Developing Capability Requirements for Training Systems

Byron R. Harder
U.S. Marine Corps Training and Education Command
Quantico, VA
byron.harder@usmc.mil

ABSTRACT

Training system requirements managers have a unique challenge in the form of documenting the qualitative and quantitative requirements of hardware and software systems whose principal purpose is the support of training outcomes for the people that use them. In the United States Department of Defense and similar organizations, capability requirements policies and processes, i.e. the Joint Capabilities Integration and Development System (JCIDS), are oriented on weapon systems. By default, training systems are considered ancillary components of the associated weapon systems. For various reasons, though, some training systems must be developed as independent capabilities rather than footnotes to larger systems. JDICS-type documents consist of two fundamental components: capability requirements, which describe "what" a system must be able to do, and performance attributes, which prescribe threshold and objective (hard minimum and reasonable maximum) measures of "how well" the capability requirements should be met. It turns out that determining performance attributes for training systems can be extremely challenging; typical weapon system attributes such as speed, range, engagement distance, weight, and survivability are essentially meaningless for training systems. The core attribute of a training system is the extent to which it enables training against a set of standards, but those standards can—and must—change as the operating environment evolves. Processes such as JCIDS are not agile enough to respond to changes to the training standards. This paper offers some approaches to determining quantifiable performance attributes for training systems in a JCIDS-type context. It also describes a methodology for adapting requirements at a pace that maintains relevance to the operating environment. Although JCIDS is the main focus, consideration is given to related processes such as Front End Analysis and Training Effectiveness Evaluation. This paper provides tools that will help training system capability developers succeed, even within processes designed for weapon systems.

ABOUT THE AUTHOR

LtCol Byron R. Harder, Ph.D. is the Synthetic Training Integration and Management Branch Head at Range and Training Programs Division, Marine Corps Training and Education Command in Quantico, VA. He is an active duty modeling and simulation (M&S) officer with a Ph.D. in Modeling, Virtual Environments, and Simulations. His background includes 21 years of military experience, including artillery, acquisition, requirements, and resource sponsorship positions. His current duties are focused on the modernization of the Marine Corps Training Environment and the requirements and resource sponsorship of ground training simulations.

Developing Capability Requirements for Training Systems

Byron R. Harder
U.S. Marine Corps Training and Education Command
Quantico, VA
byron.harder@usmc.mil

INTRODUCTION

In the United States Department of Defense (DOD) and similar organizations, policies and processes governing capability requirements, i.e. the Joint Capabilities Integration and Development System (JCIDS; DOD, 2018), are oriented on weapon systems. By default, training systems are considered ancillary components of the associated weapon systems. For various reasons, though, some training systems must be developed as independent capabilities rather than enabling components of larger systems. Included in this category are the United States Army's *Non-system Training Aids, Devices, Simulators, and Simulations* and the United States Marine Corps *Non-standard Training Systems.* There are several reasons that training systems might be categorized and developed in this way. In some cases, such as constructive simulations that model a wide variety of capabilities and stimulate a number of different operational systems, there is no single weapon system to which the training system can be logically attached. Some such programs are established to provide a collective or distributed training capability, which tends to be left out of system-specific trainers. Economies are possible in some cases where a single training system architecture can support multiple devices, e.g. marksmanship trainers that support multiple small arms weapons. In some unfortunate cases, training systems are deleted from program plans in response to budget cuts (the common refrain is, "We'll just do live training"), only to be identified later as critical shortfalls that must be developed under separate cover—typically through different funding lines and stand-alone requirements documentation.

The capability development document (CDD) as defined by the JDICS Manual (DOD, 2018), and similar types of requirements documents, use two fundamental constructs: capability requirements, which describe *what* a system must be able to do, and performance attributes, which prescribe threshold and objective (hard minimum and reasonable maximum) measures of *how well* capability requirements should be met. Even for training systems that do not use the JCIDS process or the CDD template, defining both capability requirements and performance attributes is still a good practice because it provides trade space to the solution developer to create a competitive edge without sacrificing must-have features. It turns out, though, that determining performance attributes for training systems can be particularly challenging. Typical weapon system attributes such as speed, range, engagement distance, weight, and survivability are of little use to training systems, which simply model the corresponding attributes of represented systems. The main contribution of this paper is to present some practical approaches and examples of useful performance attributes that training system requirements managers (RMs) can consider for their own requirements documentation.

The core attribute of any training system is the extent to which it enables training against a set of standards, but those standards can—and must—change as the operating environment evolves. Requirements vetting and approval processes such as JCIDS are often not agile enough to respond to changes to the training standards. To maintain relevance, training system requirements sponsors require a methodology for adapting requirements at a pace that maintains relevance to the operating environment. Offered below is an approach to solving this problem in a manner consistent with the performance attribute recommendations just mentioned.

This work begins by reviewing some background to the discussion: defining training system stakeholders, describing the basic principles of capability requirements and performance attributes, and surveying relevant literature and policies. Next, specific recommendations are presented for performance attributes that can be used for training systems. Following this, some ideas for organizing requirements that enable constant evolution against the moving target of training needs are discussed. After summarizing the recommendations, some remaining challenges in this space are mentioned.

BACKGROUND

Stakeholders

To develop effective training system requirements, the RM must understand the types of stakeholders involved. Each has important information to provide to the requirements elicitation process, and each holds a different type of leverage over the training program. The following are generalized descriptions of stakeholders found in different armed services.

Trainees are, of course, the primary focus of all other stakeholders. If the system cannot meet their training needs, then it has no value. The RM must understand the abilities of trainees (e.g. education and training background, level of physical conditioning) before the training system is used and the desired level of proficiency after training is complete. Trainees may have preconceived notions about the utility of certain types of training systems based on experience or culture, which may bias their input or their reaction to the final product.

Training Agencies are the groups responsible for training the trainees. In many cases, they can provide more specific information about required capabilities than the trainees because they have shepherded many different trainees through the learning process. They may have experience with an older version of the training system, which may color their perception of future capabilities. In some cases, such as deployable military trainers, the trainees *are* the training agency—but there still tend to be some individuals in any organization that have a deeper knowledge of how to train and how to use training systems.

Training Standards Organizations are responsible for documenting the tasks, conditions, and required performance levels (all of which will be abbreviated to *standards* below) that must be achieved by the trainees. All training system requirements should be traceable to these standards, and the standards should annotate which systems—in specific configurations—can or must be used to support training.

Resource Sponsors are those that decide how much funding will be allocated for the training system program. There may be one or more resource sponsors for a given program. Since this essentially gives veto power, it is essential to maintain the resource sponsors' confidence that system requirements are well-managed, and that the resulting system will be affordable.

The *Program Manager (PM)* is ultimately responsible for developing, testing (i.e. verifying and validating), and fielding a training system that meets the requirements provided by the RM. For DOD training systems, the PM is part of the acquisition workforce. This stakeholder has the authority to make technical decisions within the trade space provided by the performance attributes, but does not have the authority to change the language of the requirements document. The PM's team usually includes systems engineers and instructional systems specialists (among many other disciplines). The PM is evaluated based on the program's optimization of the trade space, affordability, and accreditation decisions (see below).

The *Technical Workforce* is the team that actually produces the training system. For most DOD training systems, this is a contractor. For many organizations, there may be no need to distinguish between the PM and the Technical Workforce. The Technical Workforce's stake is similar to the PM's, but in the case of a contracted effort, usually includes an additional profit motive. The PM translates training system requirements into contractual specifications for the Technical Workforce in order to optimize the coverage of the trade space.

The *Accreditation Authority* uses the results of validation testing to formally determine which training standards the training system may be used to train. In some cases, the Accreditation Authority role may be held by the Resource Sponsor or the Training Standards Organization. If unstated, the role effectively falls to the PM.

Some of these stakeholders clearly map to those one would identify for a weapon system program—Resource Sponsors and the PM, for example. The unique roles for training system programs are Training Agencies, Training Standards Organizations, and the Accreditation Authority (Trainees are to training systems as Users are to weapons systems). All of these training system stakeholders are (or should be) present in a weapon systems program since all weapons systems require training, but it seems that the focus on the weapon system's trade space often overshadows tradeoff analysis and decisions for their training systems.

Requirements Development in General

The Requirements Development process goes by several different names, including *requirements analysis* and *requirements elicitation.* One can find educational material and standards for Requirements Development under the systems engineering and software engineering disciplines; Laplante (2018) is an excellent reference. However, in the DoD, RMs for JCIDS-type requirements are usually not systems engineers or employees of acquisition commands. Rather, they tend to be subject matter experts in the mission area supported by the new capability with specific knowledge of Requirements Development, to include the particulars of the JCIDS or equivalent business process. Requirements Engineering, as a systems engineer would define it, goes farther and deeper than the Requirements Development discussed in this paper. For example, Requirements Engineering develops specified and implied requirements into a functional baseline and allocated baseline, and includes direct interaction with the Technical Workforce (usually through contractual arrangements) to develop a product baseline.

In the U.S. DOD, an organizational and legal separation exists between the RM and the PM. To simplify the matter for academic purposes, let us say that the RM does not have the authority to make decisions about what technical solutions to use, which vendors to employ, or exactly how much funding to expend on any particular component. Those decisions are the PM's to make. This is why threshold and objective performance attributes are written into the requirements document—otherwise, all attributes are essentially thresholds and the PM is forced into a "lowest cost technically acceptable" contracting strategy. The objective level of a performance attribute tells the PM when to stop expending more resources and instead focus on improving other attributes. Ultimately, the product baseline for a given increment or release fixes the level of each attribute, hopefully somewhere between the threshold and objective. In commercial applications or other situations where the RM/PM separation of duties is less strict, the RM may be directly involved with fixing the attribute levels (e.g. by writing them into a contract) as the project moves forward. Of course, the Resource Sponsor could scrub an unfavorable product baseline by removing funding, but doing so would provide no alternative.

The CDD template (DOD, 2018) includes 13 sections and 4 appendices, covering information such as the concept of operations, threats, and affordability. The present work is only focused on the Capability Discussion and Performance Attributes (sections three and five of the CDD template). For reference, the information in these sections is developed according to the sequence shown in Figure 1. A Capability Gap Assessment produces a list of capability requirements and associated gaps. Some of these are carried forward into an Analysis of Alternatives, which includes—among its many outputs—information useful for determining performance attributes. Each performance attribute is mapped to one or more capability requirements, meaning that it should help measure the achievement of that requirement. Performance attributes should be "technically achievable, quantifiable, measurable, testable, unambiguous, supported by documented trade-off analysis, and defined in a manner that supports efficient and effective [testing and evaluation]" (DOD, 2018).

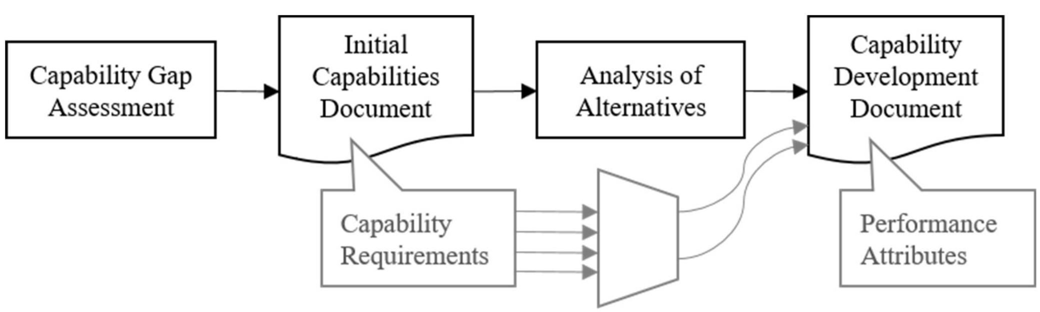

Figure 1. Simplified Capability Development Process

A capability requirement is written as "the ability to [perform a task] against/given a [threat] in order to achieve [effect] in a/under [environmental conditions] in [standard timeframe]" (DOD, 2018). Although at first glance this structure assumes a warfighting context, note that the "task" is drawn from the Universal Joint Task List or Service (Army, Navy, etc.) Tasks, which include training and force preparation categories. "Threats" could be excluded, since training systems are not meant to be employed on the battlefield; alternatively, they could include the pervasive threats of the information environment (i.e. cybersecurity) or describe the modeled threats that trainees must compete with to achieve effective training. “Effects” can be described as an increase or sustainment of unit readiness or individual skill proficiency. In some cases, it is helpful to include more specific effects such as presenting trainees with a certain type of challenge, replicating aspects of an operational system, or providing training results data. “Environmental Conditions” can be used to specify the trainees’ situation or location, e.g. in a simulation facility, in the barracks, on ranges or training areas, embarked aboard ship, or deployed to a forward operating base. The RM could also choose to use this part of the capability requirement to describe the operational conditions (e.g. visibility limitations, temperature, and other conditions whose effects on the response of the modeled platform could be simulated) that the training system must replicate. One approach to this is to simply reference the environmental conditions of the modeled operational capabilities. Finally, “standard timeframe” can refer to the duration of a mission, a training vignette (which might be shorter than an actual mission), a training session (which could be longer than a mission if multiple repetitions are appropriate), or even the overall timeframe of system usage during a deployment workup or training course. The preceding information can be difficult to cram into a single sentence and would likely be repetitive across multiple capability requirements, so it is advisable to factor out the common information into other parts of the requirements document. The fundamental core of the capability requirement statement is “the ability to [generate an effect],” which is an inherently binary statement—either the capability can generate the effect, or it cannot. The Capability Discussion should focus on a description of the desired effects without specifying any particular solutions or approaches.

Performance attributes are all about measuring the achievement of capability requirements. The definition of performance requirements (section five of the CDD) should *not* spend any time describing *what* the capability must be able to do—any such language should be moved to the Capability Discussion. Instead, this section should only describe *how* aspects of the capability will be measured and what the useful upper and acceptable lower (objective and threshold) measures are. Performance attributes are not conjured out of thin air; they typically require some sort of study or analysis to ensure that the thresholds are not too low and the objectives are both achievable and useful. There is an important relationship between performance attributes and the state of science and technology (S&T), which is measured in technology readiness levels (U.S. Government Accountability Office, 2016): thresholds must be “achievable through the current state of technology” (DOD, 2018) and objectives should at least have a chance of being achievable within a decade (author’s recommendation). Since program affordability is only based on threshold levels (DOD, 2018), there is wide latitude for specifying objective values. However, RMs are cautioned that unachievable objectives will be ignored. Setting a science fiction-level objective value is essentially abdicating the objective level specification to the PM.

In a CDD, performance attributes are categorized as key performance parameters (KPPs), key system attributes (KSAs), and additional performance attributes (APAs). The important distinction is that KPPs are “program killers,” meaning that failure to achieve their threshold levels should result in a cancellation or reset of the acquisition program. The distinction between KSAs and APAs is minimal; the former are higher priority than the latter, but the PM is still able to make tradeoffs between them freely.

Developing Training Requirements

Military documentation includes process guides specific to the development of instructional systems. Most of these are versions of Instructional Systems Development (ISD) or Systems Approach to Training (SAT; DOD, 2001). This process is also known by the acronym ADDIE for its component steps: Analysis, Design, Development, Implementation, and Evaluation. The important distinction here is that an *instructional system* (as coined in ISD/SAT documentation) is greater in scope than a *training system,* and therefore greater in scope than the focus of the present work. An instructional system includes a variety of learning modalities, such as classroom lectures, reading assignments, demonstrations, discussions, and performance—for which one option is the execution of tailored scenarios in training systems. Due to this broad scope, ISD/SAT documentation and research tends not to venture down to the level of detail required to develop specific training system performance attributes. For context, though, it is important to understand that Requirements Development (for training systems) mainly occurs during the Analysis and Design phases of the ISD/SAT process.

Most of today’s training systems employ software. The trend towards software-centric designs has made training systems become more configurable and general purpose. It is increasingly the case that a single training system can be employed in support of *several* different instructional systems. Therefore, the training system requirements development process cannot be embedded entirely within a single ISD/SAT process. The RM for a training system may be aware of a few in-development or already existing instructional systems that must be supported, but ideally, modern training systems are robust enough to support instructional systems that have not even been dreamt up yet. One example of such a training system is the Marine Corps Deployable Virtual Training Environment (DOD, 2019), which is essentially a suite of networkable gaming laptops installed with applications such as a semi-immersive infantry combat simulation. It can be used to support a variety of different instructional systems, including programs of instruction for individual skills (e.g. land navigation), small unit infantry tactics, and fire support observation.

Literature Review

The research literature on this subject area includes recommendations for training requirements development processes, fidelity tradeoffs, and interoperability. Vaughan (1979) describes a process of evaluating training system choices early in the weapon system development process. Heffley et al (1982) investigate the use of human operator control theory to determine quantitative flight simulator fidelity requirements and identify data gaps limiting the optimal employment of their technique. An approach to developing initial training system requirements based on architecture views and design layer theory is recommended by Duke and Shook (2010), who also identify the need for systems engineers and other specialties on the Training Integrated Product Team. McCallister and Gray (1999) call out shortfalls in training system requirements—to include typically cursory treatment in the corresponding operational requirements document (the predecessor to the CDD)—and they recommend a robust requirements engineering approach to improve results.

Fidelity (or *resolution*) is a common topic for training system requirements discussions. It generally refers to the similarity of a training system to the real-world system and environment that it is meant to replicate. The term *fidelity differential* (Fullmer, Woodard, and Matusof, 1990) has been used in attempts to describe how far off the training system is from the real-world ideal. Estock et al (2006) describe a process that maps simulator fidelity to training needs; this paper also includes a useful summary of research on different categories of fidelity. Schreiber, Bennett, and Gehr (2006) provide experimental results from two different aviation training systems with different levels of fidelity, using a 0-5 human evaluator rating scale (from "Capability to experience does not exist" to "very good"). Hall and Janisz (2006) discuss tradeoff analysis for different aspects of a ground combat shot adjudication system. An important takeaway from the literature is that empirical research focusing on *tradeoffs* can provide useful suggestions for the RM, since metrics used for tradeoff analyses can be considered as potential performance attributes.

Interoperability requirements can be difficult to place on a quantitative scale for training systems. Even if a known set of digital message types can be received and transmitted without error, the interpretation of those messages by the receiving systems can result in very different training outcomes. An early idea in the development of networked training systems was that trainee workload in the training environment should be comparable to the that of the real-world environment (Knight, 1990)—so interoperability efforts not leading to useful change in the relevant workload are wasted. On the other hand, traditional simulation interoperability standards can result in the "least common demonimator" problem: higher-fidelity systems tend to experience a degradation in fidelity during distributed training due to the inability of lower-fidelity systems to provide sufficiently detailed information. An example of this phenomenon is an aircraft track generated from a battle staff trainer not providing a sufficiently realistic flight path for an observer in a connected flight simulator. Solutions to this challenge are still being pursued today (Office of Naval Research, 2017).

If interoperability requirements are written only at the individual training system level, the interconnected system of systems that eventually emerges is unlikely to meet unspecified higher-level mission requirements. Guptill and Cipparone (2001) present a multi-level, nested training requirements development approach that links individual tasks all the way up to the Joint Mission Essential Task List, an approach that is somewhat reflected in the current ISD/SAT Military Handbook (DOD, 2001). Even if this type of process is followed, though, the Defense Acquisition System is not structured hierarchically. A system-of-systems construct exists in JCIDS, wherein multiple CDDs can be written against "parent" capability requirements in a common ICD, but the acquisition programs that they spawn are each managed as independent entities. Each responsible project officer is measured against the requirements of the individual system and in control of a separate budget, so efforts towards closing interoperability gaps across multiple systems may not be consistently prioritized across program boundaries.

DEVELOPING PERFORMANCE ATTRIBUTES

Although policies and processes are well-established for the development of requirements (in general) and the design of instructional systems, and despite a significant volume of research in this general area over the last several decades, RMs for training systems do not have a practical list of candidate performance attributes to draw from when developing JCIDS-type requirements. The remainder of this paper attempts to provide the first cut of such a list. The following attributes will not be appropriate for every system, and will likely be insufficient for any particular system, but the intent is to provide a starting point so that training system RMs are not starting every effort from whole cloth. Most of these performance attributes have either been used in the requirements documentation for one or more training system employed by the U.S. Marine Corps or found in tradeoff analyses for proposed training solutions.

Training Standards Accredited

The first candidate performance attribute in the list is the most important. It is difficult to imagine a training system that would not benefit from including this attribute because it directly measures the ability of the system to support training. Let S be the set of training standards that the system is mandated to support (i.e. the objective of developing the system is to provide a training medium for all elements of S). Let $A \subseteq S$ be the set of training standards accredited by the Accreditation Authority. The *training standards accredited* performance attribute is then defined as $|A|/|S|$, the fraction of required standards accredited. This should be a KPP—and, one might argue, the only KPP—since the entire point of a training system is to enable standards-based training. The threshold should be a fraction below which the return on investment would not justify the continued pursuit of the material solution. Although JCIDS argues against setting objectives at 100%, it seems justifiable in this case. As was argued above, every member of S selected by the RM should be achievable. If some standards have higher priority (e.g. safety-related standards that will have no alternative means of training), then the threshold may need to be stated with a bit more complexity. For example, one might list specific standards that absolutely must be supported, and then provide a target score for $|A|/|S|$.

If the organization allows some notion of *partial* accreditation—for example, there might be some conditions that the training system is not able to correctly replicate—then this will need to be accounted for in the calculation. The simplest approach would be to score a partially accredited standard as one-half of a standard in the count of $|A|$.

It is the author's experience that not all desired standards are fully supported by most training systems. Some standards include performance steps that simply do not fit a reasonable technology approach for the system; others might require resources (e.g. procurement of certain functional equipment replicas) that will not be available for years down the road. The malleability of training standards will be addressed below; this fact of life can cause S and A to change faster than the program can reasonably adapt. With this in mind, the RM should allow considerable trade space between the threshold and objective. For this attribute above all others, the PM should doggedly pursue the objective with each system update.

Quantified Proficiency Improvement

The next candidate performance attribute is rarely, if ever, actually used in requirements documentation. It is an alternate approach to measuring a system's ability to support training, using direct results from a training effectiveness evaluation rather than accreditation. *Quantified proficiency improvement* depends, as the name suggests, on a quantified measurement of trainees' proficiency against the associated training standards. A good example of such a metric is a marksmanship score, where increasing points are awarded for shots that land closer to the bullseye. For tasks that do not lend themselves to such clearly defined metrics, it is possible to quantify the qualitative assessments of subject matter experts (Loeffelman, 2019). The idea of this performance attribute is that trainees are tested before and after using the training system, ideally in real-world conditions with the actual equipment (that is, not the training system).

Let d_T and d_O (both greater than 0) be the threshold and objective quantified proficiency improvement differences that will be listed in the CDD. By conducting a pre- and post-test with a population of trainees, one can find the mean quantified proficiency scores $\bar{y}_{\text{pre}}$ and $\bar{y}_{\text{post}}$. Consider two alternative hypotheses, $H_1\colon y_{\text{post}} - y_{\text{pre}} < d_T$ and $H_2\colon y_{\text{post}} - y_{\text{pre}} \geq d_O$. If the null hypothesis is rejected, then either the system *failed* to meet the threshold ($\bar{y}_{\text{post}} - \bar{y}_{\text{pre}} < d_T$) or it *succeeded* in meeting the objective ($\bar{y}_{\text{post}} - \bar{y}_{\text{pre}} \geq d_O$). Otherwise, the threshold is assumed to have been met. This may seem to be a low bar, but adopting this approach would be a significant step up in accountability from the current state!

This potential performance attribute may not be appropriate for trainees that are already well-trained. If sustainment training is the purpose of the system, then it may be very difficult to detect any improvement in this metric. This does not necessarily mean that the training system has no value.

Although this performance attribute is not popular, RMs should consider it. Of all the attributes discussed here, it would be the most scientifically defensible in the context of military readiness. It would also provide a strong basis for return on investment analysis within a portfolio of learning resources. If the Experience Application Programming Interface (xAPI) is in play, then the Results field of the xAPI Statement (DOD, 2020) can be leveraged to capture and recall data to feed the quantified proficiency improvement calculation. If this performance attribute is perceived as too risky or difficult by the stakeholders, then it should be considered as a KSA or APA.

Operator Efficiency

The motivation for the *operator efficiency* attribute is measurement of the level of automation of some functions of a training system. Rather than attempt to measure this directly with something like the number of operator interventions, which can be difficult to determine *a priori* and awkward to state succinctly (there could be many different functions with different numbers of interventions), the suggestion here is to state the number of human operators required to perform a key high-level task. It may be necessary to also place a time limit on task completion—otherwise, it could be argued that just one operator can perform all tasks given unlimited time. Alternatively, if there is a way to measure the scale of the task, then the attribute could be stated as threshold and objective values of the scaling metric given a fixed number of human operators. An example of this type of alternate metric is the size of simulated units (e.g. squad, platoon, company) that can be controlled by a single human operator. This attribute may not be particularly useful when considered in isolation, but combined with other performance attribute it can help drive the PM to invest in automation features. Synergy also exists with cost attributes (not discussed further, as they are standard CDD features), since recurring person-hours of the operational staff can be reduced through a one-time investment in automation.

Time to Prepare

Similar to operator efficiency is *time to prepare.* The point of calling this out separately is that preparation time—that is, work performed by the training agency before the trainees start using the system—can be significantly greater than execution time. Preparation may refer to pre-execution or pre-assessment efforts. For this performance attribute to be meaningful, the number of operators and the complexity of the training event must be fixed. The latter could be accomplished by specifying the number of simulated units, number of tasks to be trained, number of trainee decision points, etc. This will depend on the type of training system. If fixing the complexity of the task is too limiting then it may be possible to specify the threshold and objective using a formula such as $t(x) = Ax + B$, where *x* is the number of tasks (or decision points, etc.), A, B are in units of time, B is the baseline amount of preparation time (regardless of complexity), and $t(x)$ is the parameterized time to prepare. The RM chooses threshold values $A = A_T, B = B_T$ such that all values of $t(x)$ are acceptable for the stated domain of *x*. Objective values A_O, B_O should satisfy $A_O \leq A_T, B_O \leq B_T$, and $A_O x + B_O < A_T x + B_T$ for all *x*. The values of A and B for the as-built system could be experimentally determined using regression.

Number of Scenarios Provided

Most training systems must be configured with some type of scenario in order to support training. Although it could be left to the training agency to create these scenarios, it would be impossible to conduct validation testing on the system before the scenarios exist (and it would be irresponsible of the PM to deliver a system without having conducted developmental testing with representative scenarios). The PM's team is likely equipped with instructional systems specialists that have the expertise to ensure the development of effective scenarios. Since a system that includes more scenarios "out of the box" can support a wider variety of training upon delivery, it may be useful to incentivize the development of more scenarios by making this a performance attribute. It is quite possible to deliver inadequate scenarios, so the performance attribute description should specify that the count will exclude any that fail a training effectiveness evaluation. It should also state that the set of scenarios delivered, in combination, must cover all of the tasks and conditions that the system is required to support.

Data Volume

Capturing the results of training is an important objective. Training data can support tailoring of future training to individual needs, identification of trends across the organization, and investment decisions affecting the system's continued operation and potential modifications. Thus, inclusion of a performance attribute that measures the volume of data collected may be helpful to ensure that a sufficient amount of data can be collected. If possible, specify the data volume in operationally relevant terms, e.g. minutes of voice traffic, number of position location data points, or number of xAPI Statements rather than raw megabytes of data.

Lists of Features

Generally speaking, it is much easier to determine capability requirements—that is, features that the training system must have—than it is to determine threshold and objective performance attributes to measure those requirements. If the RM is able to categorize some of a system's requirements into "must-have" and "nice-to-have" features, then these categories can be used as the basis of performance requirements. Simply list the "must-have" features as the threshold, and state that the "nice-to-have" features, in addition to the threshold, are the objective. An example of this would be a marksmanship trainer that displays the simulated impact point on the target based on weapon orientation at the time of shot (threshold), and displays the aimpoint trace from five seconds prior to one second after the time of shot (objective). (Note that a data volume approach could also be used in this example.) Unique or new features that have never been demonstrated in a fielded product should be considered for the objective category.

Fidelity

It is easy to describe fidelity differentials and the negative training that could result from inadequate modeling. Describing threshold and objective levels of fidelity, though, is difficult at the aggregate system level unless the system is a very specific part-task trainer. RMs are cautioned against language such as, "must be identical to the real-world system in every way," because the only way to truly meet such a requirement is to deliver the real-world system. Similarly, requiring perfect replication of real-world conditions is not possible; the act of modeling is essentially the choice of which aspects of the real world to abstract away based on the intended use of the model (i.e. George Box's familiar assertion that "all models are wrong"). Rather than delve further into the epistemology of simulation here (interested readers should consult Tolk et al, 2013), let it be enough to say that performance attributes should be dealt with below the system level. A few examples are provided in the remainder of this section, but due to space limitations this is only a representative sample.

Field of view is relevant for many different types of training systems, including air, ground, and waterborne vehicles as well as dismounted (i.e. infantry) trainers. Field of view can be stated in units of angular measurement such as degrees and should consider both horizontal and vertical axes. With recent interest in head mounted displays, it may be necessary to state the *field of regard* rather than, or in addition to, the field of view. The field of regard is the range of viewing angles providing realistic imagery assuming no restrictions to the trainee's head or eye movement. (The term is taken from a modeling attribute in the COMBAT XXI constructive simulation system; see https://www.trac.army.mil/COMBATXXI.pdf). It may be the case that only discrete ranges of measures provide additional value. For example, a joint terminal attack controller trainer may have an objective requirement for trainees to be able to look for aircraft behind them; increasing the horizontal field of regard from a 90° to a 120° arc does nothing to satisfy this requirement and may not be worth the additional cost. (In this example, the *list of features* approach presented above may be a better approach to meet the intent.)

Visual refresh rate is a relatively easy measure to take on a digital training system, since most modern visual systems already report this metric. It is advisable to focus on the worst-case scenario—meaning the heaviest processor workload—because even the world's best graphical processing units can be held back by an overloaded central processing unit. Visual refresh rate is an important contributing factor to the chance of simulator sickness, so the RM should consult with visual system specialists to determine the most appropriate values for threshold and objective. Values in the range 60-120 Hz are likely for real-time applications; more aggregate (i.e. battle staff) trainers could get by with lower values.

Trajectory variance and *shot distribution* are two ways of measuring the fidelity of projectile weapon models. Both depend on the existence of a reference model, since measuring physical projectile paths is likely beyond the scope of any training system program. One might be tempted to simply require that the known trajectory model be encoded in the system software, but such an approach is not always appropriate. If the training system only needs to provide aggregate information—for example, a battle staff trainer—then the expense of modeling the entire trajectory is likely wasted. If the terrain and entity models are not at a commensurate level of resolution to the weapon trajectory, then the impact points will not be accurate anyway. For some material solutions, such as commercially-available products, the trajectory or impact point model may not be accessible or modifiable. In these cases, it would be helpful to include a performance attribute that sets testable levels of variance. Since real-world projectile paths are stochastic, it would not make sense to incentivize a deterministic trajectory. The performance attribute should instead state the parameters of the distribution, with the notion that a χ^2 test (made possible by sampling the projectile location) will be used to determine the representational correctness of the model. It is important to note here that levels of precision can be counter-intuitive in a training context: if a simulated shot distribution is tighter than it would be in the real world, then the trainee's scores could *drop* when transitioning from the training system to the live range. Another potential consideration is that guided weapons follow much more complex paths that depend on parameters that vary after launch, such as movement of the target and control inputs from the trainee. Also, the accuracy of certain weapon flyouts may affect the training system's security classification.

Interoperability

Specifying levels of interoperability presents similar challenges to those discussed in the previous section. How can we be precise about requiring a system to be "more interoperable?" There are many different ways for interoperability to go wrong (Tolk & Muguira, 2003), so trying to specify attributes to cover the gamut will likely result in a requirements document that exceeds its page limit. Part of the solution is to leave technical standards to the accompanying architecture package (e.g. DOD Architectural Framework System Views and Standards Views for DOD systems), but certain performance attributes can help bring interoperability to the forefront of the PM's planning.

For modeling and simulation-based systems, correct visual representation depends not only on the correct interpretation of received messages; the appropriate visual models must be present in the runtime. If an externally published list of entity types can be referenced, then it is possible to establish a performance attribute that sets the percentage of models that exist in the system relative to the standard. This attribute could be titled *visual model completeness.* It is not advisable to use extremely broad standards (e.g. Simulation Interoperability Standards Organization, 2019), because they are too massive to assist in prioritizing investments. The RM might as well write, "model everything." A better approach is to develop a shorter list of models that are most valuable to the distributed teams that need to train together. Note that this performance attribute interacts with the fidelity category, since models with insufficient resolution for the intended use should not count on the positive side of this metric.

Federations of training systems often encounter problems with *terrain correlation*, which manifest when two or more trainees view the same logical piece of ground through the lens of different training systems. If the terrain data is captured at different levels of resolution or with different geometrical modeling approaches (either in the source database or the runtime memory), includes different types of attributes (e.g. one database might not have a field for "chain link fence"), or was collected at different times, then differences in trainee experiences can distract from learning. For example, one trainee may attempt to describe a target area to another using reference features that only exist in one of the two systems. So-called "fair fight" issues may also result when cover (from projectiles) and concealment (from observation) are not consistent across systems. One way to describe the similarity of terrain between systems is to focus on terrain elevation. A performance attribute could state that, when the same reference point (e.g. latitude and longitude) is tested in the new system and a reference system, the difference in elevation cannot be perceived by a typical human observer some percentage of the time. The definition of *visually perceptible* should depend on the training application, since (for example) infantry and fighter jet pilots tend to view objects at different distances. The details of how to determine this could be left as an implied task for the PM. Note that the term *typical human observer* does not necessarily require a human tester if the stakeholders agree on what that allowable distance is; if so, then an automated testing process can be used. Of course, elevation is only one of many terrain-related attributes that could reflect correlation issues, but most requirements do not even go this far.

Interoperability requirements present another counterintuitive issue: very precise individual training system performance attributes may *increase the risk* of interoperability problems. If the PM is forced to develop customized solutions or boxed into a boutique commercial product, the ability to reuse components (image generators, terrain databases, entity models, adjudication algorithms, etc.) across systems decreases. The RM must balance interoperability—that is, distributed training quality—against "stand-alone" training requirements. Performance attribute prioritization can play an important role here. It is worth mentioning architecture again, since requiring industry standards such as xAPI (DOD, 2020) or the Open Geospatial Consortium's Common Data Base (2020) can prevent costly proprietary integrations.

ADAPTATION TO EVOLVING TRAINING NEEDS

The lifecycle of typical defense systems, to include training systems, is measured in decades. Just getting a requirement from need identification all the way through fielding, testing, and the establishment of in-service support can take five years or longer. In comparison, training standards are updated every one to three years. If the training system requirements are based on the training standards, how can the system ever meet the mark?

This issue bears its own dedicated research, but a few recommendations can be made here. First, it is advisable not to list specific task codes from the performance standards in a requirements document such as a CDD. These documents take a long time to reach approval and can be difficult to change. Instead, the requirements document should reference the training standards documentation. These should, in turn, list which training systems currently support the standard; they should also identify standards that are in need of a training system but currently lack one. The PM might argue that this is shaky ground because the standards could change before an engineering solution is even ready to begin verification testing—but any other approach is blind to the rate of change of our modern world. Even before a system is fielded, the configuration change process should be integrated with the training standards development process. This will provide the PM the earliest possible warning that change is afoot. The RM must also be directly involved to ensure that the training, requirements, and acquisition communities are interpreting the requirements the same way in the context of changing standards. Additionally, PMs should employ agile system development practices; requirements that provide trade space enable the PM to break out of the waterfall model.

With the exception of purely hardware-based systems, the notion of *full operational capability (FOC)* is inappropriate for any modern training system. After FOC, a traditional system transitions into "sustainment mode" and is not expected to expend further research or development funds. This concept is not compatible with the need for adaptability just discussed. Most training systems should be treated with a software-centric acquisition model, with the assumption that development is never complete and there is always another release coming soon. RMs must lead-turn these efforts and provide updated performance attribute objectives as science and technology evolves. The RM's goal is to prevent the PM from ever reaching all of the objectives, enabling continual improvement.

CONCLUSION AND FUTURE WORK

The complexity of training systems has steadily increased over the years. If identifying appropriate trade space for non-training systems is important, and operational effectiveness depends on high-quality training, then surely it is also important to identify trade space for training systems. Although plenty of resources exist to guide instructional system specialists through the process of determining *what* a training system must be able to do, there are few examples to be found of actual performance attributes that RMs can consider when developing trade space for a training system. Hopefully some of the most important attributes were captured here, but surely this paper only presents the beginning of a complete list.

Although challenges related to adaptability have been discussed here, only a few approaches have been presented to better deal with them. Future work should expand the dialog towards methods and tools that better allow the training, requirements, and acquisition communities to adapt training systems to the operational environment. It appears that providing "acquisition maneuverability" within a reasonable trade space is a useful tool in such a context, but more work is needed in this area.

The fundamental intent of this work is to create a dialog about trade space among the training systems RM community. In particular, quantified proficiency improvement could benefit from focused studies and implementation efforts, with the potential reward of better interoperability of data between products and organizations.

DISCLAIMER

The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies or positions, either expressed or implied, of the U.S. Government, U.S. Department of Defense, or the U.S. Marine Corps.

REFERENCES

DOD. (2001). *Department of Defense Handbook: Instructional Systems Development/Systems Approach to Training and Education (Part 2 of 5 Parts), MIL-HDBK-29612-2A.* Retrieved June 14, 2020, from https://quicksearch.dla.mil/ qsDocDetails.aspx?ident_number=207794.

DOD. (2018). *Manual for the Operation of the Joint Capabilities Integration and Development System.* Retrieved June 14, 2020, from https://www.dau.edu/cop/rqmt/DAU%20Sponsored%20Documents/Manual%20-%20JCIDS,%2031%20Aug%202018.pdf.

DOD. (2019). Program Manager for Training Systems Product Catalog. Retrieved June 14, 2020, from https://www.marcorsyscom.marines.mil/ Portals/105/PfM/TRASYS/2019%20P_S%20Catalog.pdf?ver=2019-02-08-095828-917.

DOD. (2020). Experience xAPI. Retrieved June 14, 2020, from https://www.adlnet.gov/projects/xapi/.

Duke, D. & Shook, W. (2010). Using Department of Defense Architectural Framework (DODAF) to Identify Initial Training System Requirements. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*, Paper No. 10180.

Estock, J. L., Alexander, A. L, Gildea, K. M., Nash, M., & Blueggel, B. (2006). A Model-based Approach to Simulator Fidelity and Training Effectiveness. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*, Paper No. 2794.

Fullmer, C., Woodard, P, & Matusof, R. (1990). Interoperability: The Key to Successful Team Training and Rehearsal. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*, 414-419.

Guptill, R & Cipparone, J. (2001). Defining Mission-Based Training Requirements: Connecting the Dots. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*.

Heffley, R. K., Clement, W. F., Ringland, R. F., Jewell, W. F., Jex, H. R., McRuer, D. T, & Carter, V. E. (1982). Training Requirements Determination During Early Phases of Weapon System Acquisition. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*, 437-447.

Knight, S. N. (1990). Issues Affecting the Networking of Existing and Multifidelity Simulations. *Summary Report: The Second Conference on Standards for the Interoperability of Training Systems, Vol. III.* 40-47.

Loeffelman, G. A. (2019). *Developing A Scaled Performance Evaluation Measurement System to Evaluate Marine Performance. Defense Technical Information Center.* [Master's thesis, Naval Postgraduate School]. https://apps.dtic.mil/dtic/tr/fulltext/u2/1080318.pdf.

McCallister, F., & Gray, M. (1999). Improving Training System Acquisition Through Process-Based Requirements Engineering. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*.

Office of Naval Research, The. (2017). Broad Area Announcement N00014-16-S-BA06: Future Integrated Training Environment. Retrieved June 14, 2020, from https://www.onr.navy.mil/-/media/Files/Funding-Announcements/BAA/2016/N00014-16-S-BA06.ashx.

Open Geospatial Consortium. (2020). CDB. Retrieved June 14, 2020, from https://www.ogc.org/standards/cdb.

Schreiber, B. T., Bennett, Jr., W. B., & Gehr, S. E. (2006). Fidelity Trade-Offs for Deployable Training and Rehearsal. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*, Paper No. 2840.

SISO. (2019). *SISO-REF-010-2019: Reference for Enumerations for Simulation Interoperability,* Orlando, FL.

Tolk, A., & Muguira, J. A. (2003, September). The Levels of Conceptual Interoperability Model. In Proceedings of the 2003 Fall Simulation Interoperability Workshop (Vol. 7, pp. 1-11). Citeseer.

Tolk, A., Ihrig, M., Page, E. H., Szabo, C., Heath, B. L., Padilla, J. J., Suarez, E.D., Weirich, P., & Yilmaz, L. (2013, December). Epistemology of modeling and simulation. In 2013 Winter Simulations Conference (WSC) (pp. 1152-1166). IEEE.

U.S. Government Accountability Office. (2016). *Technology Readiness Assessment Guide, GAO-16-410G.* Retrieved June 14, 2020, from https://www.gao.gov/assets/680/679006.pdf.

Vaughan, Jr., J. J. (1979). Training Requirements Determination During Early Phases of Weapon System Acquisition. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*, 437-447.