Military Data Link Integration Application

James T. Sturdy

Honeywell
Defense and Space Electronic Systems
9201 San Mateo Blvd., NE
Albuquerque, New Mexico 87113-2227
(505) 828-5703 Office
(505) 828-5500 Fax
james.sturdy@honeywell.com

Abstract

The military uses tactical data link radios to send and receive digital voice, data, and video between vehicles and command and control facilities. These data link radios are interfaced to various mission computers and display systems. As outlined in the Joint Tactical Data Link Management Plan, a wide range of legacy military platforms will be upgraded to incorporate new data link radios through 2015 and beyond. The upgrade costs to do this will be enormous if traditional subsystem upgrade approaches are used. A need exists for a common, scalable and low cost military data link integration solution that can be used in multiple and disparate platform applications. The author will discuss such a design approach that can be used on each military platform application. The solution processes new and evolving messages with a database driven design so that the user can control message activation, deactivation, and processing instructions for each unique platform application. The database used for this capability is created by and maintained by the user. This allows a common design to work and to evolve on each unique platform without the need to modify the operational software.

Background

The military uses tactical data link radios to send and receive digital voice and data between their air, land, sea and space vehicles, and command and control facilities. On each vehicle and in each command and control facility, these data link radios are interfaced to various communications/mission computers and display systems. The data transmitted across these data link radios comply with message formats and message protocols defined by various message standards. The mission and display systems use the data contained in these messages from external sources and generate the data put into these messages sent to external systems.

These military data link radios include UHF line of sight (LOS) radios, UHF DAMA SATCOM, EHF MDR SATCOM, HF radios, Joint Tactical Information Distribution System (JTIDS), Multi-function Information Distribution System (MIDS), and Joint Tactical Radio System (JTRS). Military data link message standards can include Link 4, Link 11, Link 16, Link 22, and the Variable Message Format (VMF). Examples of mission and display systems are vehicle controls and displays equipment, mission computers, workstations, and network servers.

The Department of Defense (DoD) has recently selected the Link-16 data link J-Series message set (in accordance with MIL-STD-6016) as the standard for use on military platforms for tactical data link operations. In addition, the DoD is currently developing the Joint Tactical Radio System (JTRS) to use as the standard data link radio system. Each existing and future military platform will use the JTRS set for its tactical data link capability.

As outlined in the Joint Tactical Data Link Management Plan[1], a wide range of legacy military platforms will be upgraded to incorporate the JTRS through 2015 and beyond. These same platforms are currently deployed with existing subsystems that generate information used by or consume information provided by various existing and disparate military data link systems.

[1] Department of Defense Command, Control, Communications, Computers, and Intelligence (C4I) Joint Tactical Data Link Management Plan, dated June 2000.

When the new JTRS equipment is introduced into these platforms, there will be a need to interface the existing platform subsystems with the new JTRS equipment. Also, each existing subsystem will need to be upgraded to utilize the evolving data link message sets.

Existing implementations use a subset of the J-Series message set, or a different and older data link message set such as Link 4 or Link 11. Also, many existing subsystems were designed to interface with older data link radio equipment and are not compatible with the newer data link radio equipment and message processing protocols. These are typically point solutions unique to the specific platform they are implemented on. These point solutions include receive, transmit, and processing functions. Receive functions receive the message from the data link radio, decode the message data, and send the data to the appropriate subsystem. Transmit functions collect specific data from platform subsystems, encode the data into the proper message format, and send the message to the data link radio. Processing functions act on selected data elements to perform specific tasks such as filtering, correlation, keeping track files, and other mission specific functions. Each solution only implements the subset of messages required for that platform's mission. When future changes are needed because the military wants to add, delete or modify specific messages and message processing for the platform, the military is faced with returning to the previous point solution and paying to implement the changes. Thus, the existing solutions do not provide the military with the capability of modifying specific messages without a major product redesign on each unique platform. For example, on fighter aircraft the mission computer interfaces to the existing data link radio and performs the message processing for the message subset implemented on the specific fighter. The display system also processes those messages that contain situational awareness information, but tailors it for the specific fighter mission and display requirements.

There are many different implementations of military data link integration used in the United States as well as in NATO countries. Each of these are point solutions that were designed specifically for the vehicle they are used on. Many have recognized the high costs associated with legacy equipment modifications and vehicle integration. As an example the recently awarded JTRS Cluster 1 is already driving cost growth because of *larger-than-expected price tags for installing the radios aboard older aircraft and ground vehicles*[2]. Furthermore, *adding to the complexity of Cluster 4 is the requirement to accommodate 65 different types of aircraft*[3].

The goal of the JTRS program is ambitious even if you limit it to be an F^3 replacement for the existing legacy radios. This assumes that you do not change any of the existing data link functionality in other interfacing subsystems (e.g. mission systems and display systems) on each platform. In this manner, the JTRS can behave just like the replaced radio. And for this goal the JTRS may be ultimately successful. However, this is only a small part of the total life cycle integration need. Once the JTRS has been installed, then the capability will exist to add new messages and message processing, and this will require changes to the subsystems on each unique platform.

With approximately 28,000 data link platforms[4] in use or planned throughout the DoD, the upgrade costs for the associated existing platform subsystems will be enormous if traditional

[2] National Defense (ISSN 0092-1491), October 2003 issue, page 19.
[3] Ibid page 20.
[4] Department of Defense Command, Control, Communications, Computers, and Intelligence (C4I) Joint Tactical Data Link Management Plan, dated June 2000, page B-5.

subsystem upgrade approaches are used. Traditional upgrade approaches involve point solutions and upgrades by different integrators on each platform application. A need exists for a common and low cost military data link integration (MDLI) solution that can be used in multiple and disparate platform applications.

What is a common and scalable solution?

Both industry and the DoD have developed a number of standards that, in part, specify what a common and scalable solution should be. Some relevant examples are the Joint Technical Architecture (JTA) and Software Communications Architecture (SCA). There are many others. However, the purpose of this paper is not to repeat these standards. Instead, this paper discusses some specific features of an MDLI application that ensures commonality and scalability. In addition, this paper describes how a common MDLI application provides the military user with the capability of making their own changes to the way mission functions and data link functions are integrated on the host platform. Here the military user may be a member of a host platform program office, prime contractor, or subcontractor. The user makes these changes without the costs associated with verification, qualification, and certification that are otherwise typical when mission and/or data link systems are modified. The following text discusses some enabling features of the MDLI application.

The MDLI application implements the complete or full J-Series message set (or other message sets as required) with a database driven design so the military user can control message activation, message deactivation, and message processing instructions for each unique platform application. The database used for this capability is created and maintained by the military user. The military does not need to pay any company to do this for them. This allows a common MDLI application to work on each unique platform without the need to recompile the operational software.

The MDLI application implements a database-driven design that allows automatic re-configuration of the application for each unique platform without the need for software changes to the solution. The database used for this capability contains the interface instructions for each type of subsystem used on each unique platform and allows the common application to work on each unique platform without the need to recompile the operational software.

The MDLI application implements special message processing functions. These functions provide correlation of similar data from disparate sources, target track files, formatting of data into situational awareness display formats, automatic event triggers and associated actions, automatic triggers for transmit messages, mission recording and playback, and others. In addition, standard video outputs for data link display formats are provided.

The MDLI application provides the military with a low cost solution. The user can tailor how the MDLI application works on each unique host platform simply by updating the host platform configuration database and User Modifiable Instructions (UMI) database. This gives the user control over the use and operation of the solution without the need to pay someone to modify it for them.

The MDLI application is flexible, scalable, and reusable for each unique host platform. Through the host platform configuration database, it can be used on multiple unique host platforms to interface with available communications subsystems and other subsystems without any required modifications.

The MDLI application implements the complete set of J-Series messages and processing rules defined in MIL-STD-6016. The user need only activate or deactivate messages as required for each unique host platform. Other messages sets and protocols could also be implemented.

The MDLI application implements special message processing functions and utilities that can be activated or deactivated by the user. These message processing functions and utilities allow the user to add value to the data and messages sent by the MDLI application to available communications subsystems and other subsystems on the host platform.

The MDLI application provides standard display system interfaces to support flexible and user programmable display formats to view tactical data and legacy subsystem data. This enables the user to create new display pages without impacting the flight worthiness or qualification of other display pages or other mission functions.

The MDLI application is supported with a Ground Based Software Tool that allows users to define and create the host platform configuration databases and UMI databases on a workstation in an office environment. No special equipment or dedicated planning stations are required.

The MDLI application common solution saves the user money on training and maintenance costs across all host platforms. This savings is achieved because it is less expensive to train the users on a common solution than to train them on many different product solutions.

Finally, the MDLI application is derived from existing and proven commercial products and technologies used in safety critical applications. For example, many of the features listed above have already been proven in the Honeywell Communications Management Unit (CMU) that is utilized in the airborne data link solutions used by the commercial airlines today. The model-based CMU design ensures that these proven commercial capabilities can be easily ported to military data link platforms.

MDLI Application Description

The MDLI application can be applied to any data link message standard or group of message standards as required. However, this description uses the Link 16 tactical data link message standard as its example.

The MDLI application is a software process or service that executes on a host processor. Figure 1 illustrates how the MDLI application is implemented. This same implementation is envisioned for each platform in which the MDLI application is used. The MDLI application consists of the following functions: Data Link Message Processing, Data Link Platform Integration, Host Platform Configuration Database, Message Parameter Database, and User Modifiable Instructions (UMI) Database. These MDLI application functions are implemented in a computer system available on each host platform. The host computer system is expected to consist of a Host Applications Processor module, Image Processing Module (IPM), Input and

Output (I/O) Modules, and a computer cabinet with the necessary Computer Module Interconnects. The MDLI application functions execute on the Host Applications Processor and interface with existing software Mission Applications that are also executing on the Host Applications Processor through pre-defined data exchange protocols in the Host Platform Configuration Database. The MDLI application functions also interface with external Communications Subsystems and other Subsystems through the host computer system IPM, I/O Modules, and their associated Host Computer Module Interconnections. The pre-defined data exchange protocols consist of host computer system port addresses, message structures and formats, and data exchange command sequences. The MDLI application functions utilize these host computer resources to exchange data with external subsystems over pre-defined system interfaces consisting of I/O and Link 16 Messages. Additionally, the MDLI databases are created off the platform on a Ground Based Software Tool. These databases are then uploaded to the Host Application Processor memory through a data loader using a Data Loader Cartridge or other data loader interface. These databases are used by the Data Link Message Processing and Data Link Platform Integration functions to automatically configure the MDLI application on the host platform, and to implement user defined instructions.

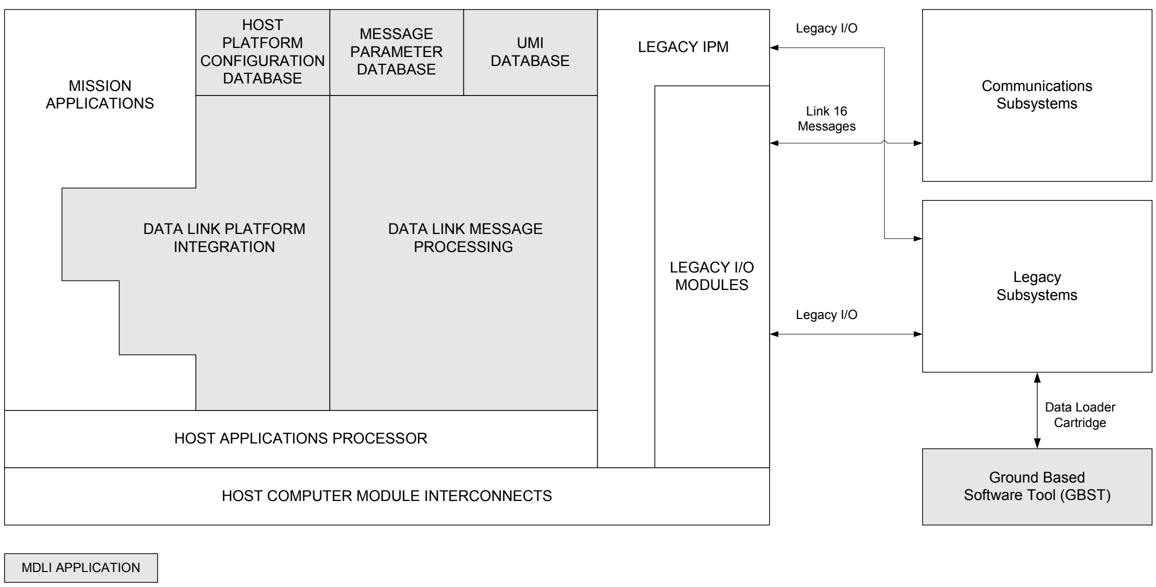

Figure 1 - Military Data Link Integration Application Overview

Figure 2 illustrates the MDLI application decomposed into its major functions - Data Link Message Processing and Data Link Platform Integration. It also includes the Host Platform Configuration Database, the Message Parameter Database, and the User Modifiable Instructions (UMI) Database. The Data Link Message Processing function implements the Link 16 message set with its processing rules and special message functions. It interfaces with the Communications Subsystems on the host platform, databases, and the Data Link Platform Integration function. The Data Link Platform Integration function implements the rules and instructions needed to interface with and interact with the various Subsystems on the host platform, as well as Special Platform Functions, under the control of the Data Link Message Processing function. Control is accomplished through the Control and Status Exchange interface.

The Data Link Platform Integration function also implements the data loader function that is used to update the Host Platform Configuration Database and the UMI Database using the host platform's Data Loader device.

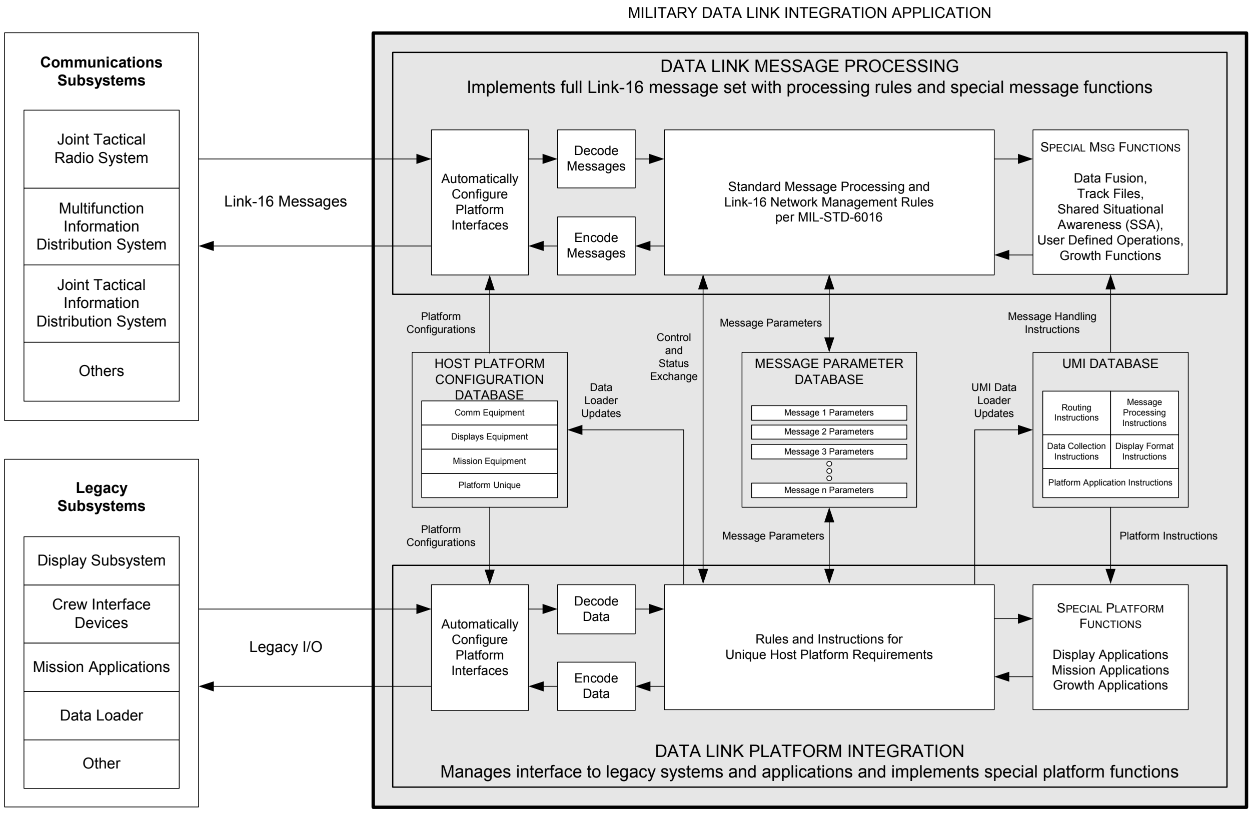

Figure 2 - Military Data Link Integration Application Implementation

Data Link Message Processing

Data Link Message Processing begins after application of power when the Host Applications Processor initiates its execution. Initialization establishes the interfaces to the Communications Subsystems using pre-defined instructions in the Communications Equipment configuration database. This database contains the instructions to identify which Communications Subsystems are available on the host platform. The database also provides Host Applications Processor interface port addresses and protocols, message structures and formats, and command sequences to accomplish data exchange for each of the available communications subsystems. After initialization is complete, incoming messages are decoded and outgoing messages are encoded at a prescheduled rate using the Standard Message Processing and Link-16 Network Management Rules per MIL-STD-6016. Message parameters obtained from incoming data are stored in the Message Parameter Database.

Special Message Functions are executed based on UMI Database instructions. The Special Message Functions task uses Data Collection Instructions to identify data parameters to be collected from Subsystems on the host platform. These collected data parameters are stored in the Message Parameter Database. Message Processing Instructions are used to activate utilities on user specified data parameters. These utilities include data fusion algorithms, creation and update of track files, creation and update of shared situational awareness (SSA) information, and other user defined data operations. The results of these utility operations are stored in the Message Parameter Database. Routing Instructions are used to identify data in the Message Parameter Database to be sent to specific Subsystems and Communications Subsystems. Display Format Instructions are used to format selected data in the Message Parameter Database for display.

The data tagged in the Message Parameter Database for output to available Communications Subsystems is formatted in accordance with MIL-STD-6016 Message Rules, and then is encoded into the appropriate message format and transmitted to the appropriate Communications Subsystem.

Data Link Platform Integration

Data Link Platform Integration begins after application of power when Host Applications Processor initiates its execution. Initialization establishes the interfaces to Subsystems using pre-defined instructions in the Displays Equipment configuration database, Mission Equipment configuration database, and Platform Unique configuration database. These databases contain the instructions to identify which Subsystems are available on the host platform. These databases also provide Host Applications Processor interface port addresses and protocols, message structures and formats, and command sequences to accomplish data exchange for each of the available legacy subsystems. After initialization is complete, incoming subsystem data are decoded and outgoing subsystem data are encoded at a prescheduled rate using the Rules and Instructions for Unique Host Platform Requirements. Incoming and outgoing data are stored in the Message Parameter Database.

Special Platform Functions are executed based on UMI Database instructions. The Special Platform Functions task uses Data Collection Instructions to identify data parameters to be collected from Subsystems on the host platform. These collected data parameters are stored in the Message Parameter Database. Platform Application Instructions are used to activate utilities on user specified data parameters. These utilities include display applications, mission applications, and other user defined data operations. The results of these utility operations are stored in the Message Parameter Database. Display Format Instructions are used to format selected data in Message Parameter Database for display.

The data tagged in the Message Parameter Database for output to available Subsystems is formatted in accordance with the Host Platform Requirements, and then is encoded into the appropriate message format and transmitted to the appropriate Subsystem.

Benefits and Capabilities

The MDLI application automatically initializes its interfaces to Communications Subsystems on the host platform. This capability allows the MDLI application to be hosted on many different host platforms without the need to modify it for different communications equipment configurations.

The MDLI application automatically initializes its interfaces for Subsystems on the host platform. This capability allows the MDLI application to be hosted on many different host platforms without the need to modify it for different legacy mission and displays equipment configurations.

The MDLI application implements user specified instructions associated with Link 16 message processing and unique host platform functions. This capability allows the user to tailor how Link 16 messages are processed, to define special message processing functions, and to define special platform integration functions without the need to modify the MDLI application for each host platform configuration. A benefit of this approach is that in a particular platform that has existing subsystems that already implement a subset of the required data link message processing, the user can instruct the MDLI application to simply pass through those messages to and from those subsystems. This is accomplished with the UMI Database instructions. Since the MDLI application implements the full set of Link-16 messages, the UMI Database can be used to instruct the MDLI application to ignore messages not currently used on the platform and pass through messages currently implemented on the platform to achieve existing capabilities. Then, when new capabilities are needed on the platform, the UMI Database can be used to instruct the MDLI application to activate new messages, implement their message processing rules, and create display formats using the new messages. These new message capabilities can be implemented without impacting the existing subsystems, or with only minor impacts to receive interface data needed from those subsystems.

Since the MDLI application is a software process or service, it can be implemented in an existing computer system on the host platform as illustrated in Figure 1. It can also be hosted on a General Purpose Processor (GPP) module as illustrated in Figure 3, or an Image Processing Module (IPM) as illustrated in Figure 4.

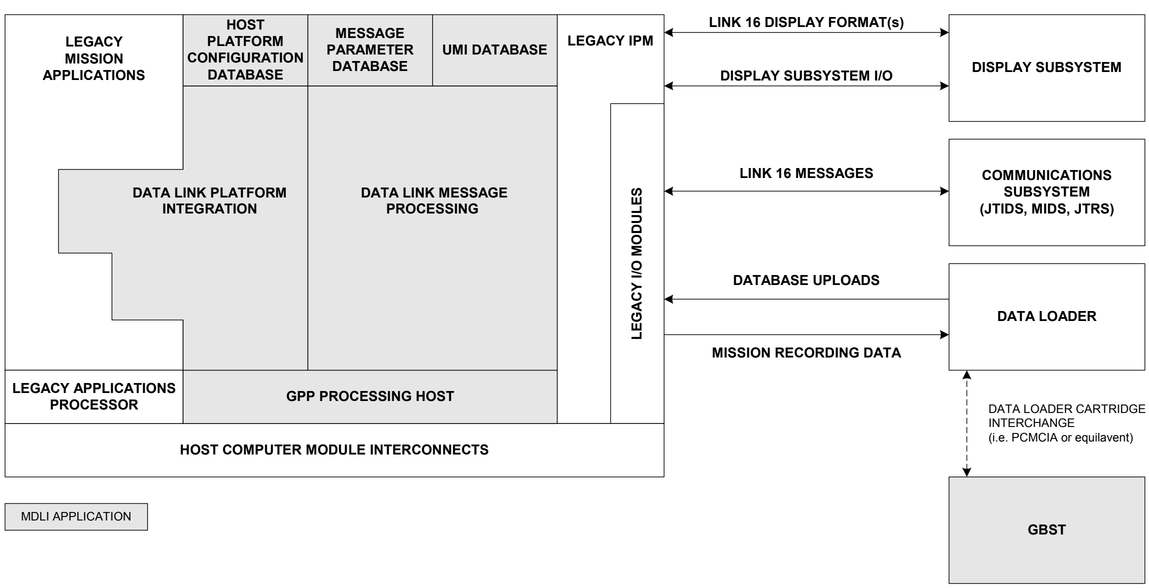

Figure 3 - MDLI Application Hosted on a GPP

In Figure 3, the Host Platform Configuration Database is used to define the interfaces between MDLI application and Mission Applications executing on the Legacy Applications Processor. The Host Platform Configuration Database is also used to define the interfaces to IPM and I/O Modules.

In Figure 4, the Host Platform Configuration Database is used to define the interfaces between the MDLI application and Mission Applications executing on the Legacy Applications Processor. The Host Platform Configuration Database is also used to define the interfaces to I/O Modules. The IPM application is replaced with the Image Processing Host module.

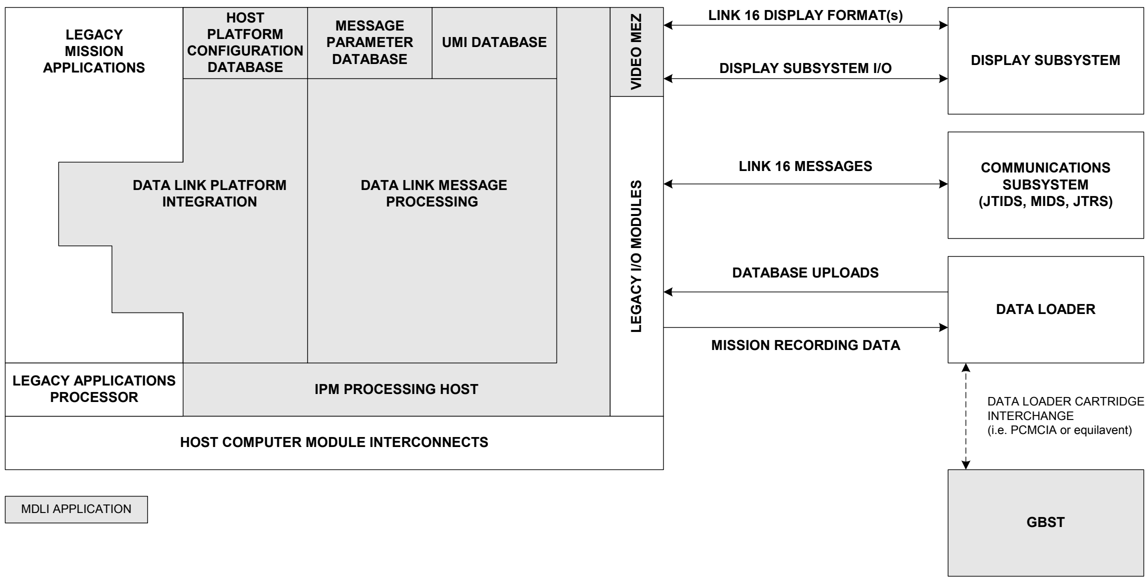

Figure 4 - MDLI Application Hosted on an IPM

The advantage of hosting the MDLI application on a General Purpose Processor module or an Image Processing Module is that these provide more flexibility in implementing the MDLI application on host platforms that do not have an existing computer system. Also, the existing computer system may not have the processing and memory resources required for the MDLI application. In these cases, the General Purpose Processor module or Image Processing Module can be integrated into any existing subsystem equipment that has a spare card slot.

CMU/CMF Heritage

The common and scalable MDLI application is possible because it is based on an existing civil aeronautical Communications Management Unit (CMU) design that accomplishes basically the same task. This existing design has been implemented as a stand alone line replaceable unit[5] and also as a software partition[6] within an integrated avionics computer platform.

Figure 5 illustrates the existing civil Communications Management Function (CMF) implementation in accordance with ARINC specification 758. As can be seen, it is very similar to the MDLI application illustrated in Figure 2. The CMU and CMF products use the same core software developed using the ObjectGeode modeling tool. The CMU/CMF implement the various network layers defined by the ISO Open Systems Interconnect (OSI) standard used to interface with different communications systems. With this layered and open architecture, the CMF can be used in systems on any mobile communications node as well as in ground-based communications infrastructures to intelligently integrate each civil communications network (VHF, HF, and Inmarsat SATCOM) together into a global communications network. The OSI standard defines seven layers that are the standard layers used today in most communications networks, including the Internet.

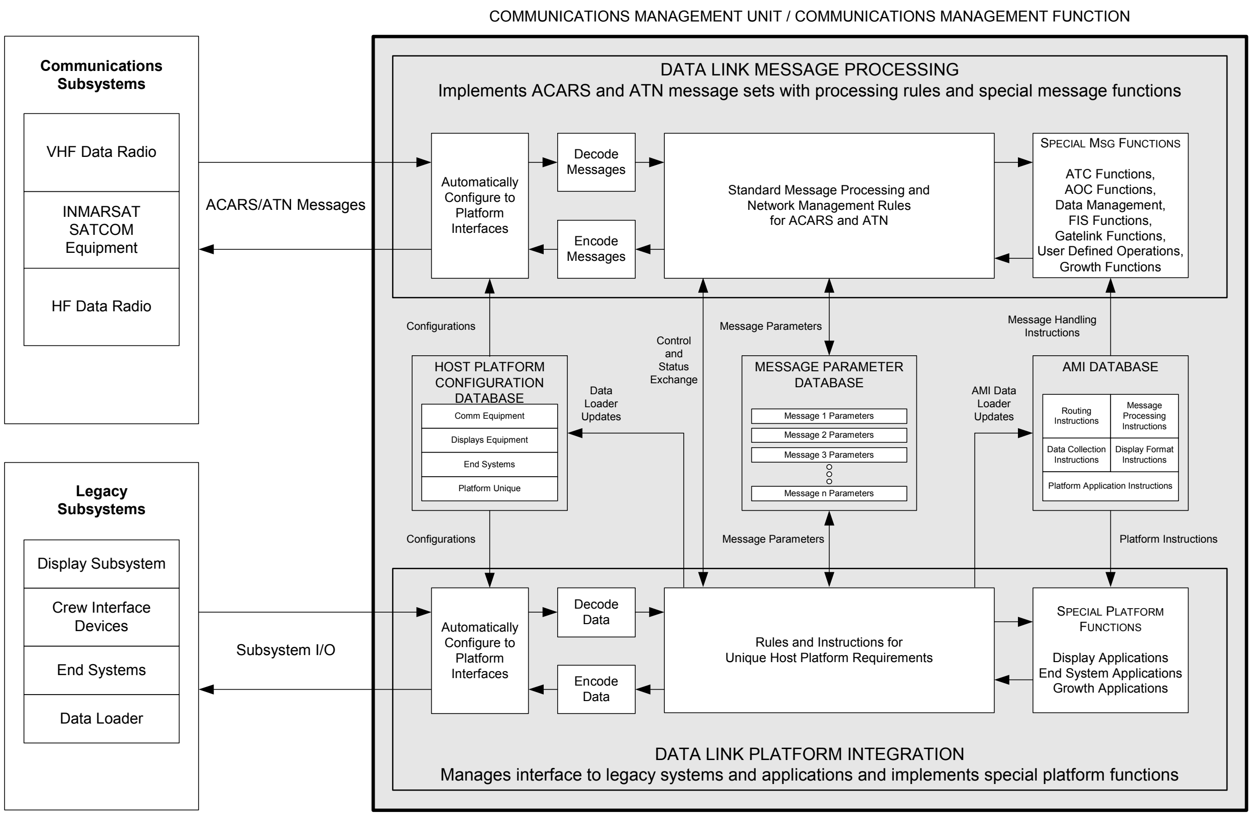

Figure 5 - Civil CMU/CMF Application for ACARS and ATN Data Link Management

Presentation layers are generally handled by each end system application. The existing civil end systems that the CMF exchanges data link messages with are ARINC 623 Airline Communications Addressing and Reporting System (ACARS) functions, Aeronautical

---

[5] Honeywell's Mark III CMU is one example of an ARINC 758 compliant civil communications management unit.
[6] Honeywell's Communications Management Function (CMF) is also ARINC 758 compliant and is used on several military tanker/transport aircraft to meet GATM requirements.

Operational Control (AOC) functions, Aeronautical Telecommunications Network (ATN) functions, and some special purpose Data Management functions. Session and transport layers are either not used or are shared between end systems and the CMF application.

As implemented today, the CMF integrates disparate civil aeronautical communications networks. Each civil end system exchanges data link messages across the various civil network paths (ARINC 618, ARINC 619, ARINC 656, and ATN) in accordance with civil communications requirements. Each network path interfaces with the selected communications radio equipment using the appropriate link and physical layers for high and low speed ARINC 429 interfaces, High Speed Ethernet (New Link) Interfaces, or MIL-STD-1553B interfaces.

The existing CMU/CMF software application has been developed using the ObjectGeode System Design Language (SDL) modeling tool. The ObjectGeode SDL was originally developed for use in the telecommunications industry and is ideal for communications networking product development. As a result, the CMF model can be rapidly modified to create an MDLI application, and the ObjectGeode autocode generation tools can be used to significantly reduce the MDLI application software development cycle.

An existing AirSim tool models the civil data link networks (VHF, HF and Inmarsat SATCOM) and external end system interfaces to the CMF. The AirSim tool is available for simulation and analysis of the CMF software in a PC workstation environment, and can be used for CMF analysis and testing, demonstrations, and training. The AirSim tool could be modified, or another tool could be used, to add models for military data link networks to support the MDLI application capabilities.

Conclusion

The DoD has over 28,000 existing or planned data link platforms that either have or will be upgraded with newer military data link radios such as JTRS. In addition to the installation of the data link radios, the platform missions will continue to evolve as new messages are implemented to increase and/or enhance the platform's capability. If traditional subsystem upgrade point solution techniques are used, the cost to implement these evolving message changes will be enormous, and perhaps unrealizable due to funding constraints. The MDLI application provides a common and scalable design solution that substantially reduces this future cost, allowing the military to make unique platform changes through user modifiable instructions in a database.

Leveraging the successful commercial CMU and CMF products, the MDLI application provides a flexible means for integrating mission and data link functions. The military user community should seriously consider the MDLI application as a solution that can greatly reduce data link integration life cycle costs.