

STRAWMAN FOR
ARINC PROJECT PAPER 830
AIR/GROUND INFORMATION EXCHANGE (AGIE)
USING INTERNET PROTOCOLS (IP)

AEEC Project Paper 830 Strawman Proposal
Aircraft-Ground Information Exchange – AGIE

Revision: - Draft 5.1

Proposed by:

Teledyne Controls

11 January 2011

Updated 28 January 2011 HB

Table of Contents

1	Introduction.....	4
1.1	Purpose.....	4
1.2	Scope.....	4
1.3	Document Overview.....	4
1.4	Related Documents.....	5
1.5	Document Precedence.....	5
1.6	Regulatory Approval.....	5
2	ARINC 830 “AGIE” Definition.....	6
2.1	AGIE Purpose/Objective.....	6
2.1.1	AGIE Goals.....	6
2.2	AGIE High Level Requirements.....	7
2.3	General Description.....	8
2.4	High Level Functional/Interface Description.....	11
2.5	Policy and Quality of Service Considerations.....	11
2.6	Certification and Partitioning Considerations.....	12
2.7	Assumptions/Constraints.....	12
3	AGIE Functional Specification.....	13
3.1	AGIE Principles of operation.....	13
3.2	AGIE Topology.....	14
3.3	Link management.....	15
3.4	Protocol Binding.....	16
3.4.1	Client/Server Protocol Binding.....	16
3.4.1.1	SOAP over http Binding.....	16
3.4.1.2	Simple XML Binding.....	17
3.4.1.3	AMQP Binding.....	17
3.4.2	Server/Server Protocol Binding.....	17
3.4.2.1	Server/Server Protocol 1.....	17
3.4.2.2	Server/Server Protocol 2.....	17
3.5	AGIE Addressing.....	17
3.5.1	The AGIE Address.....	18
3.5.1.1	AGIE Client Address.....	19
3.5.1.2	AGIE Server Address.....	19
3.5.2	Address Resolution.....	20
3.5.2.1	Address resolution sequence.....	20
3.6	AGIE data delivery management.....	21
3.6.1	Data transfer prioritization.....	22
3.6.2	Delivery status management.....	23
3.6.2.1	Delivery Date/Time Parameters.....	23
3.6.2.2	Inability to deliver.....	25
3.7	AGIE system configuration management.....	25
4	AGIE Interfaces and Protocols.....	27
4.1	AGIE message definitions.....	27
4.1.1	AGIE message attributes.....	27

ARINC Project Paper 830 AGIE Strawman Proposal

4.1.1.1	STANDARD Message Attributes	28
4.1.1.2	COORDINATION message attributes	31
4.1.1.2.1	NOTIFICATION message attributes	33
4.1.1.2.2	ENQUIRY message attributes	34
4.1.1.2.3	CONNECTION message attributes	35
4.1.2	Message Identifier.....	36
4.2	AGIE message management	36
4.2.1	STANDARD message management.....	36
4.2.2	COORDINATION message management.....	38
4.2.2.1	NOTIFICATION message management.....	38
4.2.2.2	ENQUIRY message management.....	38
4.2.2.3	CONNECTION message management.....	38
4.2.2.4	39
4.2.3	Connection Authentication.....	39
4.3	Message Attachments.....	39
5	AGIE Information Security.....	40
6	Appendices.....	41

1 Introduction

Aircraft communications is a critical element in the operations and safety of today's commercial airlines. This is becoming even more so as new aircraft being introduced into airlines' fleets, like the A-380 and B-787, involve increasingly more data intensive operations. Similarly, many airlines are incorporating electronic flight bags (EFBs), which typically also have large data requirements, into the operations of their existing legacy aircraft. This increase in the amount of operational data results in a corresponding increased demand on aircraft communications systems and their ability to handle the necessary data exchanges with the various onboard applications.

Today there can be many data communications paths to and from aircraft. These may typically involve many different communications media that could include, for example, VHF, HF, satellite, cellular, and Wi-Fi Gatelink. Some recent implementations in the satellite, cellular, and Wi-Fi Gatelink areas potentially have the capability to address the broadband communications requirements needed to help meet the increased data communications demands discussed above. These broadband capabilities are typically IP-based communications technologies. However, separate technology implementations within each of the media as well as between the media themselves currently require each aircraft application to specifically meet the unique interface requirements of each media communications path in order to use it.

It is desirable therefore to have a common aircraft communications interface that all onboard applications could use to access the appropriate media link to communicate with their ground complements. In addition, because the broadband throughput capability may not always be available along the entire path between the aircraft application and its complement on the ground (e.g., airline server), it is also desirable to allow a common store and forward capability to be implemented at the airport to address such throughput restrictions when they arise. This will help minimize the need for costly communication equipment on the ground that various users might find necessary to deploy to address throughput restrictions impacting their individual applications.

1.1 Purpose

The purpose of ARINC Project Paper 830 is to define a protocol for application-to-application information exchange between the aircraft and the airline ground infrastructure. It is intended for all types of IP communications including ground-based and satellite.

1.2 Scope

This document is intended to define the AGIE protocol to sufficient detail such that any party may develop a functional implementation of either all or part of the standard depending on the nature of the application.

1.3 Document Overview

This document is structured as follows:

- Section 2 provides a brief overview of the AGIE standard and the considerations driving this standard

ARINC Project Paper 830 AGIE Strawman Proposal

- Section 3 describes AGIE from a functional and operational aspect
- Section 4 describes the interface specifications along with message details
- Section 5 defines information security and other relating topics

Attachments/Appendices contain various figures, tables, and related information as necessary.

1.4 Related Documents

The high-level requirements for a messaging service application are addressed in ARINC Report 821, Aircraft Network Server System (NSS) Functional Definition. The ARINC 821 document serves as an umbrella document that identifies and describes the high-level requirements for various network services that are to have their detailed requirements developed in dedicated ARINC standards. Also, besides the high-level requirements, ARINC 821 itself discusses the general prerequisites and design considerations for aircraft network services as well as defines a set of services for management of network elements.

The other network services that have their high-level requirements delineated in ARINC 821 but have their detailed requirements developed in separate, dedicated standards include Avionics Interface Services (ARINC 834); routing services -- the Manager of Air/Ground Interface Connections [MAGIC] (ARINC 8xx); and messaging services -- Aircraft/Ground Information Exchange [AGIE] (ARINC 830). The latter is the subject of this document.

1.5 Document Precedence

TBD

1.6 Regulatory Approval

TBD

2 ARINC 830 “AGIE” Definition

2.1 AGIE Purpose/Objective

The aviation industry has spent significant effort defining onboard architectures and communication protocols to support the delivery of large amounts of aircraft information using the Internet Protocol (IP). For example, aircraft and applications have been developed to rely on wireless communication of these data uploads/downloads for efficient operation. AGIE’s intention is to define a common-use infrastructure such that every airline, airframe or third-party content provider does not need to maintain local servers for individual aircraft applications (such as EFB chart viewers, document viewers, electronic logbooks, flight data download systems, IFE content, etc.).

Therefore, the intention of the ARINC 830 “Aircraft/Ground Information Exchange (AGIE)” standard is to establish a rather simple, yet well defined and predictable, non-proprietary data interchange protocol and interfaces for application-to-application information exchange between aircraft applications and airline ground infrastructure using wired, wireless and optical technologies.

The specification defines a standard interface and functionality for ground applications to enable use of the “store and forward” process and its method of operation is analogous to the internet email systems and supports transfers for large data items.

AGIE further addresses authentication, authorization, transaction accounting, data integrity as well as information security and is not specific to a particular aircraft information domain.

Section 2.2 summarizes the high level requirements which the AGIE standard needs to address.

2.1.1 AGIE Goals

AGIE aims to establish a communication infrastructure within which any end-system application (aircraft or ground-based) can submit data to the AGIE data processing cloud for the purpose of having this data being automatically transported and made available for retrieval by the intended recipient end-system (also either aircraft or ground-based) application of this data without any of the end-system requiring any knowledge of the actual delivery mechanism.

Operationally this is analogous to a person depositing a letter in a mail box with the expectation that this letter is subsequently delivered by the postal service to a mail box from which the addressee will retrieve this letter at some later time.

While end-system applications need not have knowledge of *how* actual data transport is achieved:

- a) The sending system must know in what format data submittal is required and what information it must provide to assure successful data delivery
- b) The transport mechanism needs to know how data can be delivered to the intended recipient
- c) The system needs to maintain delivery status while delivery of data is still pending
- d) The receiver needs to know how to retrieve the data when available

Consequently, the AGIE definition includes two related yet operationally distinctly different types of standards:

- a) The interface specifications which define how end-system applications submit and retrieve data

ARINC Project Paper 830 AGIE Strawman Proposal

- b) The operational infrastructure which implements the actual transport of data between end-systems which those depend on

Additional goals for the AGIE standards definition must include:

- Minimize impact on existing business systems
- Minimize need for new development and unique/proprietary mechanisms
- Integrate end-to-end information security
- Design for long life cycle
- Be programming language, operating system and platform independent

2.2 AGIE High Level Requirements

The meet the AGIE purpose/objectives as top level requirements the ARINC 830 AGIE standard *shall*:

1. **Define a universal application to application messaging protocol through which aircraft installed systems can achieve reliable bi-directional information exchange with ground based systems**
2. **In addition to aircraft/ground communication, support application to application data exchange within the same aircraft or between ground-based applications**
3. **Support messaging data exchange paradigm (as opposed to streaming)**
4. **Define the data interface specifications as being independent from the data transfer mechanism used for the actual transport of data**
5. **Allow “store and forward” of data, i.e., allow the initiation of data transfers without requiring an active communication link between the aircraft and the ground being available at that time.**
6. **End-system applications are not required to be cognizant of the type of communication link used for aircraft/ground communication**
7. **Support a policy that allows the overall management of the message delivery and routing as defined by the operator.**
8. **Include means for actively managing the delivery of messages such as status tracking, retries, notifications, logging, etc.**
9. **Ensure operating system, development system and language independence.**
10. **Permit exchange of data of any type, e.g., binary, ASCII, Unicode**
11. **Permit exchange of messages of unlimited size, subject to the resource constraints of a particular implementation.**
12. **Provide means to prioritize data transport between applications and between messages within the same application.**
13. **Include a universal addressing mechanism through which all AGIE nodes (needs further definition) can be reached**

14. Support the secure data transfer of application data between AGIE end-points and systems.
15. Include modern information security provisions such as authentication, data integrity checking, etc.
16. Be capable of staging data in an AGIE server.

2.3 General Description

Consistent with high level requirements stated in Section 2.2 AGIE defines a client/server based request broker system where dedicated AGIE servers manage the data exchange and end system applications register as AGIE clients with the “nearest” AGIE server for the purpose of submitting data for transfer to and retrieve data from other AGIE clients.

AGIE clients submit data for delivery to other AGIE clients to their “nearest” AGIE server and the AGIE client for which data is intended in turn retrieves data from “its” nearest AGIE server.

AGIE defines a messaging standard where all data is exchanged as XML documents that may or may include attachments in some form for the purpose of transferring large data items.

AGIE further defines a flexible AGIE addressing scheme which operates similar to the email addressing convention although there are some distinct differences in order to account for the esoteric requirements for data transport in aerospace and aviation applications.

Owing to the analogy with email operation it is useful for the following terminology to be applied for AGIE as well:

1. Define the term “Mail/Message Transport/Transfer Agent (MTA) which takes the role of an AGIE Server
2. Define the term “Mail/Message User Agent (MUA) which takes the role of the AGIE client

Figure 1 provides a conceptual overview of the intended AGIE system architecture.

The architecture consists of two primary groups of components: aircraft and ground based components. Typically all actual transfer between the ground and aircraft occurs between AGIE servers only although the AGIE standard also allows direct link of aircraft clients with ground based AGIE servers.

The AGIE standard addresses the management of data exchange between AGIE clients only and does not address *how* data is exchanged between an aircraft based or a ground based server. Any available communication link technology may be employed to transfer data between an aircraft AGIE server and a ground based server. As such this exchange may occur via ARINC 839 “MAGIC” compliant transport mechanisms but AGIE does not specifically require this and any available means could be used for this purpose.

Depending on the type of aircraft/ground communication link being in a place for a particular system, this link may or may not be available at all time, e.g. if communication links are used which are only available while an aircraft is on the ground then during flight no communication is available with that aircraft. The AGIE standard provides for this situation.

An important aspect of AGIE is that a client need *not* be aware of

1. The type of aircraft/ground communication link that may be used

ARINC Project Paper 830 AGIE Strawman Proposal

2. If an aircraft/ground communication is active at the time a data transfer is initiated

An AGIE client can be any software application hosted either on an aircraft installed component or on a ground-based system. While typically data exchange occurs between aircraft installed and ground-based systems, AGIE does not stipulate this as a requirement. An aircraft resident AGIE client may also exchange messages with other aircraft resident AGIE clients and likewise ground-based AGIE clients can exchange messages with other ground-based AGIE clients.

As far as data traffic is concerned AGIE is a symmetric system, i.e. the same data exchange mechanics apply ground to aircraft and aircraft to ground data transfers.

DRAFT

ARINC Project Paper 830 AGIE Strawman Proposal

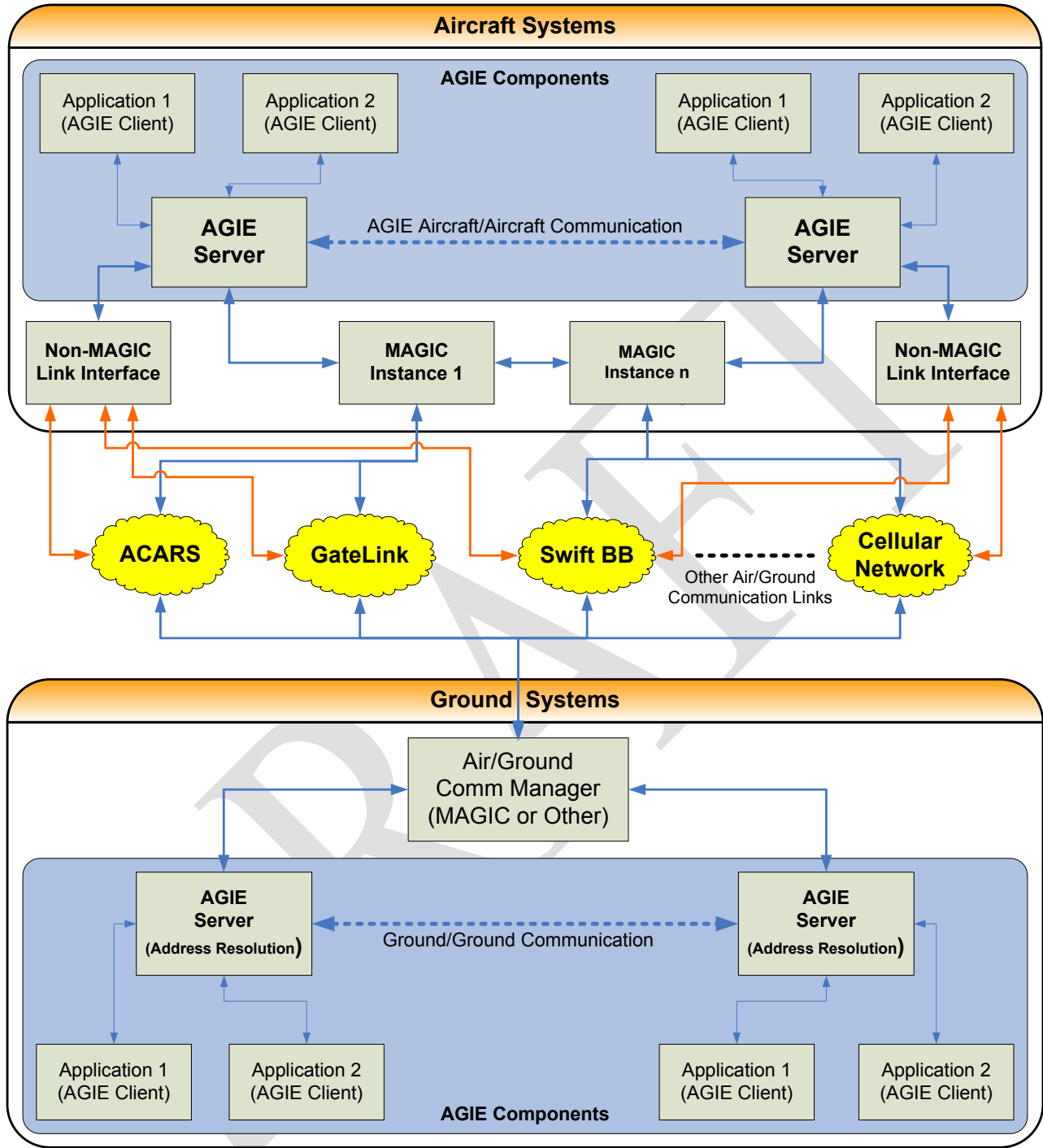


Figure 1 – Conceptual AGIE System Overview

2.4 High Level Functional/Interface Description

All AGIE data are transported as XML documents sent by AGIE Clients (or MUA) as messages to the “nearest” AGIE server (or MTA). It is the MTA’s responsibility to coordinate transferring the respective message to the AGIE Server/MTA through which the destination client/MUA retrieves that message.

Two separate levels of transport mechanisms exist as part of the AGIE standard:

- a) The interface between and AGIE client (MUA) and server (MTA)
- b) The exchange of data between two servers (MTAs)

The AGIE standard defines the following three (3) protocol choices to implement *client to server* communication (defined in Section 3.4.1):

- a) A SOAP based protocol over http
- b) Simple XML based protocol
- c) An AMQP based protocol

The implementer of an AGIE client has a choice as to which protocol is most suited for a particular application. However, an AGIE server MUST support all three to assure full interoperability.

For *server to server* communication the AGIE standard defines a minimum set of **still tbd #** protocols that MUST be supported by all AGIE server implementations, further described in Section 3.4.2. This minimum set was deliberately chosen with the goal to minimize new developments by implementers.

For the implementation of a particular server-server pair, however, a vendor may also choose the use of alternative protocols to possibly achieve a higher level of operational efficiency. Care must be taken that in such case interoperability is still assured.

For aircraft/ground information exchanges AGIE messages need to pass through some type of aircraft/ground communication link. This communication link may or may not be implemented through an ARINC 839 “MAGIC” system and may take on any form which in turn may or may not use proprietary technology..

Consequently, key to AGIE is the ability address all nodes in a consistent manner. To enable this AGIE adopts a mechanism which is very similar to conventional email addressing and is defined further in Section 0. The main challenge of addressing is to achieve a reliable mechanism which allows easy identification on how a destination server can be reached.

2.5 Policy and Quality of Service Considerations

AGIE is an application level protocol and primarily defines a common format and basic hand shake type protocol to permit applications implemented by various vendors and various systems to allow interchange of information with each other using each other’s services. As such policy and quality of service related consideration do not significantly (if at all) into the AGIE standards definition. Those types of considerations belong to the transport layer aspects of communication such as the ARINC 839 MAGIC standard.

2.6 Certification and Partitioning Considerations

TBD...

2.7 Assumptions/Constraints

TBD...

DRAFT

3 AGIE Functional Specification

This section defines the operation a functional level without defining details such as message structures. As such the following areas addressed:

1. Principles of operation
2. System topology
3. Link management
4. Addressing
5. Data delivery and status management
6. System configuration management

The content of this section extends the brief system description provided in Section 2.1 and needs to be viewed in context with the conceptual system architecture illustrated in Figure 1.

3.1 AGIE Principles of operation

The AGIE standard primary objective is to define a protocol for exchange of data between end systems/applications. To do so end system software applications register themselves as AGIE clients to an AGIE server which, from a network topology point of view, is nearest to each client.

In addition to clients registering with servers active AGIE links are also established between the AGIE servers themselves in order to establish full communication paths.

The registration process includes the establishment of an active communication link between those two components.

Note: The registration process does not constitute a network type authentication. AGIE makes the assumption that this type of authentication from an information security point of view is performed at a lower level. The registration process thus constitutes a link establishment only and is further addressed in Section 3.3.

Some AGIE clients may be registered with “their” server at all times and would only need to (re)register occasionally, e.g. in the event of aircraft power up sequence. An example of a “fixed” client is an airborne data loader, which is permanently installed in the aircraft.

Other clients may not be permanently registered to the same server. An example of this is a Class I Electronic Flight Bag (EFB). Class I EFB are typically removed from the cockpit by the flight crew after a flight and thus removed temporarily from the system. The EFB registers again as a client for another flight which may even be different aircraft which in turn uses a different server than before. It is also possible that a flight crew member registers with a ground based client, e.g. if he/she connects their EFB on the ground from within a flight operations facility or perhaps a hotel room. Nonetheless, all clients *must* to be addressable by all other AGIE system components once registered with one of the AGIE servers.

The following key requirements must be met to achieve this type of flexibility with AGIE:

- a) An AGIE system is a closed, primarily statically defined system
- b) AGIE uses an addressing mechanism which supports dynamic association of clients with servers

ARINC Project Paper 830 AGIE Strawman Proposal

- c) AGIE clients cannot be associated with aircraft tails alone but require association with some other entity such as specific LRUs on specific aircraft or even flight crew members.

These requirements are discussed in more detail in the following sections.

3.2 AGIE Topology

AGIE defines a request broker system where dedicated AGIE servers manage the data exchange and end system applications register as AGIE clients with the “nearest” AGIE server for the purpose of submitting data for transfer to and retrieve data from other AGIE clients.

This requires a flexible way for AGIE components to address each other and the respective addressing mechanics are defined in Section 3.5.

To allow flexible addressing of AGIE components, especially AGIE clients, an AGIE system **MUST** be a closed system in the following sense:

1. The comprehensive set of all clients and servers that make up the full system is known to all components at all time
2. The network topology between all AGIE servers is static

In this closed system the association between all servers is static. On the other hand the association between clients and servers can be dynamic such that any known client can register with any server and this association can change at any time. To assure full addressing capability is maintained the “new” server **MUST** make this new association available to all other servers in the system. This is further described in Section 3.5.2.

All clients **MUST** be uniquely identifiable within the system independent of associations with servers. In the event more than one instantiation of an identical application registers with a server it is necessary that each application instantiation is provided with its own unique identification. Typical scenarios of this are again EFBs or credit card authorization devices.

It will, of course, be necessary to amend the static system configuration from time to time. Such a change requires a controlled effort managed by authorized system administrators and **MUST** be rolled out globally across the system. AGIE defines special configuration notification messages for this purpose, see Section 4.

Part of the definition of a static system also includes a definition of which end-system is “allowed” to exchange data with which other end systems. This is accomplished by providing each such end system with a list of “known” clients.

An AGIE system needs to provide a central means for management of global system configuration and has this information available to all system components. To prevent the rollout of this information from becoming too complex, the AGIE standard states that this information is centrally managed on one or more ground-based servers which take the role of a Domain Name Server (DNS). All AGIE servers can enquire about the association between client and servers from DNS at all time. In the event redundant DNS are used, it is a requirement that those be kept synchronized. One of the servers is always the primary and all others the secondary servers. All AGIE servers must know how to reach the DNS nodes however. These DNS also provide the translation from AGIE node identifiers to actual IP addresses.

3.3 Link management

Within the framework of AGIE two types of communication links exist:

1. Link between an AGIE client and an AGIE server
2. Link between two AGIE servers

The initiation of link establishment is subject to the following rules/conventions:

1. A client always initiates a link with a server, i.e. this link is never initiated by the server
2. An aircraft based server always initiates a link with a ground-based server, i.e. a ground-based server cannot initiate a link with an aircraft based server
3. The precedence of link establishment between two aircraft based servers or two ground based servers is configuration defined and may be mutual, i.e. either could initiate the link
4. An AGIE client cannot establish a direct link with another AGIE client.

As AGIE is an application level (Layer 7) protocol AGIE links should be viewed as logical links rather than a physical communication link. For that reason one AGIE entity is said to be “registered” with another AGIE entity on an application level and this state indicates to each party that the ability to exchange information has been established. The actual physical link occurs outside the AGIE framework at a lower layer and is managed by the underlying protocol employed for actual transport of AGIE messages, described further in Section 3.4 “Protocol Binding”.

The “registration” process involves the exchange of specific coordination messages between two AGIE entities (defined further in Section **Error! Reference source not found.**). Within AGIE the exchange of such messages is the only mechanisms through which AGIE entities “know” that the link between the two is in fact operational.

AGIE relies on the underlying transport protocol to provide connection heartbeat messages. An AGIE implementation – irrespective of whether a server or client implementation – may therefore utilize the provisions of the underlying transport protocol to enquire about the status of a communication link.

The creation of an AGIE link, i.e. registration, is automatic and does not include any human action or any specific trigger by an application. However, in the event an application detects that the link has been interrupted the party which normally initiates the link **MUST** re-initiate the link.

AGIE does not define a specific “unregister” process. Once a link is established it remains active until interrupted in some “abrupt” fashion, e.g. through loss of power to an LRU or aircraft/ground link interruption. There is no requirement to first terminate the link from an application level.

A link is considered interrupted when the communication link is interrupted within the lower level data transport protocol.

3.4 Protocol Binding

While AGIE explicitly defines messages as XML documents along with a definitive set of AGIE message types (Section 4.1) to address all functional AGIE needs, AGIE offers some flexibility regarding the protocol binding to achieve the actual transport of the actual XML documents.

The AGIE standard specifies different protocol bindings for:

- Client to Server Communication
- Server to Server Communication.

As such the protocol requirements are different for an AGIE client compared to an AGIE server. A client is not required to support all defined protocols but a server **MUST** support all AGIE defined protocols to assure interoperability.

In addition, for server to server communication an implementer may choose to employ additional protocols to be used between two AGIE entities implemented by the same vendor for example to achieve a higher level of optimization. Nonetheless, the minimum set of protocols must also be supported to

Irrespective of the actual protocol implementation any AGIE implementation must support all message types and the actual message content is standard.

3.4.1 Client/Server Protocol Binding

AGIE defines three (3) distinct choices for the protocol used between an AGIE client and an AGIE server, i.e. based on:

1. Simple Object Access Protocol (SOAP) over http
2. Simple XML based protocol similar to the protocol defined as part of the ARINC 840 standard
3. Advanced Message Queuing Protocol (AMQP)

Each protocol is outlined in the subsequent sections and also described in further detail in the Appendix.

A particular AGIE client application has a choice as to which protocol it should implement. However, a server implementation **MUST** implement all (three) protocol with the scope of implementation as required to support this standard. Consequently, any AGIE server implementation **MUST** implement listeners for all three protocols to permit servicing a link request through each of those to assure full interoperability.

As an AGIE client *always* initiates the communication between a client and a server the latter will always know through which protocol the link was established and all subsequent communication continues to use the same protocol.

An AGIE client **MUST** implement at least one of those three protocols to assure interoperability.

3.4.1.1 SOAP over http Binding

Description of SOAP Binding goes here – needs to include reference to the actual message content, addressing and message types where applicable...

3.4.1.2 Simple XML Binding

Description of Simple XML Binding goes here – needs to include reference to the actual message content, addressing and message types where applicable...

3.4.1.3 AMQP Binding

Description of AMQP Binding goes here – needs to include reference to the actual message content, addressing and message types where applicable...

3.4.2 Server/Server Protocol Binding

The AGIE standard requires that an AGIE server **MUST** be able to initiate a link with another AGIE server at a minimum via the use of the following minimal set of protocols:

List of protocols goes here, e.g. FTP...

The list of protocols was chosen deliberately to minimize any risk for unnecessary protocol development. In addition to the minimum standard server-server protocol a particular implementation **MAY** implement proprietary protocol that may be more efficient. This is likely only practical if both respective server implementations were developed by the same vendor.

Note: The use of proprietary protocols results in loss of interoperability. Therefore, the AGIE standard requires that the standard protocols to always be supported.

3.4.2.1 Server/Server Protocol 1...

Tbd...

3.4.2.2 Server/Server Protocol 2...

Tbd...

3.5 AGIE Addressing

Addressing is a key element of the AGIE standard and **MUST** in particular accommodate the requirement to that the association between AGIE clients and AGIE server is not necessarily static. Electronic Flight Bags (EFB) are particular good examples of AGIE clients that can regularly change association with AGIE servers. For example, the same EFB may be registered with an AGIE server on one aircraft, followed by a registration with a ground based AGIE server while the pilot resides in a hotel and subsequently be associated with an aircraft based AGIE server for the next flight.

It is therefore necessary for an AGIE system to centrally maintain a database containing the following information for the entire AGIE system:

1. List of all defined AGIE clients
2. List of all defined AGIE servers

ARINC Project Paper 830 AGIE Strawman Proposal

3. List of instantaneous association between client and servers

To assure that all AGIE nodes can access this central database it needs to be a ground based system. To do so at least one ground based AGIE server MUST be configured to perform DNS type services in addition of all other AGIE server functions.

All aircraft/ground and ground/ground data exchange are automatically routed through this central server for subsequent message routing based on the destination address.

It is, however, not practical for client to client data exchange within the *same* aircraft to require the routing messages to a ground based system first. Therefore, an aircraft based AGIE server may also take the role of a central DNS but with limited scope managing information for that particular aircraft only. Should a message be received whose destination is also an AGIE entity on the same aircraft then this message can be routed within the aircraft accordingly without requiring aircraft/ground communication.

To permit adequate DNS type service an AGIE server is identified via the following minimum set of attributes and maintained as an XML document:

- SERVER_ID – specifies the formal server identifier/name given by the system integrator
- IP_ADDRESS – fixed IP address of the server
- SERVER_INSTALL – an indicator if server is aircraft or ground based may take one of the following two values: AIRCRAFT or GROUND
- SERVER_LOCATION – defines the physical location of the server and takes the following type of values:
 - Aircraft identifier (tail number of similar) if aircraft based, e.g. D-AGIE, F-AGIE or N1234A
 - Geographic location if ground based, e.g. Chicago, Paris or Frankfurt

Note: the aircraft identifier is required for aircraft servers but may be omitted for ground based servers.

- AGIE_DNS – indicates if the server performs the role of an AGIE DNS server with values as YES or NO
- CLIENT_LIST – list of formal client identifiers currently registered with this server (a server may have zero or more clients registered at any time)

It is paramount that this information be kept up to date to the best extent possible. Any time an AGIE server experiences a change in client/server association this server is required to send a coordination message to the DNS AGIE node providing the necessary updated information. This applies to the central ground based DNS as well as any aircraft based DNS AGIE nodes that perform this type of function.

3.5.1 The AGIE Address

AGIE defines two different types of addresses:

1. AGIE Client Addresses
2. AGIE Server Addresses

ARINC Project Paper 830 AGIE Strawman Proposal

The identifiers contained within the address be composed of any characters that are permissible for use with conventional email and internet addresses.

3.5.1.1 AGIE Client Address

A standard AGIE *client* address follows the same general convention as standard email addresses and takes the general form:

`<client_identifier>@agie.<airline_domain>`

The “agie” token immediately to the *right* of the “@” symbol is required and serves as an indicator that the address represents an AGIE address rather than a conventional email address.

The token `<airline_domain>` is the nominal internet domain that an airline is registered with, e.g. “united.com”, “airfrance.com” or “lufthansa.com” etc.

Examples of a full AGIE client address are:

efb.fo@agie.lufthansa.com	(First Officer EFB AGIE address)
adl@agie.united.com	(Airborne Data Loader AGIE address)
flight_ops@agie.airfrance.com	(Ground based flight operations AGIE address)

For messages to be sent within the same domain (airline) it is not necessary for a sending client to specify the full address as the domain is assumed. An AGIE server automatically expands the address to the full definition. Only in the event a message needs to be sent to a different domain the right hand side of the address is necessary.

An AGIE client identifier may be a node, may be a particular application instance or even be a physical person. AGIE intentionally does not make any restrictions on how client identifiers are defined and this is left to the discretion of the system implementer.

Moreover, the AGIE standard does not stipulate how `<client_identifiers>` are constructed and this is also up to the implementer to define a suitable naming convention.

3.5.1.2 AGIE Server Address

The addressing of a specific server rather than an AGIE client is required for the exchange of coordination messages between AGIE servers. To differentiate a client address from a server address a standard AGIE *server* address takes the general form:

`<server_identifier>.agie.<airline_domain>`

Note the absence of the “@” symbol, which signifies that this address is a server address and not a client address.

The “agie” token is again required to identify the address as an AGIE address and the `<airline_domain>` is the same as for AGIE client addresses.

The `<server_identifier>` is the formal name of any AGIE server node as defined by the attributes defined for each server. Unlike AGIE clients AGIE servers **MUST** always use the full address to assure that addresses are interpreted correctly.

3.5.2 Address Resolution

The following types of principal address resolution scenarios must be supported by AGIE:

1. An aircraft based client sends a message to another client registered with the same server
2. An aircraft based client sends a message to another client registered with a different server on the same aircraft
3. An aircraft based client sends a message off the aircraft (either to another aircraft or a ground based client)
4. A ground based client sends a message to an aircraft based client
5. A ground based client sends a message to another ground based client registered with the same server
6. A ground based client sends a message to another ground based client registered with a different server

Address resolution is the process of determining how to “find” the destination server. Actual routing is then performed in the conventional way.

3.5.2.1 Address resolution sequence

The following steps define the standard sequence of resolving AGIE addresses for messages originating from an aircraft or from a ground based component:

1. A client sends a message intended for another client to its server (i.e. the server it is registered with) – this step does not apply if the message originates from a server itself
2. The sending server checks if the receiver client is also registered with that server (which may in this case also be the DNS node) – if so the message is routed directly to the destination client
3. If the receiving client is not registered with the sending client’s server also then the message is routed to the applicable DNS node:
 1. If the sender resides on an aircraft then the message is routed to the respective aircraft DNS node which then determines if the receiver is on the same aircraft – if so the message is routed accordingly directly to the destination client without requiring ground communication
 2. If the sender resides on an aircraft and DNS determines that the destination client is not on the aircraft the message is routed to the ground based DNS for further routing
 3. If the sender is a ground based client the message is routed to the ground DNS for further routing.

In the event an address cannot be resolved the sending server is provided with a non-deliverable notification.

The need and use of a DNS type service by certain AGIE servers as already described at the beginning of Section 3.5 as required to perform full address resolution.

To assure that DNS data is always up to date it is necessary for any AGIE server (aircraft or ground based) to “report” any change in client/server association to the central DNS node by sending the respective CONFIGURATION COORDINATION message (see Section 4.2). If the change occurs

within an aircraft the aircraft based DNS as well as the central ground DNS node are informed. If the change occurs on the ground on the central ground based DNS needs to be informed. This may be accomplished by sending a complete client list to this server or only a change to that list. The DNS will update its internal records accordingly. Such changes in association would only occur on a regular basis for portable devices but not for stationary devices.

If at any time it is determined that the address path was incorrect but the destination address was resolvable anyway the sender as well as the DNS are notified accordingly. Likewise the original “incorrect” address is replaced by the “correct” address.

3.6 AGIE data delivery management

AGIE is a “store and forward” protocol. All end-system related “payload” type data are sent strictly from one AGIE *client* to another AGIE *client* with no exception. To initiate a transfer of data to a destination client the originator client sends an XML document as a message to the server it is currently registered with using the protocol the client is registered with. This server locally stores the information until it is able to transfer this message to the either the destination server to which the recipient AGIE client is registered or to the DNS otherwise, i.e. whenever a communication link is established.

Within the AGIE standard definition only two logical servers exist:

- a) the ORIGINATOR server from which a message originates after being submitted by a client
- b) the DESTINATION server which “keeps” the message until it can be delivered to the destination client

While actual data paths may require multiple physical “server hops” only the ORIGINATOR and the DESTINATION server are of relevance for an AGIE data transfer. The ORIGINATOR and DESTINATION server may in fact be the same AGIE server instantiation.

Data transfer between those two “end” servers is managed via standard TCP/IP protocol in conjunction with any other aircraft/ground link infrastructure such as MAGIC.

The recipient server locally stores the information until it is able to deliver the data to the final destination.

AGIE defines two principal types of messages, described in detail in Section 3.6.2.2:

1. STANDARD messages used to transfer end-system related data between AGIE clients
2. COORDINATION messages used to communicate any internal system management related information such as notifications, status enquires, connection coordination etc.

“Payload” data are communicated via STANDARD messages. For small data items the data can be encapsulated as part of the AGIE XML document body while for large volume transfers such data can be transferred as attachments to AGIE messages.

Each message is assigned a unique identifier which is a combination of the sender identifier plus an integer number which is unique to the server from which the message originates at the time the message is sent. Such integer may be re-used provided at the time of sending this number is unique.

The delivery of data may take two forms depending on the message type:

ARINC Project Paper 830 AGIE Strawman Proposal

1. The server notifies the client of the data and the client subsequently requests the data to be transferred to the client – this is the normal delivery mode for STANDARD messages
2. The server can also “push” the data directly to a client when a link is established without prior explicit notification if supported by the chosen protocol – this the normal delivery mode for COORDINATION messages and is also used for URGENT STANDARD messages for delivery of urgent messages or data which may need to be processed in near real-time; however, this mode *should* be used sparingly for STANDARD messages

In addition, the client may poll a server for availability of data. When a client has received indication that data are available for this data the client asked the data to be transferred via a dedicated AGIE message.

AGIE also defines the capability to broadcast messages to all clients. Although operationally this is primarily intended for ground-based applications to broadcast to aircraft clients the standard does not impose this restriction explicitly. The delivery mechanism for broadcast messages is identical to all other STANDARD messages.

The exchange of data via AGIE is completely symmetrical, i.e. the same services apply for aircraft to ground as well as for ground to aircraft messaging.

AGIE does not define how data are locally stored at any server location as part of the “store and forward” operation.

3.6.1 Data transfer prioritization

Within the framework of AGIE only two levels of data priority are defined and they are “NORMAL” and “URGENT”. The vast majority of traffic is expected to be managed as NORMAL traffic.

As a rule all data are strictly queued and actioned on a First-In First-Out (FIFO) basis irrespective of their origin and/or destinations. With respect to NORMAL versus URGENT messages the following general rules apply:

1. URGENT messages take precedence over NORMAL messages
2. Newly appearing URGENT message generally are placed ahead of the queue
3. URGENT messages are queued in FIFO order with respect to each other
4. URGENT can suspend the processing of a NORMAL message provided the URGENT message is of less than some maximum parameter defined size; otherwise transfer of message in progress continues until completion and the new URGENT message will become the next one to be serviced.

Note: this could in principle lead to a situation where no NORMAL messages are ever processed and hence the use of URGENT message should be used sparingly

The AGIE standard does not stipulate how many queues a particular server (or client) may maintain, e.g. it may be possible that a server maintains a separate queue for each client that is registered with that server. Likewise, there may be separate queues to manage more than one link to other AGIE servers.

The prioritization rules however apply to each such queue individually.

ARINC Project Paper 830 AGIE Strawman Proposal

The use of “URGENT” traffic should be restricted to cases where near real-time transaction processing is desired and/or required or in the event important information is to be delivered to clients. Moreover, “URGENT” messages are recommended to only be used for small data items.

The AGIE definition provides for a system administrator to define maximum data sizes to apply to URGENT data. Should such a maximum data size be defined for a system, then, in the event that the size of a particular data transaction exceeds this value, such transfers are actioned as “normal” only regardless as to whether this transaction is designated as “urgent” or “normal”. Moreover, if a maximum “urgent” data size is defined then this maximum applies globally to the entire system.

Data transactions designated as “urgent” are always “pushed” by the server to any registered client. In the event the destination client of an “urgent” message is not registered to the server, which holds this message, then this server needs to deliver this message as soon as this client registers with that server, provided the latest delivery date has not yet expired. The administrator will always have the option to choose this parameter sufficiently large to allow most data sizes to be treated as “URGENT” should this be necessary.

“Normal” is the default priority attribute value for all transactions.

Any additional level of prioritization is either managed on an end-system application level or within lower level transport layer, such as QoS or any prioritization rules implemented by MAGIC (ARINC 839 standard).

3.6.2 Delivery status management

All AGIE servers *must* track the status of a message until successful delivery or a non-delivery status is determined.

A sending client can enquire about the status of a message at any time while this message is in transit. This is done through special ENQUIRY messages (see Section 4.1).

AGIE must also retain a history of all transactions. The retention period thereof is likely to be largely defined based on regulatory and/or operational considerations and are in any case configuration defined. In general, for aircraft based components the retention period is typically short, e.g. several flight legs or perhaps several days. In contrast for ground based components the retention period may be much longer, e.g. years, especially for ground based servers.

AGIE does not define actual such retention time values, but does define as a requirement that transaction log retention periods are definable with different parameters definable for each of:

- Aircraft based clients
- Aircraft based servers
- Ground based servers
- Ground based clients

3.6.2.1 Delivery Date/Time Parameters

To manage delivery and associated notifications the AGIE standard defines the following message delivery related date/time parameters which are passed as part of AGIE messages and serves various purposes:

ARINC Project Paper 830 AGIE Strawman Proposal

1. Origin Date/Time
2. Delivery Date/Time
3. Creation Date/Time
4. Effectivity Date/Time
5. Expiry Date/Time
6. Latest Delivery Date/Time
7. Resubmission Date/Time

The first two parameters serve informational purpose only and may be used for purposes such as transaction logging etc. Parameters 3-5 are mainly of interest to the applications that consume the information, while the last two attributes factor into the delivery mechanism of data itself. Table 1 summarizes these attributes including their intended purpose/use.

Table 1 – AGIE Date/Time Attributes

Date/Time Attribute	Attribute Description/Definition	Used By/For
Origin Date/Time	Date/time stamp set at the moment a message is sent by the message originator.	By: Any message consumer For: Maintain internal reference of receipt and for possible receipt confirmation
Delivery Date/Time	Date/time stamp set at the moment a message has been successfully received by the intended recipient of a message. This attribute is not part of the original message and is only used for receipt notification as well as event logging purposes.	By: Any message consumer For: Maintain internal reference of receipt and for possible receipt confirmation
Creation Date/Time	Date/time at which sent data was generated (which may not be the same as "Origin_Date/Time). This attribute is generally more of relevance to attachments rather than directly encapsulated payload data.	By: Destination application For: Informational purpose
Effectivity Date/Time	Informs consumer of message of the date/time at which the payload information of a message becomes of relevance to the receiver application(s).	By: Destination application For: Establish start of data validity period

ARINC Project Paper 830 AGIE Strawman Proposal

Date/Time Attribute	Attribute Description/Definition	Used By/For
Expiry Date/Time	Defines date/time at which the payload information of a message is no longer of relevance to the receiver application(s). This attribute may be used to determine if further attempts to deliver this message may be stopped.	By: <ul style="list-style-type: none"> • Destination application • AGIE servers • Message originator For: <ul style="list-style-type: none"> • Determine if stopping delivery attempts • Provide notification to message originator • Generate alert to system operator
Latest Delivery Date/Time	Defines date/time at which a message needs to reach the intended destination at the latest. Inability to deliver data by this date/time results in alerts being generated to the system operators. This is a compulsory message attribute.	By: <ul style="list-style-type: none"> • AGIE servers • Message originator For: <ul style="list-style-type: none"> • Provide notification to message originator • Generate alert to system operator
Resubmission Date/Time	Defines an intermediate date/time value which may be used to inform the originator of a message of the inability to delivery some time prior to the Latest Delivery Date/Time date/time so that the originator may resubmit this message at its discretion. This is an optional message attribute.	By: <ul style="list-style-type: none"> • AGIE servers • Message originator For: <ul style="list-style-type: none"> • Server provides notification to message originator • Originator may resubmit message/data upon notification

3.6.2.2 *Inability to deliver*

In the event a delivery path cannot be determined, a client does not accept data which is pushed by the server or alternatively the client does not retrieve data from the server by the time the **Latest Delivery Date/Time**, **Expiry Date/Time** or **Resubmission Date/Time** has expired (see Section 3.6.2.1), the server sends a non-delivery notification to the originator of the message. If delivery cannot be made by either **Latest Delivery Date/Time** or **Expiry Date/Time** some type of alert needs to be presented to a system operator representative (e.g. system administrator) on the ground of the inability to deliver.

3.7 *AGIE system configuration management*

Aircraft are identified by unique identifiers, with which an air operator registers its fleet. This is typically the aircraft registration but it is also possible to use other means of identification provided all aircraft within the fleet can be uniquely identified within the fleet.

ARINC Project Paper 830 AGIE Strawman Proposal

AGIE only works within this closed system. Any communication to external systems must occur via one of the AGIE clients and this would typically apply primarily to ground based systems.

Also, in the event of a configuration update being received the AGIE server will push the new configuration to all the clients which are connected at that time.

At any time a client reconnects with a server the server provides, as part of the acknowledgement of the connection, the meta data of the latest configuration.

Likewise, whenever an aircraft resident server must reconcile with a ground based server that it does in fact have the latest configuration parameters at hand. Any server will keep two copies of the system configuration: the most recent one and the one preceding it.

There is a static server configuration definition file and a static client definition file. Both files are read-only XML files which can only be amended by an authorized system administrator.

On the other hand an AGIE client can connect and disconnect itself from the network at any time. However, all possible AGIE clients must be known to everyone else in the system. The dynamic aspect is to associate any known clients with a known server and this can change dynamically.

The AGIE servers must dynamically exchange information regarding which client is associated which server at any given time. The purpose of this is to allow clients to be moved dynamically such as Class I EFBs. A possible scenario is a flight crew member connects to a ground-based AGIE server in a hotel room then removes the EFB from the network and reconnects after he/she has entered the cockpit and reconnected to another server on the aircraft.

4 AGIE Interfaces and Protocols

4.1 AGIE message definitions

All AGIE data transactions involve the exchange of special AGIE XML documents as messages and messages can be exchanged between two clients, two servers or between a client and a server. However, true “payload” type messages occur between clients only and messaging between servers is used for system management purposes only.

AGIE defines two primary message *classes*:

- a) **STANDARD** messages which are used for exchange of end-system specific information between two AGIE clients only
- b) **COORDINATION** messages which represent all other non end-user “payload” related data communication and are categorized as one of the following (further described in Section 4.1.1.1):
 - NOTIFICATION messages
 - ENQUIRY messages
 - CONNECTION (management) messages

Both the STANDARD and COORDINATION messages have each standard set of attributes as described in Section 4.1.1. Some of those attributes are shared between the two message classes.

Each message requires a specific and unique **MsgID**. The value is auto-assigned by the AGIE node (client or server) from which the message originates. The respective convention for establishing the value this identifier is defined in Section 4.1.1.2.

STANDARD messages do not require any additional message type classification other than the message priority. COORDINATION class messages, however, require further type classification is defined to permit establishment of specific COORDINATION message purposes, see Section 4.1.1.2.

The AGIE standard does not define specific methods for data exchange. Instead, all AGIE messaging strictly consists of one node (server or client) sending an (XML) AGIE document to another AGIE node with the XML documents themselves containing all necessary information that may need to be conveyed to another AGIE node.

The use and inter relationship of and between COORDINATION messages is described in Section 4.2.

4.1.1 AGIE message attributes

The pre-defined set of attributes for STANDARD messages is defined in Section 4.1.1.1 and the pre-defined set of attributes for all COORDINATION messages is defined in Section 4.1.1.2.

The set of attributes are intended to be sufficiently comprehensive as part of the AGIE XML message document itself to permit complete management of the messages across an entire AGIE system.

Some attributes are compulsory for all messages, while some are required for certain types of messages only. Attributes that have default values assigned are interpreted consistent with those return values if the attribute is not part of the message itself.

ARINC Project Paper 830 AGIE Strawman Proposal

Note that for all text based message fields all leading or trailing white spaces of the content are ignored when interpreted. This allows any AGIE client or server implementation to trim such content accordingly.

Default values are assigned automatically if not explicitly provided by the sender of a message.

4.1.1.1 STANDARD Message Attributes

For STANDARD messages the **MsgClass** value is always **STANDARD** and its purpose is to convey end-system/application data between AGIE clients. Table 2 defines all attributes of a STANDARD messages that are contained within the XML message document.

Table 2 – AGIE STANDARD Message Attributes

Message Attribute	Attribute Description/Definition	Value Profile
MsgID	Unique message identifier assigned by originator of the message – this id is unique from the originators point of view	Auto assigned as defined in Section 4.1.1.2 Compulsory: Yes Default value: <i>none</i>
MsgClass	Defines if this is a “payload” or a coordination type message	For STANDARD message this value is always STANDARD Compulsory: Yes
MsgType	Specifies the priority of the message	Can take one of the following values: <ul style="list-style-type: none"> • NORMAL (default) • URGENT (for high priority delivery)
Description	Optional text field intended to provide a short subject line similar to an email subject line. This field may be used to provide a brief description for attachments or instructions relating to encryption of the associated data	Free Text Field up to 256 bytes in length Compulsory: No – but its use is encouraged Default value: <i>empty string</i>
DestinationClientAddress	Compulsory attribute defining the AGIE address(s) of the final destination of the message For multiple addresses this is an array of addresses all of which are separated from each other using the “;” (semicolon) symbol.	AGIE Client Address as defined in Section 3.5.1.1 Compulsory: Yes Default Value: <i>none</i>
SourceClientAddress	Compulsory attribute defining the AGIE address of sender of message	AGIE Client Address as defined in Section 3.5.1.1 Compulsory: Yes Default Value: <i>none</i>

ARINC Project Paper 830 AGIE Strawman Proposal

Message Attribute	Attribute Description/Definition	Value Profile
NextServerAddress	Address of the next server a message is to be routed to.	<p>AGIE Server Address as defined in Section 3.5.1.2</p> <p>Compulsory: No – when sent by clients Yes – when being forwarded from one server to another server</p> <p>Default Value: <i>none</i></p>
LastServerAddress	Address of the most recent server that is holding this message.	<p>AGIE Server Address as defined in Section 3.5.1.2</p> <p>Compulsory: No – when sent by clients Yes – when being forwarded from one server to another server</p> <p>Default Value: <i>none</i></p>
Origin_DateTime	As per Table 1	<p>Single field using the following format: <YYYYMMDD-hh:mm:ss></p> <p>Compulsory: Yes Default Value: <i>none</i></p>
Delivery_DateTime	As per Table 1	<p>Single field using the following format: <YYYYMMDD-hh:mm:ss></p> <p>Compulsory: Yes Default Value: <i>none</i></p>
Creation_DateTime	As per Table 1	<p>Single field using the following format: <YYYYMMDD-hh:mm:ss></p> <p>Compulsory: No Default Value: <i>none</i></p>
Effectivity_DateTime	As per Table 1	<p>Single field using the following format: <YYYYMMDD-hh:mm:ss></p> <p>Compulsory: No Default Value: <i>none</i></p>
Expiry_DateTime	As per Table 1	<p>Single field using the following format: <YYYYMMDD-hh:mm:ss></p> <p>Compulsory: No Default Value: <i>none</i></p>

ARINC Project Paper 830 AGIE Strawman Proposal

Message Attribute	Attribute Description/Definition	Value Profile
LatestDelivery_DateTime	As per Table 1	<p>Single field using the following format: <YYYYMMDD-hh:mm:ss></p> <p>Compulsory: Yes Default Value: <i>none</i></p>
Resubmission_DateTime	As per Table 1	<p>Single field using the following format: <YYYYMMDD-hh:mm:ss></p> <p>Compulsory: No Default Value: <i>none</i></p>
ReturnReceipt	Specifies if the receiver needs to send a special message to an originator acknowledging the receipt of the message. The return receipt is a NOTIFICATION message (see Section 4.1.1.2.1)	<p>Can take one of the following values:</p> <ul style="list-style-type: none"> • NO • YES <p>Compulsory: No Default value: NO (assumed if field is not present)</p>
MsgContent	<p>This attribute contains the message payload if not provided through an attachment. This is a compulsory attribute but may be of zero length.</p> <p>How the data is encoded is defined through the ContentType attribute.</p> <p>Additional data can be provided through Attachments (see below).</p>	<p>Free Field Text of arbitrary length but recommended to not exceed 2048 bytes.</p> <p>Compulsory: Yes Default Value: <i>empty string</i></p>
ContentType	Describes how the information in MsgContent is encoded.	<p>Can take the following values:</p> <p>ASCII UNICODE BINARY</p> <p>Compulsory: Yes if not ASCII Default: ASCII</p>
Attachments	<p>Array of values describing the size (in bytes) of the attached files as well as the respective path (URL) through which the files can be retrieved.</p> <p>This attribute is not mandatory and if omitted attachments are presumed to not be present.</p>	<p>Array of two values, the first defines the size of the attachment and the second the path/url through which the attached data can be accessed.</p> <p>If omitted no attachments are expected</p> <p>Compulsory: No Default Value: <0, ""> i.e. zero and blank string</p>

At this point a good example of a full AGIE message needs to be provided - tbd

4.1.1.2 COORDINATION message attributes

This section describes the various **types** of messages defined within the AGIE standard that serve the purpose of coordinating communication between AGIE nodes. Table 3 lists the pre-defined attributes of a COORDINATION message XML document.

Table 3 – AGIE COORDINATION Message Attributes

Message Attribute	Attribute Description/Definition	Value Profile
MsgID	Unique message identifier assigned by originator of the message – this id is unique from the originators point of view	Auto assigned as defined in Section 4.1.1.2 Compulsory: Yes Default value: <i>None</i>
MsgClass	Defines if this is a “payload” or a coordination type message	For COORDINATION messages can take one of the following values: <ul style="list-style-type: none"> • NOTIFICATION • ENQUIRY • CONNECTION Compulsory: Yes Default value: <i>None</i>
MsgType	Future specifies the nature of a particular message	Possible values depend on MsgClass value – see Sections 4.1.1.2.1 through 4.1.1.2.3
Description	Optional text field that may contain relevant information regarding a particular message type.	Free Text Field up to 256 bytes in length <i>(length limit required?)</i> Compulsory: No – but encouraged Default value: <i>empty string</i>
MsgReference	MsgID reference required by many COORDINATION type messages to which the latter applies	Same format as MsgID Compulsory: required for most COORDINATION messages Default value: <i>none</i>

ARINC Project Paper 830 AGIE Strawman Proposal

Message Attribute	Attribute Description/Definition	Value Profile
DestinationAddress	<p>Compulsory attribute defining the AGIE address of the final destination of the message which could be client or a server</p> <p>This may be an array of addresses all of which are separated from each other using the “;” (semicolon) symbol.</p>	<p>Format as defined in Section 3.5.1.1 or Section 3.5.1.2</p> <p>Compulsory: Yes Default Value: <i>none</i></p>
SourceAddress	<p>Compulsory attribute defining the AGIE address of original sender of a message which could be client or a server</p>	<p>Format as defined in Section 3.5.1.1 or Section 3.5.1.2</p> <p>Compulsory: Yes Default Value: <i>none</i></p>
NextServerAddress	<p>Address of the next server a message is to be routed to.</p>	<p>AGIE Server Address as defined in Section 3.5.1.2</p> <p>Compulsory: No – when sent by clients Yes – when being forwarded from one server to another server</p> <p>Default Value: <i>none</i></p>
LastServerAddress	<p>Address of the most recent server that is holding this message.</p>	<p>AGIE Server Address as defined in Section 3.5.1.2</p> <p>Compulsory: No – when sent by clients Yes – when being forwarded from one server to another server</p> <p>Default Value: <i>none</i></p>
Msg_DateTime	<p>Date/Time stamp at which the message was created.</p>	<p>Single field using the following format: <YYYYMMDD-hh:mm:ss></p> <p>Compulsory: Yes Default Value: <i>none</i> – auto assigned</p>
MsgContent	<p>This attribute is used for some COORDINATION message to convey relevant information and its format depends on the purpose – see Section 4.1.1.2.1 through 4.1.1.2.3</p>	<p>Free Field Text of arbitrary length but recommended to not exceed 2048 bytes.</p> <p>Compulsory: Yes Default Value: <i>empty string</i></p>

The different types of COORDINATION messages are described further below.

ARINC Project Paper 830 AGIE Strawman Proposal

4.1.1.2.1 NOTIFICATION message attributes

The purpose of this message is to provide an *unsolicited* notification of some type from one AGIE node to another, which can be server to client or server to server (typically). NOTIFICATION messages do *not* generate a response from the recipient.

For NOTIFICATION message the **MsgClass** value is “NOTIFICATION”.

Table 4 – Predefined NOTIFICATION message values

MsgType	Purpose	MsgReference	MsgContent
DATA_AVAILABLE	A server notifies a client that data are available for retrieval for that client	<i>Not used</i>	Array of MsgReference values that is available for retrieval by a client Compulsory: Yes Default Value: <i>none</i>
FETCH_DATA	Client instructs the server to send one or more messages listed in the MsgContent field of the DATA_AVAILABLE message that precedes this message. Server then sends the actual STANDARD messages	<i>Not used</i>	Array of MsgReference values that is available for retrieval by a client Compulsory: Yes Default Value: <i>none</i>
ACKNOWLEDGE	General purpose message to acknowledge the receipt of any other message irrespective of the type of message or which sender	MsgID of message being acknowledged	<i>Not used</i>
CONFIG_UPDATE	Notifies recipient that of any change in system configuration	<i>Not used</i>	Includes relevant information about system configuration changes such as adding or removing AGIE addresses – further described in Section NN . <i>(not entirely clear if this is required)</i>
MSG_RECALL	Instructs AGIE servers to no longer attempt delivery of an already sent message. This essentially cancels a message	MsgID of message being recalled	<i>Not used</i>
MSG_RESUBMIT	Send by a server after expiry of Resubmission_DateTime	MsgID of message to be resubmitted	<i>Not used</i>

ARINC Project Paper 830 AGIE Strawman Proposal

MsgType	Purpose	MsgReference	MsgContent
ALERT	Notifies a node of a certain event or that an expected event has not taken place within a predefined period of time	MsgID of message being referenced	Possible values are: <ul style="list-style-type: none"> ▪ NON_DELIVERY – inability to deliver a message by requested time ▪ LATE_DELIVERY – inability to deliver a message in time ▪ INCORRECT_ADDRESS – addressee not found ▪ LINK_DOWN – sent if aircraft/ground link has not been available for a specified period of time; used for server to clients if messages are pending for delivery between aircraft and ground (either direction) Others?

4.1.1.2.2 ENQUIRY message attributes

The purpose of ENQUIRY messages is for one AGIE node to enquire about a particular piece of information. Unlike NOTIFICATION messages ENQUIRY messages require a response by the recipient such that the **MsgType** value always come in pairs as described in Table 5.

(so far only three types of enquiry are identified but there may be more)

Table 5 – Predefined ENQUIRY message values

MsgType	Purpose	MsgReference	MsgContent
DELIVERY_CHECK	Sent by any node to enquire about delivery status of a particular message	MsgID of message being referenced	<i>Not used</i>
DELIVERY_STATUS	Response to a DELIVERY_CHECK enquiry	MsgID of preceding ADDRESS_CHECK message	Possible values are one of: <ul style="list-style-type: none"> ▪ DELIVERED – if at final destination ▪ IN_TRANSIT – if not yet at final destination ▪ FAILED – if delivery was not feasible ▪ RECALLED – if a message was recalled by the sender Plus the following: AGIE address of node that currently holds the message and Date/Time at which the message had arrived at that node
LINK_CHECK	Check on availability of aircraft/ground data link. Can be initiated from a ground AGIE member or an aircraft based AGIE member	<i>Not used</i>	<i>Not used</i>

ARINC Project Paper 830 AGIE Strawman Proposal

MsgType	Purpose	MsgReference	MsgContent
LINK_STATUS	Response to a LINK_STATUS enquiry	MsgID of preceding ADDRESS_CHECK message	Possible values are: <ul style="list-style-type: none"> ▪ LINK_OK – if a link is currently operational or <ul style="list-style-type: none"> ▪ LINK_DOWN – if no link is currently operational
ADDRESS_CHECK	Used to check if a destination address can be reached without requiring aircraft/ground data link, i.e.	<i>Not used</i>	Address of AGIE member (client or server) to be checked for ability to accessed
ADDRESS_STATUS	Response to ADDRESS_CHECK	MsgID of preceding ADDRESS_CHECK message	Return value can be one of: <ul style="list-style-type: none"> ▪ ON_AIRCRAFT <aircraft identifier> indicating the address represents a member that entity on that aircraft ▪ ON_GROUND indicating the address represents a ground based member at this time ▪ NOT_FOUND indicating that address could not be resolved by the server directly

4.1.1.2.3 CONNECTION message attributes

The purpose of CONNECTION messages is to coordinate a AGIE level communication link between two AGIE nodes. Table 6 defines the predefined attribute values for CONNECTION messages.

Table 6 – Predefined CONNECTION message values

Message_Class	Purpose	MsgReference	MsgContent Values
CONNECTION_REQUEST	For a client or a server to request establishment of an AGIE link	Not used	<i>Not used</i>
CONNECTION_RESPONSE	Response to CONNECTION_REQUEST	MsgID of CONNECTION_REQUEST message that is responded to	Can take one of the following values: <ul style="list-style-type: none"> ▪ GRANTED – response from server to requestor to “allow” connection ▪ STANDBY – response from server to notify requestor that a request cannot be serviced immediately ▪ DENIED – response from server to notify requestor of the inability to grant the request, e.g. due to lack of authentication

4.1.2 Message Identifier

All AGIE messages are encapsulated as XML documents. To assure complete uniqueness the message identifier is as follows essentially a combination of date/time and *sender* full AGIE address e.g.:

```
"<Random Seq Num>-YYYYMMDD:<AGIE identifier>"
```

For example a typical message identifier *could* be:

```
"12345678-20100312:fo.efb"
```

The random sequence number is assigned by the sending component and SHOULD be sufficiently large to minimize the possibility of duplication.

4.2 AGIE message management

This section further defines when and how certain message classes and types are used.

4.2.1 STANDARD message management

(This section is meant to address various scenarios that may be encountered while a STANDARD messages is in "transit")

Standard messages serve the purpose of conveying some type of end-system (application) information to another application, where both the sender and recipient are AGIE clients. The payload content of STANDARD messages is of no relevance to any of the AGIE nodes (that also includes the AGIE clients) and is only of interest to the sending and receiving applications themselves.

Any client can send a STANDARD message at any time to any other AGIE client anywhere in the system. However, the actual process of delivery of such STANDARD messages can be influenced by a number of factors such as:

- Location of both clients, i.e. on/off aircraft, different aircraft etc
- Link availability for aircraft/ground data exchange
- Type of protocol being employed for data transport

Scenario 1 – Message sent from aircraft client to a client on same aircraft:

ARINC Project Paper 830 AGIE Strawman Proposal

Message Sequence: Aircraft Client 1 to Aircraft Client 2

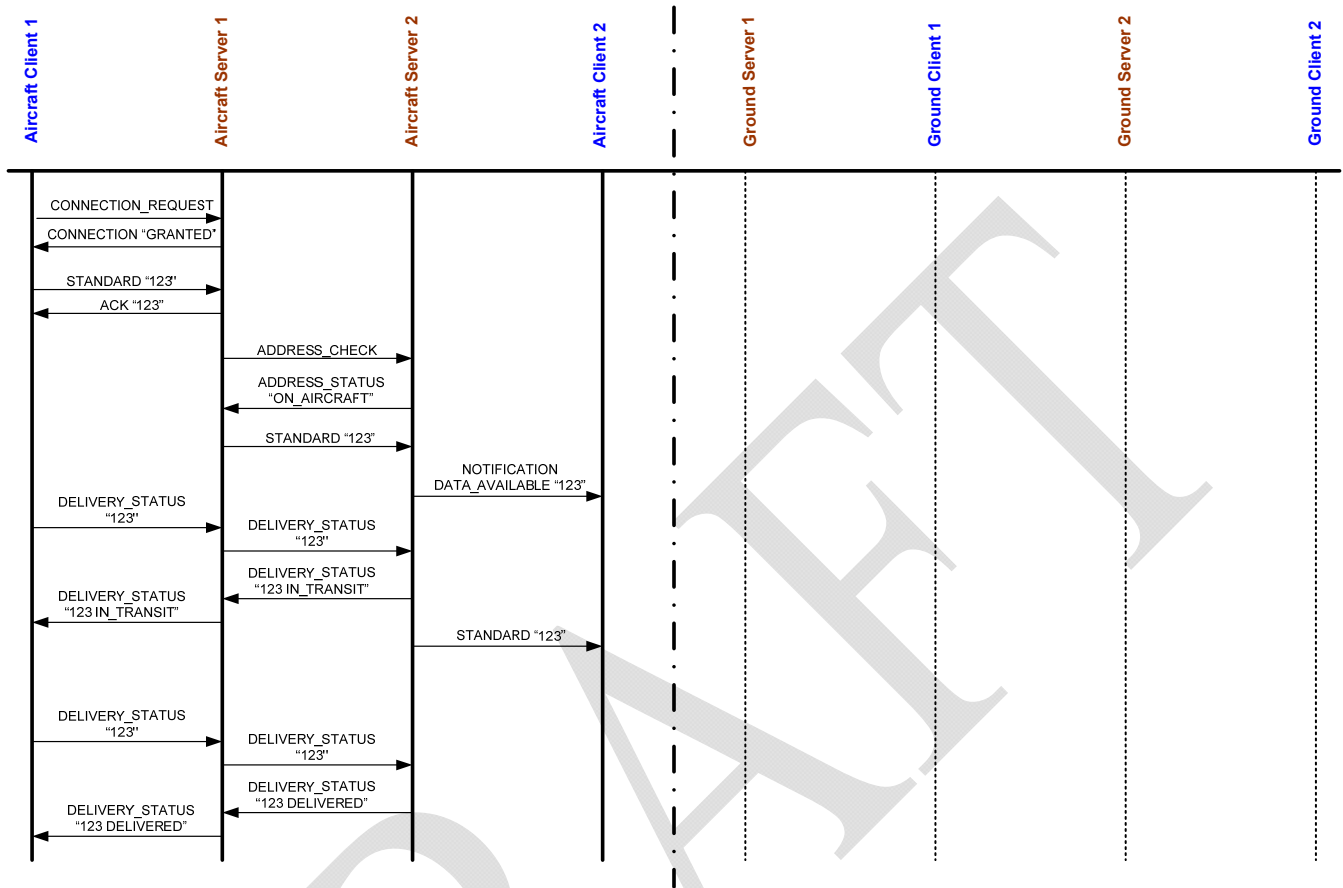


Figure 2 – Message flow within same aircraft

Scenario 2 – Message sent from aircraft client to ground client

ARINC Project Paper 830 AGIE Strawman Proposal

Message Sequence: Aircraft Client 1 to Ground Client 2

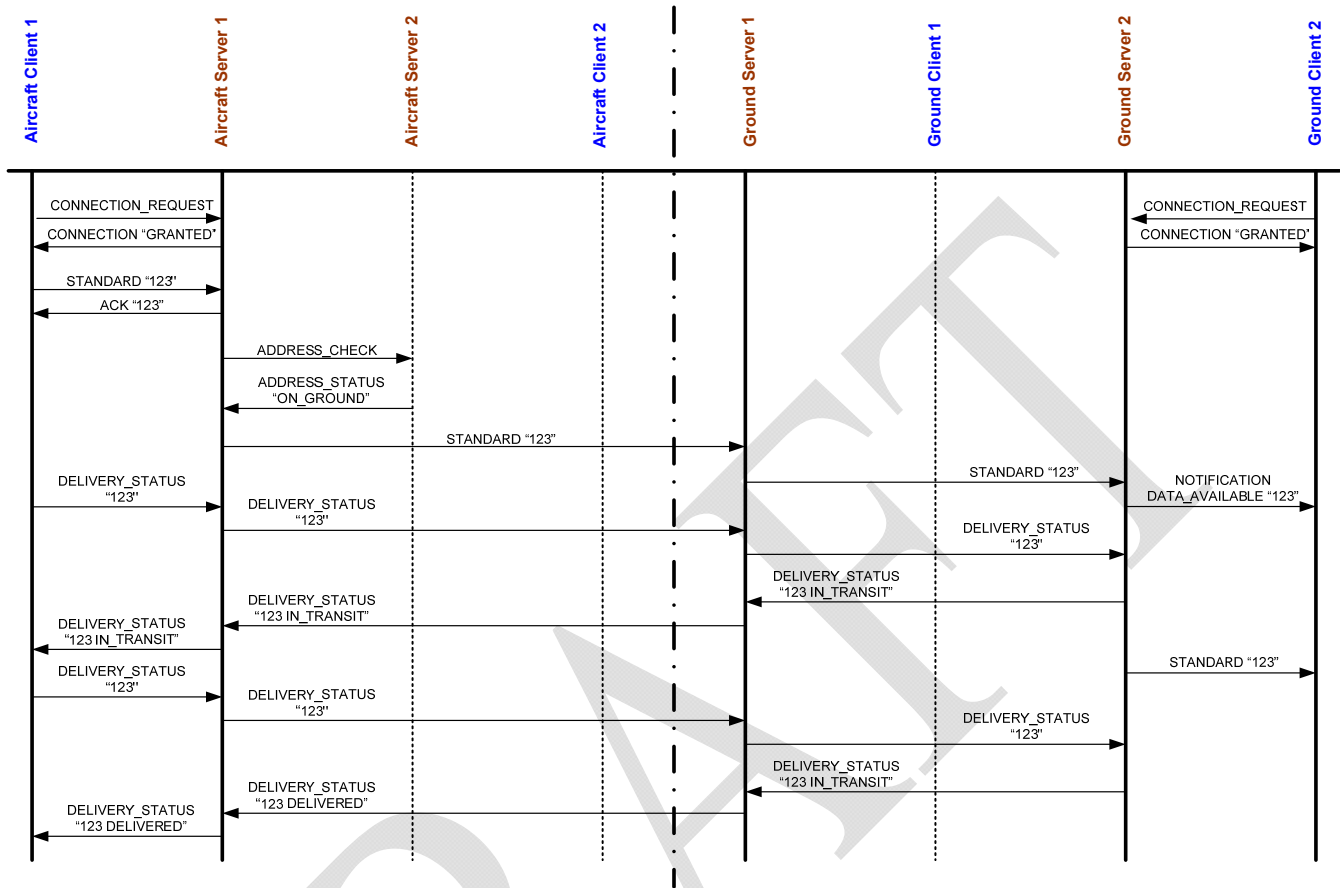


Figure 3 – Message flow from aircraft to ground

from one client to another can be influenced by a number of factors.

4.2.2 COORDINATION message management

4.2.2.1 NOTIFICATION message management

Notifying another AGIE node of something

4.2.2.2 ENQUIRY message management

Checking on various status entities

4.2.2.3 CONNECTION message management

Initiating links between AGIE nodes

4.2.2.4

4.2.3 Connection Authentication

Use of keys etc – tbd...

4.3 Message Attachments

Any STANDARD message may include an attachment but only one attachment per message. Attachments use the MIME standard.

DRAFT

5 AGIE Information Security

The AGIE standard does not stipulate any particular Information Security related requirements. Instead, AGIE makes the assumption that any AGIE entity is being properly authenticated at a lower network level.

Moreover, AGIE intentionally does not impose any data encryption type of requirements to assure flexibility. This type of information security measure is implemented by the aircraft/ground link technologies themselves and/or by the end-system applications.

AGIE's role is simply to assure data are transported to the intended destinations and delivered accurately and in a timely manner.

However, if an AGIE node is not properly authenticated at a lower communication protocol layer and therefore is not able to deliver messages for that reason then the respective delivery status **MUST** reflect this fact, i.e. use a predefined non-delivery reason.

DRAFT

6 Appendices

Detailed protocol definitions...

DRAFT