Rapid Prototyping for Simulation and Training with the Rapid Integration & Development Environment (RIDE)

Arno Hartholt, Kyle McCullough, Ed Fast, Andrew Leeds, Sharon Mozgai, Tim Aris, Volkan Ustun, Andrew S. Gordon

USC Institute for Creative Technologies

Playa Vista, CA

{hartholt, mccullough, fast, leeds, mozgai, aris, ustun, gordon}@ict.usc.edu

Chris McGroarty

US Army DEVCOM Soldier Center
Soldier Effectiveness Directorate, SFC Paul Ray Smith Simulation and Training Technology Center

Orlando, FL

christopher.j.mcgroarty.civ@mail.mil

ABSTRACT

This paper describes the current status and next steps of the Rapid Integration & Development Environment (RIDE). RIDE is a foundational Research and Development (R&D) platform that unites many Department of Defense (DoD) and Army simulation efforts to deliver an accelerated development foundation and prototyping sandbox that provides direct benefit to the US Army's Synthetic Training Environment (STE) as well as the larger DoD and Army simulation communities. RIDE integrates a range of capabilities, including One World Terrain (OWT), Non-Player Character (NPC) Artificial Intelligence (AI) behaviors, Experience Application Programming Interface (xAPI) logging, multiplayer networking, scenario creation, machine learning approaches, and multi-platform support. It leverages robust game engine technology while designed to be agnostic to any specific game or simulation engine. RIDE is freely available through Government Purpose Rights (GPR) with the aim of lowering the barrier to entry for R&D efforts within the simulation community, in particular for training, analysis, exploration, and prototyping. This paper provides lessons learned in developing this R&D sandbox, use cases that highlight how RIDE is currently being used in the broader Science and Technology (S&T) community, and an overview of next steps.

ABOUT THE AUTHORS

Arno Hartholt is the Director of R&D Integration at the University of Southern California Institute for Creative Technologies (USC ICT) where he leads the central R&D integration group. He is responsible for much of the technology, art, and processes related to integrated systems, with a particular focus on the interchange of research and industry capabilities. He has a leading role on a wide variety of research prototypes and applications, ranging from medical education to military training and treatment. Hartholt studied computer science at the University of Twente in the Netherlands where he got his Master's degree. He worked at several IT companies, from large multinationals to early start-ups, before accepting a position at USC ICT in 2005. He has over a decade's worth of experience in leading multidisciplinary research and commercial projects, with an emphasis on virtual humans, virtual reality (VR), augmented reality (AR) and serious games.

Kyle McCullough is the Director of Modeling & Simulation at USC ICT. His research involves geospatial initiatives in support of the Army's One World Terrain project, as well as advanced prototype systems development. His work includes utilizing AI and 3D visualization to increase fidelity and realism in large-scale dynamic simulation environments, and automating typically human-in-the-loop processes for Geo-specific 3D terrain data generation. Kyle received awards from Interservice/Industry Training, Simulation, & Education Conference (I/ITSEC) and the Raindance festival, winning "Best Interactive Narrative VR Experience" in 2018. He has a B.F.A. from New York University.

Ed Fast is a Computer Scientist at the University of Southern California Institute for Creative Technologies where he has been the Technical Lead on a number of projects for over a decade. While his primary projects have been virtual humans related, the main goal has been to build reusable systems for a variety of research prototypes and applications.

He previously worked in the video game industry and has enjoyed rewarding success in applying that expertise to simulations. Ed holds a BS in Computer Science from California State Polytechnic University, Pomona.

Andrew Leeds is a Technical Support Lead at USC ICT. His professional work focuses on medical and training application development and the integration of novel, multisensory VR systems. He directed and edited "A Soldier's Tale," an immersive VR narrative exhibit for the Heroes Hall veteran's museum. Andrew received a B.A. in Film and Video from Columbia College Chicago.

Sharon Mozgai is the Associate Director of the Medical Virtual Reality lab at USC ICT where she heads up the R&D efforts for clinical applications of VR, including VR Exposure Therapy, in addition to Virtual Human training and education. Her background is in psychology with a Master's degree from Harvard University. She has worked at a range of organizations, focusing on computational linguistics at MIT, organizational behavior at Harvard Business School, cognitive behavioral therapy and motivational interviewing at the San Francisco Veterans Affairs (VA) Medical Center, and VR, Virtual Humans, and AI at USC ICT and at several tech startups.

Tim Aris is a graduate research assistant at USC ICT where he works to train reinforcement learning agents. His research interests include reinforcement learning, deep learning, artificial intelligence, hierarchical systems, and AI safety. He has completed a MS in Computer Science at USC.

Volkan Ustun is the Associate Director of the Cognitive Architecture Group at USC ICT. His general research interests are artificial intelligence, multi-agent systems, computational cognitive models, and simulation. He has a Ph.D. degree in Industrial and Systems Engineering and held postdoc positions at Georgia Tech and Rice University before joining ICT.

Andrew S. Gordon is the Director of Interactive Narrative Technologies at USC ICT, and Research Associate Professor in the USC Department of Computer Science. He is the author of the 2004 book, *Strategy Representation: An Analysis of Planning Knowledge*, and the 2017 book, *A Formal Theory of Commonsense Psychology: How People Think People Think* (with Jerry R. Hobbs). He received his Ph.D. in 1999 from Northwestern University.

Chris McGroarty is the Chief Engineer for Advanced Simulation on the Advanced Modeling & Simulation Team at the US Army Combat Capabilities Development Command, Soldier Center, Soldier Effectiveness Directorate, Simulation and Training Technology Center (DEVCOM SC SED STTC). His research interests include distributed simulation, novel computing architectures, innovative methods for user-simulation interaction, methodologies for making simulation more accessible by non-simulation experts, service-oriented architectures and future simulation frameworks. He manages and leads a variety of research efforts that mature, integrate and demonstrate these technologies in a relevant Army and Department of Defense context. He received his Master of Science and Bachelor of Science in Electrical Engineering from Drexel University in Philadelphia, Pennsylvania.

Rapid Prototyping for Simulation and Training with the Rapid Integration & Development Environment (RIDE)

Arno Hartholt, Kyle McCullough, Ed Fast, Andrew Leeds, Sharon Mozgai, Tim Aris, Volkan Ustun, Andrew S. Gordon

USC Institute for Creative Technologies

Playa Vista, CA

{hartholt, mccullough, fast, leeds, mozgai, aris, ustun, gordon}@ict.usc.edu

Chris McGroarty

US Army DEVCOM Soldier Center
Soldier Effectiveness Directorate, SFC Paul Ray Smith Simulation and Training Technology Center

Orlando, FL

christopher.j.mcgroarty.civ@mail.mil

INTRODUCTION

Military simulations are complex systems, with many interconnected capabilities to support training, mission command, and operations. These simulations require realistic models of allies, adversaries, and civilians as well as the equipment they use and processes they follow. Effective systems need to be adaptive, respond to user input, be accessible at the point of need, and scale to large numbers of users, simulated agents, and other assets. In isolation, each of these are a challenge to design, develop, deploy, and maintain. Combining them into a single, effective and efficient simulation is a massive undertaking, exacerbated by the need to provide useful systems today while preparing to take advantage of technological advances tomorrow.

Given these requirements and challenges, military simulation stakeholders include a wide range of organizations and individuals, encompassing government, academia, and industry. Each of these stakeholders has their own culture, expertise, terminology, processes, and incentives, which can lead to organizational challenges that can compound the technical challenges.

These organizational aspects, together with modern technical advances, give rise to a fascinating paradox: while the potential to create more relevant, effective, and efficient military simulations has never been higher, it has become increasingly difficult to realize that potential. With an ever-growing number of technologies and services to leverage, proper assessment and validation has become more complicated. Furthermore, even if best-of-breed technologies are identified, combining them into a single system is challenging due to its inherently combinatorial complexity. This impedes decision-makers' ability to identify the state-of-the-art and define meaningful requirements and metrics in the first place. With clear technical requirements lacking, it becomes increasingly difficult to make informed decisions regarding necessary trade-offs throughout the design, development, and deployment process.

This complexity, intrinsic to both the technical and organizational processes, make it increasingly challenging to properly define what needs to be created, how, by whom, when, and even why.

Our aim is to help manage this complexity through the Rapid Integration & Development Environment (RIDE), an open research & development (R&D) platform that combines many simulation capabilities into a single framework. RIDE is a shared, rapid prototyping platform, independent of any specific game or simulation engine, with native, integrated support for synthetic One World Terrain (OWT), artificial intelligence (AI) and machine learning (ML) frameworks, scenario creation tools, networked multiplayer, battle drill performance assessors, Experience Application Programming Interface (xAPI) logging, Distributed Interactive Simulation (DIS) messaging, web services, and multi-platform support.

In this paper we describe RIDE, its background, architecture, systems, capabilities, and current applications and use cases. We end with a discussion on opportunities, limitations, and next steps. Through this, we aim to show that RIDE is an unparalleled tool in managing the complexity described above, by 1) connecting both people and technologies, 2) making requirements and trade-off decisions explicit, and 3) showing the path of the future.

BACKGROUND

Due to the complexity of military simulations, existing systems typically exhibit shortcomings. For example, in order to be able to provide the desired level of simulation fidelity, they may not run in real-time, severely limiting their applicability. They may be federated systems, which reduces interoperability, cross-system tracking, and overall efficiency. They may be outdated, not being able to leverage the latest technological advances. Finally, they may be proprietary or closed, severely diminishing the ability to grow, re-use, and integrate relevant capabilities.

To address these shortcomings, the Army Modernization priorities have directed the need for a Synthetic Training Environment (STE) that is a holistic training system which leverages the latest in real-time simulation capabilities, enables domain experts to create and modify content, and allows Warfighters to train on demand at the Point of Need. While many of the individual capabilities that the STE requires exist in one form or another, efforts to integrate them into a single environment are ongoing. In addition, no common R&D framework exists where all stakeholders can identify the possibilities for the near future. RIDE provides that framework, not only for the STE and Army training needs, but for military simulations more broadly.

RIDE is being developed at the University of Southern California's Institute for Creative Technologies (USC ICT). USC ICT is a University Affiliated Research Center (UARC), working in collaboration with the Army DEVCOM SC SED STTC and Army Research Laboratory (ARL). USC ICT has a rich history of performing research and creating systems that teach, train, assess and heal. For example, UrbanSim is a PC-based virtual training application for practicing the art of mission command in complex counterinsurgency and stability operations, consisting of a game-based practice environment, a web-based multimedia primer on doctrinal concepts of counterinsurgency and a suite of scenario authoring tools (Wansbury et al., 2010). ELITE/INOTS is a virtual human and intelligent tutoring system for instruction, practice and assessment of interpersonal communication skills for Army and Navy leadership and counseling (Hays et al., 2012). BRAVEMIND is an evidence-based Virtual Reality Exposure Therapy (VRET) system aimed at providing relief from post-traumatic stress, allowing clinicians to gradually immerse patients into virtual environments representative of their traumatic experiences in a controlled, stepwise fashion (Rizzo et al., 2014; Mozgai et al., 2020). SimSensei is a basic research clinical decision support tool and interactive virtual agent-based healthcare dissemination system that is able to recognize and identify psychological distress from multimodal signals, including signs that may be indicative of post-traumatic stress (DeVault et al., 2014). It uses MultiSense, a framework for audio-visual sensing modules, to gather rich data of a user interacting with a virtual human (Scherer et al., 2012). Many of the virtual human capabilities have been integrated into the Virtual Human Toolkit, which is available for the R&D community (Hartholt et al., 2013).

As a UARC, USC ICT is a neutral and trusted advisor in support of validating and informing Army program requirements and acquisitions. While RIDE originated to target the STE specifically, it has been designed and developed to support a wide range of research and development needs, and is used for non-Army and non-training purposes by a range of organizations, including the Office of Naval Research (ONR).

RIDE SYSTEM

Architecture

RIDE has been designed and developed from the ground up to facilitate rapid prototyping specifically for simulation researchers and developers. This is achieved by:

1. Leveraging real-time game engine technologies that provide core capabilities, including rendering, physics simulations, and audio.
2. Abstracting away from specific game engines in order to provide simulation researchers and developers with the concepts they are most familiar with.
3. Providing a drag-and-drop development environment that offers reusable blueprints of commonly used functionalities.
4. Integrating core functionalities into a common framework in order to add combinatorial value.
5. Offering all of the above through a principled Application Programming Interface (API) that unifies all RIDE functionality in a consistent and easy to use manner.

In order to support a large ecosystem that contains many different developers, researchers, technologies, applications, and organizations, RIDE follows a layered architecture (see Figure 1).

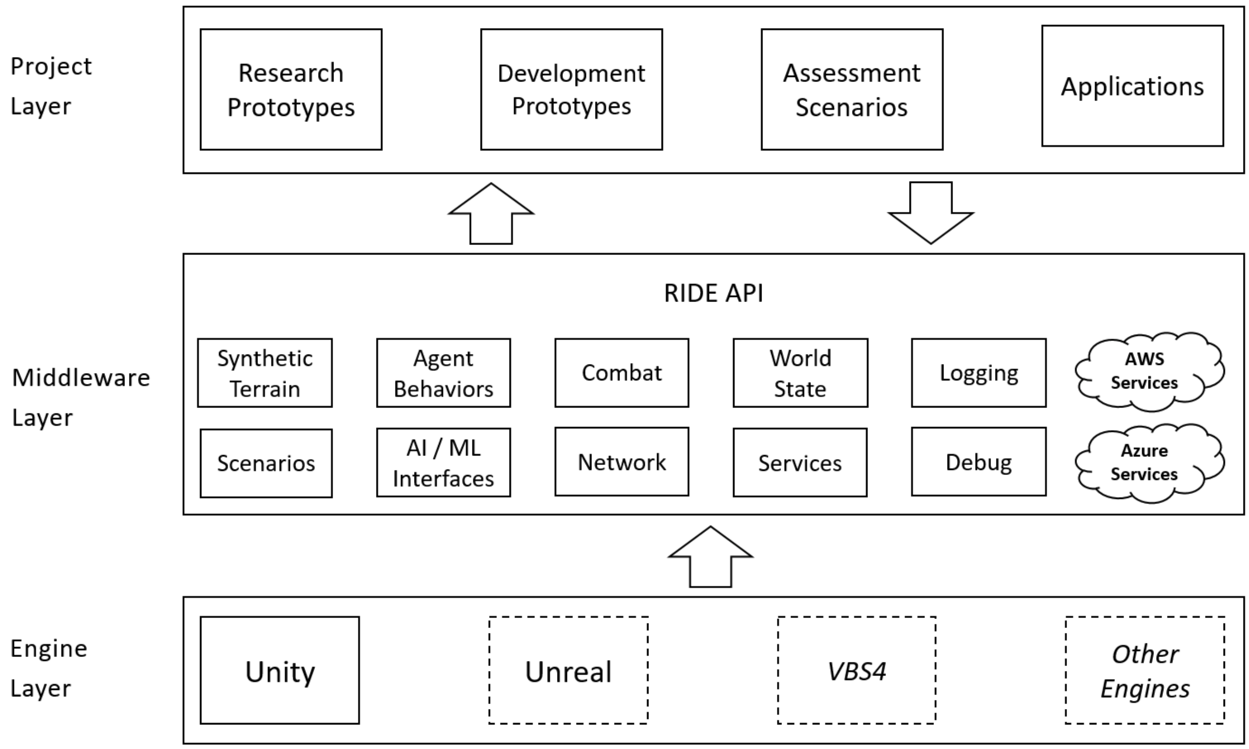

Figure 1: The RIDE architecture follows a modular and abstraction approach to leverage and integrate a range of industry and academic capabilities in a single, flexible, and easy to use R&D environment.

The Engine Layer allows RIDE to leverage robust gaming technologies that provide common capabilities, including rendering, physics, animation, pathfinding, User Interface (UI), audio, and network protocols. RIDE is agnostic to any specific game or simulation engine, which avoids vendor lock-in and enables researchers and developers to create simulation scenarios without the need to have experience with a specific engine. Currently, the main target is Unity, through the C# RIDE API, with key capabilities ported to Unreal Engine, through the C++ RIDE API.

The Middleware Layer abstracts and augments the Engine Layer with simulation-specific capabilities, including OWT, AI agent behaviors, combat system, scenario system, machine learning (ML) interfaces, and networked multi-user capabilities. RIDE is designed to provide architectural flexibility in order to facilitate R&D in support of systems design, technical performance assessment, and scalability. It contains and is extendable with permissive 3rd party assets and libraries, has native support for Amazon Web Services (AWS) and Microsoft Azure cloud web services (e.g., storage, AI services), and can itself act as a web service for any of its capabilities. The RIDE API encapsulates these available capabilities in a well-designed suite of systems and services. The API follows interface-based design, which allows for principled implementations of new technologies; new concrete implementations simply need to implement the appropriate interface in order to be used in RIDE applications. This not only provides the ability to extend RIDE with relatively little effort, it also enables multiple technical implementations of a single feature under unified interfaces, making it possible to contrast and compare, simultaneously or independently, separate approaches within a single platform, which is crucial for R&D purposes.

The Project Layer allows researchers and developers to leverage RIDE as a foundation for their own projects. RIDE provides common functionality through a drag-and-drop interface in combination with the API. This enables researchers and developers to rapidly create new scenarios as a starting point for their specific needs. The Project Layer acts as an incubation area, where new technologies and approaches can be explored safely, with mature results moving back to the Middleware Layer in order to advance RIDE and benefit all of its users.

Taken together, RIDE leverages robust industry capabilities, provides added value with each additional integrated capability, avoids reinventing the wheel, and offers researchers and developers a unified way to rapidly create, validate, and assess novel technologies and approaches, while contributing to a growing community of collaborators.

Interoperability

A key requirement for a successful R&D platform is the ability to interface with other systems, both legacy and current. RIDE follows several approaches in support of extensibility and interoperability. It supports the Distributed Interactive Simulation (DIS) messaging protocol used in many military simulations (Hofer and Loper, 1995) as well as the xAPI data format that is core to many learning and training systems[1]. In addition, it supports the ActiveMQ messaging protocol[2], in particular through the VHMsg interface, which allows communication with common virtual human and embodied conversational agent systems; see the next section for details. RIDE provides a common interface for data logging and storage. Multiple implementations allow for straightforward storage of data locally, in the cloud using AWS or Azure, or through custom means.

RESTful web services are a core pillar of the modern web, allowing vendors, producers, and consumers to interact in a flexible, secure, and real-time manner. The ability of RIDE to act as a web service provides interoperability between RIDE and other applications without the need for those applications to know about RIDE. This enables other applications (including those created years ago) to take advantage of RIDE's deep suite of capabilities with minimal work on their end. This makes RIDE highly interoperable with other systems, increasing its architectural flexibility, while maintaining efficiency and security. For example, any web browser can send and receive web requests without having to know the details of RIDE itself. It enables many simulation capabilities, especially useful for data and analysis without the need to host the RIDE platform or data locally.

Multi-Platform Support

Modern training and simulation systems require flexibility in terms of delivery mechanisms in order to allow access by end users at the point of need. RIDE has been designed and developed from the ground up to account for this need through the support of multiple target platforms. This is made possible by the multi-platform support that current game engines offer. RIDE supports development on Windows, Mac, and Linux machines, and can target applications for mobile (Android, iOS), web (WebGL), AR/VR (e.g., SteamVR, OpenXR, HoloLens), and streaming solutions (e.g., Google Stadia, Microsoft xCloud, Nvidia GeForce Now, Unity Furioos).

RIDE CAPABILITIES

One World Terrain (OWT)

Terrain data is a critical component of many simulation systems. Due to the nature of real-world terrain data, especially for military areas, sourcing existing 3D Geospatial data is a resource intensive process both in time, money, and human effort. While there are existing databases of high-quality data, such as that provided by National Geospatial Agency's (NGA's) Geospatial Repository and Data Management System (GRiD) Enterprise platform, that data is not often simulation ready or of a level of fidelity that could support ground level first person systems as would be necessary for specific use cases such as human training systems or autonomous vehicles simulators. Researchers at USC ICT have developed a fully automated pipeline that takes in high-resolution data from sources such as Small Unmanned Aerial System (sUAS) (Chen et al., 2020a) or Microsoft Bing (Chen et al., 2020b), and outputs high-resolution 3D models ready for simulation (Chen et al., 2020c). Having this foundation allows for research external to the terrain itself to take advantage of real-world data, as is the case with studying large-scale scenarios using existing data from Combat Training Centers (CTCs) such as the National Training Center (NTC) at Ft. Irwin. However, this platform can also allow for terrain research to be accelerated as well, as is in the case of the OWT Well Formed Format (WFF) data model. In an interesting research effort examining the human cognitive side of a terrain detail, when considering the resolution of textures and meshes, DEVCOM SC has explored the effect of visual fidelity on the human mind,

[1] https://xapi.com
[2] https://activemq.apache.org

aiming to find a “sweet spot” for the processing time and the resulting 3D data quality. This research has begun to define the requirements for visual fidelity, such as the mesh and texture Level of Detail, and is being led by RIDE collaborator Dr. Aaron Gardony (Gardony et al., 2021). RIDE provides an excellent foundation for testing and extending this and related work, resulting in reduced complexity of writing custom terrain loaders and testing implementations, greatly decreasing development cost and time.

Ultimately, real-world applications should be able to make use of high-quality, geospatially accurate real-world data with rich-feature attributes, and receive, collect, or process that data quickly and at very low cost. Ensuring that data is simulation ready is another important factor that underlies the use of real-world data within systems like RIDE, and being able to properly and accurately enrich that data with the necessary semantic and feature information is key to the simulation ready state of the data. Researchers at USC ICT are able to utilize the Semantic Terrain Points Labeling System Plus (STPLS+) (Chen et al., 2019) to provide this simulation information directly to the system, allowing for more effective and high-fidelity simulations than what would be possible with just mesh geometry alone. For instance, consider a simple waypoint pathing application. Utilizing a navigation mesh for path generation that is solely derived from geometry should give an accurate spatial representation of a possible path. This path wouldn’t go directly through a tree, or over a building. However, now consider enriching that pathing capability with the underlying semantic information and material classification. This path could now consider the type of surface to be traversed, adding penalties to nodes based on the ease of movement. By taking advantage of this information in a form that directly reflects smart data models, such as the OWT WFF, this information can be utilized and even updated based on dynamic events in real-time. For such a case, a region of ground material classified as dirt can apply a further penalty during a rain event as the dirt is reclassified as mud at runtime.

Many of the factors that USC ICT and collaborative researchers are exploring are to support the Army’s STE OWT program. Notably, this program’s foundational 3D data format is the Open Geospatial Consortium (OGC) standard for 3DTiles, which wraps mesh data as glTF and is excellent for storage, data transmission, and streaming due to its tiled nature and optimized compression formats. However, this data format has historically had little support on the consumption side in applications, engines, or authoring tools. RIDE has been able to provide a fertile environment for creating proxies for the OWT program data, enabling researchers to test and produce results that approximate the OWT solution, by utilizing high-resolution inset data and attribution data that mirrors the OWT WFF. As the OWT specification is updated, improved, and released, and the engine begins to provide native support for 3DTiles, the RIDE platform’s foundational systems and philosophy mean that the code reuse is high, making it simple and straightforward to switch from the proxy formats to the actual data sources without rewrites or heavy lifts resource-wise. This allows researchers to continue to progress without having to wait for government, industry, or other collaborators, as oftentimes the speed of research will outpace the authoritative solutions or programs of record (POR).

Artificial Intelligence (AI) and Machine Learning (ML)

AI and ML are transformative technologies that are of strategic importance to the US military. For the US Army in particular, they form a core area of its research and development portfolio and are an integral part of the Army Modernization Strategy. In order to reach its strategic objectives, the military needs an enabling capability to effectively and efficiently leverage AI and ML advances from the research community and industry, apply these to the military domain, and advance the state-of-the-art.

A serious bottleneck in this effort is the ability to create, collect, analyze, and validate quality data that is relevant and readily accessible, in sufficient quantities. RIDE is addressing this challenge, by combining synthesized terrain, agent behaviors, networking, and other capabilities, into a principled architecture and API, shaped directly by AI research and development needs. As such, it provides a natural environment for AI and ML experiments, augmenting the general data science hierarchy of needs (see Figure 2).

RIDE provides a particular focus on enabling researchers and developers to author, test, validate, train, execute, and integrate agent behaviors. To this end, initial agent behaviors can be scripted with either state machines or behavior trees in order to bootstrap experiments. Provided behaviors include movement, attacking, and formations, tied to dedicated combat and health systems. Alternatively, networked multiplayer allows multiple teams of users to control avatars in a common synthetic environment, each from their own location and type of hardware (e.g., desktop or mobile device).

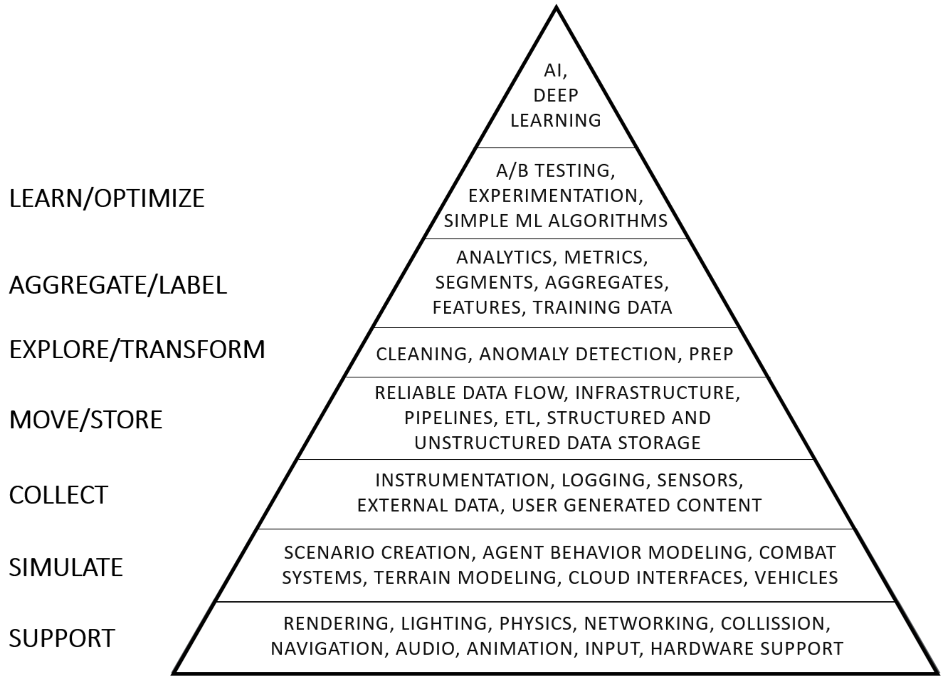

Figure 2: The original data science hierarchy of needs by (Rogati, 2017), extended with two additional layers: Simulate and Support, which are provided by RIDE.

Both human avatars and bots can be mixed and matched in a single environment, from which data can be gathered to train ML models. RIDE has been used to create and run models in TenserFlow, PyTorch and custom solutions. Results can either directly be run in RIDE through the inference model, by providing individual commands to agents, or by generating behavioral code that leverages atomic actions. See (Hartholt et al. 2021) for more details.

Virtual Humans

Virtual humans, or embodied conversational agents, are interactive, digital, embodied characters that perceive real humans and respond appropriately, both verbally and nonverbally. They act as social interface agents that add a social component to the environments in which they are embedded. They provide a standardized experience across users and can be omnipresent and indefatigable in their roles. Virtual humans have been shown to improve user's perception of their environment (Johnson et al., 2000), increase interaction time (Lane et al., 2011), and improve learning outcomes (Schroeder et al, 2013).

Virtual humans allow simulation to go beyond maneuvering and fires to include first-person perspective and conversational-based scenarios (e.g., tactical questioning, interactive After-Action Review (AAR), negotiation, etc.). Effective use requires automated or semi-automated agents that can take on the role of friendly forces, allies, enemy forces, or non-combatants. In order for these to be effective, these interactive characters need to behave in ways similar to how real humans behave. Their behaviors need to include verbal and nonverbal communication capabilities that will allow for more sophisticated scenarios (e.g., tactical questioning, mission debriefing, etc.).

While much progress has been made in creating effective virtual humans, simulating all aspects of up-close human appearance and behavior remains a huge challenge in general and within military training in particular. Creating effective military virtual role-players requires 1) a deep and shared understanding of all the interconnected technologies and approaches that make up interactive AI characters, 2) the ability to rapidly prototype a range of different types of interactive AI characters, and 3) an understanding of which factors of an interactive AI character affect the efficiency and effectiveness of training and how. RIDE forms the ideal testbed to further explore these challenges and inform trade-off decisions regarding quality, speed, and cost.

ICT has widely been recognized as one of the leaders in virtual human research and development, including basic research in cognitive architectures (Rosenbloom et al., 2016), audiovisual sensing (Scherer et al., 2012), and character animation simulation (Shapiro, 2011), as well as applied prototypes for leadership development (Hays et al., 2012), information dissemination (Rizzo et al., 2011), job interview training (Hartholt et al., 2019), and life-long learning (Swartout et al., 2016). Our approach is highly interdisciplinary with a strong focus on integrating both theory and technology into common frameworks (Hartholt et al., 2009; DeVault et al., 2014). This work has resulted in the Virtual Human Toolkit, a collection of modules, tools, and libraries designed to aid and support researchers and developers with the creation of virtual human conversational characters (Hartholt et. al, 2013). The Toolkit is freely available for academic and government purpose use. Recent efforts have resulted in the ability to develop virtual humans for a range of hardware platforms, including web, mobile, AR and VR (Hartholt et al., 2020).

Ongoing work focuses on integrating these efforts with RIDE, revolving around three main areas: architecture, capabilities, and content. The Virtual Human Toolkit architecture is modular, using a combination of message passing and direct data streams to facilitate inter-module communication, augmented with a microservices architecture. RIDE has integrated its messaging protocol, called VHMsg, that's built on top of ActiveMQ. This allows researchers and developers to interface with any of the main Toolkit modules and the capabilities they represent, including natural language processing and nonverbal behavior generation. In addition, by leveraging RIDE's dedicated web services system, commodity AI-related services are provided, including audio-visual sensing, speech recognition, natural language processing, and text-to-speech generation. Nonverbal behavior realization (e.g., lip-sync, facial expressions, conversational gestures) is directly integrated within RIDE. All these capabilities together can function on dedicated characters that are able to observe real human users, create the agent's communicative intent, and realize that intent verbally and nonverbally, synchronized in real-time.

RIDE USE CASES

Rapid Scenario Creation

RIDE's core capabilities for the use of geo-specific terrain, networked multiplayer control of virtual soldiers, and deployment of AI forces enables rapid scenario creation, with applications in training and mission rehearsal. With minimal developer effort, commanders and instructors can design and execute exercises where multiple remote participants control their virtual avatars in combat with AI-controlled enemy units in varied physical environments and force composition. To tailor the experience to specific learning objectives and experiences, RIDE includes a robust framework for authoring events for inclusion in a Master Scenario Events List (MSEL). In the simple case, RIDE's scenario system can inject varied events into multiplayer sessions at specific times, e.g. to cause an explosion of a key civilian infrastructure facility at the 10-minute mark of a scenario, leading to panic among the AI-controlled civilian population. As well, RIDE enables the triggering of scenario events based on recognized behaviors of the human participants (Feng & Gordon, 2020).

While the development of these capabilities was motivated by their application toward mission rehearsal and virtual training, the ability to rapidly create interactive scenarios in geo-specific locations in RIDE has also enabled the development of other sorts of virtual experiences, most notably for the purpose of site familiarization. In one recent effort, our team crafted an interactive tour of an urban environment that exists, in the real world, at the NTC at Ft. Irwin, California, modeled as a small city that might exist in Iraq. In the virtual tour, participants control an avatar as they follow a tour guide around the city, learning about the buildings and neighborhoods with produced audio commentary and virtual fly-over videos. In a second recent effort, our team created a multiplayer scavenger hunt game using terrain data of a university campus, where players race their avatars through campus to find the location of different buildings. Both of these recent efforts offer users a fun way to familiarize themselves with real-world places, so that they are better able to execute tasks when they physically arrive at these locations.

Operational Systems, Testing, and Evaluation

The terrain loading and toolset within RIDE is currently being used to support a number of external objectives, notably the work by Dr. Mark Dennison at the Army Research Laboratory - West (ARL-W) for Project AURORA supporting the US Army's Network Cross Functional Team (CFT) (Dennison & Trout, 2020). AURORA was designed to enable researchers to study the visualization, analysis, and actuation of battlefield data across multiple domains by collaborative teams of humans and intelligent agents that are both co-located and distributed over geographical space. Dr. Dennison and his team had been leveraging fictional terrain as well as plugins from the Unity Asset Store in order to create their prototype system. Needing high-resolution real-world data with a high level of geospatial accuracy and fidelity was critical to advancing their system and enabling demonstration and participation in the Army's Project Convergence event. This is an example of how RIDE is not necessarily required in whole to lower the barrier to entry into simulation, but that modular components of RIDE can lower the barrier to entry to specific areas of simulation. In the AURORA use case, real-world terrain was already integrated, however the effort required to source and utilize new datasets of geospatially accurate real-world terrain was still hugely advantageous. RIDE provided an immediate solution, allowing the AURORA project team to focus their efforts and attention on the Common Operating Picture (COP) instead of wasting development cycles attempting to find, package, and prepare terrain for deployment within their application.

Just as RIDE has been used internally at USC ICT to integrate disparate but related research efforts, the DEVCOM SC SED STTC Synthetic Natural Environment program has also been seeking ways to integrate and evaluate research, as well as beginning to provide a high-fidelity laboratory environment for testing and evaluation. For this effort, USC ICT, in coordination with Unity, began to create a number of foundational capabilities and systems that would enable replication of the Army's Infantry Battle Drills. Having a rich scenario with a desired outcome provides an excellent contextual wrapper for integration of disparate systems and research efforts. One example would be the React to Ambush Battle Drill. In this particular drill there are two teams, a friendly "Blue Force'' (BLUFOR) and an enemy "Opposing Force" (OPFOR) team. The OPFOR team has set up an ambush and selected a killzone area. When the BLUFOR patrol enters the ambush location, the OPFOR begins the assault. BLUFOR must react accordingly with units inside the killzone throwing smoke and assaulting directly towards the OPFOR squad, and BLUFOR units outside the killzone taking up cover positions and applying suppressive fire.

Though this scenario seems straightforward, there are a number of layers that can continually be iterated upon and improved to optimize the simulation fidelity. For instance, the quality of the AI behaviors is important, and research into behavior trees and ambush tactics has been applied to increase the realism of these tactics. Digging deeper, the take cover action is critical to the BLUFOR response, and by providing the real-world terrain as geometry, a fair result can be expected; however, by leveraging the rich feature attribution and semantic data as provided by the STPLS+ pipeline, the autonomous agents can also begin to assign quality of cover values to the geometry. A concrete block will be much more effective as a cover position than a wooden fence, and with access to the material classification, an autonomous agent can make a judgement call to maneuver more effectively, and make optimal use of cover and concealment techniques. Though in the current iteration of the Battle Drill this is being achieved through a behavior tree AI implementation, additional research is currently being conducted that can apply some of this technique to ML agents as well, as described in the section below on ML training on OWT. This is another example of how these disparate research efforts are able to be consolidated into a single codebase, tested against a common foundation, and swapped as necessary to support different use cases, research efforts, and training objectives.

ML Training on OWT

As discussed, RIDE supports a range of behavior representations, including artificial neural network-based behavior policies. Such a capability provides opportunities to train robust neural behavior policies that potentially generalize better than traditional representations. Furthermore, RIDE offers such policies to utilize in simulation models.

Previously, we showed the potential of a neural network model trained to take cover in a simple block environment in RIDE (Ustun et al. 2020). Here, we extend that work and train agents who learn to take cover in geo-specific terrains within RIDE. Within this context, "taking cover" is defined as having every point on the agent occluded from an enemy. In other words, the goal is to have an object in between the agent and the enemy such that the agent is no longer visible. An agent is also considered in cover if they are sufficiently far away from the opponent, because it would be impractical for an opponent to shoot the agent from that distance. However, in the terrains that have been used, it's almost always faster to take cover by hiding behind an object.

In this effort, agents were trained on Drone-captured geo-specific terrains of the USC campus and Catalina Island in RIDE. Coordinates of the opponent along with Unity's rays comprise the observation space. For each observation ray cast, we check whether the ray hit the opponent, whether it hit a part of the terrain, and the distance the ray travelled. The observation rays are depicted in Figure 3. We leveraged Deep Reinforcement Learning, more specifically the Proximal Policy Optimization (PPO) algorithm (Schulman et al. 2017) from within the Unity ML-Agents framework (Juliani et al. 2018), to learn behavior policies for taking cover.

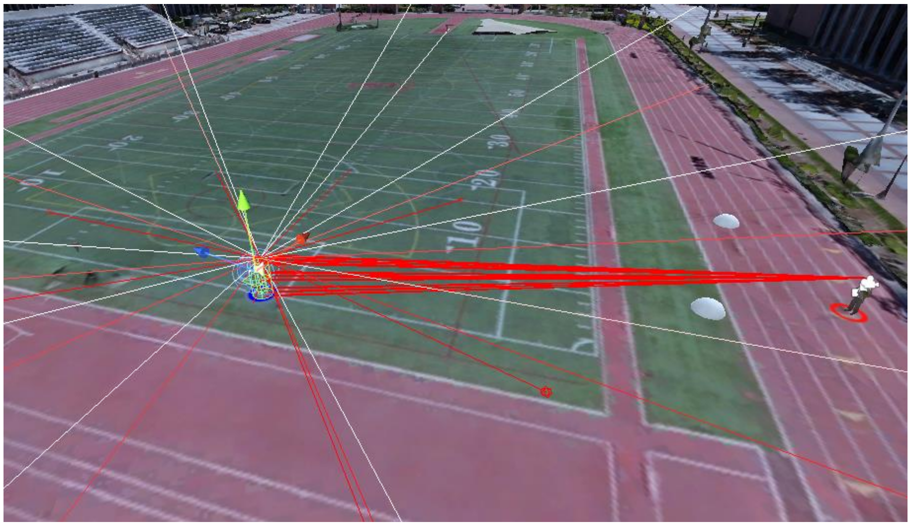

Figure 3. The rays that make up the agent's (blue) observations. Also seen are the rays testing for occlusions between the agent and the enemy (red).

Our experiments showed that the trained agents can achieve human, or near human, performance in areas they are trained in. Such performance is roughly defined as consistently having a finish time of lower than 1000 timesteps, out of a possible 5000. The median performance at the end of training is around 500, with the agent successfully finding cover roughly 97% of the time. In the test areas, agents attain slightly worse performance, with a median timesteps to take cover between 1200 to 1500 depending on the terrain. However, they frequently get stuck with little ability for error correction, so they have a success rate between 60-70%, again, depending on terrain. Thus, further progress is needed to increase the robustness on the test sets. Another experiment focused on training pretrained agents on different terrains. The most significant finding was that training on a new terrain would increase its performance on the terrain it was initially trained on (see Figure 4). In other words, taking an agent trained on USC, then training it on Catalina would increase its performance on USC.

The results are promising, and the models are considered performing well. However, in order for full deployment in RIDE, the models will need to get closer to human-level performance. Potential research avenues that are being explored to achieve this goal include: (1) Training on simpler environments to see where exactly the agents are failing to generalize; and (2) Leveraging hierarchical learning to train a controller that moves to a goal, and a meta-controller that chooses goals to move the agent into cover.

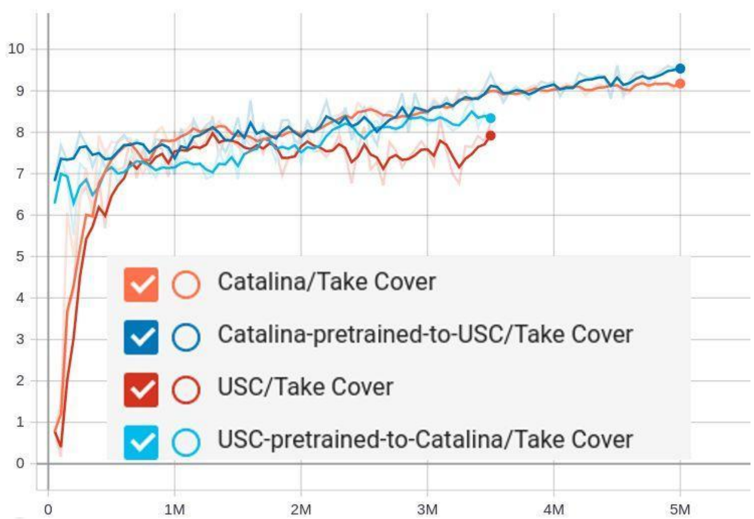

Figure 4. Reward curves of agents trained on different terrains. The y-axis is the final reward; 9 or higher resembles human-level performance. USC and Catalina were trained in their environments. Catalina-pretrained-to-USC was initialized from the Catalina model, then trained on the USC terrain, and vice versa.

Wargaming

Wargaming has seen a long tradition, with (Perla & McGrady, 2011) claiming that "When it works, wargaming can appear almost magical in its power to inform and instruct; when it doesn't work, it can appear almost childish in its oversimplifications and abstractions." Providing technical wargaming contexts with the proper level of simulation remains challenging, including not having a comprehensive adjudication of armed conflicts (Col & Caffrey, 2000), not enough emphasis on narrative storytelling (Perla & McGrady, 2011), and participants being able to "game the game" rather than focus on learning objectives (Frank, 2012). We start addressing these issues by creating the context for a virtual, historical battleground, leveraging RIDE.

The specific goal is to create a testbed for AI researchers to develop models to create Courses of Actions (CoA), which can be compared against historic CoA. The Invasion of Sicily was chosen due to its combined allied forces, rich information, and combination of land, air, and sea forces (Zuehlke, 2009). RIDE offers the core simulation framework, adding historic units, sea and air units, detailed simulation of soldier fatigue, and multi-echelon support.

DISCUSSION

We have presented the background, architecture, capabilities, and use cases of RIDE, highlighting how this integrated R&D platform provides military stakeholders with a powerful capability to explore the art of the possible, contrast and compare trade-offs, and inform program and acquisition requirements. RIDE is built from the ground up to facilitate rapid prototyping in military-specific contexts, leveraging best-of-breed technologies from both academia and industry. RIDE is an open platform with a specific focus on community building and the democratization of integrated military simulation R&D, through easy access to capabilities, onboarding tutorials, a myriad of examples, detailed documentation, system operability, and integration of technologies from all stakeholders, resulting in a unique approach to lowering barriers of entry.

RIDE offers an easy to use suite of integrated capabilities, including OWT tools and data, AI and ML interfaces, multi-platform support, NPC behaviors, logging, cloud services, and virtual humans. It has been used by over 30 organizations, from the Army and Navy to industry and academia, including the STE CFT, DEVCOM SC SED STTC, ONR, ARL, University of Central Florida, SoarTech, Engineering R&D Center (ERDC), and Unity. It forms the foundation for proofing out the next generation OWT format, identifying hybrid computing and scalability best practices, leveraging ML platforms to model AI allies, adversaries and civilians, providing intelligent tutoring and AARs, and studying the visualization, analysis, and actuation of battlefield data across multiple domains.

This work has several limitations. First, RIDE is a relatively young framework, given its two years of active development, resulting in varying maturity levels of its systems. However, this has allowed us to focus specifically on modern software architectures and approaches, with today's and tomorrow's military needs in mind. Work on RIDE is rapidly progressing, by incorporating our previous work, integrating robust industry capabilities, and leveraging a wide network of collaborators spanning industry, academia, and government. Second, RIDE leverages existing game and simulation engines and therefore inherits their strengths and weaknesses, both technically and organizationally. The engine-agnostic design of RIDE allows it to compare these strengths and weaknesses, and even to mitigate the latter, depending on the extendibility of the underlying engine. Third, our focus has been on flexible, integrated, principled systems and APIs rather than visual fidelity. We are strong advocates for open source military art assets and related models in order to provide our warfighters with the simulation fidelity they deserve.

Future work is focused on growing RIDE, in terms of capabilities, projects, and community. Specific new features on the roadmap include multi-echelon support, Multi-Domain Operations support, enhanced C++ API in addition to the C# API, rapid soldier avatars, streaming solutions, and network scalability. As a result of these and ongoing efforts, RIDE has moved beyond simulated training, to include integrated live and synthetic training, Planning, Mission Command and Operations, and advanced R&D leveraging AI and machine learning.

To maintain its strategic advantage in a world that's rapidly changing due to technical advances and the rise of near-peer adversaries, the US military needs an R&D platform and approach that facilitates rapid prototyping, flexible exploration, easy integration of stakeholders and technologies, and standardized assessment capabilities, in support of efficient decision making and technical progress. These are inherently complex challenges. We believe that RIDE provides a blueprint to manage this complexity by 1) connecting people and technologies, 2) making trade-offs explicit, and 3) showing the path of the future.

ACKNOWLEDGEMENTS

We would like to thank all of our partners and collaborators at USC ICT, DEVCOM SC SED STTC, ONR, and others, who have been instrumental in making RIDE a success. Part of the efforts depicted were sponsored by the US Army under contract number W911NF-14-D-0005. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

REFERENCES

L. T. Col and M. Caffrey, (2000), “Toward a History-Based Doctrine for Wargaming.”, AIR AND SPACE POWER JOURNAL MAXWELL AFB AL.

Chen, M., Feng, A., McCullough, K., Prasad, P. B., McAlinden, R., Soibelman, L., & Enloe, M. (2019). Fully Automated Photogrammetric Data Segmentation and Object Information Extraction Approach for Creating Simulation Terrain. Interservice/Industry Training, Simulation and Education Conference, November 29-December 3, 2019, in Orlando, FL.

Chen, M., Feng, A., McCullough, K., Prasad, P. B., McAlinden, R., & Soibelman, L. (2020a). 3D Photogrammetry Point Cloud Segmentation Using a Model Ensembling Framework. *Journal of Computing in Civil Engineering*, *34*(6), 04020048. https://doi.org/10.1061/(asce)cp.1943-5487.0000929

Chen, M., Feng, A., McCullough, K., Prasad, P. B., McAlinden, R., & Soibelman, L. (2020b). Semantic Segmentation and Data Fusion of Microsoft Bing 3D Cities and Small UAV-based Photogrammetric Data. Interservice/Industry Training, Simulation and Education Conference, Nov 30-Dec 4, 2020, in Orlando, FL.

Chen, M., Feng, A., McAlinden, R., & Soibelman, L. (2020c). Photogrammetric Point Cloud Segmentation and Object Information Extraction for Creating Virtual Environments and Simulations. *Journal of Management in Engineering*, *36*(2), 04019046. https://doi.org/10.1061/(ASCE)ME.1943-5479.0000737

DeVault, D., Artstein, R., Benn, G., Dey, T., Fast, E., Gainer, A., Georgila, K., Gratch, J., Hartholt, A., Lhommet, M. and Lucas, G., (2014). SimSensei Kiosk: A virtual human interviewer for healthcare decision support. In Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems.

Denisson, M., Trout, T., (2020). The Accelerated User Reasoning for Operations, Research, and Analysis (AURORA) Cross-Reality Common Operating Picture. (Publication No. ARL-TR-8942). https://apps.dtic.mil/sti/pdfs/AD1096359.pdf

Feng, A., and Gordon, A. (2020) A Framework for Action Detection in Virtual Training Simulations using Synthetic Training Data. I/ITSEC, Nov 30-Dec 4, 2020, in Orlando, FL.

Frank, A. (2012). Gaming the game: A study of the gamer mode in educational wargaming. Simulation & Gaming, 43(1), 118-132.

Gardony, A. L., Hendel, D. D., & Brunyé, T. T. (2021). Identifying optimal graphical level of detail to support orienting with 3D geo-visualizations. *Spatial Cognition & Computation*, 1–26.

Hartholt, A., Gratch, J., & Weiss, L. (2009, September). At the virtual frontier: Introducing Gunslinger, a multi-character, mixed-reality, story-driven experience. In International Workshop on IVA. Berlin, Heidelberg.

Hartholt, Arno, David Traum, Stacy C. Marsella, Ari Shapiro, Giota Stratou, Anton Leuski, Louis-Philippe Morency, and Jonathan Gratch. (2013) All together now: Introducing the Virtual Human Toolkit. In International Workshop on Intelligent Virtual Agents, pp. 368-381. Springer, Berlin, Heidelberg.

Hartholt, A., Mozgai, S., & Rizzo, A. S. (2019, July). Virtual job interviewing practice for high-anxiety populations. In Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents (pp. 238-240).

Hartholt, A., Fast, E., Reilly, A., Whitcup, W., Liewer, M., & Mozgai, S. (2020). Multi-Platform Expansion of the Virtual Human Toolkit: Ubiquitous Conversational Agents. International Journal of Semantic Computing.

Hartholt, A., McCullough, K., Fast, E., Reilly, A., Leeds, A., Mozgai, S., Ustun, V., & Gordon, A. (2021) Introducing RIDE: Lowering the Barrier of Entry to Simulation and Training through the Rapid Integration & Development Environment. Simulation Interoperability Standards Organization, Virtual Simulation Innovation Workshop (SIW), February 8-12, 2021, online.

Hays, M. J., Campbell, J. C., Trimmer, M. A., Poore, J. C., Webb, A. K., & King, T. K. (2012). Can role-play with virtual humans teach interpersonal skills?. Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL, December 3-6, 2012.

Hofer, Ronald C., and Margaret L. Loper. (1995) DIS today. Proceedings of the IEEE 83.8 (1995): 1124-1137.

Johnson, W. L., Rickel, J. W., & Lester, J. C. (2000). Animated pedagogical agents: Face-to-face interaction in interactive learning environments. International Journal of Artificial intelligence in education, 11(1), 47-78.

Juliani, A., Berges, V. P., Teng, E., Cohen, A., Harper, J., Elion, C., ... & Lange, D. (2018). Unity: A general platform for intelligent agents. arXiv preprint arXiv:1809.02627.
Lane, H. C., Noren, D., Auerbach, D., Birch, M., & Swartout, W. (2011, June). Intelligent tutoring goes to the museum in the big city: A pedagogical agent for informal science education. In International Conference on Artificial Intelligence in Education (pp. 155-162). Springer, Berlin, Heidelberg.
Mozgai, S., Hartholt, A., Leeds, A., & Rizzo, A. S. (2020). Iterative Participatory Design for VRET Domain Transfer: From Combat Exposure to Military Sexual Trauma". In Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (pp. 1-8).
P. P. Perla and E. D. McGrady, "Why Wargaming Works," 2011.
Rizzo, A., Lange, B., Buckwalter, J. G., Forbell, E., Kim, J., Sagae, K., ... & Kenny, P. (2011). SimCoach: an intelligent virtual human system for providing healthcare information and support.
Rizzo, A., Hartholt, A., Grimani, M., Leeds, A., & Liewer, M. (2014). Virtual reality exposure therapy for combat-related posttraumatic stress disorder. Computer, 47(7), 31-37.
Rogati, Monica (2017). The AI Hierarchy of Needs, https://hackernoon.com/the-ai-hierarchy-of-needs-18f111fcc007, accessed April 23 2021
Rosenbloom, P. S., Demski, A., & Ustun, V. (2016). The Sigma cognitive architecture and system: Towards functionally elegant grand unification. Journal of Artificial General Intelligence, 7(1), 1-103.
Scherer, S., Marsella, S., Stratou, G., Xu, Y., Morbini, F., Egan, A. and Morency, L.P., (2012). Perception markup language: Towards a standardized representation of perceived nonverbal behaviors. In International Conference on Intelligent Virtual Agents (pp. 455-463). Springer, Berlin, Heidelberg.
Schroeder, N. L., Adesope, O. O., & Gilbert, R. B. (2013). How effective are pedagogical agents for learning? A meta-analytic review. Journal of Educational Computing Research, 49(1), 1-39.
Schulman J., Wolski F.,Dhariwal F., Radford A., and Klimov O. 2017. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347, 2017.
Shapiro, A. (2011, November). Building a character animation system. In International conference on motion in games (pp. 98-109). Springer, Berlin, Heidelberg.
Swartout, W. R., Nye, B. D., Hartholt, A., Reilly, A., Graesser, A. C., VanLehn, K., & Rosenberg, M. (2016, March). Designing a personal assistant for life-long learning (PAL3). In The 29th International Flairs Conference.
Ustun, V., R. Kumar, A. Reilly, S. Sajjadi, and A. Miller. 2020. "Adaptive Synthetic Characters for Military Training". Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC).
Wansbury, T., Hart, J., Gordon, A., and Wilkinson, J. (2010) UrbanSim: Training Adaptable Leaders in the Art of Battle Command. Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL, November 29 - December 2, 2010.
Zuehlke, M. (2009). Operation Husky: The Canadian Invasion of Sicily, July 10 August 7, 1943. D & M Publishers.