Automated Decision Stations

Mark Tischendorf
Eastman Kodak Company
Artificial Intelligence Laboratory
(716) 477-1357

March 15, 1990

Automated Decision Stations

Abstract

This paper discusses the combination of software robots and expert systems to automate everyday business tasks. Tasks which require people to repetitively interact with multiple systems screens as well as multiple systems.

1. Objective

This paper describes a system created to automate decisions. Either an independent system capable of performing specific business tasks or an intelligent assistant which helps individuals by collecting information, offering recommendations, and carrying out decisions.

The term "Information System" implies an end goal of providing a person with information. The person is responsible for deciding what the information means. Our intent is to integrate the system's information with automated human decision making without altering the existing information systems.

For many potential applications, the business case compares system implementation costs with the cost of clerical labor. Therefore, implementation costs need to be as small as possible.

2. Introduction

Information systems are an integral part of many business operational environments. These systems normally provide users with information about a single domain. This results in individuals being assigned to act as interfaces between such systems. For specific tasks, a person needs to gather information by referencing multiple systems or screens, decide what to do with the information, then carry out the decision within one or more systems. These types of jobs exist because building interfaces between the existing information systems is not possible or cost effective.

Software robots combined with expert systems can emulate the type of human activity described above.

A software robot is a programming tool for automating the use of existing software. Software robot tools are also called surround tools, agents, or script files. Software robots can emulate keystrokes and monitor screen activity. In most cases software robots can automate any repetitive task that a person performs at a terminal.

Occasionally a software robot application requires significant reasoning capabilities in order to make decisions. This is where expert systems come into play. The software robot collects pertinent information and feeds it to the expert system. The expert system performs its reasoning and either tells the software robot what to do or makes a suggestion to an individual. In either case, the software robot can then carry out the decision.

The software robot acts as "the eyes and hands." It knows how to traverse systems and screens, and where to locate data on screens. The expert system, on the other hand, acts as "the brains." It reasons about information provided by the software robot. Although I've called this type of system an Automated Decision Station, one could also call this an Expert Software Robot.

3. Implementation

The hardware involved is a personal computer equipped with multiple session 3270 emulation capability. The 3270 emulation software, robot software, and expert system software need to be simultaneously resident in the PC's 640K memory. Therefore, memory restrictions are a primary constraint on software tool selection. The PC need not be a high-speed 386-based machine. Host system response time is the limiting factor on software robot execution speed. A 286-based PC works well and is sometimes easily obtainable since they're somewhat out of date.

The software robot tool we've used is AUTOMATOR-MI from Direct Technology. AUTOMATOR is capable of surrounding software on the PC as well as any type of host system. AUTOMATOR also works over a wide variety of connectivity alternatives. The current version of AUTOMATOR uses about 96K of memory.

The expert system shell CLIPS, from COSMIC/NASA, handles the decision reasoning. CLIPS is desirable due to its low memory overhead, low cost, and ability to import data files.

The robot controls execution of the other software and interfaces with the user if necessary. The robot accesses one or more systems by way of the 3270 emulation. Concurrent access to multiple systems is accomplished over separate emulation sessions. The robot collects pertinent data from these systems and creates a file on the PC containing this information. The robot then jumps from 3270 emulation into DOS where the expert system is already running but suspended. The robot starts up the expert system, which reads the pertinent data file and reasons about a decision. The expert system displays the decision on the screen so the robot can see it, then suspends itself. The robot reads the decision and jumps back into 3270 emulation to carry out the decision. Note that the robot is controlling all the activity. The robot treats the expert system as a decision-making calculator.

4. Example

We've built an Automated Decision Station to assist order entry credit checking in one of Kodak's distribution regions. A two-to three-week programming effort has yielded a system which can automatically handle about 20% of the credit referral activity. More significantly, the automated data collection considerably aids manual processing of the remaining credit referrals.

As described above, the robot collects data for each credit referral from two different mainframe systems. The pertinent information is summarized from six or more different screens. The expert system identifies the type of credit referral and performs any appropriate calculations. A printout communicates the pertinent information, recommended action, and reasoning explanation. If the referral looks okay, the system will approve it, given user confirmation. Otherwise, the system places the referral on hold for manual handling.

This application has been in use since the beginning of February 1990. As of this writing, several thousand transactions have yielded no significant problems. Future enhancements are identified to provide additional automation capability.

This Automated Decision Station offers management the opportunity to combine manual credit referral operations. We can concentrate activity from several regions into two regions (east-coast and west-coast). Alternatively, a Decision Station can be put in each region to streamline each existing process.

5. Costs / Benefits

The primary cost of this Credit Referral Expert Software Robot was the two- to three-week programming effort. A spare PC was resurrected from a storage shelf. A software robot run-time license cost $250, and a 3270 emulator board cost $750.

The benefits include:

> Reduced labor, increased productivity, and faster workload turnaround from automating repetitive terminal activity. These systems sometimes cut out the need for users to interface with any systems.

Better job quality due to the absence of typing errors.

Users of such systems gain an increased sense of self-worth. Rather than spending time keying and calculating, they are now free to concentrate on the highly skilled parts of their job.

These inexpensive decision stations offer a new way of accomplishing things. They are an alternative means of interfacing information between systems for specific business purposes. These systems are a cost-effective way to do things previously considered unjustifiable.

By capturing the rules on how to make specific business decisions, we are preserving corporate know-how. We then apply this know-how consistently to suitable problems. This know-how can also help with the training of new people.

6. Observations

This type of system is easy to introduce into new environments. The low cost certainly helps, but the implementation methodology also plays a big role. The software robot needs to surround the existing work environment. So the system is typically build right in the end-user work place. This results in close contact with both the users and their management. They see the system evolve as it is built, fostering a sense of ownership.

These Expert Software Robots lend themselves to modular implementation. It is often possible to build only the robot component, keeping the person in the loop for decision making. Then, build the expert system component when resources become available. It is helpful to use the robot to collect actual test cases, to aid the expert interviewing process.

It turns out that it is very easy to migrate these systems from prototype status into a production-worthy system. It is so easy, in fact, that we make this migration even though programming enhancements are pending. The catch here is that once in production, programming changes have to be more carefully coordinated and are therefore more time consuming.

The one disadvantage to software robots are their vulnerability to host system screen changes. If screens change in the surrounded environment, the robot can get confused. So far this has not been a problem. It has only taken minutes to fix a couple such occurrences. This does, however, imply that a trained person needs to be available to attend to these types of unexpected situations.

7. Conclusions

Automatic Decision Stations (or Expert Software Robots) are easy and inexpensive to build. The learning curve on the software tools is relatively short. These types of applications can increase productivity while improving quality. These systems offer a new way to solve problems, as well as an alternative way to view existing systems environments.