JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 3, December 2006

Using Language Technology to Increase Efficiency and Safety in ATC Communication

Margret Dora Ragnarsdottir,* and Ebba Thora Hvannberg†
University of Iceland, Reykjavik, Iceland

and

Helga Waage‡
Hex Software, Reykjavik, Iceland

Voice communication is a volatile part of Air Traffic Control (ATC) and thus the International Civil Aviation Organization (ICAO) puts great emphasis on improving communication in ATC. Research has shown that miscommunication happens on average once every hour per radio frequency where there is frequent communication, such as in Terminal Radar Approach Control (TRACON). This paper proposes that a Language Technology System (LTS) can make communication between controller and pilot more reliable and efficient, thus improving safety in aviation. An LTS can for example detect read back errors. It can also directly feed data from the voice recognizer into a Flight Data Processing System or interact with it otherwise as we show. By interviewing air traffic controllers and studying the literature, we identified these and several other examples of use of language technology in ATC. In this paper, we explore one of the identified examples, a system that takes over the communication between controller and pilot in oceanic ATC. This system is not meant to control the airspace autonomously, but only relay information. Latest advances in language technology have enabled the development of such a system. The functionality of the proposed LTS is described using scenarios and sequence diagrams. A demonstration LTS using Hex Technology was implemented. A usability test was administered to seven controllers. Their attitude towards the agent was positive and indicates that there is reason for further research. The performance and error logs of the LTS were analyzed and give guidance on further development of a fully functioning LTS for ATC.

I. Introduction

IN Air Traffic Control (ATC), it is vital that all parties understand each other and that information is delivered and received accurately. This paper proposes that language technology can be used to assist pilot and controllers in ATC to improve this communication and thereby increase safety in aviation. A substantial part of ATC communication is vocal over radio. Voice communication is very volatile and research has shown that on average one error is made every hour per radio frequency where there is a lot of communication such as in the ATC tower.[6,7] The biggest risk in aviation has been identified as being the problem of getting information correctly from the Air Traffic Controller's Flight Data Processing System (FDPS) into the aircraft's Flight Management System (FMS). Therefore, it is interesting to

Received 11 August 2004; revision received 3 September 2006; accepted for publication 22 September 2006.
* mdr@hi.is
† ebba@hi.is
‡ helgaw@hex.is

study how communication between the controller and pilot can be improved, to ensure that information is transferred efficiently and reliably.[9,20,29]

This paper discusses the ATC environment and how Language Technology can be utilized to enhance communication in this environment. Much emphasis has been on making this communication more text based. The transition from voice communication to data link is still under investigation since implementing data link will be costly for the airlines and the advantages are not clear.[9,20,29] The authors of this paper feel that it is important to investigate other possibilities of improving communication as is shown in this paper. It is maintained that the benefits of voice-based communication enhanced with language technology merit closer scrutiny. This idea was put to the judgment of professional Air Traffic Controllers in a trial and their response was encouraging for further research.

II. Environment

In ATC, information flows between the controller and pilots to ensure that the controller is in control of the airspace and knows where every aircraft is. It is imperative to the safety of the aircraft in the airspace that the information entered into the FMS in the aircraft and the FDPS in the ATC Center is the same. The communication is most frequently direct between controller and pilot as described in Fig. 1. There are two persons and two systems that all need to be synchronized, through three links of communication:

A) Communication from the controller as he enters information into the Flight Data Processing System (FDPS).
B) Radio communication between the controller and pilot.
C) Communication from the pilot as she enters information into the Flight Management System (FMS) of the aircraft.

Each link in the communication is a safety hazard, for it opens up the possibility of the information being corrupted.

Things are furthermore complicated in the oceanic environment where the communication is mediated through high frequency (HF) radio. For HF communication special radio stations are in place and the controller/pilot communication is relayed through a third party radio operator. This adds links to the communication. Continuing with

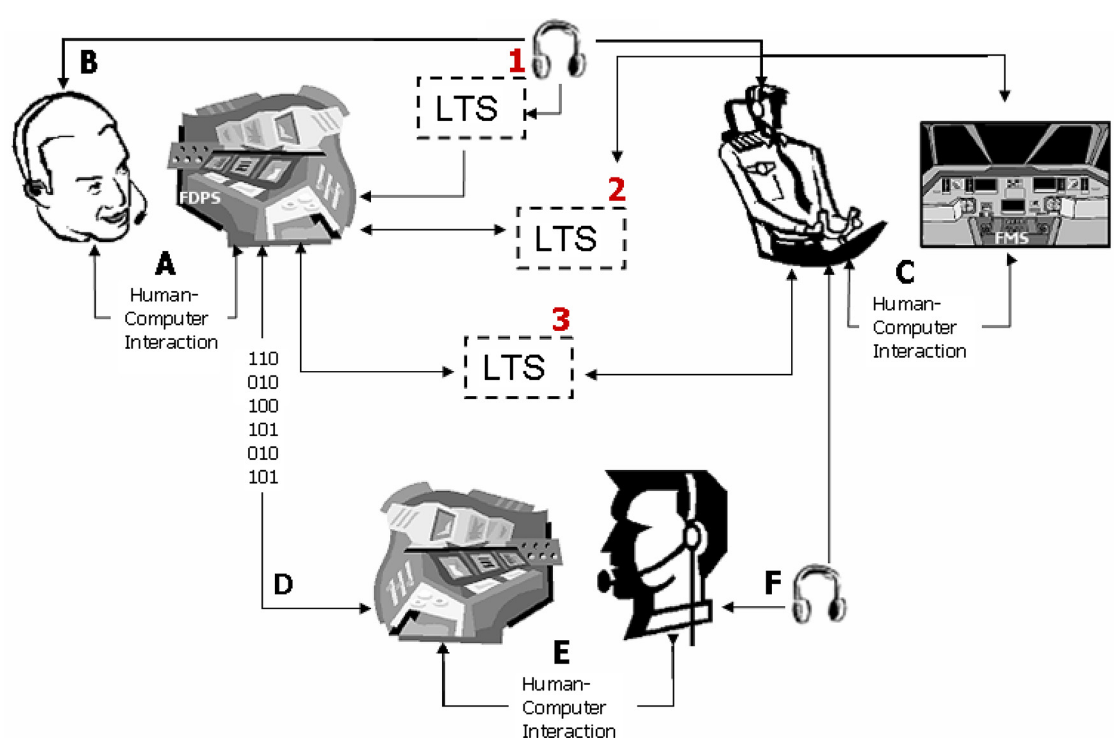

Fig. 1 Air Traffic Controller/pilot communication.

the alphabetical ordering from above, the communication links in this scenario are as shown in Fig. 1:

D) Communication between Air Traffic Controller and radio operator. The messages are most frequently sent between them as text messages via a FDPS or, rarely, directly between controller and radio operator via radio/phone (not shown).
E) Communication between radio operator and his workstation as he enters and receives information.
F) Communication between radio operator and pilot via HF radio.

A. Oceanic Air Traffic Control

To enable surveillance in oceanic airspace, the pilots have to check in with the controller of the airspace they are flying through at certain waypoints. At the outset of a flight, a pilot has to prepare and report to Air Traffic Control a flight plan with the track that will be taken and estimates of the destination times at the track's waypoints. The controller does not know exactly where the aircraft are at any given time but instead has to project their position using their last point, next point and speed of travel. For the controller to understand the traffic in her airspace and maintain safety, it is vital that the aircraft conform to the flight plan and that it is identical to the flight plan the Air Traffic Controller is using. Otherwise, it is impossible for the controller to instruct other flights where it is safe to fly. The main task of the Air Traffic Controller is to make sure that the minimum separation between aircraft is maintained. In the oceanic environment planes are not supposed to fly closer than 50–100 nautical miles radius to another aircraft. Vertical minimum separation is 1000 feet in the North-Atlantic airspace. However, since the aircraft normally conform to the flight plan, the communication is usually routine.[12–14]

To summarize, Oceanic ATC has two distinct characteristics:

1) It is dependent on voice communication for surveillance because there is incomplete radar coverage.
2) It uses High Frequency (HF) radio for communication. Very High Frequency (VHF) radio coverage is dependent on line-of-sight between aircraft and radio transmitters for communication, which is not available over the oceans and deserted terrains.[11,13]

We decided to focus our research on the Oceanic Environment (OE) because:

1) Communication in the OE is very *formal and structured*.
2) Communication in the OE is *not time critical*.
3) *Recent developments* in the OE motivate redesign of communication. This calls for discussion of new ideas.

III. Recent Development in ATC

The most important improvement initiatives in communication in Air Traffic Control involve moving from voice communication to text communication using a data link between the controller and the aircraft. This data link enables the controller and pilot to send text messages back and forth instead of communicating through the radio. The main initiatives in this direction are the Automatic Dependent Surveillance (ADS) and Controller Pilot Data Link Communication (CPDLC). ADS allows direct data link between the aircraft and controller over HF in the oceanic environment and improved positioning using satellites. CPDLC enables data communication over VHF between controller and pilot.[10]

The future outlook for Air Traffic Control is that all aircraft will be equipped with data link communication tools and that voice communication will only be used as a backup if the data link communication fails.[13] In the oceanic environment this will greatly influence the operation of the commercial operators (such as ARINC) since their workload will decrease to a mere trickle. Soon the radio facilities will be run for backup communication alone without much cut in the cost of running these facilities today.[13]

As with all new technology, there is a transition period from introduction to users' embracement, even after regulation mandates their use. It is expected that the oceanic environment will have mixed equipage (some aircraft having data link functionality whereas others will not) for a long period. This will increase controllers' workload, who have to make an additional check before communicating to see how it is appropriate to communicate with a certain aircraft.[13]

These initiatives are still debated within the aviation community. The biggest debate is about the cost that data link will bring to the airlines in new equipment for their aircraft.[23] In the current business environment, airlines are certainly not looking to add cost to their operation.[16] Another reason for the aerospace industry's hesitation to go through with this transition is that researchers do not agree that there is an advantage of text communication over voice communication. Moving the communication from voice to data link changes the flow of information between pilot and controller greatly. ATC and flying an airplane are both high memory load tasks where the human operators need to keep in mind a lot of information. This means that their working memory is loaded with information. Cognitive psychology has shown that the working memory has two types of storage that do not seem to interfere with each other, one for visual-spatial information and another for verbal-phonetic information. Visual-spatial information is gathered through vision. Verbal-phonetic information is gathered through hearing. Using both types of information in the work-environment helps the operators do their job more efficiently as they can receive more information at a time than if they are only using one type of information media. In the controller/pilot work-environment, there is a lot of visual-spatial information that the human operator needs to perceive and understand, e.g. gauges and radar screens. Using another type of input channel (verbal-phonetic) helps the human operator to perceive the mediated information. Making the communication all visual will effectively add to the already loaded visual-spatial working-memory of the operators, potentially overloading it and influencing their perception of the information and hence their situation awareness. Considerable research has been done on the difference between communicating using data link vs. voice, but the results are still inconclusive.[9,20,29,31]

While this debate has not been resolved, it is important to look into other possibilities, which could improve safety and efficiency in ATC, and that might not be as costly and might even bring further advantages. Using Language Technology Systems is potentially cheaper and faster to implement for the aerospace industry. There is no need for new equipment on board the aircrafts; it is set up in the air traffic control centers. Therefore, it is an exciting alternative, which is worth researching.

IV. Proposed Use of Language Technology

User-Centered Design (UCD) is a methodology of system design that maintains that the people who will use the system must be a part of the design process from the beginning.[29] This study uses UCD and with that in mind sought the opinion and valuable help from professional Air Traffic Controllers, right from the outset. The following eight functionalities (no particular order) for an LTS were identified through interviews with controllers and extensive literature research:

- Listening for information that is *missing*, for example by making sure that there is readback and that *all* information is read back.
- Listening for information, that is *wrong*.
- *Entering* information directly into the FDPS.
- *Logging* the communication.
- Relaying the information between controller and pilot using the *appropriate mode of communication*, especially in airspace of aircraft with mixed equipage.
- Relaying the same information in text and voice for *redundancy*.
- Serving as a *backup* for data link communication.
- Assist with *training*—the LTS can simulate communication between controller and pilot.

Detailed analysis of voice communication in ATC shows that in a little less than 1% of transmissions there was some sort of miscommunication. Even though this does not seem like a real threat to air traffic, it means that on average at least one miscommunication is made every hour on a single frequency where there is heavy communication such as in the tower. Note also that these are reported incidences. Most mistakes are caught on the fly and are never reported.[6,7] Any effort to try to reduce these errors will improve safety in aviation.

Summarizing the functionalities above, language technology could support the communication in the following ways: (see fig. 1)

1) Analyze the communication and make sure that the there is no discrepancy. This enhances the **B** communication link.
2) Analyze the communication and entering data into the FDPS and/or the FMS. This eliminates the **A** and **C** communication links.
3) Take over the communication between the Air Traffic Controller and pilot. This eliminates the **D**, **E** and enhances the **F** communication link.

Combining items 2 and 3 makes it possible to use the most appropriate way of communicating based on the aircrafts' equipage.

The limited scope we have selected for this paper is a system that takes over the communication according to scenario 3 described above and shown in Fig. 1.

Based on this analysis, we designed and implemented a limited prototype that we demonstrated in a *Wizard of Oz* usability test.[2] The purpose of the test was to get the reaction of professional Air Traffic Controllers. This would indicate whether there is reason for further research.

V. Language Technology

Language technology deals with how computers can process and use language. This has many practical applications such as automatic translation, or natural language interaction with users. Natural Language Processing (NLP) is the technology of making computers 'understand' natural language. So that it can understand speech we use Automatic Speech Recognition (ASR). To make it speak we have Text-To-Speech (TTS) generators.[19] VoiceXML uses these components. It is a scripting language for specifying the exchange of information between a user and a speech-enabled application.

A. Language Technology in ATC

Some research has already been done on how language technology can be made useful in the ATC domain. For example, Schaefer has researched how language technology is used in training of Air Traffic Controllers.[23] In 2002 an LTS was implemented in training simulators in the United States.[27]

Applying language technology in the ATC domain puts certain requirements on the Language Technology System (LTS) beyond those that are used commercially in non-safety critical fields. In this case, we must require the system to:

- *Be highly accurate and reliable.* It is important to set a minimum level of acceptable recognition that must be achieved before any system using ASR in ATC can be used effectively. If the recognition is worse than human, it will be more of a disturbance than support.
- *Be speaker independent*. The system must be able to understand everyone in the airspace.
- *Be very responsive.* A system made for time critical situations like ATC cannot slow down the communication and has to deliver warnings as soon as discrepancies are found.

Below, we will discuss how each of these requirements is achieved with language technology.

B. Natural Language Processing (NLP)

The lowest level of NLP is lexical analysis. With a dictionary, the computer can scan through text and make sure that every word in it is also in the dictionary. The lexical analyzer does not understand the relationships between the words, nor their meaning, and will allow the users to make grammatical errors. Perhaps not very sophisticated but very useful for example in spell checking.[18] This is also enough to design an LTS that only needs to understand keywords to be able to react sensibly to the user input.[9]

Moving to upper levels of language processing we come to syntax. In many languages, the rules of grammar and their exceptions are well documented. Linguistic grammar can easily be formalized and therefore it is possible to write a computer program that knows and uses it. To decipher the input, the systems build a so-called parse tree. The parse tree shows how the words in the input constitute a sentence from the grammar that the parser knows. More than one legal parse tree is often available for an input (think 'For sale—an antique table suitable for lady with thick legs.'). If the parser is required to select between the trees it commonly makes a decision based on assessed probabilities of use, based on an analysis of how the language is used. Information like that is collected in language corpora.[19] This technology is used for example in grammar checkers or dictation programs.

An LTS designed for ATC does not have to understand the meaning of utterances. It only needs to understand whether certain information is being conveyed or not. Even though ATC communication is structured, the degree still varies and so the system must be flexible. Thus, it should not focus on the syntax of the utterances, and it suffices that the system uses keyword recognition. This means listening for words or phrases irrespective of the order in which they appear.[18,20]

An ATC corpus has already been made by Ward.[28] By using such a corpus, it is possible to identify the keywords in a distress call for example. The LTS is then instructed to listen for these keywords instead of a sentence that conforms to communication regulation. This allows for flexible communication and fulfills the requirement for accuracy and reliability in the ATC environment.

C. Automatic Speech Recognition (ASR)

Automatic Speech Recognition (ASR) is the technology used to translate speech into text that can then be analyzed by NLP.

To analyze the speech, the ASR records it (see Fig. 2). Then it matches the recorded speech sounds to phonemes. Phonemes are the smallest meaningful units of language and words can be described by their constituent phonemes. From this matching it may be possible to build many words.[18,19]

Most commercial ASRs will give results with a recognition rate, which assesses the probability that the given result matches the correct word. This is important information that can help analyze the performance of the ASR. If the LTS that is receiving the result is only looking for one result, the ASR chooses the most likely candidate and sends it to the underlying LTS with the recognition rate. Many ASRs can be made to reject any result that has recognition rate below a given value. This will lower the number of false recognitions by the ASR, when the assumption about what it heard is simply wrong. However, it will also increase the number of false rejections, when it heard correctly but was not very certain about it.[1]

The human voice has a very broad range of loudness, timbre and pitch. The ASR needs to adapt to this and so focuses on the shape of the speech wave and its frequency. A related complication is that pronunciation varies widely. Pronunciation is even dependent on how the speaker feels and his health (think of someone with a bad cold).[19] To improve recognition, an ASR can be trained to listen to only one user. This will increase the recognition level substantially but it also limits the usage.[19] In ATC, this is not applicable as the system must be able to understand everyone in the airspace. So, it is important to improve the recognition with other means. One way is to use keyword recognition and as we have previously discussed. This is appropriate for the ATC environment.

Current commercial ASRs boast of 95–98% recognition rate in speaker independent single/double word trials, down to 55–80% in continuous speech.[4,5] Since we are using keyword recognition we can assume 95–98% recognition rate.

Accuracy of human speech perception is well researched and the results show that it is around 96–99%. An example of such research that is very relevant here is a study done by Cardosi which finds that between 0.73% and 3.36% of all communication is incorrectly or incompletely recognized in ATC communication, dependent on the complexity.[6] This places human speech perception accuracy between 96.6–99.2%.

To be considered useful, the system needs to improve the working situation. Therefore, the system must show better recognition than humans do. Because of the formality and limited vocabulary of ATC and with the correct

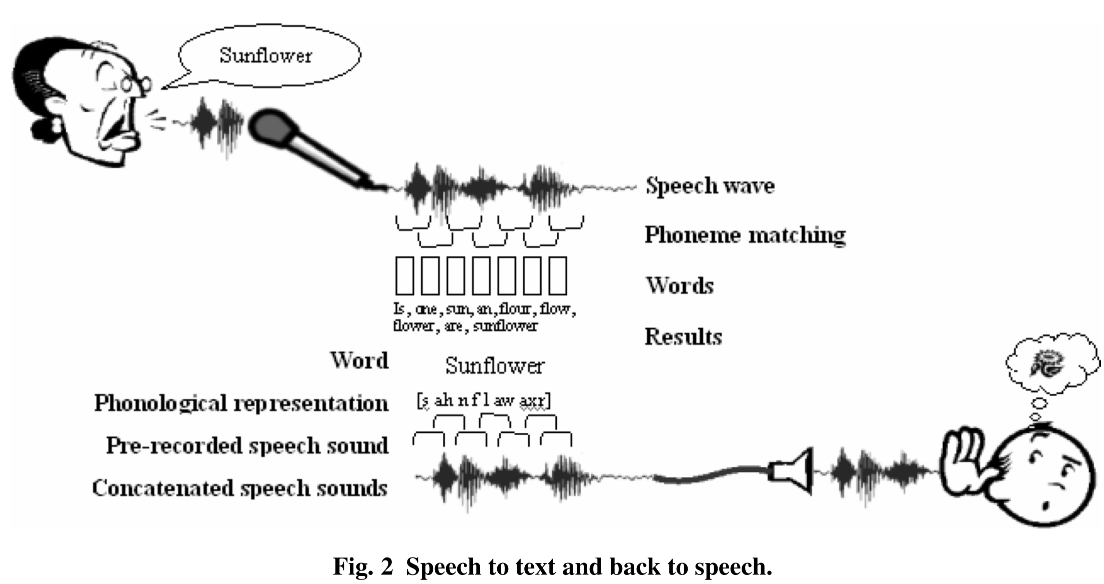

Fig. 2 Speech to text and back to speech.

setup and careful testing, the required level of accurate and reliable speech recognition can be achieved with ASR.

D. Text-to-Speech (TTS) Generation

In Text-To-Speech generation, text is translated into speech. Speech sounds are either created synthetically or from pre-recorded speech sounds. Pre-recorded speech generators have become more popular.[19,22,26]

The coarsest form of pre-recorded TTS generators has a limited vocabulary and then concatenates words to form sentences. This is very limiting for the application of the TTS.[19,22,26] More sophisticated generators are based on prerecorded input, which is cut into very short segments that represent phonemes. The TTS generator maps the letters of the words to the appropriate phoneme for pronunciation. However, the result of concatenating phonemes into a string of sounds will not sound very smooth and the flow of speech will be greatly compromised. This contributes to people feeling uncomfortable using it. It has been shown that to make the voice sound more real and to make it flow convincingly the phonemes must be cut down into smaller pieces called speech sounds. They are then concatenated from the mid speech sound before it and until the mid speech sound after it, two speech sounds (diaphones) at a time.[2,19,26]

When turning text into speech the TTS has to write out the word phonologically like we have done in Fig. 2. Here, characters are used to represent speech sounds. The TTS generator then maps the representation to the appropriate sounds and plays for the user.[19]

E. VoiceXML

VoiceXML is based on eXtendable Markup Language (XML). XML is designed to represent data. The biggest advantage is that it is a platform independent language and thus it can serve as a means of communication between applications written in different programming languages for different platforms. VoiceXML has made development of voice-based applications easier by placing all processing of the application on the receiving end of the call. The caller only needs to have the ability to send and receive sound.[1]

Figure 3 shows how VoiceXML language technology systems work. The VoiceXML scripts are run in a voice browser, the same way HTML scripts are run in web browsers. A user accesses the LTS by using any microphone and loudspeaker such as a phone. When the user initiates the communication the voice browser sends an HTTP request to the host of the LTS to request what it is supposed to show the user, in the same way as a web browser. The web server returns a VoiceXML document, which the browser shows the user by playing a sound file. The voice browser hosts both the ASR and TTS.[1,25]

In the VoiceXML standard, grammar describes the vocabulary for the context of the conversation and thus what the ASR should be listening for. That way, the ASR does not have to take the speech input and try to match it to every word in its dictionary, but only to the keywords in the grammar. If the input fits the grammar the ASR returns the input to the LTS, and any other input is irrelevant to the application. Thus, the ASR rejects it and returns an error to the LTS and so the LTS can request that the user gives a more appropriate response[1].

The voice browser controls the interaction between user and system. It receives from the LTS what it is supposed to say to the user either as a sound file or text with instructions on how it is supposed to be read by the TTS. The LTS also relays the grammar to the voice browser to instruct the ASR what to listen for next.[1]

Using VoiceXML the LTS can log the communication and thus it can be referred to later, for example to check a status report against the last report.[1,25]

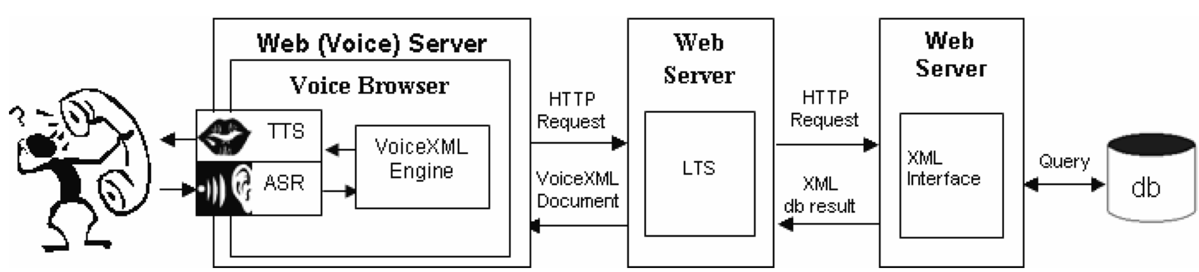

Fig. 3 VoiceXML system architecture (adapted from[1]).

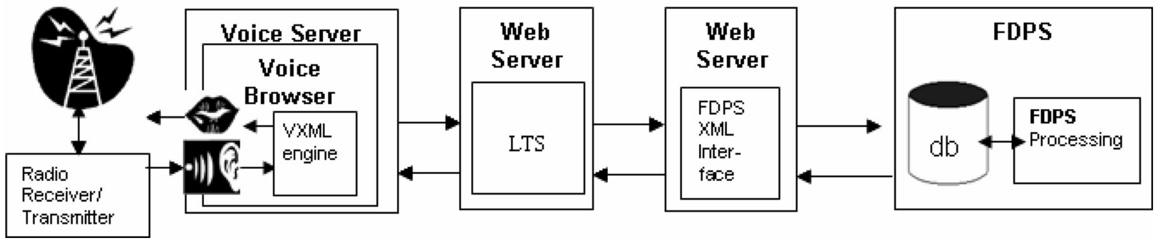

Fig. 4 System architecture including the Language Technology System (LTS).

VI. System Architecture

This paper describes research that was done at the Air Traffic Control Center in Reykjavik, Iceland. The Flight Data Processing System (FDPS) at the Reykjavik ATC Center is specially designed for the center. It manages all the data on aircraft in the airspace and gives an overview of the status. The controller can view the flight plan for each aircraft on digital flight strips and view where it currently is located either on a radar or situation display which depicts graphically the flight plans and projects the aircrafts' status. Conflict probe and text communication modules are included. The interaction with the radio operator in oceanic environment takes place through the text communication module. The controllers can manipulate the data on digital flight strips just as they did with their paper predecessors.

The system architecture for the system we are proposing for the Icelandic Oceanic ATC environment is based on the VoiceXML technology (described in Fig. 3) and is shown in Fig. 4.

Note that for this environment we are using radio rather than telephony communication. Therefore, we must allow radio communication to enter and leave the system. More importantly, for any LTS to work in this environment it will have to interact with the FDPS. Since there are many different systems involved, XML should be used as a bridge between components. The LTS makes a request to the web server, which then queries the FDPS databases and returns the result in XML format. With this architecture, the LTS will for example be able to request a flight plan for a certain aircraft directly from the FDPS database.

VII. Hex Agent Technology

Language technology systems that give the right functionality to fulfill the requirements discussed above are called conversational agents. They are types of LTSs that can listen to (through ASR), understand, and take part in (through TTS) natural language conversations.

There are many ways to create such conversational agents, and they are not technically very difficult to implement, especially using VoiceXML. One of the most innovative solutions on the market is Hex Agent Technology that enables agent designers to implement an agent with only 20% of the effort that is needed to code a conversational agent in VoiceXML.

Hex Agents are created in the Hex Agent Creator, a graphical user interface that makes it possible for agent creators to focus on the design rather than the programming of the agent. The Creator builds the agent, which is run on the Hex Agent Server. The Hex Agent Server subsequently generates the VoiceXML scripts and grammar for the voice browser.

The foundation of Hex Agents is called a template. The templates describe what triggers a particular response with the agent and is the basis of the agent's grammar. For example, the agent creator might define a *mayday* template. This template would be triggered when the agent hears somebody calling in emergencies. The templates can also be dynamic, i.e. the agent creator defines a template that is a placeholder for a call sign. The relevant call signs for all the aircraft in the airspace are then fetched from the FDPS runtime. That way the grammar would always include call signs of every aircraft in the airspace because they are likely to be heard but not include all possible call signs. This makes the matching much easier for the ASR.

When a template is triggered, the agent responds appropriately. The agents have various ways of responding. The most common response of a conversational agent is a vocal response to keep the conversation going. In ATC, a vocal response disrupts the flow of communication and thus should be used carefully. However, the agent can also input what he hears into the FDPS or trigger a more appropriate response to the air traffic controllers' monitor for example if it finds anything suspect. Using the emergency example from above, the agent might send a warning to all appropriate authorities and rescue organizations in the area immediately when it hears a distress call.

As an example, a template could include just a call sign. When a call sign is heard the agent fetches the relevant data from the FDPS. Then it listens to the communication, e.g. comparing a status report to the expected report and if everything is in order, sends an OK signal to the controller.

A demonstration Hex Agent for ATC, the ATCAgent, was designed for the purposed of administering a trial. The ATCAgent listens for the users' inputs and responds accordingly. That way it simulates the functionality of the system we are proposing. A simulation is sufficient to administer a *Wizard of Oz* usability test. The trial is described in the next chapter.

VIII. Trial

In Baum's *Wizard of Oz*, the wizard deceived his citizens to believe he had superpowers. In the same way, a simulation of an LTS can make the users believe that they are using a fully functional application if the simulation is convincing enough.

Following the *Wizard of Oz* method the demonstration was administered to a small number of controllers. The participants acted out scripted conversations with the agent and the agent responded to simulate the functionality.[2] That way we demonstrated the functionality of the system to the participants so that they could understand the system and give feedback on the idea of using LTSs in ATC communication. The trial was primarily done to investigate three things:

1) *Correctness* of the ATCAgent, which was measured with recognition rate.
2) *Reliability* of the ATCAgent, which was measured with hit/miss/false alarm rate.
3) *Satisfaction* of the Air Traffic Controllers, which was measured with a questionnaire.

The results were analyzed to show how the controllers reacted to the system and how the system reacted to them. This made it possible for us to identify things that should be emphasized in further development.

The functionality was designed by analyzing typical air traffic control tasks using scenarios and Unified Modeling Language (UML) sequence diagrams as described below.[3,8]

A. Scenarios

The trial used a few scenarios from the Oceanic Environment and is designed for case 3 in Fig. 1: Using LTS to communicate information from FDPS to pilot.

Here, we will describe how we analyzed one scenario to define the functionality of the ATCAgent.

The characters in our scenarios are:

- *Paula*, pilot of Fairline (FAL) 904.
- *Arnold*, air traffic controller.
- *Felicity*, controller from an adjacent facility.
- *Roger*, radio operator.

1. Scenario: Handover—Current Situation

This scenario describes how Arnold receives information on incoming flights to his sector and plans how to handle them when they arrive.

Please read the text with the UML sequence diagram in Fig. 5 and use the numbers in the scenario for reference.

The scenario describes how Arnold and Paula communicate through Roger from the point when Arnold first hears about Paula's flight entering his airspace until Paula has received a new clearance from him, relayed by Roger.

Felicity calls up Arnold. She has an estimate on a flight, which is half an hour from entering the airspace.

Felicity (1): "*Good morning, Reykjavik. I have an estimate on Fairline 904*."

Arnold (2): "*Fairline 904 go ahead*."

Felicity (3): "*Fairline 904 estimating 60 north 40 west at 0804, flight level 340, MACH 080, route 62 north 30 west 63 north 20 west 63 north 10 west ISVIG*."

Arnold enters the information into the FDPS (4) and reads it back to Felicity (5): "*Fairline 904 estimating 60 north 40 west at 0804, flight level 340, MACH 080, route 62 north 30 west 63 north 20 west 63 north 10 west ISVIG.*"

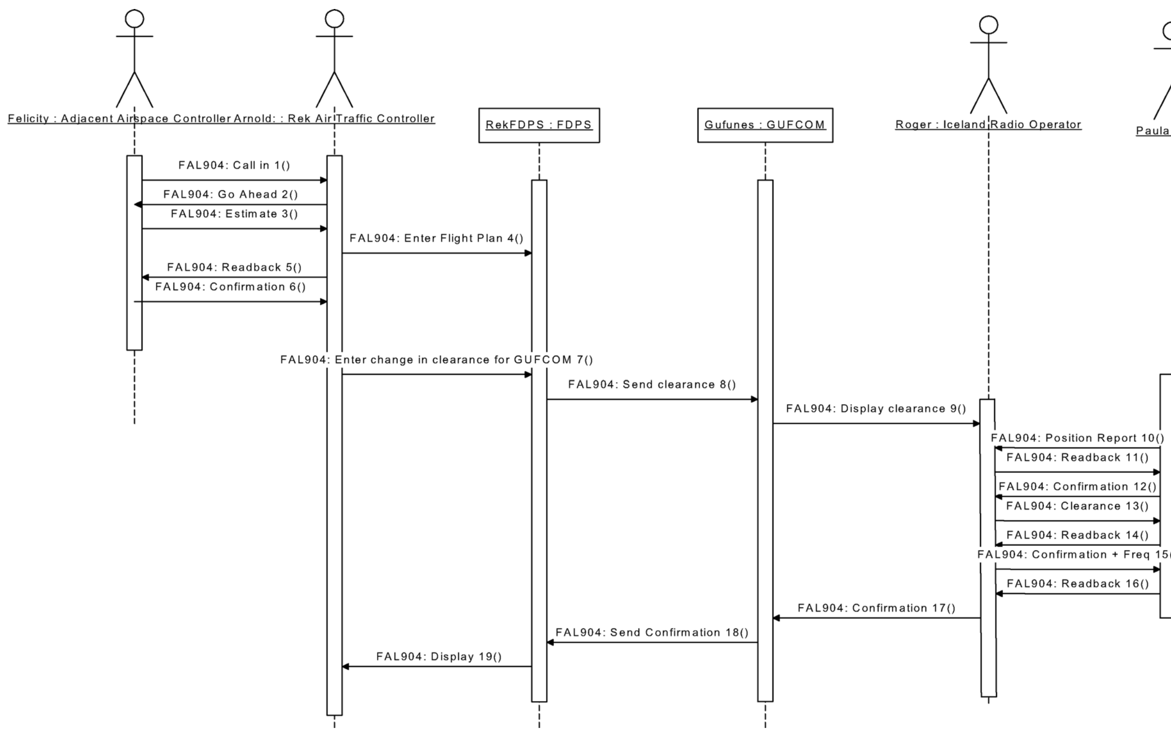

Fig. 5 Handover— current situation.

Felicity (6): "*This is correct. Thank you and goodbye.*"

Arnold: "*Thank you.*"

Arnold makes a decision that FAL904 needs to change flight level to 380 and enters it into the FDPS, which it sends to Roger, to be relayed to Paula when FAL904 enters the airspace.

Arnold (7–8):

```
TGC837 031515
FF BICCZZZX
031515 BIRDZOZF
(CLE-FAL904-REYKJAVIK OAC CLEARS FAL904 AT 62N30W CLIMB TO FL380)
```

Roger has all information on flights in front of him on his workstation (GUFCOM) (9). He receives the clearance and prepares for FAL904 to enter the airspace.

Entering the Icelandic airspace, Paula calls up Iceland Radio (10): "*Iceland Radio, Fairline 904, with you overhead 60 north 40 west at 0803, flight level 340. Estimating 63 north 30 west at 0839. Next 63 north 20 west.*

Roger responds with a read-back (11): "*Roger, Fairline 904, good morning. 60 north 40 west at 0803, flight level 340. Estimating 63 north 30 west at 0839. Next 63 north 20 west.*"

Paula confirms the readback (12): "Affirmative, Iceland Radio."

Roger now gives the new clearance (13): "*Fairline 904 from Reykjavik Oceanic. At 63 north 30 west climb to flight level 380*."

Paula reads back the new clearance (14): "*Fairline 904, at 63 north 30 west climb to flight level 350*".

Roger confirms the new clearance (15): "*Fairline 904 read back correct*".

Roger sends a message to Arnold's FDPS that the clearance has been relayed (17–18):

```
GTC512 031518
FF ENOBZOZX BIRDZQZX KIADXAAY BICCYSYW
031518 BICCZZZE
(RBK-CLE-FAL904-REYKJAVIK OAC CLEARS FAL904 AT 62N030W CLIMB TO FL380)
```

```
FAL904 RB
VD3JY/1518 5337
```

The FDPS displays to Arnold that the clearance has been delivered (19). Arnold believes FAL904 is climbing to FL380 whereas FAL904 is really climbing to FL350.

2. Scenario: Handover—The LTS Relays the Communication and Catches a Potentially Dangerous Situation

In this interaction we have added the LTS into the communication as described in Fig. 1, case 3. This makes the communication between Arnold and Paula more direct and Roger in effect redundant. After the handover from Felicity, Arnold enters the clearance into the FDPS. The LTS receives the clearance and waits for FAL904 to enter the airspace. The numbers in the text refer to Fig. 6.

When Paula checks in with Icelandic ATC the LTS recognizes the call sign and retrieves the clearance for FAL904 from the FDPS (9-10). For Paula the communication is exactly as described before until the point where the LTS has relayed the change in altitude and recognizes a discrepancy in the read back (17): "*Fairline 904. Correction, cleared to climb to flight level 380*".

Paula responds (18): "*Fairline 904, climb to flight level 380*".

The LTS compares the read back to the clearance (19) and confirms the new clearance (20): "*Fairline 904 read back correct*".

The LTS inputs the information to the FDPS that the clearance has been confirmed as before (21–22). The FDPS displays the confirmation to Arnold as before. Now there is no discrepancy.

B. Trial Environment

The above scenarios were used to define the functionality of the ATCAgent. In the trial, the ATCAgent simulated the part of the LTS as described in the scenarios. In the trial roles were assigned to the participants and administrator, to act out with the LTS. To make sure that the ATCAgent was following the conversation it always gave a response

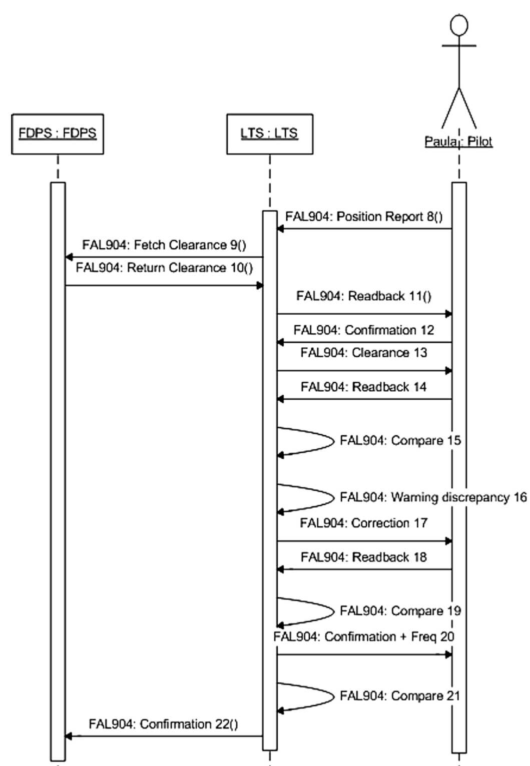

Fig. 6 Handover—includes the LTS.

to the input from participant and controller. This would of course not be acceptable in normal circumstances since it slows down the communication substantially. However, it was more important for this trial to keep everyone synchronized than to keep up the pace of the communication. The ATCAgent also gave warnings when it did not understand the user and responded appropriately when discrepancies were made in the communication according to script in order to keep the communication on track.

The trial was administered to seven Air Traffic Controllers. The seven participants were on average 34 years old with 11 years of experience. The gender distribution was roughly equal. Their primary work is in the oceanic or en-route environment. On average each trial run lasted around 30 minutes.

As described in Fig. 4, during the trial the ATCAgent was called up over the phone. The utterance of the participants was received by an ASR that matched it, if possible, with known words in the grammar from the ATCAgent. The Voice Browser then sent an HTTP request to the Hex Agent Server with the input. The ATCAgent returned a VoiceXML document with the appropriate response and the grammar for the next input. The answer was delivered to the participants using a TTS generator.

C. Results

The trial was primarily done to investigate three things:

1) *Correctness* of the ATCAgent which was measured with recognition rate.
2) *Reliability* of the ATCAgent which was measured with hit/miss/false alarm rate.
3) *Satisfaction* of the Air Traffic Controllers which was measured with a questionnaire.

During the trial two problems occurred that in effect ruined the trial in 3 out of 7 cases. First of all the telephones used for the trial stopped working properly and secondly static appeared on the phone line. This caused the ATCAgent not to hear the participants' input and thus it was unsuccessful in demonstrating the intended functionality. In the discussion below we will distinguish between the results for successful vs. unsuccessful trial runs.

1. ATCAgent Correctness

This difference between the successful and unsuccessful trial runs was best seen when we analyzed the results of the ASR. The automatic speech recognizer returns two types of events: normal events that describe that it recognizes the input from the participant and error events that describe different reasons for the voice recognizer not recognizing the input.

In the successful trials the voice recognizer returned error events in 15% of the cases versus 31% of the cases in the unsuccessful trial runs. This meant that on average the number of error events in each of the unsuccessful trial runs were more than 4 times more common than in the successful trial runs. On average the voice recognizer returned errors in 25% of the cases.

Even if the response of the ASR were error events in 25% of the cases the ATCAgent responded appropriately in 89% of the cases. It was designed to ask the participants to repeat their input if it didn't understand and thus get back on track quickly again in case of an error. In the case of the successful trial runs the correctness of the ATCAgent went up to 97% and, surprisingly, only down to 87% in the unsuccessful runs.

2. ATCAgent Reliability

Figure 7 depicts the origins of the errors in the successful trial runs. The majority of these are in the recognition part which can be traced back to the fact that the participants were not native English speakers, they were using inferior recording equipment (a telephone) and talking over a telephone line to the ASR. The former issue is being addressed by ASR developers. The latter can be greatly improved with better equipment.

It is interesting to note that the proportion of errors that could be assigned to the ATCAgent are not different for successful and unsuccessful trial runs. The main difference is that for the successful trial runs almost no errors were caused by interaction between the two. This indicates that the reliability of the ATCAgent is not dependent on the reliability of the input.

3. Controller Satisfaction

The difference between the successful and unsuccessful trial runs was further visible in the response of the two groups to the ATCAgent. The unsuccessful participants were much more negative and skeptical of the ATCAgent and

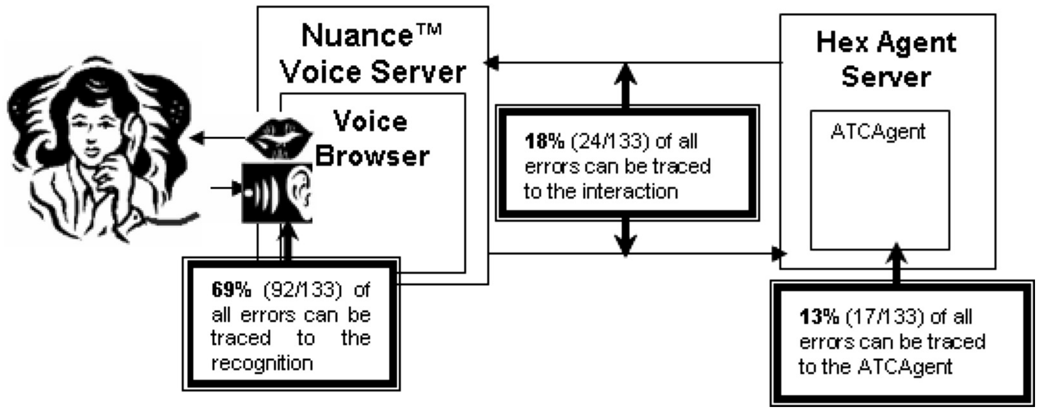

Fig. 7 Origins of error events.

its possibilities than were the successful participants. A questionnaire was administered after the trial that asked the participants to assess the ATCAgent and when we drew up the correlation between the assessment and the number of error events that the participants experienced during the trial we got an indication of a very clear correlation (see Fig. 8). The sample is however too small to calculate correlation statistics.

In Fig. 8 the dissatisfaction is the summed up answers of the questionnaire that was administered. The questionnaire included six questions, where an answer of one was very positive and five was very negative. This means that the most negative score was 30, and the most positive score was six. The error events experienced were based on a counted number by the researcher during the trial since not all error events in the ASR resulted in an error event in the ATCAgent and the experience of the participants was based on the number of error events they actually heard rather than the number generated by the system.

As can be seen in Fig. 8 the participants that took part in the successful trial runs were very positive towards the ATCAgent and its possibilities. When discussing the satisfaction of the controllers we feel justified in only looking at the response of those that took part in the successful trial runs, since the participants in the unsuccessful trial runs did not experience the functionality of the ATCAgent that we wanted them to assess.

The controllers in the successful trial runs expressed a positive attitude towards the ATCAgent. They thought it would increase safety and was reliable and said that they would like to have an agent like this one for support.

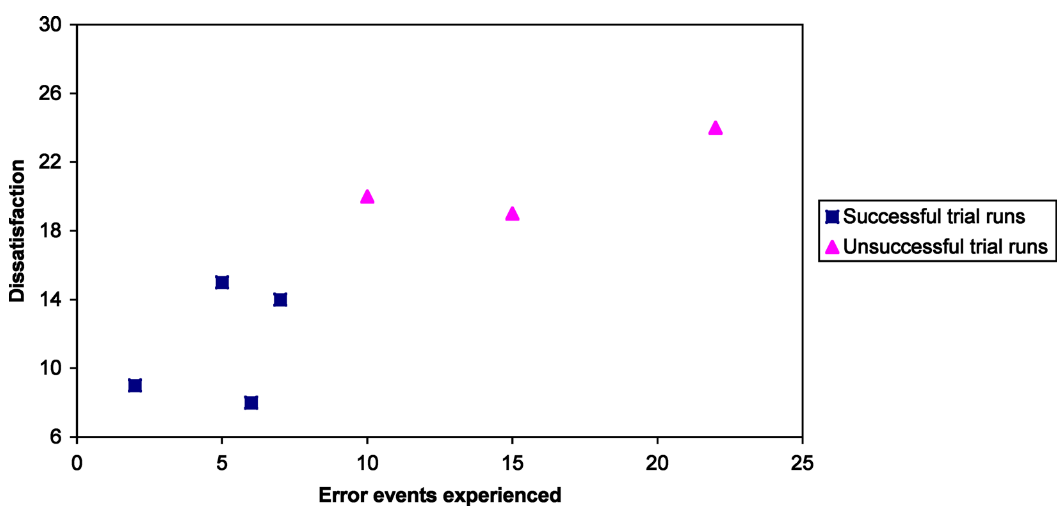

Fig. 8 Correlation between dissatisfaction and number of error events experienced.

In short, the trial results gives just cause to believe that further research should be carried out on using language technology in Air Traffic Control.

IX. Conclusion

We feel that the results of the trial were encouraging and therefore that our idea of using language technology in Air Traffic Control (ATC) should be investigated further. The system's response to the controllers was above expectation; it responded appropriately in 89% of the cases. Understandably, this is not acceptable in a safety critical environment but from a demonstration agent this result can be greatly improved. The following topics should be investigated further:

First, the Language Technology System needs to be adapted into the ATC technical environment, most importantly to read and write data to the Flight Data Processing System (FDPS) database.

Second, all types of communication between air and ground need to be thoroughly documented. The first step in this has already been made with the definition of a corpus.[17,28] When this is done, the different types of communication can be analyzed for keywords that the Language Technology System (LTS) listens for and recognizes. If this is done properly it will make the design of the LTS easier because the flow of the communication defines the system.

Third, we need to make sure that the voice server is able to cope with the environment and that it gives the LTS what it needs. Configuration should be focused most importantly on:

- *Filtering* of radio static, which is not a trivial matter, especially in High Frequency radio. Because of the high level of interference this needs extensive testing.
- *Types of input*—Analyze when to expect long/short correspondence and how to react differently so that it allows the user to take a breath.
- *Recognition rate*—Careful testing can show what the perfect balance between too many rejects and too many falsely recognized inputs is.

Fourth, the design should be able to allow the LTS to stand by without losing the state it is in between communication.

Fifth, there needs to be a way for the LTS to display warnings to the user either through the FDPS or a separate program that can pop up windows or send other types of indications to the users. These indications should be carefully designed in cooperation with the users.

To introduce language technology into ATC an important first step is to develop a fully functioning Language Technology System for the oceanic environment. This can serve two equally important functions:

- Continue the research of the technology needed to make such a Language Technology System function flawlessly.
- Build a reputation for a Language Technology System in this field.

Every air traffic controller in training begins by controlling the oceanic airspace and then improves her skills in order to control more complex airspaces such as in the tower. The LTS has to go the same route to build trust with the people it is aimed to support, the controllers. It should start in training (as described in[27]) and then move to a live airspace, first in the oceanic environment. With further development, an LTS would be able to assist where it is most needed, in time sensitive situations where the information is critical and no time for read backs and double checks i.e. during approach and landing.

When an LTS like this has proven its worth, further research should be done on how artificial intelligence could be incorporated into the system, so that it can make recommendations on controlling the airspace in addition to monitoring the communication. For example, the LTS can easily do the calculations needed to clear a request for altitude change from an aircraft. The conflict probe that is already in place in the ATC Center in Reykjavik is a step in that direction.

Acknowledgments

The Icelandic Civil Aviation Administration and Hex Software supported this research.

References

[1]Andersson, E. A., Breitenbach, S., Burd, T., Chidambaram, N., Houle P., Newsome, D., Tang, X., and Zhu, X., *Early Adopter VoiceXML*, Wrox Press Ltd., Birmingham, UK, 2001.

[2]Balentine, B. and Morgan, D. P., *How to Build a Speech Recognition Application*, 2nd ed., Enterprise Integration Group Press, San Ramon, CA, 2001.

[3]Booch, G., Rumbaugh, J., and Jacobsen, I., *The Unified Modeling Language User Guide*, Addison Wesley, Reading, MA, 1999.

[4]Broughton, M., *Measuring the Accuracy of Commercial Automated Speech Recognition Systems During Conversational Speech.* From the Workshop on "Virtual Conversational Characters: Applications, Methods, and Research Challenges," 29th November 2002, Melbourne, Australia http://www.vhml.org/workshops/HF2002/papers/broughton, accessed January 2003, 2002.

[5]Cane, J., *Comparing Dragon NaturallySpeaking and IBM ViaVoice Gold, ENW International*, http://www.enw-ltd.com/vv-dragonComparison.htm, accessed January 2003, 1998.

[6]Cardosi, K., An Analysis of En Route Controller-Pilot Voice Communications. Final Report No. DOT-VNTSC-FAA-93-2, 1993.

[7]Cardosi, K., Falzarano P., and Han S., *Pilot-Controller* Communication *Errors: An Analysis of Aviation Safety Reporting System (Automatic* Speech *RecognizerS) Reports*. DOT/FAA/AR-98/17, http://www.volpe.dot.gov/opsad/docs/asrs1.doc, accessed January 2003, 1998.

[8]Carroll, J. M., *Making Use: Scenario-based Design of Human–Computer Interaction*, MIT Press, Cambridge, MA, 2000.

[9]Dunbar, M., McGann, A., Mackintosh, M- A., and Lozito, S. *Re-examination of Mixed Media Communication: The Impact of Voice, Data Link, and Mixed Air Traffic Control Environments on the flight Deck*, NASA/TM–2001-210919, Moffet Field, CA, 2001.

[10]EUROCONTROL—European Organization for the Safety of Air Navigation, *Automatic Dependent Surveillance (ADS)*, http://www.eurocontrol.int/eatmp/work/ads.html, accessed January 2003, 2002.

[11]FAA—Federal Aviation Administration, *Study 3.12 Oceanic Controller Job/Task Analysis; Final Report, Volume IV, J/TA Summary & Interfacility Comparisons*, FAA, Washington, D.C., 1994.

[12]FAA—Federal Aviation Administration, *Advanced Technologies and Oceanic Procedures (ATOP) Result Report: Oceanic Controller Job/Task Analysis (J/TA) Year 2000 Revisions*, FAA, Washington, D.C., 2000.

[13]FAA—Federal Aviation Administration, *Strategic Plan For Oceanic Airspace Enhancements and Separation Reduction*, Draft 1.0, FAA, Washington, D.C., 2002a.

[14]FAA—Federal Aviation Administration, *Blueprint for NAS Modernization. 2002 Update*, FAA, Washington, D.C., http://www1.faa.gov/nasarchitecture/blueprnt/2002Update/PDF/2002Update_complete.pdf, accessed January 2003, 2002b.

[15]Finlay, J. and Dix A., *An Introduction to Artificial Intelligence*, UCL Press, London, 1996.

[16]Fraser, T., *World Airlines Lost about $13 Billion in 2002*, Associated Press. http://www.montereyherald.com/mld/charlotte/business/4893873.htm, accessed January 2003, 2003.

[17]Godfrey, J. J., *Air Traffic Control Complete.* ISBN: 1-58563-024-1 http://www.ldc.upenn.edu/Catalog/LDC94S14A.html, *accessed January 2003*, 1997.

[18]Hausser, R. R., *Foundations of Computational Linguistics: Human–Computer Communication in Natural Language*, 2nd ed., Springer, Berlin, 2001.

[19]Jurafsky, D. and Martin J. H., *Speech and Language Processing*, Prentice Hall, Upper Saddle River, NJ, 2000.

[20]Prizno, V., *Data-Linked Pilot Reply Time on Controller Workload and Communication in a Simulated Terminal Option*, DAA/FAA/AM-01/8.

[21]Ragnarsdottir, M. D. *"Do You Copy?" Using Language Technology to Support Communication in Air Traffic Control*, Unpublished M.Sc. Thesis. University of Iceland, 2003.

[22]Rosen, S. and Howell, P., *Signals and Systems for Speech and Hearing*, Academic Press, London, UK, 1991.

[23]Rurup, A. Y., Machin, C., Hawley, M., Pandya, A. and Dell'Orto, L., "A Comparative Study of ADS-B Link Costs by the FAA and by Eurocontrol", FAA-Eurocontrol Workshop, Toulouse, 3–5th June 2002, http://www.eurocontrol.int/eatmp/events/docs/avionics/Rurup2.pdf, accessed January 2003, 2002.

[24]Schaefer, D., *Context-Sensitive Voice Recognition in the Air Traffic Control Simulation*, EEC Note No. 02/2001, http://137.193.200.177/ediss/schaefer-dirk/meta.html, accessed January 2003, 2001.

[25]Tober, E. D., Marchand, R., and Ferrans, J., *The VoiceXML Forum's VoiceXML Tutorials*, Voice XML Forum, http://www.voicexml.org/tutorials/intro1.html, accessed January 2003, 2001.

[26]Van Santen, J. P. H., Sproat R. W., Olive, J. P., and Hirschberg J. (eds.), *Progress in Speech Synthesis*, Springer, New York, 1995.

[27]Voice Recognition Trains Next-Generation of Air Traffic Controllers, *Speech Technology Magazine*, http://www.speechtechmag.com/pub/industry/1029-1.html, accessed January 2003, 6 Aug. 2002.

[28]Ward, K., *A Speech Model of Air Traffic Control Dialogue*, Unpublished M.Sc. Thesis, Oregon Graduate Institute of Science and Technology. http://citeseer.nj.nec.com/ward92speech.html, accessed January 2003.

[29]Wickens, C. D., Goh, J., Helleberg, J., and Talleur, D. A., *Modality Differences in Advanced Cockpit Displays: Comparing Auditory and Vision for Navigational Communications and Traffic Awareness*, ARL-02-8/NASA-02-6, http://www.aviation.uiuc.edu/new/html/arl/report_fulltext.html, accessed January 2003, 2002.

[30]Wickens, C. D., Gordon, S. E., and Liu, Y. "*An Introduction to Human Factors Engineering*", Addison Wesley Longman Inc., New York, NY, 1997.

[31]Wickens, C. D., and Hollands, J. G., "*Engineering Psychology and Human Performance*", 3rd ed., Pearson Education, Upper Saddle River, NJ, 2001.

Ellis Hitt
Associate Editor