

**Agent Architecture
for
Aviation Data Integration System**

by

Deepak Kulkarni, Ph.D., NASA

Yao Wang, Ph. D., NASA

May Windrem, SAIC

Hemil Patel, SAIC

Mei Wei, NASA

1	Introduction.....	3
2	Key features of the architecture	3
3	System Architecture	5
4	Summary.....	6

1 Introduction

This paper describes the proposed agent-based architecture of the Aviation Data Integration System (ADIS). ADIS is a software system that provides integrated heterogeneous data to support aviation problem-solving activities. Examples of aviation problem-solving activities include engineering troubleshooting, incident and accident investigation, routine flight operations monitoring, safety assessment, maintenance procedure debugging, and training assessment. A wide variety of information is typically referenced when engaging in these activities. Some of this information includes flight recorder data, Automatic Terminal Information Service (ATIS) reports, Jeppesen charts, weather data, air traffic control information, safety reports, and runway visual range data. Such wide-ranging information cannot be found in any single unified information source. Therefore, this information must be actively collected, assembled, and presented in a manner that supports the users' problem-solving activities. This information integration task is non-trivial and presents a variety of technical challenges. ADIS has been developed to do this task and it permits integration of weather, RVR, radar data, and Jeppesen charts with flight data. ADIS has been implemented and used by several airlines' FOQA teams. The initial feedback from airlines is that such a system is very useful in FOQA analysis. Based on the feedback from the initial deployment, we are developing a new version of the system that would make further progress in achieving the following goals of our project:

- Deliver the requested data to the users immediately in the presence of limited network bandwidth
- Ensure that ADIS continues to run even when its components, such as servers network connections or disks, fail
- Build and deploy information system that is more secure, scalable and reliable
- Make ADIS highly configurable in order to meet the needs of all airlines as well as the interface requirements of various flight analysis software, such as APMS and AGS
- Share data/information across multiple platforms, and applications to make it easier to connect to any software system.

2 Key features of the architecture

The key features of our proposed approach to meet these challenges are as follows:

- Use of agent architecture
An agent-based architecture allows ADIS to attain the maximum flexibility to continuously collect, parse, and transfer data. ADIS agents autonomously collect data when available, operate continuously, and collaborate with other agents. ADIS agents can reconfigure themselves to interface with external clients,

Internet-based servers, and local machines. Maximum flexibility of such interfaces would be ensured by implementing the system as a collection of agents.

- **Distributed agents**

Distributed agents are used to increase ADIS availability and to reduce the workload on machines that contain sensitive data. ADIS security requirements dictate that there should be minimal processing performed on machines that store sensitive information. The ADIS reliability requirements dictate that processing should be distributed to eliminate single points of failures. Use of distributed agents would enable ADIS to meet these requirements.

- **Mobility**

Mobility would be a general characteristic of ADIS agent. An interesting type of ADIS agent is the personal assistant agent. Because a user can access ADIS from different locations, the personal agents have to be mobile in order to transfer the user profile to the system where the user accesses ADIS. The customers of ADIS can be airlines or government agencies. The user profile of an airline consists of information such as airports of interest, periods of interest, and query statistics. This type of information can be both proprietary and sensitive. However, ADIS needs this information in order to optimize the performance of various agents. The personal agents that encapsulate the user interface will safeguard the user profiles. Another situation in which ADIS needs to deploy mobile agents is when one or more machines in the system go down and the agents running on the failed machines have to migrate to other machines to continue running.

- **Use of standards**

ADIS will use standardized ontology and agent communication languages, such as XML, SOAP, RDF, OWL and ACL, in building the ADIS agents. This extensive use of standards will allow ADIS to communicate with software agents provided by vendors and achieve flexibility in configuring and extending the ADIS system. Another advantage of standards is that our customers can easily access the services provided by ADIS.

- **Coordination**

ADIS would use a coordination scheme to match agents that provide services with agents that need their services. Currently, ADIS is deployed on multiple servers with different services and security profiles. The scheme would coordinate the services between redundant agents running on different servers.

- **Security agents**

The security agents will provide threat protection for ADIS agent systems. It will identify and prevent malicious behavior, thereby eliminating known and unknown security risks and helping to provide robust protection for ADIS data and

computer systems. The ADIS security agents will aggregate and extend multiple security functions by providing host intrusion prevention, multi-layer firewall capabilities, malicious mobile code protection, operating system integrity assurance and audit log consolidation.

- **Adaptive and proactive**

To provide high quality of service (QoS) is an important objective of ADIS. In order to meet this objective, ADIS agents are designed to be adaptive and proactive.

We will now describe the proposed agent-based architecture in more detail.

3 System Architecture

Agents can be categorized into Interface Agents (IA), Information Collection Agents (ICA), Utility Agents, Coordinator Agents, and Mediator Agents. An Interface Agent is a communication interface wrapper between humans and computers, or applications and other systems. An IA encapsulates user queries in the proper ADIS forms and publishes the user profiles and forms to mediators or to other agents. Agents of this kind include human computer interface agent, application agents, and personal assistant agents. A personal assistant agent is a mobile agent that ensures that user profile information about airline preferences including airports and periods of interest is made available to the other agents in the system. For example, a download agent can use this information to vary its downloading strategy. Profiles may or may not be directly specified by the end users. In some cases, personal assistant agents can create a profile based on an analysis of patterns of past queries by an end user.

A Mediator is a special agent that is central to the operations of ADIS. It acts as a liaison between other ADIS agents. The agents that provide services will first register the type of information that they provide with the mediator. The agents that request services will also register the type of information that they need with the mediator. The mediator first uses public keys to authenticate the agents and then uses the ontology to translate the messages. After performing these utility functions, the mediator matches up the requester and a provider of the information. This agent can perform a variety of functions:

- refer a query from an Interface Agent to Information Collection Agents
- monitor the progress of the query
- transmit the results of the query
- perform translation according to a common ontology
- perform bookkeeping

An Information Collection Agent provides an information extraction wrapper for information sources. Agents of this type include remote data agents, database agent, data

collection agent, and wrapper agent. For example, Data Collection agents handle data downloads from remote servers to local machines. Data Collection Agents can download data from remote data sources to central servers and from central servers to airline facilities.

Utility agents provide general services such as safety assessment, data mining, data summarization, QoS/Network management and scheduling. For example, data summarization agent would summarize aviation information into a succinct summary. As there is a large amount of aviation data relevant to a problem that an analyst typically works on, the analyst faces the problem of information overload in using this information. While a weather or digital Airport Terminal Information System report may give a lot of details about weather and terminal condition, an analyst may often be interested in only knowing if the conditions are of a certain kind -- VFR, IFR, etc. Similarly, while air traffic control tracks may have a lot of details about the positions of different aircrafts, the analyst may not have time to study these tracks. A brief summary of ATC situation may suffice in the analysis of many problems. Another important utility agent is the network agent. Network agents configure and secure airlines' access to ADIS to meet the security requirements of the airlines. Security requirements vary from one airline to another. Some airlines permit the machine that runs flight data analysis to access the Internet. Other airlines only permit the connection of the flight data analysis machine to the Internet through a proxy server and a firewall. The airlines with the strictest security requirements, machines with flight data cannot connect to the Internet either directly or indirectly. For these airlines, the ADIS data download agent has to run on a computer with access to the Internet in order to download data.

Coordinator agent does the function of locating the best suited agent for a particular request of service. Matched agents communicate using standards such as XML, SOAP and Agent Communication language.

All ADIS agents will have the basic characteristics of autonomy and cooperation as well as adaptation and mobility.

4 Summary

We have developed Aviation Data Integration System (ADIS), a system that permits integration of weather, RVR, and radar data with flight data. The initial feedback from airlines is that such a system is very useful in FOQA analysis. To provide much higher quality of service, we are now implementing the system using an agent based architecture described in this paper. Agent based architecture is ideally suited for an application that integrates heterogeneous data. The autonomy and adaptability of agent architecture ensures continuous data collection and timely responses to external events, such as changes in data update rates. Also, the mobility of agents ensures the continuous

operation of ADIS in case of hardware failure by automatically transferring operations to a system that is running.