December 2001

Air Traffic Information System

Karl Gangle
Illinois Institute of Technology

Martha Evens
Illinois Institute of Technology

Follow this and additional works at: http://aisel.aisnet.org/amcis2001

Recommended Citation
http://aisel.aisnet.org/amcis2001/307

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2001 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
AIR TRAFFIC INFORMATION SYSTEM

Karl Gangle
Illinois Institute of Technology
kgangle@prodigy.net

Martha Evens
Illinois Institute of Technology
evens@iit.edu

Abstract

Government regulations define how an industry operates within limitations. E-Government takes these regulations to form an information system to manage that regulated industry. This paper proposes an Air Traffic Information System to replace the current federal aviation system. The ISO Air Traffic Network is a ground and airmobile network to connect the computers located at control towers, air route traffic control centers, and airplanes. The Air Traffic Communication Protocol includes air traffic management, voice communication, radar communication, and airplane location information. An Airplane System is discussed using an Airplane Communication Protocol to network the different airplane parts together.

Keywords: Air traffic communication protocol, airplane system network, airplane communication protocol, fly-by-message, e-government, e-regulations, object-oriented regulation language

Introduction

E-government is a young concept of representing a bureaucratic system. The United State Government regulates and manages many sectors of society by a set of laws known as the Code of Federal Regulations. Regulation laws exist for aviation, stock exchange, pharmaceutical, and banking. These regulation texts can be converted to an object-oriented regulation language forming e-regulations. This language allows the regulated system to be modeled on a personal computer and integrated into information systems. The regulation language allows monitoring the information system for violations.

This paper presents software components for a new federal aviation system concept to manage an air traffic regulated system. The system is called the Air Traffic Information System. This system is under the jurisdiction of the Federal Aviation Administration and their regulations. The software components can be modeled or implemented. This new aviation system is a large ground and airmobile network. The ground network connects client and server computers located at control towers and air route traffic control centers. The mobile network connects ground computers to computers on board airplanes. An Air Traffic Communication Protocol is proposed allowing all the computers to communicate with each other.

The current air traffic system defined in the regulation text (US Government, 2000) is not a computer network system. Pilots and controllers who use radios to communicate need to change frequencies often (Nolan, 1999). This is an extra task that can be delegated to the computer. Thus allowing the pilots and controllers to focus on their real work of flying airplanes and assisting airplanes. The following sections discuss the old and new air traffic control system.

CAPLE Architecture Overview

Over the years the United State Government agencies have developed regulation rules to define safe and efficient use of industrial systems for products and people of society. There are regulations for many industry sectors, for example aviation, pharmaceutical, banking, and stock exchange. This paper uses the aviation industry as an example.

Computer Aided Policy and Law Engineering (CAPLE) (Gangle and Evens, 2001) is a simulation tool that model regulations to identify deficiencies and study behavior. The regulation text is parsed into event-condition-action (ECA) rules and entity objects. Unfortunately, the existing regulations have deficiencies, for example objects are not accurately defined and numbers are not associated to attributes. The regulations are rewritten in an object-oriented format eliminating these deficiencies and producing more accurate object definitions. These objects are used to model a managed industry, in our case the aviation industry.
CAPLE is organized as a two layered architecture. See Figure 1. The top layer contains preprocessor and runtime parts. The preprocessor part consists of the r++ and Object/ECA parser. The parser uses object language grammar to analyze object-oriented regulation text producing ECA rule and object definitions in the Regs language. Regs is an object-oriented regulation language. The r++ preprocessor translates Regs into C++. The definitions are stored in the object base.

The bottom layer consists of a message router and an object base. The message router is connected to instantiated objects using message queues or shared memory. The object base uses persistent object storage.

The Object base manages objects. An object contains object, method, message, and ECA rule definitions. The object engine receives messages to create, delete, execute method, set attribute value, and more. ECA objects are independent threads allowing automatic reaction to events.

CAPLE uses objects communicating with messages to model a regulated industry. The following sections use these objects and messages to form implementations of aviation industry objects, applications, and message protocols for inter-object communication.

**Air Traffic Information System**

The Air Traffic Information System contains Air Route Traffic Control Center (ARTCC), Air Traffic Control Tower (ATCT), and airplane nodes. See Figure 2. The ARTCC and ATCT nodes use the Air Traffic Network to connect many server and client computers. The server contains object bases, message communication, and message processing. The object base contains the previous section’s entity and regulation ECA rule object definitions and their instantiations. The ECA rules detect regulation violations. The ARTCC and ATCT client computers execute application software. The following sections discuss software descriptions and how the system operates.

**The Current System**

The current aviation system requires pilots and controllers to communicate and coordinate with radio, radar, and flight control strips. Phrases exist for pilots and controllers to communicate allowing effective airplane management.

Radio is the primary communication mechanism between pilots and controllers. The pilot changes radio frequency when communicating with different ATCTs and ARTCC regions. Each region can have 20 to 80 sectors. Each controller manages his or her sector using a radio frequency. When an airplane changes sectors, it also changes controllers and frequencies. So, over the course of a flight a pilot will change many radio frequencies.

Radar is a type of radio that identifies airplanes. It is the acronym for RAdio Detection And Ranging. ATCT and ARTCC transmit radar in their area. Airplanes reflect an image back to the receiver. Airplanes with Mode C transponder transmit the airplane’s location, altitude, and transponder code.

Before a flight starts, the pilot must file a flight plan that is entered into the computer. When the flight starts, the aviation information system prints out a flight control strip at each ARTCC the plane will come in contact with. The controllers use flight control strips to monitor the course of each flight. So, a large amount of strips are print out.

The ATCT controllers have a variety of management responsibilities. Several among them include clearances for landing, taking off, leaving airport airspace, and entering airport airspace. These tasks use phrases forming a communication protocol.

The current aviation system has the following problems. First, reducing or eliminating all the radio frequency changes pilots must do. Second, change the concept of radar. Third, eliminate the printing of flight progress strips. Solutions to these problems will be addressed in the following sections.
Air Traffic Network

The new aviation system revolves around a ground based computer and airmobile network. The ground network uses the Internet Protocol (IP) to connect the ATCTs, ARTCCs, and airport vehicles. The airmobile network uses the Aviation Protocol (AP) to communicate between airplane nodes. The Air Traffic Network is an Open Systems Interface (OSI) architecture (Stallings, 1990). See Figure 3.

The Application Layer consists of egovernment user applications. These applications are for users in the airplane, ATCT, ARTCC, and national air traffic management centers. The following sections discuss the application software in more detail.

The Presentation layer in Figure 3 provides a set of service methods that allows any application to create and send messages using the Air Traffic Communication Protocol. See Figure 4 for the Air Traffic Communication Protocol messages. Other services are activating message encryption/decryption and setting the message receive interrupt vector.

A special message pair replaces the radar system. The ARTCC or ATCT transmitter broadcasts a RequestCurrentInformation message to all airplanes. The airplanes receive this message using a message transmitter and receiver (MTR) device connected to a computer. The onboard computer processes the RequestCurrentInformation message by creating a CurrentInformationReply message. The message fields consist of latitude, longitude, altitude, speed, and status. The responds is created and transmitted. The ATCT and/or ARTCC server receives the message and places it in a queue. The server finds the airplane object in the object base and stores the message information in the object. This information is used and displayed by the Airplane Traffic Management and Sector Traffic Management applications.

The Session and Transport Layers contain the Transport Control Protocol (TCP). TCP breaks the message into smaller packets for transmission. On the receiving end, the packets are reassembled into the original message and passed to the next layer above. Checksum is provided and other features.

The Network Layer uses the AP and IP. The AP is like the IP with one main difference – the dotted address. The AP was created with a new address space because the Internet address space might not contain enough addresses for all the airplanes. AP uses ccc.mnn.nnn for an address-coding scheme. The c’s are three bytes indicating a country code. The n’s are six bytes for airplane network addresses. AP also supports sending, routing, and receiving messages.
The Link and Physical Layers interface with the transmission mechanisms used by the aviation system. These mechanisms are satellite, radio, and high speed dedicated lines. The ATCTs and ARTCCs use an existing radio network throughout the country (Illman, 1999) to communicate with airplanes.

### Airplane Systems Network

Today’s airplanes are built using either wire cable, “fly-by-wire” (NASA, 1999, NASA, 1996), or computerized methods. Airplanes today do not operate using messages. Messages are the common theme in this new Air Traffic Network and Air Traffic Information System. To stay with this theme, airplanes will have to adapt. The existing airplanes will install an MTR. New airplanes will incorporate a new method for airplane operation called the Airplane Systems Network. This method uses fly-by-message consisting of the Airplane Communication Protocol. These messages transmit on a fiber optic ring around the fuselage. A repeater (R) allows an extension ring to be placed in the wings, rear stabilizers, and vertical stabilizers. The extension ring is present because if appendages break off then the network is still operational.

The fiber optic ring connects to a network interface control card (NICC) controlling a system component. There are many different systems on board an airplane with a corresponding network address space. The address space consists of two three-byte sequences, sss.ccc. The s’s indicate a system identification number. The c’s indicate the system’s component identification number. Each system is assigned a number along with each component. See Figure 5. For example, the hydraulic system has 003 for a system address. Example component addresses are landing gear is 003.001, thrust reversers is 003.002, nose wheel steering is 003.003, and brakes is 003.004.

The Airplane System Protocol contains messages that are sent from the Flight Control Management application on the central computer to an NICC with the sub-system destination address. Figure 6 contains messages to accomplish this task. The computer contains all the current values of the different sub-system attributes. These values and menu options are displayed on several touch screens (D) replacing the old switches in the cockpit. See Figure 7.

The main computer is connected to the MTR allowing ATCT and ARTCC messages to be process automatically and manually. The computer processes automatic messages, such as RequestCurrentInformation, GPS, and NeighboringAircraftList. Manual messages require the pilot to type or speak text to create a flight plan or talk to a controller or pilot. The message is then sent on to an ARTCC, ATCT, or airplane.

### Air Traffic Control Tower

The Air Traffic Control Tower (ATCT) is a node in the egovernment information system. It manages the airport and surrounding airspace. In the tower there are three egovernment users Flight Data Controller, Ground Controller, and Local Controller (Illman, 1999)(Nolan, 1999). This section discusses the application software used by different controllers.

The Flight Data Controller uses the Flight Data Management (FDM) application software. This software primarily manages Flight Progress Strips automatically sent by the computer, reviews and stores daily records, and enters information for the Automatic Information message. All the active strips are displayed in the Flight Progress Strip Window. The strip fields are aircraft identification, revision member, request origination, aircraft type, network address, departure time, flight altitude, departure airport, flight route, and destination airport (Nolan, 1999). The window has a menu for all strips. The menu consists of save, send, request IFR, and close buttons. Save forms a StoreFlightProgressStrip message. The message gets sent to the object base server and content stored. The send button allows the strip to be sent to someone else, for example the Clearance Deliver Controller. The close button closes the strip. Lastly, the request IFR button is used when the controller has detected that the flight has not been cleared for an Instrument Flight Rules (IFR) flight. The button contacts the ARTCC using the RequestIFRClearance message. See Figure 8. The response is returned in the IFRClearanceReply message.
The Clearance Delivery Controller uses the Clearance Delivery Management application software. The software receives airplane clearances from the FDM and transmits departure clearances (Nolan, 1999) to airplanes. The departure clearance message consists of airplane identification, clearance limit, departure procedures, flight route, and altitude.

The Ground Controller uses the Ground Management application software. This software monitors and graphically displays the airport ground area, airplane, and vehicle traffic. The controller sends instructions to taxiing aircraft and vehicles as they move across active runways, inactive runways, and other airport parts. Thus avoiding collisions. The monitoring takes place by broadcasting the RequestCurrentInformation message to the airport ground and airspace. The location data is stored in the airport object base. It is used by controller application software.

The Ground Management application sends a RequestAirportGroundTraffic message to the ATCT server. The server has an object base that contains an airport traffic object with the locations of all the airplanes in the airport airspace. See Figure 9. A method is executed to obtain all the airplane and vehicle locations on the ground and sends an AirportGroundTrafficReply message to the application. The application displays a symbol at each traffic element’s location. The traffic display allows the controller to complete his/her task of sending taxiing instruction to traffic. Example instruction messages are taxi to a runway, taxi to a terminal, or avoid active runways. The software also allows early detection of accidents allowing the controller to send instructions to avoid collisions.

The Local Controller manages the separation of arriving and departing aircraft using the Airplane Traffic Management software. The controller uses the software to issue instructions by sending several messages to airplanes. The AirplaneSeparationInstruction message is sent to departing airplanes indicating they must hold until the arriving plane has landed then the departing plane can take off. See Figure 10. Therefore, the spacing is required between airplanes. The above instructions can also be sent in a Departure message. The later message tells the pilot the departure runway, after air born the direction to fly, and if cleared for take off. If the pilot has not been cleared, then the controller sends a TakeOffClearance message indicating it is OK to take off.
Air Route Traffic Control Center

The egovernment aviation system is broken down into areas (Illman, 1999)(Nolan, 1999). Each area is managed by an ARTCC. An area has many sectors each with a Sector Traffic controller. The controller uses the Sector Traffic Management application to graphically display separated traffic and communicate with the airplanes in their sector. The main task for each Sector Traffic controller is to separate airplanes into different levels. The RequestLevelChange, LevelChangeReply, and LevelChange messages are used to communicate between the controller and pilot.

The Sector Controller is also responsible for handing off airplanes flying into a new sector. In the old system, each controller has there own frequency. So, Pilots frequently had to change radio frequencies. In the new system, this task is off-loaded to the ARTCC server. When a pilot nears a sector boarder, the ARTCC server detects this because it knows the location of the airplane and the sector boundary. The server sends a HandOff message to the airplane with the new sector controller’s frequency. The airplane’s computer receives the message and automatically changes the transmission frequency. So, the pilot just has to concentrate on flying the airplane.

The Flight Data Controller helps by receiving and sending information to other sector controllers using the Fight Data Management application. This information includes weather and flight progress strips. After a pilot has filed a flight plan, the plan is sent to the server, stored in the object base, and FlightProgressStrip messages are automatically sent to all ARTCC’s along the flight route. These messages are displayed on the controllers Flight Progress display. The controller reviews and makes any changes. They are then sent to the Sector Controllers.

Summary and Future Work

The past sections have discussed different aspects of the Aviation Information System. CAPLE analyzes regulation text for entity and ECA rule objects creating a model of the regulated industry. These objects are implemented in the Aviation Information System using client and server computers at ATCT and ARTCC locations. The server contains an object base with the regulation entities and ECA rule objects. These rules are used to verify the airplane and aviation system’s compliance with regulations. The client computers execute application software for the different controllers. The computers communicate using TCP/IP and communicate with airplanes using TCP/AP. The Air Traffic Communication Protocol defines messages for airplanes, ATCTs, and ARTCCs to communicate with each other. A message conversation is used to replace radar. The airplane is modified to implement fly-by-message in the Airplane Systems Network.

Future work involves several items. First, the message transmitter and receiver must be able to send and receive a large volume of messages in a small time. Second, the Aviation Protocol must be integrated. Third, integrate the Airplane Systems Network into airplanes. Fourth, create an object-oriented regulation text to define e-regulation objects and rules defining and enforcing a regulated industry. Fifth, develop a computer regulation language to specify information system components that manage and regulate industry sectors.
Figure 8. Flight Progress Strip (FPS) Scenario

Figure 9. Taxi Instruction Scenario

Figure 10. Departing Airplane Scenario

References

