INTERNATIONAL STANDARD

ISO 22902-1

First edition
2006-11-01

Road vehicles — Automotive multimedia interface —

Part 1: General technical overview

Véhicules routiers — Interface multimédia pour l'automobile

Partie 1: Vue d'ensemble technique générale

Reference number
ISO 22902-1:2006(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 22902-1 was prepared by Technical Committee ISO/TC 22, *Road vehicles*, Subcommittee SC 3, *Electrical and electronic equipment*.

ISO 22902 consists of the following parts, under the general title *Road vehicles — Automotive multimedia interface*:

— *Part 1: General technical overview*

— *Part 2: Use cases*

— *Part 3: System requirements*

— *Part 4: Network protocol requirements for vehicle interface access*

— *Part 5: Common message set*

— *Part 6: Vehicle interface requirements*

— *Part 7: Physical specification*

Road vehicles — Automotive multimedia interface —

Part 1:
General technical overview

1 Scope

This multimedia and telematics standard is based on the AMI-C specifications and reference documents for automotive industry.

It is established to facilitate the development, promotion and standardisation of automotive information and entertainment system interfaces to motor vehicle communication networks.

Technical glossary is a compilation of terms and definitions used in AMI-C technical publications.

User guide describes the organisation and classification and scope of the reference information for ISO specifications listed below.

Architectural overview describes the structural, functional, and applications views of the AMI-C architecture.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 22902-2, *Road vehicles — Automotive multimedia interface — Part 2: Use cases*

ISO 22902-3, *Road vehicles — Automotive multimedia interface — Part 3: System requirements*

ISO 22902-4, *Road vehicles — Automotive multimedia interface — Part 4: Network protocol requirements for vehicle interface access*

ISO 22902-5, *Road vehicles — Automotive multimedia interface — Part 5: Common message set*

ISO 22902-6, *Road vehicles — Automotive multimedia interface — Part 6: Vehicle interface requirements*

ISO 22902-7, *Road vehicles — Automotive multimedia interface — Part 7: Physical specification*

3 Terms and definitions

For the purposes of all parts of ISO 22902, the following terms and definitions apply.

3.1 Terms and definitions

3.1.1
1394 Automotive
the automotive version of IEEE 1394 network technology

3.1.2
AMI-C component
component that meets AMI-C specification(s)

3.1.3
AMI-C guideline
AMI-C statement of what should be done on a subject for which AMI-C adopts a standard or specification from another organization

3.1.4
AMI-C recommendation
AMI-C guideline that AMI-C proposes to another organization for adoption

3.1.5
AMI-C requirement
requirement of one or more interfaces for one or more elements in the architecture; requirements do not address performance and quality issues

3.1.6
AMI-C specification
automotive multimedia interface specification published for connecting or implementing information, communication, entertainment systems to motor vehicles

3.1.7
Application
distinct executable software element that provides functionality or utility to a user

3.1.8
architecture
description of the key elements of a system, their functions, and the interrelationships among the elements

3.1.9
ASN.1 Reference
the AMI-C CMS has been developed in ASN.1. In this textual document the messages are presented in a tabular format for readability. The CMS in ASN.1 is available as a download from the AMI-C web site, although ASN.1 has a large number of type definitions, only the following definitions are used in the AMI-C CMS

3.1.10
audio Gateway
[Bluetooth] device embedded with the Bluetooth transceiver that serves as the gateway (access) to the phone audio for both input and output; typically, devices acting as audio gateways are cellular or mobile phones these devices are implemented with the Hands-Free Profile

3.1.11
boolean
logical values TRUE and FALSE

3.1.12
bundle
[software] mechanism to organize the delivery of software to the platform/OSGi framework

3.1.13
common message set
CMS
sequence of network messages between two functional modules that enable the exchange of data

3.1.14
Car Communication Application Promotion
CCAP
Bluetooth Hands-Free phone application guideline for implementers to improve device interoperability and enhance customer convenience by providing consistent user experience

3.1.15
choice
between types

3.1.16
common mode bulk current injection
CBCI
test method used to determine a component's or subsystem's immunity to electromagnetic fields, "common mode" refers to the phase at which the signal is applied

3.1.17
commonality
the possession, along with one or more other systems, of a certain attribute or set of attributes: the characteristic of being a shared feature or attribute

3.1.18
component
a device that is one of the individual parts of which a vehicle is made up; especially a part that can be separated from or attached to a system

3.1.19
configure-ability
characteristic of a system that supports the rearrangement of features and attributes to support different features and uses

3.1.20
consumer connector
connector intended for use in applications where the consumer regularly mates and un-mates the connector for the purpose of installing and removing components or devices

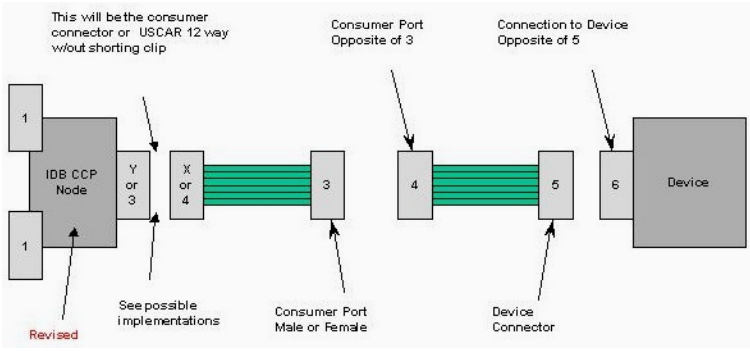

Figure 1 — Consumer device connection — General configuration

3.1.21
consumer convenience port
CCP
connection point for portable consumer devices to connect to an AMI-C network

3.1.22
controllers
computer-based units performing a dedicated function or feature

NOTE They frequently use network devices to gather information to control a system.

3.1.23
custom access adaptor
interface device between an AMI-C interface connector and an embedded device to a customer access connector

3.1.24
customer access connector
the connector accessible to a driver or passenger for the connection of an external or consumer device to an AMI-C compliant network system

NOTE It provides access to the system signal set (power, network data, power mode, etc.).

3.1.25
device
low-level components with little or no programmable intelligence

NOTE They are generally driven by other components and customer commands.

3.1.26
differential Mode Bulk Current Injection
DBCI
test method used to determine a component's or subsystem's immunity to electromagnetic fields

NOTE Differential mode refers to the phase at which the signal is applied.

3.1.27
embedded
components or devices that are installed by the automaker

3.1.28
embedded component
component installed in a vehicle by the vehicle manufacturer or service technician using conventional wiring and mounting technologies

3.1.29
embedded connector
separable interface that is an integral part of an embedded component or device

3.1.30
execution environment
(see software execution environment)

3.1.31
expandability
characteristic of a system that supports and/or facilitates the addition of new features and capabilities

3.1.32
functional module
FM
an abstraction within the AMI-C network architecture that is used to address a device

EXAMPLE Audio-related functions such as accessing vehicle audio resources, changing volume, etc., are grouped in an audio functional module.

3.1.33
gateway
interface between two networks carrying different protocols

3.1.34
Hands-Free Car Kit
HFCK
[Bluetooth] car kit composed of two major components:

— Hands-Free Unit (HFU): Device that the customer can use without need to manipulate by hands, the most well-known is hand free phone;

— User Interface (UI): Appropriate interface to facilitate complete phone operations from the hands-free unit.

3.1.35
Hands-Free Profile
HFP
[Bluetooth] specification defining the minimum set of functions required to use a mobile phone together with a hands-free device in a car via a Bluetooth link

3.1.36
Hands-Free Unit
HFU
[Bluetooth] device embedded with the Bluetooth transceiver and implemented with the Bluetooth Hands-Free Profile

NOTE The device serves as a remote unit to the audio gateway. It provides audio input, output and control functions of the audio gateway over the Bluetooth link.

3.1.37
host
general-purpose computers having the capability to execute a variety of application software

NOTE Hosts easily allow a user to select and change software on demand.

3.1.38
implementation
physical realization of a specification or specifications

3.1.39
instance number
I-Num
a number statically or dynamically assigned to each functional module that shares the same function type

3.1.40
interchangeability
characteristic of a system that allows hardware and/or software components to be replaced by other components with no perceptible change in system function or performance

3.1.41
integer
whole numbers (positive or negative), possibly named

3.1.42
interoperability
characteristic of providing an intended function in coordination with other components, the characteristic of sharing information with other system functions or components to provide additional functionality

3.1.43
isochronous data
data that arrive at the destination at the same rate that they leave the source; the word “isochronous” means “of equal time”; isochronous data are used for streaming video and audio data

3.1.44
L2CAP
(see Logical Link Control and Adaptation Protocol)

3.1.45
link key
[Bluetooth] authentication key used to establish a link between devices

3.1.46
listener
[software] Java object that responds to external events such as a change in the host operating state OR an application with a well-defined interface that does something useful for another application of a user

3.1.47
local functional modules
functional modules that are within the same AMI-C component

3.1.48
logical address
address used to identify a functional module and is made up of the combination of F-Type and I-Num

3.1.49
Logical Link Control and Adaptation Protocol
L2CAP
protocol supporting higher level protocol multiplexing, packet segmentation and reassembly, and the conveying of quality of service information

3.1.50
object body
has, Message Type (Msg Type), Message Class, Object Type and Operand

3.1.51
object type
an identification of object in a functional module

EXAMPLE In the case of body module, mirror, window, seat and so on.

3.1.52
octet string
byte strings

3.1.53
enumerated
enumeration of identifiers (state of a machine, for instance)

3.1.54
operand
data in a message

3.1.54.1
message type
there are six kinds of message type for application messages; INQUIRE, REPORT, SET, CONFIRM, COMMAND, and WARNING

3.1.54.2
message class
the message class shows a class within which a message is comprised, the message class defines thirteen types:

- Management – Network device management, audio/video stream management, and service discovery.
- Core – Information that are originally inherent in a vehicle (VIN, static configuration information, etc.).
- Body Module – Control and status related with body module (window, seat, mirror, light, trip meter, vehicle speed, etc.).
- Powertrain – Status related with powertrain (Oil temperature, coolant temperature, gear, etc.).
- Vehicle Diagnostics – Message for vehicle diagnostic (ISO 15031-5 emission related, ISO 14229-1 non emission related).
- Amplifier – Control related with amplifier and codec (volume, fade/balance, codec, etc.).
- General Player – Common functions for disk and tape player.
- Disk Media – Audio/video player for disk media (CD, MD, DVD, MP3, etc.).
- Tape Media – Audio/video player for tape media (audio tape, VCR, etc.).
- Tuner – Audio/video broadcast tuner (AM, FM, XM, TV, etc.).
- General Phone – Basic phone functionality (dial, hang-up, phone book, etc.).
- Advanced Phone – Advanced phone functionality (short message, multi party call, conference call, etc.).
- Text Display – Command to display simple texts and input texts.

3.1.55
mode
the power level of a system, modes include ON, OFF, and SLEEP; a system is in a mode, a device or a component is in a state

3.1.56
modularity
exhibiting separable functions embodied within discrete and removable hardware or software components or modules

3.1.57
MSC
Modem Status Command

3.1.58
network adaptation layer
network layer that adapts the messages from the functional modules to those of the network specific layer and resolves addressing issues (optional use of a registry)

3.1.59
network transport layer
network layer that puts network transactions in packets according to the network transport protocols of the specific network

3.1.60
network transaction
sequence of network messages between two functional modules enabling the exchange of data

3.1.61
network transaction
CBCI
See common mode bulk current injection

3.1.62
NULL
Includes the single value NULL, used for delivery report or some alternatives of the CHOICE type (particularly for the recursive types)

3.1.63
operability
the capability of being used or operated effectively to provide an intended result or function

3.1.64
pairing
[Bluetooth] the creation and exchange of a link key between two devices; the devices use the link key for future authentication when exchanging information

3.1.65
PMODE line
signal wire used to control and sense wake-up and sleep states in an AMI-C system

3.1.66
PMODE signal
signal carried by the PMODE line or optionally a signal on the AMI-C network identifying the power mode of the network

3.1.67
portability
characteristic of a component or a solution that allows it to be used effectively in multiple locations or application venues

3.1.68
preference
default value preferred by a user, such as a radio preset value

3.1.69
preference service
service that returns the preferences for a given vehicle occupant

3.1.70
proof-of-concept implementation
implementation to validate requirements and/or specifications

3.1.71
radio frequency communication
[Bluetooth] the radio frequency oriented emulation of an RS232 serial communication over an L2CAP channel

3.1.72
reference implementation
fully functional implementation against which other implementations can be evaluated

3.1.73
registry
address-mapping table between the logical address of a functional module and the specific network address of a component that contains this functional module

3.1.74
remote functional modules
functional modules that are in separate AMI-C components

3.1.75
requirement
statement of something that shall be followed for a particular purpose; an essential condition, it may not have technical details for implementation

3.1.76
resource manager
a functional module that contains the ability to perform the following functions:

— Provide dynamic I-Nums to the (FMs) that request them when a new component joins the network.

— Deallocate the I-Nums of the FMs that belong to a component that unplugs from the network.

— Allocate the multicast group ID when requested.

3.1.77
run level
the execution state of a multimedia system in a vehicle

3.1.78
run time environment
the total set of software running on a computing platform that supports the execution of applications/programs on that platform

3.1.79
scooping
undesirable condition in which the female connector housing assembly makes contact with the male terminals during the improper insertion of the female connector housing assembly into the male connector housing assembly

3.1.80
security
the characteristic of being free of risk or danger

3.1.81
sequence
ordered structure of values of (generally) different types

3.1.82
sequence of
ordered structure of values of the same type

3.1.83
service
software that provides a useful function to another entity

3.1.84
Service Level Connection
SLC
[Bluetooth] the synchronized high-level connection that refers to the presence of a RFCOMM connection, and assumes that the Hands-Free Unit has synchronized itself to the state of the AG using the specified Service Level Connection initialization procedure

3.1.85
software execution environment
the set of Java classes that are present on the platform where Java program runs

3.1.86
specification
detailed, exact statement of required particulars, especially a statement prescribing the construction, operating environment, functions, performance, or quality of a product

3.1.87
specific network
one of the networks either endorsed by AMI-C or determined to be a candidate for later endorsement by AMI-C

3.1.88
start level
OSGi defined state that constrains the order in which services are started when the system boots

3.1.89
state
the power and activity level of an AMI-C device, these include ON, OFF, and SLEEP; contrast with mode, system is in a mode and device is in a state

3.1.90
subscription
subscription is a service that allows a requestor to register for updates to a parameter without explicitly requesting each update.

NOTE There are two types of Subscription services, Periodic and Notification. Periodic Subscription is one where the basis for a message being sent is the expiration of a timer, and a Notification Subscription is one where the basis for the transmission of the message is that the parameter contained in the message has changed.

3.1.91
Synchronous Connection Oriented
SCO
one of the two Bluetooth data link types; a synchronous (circuit-switched) connection is used for full duplex audio communication between two devices

3.1.92
system transaction
data transaction performed by a component control module and characterized by its Sys bit field (in the AMI-C header) set to ‘1’B; it is always broadcasted

3.1.93
testability
the ease of being tested for proper function(s) against some predefined criteria

3.1.94
upgradeability
characteristic of a system that allows the replacement of existing hardware and/or software components or modules for improved or extended performance

3.1.95
vehicle interface
interface between the system and the vehicle manufacturer's proprietary systems, information, and signals

3.1.96
vehicle services interface
proper subset of a vehicle interface that provides access to services commonly found on body and/or chassis networks

3.2 Abbreviations

'0101'B

binary string

(1..255)

1 to 255

(1 | 255)

1 or 255; "…": new objects may dynamically be added

3.2.1
AG
see Audio Gateway

3.2.2
API
application programming interface

3.2.3
ASP
application service provider

3.2.4
CCAP
see Car Communication Application Promotion

3.2.5
CCP
see consumer convenience port

3.2.6
CISPR
Comité Internationale Spécial des Perturbations Radioelectrotechnique (IEC committee)

3.2.7
'EF35'H
hexadecimal string

3.2.8
HFCK
see Hands-Free Car Kit

3.2.9
HFP
see Hands-Free Profiles

3.2.10
HFU
See Hands-Free Unit

3.2.11
HMI
human machine interface

3.2.12
MD
mini-disk

3.2.13
OCTET STRING [SIZE(8)]
eight-bytes string

3.2.14
OEM
original equipment manufacturer. In the automotive industry, OEM represents vehicle manufacturers

3.2.15
PAP
see Phone Access Profile

3.2.16
POF
Plastic Optic Fiber

3.2.17
PN
[Bluetooth] Parameter Negotiation

3.2.18
RFCOMM
Radio Frequency Communication

3.2.19
RPN
[Bluetooth] Remote Port Negotiation

3.2.20
SABM
[Bluetooth] Set Asynchronous Balanced Mode

3.2.21
SCO
see Synchronous Connection Oriented

3.2.22
UI
user interface

3.2.23
SLC
see Service Level Connection

3.2.24
VR
voice recognition

3.2.25
VSP
vehicle service provider

4 Introduction to the User Guide

To bring existing open technologies into automotive environment, the subject ISO specifications are provided: ISO 22902-2, ISO 22902-3, ISO 22902-4, ISO 22902-5, ISO 22902-6.

The specifications can be used directly, and have adapted other available specifications by modifying them or supplementing them with additional technical contributions to make the technologies compatible with the vehicle environment. The adopted, adapted or created specifications include required technical elements that were not available in the existing body of work.

The following connections and relationships among the ISO specifications are worth noting:

1) Vehicle interface, network and host are three architectural components of AMI-C architecture.

2) Among them, the vehicle interface (ISO 22902-6 Vehicle interface requirements) is a required component.

3) *AMI-C physical specification* is applicable to all implementations of ISO 22902-4 Network protocol boundary requirements and ISO 22902-6 Vehicle interface requirements.

4) ISO 22902-2 Use cases is an informational document used to help non-technical readers understand the context of the specifications developed. The use cases examined are neither an exhaustive or all-inclusive set, but rather a common set of examples for illustrative purposes.

ISO 22902-3 is a general technical reference, and is not a detailed requirements document. The purpose of this information is to identify the general system characteristics implicit in the architecture provided by the other specifications.

5 General architecture

The AMI-C architecture is the framework upon which the AMI-C system is formulated and configured. This framework defines how the various components of an AMI-C system are interconnected, how the system logically operates, and how the system meets the AMI-C system requirements.

5.1 Architectural views

An architectural view is a representation of the AMI-C system from the perspective of a related set of concerns. The views used in this document are structural, functional and application views. Each of these architectural views responds to different types of inquiries about the architectural description. The views and related inquiry types are as follows:

— Structural view addresses logical elements and how they inter-relate.

— Functional view addresses what the system does and what the basic behavioral aspects are.

— Application view addresses how the architecture enables applications and implementations to access services.

5.2 Structural view

The logical elements of the architecture are:

— Vehicle interface – An interface to other vehicle systems, information, and signals.

— Network(s) – Standard network and message set to provide data interchange and control between components.

- Network protocols and services.
- Network nodes: devices, controllers and gateways.

— Host – Computer used to execute cross-platform, downloadable application code.

- System services.
- Applications.

An actual implementation of the architecture shall have a vehicle interface, and may have one or more networks and/or one or more hosts.

5.2.1 Vehicle interface

The Vehicle Interface (VI) is the only required component of the architecture. The VI is the interface between the manufacturer's proprietary systems and the system. While the VI performs functions traditionally ascribed to a gateway, it does so in a way that does not reveal any of the protocols, addressing mechanisms, physical devices or functional partitioning of the manufacturer's proprietary systems.

The VI creates for devices a standard interface to information and services from the manufacturer specific portions of the vehicle. To other devices on the network, the VI appears as a black box. This approach enables the VI to provide a measure of security to the proprietary side of the vehicle. Since the services of the vehicle are exposed through the VI's standard interface, devices never have direct communication with manufacturer specific devices. The VI receives requests for information or services, translates them into appropriate messages and/or protocols, and then forwards the requests to the proprietary network. Additionally, the VI provides information regarding the required power status of the network. This includes the control of specific hardware to control the power moding of networks and devices, if required by individual networks. The VI functions are also made available to applications running on hosts through an appropriate API.

Vehicle services provide devices and applications with three main mechanisms for retrieving information or performing command functions:

- Request/command: The request/command mechanism consists of a single command to a service target, or request for information from a device on the proprietary side of the vehicle. The command may be answered by a confirmation message. The request for information must always be answered with the requested data.
- Subscription: The subscription mechanism consists of a single request for subscription to a piece of vehicle interface information, followed by periodic updates of the information.

Notification of the change of a particular piece of vehicle interface information, followed by messages generated by the vehicle services' detection of a change in the parameter.

An individual service always supports the request/command mechanism, and may support either of the subscription or notification mechanisms.

5.2.2 Network

It Supports a set of wired and/or wireless networks that meet a set of defined requirements. The network system requirements are found in the system requirements. Based on these requirements, it adopts existing network specifications that meet the requirements, extends existing specifications with developed specification statements, or develops a specific specification where none exists.

An network transmits both control data and content data, between the devices on the network.

A network consists of two types of elements: devices and a protocol.

A gateway connects two networks, while the vehicle interface connects one or more networks to a manufacturer's proprietary services, some or all of which may be provided across the manufacturer's proprietary network. A network may have one or more gateways to other networks, but the resulting inter-network topology shall not contain a loop.

The system may be upgraded by the addition of new gateways, networks, devices and hosts. The standard system must support upgrades through implementation of a discovery protocol on each local network that supports the addition of this standard devices to the network up to its capacity.

Communication on this standard network

The Figure 2 shows one of the typical applications of a system in a vehicle from the viewpoint of the network architecture. The diagram depicts two major domains:

— The domain of this standard is completely open to any devices connected with one or more networks;

— The automaker's domain: The automaker's domain is proprietary to a vehicle manufacturer's specific design and controls all the services available to the domain (e.g. amplifier and speaker, or vehicle-related information like vehicle speed). Access to these services is managed through an AMI-C vehicle interface.

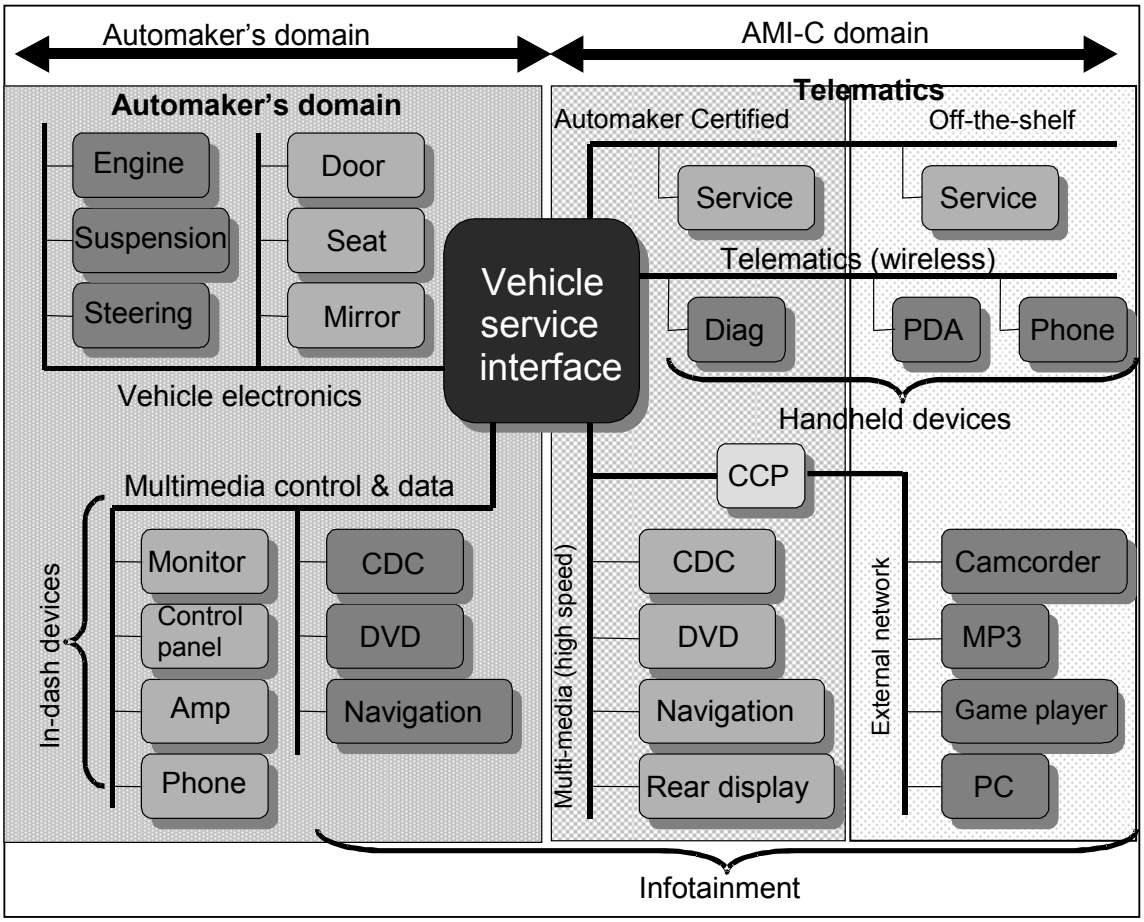

Figure 2 — Domains in network configuration

A functional module is an abstraction within the network architecture that represents a group of related functions associated with a component. As an example, a network component could be a digital television receiver that would implement several functions including tuner, decoder, audio amplifier, etc., some of which could advertise themselves as a functional module. There can be multiple instances of same functional modules in one vehicle.

See Figure 3 for a functional module, which can also access and use resources in the automaker's domain, such as powertrain functions, body functions, diagnostics, etc. Although figure depicts the audio and display resources in the automaker's domain, they can also be in the AMI-C domain.

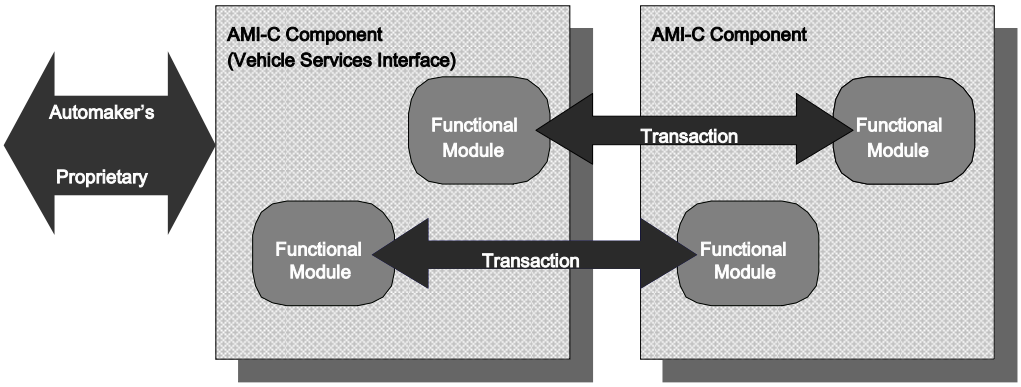

Figure 3 — Logical communication between functional modules

There are two specific functional modules that must be implemented by each vehicle, one on the network and the other in the vehicle interface. For a function module on the network, there must be a corresponding function module in the vehicle interface in order to discover the configuration of the vehicle and its functional modules.

A functional module type and instance number identify each functional module. In order to identify each functional module uniquely, an instance number is assigned to it. This identification may be used directly in some messages or it may be converted to another format. The network adaptation layer is the software that handles this conversion if required.

Whereas the Network Communication Model commonizes the addressing formats, the Common Message Set provides for common messages among the various networks. Any network must support the vehicle-related elements of the Common Message Set that will be used by the attached devices or gateways. A gateway is required between two heterogeneous networks. The gateway may run on the same platform as a host, but is logically distinct from it. The architecture permits multiple gateways in various configurations. Vehicle manufacturers and after-market suppliers may choose to integrate several logical gateways into a single physical package.

Network power management

All networks shall support three distinct power modes:

— ON: supports all device functions on the network.

— OFF: implies that all device functions have been shut down.

— SLEEP: results in low power consumption by all devices, with no data communications.

Transitions between power modes can be carried out by either messages on the bus or a hardwired signal. The message or hardwired signal notifies components to change to a specific state, based on requests initiated from sources such as the ignition switch, an active system timer, external signal, or the waking of other modules. The detailed behavior of those components is network and implementation dependent.

5.2.3 Host

AMI-C defines a host as a computing platform that can execute software other than embedded software that was provided with the device. An Host is a host that contains a specific software architecture that enables it to

execute generic application software. The software architecture provides a common execution environment that allows applications to be written to run on the host platforms of any vehicle.

The software architecture is based on the capability of Java to provide a platform independent program execution environment. The software interfaces are defined in Java. Any host must have a Java virtual machine and implement the AMI-C software interfaces. In addition to using Java, this standard has chosen to specify the OSGi framework as part of the execution environment, so an AMI-C host must provide an implementation of this framework. AMI-C designates a set of services, each of which has an interface to applications. The collection of these interfaces constitutes the AMI-C application programming interfaces or APIs. Applications can use these interfaces to access services running on the host in a consistent, platform-independent manner.

Figure 4 shows an example of an host connected to an network.

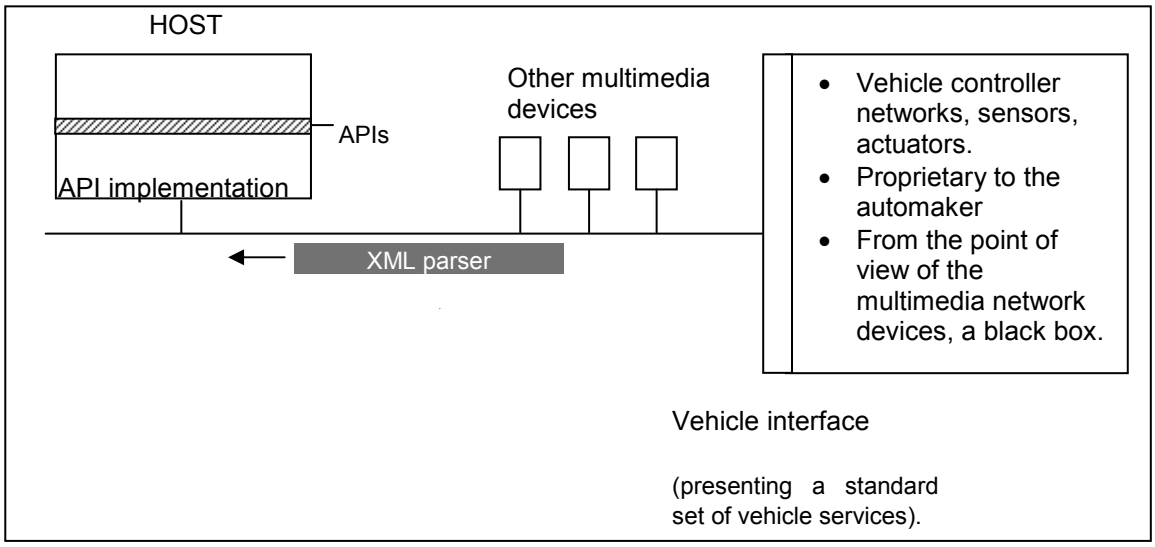

Figure 4 — Example of an host connected to an network

There are two points worth noting. First, the implementations of the APIs are specific to the network(s) connected to the host. The APIs themselves are network independent except for those functions dealing with the network, such as network management. Second, some aspects of the host platform, such as the processor and operating system used, are not specified by this standard.

Figure 5 shows the host software structure. The basic platform consists of an operating system and low level drivers for the network(s) to which the host is attached. At the next level there is a Java runtime environment as well as application level network drivers for the Common Message Set. The Java virtual machine supports platform independence for the software above it. The choice of operating system and Java implementation is not mandated in this document.

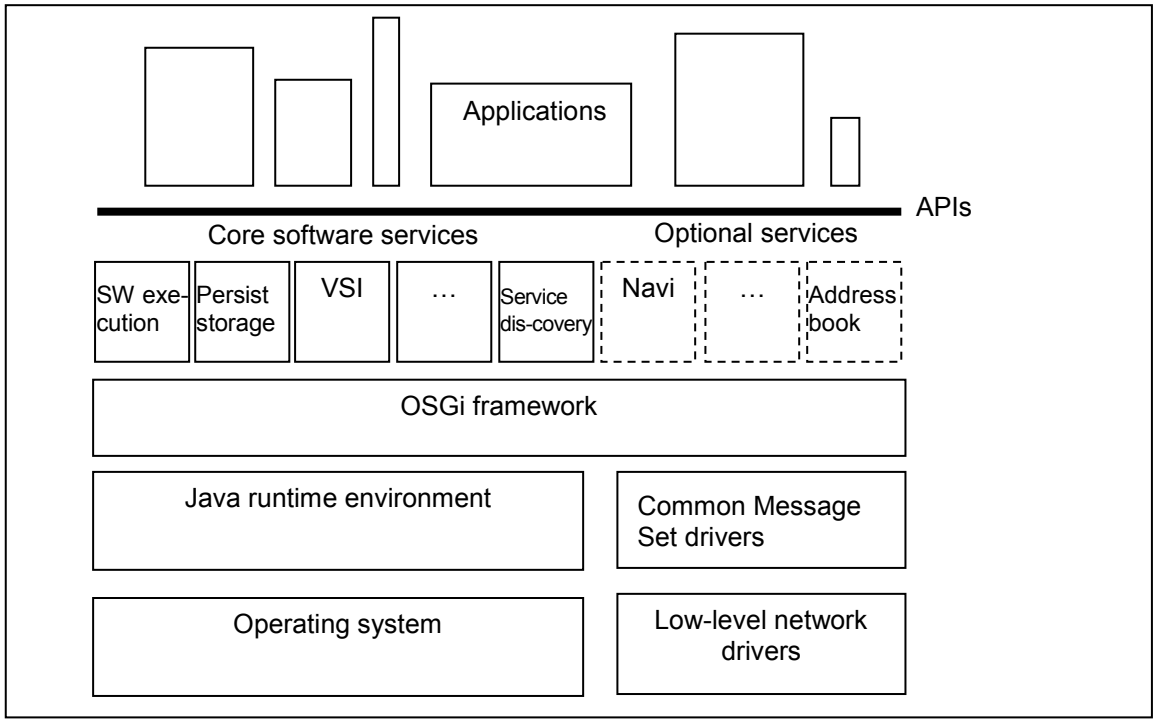

Figure 5 — Host software structure

The host platform must have an implementation of the framework defined by the Open Services Gateway Initiative (OSGi). The framework provides an execution environment for services that provides lifecycle management, storage management, security and service discovery. The set of services managed by the OSGi framework includes a set of core services that must be present on each AMI-C host and a set of optional services. It may also include additional services not defined by this standard. The core services cover the areas of software execution, software lifecycle, security, persistent storage, internationalization, vehicle service interface, resource management, service discovery, communication and human machine interface (HMI). Some of these areas, such as software lifecycle, are covered by OSGi defined service interfaces. Some, such as vehicle services, are defined by this standard, and others are covered by a combination of OSGi defined services and this standard defined services. The optional services include user profile services, address book service and off-board navigation services.

One of the major functions of OSGi specification is to define a standard format for packaging services and applications. This allows new applications/services to be added in a consistent manner across all platforms. The standard OSGi package for a service is called a bundle. A bundle has two aspects. First, it is a Java archive (jar) file that contains the code and data to implement a set of services. It may include such items as Java class files, text/HTML/XML files, and graphics files. It also contains a manifest file that contains information needed to install the bundle, as well as information about dependencies on other resources. Second, a bundle is a functional module that may exist in different states during its lifetime. Figure 66, which is taken from the OSGi specification [OSGI2001], shows the bundle states and their transitions.

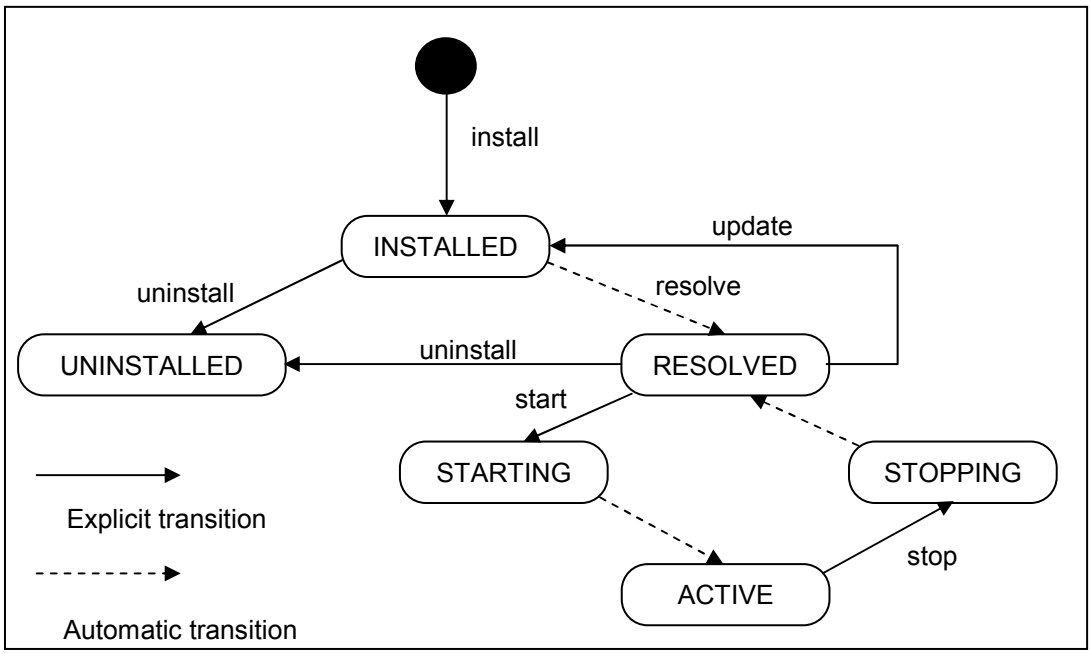

Figure 6 — Bundle states and their transitions

Each application or service on the platform must be written as an OSGi bundle. This requires writing a bundle activator that registers the services in the bundle with the OSGi framework.

5.3 Functional view

A functional view of an architecture customarily describes what a system does. It identifies, defines, and organizes the functions, processes, activities, and behavior of a system. The functional view of the architecture describes the system functions and behavior that have been chosen to meet the system requirements.

The architecture is agnostic to the functionality of the applications and implementations that it supports. Therefore, the subjects of a functional architecture are systems services provided by an system. The primary functions are: service discovery, security, system management and HMI services.

5.3.1 Service discovery

AMI-C's service discovery provides a mechanism to query for particular services and determine service information or other characteristics.

Four architectural elements provide service discovery: Bluetooth, OSGi, MOST, and Network Communication Model (on 1394 Automotive). Any two of these elements shall have a bridge between them.

For an system with a host, service discovery will be provided by the OSGi framework. For an system with a host and a Bluetooth interface, the network service discovery protocol will be bridged to the OSGi service discovery protocol. For an system with a host and 1394 Automotive network, the service discovery protocol described in the Network Communication Model Common Message Set will be bridged to the OSGi framework. For an system with a host and a MOST network, the MOST service discovery protocol will be bridged to the OSGi framework.

For a networked system without a host, the native service discovery protocol, if any, shall be used. If a native service discovery protocol does not exist, the Common Message Set and Network Communication Model define the service discovery protocol. In this protocol, devices connected to the network are grouped into service categories based on their characteristics.

The discovery of services available to application software includes the ability to inquire about the current set of available services and the ability to track that set for changes, such as the appearance of a new service.

Devices on the network must be discovered before the services provided by those devices can be discovered. Device discovery uses protocols specific to a given network, but service discovery should be network independent.

5.3.2 Security

The purpose of system security is to protect the devices, applications, data, and operation of the system and the vehicle from theft, unauthorized installations, unauthorized information transfer, and corruptions.

The mechanisms used to accomplish security include authentication, authorization, and encryption. Both devices and applications may use one or more of these mechanisms to satisfy their security requirements.

Authentication verifies identity. Authorization limits what actions are allowed based on the assumed identity. Encryption hides information from unwanted discovery.

The following contents in this section are applicable to network security:

1. Network categorization based on connection type

The architecture addresses three types of network connectivity: wired (e.g. MOST), short-range wireless (e.g. Bluetooth), and long-range wireless (e.g. GPRS). Each connection type has different requirement regarding network security.

2. Vehicle safety

In the architecture, all vehicle safety systems (such as anti-lock breaking system, or airbag supplemental restraint system) are isolated from the telematics and/or multimedia systems and networks by the vehicle service interface. The vehicle service interface ensures that systems cannot interfere with vehicle safety systems. This standard does not specify how the vehicle service interface will be designed because the design is dependent on the attributes of the specific vehicle platform.

3. Integrity of network transactions

A network technology has transaction integrity if the network protocol ensures that the intended recipient of a message receives the content of a message unchanged from what was sent by the sender and the recipient is provided with a means to verify that there had been no change to the content. Transaction integrity may not be required for all transactions.

4. Access authorization

A network technology provides access authorization if the network protocol ensures that the owner of a network object (such as a service, or data element, or device) can designate which network elements (such as nodes, devices or services) can access the object (e.g. invoke a service, change the state of a parameter, or read a data object). In general, authorization is not possible without authentication.

5 System authentication

A network technology provides authentication if the network protocol provides a means to designate devices and services that can be on the network and denies network access to devices and services that are not so designated.

The short-range wireless technology of AMI-C for Release 2 is Bluetooth. The Bluetooth core specification includes device authentication through pairing and service authentication through encryption.

Authentication over long-range wireless (packet data cellular such as GPRS or CDMA2000 1X) is carrier and application dependant.

6. Privacy

A network technology provides privacy of network transactions if the network protocol ensures that only the sender designated recipient of a network message can read it.

The Bluetooth core specification contains encryption of content transferred over-the-air. Likewise, long-range wireless (commercial digital cellular) provides over-the-air encryption.

For a wired network, third party access to content of a private transaction can be prevented by implementation specific application level encryption and/or by the security inherent in a closed, installed network.

Some types of content may be subject to external requirements for content protection. An example is the Digital Transmission Content Protection (DTCP) requirement for copyrighted video media. Both 1394 Automotive and MOST have a DTCP approved content transfer specification.

Table 1 summarizes network security capabilities for AMI-C Release 2.

Table 1 — Network security capabilities for AMI-C Release 2

Connection type		Wired network	Short-range wireless	Long-range wireless
Bus		1394 Automotive MOST	Bluetooth	Cellular network
Vehicle Safety		An AMI-C device is not a part of vehicle safety components. Any data transmission between AMI-C system and vehicle safety system is done through the vehicle interface.		
Integrity of network transactions		CRC (Cyclic redundancy check)	CRC	Yes (1)
AMI-C system authorization	Mechanism to provide authorization	IMP	No	N/A
AMI-C system authentication	Mechanism to authenticate devices	No	Yes Bluetooth paring	N/A
	Mechanism to authenticate services	No	Yes BT layer 2 security	Yes (1)
Privacy (prevention of interception)	Data sources	IMP	IMP	IMP
	Data flows	YES for DTCP for DVD stream data (2)	Yes Bluetooth layer 2 encryption	Yes (1)
	Contents protection	Based on each contents/media restriction		

YES means that the feature is present in either an AMI-C specification, a normative reference in an AMI-C specification, or is present in existing commercial implementations.

NO means that network technology is incompatible with this feature.

IMP means that the network technology provides the capability for the feature to be added by the implementer.

(1) Dependant on the individual network or service provider

(2) Both 1394 and MOST have accepted DTCP specifications. Although DTCP is intended to protect content on an isochronous stream, it can be implemented to provide asynchronous channel protection.

5.3.3 System management

System management is the function controlling the availability and capabilities of structural components. AMI-C specifies these interfaces and APIs. AMI-C does not provide system management policies, applications or tools.

AMI-C Release 2 includes

- System power management
- Start up and shutdown
- Device and service discovery

Release 2 does not include

- Service parameter control
- System resource management (e.g. network bandwidth)
- Fault management
- Tacking and logging
- Performance monitoring
- Diagnostics

Release 2 provides the following system management functions:

- Power control from the vehicle-side, as part of the vehicle interface. This is coordinated with three generic network states:
  - ON
  - OFF
  - SLEEP
- A set of activity states and messages at the vehicle interface level.
  - Boot state
  - Shutdown state
  - Sleep state
  - Active state
  - Diagnostic state

5.3.4 HMI services

Based upon the requirement to provide with automakers the ability to maintain control over how the vehicle user interface (UI) is implemented, specifies a content-based approach to HMI. For a content-based HMI, a device or application communicates HMI content information in predefined data structure. Only content is communicated, not presentation format. The specific characteristics of the vehicle's UI devices including the

method for user input and display characteristics are unknown to the application. The content-based approach has the added benefit of making applications or embedded devices portable.

The HMI specification assumes that a central entity (i.e., the HMI manager) will be used to manage access and use of UI resources. The HMI manager has full knowledge and control of the presentation and input of the vehicle UI devices. The HMI manager is vehicle specific. AMI-C does not place explicit requirements on the HMI manager, thereby allowing the automakers to scale the level of complexity for their implementation.

Figure 7, shows the HMI manager as the intermediary between the vehicle UI devices and the network or host application. The communication language between applications and the HMI manager is via the specified XML called Vehicle User Interface Markup Language (VUIML). The HMI manager uses rules specified by automakers for how to present that information on that vehicle's UI devices. Conversely, when the user makes an UI device input, the HMI manager provides a VUIML message to the respective application.

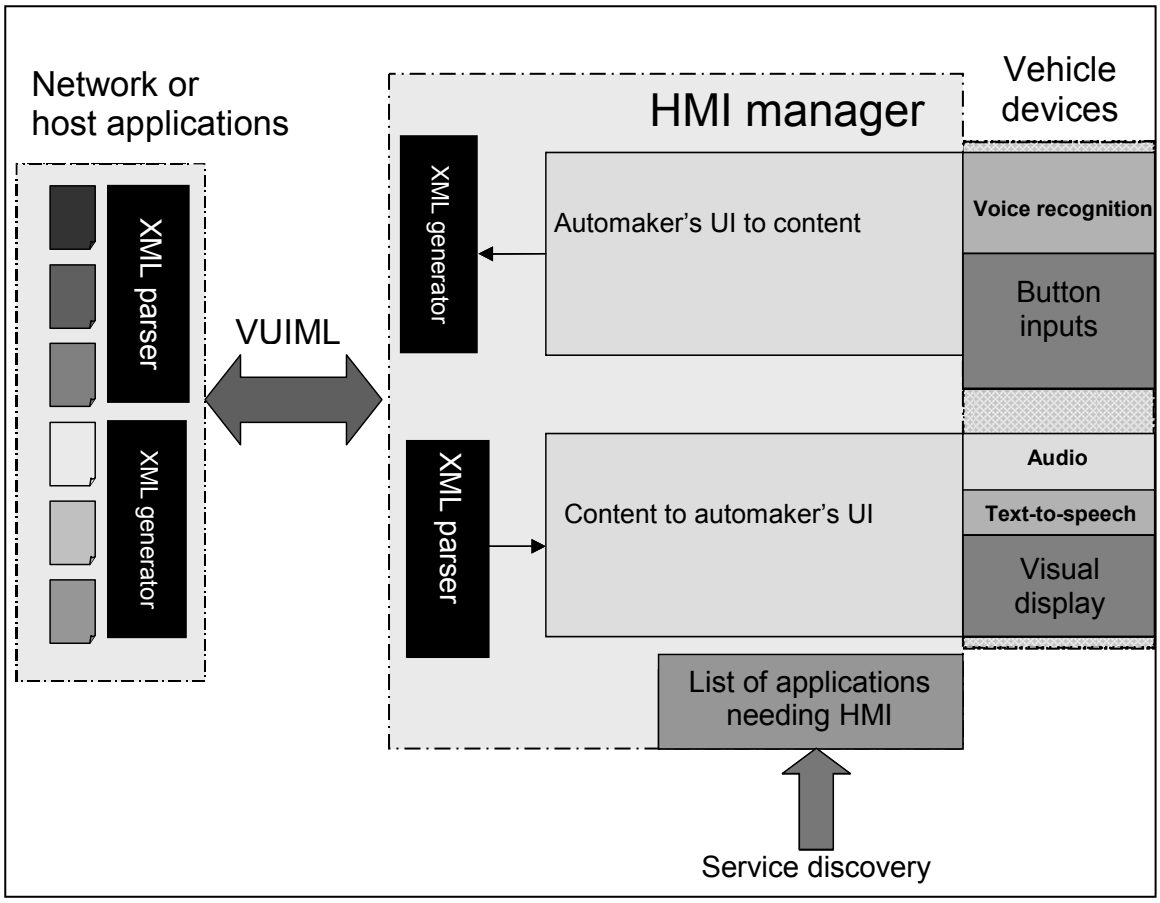

Figure 7 — Block diagram of an HMI manager

5.4 Application view of the architecture

The application view of the architecture focuses on the relationship of the use cases to the interfaces. An application view of architecture customarily identifies the relationship between the uses cases and the applications that meet the requirements implicit in the use cases. An application view also defines the applications and the relationships among the applications. It is important to note that does not specify the applications.

A use case describes the provision of user perceptible utility by a system or an element of a system. Within the domain, the provision of utility is by a network implementation or by an application on a host computing platform (or possibly both).

This standard provides common interface specifications between implementations or applications and the services of the vehicle, multi-media network or the computing system. Therefore, the interface specifications enable the use cases.

For a multi-media network implementation, the interface is an element of the network protocol, customarily one or more application level messages.

For a software application, the interface is customarily an application programming interface (API).

Figure 8 shows the application view of AMI-C use cases.

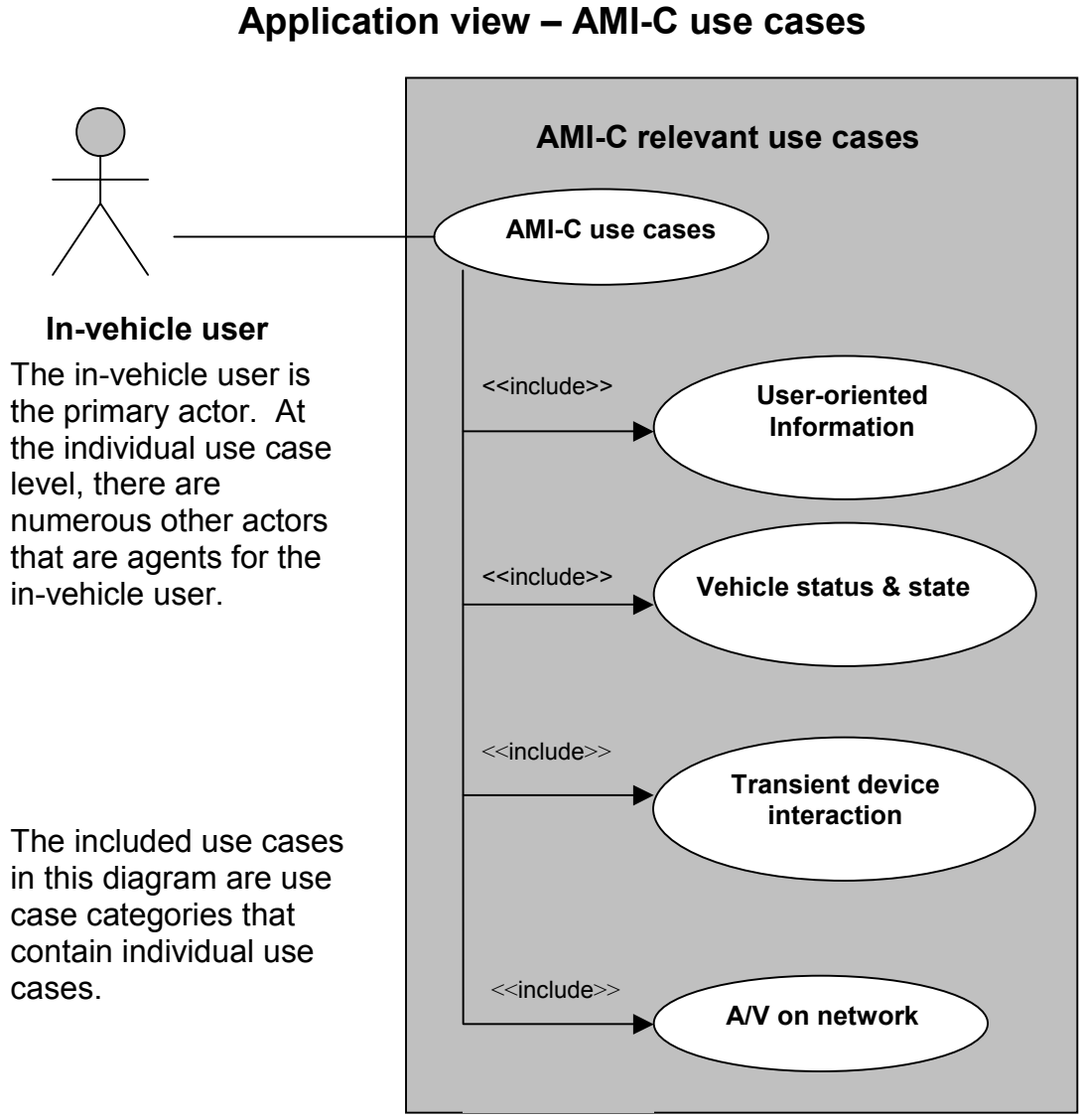

Figure 8 — Application view of AMI-C use cases

5.4.1 Driver information applications

Use cases in this category address the provision of real-time and/or location specific information to driver. (Note: This section and the following few sections will use the abbreviated use case names as in AMI-C 1001, AMI-C use cases.)

Examples of use cases:

- GUID 1: On-board navigation system.
- GUID 2: Off-board navigation system.
- GUID 3: Navigation with communication system.
- GUID 4: Traffic information using multiplex broadcasting.
- ITS 1: Obtaining traffic information from vehicle service provider.
- GUID 6: Obtaining roadside information.

5.4.2 Vehicle status applications

Use cases in this category address the real-time, event driven vehicle status message to or transaction with a service center. Examples of such use cases include:

- CUST 1: Acquisition of customer data.
- EMER 1: Automatic vehicle accident notification.
- EMER 2: Emergency call with vehicle diagnostics.
- EMER 3: User-initiated emergency call.
- FLEE 1: Submitting vehicle transit data.
- FLEE 2: Installing fleet management software.
- SERV 3: Diagnose system fault.

5.4.3 Transient device interaction

Use cases in this category address the interaction between or among vehicle services, vehicle HMI and one or more personal devices. Examples of such use cases:

- MOBI 2: Mobile phone voice call.
- MOBI 3: Mobile phone data call.
- MOBI 6: Synchronizing personal data between vehicle and a personal mobile device.

5.4.4 Network audio/video applications

Use cases in this category address the audio and video entertainment, primarily on a wired network. Examples of such use cases include:

- ENTE 4: Receiving broadcast analog/digital audio.
- ENTE 5: Audio media player (cassette, DVD, CD)
- ENTE 6: Receiving broadcast analog/digital video.
- ENTE 7: Video media player (DVD, VCR).
- ENTE 8: Video game player.

6 System architecture configurations

The AMI-C architecture defines two system levels:

— Level One: Systems without an Java host.

— Level Two: Systems with one or more Java hosts.

Either level shall provide a vehicle interface with a basic set of services defined by the standard, either through message level interaction (with an available, supported network) or through application programming interfaces (APIs) (in which case a host is available). Several network topologies are possible at each system level.

7 Relationship between uses cases and interface specifications

Figure 9 provides a simplified overview of an application as a user of a service. The services interface provides a standard way for an application to access a service.

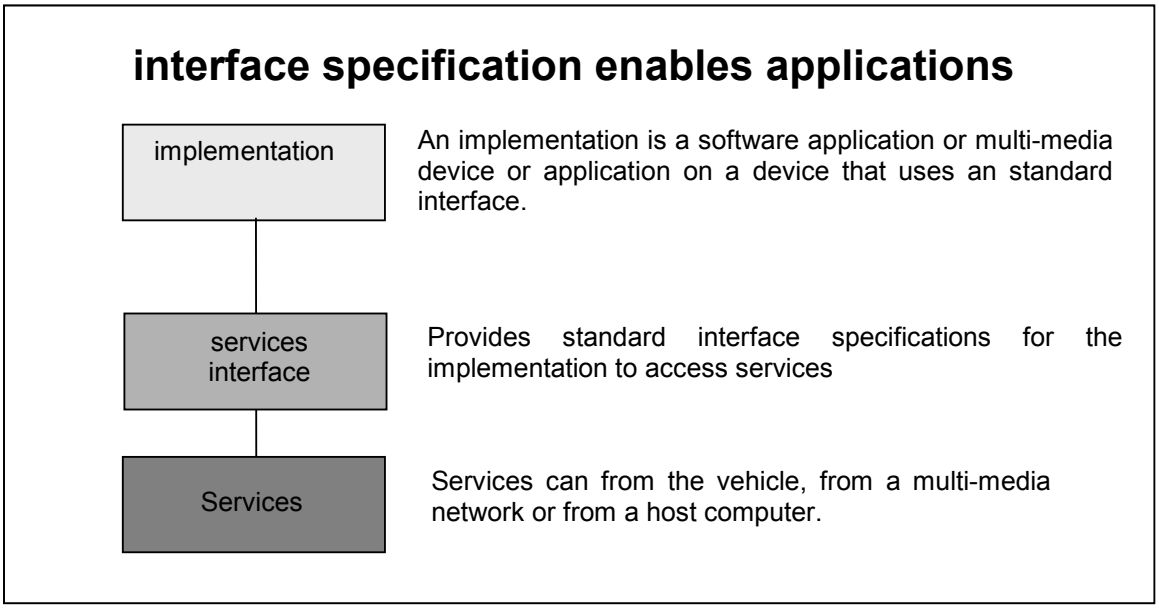

Figure 9 — Interface specification enables applications

For definition, an actor is any participant in a use case. Figure 10 describes the relationship between a use case actor and the specified interfaces. The primary actor is the in-vehicle user. Although there can be other actors, they can be viewed as agents for the in-vehicle user. This actor interacts directly with vehicle HMI, which in turn interacts with an application through an specified interface. The application provides the functionality for the user perceptible utility of the use case. The service required for the functionality of the application is accessed through an service interface.

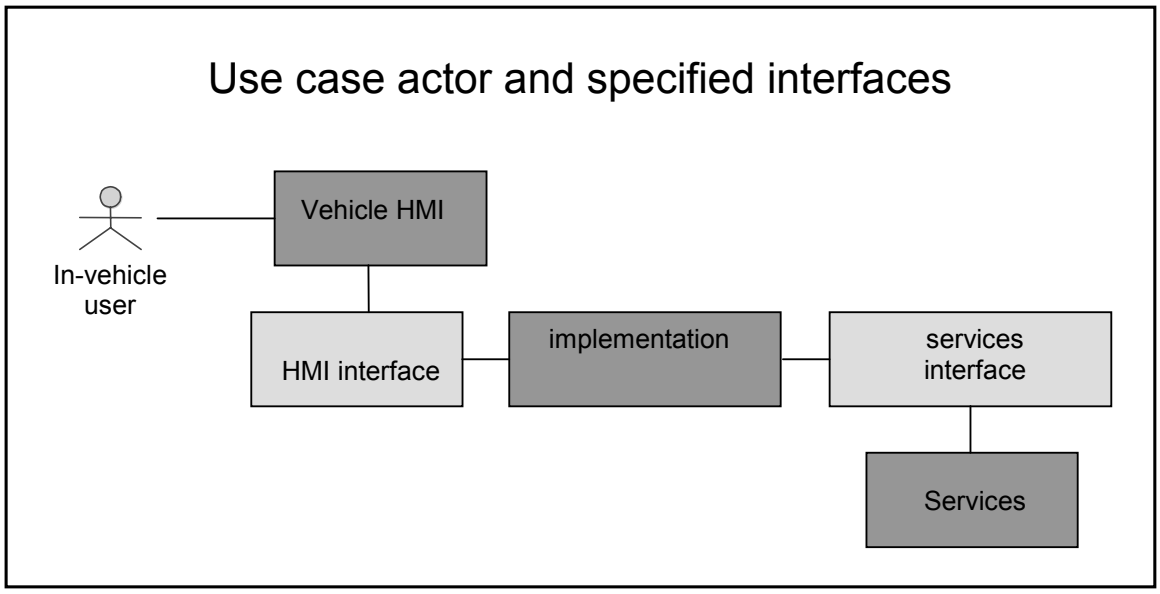

Figure 10 — Use case actor and specified interfaces

This standard specifies interfaces to services not applications. However, the interface requirements of possible applications need to be considered to identify which service interfaces are required. Figure 11 provides a description of the process used to identify the interface specifications required to support a use case. The highest level of this process is as follows:

— Step 1. An abstract or generic application is outlined which supports each use case.

— Step 2. The individual service requests needed to support the functionality of the application are identified.

— Step 3. Similar service request are collected into service access requirements.

— Step 4. An interface specification for accessing a service is developed based on the requirements.

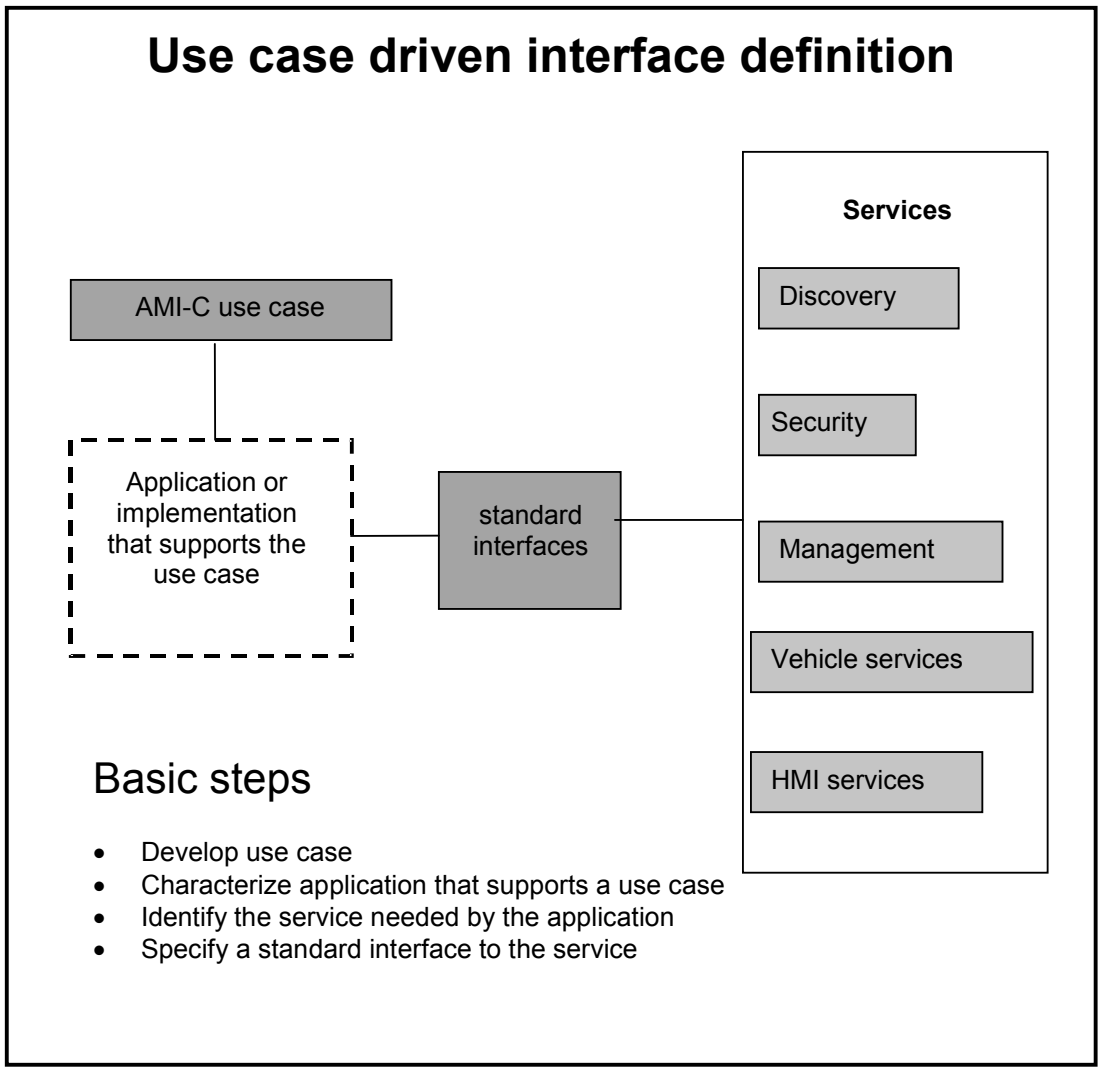

Figure 11 — Use case driven interface definition

ICS 43.040.15

Price based on 29 pages