ABSTRACT

Large scale software simulations generally require lengthy development time and can be extremely costly. A plethora of simulations of various kinds have been developed over the years. The type of simulation which meets most of the user's requirements already exists, and can often be found at low or no cost. The benefits of using existing software include low cost and practically zero development time. The drawbacks may be the lack of documentation, support, and inadequately met requirements. However, these problems can exist in any software acquisition. Well defined requirements should be the initial part of any software design or purchase, and are often the key issue in its successful application. The use or modification of readily available simulations should be examined as viable alternatives to developing or purchasing software. This paper examines the pros and cons of the various alternatives to software development.

INTRODUCTION

This paper describes the evolution of an Air-to-Air Combat Simulation model that was developed in the Air Force Avionics Laboratory, Mission Avionics Division, Applications Branch for the purpose of providing an in-house capability to analyze various Air-to-Air engagement scenarios. Many simulations have been developed during the analysis and evaluation of avionics systems over the years, and a recent survey of electro-optical computer models showed that about 130 existed in 1981 and 200 in 1984 (Ref 1). There has been, and sometimes still is, a tendency to re-develop software. This may not be too bad for relatively simple models which don't require a great deal of development effort. However, as the complexity of the simulation increases software re-development becomes a lengthy and costly process.

Fortunately, as the complexity of these models increases, so does the generic nature of the code, making it re-usable for many applications. This has become possible because of better software practices and newer languages. For example, Ada allows the models (such as sensor and aircraft models) included in various simulations to be treated at a higher level of abstraction (Ref 4). Many simulations were, and are still, written in older languages, like FORTRAN, and don't have the nice structure and modularity of some of today's programs. This frequently makes the decision of whether to use an existing simulation or develop a new one difficult.

Very few typists or accountants would consider developing a word processor program or a spreadsheet. Their primary role is as tool user, not tool developer. An engineer, on the other hand, is often not only the tool user, but also the developer. Engineers may not be familiar with all the software available. It is up to them to find out what is available. They may be reluctant to use software developed elsewhere because they have had a bad experience with using someone else's software. Often, they suffer from the second system effect (Ref 3), trying to develop ideal software. Another individual, the collector, may be all too anxious to acquire promising simulations, only to find out well into the project that they don't meet his needs.

The remainder of this paper will illustrate a procedure for making these decisions. Development of an in-house Air-to-Air Analysis capability will be used as an example. Deciding whether to build or acquire, and factors for choosing a simulation will be presented to show some of the salient points and subtleties involved. This procedure can be divided into roughly five steps: (1) Top Level Simulation Specifications, (2) Define Options/Constraints, (3) Develop The Selection Factors and Process, (4) Survey Available Software, and (5) Perform the Selection Process.

TOP LEVEL SPECIFICATION

Whether you're going to buy a word processor or develop a sophisticated simulation, some some definition of what is suppose to do is an absolute necessity. This may seem like stating the obvious. A belief exist that a simulation, is a panacea that will answer all problems. Since a software development can't be done in a vacuum, the program project objective will often provide a start at a top-level specification.

In the case of the Air-to-Air Combat
Simulation effort, the objective was "...to analyze system and mission performance in air-to-air engagements. Evaluation is to be performed for a variety of air-to-air missions as functions of variations in aircraft performance, sensor and weapon capability, fire control and other parameters." Admittedly, this was vague, but it did give us an idea of what was required. Just about any parameter can be varied, models or logic can be changed, and some bottom line figure of merit or charts (see Fig. 1) relating these parameters or changes to mission effectiveness can be produced. It is also known from past efforts that bottom line figures are not enough. It was also desirable to know what intermediate parameters affected the results and why.

This started to tell us that the simulation would have to be rich in the required models (sensors, weapons, platforms, fire control algorithms, pilot logic, etc.). The models would also need to be generic, that is, have a sufficient level of abstraction to be varied parametrically rather than by code modification.

We went as far as listing the types of models, and considered possible simulation structures and implementations, but listing these would be beyond the scope of this paper. This rather high level description told us to expect a monumental development task. We began to look at our options and identify constraints.

**OPTIONS AND CONSTRAINTS**

The options were fairly simple: (1) develop the simulation ourselves, (2) buy an "off the shelf" simulation, (3) contract simulation development to our specification or (4) find a government owned simulation which would be satisfactory. The constraints, as usual, were time, budget, and manpower. The desired timescale was one year; the budget was effectively zero; and the available manpower was two and a half man-years. We also had to realize our experience limitations. Although we had expertise in some of the areas, we certainly could not claim expertise in all of the areas.

Figure 2 shows the trades for this project. Finding an available government owned simulations were selected. Past experience and discussion with several who have built similar simulations indicated that at least five and possibly ten man-years would be required to do the job correctly. This eliminated option one, build our own. It also eliminated option four for both budget and time reasons. Option three might still be open if the budget constraint could be relaxed and, indeed, in this project we also considered purchasing an "off the shelf" simulation in the event we were unable to find suitable government owned software. Choosing the third option, however, would not really have changed the rest of the process. Actually the constraints on most projects won't necessarily be so rigid.

**SELECTION FACTORS**

<table>
<thead>
<tr>
<th>FACTORS</th>
<th>AVAILABILITY</th>
<th>UTILITIES</th>
<th>DOCUMENTATION</th>
<th>USER SUPPORT</th>
<th>MODELS</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVAILABLE SIMULATIONS</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>AASPEM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AIMTACS</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td></td>
</tr>
<tr>
<td>PACAM 8</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td></td>
</tr>
<tr>
<td>TAC BRAWLER</td>
<td>✗</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>✗</td>
</tr>
</tbody>
</table>

**FIGURE 2. OPTIONS AND CONSTRAINTS**

**FIGURE 3. SIMULATION SELECTION**
Figure 3 shows the selection factors which were used and the simulations considered. Factors were reached by discussion among the team members. At first, this seemed like a list of buzzwords. It was necessary to define what each factor really meant and ways to evaluate them. Ideally these should be done before any simulation is considered.

Availability meant that the simulation can be acquired almost immediately at no cost. This factor was evaluated by actually acquiring the code and supporting documentation.

Usability refers to flexibility, maintainability, modularity and usability. These factors were evaluated by examining the code and documentation. Example problems were created and estimates of how easy or difficult it would be to set them up in a particular simulation were performed. Could it be done by data changes only or would the code have to be modified? If the code had to be modified, did the simulation structure lend itself to easy modification? Finally, each simulation was installed, compiled and a test case executed on our VAX 11/780 computer.

Documentation refers to the clarity of the documentation. Is it sufficient to run the code and set up test cases? Could the code be modified with confidence using the existing documentation?

User support is the degree to which the code is supported. Do people exist who could help us use the code and solve problems with the code that might occur?

Models means the variety and level of models included in the code. Are all the necessary models there? Were the models of the appropriate level of detail and fidelity? Did data exist for specific instantiations of the model?

Note that for each factor there are specific questions to be answered or actions to be performed. Only one factor, the models, is specific to the Air-to-Air Combat effort. The rest involve software related issues, and are more widely applicable. Although some of the questions may be subjective, this approach gets away from attempting to rate the factors on an arbitrary numerical scale of individual opinion.

SURVEY OF AVAILABLE SOFTWARE

The next step in the selection process was to find available government owned software. We had several available sources, including the SURVIVAC repository (Ref 2), available Defense Technical Information Center (DTIC) surveys, previous software developments, and personal contacts. Several contractors were also contacted, since purchasing a simulation was a fail-back position if we were unable to find one which met our requirements. This survey was only a screening process, an attempt to find software that looked promising for more detailed evaluation.

Eventually, four engagement level simulations were found which seemed to fit our needs. These were the Air-to-Air System Performance Evaluation Model (AASPEM), Advanced Identification MultipleTarget Air Combat Simulation (AIMTACS), The Piloted Air Combat Analysis Model (PACAM), and TAC BRAWLER. The selection process was applied to these.

SELECTION PROCESS

The result of the selection process is shown in Figure 3. All of the simulations except TAC BRAWLER were easily acquired. There was also some indication that the models in TAC BRAWLER were at too high a level of detail for the type of conceptual design studies we wanted to perform.

AIMTACS was developed specifically for an Air-to-Air identification problem. Consequently, some of the models, like the missile models, were at too simple a level to be useful. The code did not appear to be modular enough to easily modify.

The difficult choice was between PACAM and AASPEM. PACAM had good documentation and met most of our requirements. AASPEM had poorer documentation, but was better in the variety of engagements it could handle. AASPEM also had an active users group. While the users group may not qualify as user support in the strictest sense, at least there were people we could contact if we had problems with the code. We decided that AASPEM was probably better, and was selected as our simulation baseline.

CONCLUSION

The entire specification and selection process took about five man-months. We were able to use the simulation to support several internal study efforts, accomplishing the initial objective. If we had attempted to develop our own simulation, the project would have required several years considering available resources. If funds were available and purchasing a simulation were selected, the initial specification and evaluation would have had to be accomplished anyway, and the procurement cycle would have added additional time.

In this case careful selection of readily available software resulted in considerable savings of both time and manpower. None of the steps involved would have been wasted, even if we were unable to find a suitable simulation and had been forced to fall back on one of the other options. The top level specification would have been a starting point for an internal
development. The process of developing selection factors would have been applicable to a contracted effort.

The five-step technique presented here, provides alternatives to full scale simulation development, and allows the program manager to keep his options open.

REFERENCES


