is based on valid source and destination IP addresses. Hence, XPORT must be able to deduce valid IP addressees from the information passed to it.
- *Well Known Port Mapping:* XPORT will be required to detect any calls to well known TSEL and map them to a corresponding well known TCP port.
- *Interface to TCP/IP:* XPORT will invoke TCP/IP services using either a Sockets or TLI API.

Following sections contain a more detailed description of the above functions.

4.1. Address Translation

There are two options to achieve ATN to IP address inter-working depending upon whether the upper layers are aware of IP addresses or work strictly with ATN NSAPs.

4.1.1. IP addresses embedded in NSAP

Under this option, all applications will deal only with IP addresses. However, to maintain the structure of ATN application PDUs, IP addresses must be embedded in NSAP. RFC 1888 specifies a mechanism to carry IP addresses in 20 byte NSAP field. Using this mechanism all applications can contain valid IP addresses in NSAP fields without disturbing the PDU structure. XPORT will extract the IP address passed by upper layers and invoke TCP/IP services using the published API. In the reverse direction XPORT will package the IP address as per RFC 1888 in NSAP. The same general mechanism will work for both IPv6 and IPv4 since IPv6 specifies a mechanism to embed IPv4 addresses in IPv6 address structure.

4.1.1.1. IP addresses embedded in NSAP as per RFC1888

An ATN address structure is given in figure 4.2.

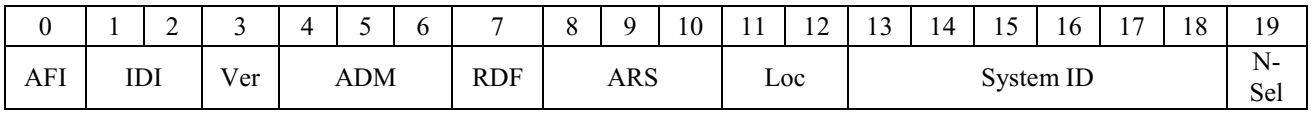


Figure 4.2 20 Byte ATN Address Structure

Explanation of each of these fields is given in Appendix-B. AFI value currently assigned for ATN addresses is 47 and IDI is 27. According to RFC1888 to embed an IPv6 address in an ATN NSAP address, AFI should be set to 35 and IDI should be set to 0. Next 16 bytes contain an IPv6 address and N-Sel is set is 0.

To embed an IPv4 address in NSAP, first use the rules to embed an IPv4 in IPv6 format and then embed that address in ATN NSAP as described above. Four byte IPv4 addresses are embedded in 16 byte IPv6 addresses by setting first 10 bytes of IPv6 address to zero, bytes 11th and 12th to 0xFFFF and last four bytes to the actual IPv4 address. RFC 2373 provides additional details on embedding IPv4 addresses in IPv6 address format.

Figure 4.3 shows an IPv4 address, 10.1.1.89, embedded in ATN NSAP structure.

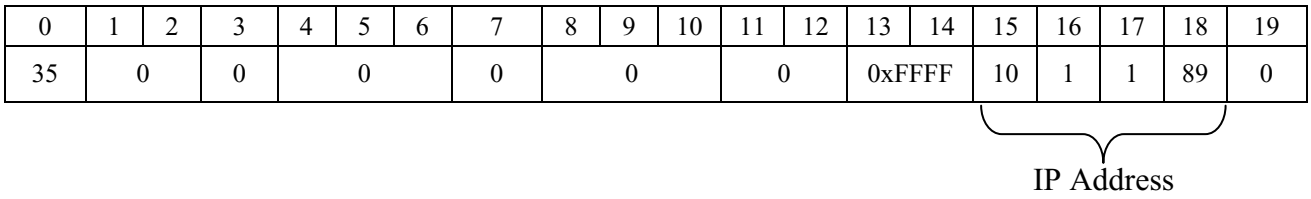


Figure 4.3 IPv4 address embedded in 20 Byte ATN Address Structure

Please note that the use of the scheme described in RFC1888 will require formal approval by ISO.

4.1.2. ATN applications speak true NSAP

Under thus option, XPORT will be responsible for translating addresses to IP. Both NSAP and TSEL may require translation. It is estimated that the potential number of addresses that an aircraft will be dealing with, are small enough that a static table will suffice to map ATN addresses to IP addresses. XPORT will use this table to do the address translation. Well known TSELs currently in use by ATN applications may have a conflict with well known TCP ports. XPORT will be required to identify connections to any of these well known TSELs and map them to well known TCP ports defined for ATN applications. Again, a static table on ADN end points should suffice to perform the mapping.

4.2. Interface to Layers Above and Below

XPORT will pose as TP4 to upper layers and transform the TP4 primitives to corresponding TCP/IP API calls. Hence, upper layers will invoke XPORT services using the same primitives as for TP4, thus preserving the interface between the layers. XPORT will, in turn, interface with TCP using Sockets or some similarly defined API. Table 4.1 shows a mapping of TP4 primitives to Sockets calls.

Table 4.1 TP4 Primitives <-> Socket Calls

TP4- Primitives		*Equivalent Socket Calls**
T-CONNECT	request	socket(), bind(), connect(), setsockopt()
	indication	Return from accept(), getsockopt() following socket(), bind() and listen().

ATTACHMENT 8-1

	response	No applicable API call
	confirmation	Return from connect().
T-DATA	request	recv(), send()
	indication	Return from recv(), send(), select()
T-EXPEDITED DATA	request	Send() with MSG_OOB flag set
	indication	Asynchronous events, recv() with MSG_OOB flag set
T-DISCONNECT	request	Close(), setsockopt()
	indication	Asynchronous events, error return, getsockopt()

*Sockets API can vary slightly from one implementation to another.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

5. DISTRIBUTED ATS APPLICATION ARCHITECTURE (ALTERNATIVES)

5.1. Introduction

This section references a scenario from ARINC Specification 637 (Attachment 5, Figures 5-1, 5-2), where FMC is running on TCP/IP and CMU is communicating using ATN stack. The Gateway required for translating TCP/IP <-> ATN connections has been defined at the Application Layer. The shortcomings of using the Gateway at the Application Layer was discussed in section 3.3.1.1. Another approach suggested the creation of an interface at the Dialog Service layer. The following section describes an alternative approach where the Gateway would be defined at the Transport Layer.

5.2. Transport Layer Gateway for Distributed ATS Application Architecture

The scenario matches the approach taken for Transport Layer gateway as described in section 3.1.1.2. The scenario is depicted in figure 5.1

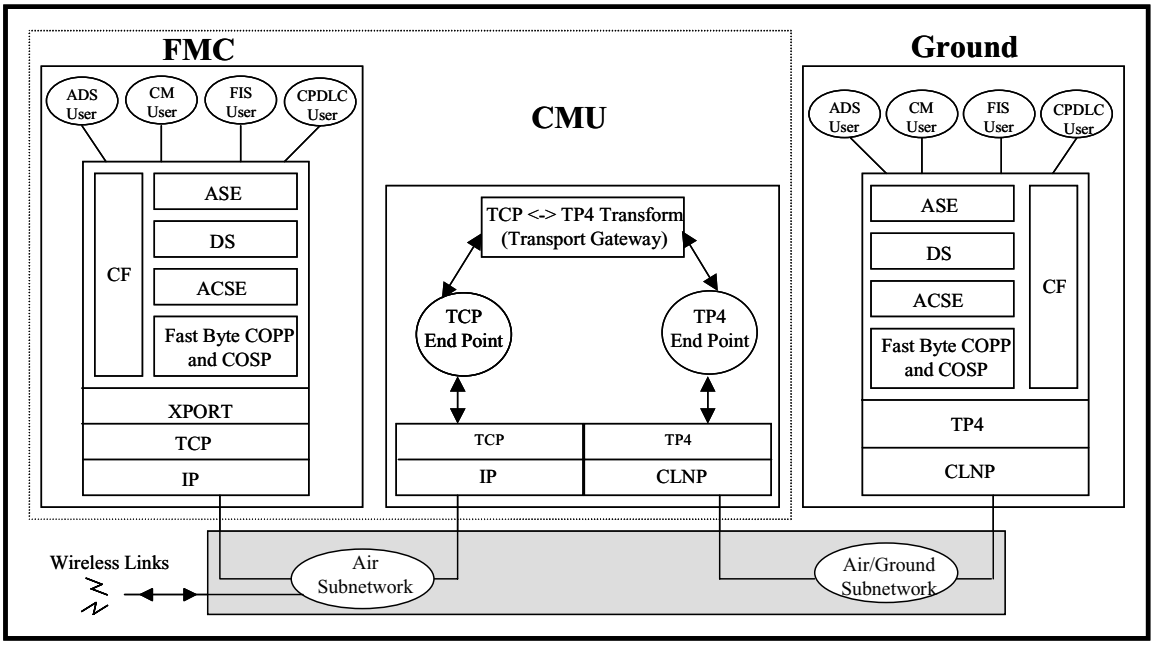


Figure 5.1 Distributed Architecture

In this scenario, FMC supports ATN applications over TCP/IP. XPORT Switch interposed in the stack provides all the functionality needed for ATN Upper Layers to interface with TCP/IP as detailed in the previous section.

CMU acts as a Transport Gateway as described in section 3.3.1.2 and 3.4.1.1. It terminates and maps TP4 connections from the Ground to TCP connections to FMC. Transport Gateway is responsible for translating IP addresses to ATN addresses and vice versa.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

6. GLOSSARY

Reference the Glossary shown in 664 Part 1.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

7. APPENDIX A – ATN ARCHITECTURE SUMMARY

7.1. Introduction

This appendix summarizes the ATN protocol architecture.

TBD.

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

8. APPENDIX B – ATN ADDRESS STRUCTURE

8.1. Introduction

This appendix describes the ATN Address Structure.

8.2. ATN Address Structure

ATN uses 20 bytes NSAP address structure as shown in figure B.1.

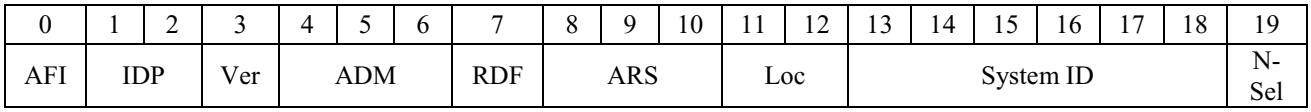


Figure B.1 20 Byte ATN Address Structure

Table C.1 describes various fields of the address structure:

Table B.1 ATN NSAP Addressing

Field	Size (octets)	Assigned Value	Comments
Authority Format Identifier (AFI)	1	47	ISO 6523 format
Initial Domain Identifier (IDI)	2	27	ATN NSAP Address
Version (Ver)	1	0x01–Fixed AINSC 0x41- Mobile AINSC 0x81- Fixed AINSC 0xC1 – Mobile ATSC	
Administration (ADM)	3		Three character alphanumeric ISO 3166 Country code; Or assigned by ICAO.
Routing Domain Format (RDF)	1		
Administrative Region Selector (ARS)	3	24-bit unique identifier for Fixed AINSC and ATSC Domains assigned by ADM; 24-bit ICAO Aircraft Address for Mobile AINSC and ATSC domains.	
Location (Loc)	2	Any value	
System Identifier	6	Any value	
Selector (N-Sel)	1	Any value	

For additional information on ATN addressing please refer to vol. 5 of ATN SARPS.

9. APPENDIX C – MOBILITY

The following sections illustrates briefly the concepts of Mobile IPv4 and IPv6.

9.1. Mobile IPv4

Mobile IP allows mobile nodes to use two IP addresses – a static address called the *home address* which identifies TCP connections, and, a dynamic address called the *care-of-address* which changes at each new point of attachment and can be thought of as the mobile node's topologically significant address; it indicates the network number and thus identifies the mobile node's point of attachment with respect to the network topology. The home address makes it appear that the mobile node is continually able to receive data on its *home network*, where Mobile IP requires the existence of a network node known as the *home agent*. Whenever the mobile node is not attached to its home network (and is therefore attached to what is termed a *foreign network* which requires the existence of a network node known as *foreign agent*), the home agent gets all the packets destined for the mobile node and arranges to deliver them to the mobile node's current point of attachment.

Whenever the mobile node moves, it *registers* its new care-of address with its home agent. To get a packet to a mobile node from its home network, the home agent delivers the packet from the home network to the care-of address. The further delivery requires that the packet be modified so that the care-of address appears as the destination IP address. This modification can be understood as a packet transformation or, more specifically, a *redirection*. When the packet arrives at the care-of address, the reverse transformation is applied so that the packet once again appears to have the mobile node's home address as the destination IP address. When the packet arrives at the mobile node, addressed to the home address, it will be processed properly by TCP or whatever higher level protocol logically receives it from the mobile node's IP (that is, layer 3) processing layer.

In Mobile IP the home agent redirects packets from the home network to the care-of address by constructing a new IP header that contains the mobile node's care-of address as the destination IP address. This new header then shields or encapsulates the original packet, causing the mobile node's home address to have no effect on the encapsulated packet's routing until it arrives at the care-of address. Such *encapsulation* is also called *tunneling*, which suggests that the packet burrows through the Internet, bypassing the usual effects of IP routing.

Mobile IP is best understood as the cooperation of three separable mechanisms:

- Discovering the care-of-address
- Registering the care-of-address
- Tunneling to the care-of-address

Figure C.1 shows an high level example for implementation of Mobile IP. Further details on Mobile IP can be obtained from the IETF standard RFC 2002.

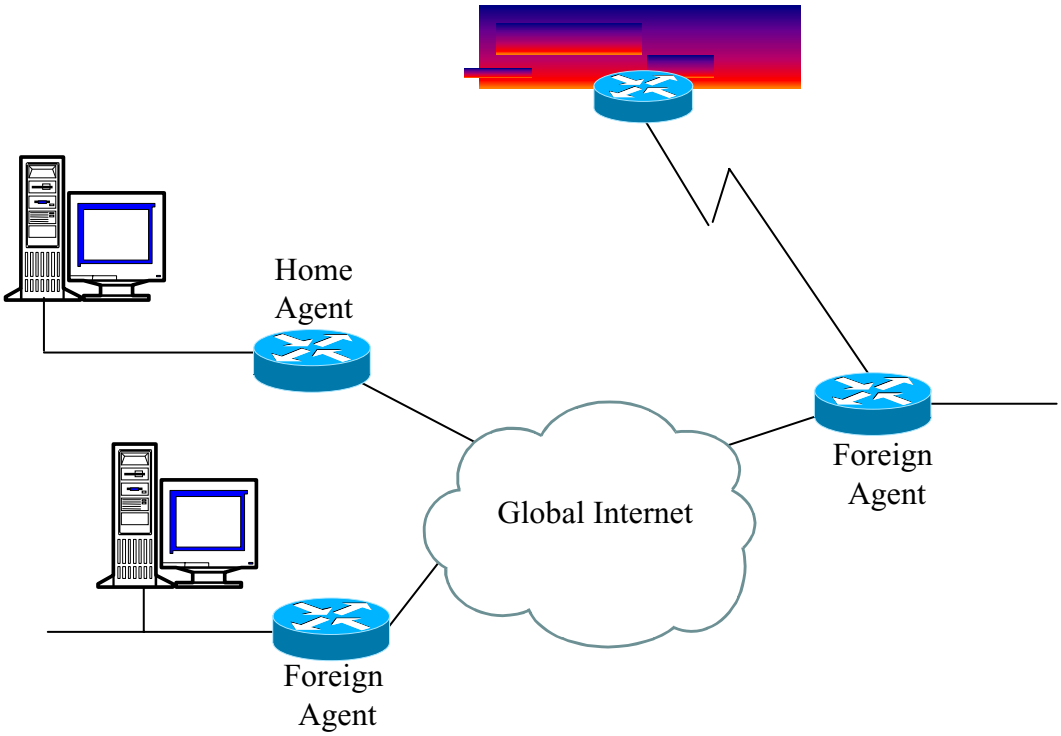


Figure C.1 Mobile IPv4 implementation

9.2. Mobile IPv6

Mobility Support in IPv6, follows the design for Mobile IPv4. It retains the ideas of a home network, home agent, and the use of encapsulation to deliver packets from the home network to the mobile node's current point of attachment. While discovery of a care-of address is still required, a mobile node can configure its a care-of address by using Stateless Address Autoconfiguration and Neighbor Discovery. Thus, foreign agents are not required to support mobility in IPv6. IPv6-within-IPv6 tunneling is also already specified.

Route Optimization

IPv6 mobility borrows heavily from the route optimization ideas specified for IPv4, particularly the idea of delivering binding updates directly to correspondent nodes. When it knows the mobile node's current care-of address, a correspondent node can deliver packets directly to the mobile node's home address without any assistance from the home agent. Route optimization is likely to dramatically improve performance for IPv6 mobile nodes. It is realistic to require this extra functionality of all IPv6 nodes for two reasons. First, on a practical level, IPv6 standards documents are still at an early stage of standardization, so it is possible to place additional requirements on IPv6 nodes. Second, processing binding updates can be implemented as a fairly simple modification to IPv6's use of the destination cache.

Security

One of the biggest differences between IPv6 and IPv4 is that all IPv6 nodes are expected to implement strong authentication and encryption features to improve Internet security. This affords a major simplification for IPv6 mobility support, since all authentication procedures can be assumed to exist when needed and do not have to be specified in the Mobile IPv6 protocol. Even with the security features in IPv6, however, the current working group draft for IPv6 mobility support specifies the use of authentication procedures as infrequently as possible. The reasons for this are twofold. First, good authentication comes at the cost of performance and so should be required only occasionally. Second, questions about the availability of Internet-wide key management are far from resolved at this time.

Source Routing

In contrast to the way in which route optimization is specified in IPv4, in IPv6 correspondent nodes do not tunnel packets to mobile nodes. Instead, they use IPv6 routing headers, which implement a variation of IPv4's *source routing* option. A number of early proposals for supporting mobility in IPv4 specified a similar use of source routing options, but two main problems precluded their use:

- IPv4 source routing options require the receiver of source-routed packets to follow the reversed path to the sender back along the indicated intermediate nodes. This means that malicious nodes using source routes from remote locations within the Internet could impersonate other nodes, a problem exacerbated by the lack of authentication protocols.

- Existing routers exhibit terrible performance when handling source routes. Consequently, the results of deploying other protocols that use source routes have not been favorable.

However, the objections to the use of source routes do not apply to IPv6, because IPv6's more careful specification eliminates the need for source-route reversal and lets routers ignore options that do not need their attention. Consequently, correspondent nodes can use routing headers without penalty. This allows the mobile node to easily determine when a correspondent node does not have the right care-of address. Packets delivered by encapsulation instead of by source routes in a routing header must have been sent by correspondent nodes that need to receive binding updates from the mobile node. It is a further point of contrast to route optimization in IPv4 that, in IPv6 mobility support, the mobile node delivers binding updates to correspondent nodes instead of to the home agent. In IPv6, key management between the mobile node and correspondent node is more likely to be available.

Other features supported by IPv6 mobility include

- coexistence with Internet ingress filtering;

- smooth handoffs, which in Mobile IPv4 is specified for foreign agents as part of route optimization;

- renumbering of home networks; and

- automatic home agent discovery.

10. APPENDIX D – SECURITY

The Security Architecture for the Internet Protocol [RFC 1825] defines a framework for security at the IP layer. There are two specific headers that are used to provide security in IPv4 and IPv6. These are the IP Authentication Header [RFC 1826], and IP Encapsulating Payload [RFC 1827] header.

10.1. IP Authentication Header

The IP Authentication Header is a mechanism for providing strong authentication and integrity checking for the IP header and payload. It might also provide non-repudiation, depending on which cryptographic algorithm is used and how keying is performed. For example, use of an asymmetric digital signature algorithm, such as RSA, could provide non-repudiation.

The Authentication header is usually placed between the IP header and the upper layer header (e.g., TCP) in order to protect the entire packet against being modified in transit and to authenticate the sender. The Authentication header may also appear after any other headers which are examined at each hop, and before any other headers which are not examined at an intermediate hop.

The following are example high level figures of IP packets with the Authentication Header.

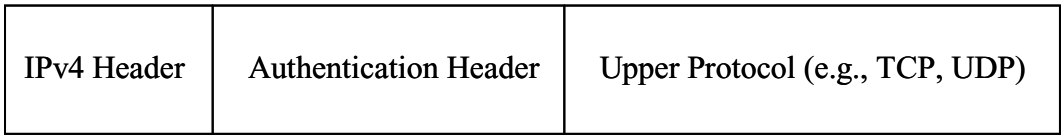


Figure D.1 IPv4 Example

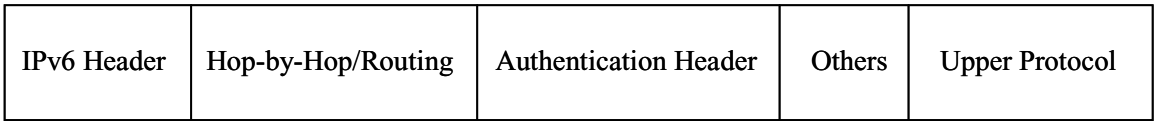


Figure D.2 IPv6 Example

The format of the IP Authentication Header is shown in Figure D.3.

The Next Header field is used to identify the protocol or header which follows the *Authentication Header*. The Next Header can be an IPv6 extension header, an upper-layer header, such as TCP or UDP, or IPv6 or IPv4 itself in the case of a tunnel. The Length field specifies the size of the Authentication Data field measured in 32-bit words. The Security Parameters Index field tells the receiver how to interpret and, therefore, how to verify the message digest or digital signature present in the Authentication Data field i.e., the Security parameter Index tells the receiver which

of the many possible security associations was used to compute the value in the Authentication Data field.

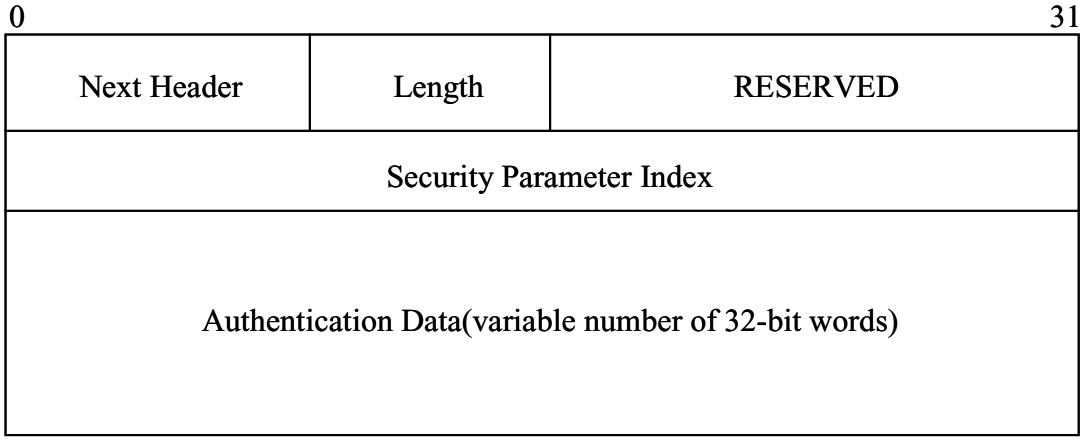


Figure D.3 IP Authentication Header

10.2. IP Encapsulating Security Payload (ESP)

The Encapsulating Security Payload is a mechanism for providing integrity and confidentiality to IP packets. It may also provide authentication, depending on which algorithm is being used. It is used similar to the *Authentication Header*, being placed between the IP and upper-layer header or between two IP headers in case of a tunnel.

The ESP packet-format consists of a 32-bit Security Parameter Index field followed by a bunch of encrypted information that is specific to the encryption algorithm being employed. It is much more useful to look at a specific example of the *Encapsulating Security Payload*. One such example is shown in Figure D.4, as defined for the Triple Data Encryption Standard (Triple-DES) encryption algorithm. The ESP syntax itself is shown in Figure D.5 and D.6.

In Figure D.4, the shaded fields are encrypted before being transmitted, while the unshaded fields are sent as plain text. The Security parameters Index field provides identical functionality as in the case of the *Authentication Header.* A randomly chosen Initialization Vector is common in many encryption algorithms and is used to ensure that identical plaintext messages produce different encrypted, ciphertext messages.

Following these two plain text (unencrypted) fields is the encrypted payload. The encrypted payload consists of the Upper-Layer Header and User Data followed by some padding and the Next Header field. The Next Header field identifies the protocol or header contained in the Upper-Layer Header and User Data fields.

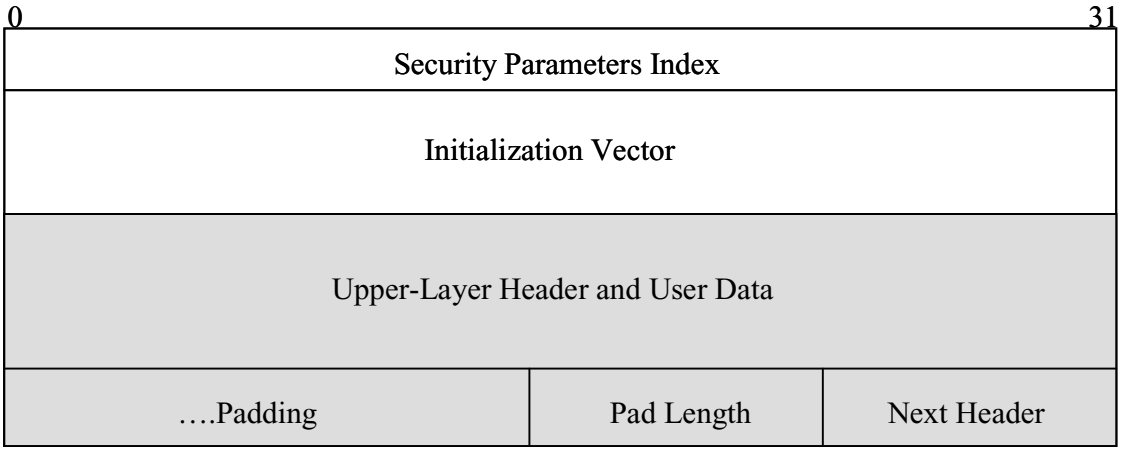


Figure D.4: IP Encapsulating Security Payload (Triple-DES)

Random data can be placed in the Padding field to make it very difficult for a "Bad Guy" to guess the length of the "real" data that is being protected by encryption. This is useful for small messages that are common in many remote login programs, some of which are only a few bytes in length. The Pad Length field tells the receiver how many Padding bytes were added to the original payload.

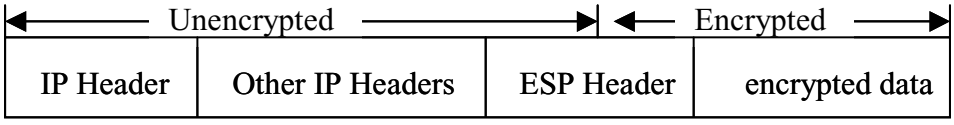


Figure D.5: High level diagram of a secure IP Packet

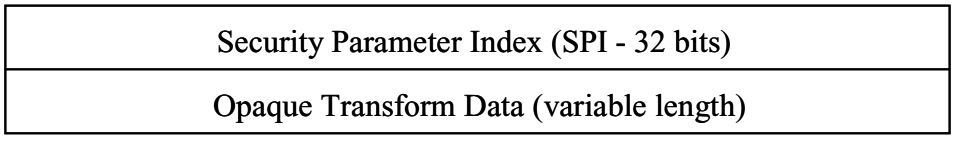


Figure D.6: ESP Header Syntax

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

11. APPENDIX E – QUALITY OF SERVICE

QoS is a set of service requirements to be met by the network in transporting a "flow". A flow is a sequence of packets sent from a particular source to a particular destination (or a group of destination nodes) for which the source desires special handling by the intervening routers. The primary goal of IP QoS is to provide priority including dedicated bandwidth, controlled jitter and latency, and improved loss characteristics. QoS provides end-to-end service guarantees and policy-based control of an IP network's performance measures, such as resource allocation, switching, routing, packet scheduling, and packet drop mechanisms.

IP Quality of Service is intended to deliver guaranteed as well as differentiated Internet services on the Internet or any IP-based network. Guaranteed and differentiated services provide different levels of QoS, and each represents an architectural model for delivering QoS. The IntServ architecture has been defined to deliver guaranteed service, while the DiffServ architecture provides a framework within which service providers can offer customers a range of network services, each differentiated based on performance.

11.1. IntServ Architecture

The Internet Engineering Task Force (IETF) set up the IntServ model to better meet the needs of emerging and diverse real time applications. It aimed to clearly define the new enhanced Internet service model as well as to provide the means for applications to express end-to-end resource requirements with support mechanisms in routers and subnet technologies. Two services – guaranteed, and controlled load are defined under this model. *Guaranteed service* provides deterministic delay guarantees, whereas *controlled load service* provides a network service close to that provided by a best-effort network under tightly loaded conditions.

Resource Reservation Protocol (RSVP) is suggested as the signaling protocol that delivers end-to-end service requirements. RSVP is used to signal QoS information using control messages that are different from the actual data packets. RSVP signaling results in certain resource guarantees along the traffic's routed path.

The IntServ model requires per-flow guaranteed QoS on the Internet. With thousands of flows existing on the Internet today, the amount of state information required in the routers can be enormous. This can create scaling problems, as the state information increases as the number of flows increases. This makes IntServ hard to deploy on the Internet.

Later in 1998, the DiffServ model was formed under the IETF to serve as a bridge between IntServ's guaranteed QoS requirements and the best-effort service offered by the Internet. DiffServ provides traffic differentiation by classifying traffic into a few classes, with relative service priority among the traffic classes.

11.2. DiffServ Architecture

The DiffServ approach to providing QoS in networks employs a well-defined set of building blocks from which a variety of services can be built. Its aim is to define the differentiated service (DS) byte and the Type of Service (ToS) byte from the IPv4 header, and the Traffic Class byte from IPv6, and mark the standardized DS byte of the packet such that it receives a particular forwarding treatment or per-hop behavior (PHB), at each network node.

The DiffServ architecture provides a framework within which service providers can offer customers a range of network services, each differentiated based on performance. A customer can choose the performance level needed on a packet-by-packet basis by simply marking the packet's Differentiated Services Code Point (DSCP) field to a specific value. This value specifies the PHB given to the packet within the service provider network. Typically, the service provider and customer negotiate a profile describing the rate at which traffic can be submitted at each service level. Packets submitted in excess of the agreed profile might not be allotted the requested service level.

The DiffServ architecture only specifies the basic mechanisms on ways to treat packets. A variety of services can be built by using these mechanisms as building blocks. A service defines some significant characteristic of packet transmission, such as throughput, delay, jitter, and packet loss in one direction along a path in a network. After a service is defined, a PHB is specified on all the network nodes of the network offering this service, and a DSCP is assigned to the PHB. A PHB is an externally observable forwarding behavior given by a network node to all packets carrying a specific DSCP value. The traffic requiring a specific service level carries the associated DSCP field in its packets.

All nodes in the DiffServ domain observe the PHB based on the DSCP field in the packet. In addition, the network nodes on the DiffServ domain's boundary carry the important function of conditioning the traffic entering the domain. Traffic conditioning involves functions such as packet classification and traffic policing and is typically carried out on the input interface of the traffic arriving into the domain. Traffic conditioning plays a crucial role in engineering traffic carried within a DiffServ domain, such that the network can observe the PHB for all its traffic entering the domain. Figure E.1 illustrates the DiffServ architecture. The two major functional blocks in this architecture are shown in Table E.1

Table E.1 DiffServ Functional Blocks

Functional Blocks	***Location***	***Enabling Functions***	***Action***
Traffic Conditioners	Typically, on the input interface on the DiffServ domain boundary router	Packet Classification, Traffic Shaping, and Policing	Polices incoming traffic and sets the DSCP field based on the traffic profile
Per-hop behavior	All routers in the entire DiffServ domain	Resource Allocation, Packet Drop Policy	PHB applied to packets based on service characteristic

			defined by DSCP

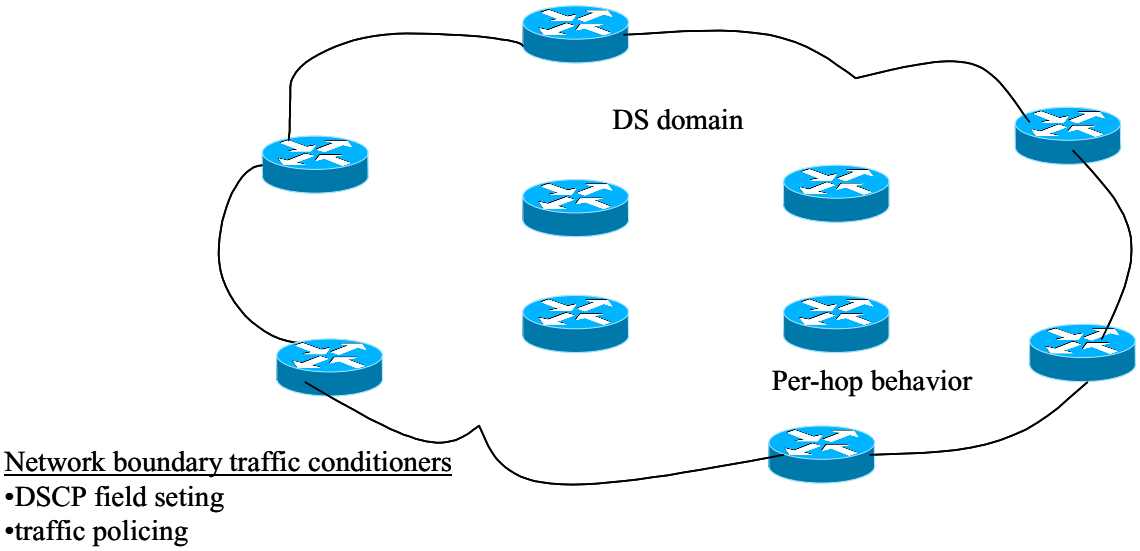


Figure E.1 DiffServ Architecture

ATTACHMENT 8-1

ARINC 664 AIRCRAFT DATA NETWORK
PART 8 UPPER LAYER and USER SERVICES
WORKING PAPER V4.0

12. APPENDIX F - ACRONYMS

ADM	Administration
AEEC	Avionics Electronics Engineering Committee
AFI	Authority Format Identifier
AH	Authentication Header
AINSC	Aeronautical Industry Service Communication
AOC	Airline Operational Communications
ARINC	Aeronautical Radio, Inc.
ARS	Administrative Regional Selector
ATM	Air Traffic Management
ATN	Aeronautical Telecommunication Network
ATS	Air Traffic Service
ATSC	Air Traffic Services Communications
CLNP	Connectionless Network Protocol
CLTP	Connectionless Transport Protocol
CLTS	Connectionless Transport Service
CM	Context Management
CNS	Communications, Navigation, and Surveillance
COPP	Connection Oriented Presentation Protocol
COSP	Connection Oriented Session Protocol
COTS	Commercial-Off-The-Shelf
CPDLC	Controller Pilot Data Link Communication
ES	End System
ESP	Encapsulating Security Payload
FAA	Federal Aviation Administration
FANS	Future Air Navigation System
FMC	Flight Management Computer
FMS	Flight Management System
GA	General Aviation
GRC	Glenn Research Center
ICAO	International Civil Aviation Organization
ICMP	Internet Control Message Protocol
IDI	Initial Domain Identifier
IDP	Initial Domain Part
IDRP	Inter-Domain Routing Protocol
IETF	Internet Engineering Task Force
IFR	Instrument Flight Rules
IP	Internet Protocol
IPng	Internet Protocol Next Generation
IPsec	Internet Protocol Security

ATTACHMENT 8-1

IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IS	Intermediate System
ISO	International Organization for Standardization
LOC	Location
NAS	National Airspace System
NASA	National Aviation and Space Administration
NSAP	Network Service Access Point
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
QoS	Quality of Service
RD	Routing Domain
RDF	Routing Domain Format
RDI	Routing-Domain Identifier
RFC	Request for Comments
RSA	Ron Rivest, Adi Shamir and Leonard Adleman
RSVP	Resource Reservation Protocol
SARPs	Standard and Recommended Practices
TCP	Transmission Control Protocol
ToS	Type of Service
TSAP	Transport Service Access Point
TSDU	Transport Service Data Unit
TSEL	Transport Selector
UDP	User Datagram Protocol
ULCS	Upper Layer Communications Services
VDL M1	VHF Data Link Mode 1
VDL M2	VHF Data Link Mode 2
VDL M3	VHF Data Link Mode 3
VDL M4	VHF Data Link Mode 4
VDL	VHF Data Link
VER	Version
VHF	Very high frequency
VoIP	Voice over Internet Protocol