A Multi-level Universal Specification for Intelligent Characters (MUSIC)

Todd W. Griffith, Jason R. Potts, Colin A. Puskaritz

Discovery Machine, Inc.

Williamsport, PA

Lt Col Byron R. Harder

USMC TECOM

Marine Corps Training and Education Command

Quantico, VA

ABSTRACT

Branches of the military are wasting time and money by duplicating efforts for proprietary scripted training systems. Training simulations offer tremendous promise in reducing costs and increasing realism but fail to achieve either aim. Efforts to create intelligent agents have been undertaken, but often are constrained to one simulation environment, leading to limited success. Standardization has failed, due to the fact that competing companies cling to their own solution as the "best and only way". The Multi-level Universal Specification for Intelligent Characters (MUSIC) provides a standard that participating simulation systems (PSS) can adopt to gain access to intelligent characters that can act within their environment.

MUSIC will not require each participating cognitive system (PCS) to code in any particular programming language, but will provide rules of when and how to display situations to trainees within a standalone or federated simulation. Each simulation, regardless of its fidelity, will be able to leverage aspects of MUSIC to enhance training. By specifying things like Levels of Detail (LoD), simulations can automatically adjust based on the current training need and their capabilities. For instance, a Semi-Automated Forces (SAF) system or a bird's eye view simulation of a village may only need to show basic details, whereas first person 3D simulations might require an elaborate set of animations. This paper discusses the benefits of multiple levels of detail and explores different examples of how MUSIC can be deployed as a foundation for adaptable and scalable training solution that focuses finite investment dollars and maximizes returns.

ABOUT THE AUTHORS

Todd W. Griffith, Ph.D. has been working in the area of intelligent systems research for 25 years, and has published papers in the areas of cognitive science, HCI, and intelligent systems. He is frequently invited to present at workshops, conferences, and panels. Prior to founding Discovery Machine, Inc. (DMI), Dr. Griffith taught computer science at Georgia Tech and Bucknell University. Dr. Griffith has focused his research on building knowledge acquisition and deployment tools that enable SMEs to encode their own mental models and problem solving strategies on the computer. He has obtained grant funding through DARPA, NASA, NAVAIR, ONR, ERDC, MDA, USAF, and NSF. Dr. Griffith holds four U.S. Patents, and the RESITE® Suite, Knowledge Service Engine, Behavior Creation Toolkit, and Behavior Authoring Console for VBS are commercial realizations of his research.

LtCol Byron R. Harder, Ph.D. is the Chief Integration Officer for the Live, Virtual, and Constructive Training Environment at the U.S. Marine Corps Training and Education Command. He holds a Ph.D. in Modeling, Virtual Environments, and Simulation from the Naval Postgraduate School. His research interests include automated planning, behavior modeling, computer networks, software engineering, IT management processes, algorithms, and theory of computation.

Jason R. Potts is the Vice President of Software Development for DMI, overseeing the software development of their technology offerings, as well as the application of their technology to customer problems across a variety of domains. Since graduating with honors from WPI with a Bachelor's of Science degree in Computer Science, Mr. Potts has been designing and developing tools facilitating knowledge capture and representation of human expertise.

Colin A. Puskaritz is the Product Development Manager for DMI. He is a 2009 cum-laude graduate of Lycoming College where he studied business administration and psychology, ultimately earning a Bachelor of Arts Degree. Over the past ten years, he has focused his efforts on artificial intelligence and its applications to instruction and simulated training. In that time, he has focused on addressing customer needs by leading knowledge capture efforts, coordinating product development initiatives, and managing project deliverables.

A Multi-level Universal Specification for Intelligent Characters (MUSIC)

Todd W. Griffith, Jason R. Potts, Colin Puskaritz

Discovery Machine, Inc.

Williamsport, PA

Lt Col Byron R. Harder

USMC TECOM

Marine Corps Training and Education Command

Quantico, VA

INTRODUCTION

Intelligent agents are commanding a greater role in military training simulation applications. The increased use of Artificial Intelligence (AI) has emerged as a priority requirement to reduce the number of operators needed for training as well as improve the efficacy of training. For this reason many simulation systems provide some capability to script or create automated agents. The issue, however, is that every simulation is different. Military simulations come in many different varieties. There are large-scale constructive simulations often called computer generated force (CGF) or semi-automated force (SAF) systems. There are 3D immersive simulations for individuals or small units. There are part-task trainers (PTTs) for flying aircraft, driving vehicles, or maneuvering ships. There are also specialized trainers that use domes or virtual reality to train Joint Terminal Attack Controllers (JTACs) and weapons trainers for marksmanship. Additionally, there are trainers for aspects of the Theater Air Control System such as the Joint Theater Air Ground Simulation System (JTAGSS) that trains Air Support Operations Center personnel. This list is by no means exhaustive. In all of these simulation-based trainers the need for greater automation is evident. Each trainer, however, implements its entities in a different way; many relying on unique scripting languages and nearly all hard-coding behaviors to a specific terrain. The downside of this hard-coded approach is that knowledge acquisition work results in intelligent agents that are usable only in the environment for which they were captured. This is problematic for three reasons:

1) The subject matter expertise required to create behaviors for a specific simulation can be difficult to obtain. Subject Matter Experts (SMEs) are in high demand for their expertise and it is costly to take them away from what they do best, to work with knowledge engineers.
2) Once an agent is captured it is very difficult to transfer that agent to a new simulation system because each simulation may have different implementations, capabilities, functions, sensors, etc.
3) A large knowledge capture effort for one simulation system can lock a military organization into that simulation due to the vast amount of work required to translate that knowledge into a more modern simulator.

This paper describes a solution to these problems – the adoption of a Multi-level Universal Specification for Intelligent Characters (MUSIC) that enables the DoD to separate knowledge capture from the specifics of the simulation environment. Figure 1 shows how knowledge capture is done today. Models are developed in many disparate systems using scripting, finite state machine representation, behavior trees or goal-directed cognitive models. Each knowledge capture effort requires a Subject Matter Expert (SME) to work with developers, scripters or knowledge engineers to define the intelligent agent for the simulation of choice. Those efforts must then be duplicated every time a new simulation is introduced. Figure 2 shows how MUSIC can be used to capture a single knowledge representation that applies to all participating simulations and provides a framework that can be leveraged by future simulation system that are not yet developed.

Figure 1. Repetition of Disparate Knowledge Capture with Subject Matter Expert

Figure 2. MUSIC Knowledge Capture with Subject Matter Expert
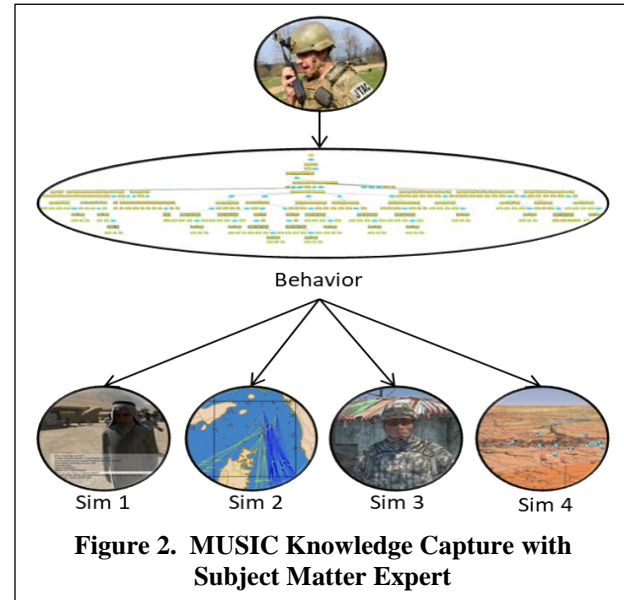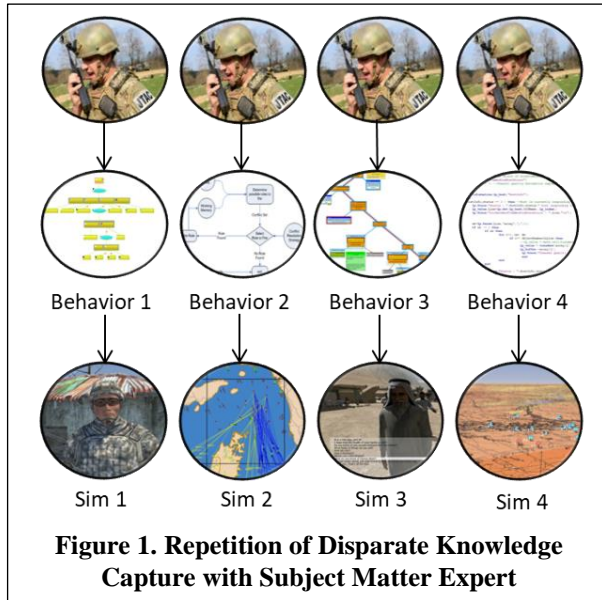
OBJECTIVES

What MUSIC Is

The Multi-level Universal Specification for Intelligent Characters (MUSIC) provides a standard that participating simulation systems (PSS) (e.g. simulation systems, immersive gaming environments, and semi-automated forces (SAF) systems) can adopt to gain access to intelligent characters that can act within their environment. These intelligent characters are driven by participating cognition systems (PCS) that drive the character's decision making. Broadly defined, characters include entities, such as vehicles, ships, or aircraft, but also avatars, agents and intelligent devices that can be leveraged within the participating simulation. MUSIC is a message passing specification that works with the existing High Level Architecture (HLA) or Distributed Interactive Simulation (DIS) message standards. MUSIC requires that each participating system implement the message specifications it describes.

At present, MUSIC is being developed as an extension to the IEEE 1278 Distributed Interactive Simulation (DIS) standard. DIS is a network protocol standard that provides a way for simulations and supporting systems to share information as part of a federation of systems. For example, a SAF system can share information about its entities through an entity state protocol data unit (PDU). This information typically describes things like the entity's location, orientation, velocity, etc. In a typical use case, one simulation may control the red aircraft, while another simulates white shipping, a third controls the blue forces, and yet a fourth may supply weather information. Using an interoperability solution such as DIS or HLA allows all the simulations to render all the entities and effects, thus providing an integrated common operating picture.

At first, the idea of a standard network protocol would seem to solve the problem. If one needs to reuse an intelligent entity developed for one specific simulation, one simply runs it in that simulation and then uses DIS PDUs to render it in all other federated simulations. In fact, this is what occurs in many distributed training commands today; a large number of simulations are federated together and the best entity representations from each are used to create the full operating picture. In such distributed environments, the behavior models of entities are generally contained within a single simulation. This has two disadvantages. First, it creates a potentially unnecessary dependency on the simulation which contains the behavior models for the entities, even if the end user (trainees) have no exposure to the primary features of that simulation. Second, it limits entities to only a single simulation's internal representation of the simulated world when making decisions about how to behave.

As an example, imagine a distributed training scenario that uses a system such as the Joint Semi-Autonomous Forces (JSAF) system federated together with an immersive 3D simulation like Virtual Battlespace (VBS). Perhaps

in this scenario, we are targeting two primary training audiences in a joint exercise, sailors on board a destroyer off the coast, as well as Marines on the ground in a coastal village. In this scenario, we have a population of civilians that are "owned" by the VBS system, driven by behaviors written in VBS script to appear as intelligent actors for the Marines to interact with, while also engaging in civilian fishing and shipping activities which impact the sailors interacting with the JSAF driven side of the simulation.

Due to the limitations imposed by requiring that the behaviors of these civilians exist internally to VBS, we are left with a situation where we need to run VBS to support the scenario, even if we were to refocus the scenario on only training the sailors on board ship. You might still want the same types of fishing traffic, but without Marines interacting with the population directly, you no longer need an immersive 3D simulation representing the civilians. However, if you were to remove VBS from the equation, you would need to then recreate some version of all of those civilian behaviors in JSAF in order for them to behave similarly. The MUSIC specification changes this in one very important way. The behaviors are not run by any one simulation. The behaviors are instead run independently by a PCS. This means that a single knowledge representation captures the behavior for all participating simulation systems, and each can render the behavior leveraging as much or as little detail as that simulation requires.

The MUSIC standard also requires that each intelligent character be defined to function at several levels of detail (LoD). Each participating system must support at least the lowest LoD in order to leverage the standard. The LoD concept is analogous to the concept used in computer graphics, where the system only renders details when the viewer is close enough to notice them. One does not need to see an avatar light a cigarette when observing a crowd from 20,000 feet in the air[1], nor can one engage that avatar in conversation from this distance. One does, however, still observe the avatar navigating from one place to another. By incorporating this type of information into the behavior specification and execution standard, MUSIC enables dynamic scaling of processing requirements for populations of characters based on real-time needs during training.

Finally, MUSIC requires each PSS to expose people, places, things, and environmental information to PCS characters. Characters can sense things around them; they observe objects in the terrain; they observe other characters and players; they observe the weather; they may have sensor systems such as radar or sonar. Characters can identify salient features within a terrain such as roads, street lights, hills or rivers. A PCS is most effective when these salient features are represented semantically within the simulation and can be obtained by the PCS when a character needs them.

What MUSIC Is Not

MUSIC is not a specification language for how to build intelligent characters. It is not a scripting language or a schema for building agents. A MUSIC character can be built in any programming or scripting language and driven by any decision making or cognitive engine. The specification does not, for example, describe how one should build an intelligent aircraft entity. MUSIC does not evaluate the quality of characters or the depth of their intelligence.

Similarly, MUSIC does not specify how participating systems are designed. MUSIC is limited to the interface between a PSS and a PCS through a message specification protocol. MUSIC does not control how each PSS carries out effects; for example, the gesture for "go away" may be visually different in different PSS instances.

MUSIC Compliance

Any PSS that implements and exposes the lowest LoD functions is considered MUSIC compliant. Any PCS that provides characters that can act within all MUSIC compliant PSS's is considered MUSIC compliant.

---

[1] There are circumstances where using high powered telescopic imaging that you could see that detail. The point, however, is that some user needs to be focused on that detail in order for it to be rendered.

How MUSIC Should Be Used

Training communities have turned to virtual simulation as a cost effective way to train. These systems reduce many of the costs associated with live training, including the cost of fuel and equipment. One of the highest remaining costs, however, is operator time. Intelligent characters can help to reduce operator time and increase training fidelity. The issue today, however, is that every simulator requires different intelligent characters to operate. These characters are usable only in the simulation for which they were designed, and are often limited to a single training scenario. As such, training communities are locked into simulation systems that are antiquated, or they are required to redo all of the knowledge engineering work needed to capture, encode, and deploy the "intelligent" characters into a new simulation environment.

MUSIC separates the knowledge engineering required to encode intelligent characters from the specifics of the simulation environment. It provides training communities with more choice, so they do not become locked into an environment due to time invested in scripting. Using MUSIC, once the work is done to capture, encode and deploy an intelligent character, it can be transferred to any MUSIC-enabled system. Thus, MUSIC can be used to focus knowledge engineering efforts toward the goal of capturing intelligent behaviors that can be used across simulations. MUSIC also opens new training possibilities by enabling automated human terrain to be leveraged in training environments, including environments containing disparate federated simulation systems.

The idea of MUSIC is to elevate the control of intelligent characters to be a service rather than an embedded function in a monolithic simulation. The more broadly adopted MUSIC becomes, the more libraries of intelligent characters can be exchanged and disseminated to an entire organization. The DoD, for example, could set up a repository of intelligent characters that could be leveraged in any simulation. In the future, MUSIC could become part of Modeling and Simulation as a Service (MSaaS) to apply to broad scale simulation efforts.

In addition, when working with connected or federated PSS and PCS instances, MUSIC shall ensure that each system renders the characters in the most effective manner. In a federation of simulations, one simulation traditionally "owns" the entity. This system provides the "entity state data" to the federation and each simulation renders the entity. Prior to MUSIC each simulation would render to the lowest common denominator, meaning even if a system had an ability to change a character's facial expressions or gestures that information would not be represented in the federation and thus the simulation would not be able to render it. MUSIC opens the door to these capabilities by rendering the effects in simulations that can render them but not in simulations that cannot.

Types of Intelligent Characters

The term "intelligent characters" is used broadly to include intelligent entities, units, devices, and agents. Each intelligent character is capable of interacting with the simulation in some way. However, not every PSS is required to support every intelligent character type.

Figure 3. Human avatars populate a market and use market stall devices in VBS3

Entities are characters that move within the simulation and include things like avatars, vehicles, and animals. Avatars are human-looking characters with which a trainee can interact. Vehicles are entities which may contain several humans within a platform or no humans at all. Things like ground vehicles (e.g. cars, trucks, tanks, etc.), surface vehicles (e.g. ships, boats), sub-surface vehicles (e.g. ships, boats), rotary-wing aircraft (e.g. helicopters), fixed wing aircraft (e.g. fighter planes), and/or missiles/projectiles (e.g. tomahawk, hellfire) are examples of vehicles. Animals are the simplest type of entity whose primary purpose is to enhance simulation experiences in areas like patterns of life.

Units represent a collection of entities which operate as part of a group. Each unit can be comprised of a combination of entities but will carry out a coordinated set of tasks. For example an infantry platoon might consist of forty-three individual Marines while a fire team might consist of five Marines. In both cases, the units would

interpret surroundings, adapt to the environment, and make decisions in conjunction with other entities in their unit in order to accomplish a set of common objectives.

Devices are things that players or characters can interact with to achieve an effect in a simulation. Some examples include a surface to air missile (SAM) site, an oil or natural gas wellhead, or a street light. With each of these examples, the trainee could impart some sort of change on the environment through an interaction (e.g. fire a weapon, open a gas valve, or turn on a street light respectively).

Agents do not have a visible presence within a simulation environment. They are more often "God's eye" components for instructors to utilize. Common uses of agents might be instructor control tools which are used to augment a real-time training event, analysis agents which interpret some aspect of a simulated event, after action review elements used to monitor trainee progression and provide feedback upon completion, etc.

Figure 4. Human avatar interacts with wellhead device in RESITE® Unity Simulation

Different training contexts require different types of intelligent characters. For example training using pattern of life, might require more entities and devices, but may have less of a need for agents. Conversely, an entire fleet synthetic training event might require many agents to analyze performance, assist instructors, and aid stressed trainees. No matter the training need, a combination of the above intelligent agents is essential to optimal results.
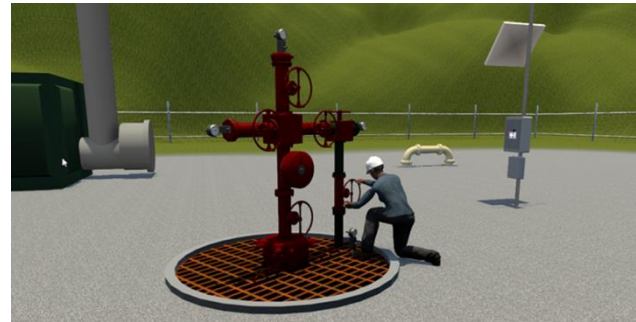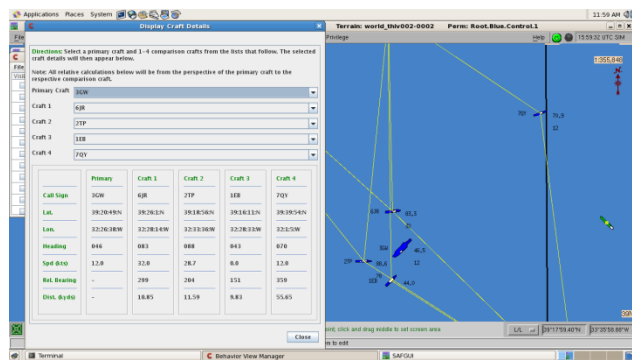

Figure 5. Surface ship vehicles performing mission in Joint Semi-Automated Forces (JSAF) while Instructor views agent summary

INTEGRATION AT THE COGNITIVE LEVEL

The MUSIC standard enables simulations to offload character decision making to an external cognition engine; thus separating the cognition from the simulation. In federated systems leveraging DIS or HLA, one simulation will "own" the character but the MUSIC enabled PCS shall drive the character's decision making in every PSS. MUSIC characters should leverage the physics provided by the PSS that owns the character including any navigational meshes or collision boundaries. Whenever possible the physical capabilities of a vehicle should be obtained from the PSS and provided to the PCS. This means that a character that has associated animations may display those animations in a 3D immersive simulation but would not in a 2D semi-automated forces (SAF) simulation.

Another way to think about MUSIC is that it enables the system to have "automated operators" controlling the characters within the PSS. The PCS controls the decisions that the characters make, while the PSS controls the rendering or physical results for the characters within the environment.

Discovery Machine, Inc. (DMI) has demonstrated this cognitive integration with numerous SAF, immersive systems and task trainers. The models used in these simulations are structured in the same way but each leverages a different set of low level functions. DMI's knowledge capture methodology enables SMEs to introspect and articulate their decision making in an iterative manner, to rapidly capture expertise needed for intelligent characters.

MUSIC shall enable the results of knowledge capture to reach a larger number of simulation systems while more efficiently limiting the time SMEs need to spend in knowledge acquisition sessions.

LEVELS OF DETAIL

Almost all military training simulations today *model* the world in 3 dimensions. Many, however, do not *render* anything in 3D, instead preferring a top-down view showing those entities on a map, sometimes called a plan view display (PVD). A PVD can show both single entities as well as aggregated groups of entities, i.e. units. Some PVDs will show individual entities when you are zoomed-in on the unit but aggregated when you are zoomed-out. This is a simple case of providing dynamic "levels of detail" to the user. Levels of Detail (LoD) is a concept first used in 3D immersive video games to save on processing. The concept is a simple one, as objects recede they require less graphical detail. For example, a tree up close would be rendered with all its leaves, while a distant tree would show a patch of color representing those leaves.

The MUSIC standard uses an analogous concept for behaviors of entities. As the trainee's focus of attention shifts away from an intelligent agent (also called a non-player character (NPC)), its behavior can be rendered in less detail. For this example, consider an intelligent agent representing a shop keeper in a pattern of life (PoL) training simulation. Following its PoL behavior, the shop keeper NPC takes a break at 11:20a.m. and goes outside to smoke a cigarette and make a call on his cell phone. If one is in an immersive simulation and this occurs nearby, one would see an animation play showing the NPC smoking the cigarette and then making the call. If, however, one is viewing these characters from a SAF using a PVD, then the animations would not be shown; only the movement of the character leaving the store and returning after some time. Using dynamic LoD in the immersive simulation, the only time those actions would be rendered is when a user is focused on them. The participating cognitive system would know they occurred, how long they took, and any outcomes, but would not have to actually execute those actions within the simulation. This is significant because it can serve to offload processing from the simulation to the cognitive system which can be on an entirely separate server.

There are other ways in which LoD can be used dynamically to reduce the amount of computational load on the system. All players within an immersive environment have a field of view that is displayed on the screen or in their Virtual Reality (VR) headset. Any characters that do not fall within some player's field of view does not need to be rendered. The behaviors of these characters can be simulated by the PCS without a full rendering of those behaviors. The character may have smoked a cigarette and may have made a cell phone call. This could have effects such as the player having one fewer cigarette, and a piece of information may have been propagated to another character, but neither of these effects would require the system to run the animations.

This leads to a more important purpose for LoD: the acquisition of knowledge from SMEs. Most knowledge capture done today is constrained by the simulations for which it is being created. Why capture knowledge that will never be used by the simulation? In general this makes sense. If one is constrained to use a particular simulation, then one should tailor knowledge capture to that simulation. Unfortunately, this is what leads to knowledge capture being done over and over for the same expertise. The government ends up paying for the same knowledge capture dozens of times. As intelligent agents become more prevalent this repeated knowledge acquisition will increase. The MUSIC specification enables the capture of behaviors with the ability to use them in any participating simulation system, which means that the knowledge capture only needs to happen once. It is possible that knowledge capture efforts to support generalizable behaviors may be more costly than to support a single PSS, but as simulation systems adopt newly developed behaviors, minimal new investment would need to be made. Thus, the total effort required would result in a net savings of both time and money. LoD helps this process by freeing the knowledge acquisition from the constraints of any particular simulation. If a behavior is represented in too much detail for a simulation, the participating simulation only needs to leverage those aspects of the behavior that are relevant. For example, a SAF would not require the animations for smoking, whereas an immersive simulation would. The same behavior, however, can be used in either simulation, so long as it participates in the MUSIC specification.

When designing an intelligent character for MUSIC one must account for levels of detail (LoD). Not all participating systems will have all capabilities. The lowest LoD shall include only functions that all participating systems can simulate. This includes basic movement, orientation and sensor capabilities (many of which are provided by standard DIS and HLA implementations) but also access to characters within their proximity, semantic terrain features and environmental features. Stated another way, every system participating in MUSIC shall be able to simulate every function at the lowest LoD. The MUSIC standards body shall determine where this lowest level resides.

Any MUSIC character that uses functionality above the lowest LoD must still function in a simulation at the lowest LoD. This means that PCS intelligent character developers must provide some mechanism for scaling their characters. Stated another way, if the PSS cannot render the MUSIC message, the PCS character must be able to ignore that message in a graceful manner.

Levels of detail apply to the capabilities of characters within the MUSIC specification. Every character must meet the minimum LoD to be a MUSIC character.

MUSIC SIMULATION INTERFACE LAYER

Both DIS and HLA provide a basic integration layer for enabling simulations to share a common operating picture (COP) that displays known entity locations and their movements. Existing federation object models (FOM) for HLA and Protocol Data Units (PDU) for DIS do not dictate how characters perceive their environment nor do they offer hooks into characters required to demonstrate cognitive activity. Both HLA and DIS, however, allow integrators to specify arbitrary information that gets published to the federation. This arbitrary information can be used to establish the same effect as MUSIC for a given federation, but does not provide any standard for PSS or PCS developers to follow when building their FOM or customized PDUs.

The MUSIC standard shall specify the features and elements needed in a FOM or PDU specification to enable any MUSIC enabled PCS to provide characters for a PSS. It also describes the messages that each PSS must support to enable the PCS to control its entities. See Figure 6 for an example of how various PCSs might be connected to PSSs using MUSIC.
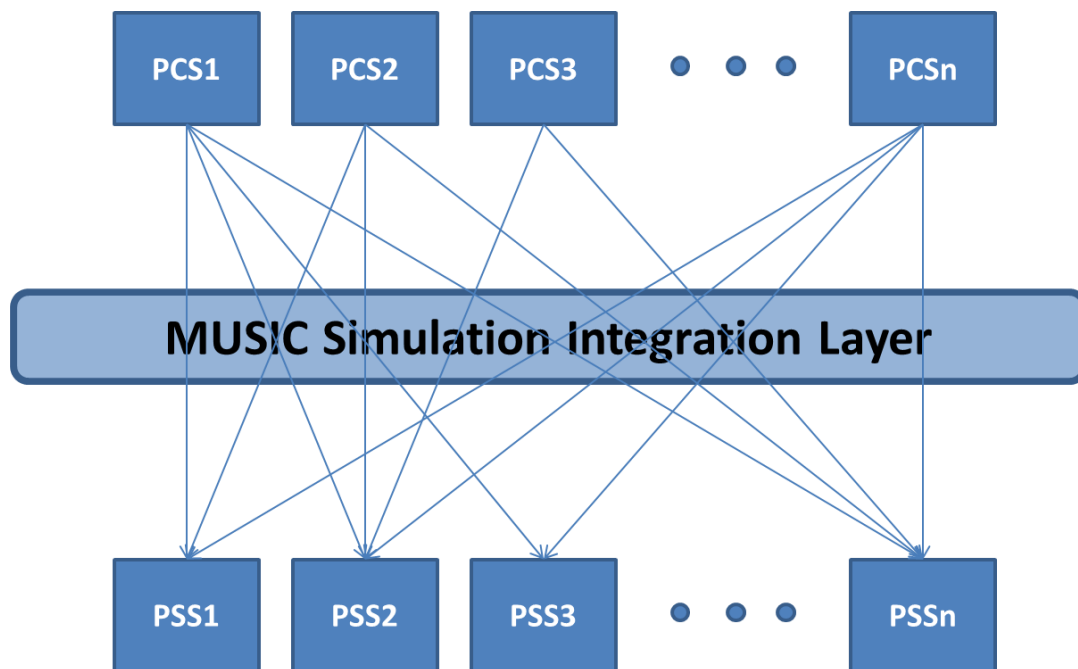

Figure 6. This diagram depicts how each PCS can provide information to one or more PSSs.

By moving the execution of entity behaviors out of a single simulation into a standards based execution designed to function in a distributed environment, MUSIC enables behaviors to automatically adapt their execution to a varying set of capabilities provided by multiple connected simulations. To illustrate this, again consider the case of a SAF system connected to an immersive 3D simulation. The SAF might "own" a large number of civilians, simulating

their movement and pushing them out to the network. The immersive simulation renders a detailed 3D model of those characters as necessary. The SAF is well suited to modeling a large number of entities, but not particularly well suited to managing things like playing animations, since they mean nothing in the SAF environment. We could push behaviors directly into the 3d simulation, but it may be less well suited to the scalability issues that arise from the large numbers of entities that SAF's often deal with.

MUSIC offers a way in which both systems can contribute to a hybridized solution. A single behavior model, executing separately from either simulation, makes calls out over the network to actuate the entities. The behavior model defines the activities of the entities, without regard to the individual capabilities of the underlying simulations. When the behavior model calls for the entity to move, the simulation that owns that entity and is capable of moving it does so, and other simulations see that movement. When the behavior model requests that an animation be played, simulations where such an action means something (like an immersive 3D simulation) do so, while others (like a SAF) ignore the request. In this way, the behavior's execution is automatically supported by the underlying simulations, expressing itself in a way that most makes sense based upon what is being used.

TERRAIN REQUIREMENTS

Each PSS must either provide semantic markers within its terrain or be federated with a terrain database that provides this information. A semantic marker is a geotagged location paired with a type. For example, the type may be a building and may have sub-types such as dwelling, restaurant or shop. MUSIC shall specify the minimal set of accessible terrain features that a low LoD character would need to act within a PSS. For example, in a pattern of life, all characters need to know where dwellings, restaurants, shops and places of worship are within their environment. Similarly, any combat entity may need to know about choke points, high ground and natural barriers such as rivers or canyons.[2]

For ground entities such as cars, trucks, tanks, dismounted infantry, or civilians, the use of terrain in both movement and decision making is critical to intelligent decision making. Unlike avatars or entities that are driven by human operators, automated characters cannot use visual processing to navigate their terrain. Machine vision has come a long way, but it is still too computationally expensive to apply to hundreds or thousands of automated characters within a simulation. The alternative is to use navigational meshes and semantic geotagging of places and objects. A navigational mesh, i.e. navmesh, is a web of points woven into a terrain that guide automated characters in their movements. A geotag is a marker in the terrain that tells characters that something is there. A semantic geotag tells the character what that something is; for example, a tree, coffee shop or street light. Characters can navigate to specific places or objects by traversing the navmesh to an identified geotag.

Characters can leverage semantic terrain information to make more intelligent decisions within the simulation environment. One objective of MUSIC is to standardize how characters receive this information. The current DIS specification makes use of enumerations to specify things such as vehicle types or weapon types. The MUSIC specification shall extend this specification to additional categories of places and objects one might find in a terrain. Places can be permanent places such as a police office or dynamic places such as an insurgent meeting place. The temptation is to create a full ontology of everything or use existing ontology systems such as CYC to represent all the semantic categories of objects and places (Lenat, Prakash, & Shepherd, 1985). This, however, would be overkill. The semantically tagged terrain does not need to support inferences about object relationships (e.g. trees have-part branches) but rather simply know which objects are present. The character's behaviors can then use that information to act or leverage its own internal ontological representations to infer whatever is needed. The purpose of a geotagged terrain is to compensate for the character's lack of image processing capabilities, not provide the character with an inferencing capability which is the province of the PCS.

It should also be noted that MUSIC does not require that terrain information come directly from the participating simulation system (PSS). Any terrain database can provide the semantic geotags for the participating cognitive system (PCS) as long as it is correlated to the PSS. Geospatial information systems (GIS) commonly have correlated layers of information that are leveraged by human users. MUSIC characters can also leverage this

---

[2]This can get complex when terrain means different things to different echelons of command and types of units (a choke point for a battalion might not be a choke point for an individual).

information so long as it is provided in a consistent manner. The use of DIS enumerations could enable this consistency. For instance, a tree might be object 10, where a deciduous tree is 10.1 and a maple tree is 10.1.2 and a Japanese maple is 10.1.2.15. The MUSIC character can thus use as much information as they require about the object. A character may hide behind the nearest tree, or find an Improvised Explosive Device (IED) component hidden under the Japanese Maple tree.

ENVIRONMENTAL REQUIREMENTS

Some PCS implementations have characters that can react to environmental conditions such as weather or bathymetry. The MUSIC standard shall define messages to provide this information from the PSS to the PCS. The PSS can provide the information directly or via an associated federated service. MUSIC characters, however, should still function gracefully when this information is not obtainable. The behavior of a character without this information, however, will be different than when the information is available. For example, a submarine that can use layer depths to hide will behave differently than a submarine that does not have access to that information. MUSIC shall define the messages that enable characters to obtain that information.

Environment is again an area that is different for automated characters than it is for human controlled characters. If, for example, one is a player in an immersive simulation in which fog or a dust storm has decreased visibility to 10 meters, this can simply be seen by the player in the environment. The player would then not be able to see adversaries 20 meters away. An automated character, on the other hand, would need to leverage that visibility to constrain its field of view. Since the character does not actually "see" the adversary in the traditional sense it needs to leverage that information to simulate the lack of visibility.

CURRENT EFFORTS

DMI is leading an effort to provide an initial specification by working with the Air Force and the Joint Staff to introduce MUSIC to SISO and position the specification for wide adoption throughout the DoD. DMI is leveraging current efforts for the U.S.M.C. in patterns of life (PoL), U.S. Navy in Future Operations Planning for Theater Anti-Submarine Warfare (TASW) and U.S. Air Force in Air Support Operations Center (ASOC) training to provide initial MUSIC enabled characters. These characters demonstrate character interoperability and simulation independence. The key to this is that behaviors are not rewritten or modified to work in different environments, but are able to leverage the specific aspects of each simulation system to carry out their decision making.

The initial MUSIC message specification is under development and shall be ready for review and initial distribution in 2018.

CONCLUSION & FUTURE WORK

The work completed to date as described in this paper is backed by research in goal-oriented cognitive architectures. Based on success implementing behaviors models for a variety of simulation systems within many branches of the DoD, it is our belief that MUSIC shall reduce costs for capturing and deploying automated characters in simulation. Only with the ability to adapt to changing simulation environments, is it possible for the DoD to effectively and efficiently leverage knowledge capture from Subject Matter Experts. The increasing need of AI to reduce the efforts of operators and reduce the costs of simulation training can only be achieved by a concerted effort to maximize SME time and reduce replication of expensive knowledge acquisition. As simulation systems become increasingly commoditized they are attempting to differentiate themselves by adding AI capabilities. All 3D simulations have similar visual capabilities, but they vary greatly in their support of AI entities and characters. The temptation from DoD sponsors will be to fund the simulation systems to directly implement AI solutions into their environments. This is a mistake. The DoD shall be better served by recognizing that cognitive systems are best produced by companies that specialize in cognitive systems, while also supporting a standard such as MUSIC to enable their simulators to make use of the behaviors. To do otherwise, will trap each DoD organization into a situation where they cannot choose another simulator without first committing to redoing all of the acquisition that was done to support it.

ACKNOWLEDGEMENTS

REFERENCES

Garcia, C.J. & Griffith, T.W. (2005) *A Composable Behavior Modeling System for Rapidly Constructing Human Behaviors,* Proceedings of I/ITSEC 2005.

Griffith, T.W. (2006) *Domain Specific Knowledge Capture Interfaces for Behavior Modeling,* Proceedings of I/ITSEC 2006

Harper, J. (2017, November 17). Training Investments Expected to Shift to Simulations. Retrieved January 22, 2018, from http://www.nationaldefensemagazine.org/articles/2017/11/17/training-investments-expected-to-shift-to-simulations

Hegde, M., Allison, D. & Griffith, T. (2014). Reflecting on Cognitive Process Models to Enhance Adaptive Training. *MODSIM World Proceedings.*

Hegde, M., Allison, D. & Griffith, T. (2014). A Framework for Enabling Virtual Observer Controllers in Synthetic Training. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference.*

Lenat, D., Prakash, M., & Shepherd, M. (1985, Winter). CYC: Using Common Sense Knowledge to Overcome Brittleness and Knowledge Acquisition Bottlenecks. AI Magazine, 6(4), 65-85.

Maxwell, D. (2017). Large Scale Testing and Evaluation of Virtual Environments for Infantry Soldier Tasks Comparing Mental Effort for Live Versus Virtual Training Performance Assessments. *Proceedings of the MODSIM World Conference 2017.*

Potts, J.R., Griffith, T., Roth, K., & Snyder, J.. (2012) Customizable Speech Centers for Automated Entities within Simulation. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference.*

Potts, J., Griffith, T., Sharp, J., & Allison, D. (2011). Subject Matter Expert-Driven Behavior Modeling Within Simulation. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2011.*

Schatz, S., Folsom-Kovarik, J. T., Bartlett, K., Wray, R. E., & Solina, D. (2012). Archetypal Patterns of Life for Military Training Simulations. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2012.*

Sharp, J. J., & Potts, J. R. (2011). Improving Trainee Engagement Levels through Adaptive Entity Behaviors. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2011.*

Snyder, J. K., Morse, S. R., Potts, J. R., & Griffith, T. (2013) Cognitive Projection of Future States by Autonomous Entities. To appear in the Interservice/Industry Training, Simulation, and Education Conference.

Stensrud, B. S., Purcel, E. R., Fragomeni, G., & Garrity, P. (2012). No More Zombies! High-Fidelity Character Autonomy for Virtual Small-Unit Training. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2012.*

Zimmerman, L. A., Mueller, S. T., Marcon, J. L., Daniels, J. B., & Vowels, C. L. (2011). Improving Soldier Threat Detection Skills in the Operational Environment. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2011.*