Development of game-based training systems: Lessons learned in an inter-disciplinary field in the making

Talib Hussain, Wallace Feurzeig, Jan Cannon-Bowers, Susan Coleman, Alan Koenig, John Lee, Ellen Menaker, Kerry Moffitt, Curtiss Murphy, Kelly Pounds, Bruce Roberts, Jason Seip, Vance Souders and Richard Wainess

Abstract
Modern computer gaming technology offers a rich potential as a platform for the creation of compelling immersive training systems, and there have been a number of game-based training systems developed in recent years. However, the field is still in its infancy. Improved understanding is needed on how to best embed instruction in a game and how to best use gaming features to support different types of instruction. Further, the field is inherently inter-disciplinary, requiring instructional system designers, software developers, game designers and more, yet there are no established development methodologies to ensure effective coordination and integration across these disciplines. We introduce a collaborative effort that is investigating how to improve the craft and science of game-based training. We present our experiences in creating a flooding control training system for the U.S. Navy Recruit Training Command, and discuss the inter-disciplinary development issues that we encountered. We present the lessons we learned and our views on how to advance current methods to support the consistent production of effective game-based training.

Keywords
Game-based training, serious games, simulation-based training, instructional design, military training, automated assessment

Introduction
Computer games of various kinds have been used for education and training purposes for over two decades with varying degrees of success (O'Neil et al, 2005; O'Neil and Perez, 2008). As computer gaming technology has matured and increased in capability, the opportunities available for delivering immersive learning experiences have increased (Bonk and Dennen, 2005; Hill et al., 2006; Hussain et al, 2008; Johnson et al., 2007; Roberts et al., 2006), and so has the challenge of creating experiences that are pedagogically effective (Diller et al., 2004; Hussain and Ferguson, 2005). A training game is imbued with a purpose - to provide experiences which lead to specifics gains in the student's knowledge and/or skills. A good training game will consistently balance the instructional goals with the goal of motivating the player. However, a poorly designed training game will sacrifice one or more fundamental elements of the gaming experience in order to attempt to satisfy the training goals, or will sacrifice effective pedagogy in order to attempt to keep the game compelling. The former may be a great training system, and even a great simulation-based training system, but doesn't pass muster as a game-based training system since the players don't enjoy it. The latter may be a great game, but doesn't pass muster as a training system since it does not produce the desired
learning outcomes. Developers of game-based training systems know this, but achieving this synergy between instruction and engagement is a poorly understood art.

The challenges facing us as a discipline are:

1) An enhanced understanding of the elements of game design and pedagogical design that are crucial to game-based training and how to balance those elements effectively,
2) An enhanced understanding of how to assess the success of a game-based training application, and
3) The creation of development methodologies that lead to repeatable successes, especially for non-commercial training programs that are limited in the scale of effort that can be supported.

We introduce the initial results of a multi-disciplinary effort sponsored by the Office of Naval Research to directly address the issue of how to best create effective educational immersive computer games. The team for our project included researchers and content developers from the fields of instructional design, story-based training and entertainment, movie production, human performance assessment, game engines, commercial games, game-based training systems, simulation and modeling, intelligent tutoring systems, graphic design and modeling, system integration and educational science. As professionals in our respective fields, we each brought different perspectives on the interactions of different aspects of gaming and pedagogy to the table.

Our effort had two mandates. The first was to conduct applied and empirical research on tools and methods for enhancing the art and science of educational games. In particular, our initial focus was on identifying extensible design and development methods that support efficient creation of training games with strong pedagogy and embedded assessment. The second mandate was to create prototype training systems for the Navy that met real training needs and embodied sound pedagogy. The customer for our first training prototype was the U.S. Naval Service Training Command. The goal of this prototype effort was to provide training on how to perform effective flooding control aboard a simulated naval ship. The Flooding Control Trainer that we developed is intended for use at the Navy's boot camp to augment the classroom-based and hands-on instruction that is currently provided to over 40,000 recruits per year.

In a seven month period, our team members worked closely together, with subject matter experts and with our Navy customer to produce a prototype that met our customer's needs and was effective. We present here a description of the process that we followed, the tensions we encountered during the effort, and the lessons we learned on how to work in an interdisciplinary manner to achieve an instructionally strong and enjoyable outcome. We discuss our next steps in the project to refine and formalize our process, as well as our thoughts on where the field needs to focus going forward to ensure longevity and success as a discipline.

**Background**

From a theoretical perspective, games hold promise as effective teaching devices because they can provide instructional environments that embody key principles derived from learning science. For instance,

- Interactions that facilitate player engagement in a compelling task environment should facilitate learning. Practicing in this type of environment is consistent with notions about the development of expertise (see Glaser, 1989; Chi et al., 1988; Bransford et al., 1999) and
anchored instruction (e.g., Bransford et al., 1990; CGTV, 1992; 1997; 2000). The game world provides a context in which appropriate mental models are developed and refined through repeated exposure to important cue patterns.

• Games provide an excellent model-based world to foster reasoning. Students are able to manipulate variables, view phenomena from multiple perspectives, observe system behavior over time, draw and test hypotheses and compare their mental models with representations in the external world. These features are consistent with the model-based reasoning concepts advocated by learning researchers (Gentner, 1983; Raghavan et al., 1997; 1998; Leher & Schauble, 2000; Cartier & Stewart, 2000; Zimmerman et al., 2003; Stewart et al., 2005).

• Game-based tasks are expressly designed to help players progress toward goals. Moreover, the goals are concrete, specific and timely. The vast literature on goal setting in instruction suggests that this characteristic property of games should enhance learning (see Locke et al., 1981; Locke & Latham, 1990; Schunk & Ertmer, 1999).

• Interaction frequency is very high in games. These often require decisions and inputs by players several times a minute. Thus, games provide a highly active learning environment, the kind of environment associated with effective instructional system design (see Rothman, 1999; Chi, 2000; Mayer, 2001; Vogel et al., in press).

• Well-designed games provide the player with constant successes. Many small tasks are embodied along the way in the pursuit of a greater goal. The result is that the game generates a feeling of constant accomplishment, a feature likely to enhance self-efficacy, which has been consistently shown to improve learning and motivation (see Bandura, 1977; 1986; Gist et al., 1989; 1991).

• Games provide a continuous source of feedback so that players know where they stand with respect to their goal accomplishment. This is crucial since feedback improves learning through both its informational and motivational qualities (see Bransford et al., 1999; Salas & Cannon-Bowers, 2000).

• Game play tends to be self-regulating, an important feature of effective learning. Players are motivated to accomplish the next challenge and they know where they stand relative to their goals. (see Kanfer & Ackerman, 1989; Schunk & Ertmer, 1999; Pintrich & Zusho, 2000; Kanfer & McCombs, 2000; Schunk & Zimmerman, 2003).

• Engagement and immersion are high in well-designed games. Literature is beginning to investigate these concepts as psychological states associated with effective performance and learning (see Csikszentmihalyi, 1990; Gerhard et al., 2004) and to examine what contributes to them (see Baylor, 2001; Gerhard et al., 2004; Moreno & Mayer, 2004).

• Challenge and competition are hallmarks of good games. These features have been found to be motivational under certain conditions (see Epstein & Harackiewicz, 1992; Reeve & Deci, 1996), and may be useful to motivate learning.

• Perhaps due to factors listed above, time on task is very high for games. It is not uncommon for players to spend hours a day on engaging games, and to continue interacting with a single game for years. From a strictly time-on-task perspective, we would expect that learning would be enhanced when trainees engage quality learning content for longer periods of time.

Despite this promise, however, many game-based training applications suffer key deficiencies leading to poor learning outcomes, such a poor instructional design, poor (or no) performance assessment, limited training scope, low levels of reusability, and lack of appeal to students (Hussain et al., 2008; O’Neil and Perez, 2008). These problems are due in part to the
limitations of current gaming technology - for example, commercial games typically do not provide the ability to capture the relevant performance data at the right level of detail for tracking and assessment of trainee performance. However, they are also due in large part to the fact that there are no clearly proven methods for the development of effective training games. Further, most organizations and companies developing game-based training systems have only a few years of experience doing so and their experience is usually limited to work on one or two systems. Generally, designers and developers apply methods that have been appropriated from other fields, such as general software development, simulation-based training, computer-based training, commercial game development, and intelligent tutoring systems. The lessons learned from these ad-hoc approaches to game-based training development are rarely shared. Hence, insights are lost, pitfalls are re-encountered and useful community-wide guidelines are not formed. Further, different development teams contain expertise in different fields, and hence some elements of game-based training design tend to be emphasized at the cost of others.

Our project team was a highly multi-disciplinary set of experts, each with varying degrees of expertise with game-based training, but all with deep knowledge within their core disciplines. The development team included members from seven different organizations that together provided expertise in game-based training, educational science, commercial game development, instructional design, human performance assessment, story-based training, simulation-based training, computer-based training, military training systems, graphic design and modeling, animated movie development, entertainment and media production, intelligent tutoring systems, game engine development, agile software development methods, and systems integration.

Collectively, these experts brought to bear a rich subset of the practices currently used for developing game-based training. The key practices that we chose to start with based on preliminary discussions included:

- The manner in which the system will be used impacts instructional design choices, so pay close attention to requirements gathering.

- Remain focused upon learning objectives throughout system design and development.

- Develop the story outline early on, base it on the learning objectives and iterate as system design proceeds. Taken to an extreme, one of the teammates believed that "It all starts with story". According to Jerome Bruner (1990), plots are a creation of "transactional relationships" between reality, memory, and imaginary/narrative worlds. Transactional connections help learners use what they know in order to contextualize what is unknown, meaning that since the human brain needs story to provide context. An effective story provides a basis for addressing requirements and framing all content development in order to produce a consistent and compelling product.

- Incorporate assessment needs as early as possible during system design and development. Post-development efforts to graft assessment on a system not designed for it lead to significant problems in capturing the type, quality and quantity of data required for effective assessment.

- Agile development methods work very effectively in fast-paced game-based training development efforts.

- Early involvement of all stakeholders in design and ongoing involvement during design and development iterations leads to a more robust product and helps avoid significant pitfalls.
• Keep the customer involved throughout to ensure the product meets stated and unstated requirements (and adapts to meet changing requirements).

Development Process

In March 2008, our team began the development of a game-based training prototype. Over the course of seven months, we followed an agile development methodology to iteratively create and refine our instructional design, and develop and refine our initial product prototype, which we delivered successfully at the end of September. During our effort, we encountered a number of issues, many of them due to tensions caused by differing perspectives of the project stakeholders based on their backgrounds and relative priorities. All key issues were resolved, practically-speaking, to result in a product that met our customers’ needs and satisfied the immediate goals of the project. However, a number of the tensions remained as background issues and recurring discussion themes. In this section, we summarize the key tasks of the project in a roughly chronological manner, and identify the key tensions that occurred along the way. Figure 1 illustrates the project milestones that are discussed in more detail in this section. The following section expands upon these tensions and identifies the methods used to address them and/or the lessons learned from our experience.

Figure 1: Timeline of Project Milestones

Our project, a three-year effort that started in February 2008, had two mandates - to conduct applied and empirical research into tools and methods for enhancing the state of the art and science of educational games, and to ensure that these developments would have long-term value to the Navy. In order to achieve the latter mandate, we created a prototype training system for the Naval Service Training Command (NSTC) to enhance the Navy’s Recruit Training Command (RTC) curriculum. Our NSTC customer has a background in educational science and
deep knowledge regarding the training needs and culture at RTC. From RTC, we drew upon the training staff as subject matter experts (SMEs). Thus, the stakeholders of our effort included our transition customer (NSTC), our program manager (ONR), the subject matter experts and the members of our team.

To support the first mandate, we identified a need for a game-based training testbed allowing the explicit control of diverse variables in order to empirically study the impact of gaming and instructional features upon learning outcomes. This testbed capability formed an additional requirement for the initial training system we created. The focus of the paper is the process we followed in developing the training system product and our lessons learned from that effort. We describe certain interim and final elements of the product to support our discussion. However, a complete description of the final product is not given.

**Selection of training requirements, domain and gaming platform**

In developing our training product for RTC, our initial step was to identify the specific training needs. At RTC, recruits currently undergo six weeks of intensive training on fundamental navy skills and end their training with an intense exercise called Battle Stations 21 (BS21). BS21 is a real-life simulation environment in which recruits are exposed to seventeen simulation scenarios during a single night. The facility is a building designed to give recruits the experience of being aboard a ship, and contains a simulated dock, a simulated exterior of a destroyer, and several floors that simulate different decks of a ship.

The goal of our effort was to provide supplementary training to augment the current classroom-based and hands-on instruction in order to produce better prepared sailors. While overall performance of recruits in BS21 is excellent, students frequently exhibit key errors in several of the BS21 scenarios due to training gaps on those skills. Our requirement was to provide a compelling virtual training system to address some of those gaps.

At this earliest stage of product development, past experience has shown that it is critical to fully involve the customer. With any training system, the focus of the training must be driven by customer requirements, and the requirements must have a direct relationship to the learning outcomes desired. With any training application, it is important to choose a training domain that is suitable for the type of training possible with the technology used, or, alternatively, to choose the right type of training technology for the type of training desired. In our case, the customer desired an initial product delivery as soon as possible and wanted the product to provide immersive training geared toward single players. The customer desired that the training address a domain that would have high payoff in terms of improving recruit performance in BS21, but also wanted the application to provide familiarization with operating within a (virtual) ship environment. Within those constraints, we had a fair amount of leeway.

In initial discussions with our customer and with SMEs, four key training domains were identified as high benefit: controlling a flooding situation, standing a bridge watch, handling rope and navigating within a ship. Of these, it was determined by all stakeholders that handling rope was the least appropriate for an immersive gaming environment. Navigating within a ship was determined to be of secondary importance in that it could be embedded within training focused on either of the other two domains.

In these early discussions with the customer, it quickly became apparent that the high-level goals of the customer and the project could be best attained by leveraging an existing, open-source game-based simulation prototype developed for the same customer. That application was based on the game engine Delta3D and contained a high fidelity representation
of the ship interior of BS21 with a first-person perspective (see Figure 2). The drawback to the application was that it had minimal pedagogical infrastructure and minimal gaming elements. The advantage of the application was that it would avoid the need for an intensive graphics development process, and, since it was open-source, would allow us to add the pedagogical elements we needed. One of our team members had been the software developer for that application, which also provided us with immediate expertise. From the customer's perspective, reusing the application would justify earlier investments. Further, the customer already knew its strengths and weaknesses and was able to give us specific feedback on what he wanted improved for our product. The earlier application did not, however, contain virtual characters. Preliminary task analysis determined that it would be pedagogically appropriate to provide flooding control training without virtual characters since those skills can be performed by a single individual alone in real-life situations. Bridge watch, in contrast, inherently involves a team of people and thus the application would require augmentation with simulated characters to provide effective training. The bridge watch domain was deemed too risky to meet our aggressive development schedule.

Figure 2: Available 3D Simulation Environment of BS-21 Interior

The final choice of training domain for our initial product - flooding control - was made in early April for practical reasons based on the nominal choice of training platform and the customer's view of which domain would have higher product acceptance at RTC. The process of making this decision revealed one of the first key tensions in the project.

- The basis for technology decisions – pedagogical or technological
Knowledge Acquisition

Once the choice of the flooding control domain was made, we undertook the task of knowledge acquisition for the flooding control domain. The knowledge acquisition methods we employed included a traditional, formal cognitive task analysis (CTA) as well as a subject matter analysis focused upon the potential elements for flooding control game scenarios, termed here a scenario element analysis. The analyses were based upon reviews of training materials in use at RTC, observations of recruits performing the flooding control scenario (and others) in BS21, and discussions with SMEs. In mid-April, a SME session was conducted face-to-face with six trainers from RTC. During this session, the questioning was led by our instructional designers, but representatives of most team members were also present and contributed to the discussions. This session, as well as a couple of follow-up face-to-face and teleconference sessions, provided two initial products.

The first was a traditional breakdown of learning objectives and the identification of expected behaviors, common errors and consequences. The CTA determined that there were four key categories of learning objectives - those related to communication, situation awareness, appropriate decision-making and leadership. The communication learning objectives, for example, included reporting a situation immediately, communicating relevant information and reporting when an action is completed. Decision-making objectives included maintaining watertight conditions, requesting permission before taking action as appropriate, and performing proper containment procedures. In addition, the CTA determined that there were certain skills that the recruits were expected to know based on the current curriculum, certain skills that it would be desirable for them to know if training could be enhanced, and certain skills that were very important in real-life, but that were beyond what could be expected of a normal recruit.

The second product was a breakdown of the elements of the flooding control mission that suggested the potential structure, actions and variations for a simulated flooding control scenario. The scenario element analysis determined the typical timeline involved in a flooding control mission and the specific elements of the different phases of the mission that suggested the potential actions and variations to be included in a simulated scenario. These phases were broadly captured as: the discovery phase (the actions surrounding the event that identifies the need for flooding control), the preparation phase (actions in getting assigned and ready to combat the flood), the transit phase (actions taken en-route to performing flooding control), the casualty combat phase (the procedures and options available while trying to control the flood), and the completion phase (what actions are performed at the end of the situation). These phases summarized the SMEs’ views of the typical way in which a flooding mission occurs and what they felt needed to be reinforced with the recruits. Within each phase, a number of activities, possible actions and scene variations were identified. For instance, in the transit phase, additional hazards could be encountered, such as additional leaks, additional flooding locations, unsecured items in hallway, injured personnel and watertight doors not closed appropriately. During different phases, inappropriate setting of a watertight boundary could lead to various complications, including allowing a flood to spread or trapping a shipmate. In addition to multiple phases of a mission, the scenario element analysis identified multiple roles involved in different phases of a flooding control mission, including a first-on-scene sailor, a damage control (DC) team leader or member, and damage control central (DCC). The SMEs indicated that the recruits should be taught certain skills associated with certain roles, even if they would typically not be in those roles for some time.
The products of the CTA and scenario element analysis were then merged to form a third product that mapped the learning objectives in context across the phases and roles (e.g., Figure 3 shows one of 26 learning objectives mapped). This product was prepared by May 1 and provided inputs to our story development process.

<table>
<thead>
<tr>
<th>Learning Objective</th>
<th>Expected behaviors (enabling objectives)</th>
<th>Discovery Phase</th>
<th>Preparation Phase</th>
<th>Transit Phase</th>
<th>Casualty Combat Phase</th>
<th>Completion Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication - Demonstrate they know (a) whom to contact and (b) what to say and (c) how to listen and repeat back</td>
<td>Use best available means to communicate</td>
<td>Role: First on scene</td>
<td>Specific objective: Notify someone of the flooding situation immediately</td>
<td>Expected behavior: Report flooding immediately upon discovery. Use best available means to communicate (phone, yell, send messenger)</td>
<td>Common errors: Do not communicate. Ignore problem, isolate location without permission, combat casualty on own. Consequences: No one else knows. Leak/flood progresses quickly and DC has less time to get there. Scenario variations: Individual injured or dies while combating. Individual electrocuted due to unperceived electrical hazard. Ship capability reduced. Ship sinks.</td>
<td>Role: DC team member</td>
</tr>
</tbody>
</table>

**Figure 3: Learning Objectives and Common Errors in Context of Flooding Mission Phases and Roles**

During the knowledge acquisition process, several tensions arose, including:

- **Identifying all the learning objectives versus selecting the objectives to be addressed in the first game prototype**
- **Balancing cognitive elements with experiential elements**

**Story Development**

Great stories share at least six elements that set them apart from “scenarios” or “case studies”. They are: setting, characters, conflict/resolution, plot, voice, and emotion. Compelling stories, and those experienced in game environments in particular, are often called “immersive.” However, immersion doesn’t just happen; it takes purposeful effort on the part of the storymakers or designers. To achieve a great story, one must be intentional about combining the six elements in a way that will draw the audience (in the case of a game, the players) into the story so that they can see themselves as part of it. To do this, the story must be easy for them to relate to. To be successful as a learning environment, it must also create an “envelope” that will carry the learning tasks so that the learner literally “embodies” them.

Generally, fodder for narrative development in a learning product can be gathered in early conversations with the customer. In our effort, during the knowledge acquisition process, we determined through discussions that the basic objective of our initial product would be to teach recruits the steps required to stop (or mitigate the flow of) a leak on board a ship. We were thus able to begin to zero in on some of the key story elements from the beginning. For instance, from these early discussions, we knew the following about the eventual story:

- The story **setting** would be on (some type of) a ship.
- At least one of the **characters** would need to be a sailor on board that ship.
• The conflict to resolve would be a leaky pipe or breach in the ship’s hull that the sailor would need to fix or at least mitigate the effects of.
• The plot would need to include some sort of potential flooding scenario.
• We believed that the emotion of fear could be used to engage the learner as a crisis ensued. We also believed that the Navy values of “Honor, Courage, and Commitment” that were instilled in recruits during basic training could be used to provide motivation to overcome the fear and lack of initial skills, competence, and confidence to perform the mission.
• Voice, or who the “teller” would be, was not established until later.

On May 6-7, we held a two day story conference. We came together as a team of instructional designers, programmers, writers, and artists to begin to create the metastory for the game. Our customer participated and provided subject matter expertise as well as general guidance. The instructional designers kicked off day one by reviewing the learning objectives with the group so that we would stay focused on those throughout the day. The group was led by experienced story development facilitators through activities chosen to help us experiment and “play” with character development, setting, and plot points that would drive the learning objectives involved with a flooding scenario. It was important to develop a metastory that would be believable and that a new recruit would identify with as well as one that would set us up for the kind of conflict or crisis that could incorporate other possible disaster prevention/recovery scenarios (e.g., fire fighting) over time.

During the discussions, our customer emphasized the need to reinforce the pattern of "look-report-do" when training recruits. This led us to focus upon the situation awareness, communication and decision-making learning objectives, and to de-emphasize leadership. To further focus the story, our customer in particular and the team in general prioritized all the enabling learning objectives based on training requirements and perceived utility.

After the story conference, the story developers used the notes from the two-day event and developed a draft of the metastory which included several possible game levels. During the first pass at the story, we decided that the background story ("backstory") would be told in the third person by a narrator that was not a part of the story itself. This backstory eventually became the introductory movie or “intro cut scene” of the game.

During story development, a key tension was:
• Story first versus design first

Game Design

On May 29-30, we held a two-day team meeting to flesh out the specific training scenarios for our product. A representative from the customer attended, but no SMEs did. Using the learning objectives and story construct as a basis, the team brainstormed on how we wanted to provide training within the game. We made the decision to couple a guided discovery instructional strategy with an adventure-style gaming strategy. Following a common gaming approach, we decided to create a game with multiple levels of increasing difficulty. Based on the scenario element analysis, we decided to structure each level as a single mission, possibly with some “twists” to add to the sense of adventure. To support our instructional and gaming objectives, we identified a general mechanism for using dialog and messages from game characters to provide mission objectives, story and adventure elements, and instructional guidance and feedback. To avoid the need for animating the non-player characters, we designed
the game in such a way that these characters would be elsewhere in the ship, out of view. Further, we incorporated the capability to provide students with opportunities to fail, possibly catastrophically, depending upon their errors.

Early on, we realized that we would need a tutorial level to train students on how to play the game. This is a standard mechanism used in commercial games to bring players up to speed. The instructional designers deemed it important that this tutorial level would introduce the basic instructional mechanisms to lay the foundation for how the game would deliver its training. The game designers emphasized that the tutorial level should reinforce the mission metaphor to introduce the gaming strategy and to set student expectations. The story developers encouraged the incremental introduction of story elements throughout all levels, including the tutorial level. We converged on a tutorial level design that met all these goals.

Using the multiple levels of expertise identified in the CTA and our earlier rankings of the relative priorities of different learning objectives, we fleshed out the basic enabling learning objectives for the initial game levels. By the second day, we had designed the outline of tutorial level and three training levels of increasing difficulty. For each level, we identified which behaviors we wanted to assess, the key story developments, potential instructional scaffolding, desired immersive elements (such as sounds, lighting and game objects and effects), and the general flow of the gameplay.

<table>
<thead>
<tr>
<th>Story Development: Approaching area of operation, prepare for underway replenishment (UNREP)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Tutorial Level</strong></td>
</tr>
<tr>
<td>Inspection: Inspect and secure repair locker</td>
</tr>
<tr>
<td>Objectives: Game mechanics, Communicate using phone.</td>
</tr>
<tr>
<td>Assess: Time taken. No failure and no negative feedback. High guidance.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Story Development: Ship collides with oiler during UNREP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mission 1 Level</strong></td>
</tr>
<tr>
<td>Investigate and Repair: Assess and repair leaking pipe</td>
</tr>
<tr>
<td>Objectives: Communicate with DCC, Maintain watertight integrity</td>
</tr>
<tr>
<td>Assess: Intent and accuracy of actions</td>
</tr>
<tr>
<td>Feedback: Guidance primarily in response to errors. Negative feedback and potential catastrophic failure</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Story Development: Approaching area of operation, prepare for underway replenishment (UNREP)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mission 2 Level</strong></td>
</tr>
<tr>
<td>Repair Complex Flood: Assess and repair multiple leaks</td>
</tr>
<tr>
<td>Objectives: Establish boundary, Apply appropriate repair, monitor and report situation, prioritize decisions, dewatering</td>
</tr>
<tr>
<td>Assess: Intent, accuracy of actions and effectiveness of decisions</td>
</tr>
<tr>
<td>Feedback: Low guidance, multiple ways to fail, complete success possible but order of decisions matters</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Story Development: Approaching area of operation, prepare for underway replenishment (UNREP)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mission 3 Level</strong></td>
</tr>
<tr>
<td>Rescue: Combat flood and ensure safety of equipment or personnel at risk</td>
</tr>
<tr>
<td>Objectives: Maintain safety of ship. Assess and report changes</td>
</tr>
<tr>
<td>Assess: Appropriateness of decisions</td>
</tr>
<tr>
<td>Feedback: Low guidance, complete success not possible – hard choices necessary, multiple “passing” mission completion states</td>
</tr>
</tbody>
</table>

*Figure 4: Potential Game Levels Produced during Game Design Session*

At this point we captured and prioritized all the desired instructional and gaming capabilities. We determined that the initial prototype would focus on the tutorial mission and the first training mission ("Mission 1"). This provided the basis for the software requirements and the launching point for further instructional design.
The tension revealed here was:

- Gaming strategy versus instructional strategy

**Initial Instructional Design**

Following our May 29-30 meeting, the instructional design process began in earnest. Our instructional designers, assessment experts and game developers worked iteratively to identify the specific instructional elements of the proposed “Mission 1” game level based on our cognitive task analysis and our guided discovery instructional strategy.

An initial instructional design was developed by mid-June that elaborated on the learning objectives associated with each level and suggested the general instructional approaches to be used in the product. The guiding principles of this initial design were to

- Address objectives in story, rules, consequences
- Design consequences to dispel misconceptions
- Provide feedback to promote learning
- Use scaffolding to guide players and allow them to experience actions beyond their level to reinforce learning objectives
- Use gaming technologies to facilitate the development of mental models

The instructional approach was a holistic design for providing part-task training within the context of the whole task. It specified multiple forms of feedback for alerting players to deficiencies and providing opportunities for players to demonstrate skills again in similar but not necessarily identical situations. It attempted to carefully control variations to ensure that the player did not just learn the game mechanics, but also understood the underlying facts, concepts or principles.

The initial design also identified a general approach for assessment within the game. The types of assessment envisioned included completion of tasks, accuracy, time taken and steps taken. The intent was to assess learning objectives throughout the game by the actions the player would take. These assessments were also expected to affect play. The player’s actions could be limited until he or she demonstrated a predetermined level of mastery. A final performance assessment would be given at the end of the game that would detail strengths and weaknesses. In this final assessment, the player would be informed of how well he or she did on each of the game performance criteria and how to improve performance in the future. Drawing upon the CTA, the scenario event analysis and the basic game level design, the initial instructional design identified a series of key scenario events, the associated learning objectives and the associated assessment requirements (see Table 1).

**Table 1: Mission 1 Level Scenario Constraints and Learning Objectives**

<table>
<thead>
<tr>
<th>EVENT</th>
<th>ENVIRONMENTAL REQUIREMENTS</th>
<th>LEARNING OBJECTIVES</th>
<th>ASSESSMENT REQUIREMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Readiness set</td>
<td>Alarm sound</td>
<td>Decision-Making</td>
<td>Learner immediately leaves for general quarters assignment</td>
</tr>
<tr>
<td>Readiness condition is set as a result of ship-wide “event” - General Quarters</td>
<td>Lighting Effect</td>
<td>• Respond to readiness condition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Announcement over the sound system</td>
<td>• Situation Awareness</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Report to proper location</td>
<td></td>
</tr>
<tr>
<td>Navigate Ship</td>
<td>Circle X, Y, &amp; Z doors to open/close</td>
<td>Decision-Making</td>
<td>• Used phone to ask permission for opening doors</td>
</tr>
<tr>
<td>(recurs throughout game)</td>
<td>Limit space to task area – Control where they can go.</td>
<td>• Comply with restrictions on doors</td>
<td></td>
</tr>
<tr>
<td>Learner navigates within the ship</td>
<td>Phone on wall in destination</td>
<td>• Situation Awareness</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Identify location</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Communication</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Ask permission before opening door</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Used phonetic alpha/num</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Path taken</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Repeated back instructions</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Located the proper</td>
<td></td>
</tr>
</tbody>
</table>
The design specified several different types of feedback to be used in the game, such as:

- Natural consequences would be used, when possible, to indicate the effects of the player's action. The player would be made aware of the relationship between their action and the consequence (e.g., the player must know that turning the valve cuts off the water supply for fighting a fire).
- Hints would be given to reinforce or correct declarative knowledge, for example by telling the player the general rule or protocol.
- Catastrophic failures that immediately stop the game would be used to emphasize critical errors made by the player.
- Game scores would be used to provide feedback on the key behaviors to be developed and to reward the attainment of desired learning outcomes.

Finally, the design specified several possible types of scaffolding to consider, including:

- Limiting the play space and/or choices offered to avoid cognitive overload and focus attention on key cues in the environment.
- Providing hints that focus on the facts, rules, and or steps in a procedure.
- Providing just-in-time information, particularly at novice levels, when the player is not expected to know that information or when that information is not critical to achieving the learning objective.
- Modeling or demonstrating procedures using gaming techniques or incorporating embedded videos.
- Providing advance organizers, such as a schematic of the ship to aid navigation.
- Providing characters to guide the player in decision making when a player has been unsuccessful after several attempts.
- Providing access to “mini games” as needed to increase proficiency in basic skills.
Following a traditional game technique, the notion of a "score" was initially suggested as a form of implicit feedback, and the use of specific warning messages in response to student errors was suggested as a form of explicit feedback and guidance.

During this initial instructional design process, two key tensions were revealed:

- **Directive instruction versus guided discovery.**
- **Feedback for learning versus feedback for motivating gameplay.**

**Assessment Strategy**

To accomplish the task of embedding effective assessment in our game, our design process dictated that we begin by examining the damage control domain - a broad class of activities of which flooding control is a part. Working closely with the instructional design team, the assessment experts identified the key constructs and relationships that formed the sub-domain of flooding within the damage control domain and mapped these constructs and relationships into a Flooding Ontology (a visual representation of the constructs and that make up flooding and how those constructs are linked). The ontology was divided into three broad constructs – Situation Awareness, Communications, and Decision Making—as these encapsulated the cognitive and behavioral elements relevant to addressing a damage control situation aboard a ship. The goals and objectives of the training, along with the assessments, would be designed around these core concepts and relationships. Given this general Flooding Ontology, the specific elements relevant to the Mission 1 level were identified based on the initial instructional design (e.g., see Figure 5).

![Figure 5: Partial Flooding Ontology with Mission 1 Related Elements (in gray)](image-url)
With this ontology completed, we drafted a preliminary set of functional game requirements necessary for player performance to be sufficiently assessed. These requirements were specific features that needed to be represented in the game to serve as indicators of performance. They included elements the player could use to communicate with Damage Control Central (to assess communication skills), opportunities for the player to choose from a variety of apparatus the appropriate one to contain the leak (to assess content knowledge and decision making), and opportunities for the player to interact with different types of doors under various material readiness conditions (to assess the player’s understanding of readiness conditions and adherence to permission protocols).

Based on the initial instructional design and the general approach to gameplay, we proceeded to flesh out the detailed automated assessment strategy to be applied. The key learning objectives were mapped to key assessment requirements. We then identified, for each assessment requirement, the specific metrics that were possible to determine automatically in the game. These were tied as appropriate to specific scoring computations or feedback actions. The final assessment strategy incorporated the pedagogical objectives while respecting the need to have assessment fit in naturally with gameplay. The game score was defined to be monotonically increasing (i.e., total score could only increase in value, and would not decrease due to errors). The second iteration of instructional design, incorporating an assessment and feedback strategy for all learning objectives, was completed in mid-July.

In our assessment strategy, we attempted to identify behaviors that could lead to a failure of the mission. These serious errors would produce a demerit that interrupted gameplay to provide important feedback. Table 2 provides an example of the assessment strategy for a single learning objective (i.e., Follow safety protocol), showing implicit feedback in response to accurate behavior via a change to the game score, as well as explicit "demerit" feedback in response to errors. Many of the details of the assessment strategy changed during subsequent refinements.

Table 2: Sample Specification of Assessment and Feedback for a Learning Objective

<table>
<thead>
<tr>
<th>Skill Area</th>
<th>Learning Objective</th>
<th>Assessment Requirements</th>
<th>Metrics</th>
<th>Positive Indicator</th>
<th>Negative Indicator</th>
<th>Game Score Feedback Effect - Success</th>
<th>Numerical/ Aggregate Metric</th>
<th>Scoring/ Feedback Effect - negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision-Making</td>
<td>Follow safety protocol</td>
<td>Did not enter the flooding compartment without permission.</td>
<td>Intent</td>
<td>Asked for permission to enter a room in which student had (incorrectly) reported flooding</td>
<td>Never ask for permission to enter a room</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Accuracy</td>
<td>Asked for permission to enter the actual flooding room without asking permission</td>
<td>Enter flooding compartment without asking permission</td>
<td>10 points if student enters the room after requesting and receiving permission</td>
<td>1 count per flooding compartment; 1 count per “first” entry into flooding compartment; 1 count per correct request to enter flooding compartment</td>
<td>Demerit: A demerit is issued if the student enters the flooded compartment without receiving permission.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Multiple instances (flooding compartment)</td>
<td>N/A - only one flooding compartment in this level</td>
<td>Only keep track of requests prior to the first entry into the flooding compartment</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Multiple occurrences (Over time)</td>
<td>Only keep track of requests prior to the first entry into the flooding compartment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Accuracy percentage: Total correct request / Total flooding compartment
As the detailed assessment strategy was being developed, preliminary instructional logic was also created to identify when to capture data for assessment of player performance, as well as to identify possible situations for performance-based feedback opportunities. The logic for several of the behaviors involved in a flooding control mission was captured by mid-July using flowcharts such as that in Figure 6. The motivation behind these initial flowcharts was to capture an "ideal" trainer by working forward from the ontology and recommended functional assessment requirements to produce a guided discovery instructional strategy that would also support effective assessment. Consideration of gaming strategy, story and mission flow were left as future integration exercises. The logic allowed for multiple attempts to accomplish tasks while providing different guidance and constraining user actions depending upon the errors made (and amount of error repetition).

![Flowchart Diagram](image)

*Figure 6: Preliminary Guided-Discovery Instructional Logic for Obtaining Appropriate Permission Before Opening Doors (light gray = direct instruction, dark gray = implied instruction, black = goal achieved)*

After mid-July, a collective effort was made to resolve the different representation formats preferred by the instructional designers, assessment experts and game designers and to integrate the various aspects of the product design. The most commonly agreed-upon format was the use of flowcharts. Using the preliminary instructional logic flowcharts (e.g., Figure 6) as a starting point, new flowcharts were defined for the key learning objectives of the mission 1 level. Figure 7 illustrates a flowchart specifying the instructional logic for "Following safety protocols", as determined by donning the appropriate personal protective equipment (PPE). Each flowchart provided a specific set of rules indicating under which circumstances specific guidance messages...
(dashed border, light-gray), positive reinforcement messages (dotted border, light gray) or negative feedback messages (solid border, light gray) should be given. These flowcharts integrated the assessment strategy and basic character interactions to identify the instruction, scoring and dialog to be implemented in the game. New flowcharts covering all key situations in Mission 1 were completed in mid-August. During this revision, a second score mechanism was added to keep track of penalties associated with errors. Specifically, when a negative feedback message was given in response to an error by the student, a negative demerit score value was decreased by some amount. The demerit score (trapezoid) and positive score (hexagon) were separate values. The possibility of mission failure due to a high total demerit score (e.g., −1.0) was also introduced.

**Figure 7: Complete Instructional Logic for Learning Objective of “Following Safety Protocols: Don Appropriate Personal Protective Equipment (PPE)”**

During the development of the assessment strategy and the associated instructional design revisions, two key tensions arose.

- **Developing an objective versus developing methods for assessing attainment of that objective**
- **The right documents for sharing knowledge**
**Software Development**

By mid-June, a software requirements document and an initial software design document were completed that identified the key changes needed to the existing simulation in order to support our desired training game design. In addition to supporting the instructional levels we had designed, we adopted an additional requirement to adapt the gaming infrastructure so that pedagogical elements of the training would be specified as much as possible using a data-driven approach rather than implementing a hard-coded game. The goal of a data-driven approach was to facilitate rapid changes to instructional content.

An aggressive agile development schedule was determined with four end-of-month deliverables, the first at the end of June and the last for our final product at the end of September. The first deliverable would incorporate a basic mission flow (incorporating a briefing screen and a debriefing screen) and include some preliminary user interface elements (such as feedback and information windows). The second deliverable would provide an end-to-end walkthrough that exercised basic missions and basic forms of all interface elements. The third deliverable would include the data-driven infrastructure and all key graphics and animations needed to support the tutorial and main mission. The final deliverable would have a fully functional tutorial and the main mission.

In particular, the data-driven design adopted was as follows. A mission would be comprised of multiple tasks. For each task, specific trigger events would initiate it, and completion of the task would in turn initiate one or more other tasks. The specific tasks in a scenario and all details concerning each task, such as its triggers and its description, were specified in the scenario data file. Further, a mission could contain multiple "score" objects, each maintaining a distinct value representing user performance. A score object could be triggered by direct player actions, such as dialog choices or entering a room, as well as by indirect actions, such as completing a task. The score object would contain specific feedback messages to provide to the user upon being triggered (e.g., via a "demerit" message or an instructional guidance message), and would maintain internal state reflecting the number of times triggered. The message could vary depending upon the number of times triggered. The specific score objects and all details concerning them, such as their triggers and messages, were specified in the scenario data file.

The key tension of the software development were

- **Hard-coded versus data-driven implementation**

**Introductory Movie**

In late June, as we iteratively refined our story, we realized that our product would be particularly enhanced by the incorporation of an introductory movie scene that would present the backstory and lay the foundation for motivating the student. The use of introductory movies and cut-scenes within and between game levels is a powerful method used in commercial games to enhance the player's sense of immersion in the game. For our product, we determined that the introductory movie needed to:

- Motivate the desire to play
- Motivate the desire to serve in the Navy
- Introduce the backstory
- Promote the Navy core values of Honor, Courage and Commitment
• Introduce the ultimate game objective
• Orient the student’s approach to playing the game

The introductory movie design was captured in a table that divided the script into small one- or two-sentence segments to facilitate the association of dialog with the appropriate on-screen visual. The team met via phone conference to discuss what should be seen while the narration was spoken. In order to hasten development and save money it was decided that characters would only be seen when absolutely necessary to the story. Likewise, we decided that in order to allow the learner to relate to the game character, the player would not be identified as male or female and that any time the learner had to “speak” it would be via text only. This meant that the learner would not ever see or hear himself or herself as a character in the game. This kept the player “generic” and allowed the learner’s imagination to fill in the gaps. Thus, a player could be immersed in the game without the necessity of choices of character art or voicing.

An artist then created storyboards (sketches) for the intro movie. The sketches were scanned and placed into the draft of the script. This document was shared with the larger team (including the subject matter experts) and feedback was incorporated into successive drafts. To stay true to story and reality, we decided it was necessary to have one non-player character (the officer of the deck who welcomed the learner aboard). Once the storyboard and script were approved by the development team and SMEs, the artists began to create the art and animations for the movie. For maximum effect, a 3D modeling approach with realistic ships and naval scenes was used. After a number of iterations with frequent SME feedback on the accuracy and suitability of our models, the graphic rendering of the movie was completed.

In parallel, a narration and sound script for the scene was finalized with inputs from several team members and from our SMEs. This process also involved the creation of a “scratch track” of the narration of the introductory movie that was reviewed by our customer. Once the intro video was created, it was passed to our sound production team. Professional voice talent was used to record the narration for the movie, and the audio for the movie was finalized and enhanced with sound effects and music to add to the drama. The introductory movie was then incorporated into the game to be played before the tutorial mission.

There were minimal tensions raised during the development of the introductory movie. However, since the development of the movie was a substantial task, it was important to ensure that it was general enough to accommodate further story refinements in the game itself. During development, the specific wording used in the narration was varied slightly several times to reflect changes in the specific missions (particularly in the mission briefings).

**Iterative Review, Refinement and Testing**

The instructional design, game level design, introductory movie and software design were all refined in an iterative manner after the end of July. Efforts were conducted in parallel and discussions on one aspect of the product often involved team members working on other aspects. Customer design and product reviews were held bi-weekly (on August 4, August 17, September 4, September 17 and October 8). Three rounds of product testing with test subjects were held both before and after the initial product release. A pilot usability test was held on September 18, an end-user usability study on October 22-23, and a pilot validation study on November 17. Following each review and test, desired improvements were identified based on
feedback and observations. These were then prioritized based on the importance of the changes and time/resource constraints.

In particular, four instructional elements were iterated upon and refined right up until the beta release in mid-September: instructional objectives, assessment methods, dialog content, and mission debriefing.

On-going refinement of the instructional objectives of Mission 1 led to the following final set of objectives being approved by the customer on September 15 (see Figure 8). For each broad skill category (Situation Awareness, Communication, Decision making), several terminal objectives were defined. For each terminal objective at least one enabling objective was defined.

![Diagram of instructional objectives]

Figure 8: Final Instructional Objectives: Skill Areas - Terminal Objectives - Learning Objectives

In the final implementation, the student's performance against every terminal objective was assessed automatically via the student's actions in the game and choices in dialogs. Dialog interactions formed a key method for assessing user performance against a variety of communication and situation awareness learning objectives. These assessments were context-sensitive (i.e., the same dialog choice may be correct or incorrect depending upon prior user actions). As shown in Table 3, a single dialog interaction could result in errors against different objectives (e.g., reporting appropriate versus accurate information), and different types of feedback (e.g., dialog responses versus demerits).
### Table 3: Assessment and Feedback based on Student's Dialog Choices

<table>
<thead>
<tr>
<th>Dialog choice</th>
<th>Assessment</th>
<th>Feedback</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;DCC, this is Seaman Taylor. I'm dressed out and ready to help&quot;</td>
<td>Incorrect</td>
<td>Dialog response from DCC: &quot;Seaman Taylor, DCC. I asked you to go to compartment 1-80-1-Q and report when ready to enter. Get on it!&quot;</td>
</tr>
<tr>
<td>&quot;DCC, I am at space 1-80-1-Q. Request permission to enter and inspect.&quot;</td>
<td>Error against 'Report Appropriate Information to DCC' learning objective</td>
<td>Dialog response from DCC: &quot;Sailor, this is DCC. Who is this?&quot; and Demerit (e.g., &quot;A proper report should include your name and all relevant information about a situation.&quot; and 0.1 demerit score change)</td>
</tr>
<tr>
<td>&quot;DCC, this is Seaman Taylor. I am dressed out and ready to enter 1-80-1-Q. Request permission to enter and inspect compartment.&quot;</td>
<td>Correct if student is actually reporting from a phone near 1-80-1-Q. If reporting from wrong location, then error against 'Report accurate information to DCC' learning objective</td>
<td>Seaman Taylor, DCC. Aye. Permission to enter and inspect 1-80-1-Q granted. Report situation as soon as possible. Dialog response from DCC: &quot;I am not showing you near that space. Go find 1-80-1-Q and call me from the nearby phone.&quot; and Demerit (e.g., &quot;You made a bad report. The DCC trusts you to be the eyes on the scene. Reassess the situation and give the DCC a better report.&quot; and 0.3 demerit score change)</td>
</tr>
</tbody>
</table>

The mission was comprised of multiple tasks, and the student's actions were interpreted in the context of the current sub-task(s). At the end of the mission, the student was provided a debrief that summarized their performance against every terminal objective on a three-point traffic light scale (green, yellow, red). An analogous design was adopted for the tutorial mission, though with a much reduced set of terminal and learning objectives.

Tensions that arose during the iterative refinement process included:
- **Designing for the novice while keeping the gamer happy**
- **Balancing revisions to instructional design with meeting software deadlines**
- **Gaming strategy versus instructional strategy**

### Multidisciplinary Tensions and Lessons learned

**The basis for technology decisions - pedagogical or technological**

The tension revealed by the intertwined choice of training domain and gaming platform concerns the weight that should be given to pedagogical or technological factors when making decisions about which training technology to use. In our case, the decision was made primarily for technological reasons, and not for pedagogical ones. For instance, the view of the instructional designers and assessment experts was that the domain should have been chosen first, the cognitive task analysis performed next, and then, based on the key learning objectives identified, the technology choice should have been made that would best suit those objectives. Thus, choices such as 3D immersive versus 2D interactive environments, single-player versus multi-player, simulated characters versus no characters, and so on should, in principle, have been driven by pedagogy. The lesson learned is that the practical considerations of customer preferences, schedule and available resources can significantly impact these choices.
**Identifying all the learning objectives versus selecting the objectives to be addressed in Level 1 of the game**

The multi-disciplinary approach to instructional design had a clear effect upon the decisions made on how to sequence training in the game. The instructional designers, assessment experts, game designers and story developers all agreed upon a layered approach in which the training was broken up into multiple levels of increasing difficulty. The challenge was determining what was meant by “increasing difficulty”. Everyone had a slightly different understanding of how the learning objectives might be linked to form a beginning level scenario that would be both engaging and prepare the target audience for a live simulation of the entire flooding process. Many interesting, collaborative discussions occurred in refining exactly what would occur in the earliest missions.

The instructional designers envisioned a layered approach focusing on the key steps in the flooding control process in all levels, but varying the complexity of the skills expected. In order to allow the students to experience the entire process in earliest levels, instructional designers advocated that some of the smaller steps in the process be provided to the learner or that the environment be structured in such a way that the player would not have to demonstrate more specific skills until higher levels of the game. Their primary goal was to build a foundation to ensure that players had a basic understanding of what happens in a flooding situation, and to reinforce all the key process steps in all the levels. The story developers encouraged the idea that not all information about the situation needed to be provided at the beginning, and that an incremental introduction of new story elements to provide context for new learning objectives over successive levels would be effective. The game designers wanted to keep the early levels simple and slowly build up the complexity of the gaming skills needed over multiple levels. Unlike the instructional designers, they did not feel that all aspects of the process needed to be present in every level (i.e., certain levels could focus upon certain elements of the process). The assessment experts were concerned primarily with cognitive overload and wanted to ensure that new skills and information were introduced in an incremental fashion over successive levels.

The decisions regarding which learning objectives to focus on in the first level were made with the intent of satisfying these goals as much as possible, and multiple refinements were needed over time to reach consensus. For example, there were several potential learning objectives pertaining to the rules for opening and closing doors under various readiness conditions. In our developed ontology all possible types of doors and hatches and appropriate entry rules were presented. In a classroom, these rules would likely be learned and tested with a paper and pencil test. An embedded tutorial addressing these rules was possible, however, we did not want this to be our primary instructional target for the beginning level of the game. To do so would burden the learner with information overload and require many permutations for mastering the distinctions governing the handling of different types of doors when in reality, given the readiness condition already set by the collision, most doors would require the same treatment. We resolved the issue by limiting the kinds of doors visible to the player so that the correct procedure would be technically correct while presenting limited cognitive load. Thereby, the learner could focus on the larger steps required to identify, report, and combat the flood. This solution managed the cognitive load appropriately, maintained cognitive and physical fidelity, kept the learner engaged in the story and prevented negative training.

A lesson learned from this key tension is to maintain communication and seek mechanisms for closer collaboration. Presenting the learning objectives and developing an ontology are critical to developing the appropriate story, rules, and consequences in the game.
Instructional designers are accustomed to look at the relationships among objectives in terms of the development of student knowledge and skills and designing for a scaffolded progression based on skill level. Further research is needed to determine whether and when, for game-based training: every skill must be taught from the bottom up in level 1; levels should provide part-task training; or hybrid solutions will be effective. There is more than one way to develop skills, and gaming technology offers unique opportunities to develop skills that are not available through the use of other instructional media. We can select skills on which to focus in a specific scenario and manipulate the physical fidelity of the environment to meet the instructional design. In a game, this can mean restricting the environment or game space or sequence of events or having other characters perform tasks that are beyond the expectations for the level 1 player. There is a fine line between selecting tasks that would be fun but not expected to challenge the learner, and selecting tasks that would confuse or distract the learner at level 1.

**Balancing cognitive elements with experiential elements**

A difference in perspective between the instructional designers and the game designers was the basis for understanding and characterizing about the domain. During the knowledge acquisition process, the instructional designers gravitated towards a focus upon the cognitive elements of the domain (via a CTA) while the game designers gravitated towards a focus upon the context in which learning would occur in the domain (via a scenario element analysis). The goals of a CTA and a scenario element analysis are different. The former is focused on performance and associated knowledge states. The latter is focused on identifying the environmental contexts in which simulated activities occur, the steps of and interactions among those activities, and the possible variations in the details of those activities or contexts. Both analyses were conducted with collaborative discussions among team members. However, there were frequent misunderstandings of how to characterize the domain. Eventually, it became apparent that the two analyses were capturing complementary information. The exercise of merging them (e.g., Figure 3) resulted in several interesting discussions in which some previous misunderstandings were resolved. One benefit was that additional enabling objectives were identified. For example, in the context of the transit phase, an enabling objective for reporting a situation immediately is reporting any hazards encountered en route to the flooding area. On the downside, the merged document was difficult to follow for team members who had not directly been part of the knowledge acquisition process.

The lesson learned is that both the cognitive and context aspects of a domain are critical for game-based training and each leads to different ideas on how to structure the learning experience in a game. More investigation into effective methods for characterizing and refining the two domain aspects is needed.

**Story first versus design first**

In our effort, a lot happened very quickly. There was a general feeling that the story development (i.e., the StoryJam™ held on May 6-7) occurred too early in the process since the learning objectives from the CTA were not fully understood and had not yet been prioritized. However, without the forcing function of trying to develop a focused story to support the training, the necessary type of filtering and prioritization of the objectives may not have occurred in a timely manner. Throughout the successive refinements of the instructional design and game design, the basic story elements established at the early meeting provided positive guidance that encouraged convergence of ideas and consensus on decisions. The story decisions were also
remarkably stable. For example, the supporting backstory and story elements used in making decisions in the May 29-30 game design meeting were largely present in the final design several months later. The participation of SMEs as well as the customer in the StoryJam was instrumental in establishment of a credible story.

Further research is needed to identify the best way to time and use the story development process to positive effect in game-based training development. However, a clear lesson learned is that a collaboratively developed story facilitates collaborative decision-making during subsequent development.

**Gaming strategy versus instructional strategy**

The key area of difficulty reflecting the multidisciplinary conflicts in the team was the relative emphasis of instructional elements and gaming elements. There are many different types of computer games and there are many different ways to structure instruction. In developing game-based training, open questions include what type of game is best suited to a particular instructional strategy as well as what type of instructional strategy is best for a particular type of game. When choosing a gaming strategy, one is typically choosing a particular suite of game mechanics (interactions the player may have within the game) and a particular type of event flow. Likewise, when choosing an instructional strategy, one is typically choosing a particular suite of instructional interactions with the student and a particular organization to those interactions.

A general methodology for determining how to provide game-based instruction for a particular domain is to map the enabling objectives to specific game mechanics and specific encounters within each level to ensure that the game player is learning what the designer has set out to instruct. During the knowledge acquisition process of our effort, we discussed a variety of possible game mechanics for various enabling objectives. In fact, the SMEs would occasionally proactively suggest a means for how they would “teach this in a game”. In our case, these initial characterizations of the instruction led to an early choice of two “complementary” strategies - guided discovery instruction using adventure style gaming. Guided discovery instruction uses implicit and explicit interventions to encourage and focus a student's exploration of the training domain to achieve the learning goals. Adventure style gaming uses carefully placed hints and clues to encourage the player to continue to explore the game world and achieve the adventure's goals.

This choice led to a lot of synergy at the beginning of the design process. Instructional designers would suggest a need for an intervention, and the game developers could easily map this into an event furthering the adventure. However, as the instructional design process progressed, issues of how to embed new instructional elements often turned into discussions of how the "gameplay" might be adversely affected. Issues that caused the greatest tension were the instructional designers’ and assessment experts' desires for increased guidance and feedback explaining all the student's mistakes as they were made, explicit didactic information on every element of relevance in the scenario (e.g., a help lookup facility), and increased scaffolding, such as the use of a compass-like aid to assist in navigating around the ship. Their goal was to reduce the student's cognitive load and to ensure that the student formed good mental models as early as possible during training. Countering this, the game designers wanted to minimize the amount of non-embedded information (i.e., not delivered as a natural part of interacting with the environment and other characters) to reduce negative impacts on the player's immersion and to maintain the sense of adventure. After many discussions and necessary compromises on
everyone’s part, we succeeded in resolving most issues to produce a final instructional design and
game design that supported one another.

Hence, the two related lessons learned are that linking gaming strategy to instructional
strategy can lead to good synergy of design and a compelling experience, but that an early choice
of gaming strategy can lead to difficulties in trying to incorporate incompatible instructional
elements later on.

**Directive instruction versus guided discovery**

Expectations for the design of directive instruction and discovery learning are not the
same. Our task was to pull the best elements of each into an engaging game that fosters learning
through guided discovery. Just how much guidance is required in guided discovery? Which are
the elements that must be more closely guided? When is there no harm in letting learners play
until they get it right versus when must one guide them so they are not bogged down by details
that are insignificant and detract from learning (i.e., avoiding situations in which the learner
becomes overwhelmed with less important details and misses the key learning opportunity)? A
prime example was in reading the compartment identifiers, or “Bull’s Eyes”, which indicate the
location and type of compartments. We purposefully focused the player’s attention on two of the
elements in the four element series of number and letters that defines a compartment identifier.
The goal was for the player to learn to use the two elements well enough to navigate fore and aft
and to recognize which side of the ship they were on. Once these tasks were mastered the
additional elements could be addressed. In the game, however, how long should the player
wander before being guided to information that will help him or her reach the intended
destination?

The lesson learned is to use the cognitive task analysis to focus on identifying the actions
a student must make to carry out a task as well as the typical wrong paths a student can take. In
experiential learning, we include these actions in the environment and make some of them
options – deliberately selecting these to promote successful task development while not
overwhelming the student.

**Feedback for learning versus feedback for motivating gameplay**

Games offer opportunities for players to see the consequences of their actions in ways
other instructional methods do not. Feedback in the form of natural consequences of actions can
be powerful; the trick is to ensure that the feedback is driven by the learning objectives and
emphasizes the cause and effect relationship between learners’ actions and the consequences so
they understand what has happened and why. Consequences of actions in games often provide
the “wow” effect garnering the player’s attention but not always making clear exactly what the
player did to give rise to them. For example, one of the early feedback responses considered was
to have a Chief non-player character yell at the player’s character when certain key mistakes
were made. This kind of feedback was not necessarily appropriate from a chain of command
perspective and did not support the learning objective. However, from a gaming/story
perspective, it added some emotional stress and made the experience more exciting. Providing
the player with appropriate consequences resulting from his or her actions had to be carefully
considered as part of the entire gaming/instructional strategy. The experiential aspect of the
game environment opens possibilities that should continue to be explored while maintaining the
instructional focus. Decisions about which type of corrective feedback to provide immediately
within the game and which to provide in the post-game debrief remain important questions for research.

The lesson learned is that finding innovative ways to provide feedback that promotes reflection requires concerted effort, ongoing discussion and continued research. The game design must examine the intended outcomes and the paths people take that compromise those outcomes. We can then devise strategies to promote awareness and reflection, such as by setting up distractions to entice players into making common errors and providing feedback to help them refine their thinking.

**Developing an objective versus developing methods for assessing attainment of that objective**

A game story line must be developed so as to create situations that challenge learners with respect to the learning objectives and that provide opportunities for learners to react and receive feedback on their actions. Therefore, the objectives as well as the assessment strategy need to be developed first in the development of the game. In formal schooling, assessments often occur after a learning event (e.g., end of module test). As a team with such a short development cycle, it was easy to slip into this way of thinking. Also, team members had differing expectations about the role of assessment and even about the meaning associated with it.

Assessment is another area where gaming technology offers new possibilities for learning. Using both time to complete and path taken are possible options in addition to “correct” actions. One promising discussion focused on assessing intent and accuracy as one way to understand reasons for a player’s actions and to tailor future trigger events so as to hone skills appropriate to that player's needs. This kind of assessment also has implications for the type of feedback provided. Van Merrienboer and Kirschner (2007) for example make the following recommendations for assessing intent and accuracy:

- If the learner makes an error that conveys an incorrect goal, the feedback should explain why the action leads to an incorrect goal and should provide a hint or suggestion as how to reach the correct goal.
- If the learner makes an error that conveys a correct goal, the feedback should only provide a hint for the correct step or action to be taken and not simply give away the correct step

The lesson learned is that objectives and assessments must be linked at the outset of the development process.

**The right documents for sharing knowledge**

A key challenge throughout the effort was conveying our thoughts and ideas to each other in an effective manner. Most stakeholders in the effort tended to think and operate at a different level of abstraction or with a different focus. Further, different stakeholders entered with different preconceptions about the motivations of other stakeholders. Exacerbating this quintessential inter-disciplinary communication issue was the fact that our team was widely distributed geographically. Despite roughly monthly face-to-face meetings and frequent discussions by telephone, numerous misunderstandings would arise and persist.

The knowledge sharing issue was particularly revealing in the documents that we generated to share our instructional design ideas. Every single performer had a different
preferred format for capturing their ideas, even if the general type of document was similar. For instance, each of us had a different view about what a storyboard entailed. This confusion led to various persistent misconceptions. For instance, the game designers tended to conclude that the instructional designers were seeking to provide learning that was linear in nature due to the list or table formats they used to capture instructional events.

At several points during the effort, attempts were made to integrate the different perspectives and create a single format that met the needs of all performers. During knowledge acquisition, a table format merging the key CTA information and the scenario element analysis was created (see Figure 3). During development of instructional and assessment logic, a rich flowchart format was used to draw together inputs from several sources (see Figure 7). During the specification of the dialog, three different formats were explored (a dialog tree, a storyboard with optional paths, and a linear list with linked dialog fragments), but ultimately the actual dialog implemented in the game prevailed as our means of discussing and reviewing dialog content.

There is a strong need for effective collaboration mechanisms in a diverse game-based training development team. While we identified some collaborative document formats, further research and refinement is needed to find those communication means that are most effective for game-based training design.

**Hard-coded versus data-driven implementation**

In the field of game-based training, there is always a tension between implementing exactly what is needed for the specific instruction and implementing general game mechanics which can be used to support a variety of instruction. In many cases, the former approach is used, leading to a system that requires significant effort to adapt. Due to our broader project goals, we adopted the latter approach. In addition to a data-driven task mechanism, several data-driven feedback mechanisms were implemented.

The software development iterations during July and August were focused on implementing a data-driven infrastructure and basic support for the game mechanics we wanted in our initial prototype. By the end of August, a preliminary implementation of the tutorial and Mission 1 levels had been implemented. However, further revisions to the instructional content continued right up the end of September. These revisions were easy to incorporate into the system, and we made numerous revisions to the instructional content without slipping our software deadlines.

A key lesson learned was that the focus on a data-driven approach to specifying the instructional logic in the game greatly reduced the stress of making changes to the game and avoided the need for any significant re-factoring of the code.

**Designing for the novice while keeping the gamer happy**

Aside from age, Navy recruits are the very embodiment of diversity – 40,000 recruits each year who come from every conceivable background. It was essential that our training system be accessible to everyone from super users to computer neophytes and be effective with everyone from expert gamers to non-gamers. However, since our target population truly are novices in the domain of Navy skills, it was critical that the instruction be delivered at an appropriate level and not be overly challenging. In making our instructional design and game design decisions, there was an ongoing tension between designing for the novice and ensuring
that the game was usable and appealing to a very broad population. This tension revealed several lessons learned.

The interface of the game needed to be simple to accommodate the broad range of users. Further, a simple interface is a common trait of many good games. However, in our effort, being simple was often at odds with being comprehensive. The more instruction, data, feedback, interventions and details we added, the more complex the interface became and consequently, the less usable. In the end, we decided to postpone some instructional interventions in order to maintain a simple, clean interface.

Dynamic interaction is the most obvious quality of games. After all, if one can’t interact, a game reduces to a spectator experience. However, what types of interactions are appropriate for novice game players, while retaining appeal for the experienced gamer? In our game, the player assumed the role of a Seaman recently assigned to his or her first ship. By making the role in the game reflect the near future for all recruits, issues of experience in gaming were partially avoided (e.g., all the recruits are new to being aboard a Navy ship, so navigating around a ship effectively requires diligent attention to details in the environment). Further, we made actions in the game reflect real-world actions. Since these actions (e.g., navigating within a ship, repairing a leak using correct procedures, communicating properly with a superior officer) are not typical of a commercial game, both the inexperienced and experienced gamer still needed substantial learning in order to do it right.

Each mistake made by a player creates a brief teachable moment where corrective guidance can make a huge difference. However, the right level of guidance to give to a novice can conflict with the degree of freedom expert gamers expect. Our approach was to provide both non-interruptive and interruptive feedback of varying detail depending on the nature of the mistake made while keeping the player immersed in the story as much as possible. Minor errors would result in some guidance provided in a small pop-up window that was not intrusive. Conversely, when the student made a critical error, a demerit would interrupt gameplay visually and aurally with a specific message about the error. For both hints and demerits, the initial message would be somewhat general. If the error was repeated, subsequent messages would be more detailed.

These decisions were borne out by results of product testing across all a variety of users. For instance, a student's first demerit stops mere button-clicking behavior and leads quickly to increased attention by both gamers and non-gamers.

**Balancing revisions to instructional design with meeting software deadlines**

In addition to our data-driven approach, another key contributor to our success was the use of an agile software development methodology. A particularly relevant aspect of agile development is frequent deliveries. One of the key tenants of agile development is getting working iterations of the software into the hands of stakeholders as early and as often as possible. With six geographically distributed groups, frequent releases helped mitigate communication issues and maintain a cohesive vision for the product.

As in many software products, the requirements provided by the customer may change over time. In our case, there was an increasing emphasis placed upon the depth of training to deliver. Initial requirements were for a skill practice environment with limited guidance. Later requirements were for a training environment with limited assessment that could produce some validated learning outcomes of certain skills. The final requirements, determined several months
into the project, were for a training system with focused guidance, general assessment on all key skills, comprehensive feedback in response to errors, no negative training and no irrelevant elements that would produce validated learning outcomes across most skills taught. Our agile approach to software development, iterative approach to instructional design and frequent interactions with the customer enabled us to accommodate these changes in requirements.

Although key to our success, the agile method was not without issues. In fact, it caused some interesting tensions and challenges among the team. In order to produce frequent releases, software development moves extremely quickly. Yesterday’s ideas become tomorrow’s implementation. This put an extra burden on quick turn arounds and the rapid iteration of ideas and forced the team to focus on what’s important. Ideas that fell into the “could-be” or “might be nice” categories quickly got pushed aside by “must have” and “critical” requirements. Whenever an idea cropped up, it was immediately weighted for its instructional or gameplay value and then balanced against the rest of the to-do items. The most important items always got done and the rest got set aside for another day. The result was a game that was delivered on time and within budget, met all customer needs, and passed the usability tests with flying colors.

Future Research Directions

In the next two years of our project, our plan is to investigate further issues related to the development of game-based training and to build upon the lessons learned so far to identify design and development methods that support consistent and effective outcomes. We plan to create additional refinements of the flooding control training system, as well as to create additional training systems in other domains for the same customer. These development efforts will inform and be supported by efforts to create better authoring and assessment tools for game-based training (e.g., of which our data-driven infrastructure was an early step). To ensure that our advances are well-founded, we will conduct a variety of empirical investigations into the effectiveness of different gaming features for training, and the interaction between different gaming and instructional approaches.

The field of game-based training is at an interesting crossroads as it moves from a poorly understood cottage industry to a well-founded discipline of its own. The multifaceted nature of game-based training is both its strength and its weakness - no other medium can provide as rich and varied learning experiences, but few other instructional mediums are as difficult to get right. There is a need for increased communication across practitioners to share processes, mechanisms, and lessons learned. We believe our continuing efforts in this project will provide seminal contributions towards formalizing game-based development methodology.

Conclusions

Conducting research on instructional games begs the question, “How can games be made instructional and engaging?” A critical battleground for this debate is the learning objectives and how they are used during design and development of the game. Respecting the role of learning objectives was an easy point of agreement for the team in the abstract. However, it was difficult to reach a common understanding of what that meant in terms of game play, story, actions, dialog choices, feedback, guidance and the game environment. Instructional design based on learning objectives can limit distractions that overload the player with extraneous information that interferes with learning. However, the objectives alone do not guarantee focus on relevant information. Structuring the story must work in concert with treating the consequences of errors to help the player develop a mental model of what actions lead to the desired outcomes. The
interactions between the player and the game must maintain the player’s sense of immersion while also ensuring that the experience remains focused on the skills being taught. Finding the balance between adhering to the learning objectives while maintaining the characteristics associated with a game was a persistent challenge that revealed itself in a variety of ways.

A second key battleground for this debate concerns the processes followed in producing an appropriately instructional and fun game. In general, the agile approach adopted in this effort resulted in ongoing priority conflicts between the instructional, gaming and story mindsets. The instructional designers by nature preferred a more traditional waterfall approach in which all key decisions regarding learning goals, content selection, and specification of instructional methods and strategies would be made prior to involving considerations about the story, the technology and the gaming style. Their concern was that that key components, affordances, and experiences in the training system necessary from a pedagogical and assessment standpoint would be undermined or inadequately represented if instruction came second. The gaming developers by nature preferred clear direction on the basic interaction modes desired for instruction and the basic structure of a game level so that the associated software interfaces and graphics could be created in a timely manner, and were content to leave easy-to-change data details (e.g., the specific dialog and instructional messages, or sequencing of activities within a level) as part of an iterative refinement process led by the instructional designers. Finally, the story developers by nature preferred a strong early investment in coming to agreement on the basic story in order to ensure that successive iterative refinements of the instructional design or technology could be shaped to maintain a powerful story. From their vantage point, powerful stories create powerful experiences that invite participation while information and data, on the contrary, invite critique. The danger of ignoring the story aspect of a learning game at any stage of the process is that achieving a powerful story becomes difficult.

Of the many lessons learned in this effort, most involve the challenge of effective communication among professionals from diverse backgrounds. It is important to clearly articulate each team member’s priorities as early as possible and work together to devise methods for sharing and integrating information that will accommodate any competing priorities. From this, specific critical paths and components must be identified that, if organized and prioritized correctly, can inform the overall design and development approach. Since the process of combining the craft of game design and the science of learning has yet to be established, the lessons learned from this project may serve to inform the process and lead to an efficient and effective model for the development of games for learning.

Acknowledgments

The research reported in this paper was conducted under the management of Dr. Ray Perez under the Office of Naval Research contract number N00014-08-C-0030. We would like to thank our customer Dr. Rodney Chapman, Chief Learning Officer of the Naval Service Training Command, and his staff for their active participation in refining and guiding our product design, and Lt. Greg Page from the Recruit Training Command for his assistance in providing subject matter expertise. We would like to thank Dr. Ray Perez and Mr. Paul Chatelier of the Potomac Institute of Policy Studies for identifying opportunities for putting our research efforts in context of the Navy's current needs. We would like to thank Steve Greenlaw of Intelligent System Design for providing subject matter expertise. Finally, we would like to thank the software designers and engineers who contributed to the development and testing of the product: Chris Rodgers and Brad Anderegg from Alion Science and Technology, Rachel
Joyce and Julian Orrego of UCF, and Erin Panttaja, Todd Wright and John Ostwald of BBN Technologies.

References


