Agent aided aircraft maintenance

Onn Shehory, Katia Sycara, Gita Sukthankar and Vick Mukherjee
School of Computer Science
Carnegie-Mellon University
{onn,katia,gitars,vick}@cs.cmu.edu

Abstract

Aircraft maintenance is performed by mechanics who are required, by regulation, to consult expert engineers for repair instructions and approval. In addition to their own experience, these engineers rely on external information sources, which are often inadequately indexed and geographically dispersed. The timely retrieval of this distributed information is vital to the engineers' ability to recommend repair procedures in response to the mechanics' requests. This problem domain is well suited for a multi-agent system: it consists of distributed multi-modal information which is needed by multiple users with diverse preferences. In this paper, we describe an implementation of such a system, using the RETSINA multi-agent architecture. Such an implementation reinforces the importance of multi-agent systems, and in particular the usefulness of the RETSINA infrastructure as a basis for the construction of such systems.

1 Introduction

Agent aided information retrieval and decision support have been a focus of the agent research community for several years. Although theories and simulations of multi-agent information retrieval systems abound, real applications are scarce. One reason for this scarcity is the lack of real problem domains which are sufficiently distributed, rich in information and complexity, without being too complex. Such domains would have facilitated the development of application systems and allowed the examination of the applicability and usefulness of multi-agent technology developed in research labs.[1]

In this paper we present such an application. Based on our experience with the RETSINA multi-agent infrastructure [7], we implemented a system to solve an existing real-world problem. Specifically, we developed a multi-agent framework that provides information retrieval and analysis in support of decision making for aircraft maintenance and repair in the U.S. Air Force. Although the solution was developed for a specific type of aircraft, the agents and the interactions among them were designed to apply to a range of similar maintenance scenarios.

Maintenance of complex vehicles such as aircraft and seacraft is a complicated task. Usually, this task involves several people, including mechanics, inspectors, engineers and possibly other experts. In addition, the amount of information involved in the process is very large. This information is typically multi-modal and available in multiple storage media (e.g., hard-copy and electronic). Commonly, the information is also geographically distributed. A major goal of the maintenance process is to perform the most appropriate repair procedures in the most efficient way and in the shortest time. Since maintenance-related processes rely on relevant information, comprehensive and timely information delivery to the individuals involved in the maintenance can significantly benefit the process.

The papers is organized as follows: we first present the properties of the problem in section 2. Then, in section 3, we briefly describe the RETSINA multi-agent infrastructure, which we used as the basis for the development of our solution. In section 4 we describe the solution we provide for the problem at hand. Finally, in section 5, we discuss the advantages and the limitation of the solution and indicate future research directions.

2 The problem

Standard aircraft maintenance in the U.S. Air Force involves the following steps: when inspecting an aircraft, a mechanic who indicates a possible discrepancy must consult an engineer to decide on the required repair. The engineer, in turn, may need to consult external sources of information. These include manuals, historical maintenance data and other, remotely-located experts. (Until recently, no automation was introduced to the consultation processes of this information-rich environment.) Hard-copy repair manuals are used, thus search for relevant information may be both time consuming and incomplete. Historical data (e.g., records of previous similar repairs) is scarcely used, since it is stored in paper format with no search mechanisms, usually only kept for short periods of time, and may be distributed along remotely located service centers. Expert engineers may be located remotely, and their advice is available by voice or fax messages. These are usually delayed for hours or days. All of these factors contribute to a slow, inefficient inspection and maintenance processes.

The repair process consists of the following steps:

[1] Such research is presented in multiple publications. For examples, surveys and pointers to additional research refer to [2, 3, 4, 5].

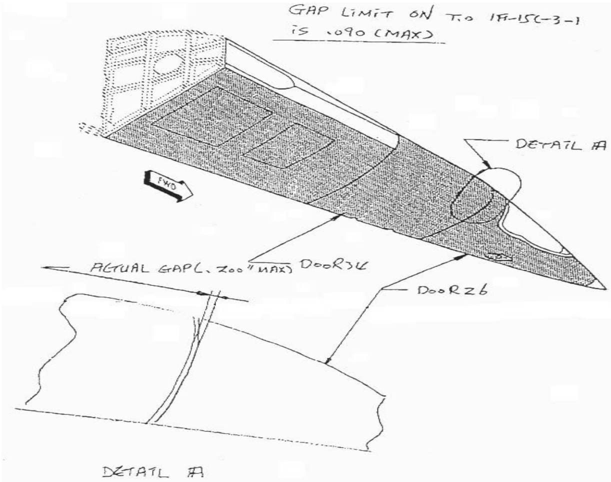
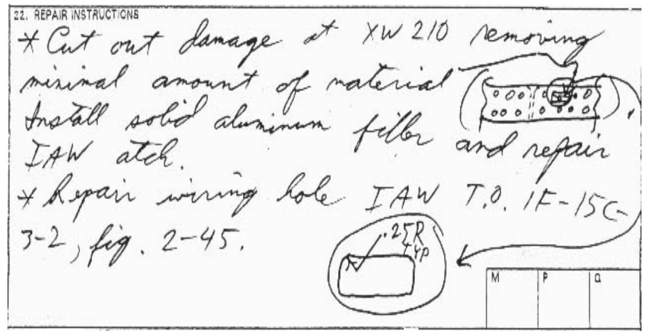


Figure 1: Part of a graphical description attached to a form.

- An aircraft arrives at a maintenance service center for regularly scheduled maintenance, which must be completed within a limited time period. This period varies for different aircraft.
- Mechanics inspect the aircraft and locate discrepancies. For each discrepancy a mechanic finds, he/she performs the following:
  - In order to describe the discrepancy in an adequate manner, the mechanic, in addition to relying on his/her experience, consults manuals and other, more experienced mechanics.
  - The mechanic fills in a 202a form, which is a standard Air Force form for reporting aircraft discrepancies. To the form, the mechanic may attach supporting information such as graphical illustrations (as in Figure 1).
  - The 202a form is sent to an engineer for advice on the required repair and authorization to perform repair procedures. Engineers may be located remotely.
- An engineer, upon receipt of a 202a form, performs the following:
  - Uses own experience, manuals and historical repair information to find the appropriate repair for the discrepancy described in the 202a form.
  - Fills in a 202b form, which is a standard Air Force form for discrepancy repair instructions. To this form the engineer may attach graphical illustration to clarify the required repair procedure.
  - Files 202a and 202b forms for future use as historical repair information.
- Upon receipt of a 202b form from an engineer, the mechanic performs the repair as instructed.

The current repair process has several problems, as described below:

- The majority of the information, both historical repair information and manuals, is found in hard-copy, and part of it is hand-written.


22. REPAIR INSTRUCTIONS

* Cut out damage at XW 210 removing minimal amount of material. Install solid aluminum filler and repair IAW atch.
* Repair wiring hole IAW T.O. 1F-15C-3-2, fig. 2-45.

.25R TYP

M	P	Q

Figure 2: Part of a hand-written 202b form.

- Mechanics and engineers spend precious time on:
  - Browsing manuals and searching for historical repair information.
  - Drawing graphical discrepancy and repair illustrations.
  - Mechanics spend time, idle, waiting for 202b forms to arrive from engineers in reply to their 202a forms.
- Old, valuable discrepancy and repair information is not used.
  - When stored in a remote location, historical information is inaccessible.
  - When stored locally, hard-copy information is difficult to browse through, especially when looking for keywords within free text sections of the 202 forms. With regards to the information needs of mechanics, using paper manuals during inspection for diagnosis, is inefficient and at times impossible due to physical constraints of the inspection environment.[2]
  - Hand-written information (as seen in Figure 2), both from historical forms and from the current 202 forms, may have a limited comprehensibility. This problem intensifies due to deterioration in the quality of such information when it is transmitted via fax machines or photo-copied.
  - Historical forms are kept only for two years, then disposed.
- Time and effort are spent on paperwork and filing. This time should instead be used for diagnosis and repair.
- As a result of being held in paper format, the information in the manuals is not always updated in a consistent manner.

To summarize, the problem with which we deal consists of decision support in a physically distributed environment, rich in multi-modal information. This is the type of problem for which the RETSINA multi-agent system is most appropriate.

[2] During aircraft maintenance, mechanics may need to enter narrow spaces where using a hard-copy manual is impracticable.

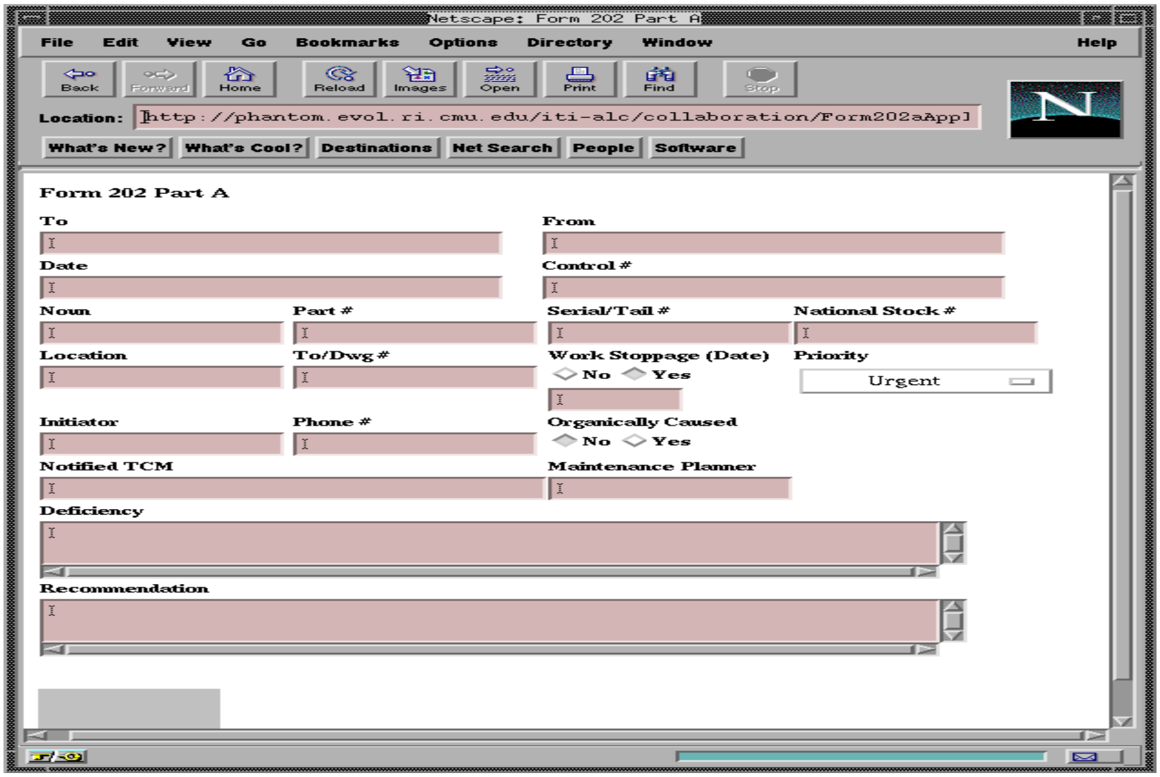


Figure 3: A HTML format 202a form.

3 The RETSINA multi-agent infrastructure

RETSINA [6, 7, 8] (REusable Task-based System of Intelligent Networked Agents) is a multi-agent infrastructure that was developed for information gathering and integration from web-based sources and decision support tasks. Each agent in RETSINA specializes in a specific class of tasks. When the agents execute tasks or plan for task execution, they organize themselves to avoid processing bottlenecks and form teams to deal with dynamic changes in information, tasks, number of agents and their capabilities.

In RETSINA, the agents are distributed and execute on different machines. Based on models of users, agents and tasks, the agents decide how to decompose tasks and whether to pass them to others, what information is needed at each decision point, and when to cooperate with other agents. The agents communicate with each other to delegate tasks, request or provide information, find information sources, filter or integrate information, and negotiate to resolve inconsistencies in information and task models. The system consists of three major classes of agents:[3] *interface* agents, *task* agents and *information* agents.

Interface agents interact with users receiving their specifications and delivering results. They acquire, model and utilize user preferences. The main functions of an interface agent include: (1) collecting relevant information from the user to initiate a task, (2) presenting relevant intermediate and final results, (3) requesting additional information during task execution. The interface agents hide the underlying structural complexity of the agent system.

[3] Deviations from this strict categorization may occur. For instance, there may be a hybrid of two types, such as interface+task agent.

Task agents formulate plans and carry them out. They have knowledge of the task domain, and which other task agents or information agents are relevant to performing various parts of the task. In addition, task agents have strategies for resolving conflicts and fusing information retrieved by information agents. A task agent (1) receives user delegated task specifications from an interface agent, (2) interprets the specifications and extracts problem solving goals, (3) forms plans to satisfy these goals, (4) identifies information seeking subgoals that are present in its plans, (5) decomposes plans and cooperates with appropriate task agents or information agents for plan execution, monitoring and results composition.

Information agents provide intelligent access to a heterogeneous collection of information sources. They have models of the information resources and strategies for source selection, information access, conflict resolution and information fusion. Information agents can actively monitor information sources.

4 The solution design

Given its properties, we found the RETSINA infrastructure appropriate to solve the problem considered. Below, we analyze the information sources and then describe the agents developed to handle these sources and the interactions among them. Note that some of these agents are hybrid agents, though all share the basic RETSINA agent (internal) architecture.

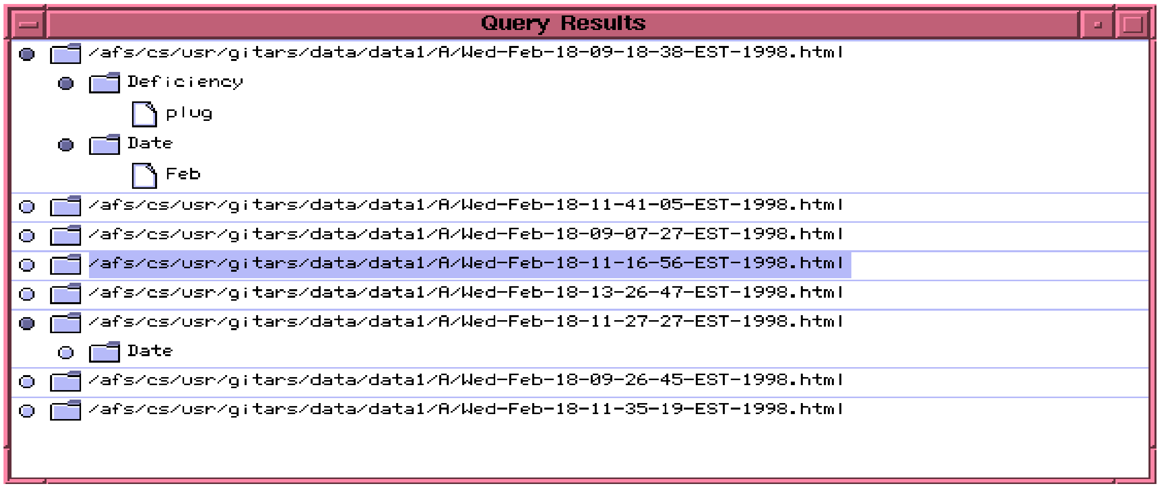


Figure 4: The result of the query, as presented by the form agent, is a list of matched forms. The matching fields and keywords can be viewed by clicking on a specific form entry.

4.1 Information sources

There are several sources of information relevant to the inspection and maintenance processes. Note that at this stage of system development some effort was already put into converting information into electronic formats. This is expressed in the description of the information sources.

- The current 202a form is the form filled in by the mechanic for the current fault encountered during inspection. A 202a form, in the new, electronic version, is in HTML format (see Figure 3) and has a unique identifier. Graphical illustrations, free text, voice messages and digital photographs can be attached to the form (all held in files of standard graphical and audio formats, e.g., JPEG). The form is located on the wearable computer held by the mechanic.

- Historical 202 (a and b) forms are collections of forms from previous aircraft inspection, consultation and maintenance procedures. Some of the historical forms were typed in manually (from paper forms). These, in addition to the attached graphical file, may have an attached text file (ascii) that includes the text that appears in the graphical illustration. Each service center[4] has its own collection of historical 202 forms on local disk. Currently, there is no indexing system for these forms.

- Manuals are partly available in HTML format while other parts are in PDF format. It is not clear to what extent are the manuals indexed, but indexing should be an inseparable part of them in the future. Manuals are found on local disks of every service center.

- Expert engineers may be located remotely and not necessarily available on demand. They provide unstructured voice, text and graphical information relevant to problems addressed to them.

In the aircraft maintenance system we use the above sources of information.

[4] Some type of planes have a single service center. In such cases distribution of historical information does not exist. Nevertheless, we develop our solution to fit to a range of aircraft maintenance scenarios, where distribution is an issue.

4.2 Agent types

We developed an agent system that provides information gathering, filtering and fusion in support of maintenance decisions. The system has been implemented and is tested with real data from a U.S. Air-base. It will be fielded and installed at the Air-base in a few months. The system is comprised of three types of agents, as follows:

- A *form agent*. Its role is to analyze the current 202a form it receives from a mechanic, characterize the problem presented in the form, and request information which is relevant for the solution of this problem. Upon the receipt of such information it merges, filters and presents it in a meaningful and comprehensible manner to the engineer. A form agent is a hybrid of a task agent and an interface agent. It combines an interface to the engineer (typical to interface agents), task decomposition and delegation and some complex information fusion (typical to task agents).

- A *history agent*. Upon request from another agent (probably a form agent), it searches for historical forms which are relevant to the problem presented in the request. The relative relevance of the forms is computed and forms that pass some relevancy threshold are sent as an answer to the requester. A history agent is an hybrid of an information agent and a task agent. It combines capabilities of information gathering and filtering (which is typical to information agents) as well as some complex analysis of the information (which is typical to task agents).

- A *manuals agent*. Upon request from another agent, it locates, in the manuals, data relevant to the problem presented in the request. Index search is assumed. A manual agent is an information agent.

The details of the functionality that we specifically developed and implemented for the agents described above are as follows.

4.3 Agent functionality

The history agent and the manuals agent are both, to a large extent, information agents. As such, they can receive requests for information and reply with the required information. Both of them receive the same type of requests—a single shot query. The content of the query includes one or more part-numbers and one or more fault descriptors for each fault in the request.

The history agent parses each historical form to which it has access, usually limited to local archives, searching for complete or partial matches. (We leave some flexibility to allow a search using an ontology to increase hits on less obvious cases). The forms which match the designated problem are inserted into the content of the reply message. In contrast, the manuals agent does not need to perform parsing. It only has to search, using indexes, through the manuals. Matches are inserted into the content of the reply message.

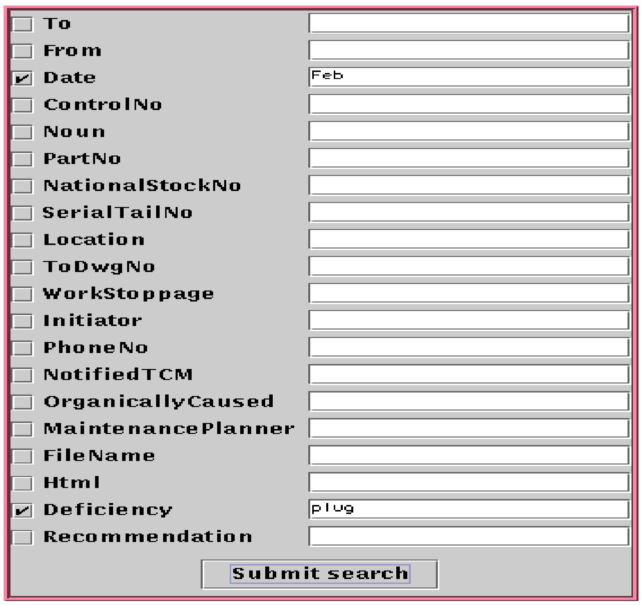


Figure 5: A table for manual submission of requests for keyword searching.

Figure 6: The hand-held computer used by a mechanic.

The form agent, upon request from the engineer (a button click), performs several actions. It first retrieves a list of keywords (of fault descriptors) from local file. Optionally, the engineer may also request for a search manually. This is performed using a field table, where the engineer can insert the requested keywords (a snapshot of this search request window is shown in Figure 5). Then, it parses the current form to find part-numbers and keywords. These are inserted into the content field of an outgoing request. The form agent then first sends (this functionality may be button controlled by user) a request to a manuals agent. If the results are null (or the user requests additional information) the form agent sends a request to a local history agent. If the results are not satisfying, it sends requests to remote history agents (in other service centers). Any non-empty result that arrives during this process is presented on a results window. A snapshot of this window in shown in Figure 4. Initially, the results of the query is a list of the addresses of the forms which were found relevant (according to keyword matches), ordered by relevancy. The user may click on items on this list to find what fields and what keywords resulted in a match. In addition, the user may request to display a selected form. In this event, the form is displayed with the relevant keywords highlighted. Text and graphics should be arranged in a reasonable way (this is still in a development stage). After the automated consultation is finished, the form agent adds the current 202 forms to the historical database of these forms.

4.4 Data processing and flow

Using these three types of agents, we have built a multi-agent system in which each specialized type of agent may have several instances in the system. Below, we describe the processing and flow of information in the computerized system, as seen in Figure 7. The process begins with a mechanic (who is sometimes required to be certified as an inspector), inspecting the plane. The mechanic uses a wearable computer (a Fujitsu 1200 as in Figure 6) with a touch-screen, microphone and a digital camera. When a discrepancy is found, the mechanic fills in an electronic 202a form (Figure 3), and when necessary and practicable, adds voice notes and digital photographs. The 202a form with the attachments is sent to an engineer. At this point, the mechanic waits for a reply from the engineer.

The engineer, with the support of a form agent on his/her workstation, extracts keywords from the 202a form. Using these keywords, the form agent automatically requests for relevant historical forms from history agents and for relevant manual pages from a manuals agent. These requests may also be activated, controlled and edited by the user (engineer). At this point, the form agent waits for the requested information to arrive, in reply to its requests.

History and manuals agents are located on central computer networks of service centers, on which the archival information they need to access is located as well. Upon receiving a request for information, history agents perform a search on the historical 202 forms archive, and conduct a relevancy analysis. They reply with a list of relevant forms, the reason for their selection and the level of relevancy. A manuals agent performs a simple search in an indexed manuals database and replies with the results of this search.

Upon receiving replies from history and manuals agents, the form agent merges results and displays them to the engineer. Using this information the engineer can decide upon the appropriate repair procedure, fill in an electronic 202b form, attach to it graphical description grabbed from manuals and historical forms (or draw new ones, if necessary), and send it to the mechanics wearable computer.

The information flow and processing end when the me-

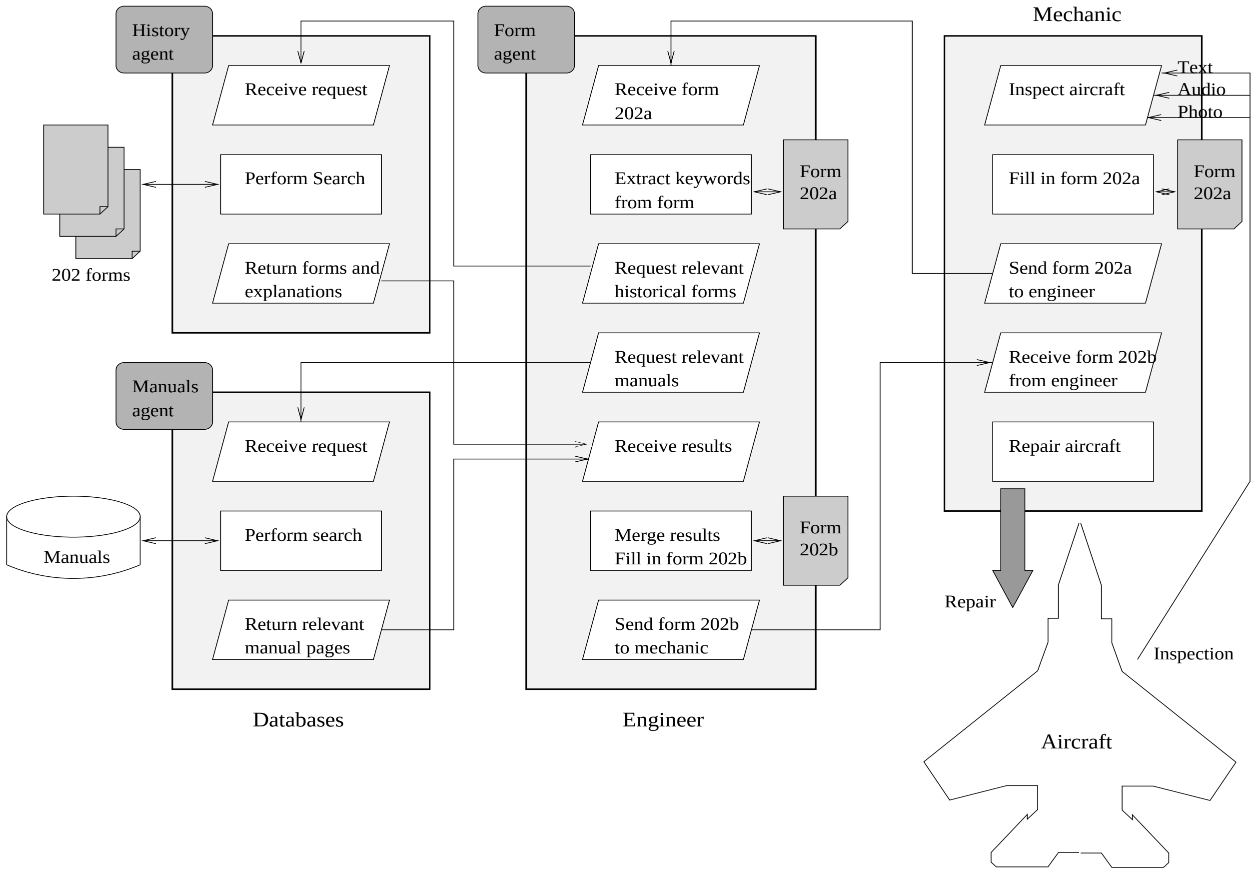


Figure 7: Data processing and flow in the computerized system.

chanic receives and opens the 202b form on the wearable computer. The details in the 202b form and the approval of a repair procedure allow the mechanic to execute the actual repair.

4.5 Multi-agent organization

The description of information flow and processing provides only a partial view of the system developed. At least as important is the way in which the agents are organized to provide the required processing and flow. Therefore, it is necessary to provide a description of the multi-agent organization. Such presentation also provides an insight into the distribution, reuse and scalability of the system. A graphical illustration of the multi-agent organization is presented in Figure 8. As depicted there, multiple mechanics use each a wearable computer in the inspection process to compose a 202a form. These forms are sent to form agents. There may be multiple form agents and each form agent may handle several 202a forms. Each form agent may request information relevant to the forms it handles from multiple history agents. This is necessary since historical archives of 202 forms may be distributed over multiple service centers. Manuals agents and history agent may be requested for information by multiple form agents. In a closed system agents know about one another in advance. Since agents may go down or new agents appear, there is need for open system organization. The RETSINA multi-agent infrastructure allows for such an open system organization through the presence and interaction with middle agents [1].

5 Discussion

The application of a RETSINA multi-agent infrastructure to the aircraft maintenance problem and its implementation provide the following advantages.

- Automatic location and retrieval of relevant information. This information is necessary to provide advice regarding to the potential risk discrepancies pose and support decisions upon the appropriate repair methods.
- Historical repair data is utilized, thus information about discrepancy re-occurrence can be re-used. In addition, long term learning of typical problems of the aircraft can be performed.
- The access to manuals via an electronic, indexed database increases the efficiency, accuracy and completeness of information retrieval from the indexes, thus reduces the probability of mistaken decision for repairs.
- As a result of electronic retrieval, fusion and transmission of information, the average repair time is reduced.

In conclusion, the increased speed and quality of information gathering, filtering and merging for repair decision results in timely and qualitative aircraft maintenance. In addition, since the underlying architecture of the system is an open multi-agent system, it is possible to re-use it for similar domains (and there are myriad distributed repair domains which can benefit from such a system). The openness of the

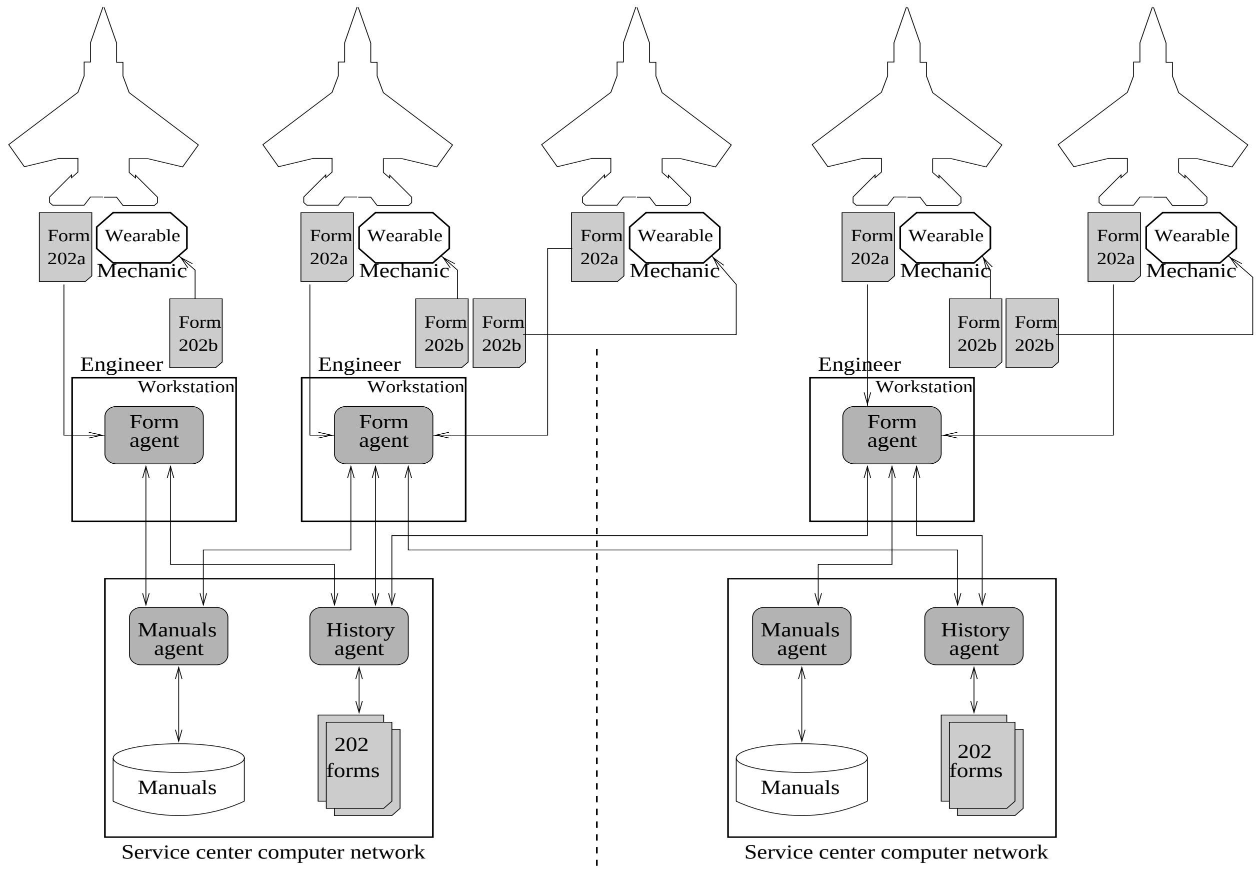


Figure 8: The multi-agent system organization.

system facilitates the automatic addition of new agents; as new resources[5] become available, agents can be created to exploit them, without modifying the existing system. This openness also allows for dynamic appearance and disappearance of agents and information sources without hampering the overall system performance, since RETSINA agents know how to find alternative agents when their initial choice is not available. We believe that multi-agent systems, and in particular the RETSINA infrastructure, provide a good solution to this type of decision support problem, as demonstrated by our prototype of agent-aided aircraft maintenance system.

6 Acknowledgement

This work could not have been performed without the cooperation and support of Dan Siewiorek, Dick Martin and Jane Siegel from the Wearable Computing project at Carnegie Mellon university and the Warner Robins Air Force Base, Georgia. Funding for this work has been provided by the ONR grant #N00014-96-1-1222.

References

[1] K. Decker, K. Sycara, and M. Williamson. Middle-agents for the internet. In *Proceeding of IJCAI-97*, pages 578–583, Nagoya, Japan, 1997.

[2] Michael N. Huhns and Munindar P. Singh, editors. *Readings in Agents*. Morgan Kaufmann, San Francisco, 1998.

[3] N. Jennings and M. Wooldridge (editors). *Agent technology*. Springer, 1998.

[4] N. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. *Autonomous agents and multi-agent systems*, 1(1):7–38, 1998.

[5] H. Nwana and N. Azarmi (editors). *Lecture notes in artificial intelligence vol. 1198: Software agents and soft computing*. Springer, 1997.

[6] K. Sycara, K. Decker, A. Pannu, and M. Williamson. Designing behaviors for information agents. In *Proceeding of Agents-97*, pages 404–412, Los Angeles, 1997.

[7] K. Sycara, K. Decker, A. Pannu, M. Williamson, and D. Zeng. Distributed intelligent agents. *IEEE Expert – Inteligent Systems and Their Applications*, 11(6):36–45, 1996.

[8] K. Sycara and D. Zeng. Coordination of multiple intelligent software agents. *International Journal of Intelligent and Cooperative Information Systems*, 5(2 & 3):181–211, 1996.

[5] Examples of such resources include: information sources; methods for processing information; user preference learning.