“Customer In the Loop” Development of Training Simulators

Peter Freed, Managing Director
Cirrus Real Time Processing Systems Pty. Ltd.
Peter.Freed@cirrusrts.com.au

Abstract. Cirrus Real Time Processing Systems ("Cirrus") has been engaged by the Defence Materiel Organisation (DMO) to develop the Mine Hunting Simulation System ("MHSS") which will be used to streamline the training of RAN personnel in the part-task of mine-hunting. For the MHSS to create real benefit to the end users, it must address issues quite specific to the functional domain of these end users, as opposed to (and in addition to) the typical engineering issues inherent to the development of any complex software system. To address this, the development process has revolved around the operation of an Integrated Project Team comprised of the developer (Cirrus), the acquirer (DMO Project Office SEA 1297) and the end users (HMAS WATERHEN Mine Warfare Faculty).

By effectively embedding the customer into the engineering design loop, the engineering outcomes of this project have been appropriately guided to align with the end users’ functional requirements.

This paper describes the successful application of the IPT project management structure. Practical examples of the involvement of the customer in the engineering design process are provided, illustrating the clear benefit of this approach.

The successful outcomes of the "customer in the loop" methodology are also illustrated.

1. INTRODUCTION

Software development projects by their nature may be rightfully regarded as being “technically risky.” In consequence, a variety of differing engineering and contractual approaches have been proposed to attempt to address and manage the inherent risk.

Evolutionary acquisition (EA) is one such approach, wherein the acquirer and the developer agree (contractually) to develop a complex software system in stages.

The intent of EA is to allow the development of detailed specification of the “end-product” to be deferred until late in the development cycle, so that end users have an opportunity to experience early “builds” of the system and to provide informed feedback.

In essence, EA purports to embed the end user into the engineering development process.

However, as with any management approach that might claim to “solve” the problems evident in earlier approaches, it is prudent to investigate whether EA in practice follows the theory.

In this article, the application of EA in practice is considered, drawing on Cirrus’ experience in developing the Mine Hunting Simulation System (MHSS) via EA.

1.1 Introduction to the MHSS Programme

In March 2003, Cirrus Real Time Processing Systems ("Cirrus") was contracted by the Mine Warfare Command Support System (MWCSS) project office of the (DMO Project Office SEA 1297) to develop a Mine Hunting Simulation System (MHSS). This contract enshrines the EA model of software development, with an Integrated Project Team (IPT) being formed comprising MWCSS (the acquirer), the Mine Warfare Faculty (MWF) of the Royal Australian Navy (RAN) (the end user), and Cirrus (the developer).

Via this IPT, the end user is effectively embedded into the development process.

1.2 Technical Objectives of the MHSS Programme

The MHSS is being acquired to provide part task training for RAN personnel in mine hunting skills. To this end, the MHSS must provide a simulation of the sonar and tactical display equipment incorporated into the RAN’s Mine Hunter Coastal (MHC) vessels, and their use in realistic mine hunting scenarios.

The intent is that MHSS training will allow mine hunting trainees to:

- become familiarised with the RAN’s mine hunting equipment,
- learn how to operate the mine hunting equipment according to Standard Operating Procedures (SOP’s), and,
- learn how differing choices of how to operate the equipment will affect the overall performance of mine hunting operations in differing environmental conditions.

This creates two technical challenges relating to simulation fidelity:

1. Equipment operability. The manner in which an end user interacts with the MHSS must reasonably
match that which prevails in the at-sea mine hunting equipment.

2. Acoustic modeling chain. To allow the end users to “see” via their sonar displays the effects of differing processing choices and environmental conditions, the need arises for a sophisticated acoustic model. Such a model would necessarily cater for a whole host of effects that impact the sonar picture displayed to the operator – signal levels, aperture effects, geometries of the sea-bed relative to the aperture, reverberation effects and signal processing artifacts etc.

In addition to these challenges, the use of the MHSS for training purposes creates further technical challenges:

1. The need to create a large (10) number of seats, to support the bulk training of students per class. This implies a networked training simulator, which in turn creates the need to suitably arrange an appropriate distribution of processing functions across the network.

2. The need for an instructor to control a training session, to monitor the performance of the trainees, and to collect and manage assessment data for each student.

All of these factors must be addressed simultaneously by the software design. Quite clearly, the MHSS development can certainly be regarded as technically risky.

How has EA faired in this example?

In the following sections, a sample of cases that typified end-user interaction during the MHSS development under EA is examined.

2. FINESSING REQUIREMENTS WITH END USER INTERACTION

The following three case studies of end user interaction have been selected to provide examples relating to issues with progressively greater levels of technical complexity.

2.1 Case Study I : Colour Matching

The RAN’s mine hunting equipment includes many computer generated displays and key-pads.

Error! Reference source not found. is a sample of an MHSS display page, within which can be identified data regions containing sonar data, plus a variety of “trimmings”; labels, totes and data axes.

Clearly, the training value provided by the MHSS will be promoted by selection of colours for the display features that reasonably match those extant in the RAN’s ship-fit equipment.

This process is reasonably straightforward for the display “trimmings” and keypad labels. The task of setting colouration of data in the sonar data display areas is somewhat more challenging to match.

Under the EA model, the approach taken to specify and implement these colourations was straightforward. After Cirrus’ engineers developed the framework software, the end users sat with Cirrus’ engineers and watched the data displays update while the engineers adjusted various settings in the software. This process continued under the end users’ direction until they had attained settings that they were pleased with.

2.2 Case Study II : Towed Body Control

The MHC’s mine hunting sonar provides a “variable depth” mode, which allows the sonar aperture to be lowered on a cable below the hull.

The depth at which the “towed body” (i.e. the sonar aperture) is deployed has a considerable impact on the performance attained by the sonar for both search and classification tasks.

From an operational point of view, to avoid damage, the towed body must also be kept clear of the sea-bed. Consequently, as a training simulator, it is important that the MHSS provides simulation of the control for this towed body.

On the MHC, the deployment of the towed body is achieved via electronic controls mounted at a separate cabinet. These controls include data keys that allow an operator to enter the desired towed body depth, “action” buttons to start/stop deployment, proximity alarms that may be set to sound within certain distances of the sea-bed, and automatic retrieval functions that can be set to avoid sea-bed collision.

These controls, while reasonably straightforward in concept, have certain interdependencies that are not immediately evident.

As a software developer without any prior experience with this device, Cirrus did not know in advance the nature of these interdependencies.

Rather than place the onus on the end users to specify in great detail the behaviour of this device, the EA structure allowed Cirrus to develop an interim version of a simulator for the towed body control. Along the way, the Cirrus engineers made various “educated assumptions” regarding the interdependencies of the controls.

The end users then sat down and “played” with this interim version. When presented with a working model, the end users had no problem in articulating the differences between the software’s behaviour and the simulation behaviour that they sought.

The original system behaviour assumed by Cirrus’ engineers, coupled with the end users’ verbalised descriptions of differences, in effect formed a detailed specification of the behaviour of this device.
A software module implementing the algorithm was developed and tested. The result of this reliance on first principles was that a truncated conic object appeared in the sonar data window with a shape that can be best described as a “smudged parallelogram”.

Mid-way through this process, at the time where this software module was undergoing initial unit testing, the appearance of these contacts did not fully tally with the end users’ collective experience of how they “should look”. They did not expect to see a smudged parallelogram, but rather a stack of smudged rectangles.

While this description was useful, for the engineers to appropriately simulate the effect, they needed to better understand why this shape came about. After all, the final appearance of the contact in the data window incorporates many effects (e.g. relative position to the array) which would vary dynamically. To merely “copy” a canned representation of the stacked rectangles to the display would be inadequate in training the end users how these objects might appear for example, from different aspects or ranges.

This prompted an ad-hoc discussion regarding the appearance of these contacts in which it was revealed that a further physical effect (the scouring of the seabed near the contact) might often be expected to affect the sonar image of these types of objects.

Further “directed questioning” by Cirrus engineers sought the end users to draw on their operational experience with the ship-fit equipment to describe how the shapes of the images would vary when viewed from different orientations.

In this fashion, end user feedback allowed Cirrus engineers to develop what was in effect, a highly detailed specification of the desired imaging behaviour of the MHSS for truncated conic contacts. In parallel with this, Cirrus gained the necessary knowledge of the physical process to re-engineer its model of truncated conic objects.

**Error! Reference source not found.** illustrates the MHSS sonar data display containing a truncated conic contact, which is visible with the characteristic stack of rectangles.

Once again, presentation to end users of an interim result that was “in the ballpark, but not quite right,” prompted the provision of highly detailed information from the end users of what the correct behaviour should be.

In this case, the detailed requirement extracted from the end users in this fashion covers a highly complex area of functionality, incorporating mathematical algorithms and their implementation in software.

### 3. DISCUSSION OF “CUSTOMER IN THE LOOP” DEVELOPMENT

The three examples provided above are typical of the very large number of interactions that took place within the IPT relating to the detail of the behaviour sought from the MHSS by the end users.
Viewed as a whole, a number of common threads emerge.

It is apparent that at the outset of a software development project, it is not straightforward to elucidate from end users their requirements in detail for a system.

However, the experience in the MHSS IPT is that once a “working prototype” becomes available, the end user will provide a considerable volume of information relating to their actual requirement.

It is apparent that at the outset of a software development project, it is not straightforward to elucidate from end users their requirements in detail for a system.

This information will usually be received in verbal form within a free-flowing discussion, and will usually take the format of a description of what changes are needed from the existing working prototype.

The volume of information provided is entirely adequate to identify requirements at a level of detail that will result in a final product that satisfies the end users’ actual (if not always stated) needs.

However, this approach does place some additional responsibilities on the developer.

It is incumbent on the developer to ensure that the working prototype is “in the ball-park”, so that the end users have a reasonably representative basis from which to identify the necessary changes.

Where the nature of the functionality being implemented is highly complex, it is incumbent on the developer to direct the flow of the discussion so as to extract from the end user the key items of information that underlie and drive the desired output behaviours.

Finally, it is incumbent on the developer to translate the information relayed during ad-hoc discussion into formal requirements.

It is also worth comparing how a non-EA process may have managed the development of requirements for the MHSS.

As an example, the “fixed price, build to fixed specification” is a common model for software development projects.

With this model, the end user will be involved during the pre-contract requirements definition phase, and then at test and acceptance; i.e. the end user is not embedded into the development phase.

With this approach, the onus is on the end user to specify in detail all of the behaviours of the system under development up-front.

In the case of the MHSS, it is not obvious that this process could have been conducted effectively at the outset. Our experience has been that the end users have been most forthcoming with information describing how the MHSS should operate (i.e. requirements) as and when a working prototype has been available to comment on.

It is also worth noting that it would have been very difficult foreseeing in advance which particular areas of functionality needed highly detailed requirements, and which did not.

Consequently, it is likely that some functional areas may be over-specified, whereas other areas may be under-specified.

This implies that the acquisition organisation may incur either wastage of resources (for those areas that become over-specified), or expose the end user to acquisition risk (for those areas that are under-specified), or possibly both.

4. END USER INTERACTION – OVERALL RESULT

As noted above, the IPT was formed in March 2003 following contract award in that month.

Cirrus delivered to the MWF the first operational MHSS (“R2”) in December 2003, which is now in reliable operational use. This represents a development schedule that is somewhat more aggressive than industry norms for a complex software development project.

It is reasonable to suggest that EA is a contributor to the success of this programme.

By the same token, the author acknowledges that other factors might also have contributed to the success of this programme. Furthermore it is recognised that successful application of EA to one project does not necessarily imply that application to other complex software development projects will by itself result in successful project execution.

Nevertheless, given that (in this case) the in-theory advantages of EA were indeed borne out in practice, the EA approach to software development should be seriously considered by other parties embarking on complex software development programmes.
Figure 1: Typical mine hunting sonar data display. In this instance a Plan Position Indicator (PPI) display illustrates sonar returns from a swathe of the sea-bed.

Figure 2: The towed body control. This feature provides a simulation of the appearance and behaviour of the MHC towed body control cabinet.
Figure 3: A sonar data display, containing a “truncated conic” object, which is also visible in expanded form in the smaller data area.