

Developing xAPI Enabled Virtual Advisors

Joel Scharlat
IVEA Consulting
Ashburn, VA
jscharlat@iveaconsulting.com

ABSTRACT

The Experience API (xAPI) has the potential to collect all manner of data associated with the experience of a learning event. This experience includes not only the traditional content of the learning objective, but also can include information about the learning environment, the learner (physiological, social (including other learners), psychological, etc.) and potentially about the instructor, delivery method and other environmental factors. As implementation of the xAPI grows, the amount of data available increases. This data, when provided with enough breadth and depth can be useful in understanding what comprises an ideal learning experience. That experience will be unique to each learner and can even vary among different instances of learning for a single learner. Traditional learning environments (i.e. one teacher to many students) cannot possibly provide the highly personalized learning experience that can otherwise be enabled by the xAPI due to physical limitations faced by a single instructor. A natural evolution of learning spaces that can take advantage of the personalization information provided by xAPI are immersive virtual environments (IVEs) where virtual advisors and instructors can be utilized. While there has been some discussion on the limitations of learning in virtual worlds, they are less related to content delivery than they are about comfort and cultural acceptance. That notwithstanding, IVEs (virtual worlds) provide an ideal location/platform to provide highly personalized learning experiences. This paper discusses how avatars can be created and personalized based on a learners likes and dislikes and how, with certain design criteria, can be created to maximize factors such as trust and influence. This paper is a first step in defining what data could be useful when collected for the creation of a personalized learning experience, as well as defining a process for the design and the deployment of an avatar based virtual advisor.

ABOUT THE AUTHOR

Mr. Joel Scharlat is the President and owner of IVEA Consulting, a Service Disabled, Veteran Owned Small Business (SDVOSB) that focuses on the operational use of metaverse technologies (virtual worlds, augmented reality, mirror worlds and lifelogging). He is active in the Marine Corps Reserves after 13 years on Active Duty in the Marine Corps where he flew combat missions as a helicopter pilot then more recently focused on strategy, policy and requirements development for tactical networking equipment. While in the Marine Corps, he attended the Naval Postgraduate School (Joint C4I) where he wrote his thesis on conducting Information Operations in Immersive Virtual Environments. He also served as an Associate Fellow with the Chief of Naval Operations (CNO) Strategic Studies Group studying how Naval Forces would fight in Cyberspace in the 2030 timeframe.

Developing xAPI Enabled Virtual Advisors

Joel Scharlat
IVEA Consulting
Ashburn, VA
jscharlat@iveaconsulting.com

INTRODUCTION

Arguably, the greatest contribution of the release of the Experience API (Application Programming Interface) or xAPI specification is the amount of data that will be available to the training and learning community. This increase in data is a result of the flexibility of xAPI to collect data about more than just traditional learning experiences. Experience API also allows content creators and managers to track formal and informal learning data that describes the learning experience of the user and can come from a larger pool of learning platforms to include mobile devices, simulations (e.g., 3D virtual worlds) and serious games. Additional flexibility is gained because the xAPI, similar to activity streams, allows for the collection and storage of data in a Learning Record Store (LRS) or data management system. This increase in the amount, and types, of data collected can create a paradigm shift in learning and training as content creators and managers take advantage of larger repositories of data with creative approaches to enhance learning experiences. This paper explores one potential use of this explosion of personalized data: avatar-based virtual advisors hosted within immersive virtual environments (IVEs). These advisors would be customized and informed by the myriad data points available as the xAPI specification sees greater rates of adoption in learning systems. Subsequent sections will introduce both the xAPI specification and IVEs and discuss the significance of IVEs to the learning community. Effective use of virtual advisors will hinge on proper design informed by specific data. To that end, design criteria will be presented within an introductory process defining how a virtual advisor can operate. Data collection considerations with respect to the xAPI will be discussed and the entire concept will be further highlighted with an example. Finally, a short discussion on a few next steps that can be accomplished to further this concept will conclude the paper.

Experience Application Programming Interface Specification

According to the Advanced Distributed Learning (ADL) initiative, the xAPI specification is a component of the larger Training and Learning Architecture (TLA) being developed by the same organization. The specification was released as version 1.0.0 on April 26th, 2013 as the culmination of over twelve months of work by a representative group of developers, users, stakeholders and vendors within the learning community. A collaboration between industry, academia and government, the xAPI specification is essentially the next generation activity beyond the long standing Shareable Content Object Reference Model (SCORM).

First released in 2000, SCORM was developed in response to the need for common specifications and standards for e-learning. It is a technical specification developed to provide a common “language” for e-learning platforms to use when communicating with each other for interoperability. The SCORM specification has undergone numerous revisions, yet, despite the updates, there are still limitations to SCORM that allow for *simple* and *standard* approaches to using SCORM on mobile devices, with simulations, in virtual worlds, and in complex integrations across multiple environments. The xAPI specification was developed to address many of the shortfalls; most relevant to this paper is the increase in flexibility that the xAPI specification provides beyond the SCORM specification. With the implementation of the xAPI specification, the deployment and tracking of e-learning activity is no longer constrained to the web browser on a computer. Learning activities can be tracked when they occur on mobile devices, within simulations (including 3D virtual worlds) or even when they are not conducted on a digital device (like reading a research journal or attending a conference).

The xAPI is designed to enable more robust learning environments by removing the rigid SCORM-to-LMS communication and allowing data to be tracked any time learning occurs, regardless of where it occurs. The xAPI allows broader levels of tracking across a wide variety of environments, and potentially decouples the data from the LMS to provide more complete and interoperable views of data across numerous environments. Overall, this means that data tracked about a learning activity is no longer constrained to only that data retrieved from specific courses or

dedicated computers and ultimately acknowledges that learning occurs independent of location or platform. This tracking is made possible through the introduction of a format based upon activity streams.

Activity streams are an open format specification that allow interaction with other web services implementing the open format activity stream specification and are used to describe an activity in the format of <actor, verb, object>¹. An activity stream describing research activities on the topic of xAPI, for example, would look something like <Bob read 'Experience API Version 1.0.0 Released! '>. This information is then stored in a LRS and can be accessed by other systems capable of reading activity streams. Flexibility is increased inside of a traditional learning environment because much higher fidelity data can be collected in a context that makes sense and tells a more complete story about how and what a learner is learning. Outside of the traditional learning environment (often called informal learning environments), flexibility is introduced as content creators and managers as well as third party developers will have access to information about the learner that can be used for and by other activity stream enabled environments. While learning data from SCORM enabled learning environments could only record formal learning data contained within the specific module, the xAPI enables more platforms to collect both formal and informal learning data including mobile platforms, social network environments and even IVEs. This cross-platform approach makes data available in a user-centric way and frees data about and for the learner to be used in creative ways.

Immersive Virtual Environments

In order to properly define an IVE, it is important to understand it as a subset of technologies that define the larger metaverse. The term metaverse was first coined in 1992 by author Neal Stephenson in the book *Snow Crash* and was described as a 3D virtual space accessed through glasses worn by a user. That vision was later broadened by a group of futurists creating the Metaverse Roadmap - a document outlining a path to the future as characterized by Stephenson's work. In the roadmap, the metaverse is defined as "...the convergence of 1) virtually enhanced physical reality and 2) physically persistent virtual space. It is a fusion of both, while allowing users to experience it as either." (Smart, Cascio, & Paffendorf, 2007). Figure 1 depicts the two continuum used in defining the framework for metaverse technologies. It shows four quadrants where technologies and applications exist to enable the metaverse; all are equally beneficial in realizing the immersive experience. For example, lifelogging technologies provide data about a person through continual updates on location and activities. Augmented reality brings the virtual world to the physical world through technologies that overlay digital information on top of the physical world. Mirror worlds provide physical world information to digital worlds and virtual worlds are generally described as digital representations of physical worlds. While virtual advisors can be applied across technologies found in the augmented reality, mirror worlds, or virtual worlds quadrants, the primary focus of this paper will be on technologies that reside in the virtual worlds quadrant. An IVE can best be described as an environment augmented by or created with digital information such that a user experiences the environment as if it were real. What makes the experience immersive is the sense of "being there". For the remainder of this paper, it should be noted that the terms IVE and 3D virtual worlds are synonymous, especially when viewed as platforms for learning environments.

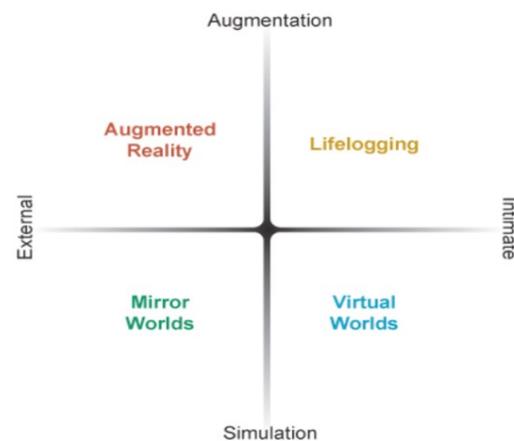


Figure 1. Graph of continuum used to define the metaverse. IVEs reside in the bottom right quadrant where technologies are defined as being user centric (intimate) and simulation based. (Taken from Metaverse Roadmap (Smart, Cascio, & Paffendorf, 2007))

¹ The JavaScript Open Notation (JSON) definition of an activity stream includes a 'target' after the 'object' noted in the xAPI activity stream which can be found at <http://www.activitystrea.ms/specs/json/1.0> and while the xAPI specification is generally depicted as written above, more complex statements can be written.

EVOLUTION OF LEARNING ENVIRONMENTS

The majority of formal learning environments are based on a one teacher to many student ratio despite research highlighting the value of a one-to-one, teacher-to-student ratio (Bloom, 1984). Except in limited cases, a student rarely has the sole attention of a single instructor. There are benefits to this approach, mainly that a standardized curriculum is provided to the maximum number of students possible. However, there are downsides as well. Although the instructor is teaching the same lesson plan to all of the students, the delivery method is heavily influenced by the instructor. Instructors are defined by their life experiences, their likes and dislikes, and through their traits and constraints. These qualities make them uniquely who they are and help define their teaching style. The same fundamental principle applies to the students - they too have unique traits, qualities and experiences, all of which shape their ability to learn. The more a student's learning style matches up with an instructor's teaching style, the better a student learns ("Knowledge of Student Characteristics," n.d.). In a physical classroom setting, an instructor cannot be everything to everyone, so gaps exist in this one-to-many model. E-learning can make strides in closing the gap, but the model is still essentially one-to-many (albeit, a much larger "many"). In the case of e-learning, "one" refers to a pre-made lesson plan designed by an instructor (with a certain teaching style) and encompassed in a rigid series of slides, frames and modules. The current e-learning approach has the ability to incorporate different teaching styles beyond the content creator's style in an attempt to connect with more learners and potentially moving closer to a one-teacher-to-one-student model. By incorporating IVEs into the picture, learning environments can move even closer to, if not achieve, a true one-to-one environment; one where each student can essentially bring their personalized avatar-based instructor to the learning environment.

The personalized avatar-based instructor model can be more easily developed within an IVE and would require extensive knowledge about its user, the learner. Some of this data can be gathered through the use of the xAPI and the following sections describe an avatar-based virtual advisor that can take on multiple roles for the user. The modality, or the mode of interaction with the user, doesn't necessarily matter except that ideally the form the virtual advisor takes would be most appropriate for a given situation, which can be partially defined by location data and other data providing context. For example, the virtual advisor could be text-based, such as a pop-up message on a computer or phone when conditions require minimal noise intervention or when resources aren't available for any other modality. Similarly, the virtual advisor could interact with the user through auditory means, such as voice feedback or visually as exemplified by avatars in IVEs. In this discussion, the focus of the virtual advisor will be on providing instruction and help (tutorship) during periods of formal learning and training and so the remainder of this paper will address avatars in IVEs as the preferred modality.

Next Generation Learning Environments

Before describing an avatar as a virtual advisor, it would be beneficial to discuss IVEs as learning environments as that is where an avatar currently has the most value. Education within IVEs is not a new concept. For years, universities, businesses and even government agencies have designed and maintained entire campuses that rival their physical world counterparts ("A serious Second Life for business schools," n.d.; "Second Life Education Directory," 2013). These campuses and various "school houses" have been used to conduct distance learning and provide training and education opportunities. IVEs can be used to create new and exciting opportunities (and places and experiences as well), yet the majority of classrooms and learning spaces developed in IVEs very closely (accurately) resemble traditional classrooms found in the physical world. Essentially, IVEs offer the opportunity to create unique experiences that would otherwise be resource prohibitive in the physical world and is a trait that should be taken advantage of.

IVEs, by their nature of being a software product, are fundamentally more flexible than the physical world. It should be noted that this flexibility is not trivial by any means – a fair amount of software development is required to create and change virtual worlds. Environments like Linden Labs' Second Life® were specifically built to allow users to design their avatars and even the entire virtual world around them ("Linden Labs," 2013). IVEs are often free of many constraints encountered in the physical world, allowing content developers and consumers to create and take part in new experiences that would otherwise be impossible in the real world. With the proper resources, designers can create anything they like and make the virtual world around users highly personalized. APIs and scripting languages are sometimes provided to simplify the process and most of the time, these changes can be applied dynamically, while the virtual world is still running thus eliminating the requirement to shutdown, restart the

environment and interrupt the user's experience. For an IVE to lend itself to hosting a virtual advisor, dynamic scripting is key as changes to the IVE, and advisor, have to occur in response to events and actions taken by (or happening to) users in the IVE.

VIRTUAL ADVISORS

The concept of a virtual advisor is, at its most fundamental level, a digital representation of the meaningful aggregation of data about and for a user. Figure 2 depicts a high level concept of a virtual advisor construct and shows the multiple data sources that could provide data to inform the virtual advisor. The various roles a virtual advisor could play are all centered on the user, highlighting the personal connection between a user and their virtual advisor. The “teacher/mentor” role is one role that offers the opportunity to radically shift the one-teacher-to-many-students paradigm to a one-teacher-to-one-student model. The following sections will discuss avatars in the teacher/mentor role.

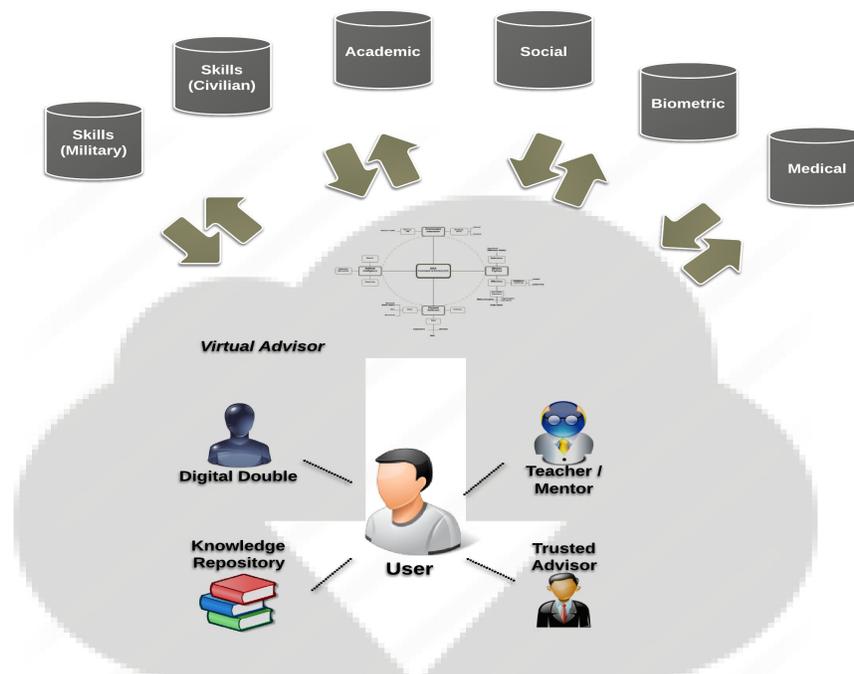


Figure 2. Concept of a Virtual Advisor informed by multiple data stores, providing different roles in a user-centric model.

Avatars as Virtual Advisors

Avatars are virtual representations of users in IVEs². They allow a user to experience the IVE by providing presence within the IVE. Avatars are also the user's primary interface for interaction with the IVE. There are many forms an avatar can take, ranging from simple to complex and fantastical to realistic. Most IVEs allow some level of customization of a user's avatar, whether it is as simple as choosing a gender among pre-defined avatars or creating a fully customized avatar. An avatar represents a good choice as a manifestation of a virtual advisor because people can associate with avatars. It has been shown that when users are allowed to create and customize their own avatar³, it creates a sense of ownership in that avatar, creates a psychological bond between the user and the avatar and can

² With the rise of non-player-characters (NPCs) in virtual environments, the definition of an avatar should expand to include NPCs as well, despite not meeting the criteria of “user”. This paper includes them in the definition.

³ It should be noted that much of the research presented here discusses avatars as agents of the user and doesn't directly discuss non-user controlled, or “third party”, avatars interacting with a user. The research presented suggests that the findings may equally apply to such third-party avatars, and so is presented as such. However, more research focusing on the effects of other avatars would be required.

enhance user engagement within an IVE (Swayne, 2013; Morie, 2008). Additionally, there are studies showing that the outward appearance of an avatar can affect a user's behavior. This behavior change is noticed not just within the IVE, but it can also transfer to the physical world (Yee & Bailenson, 2007; Bailenson, Beall, Blascovich, Loomis, & Turk, 2004). Some of these points will be discussed in more detail in later sections, but these phenomena lend themselves to using virtual advisors to depict "perfect" instructors and mentors by maximizing knowledge transfer.

Design Considerations

The process and outcome of the design of a virtual advisor becomes an important focus because not only can users have control over the appearance of their virtual advisor, but as mentioned above, there is a growing body of research highlighting the fact that an avatar's appearance (and behavior) can affect a user's behavior both inside the IVE and out in the physical world. Consideration should be given to designing a virtual advisor using criteria that will result in a meaningful and influential virtual advisor for a user with the goal of maximizing the effectiveness of knowledge transfer. One of the overarching criteria is trustworthiness or simply trust. Another closely related criteria is likability and familiarity.

Trust

Trust plays a large role in the efficient and effective transfer of knowledge (Levin, Cross, & Abrams, 2002). This trust is composed of two types of trust: benevolence- and competence-based trust. Benevolence-based trust is that based on the likability or "niceness" associated with a person. Competence-based trust is that trust that is afforded someone based on their verified level of experience. People who consistently prove they know what they are talking about earn high levels of competence-based trust (Levin, Cross, & Abrams, 2002). Both types of trust are affected by the strength of the ties between two people.

Mark Granovetter, a prominent social scientist, defined a tie as "...a combination of the amount of time, the emotional intensity, the intimacy (mutual confiding), and the reciprocal services" (Granovetter, 1973) provided between two social actors⁴. A strong tie would be a tie between two social actors that exhibits high levels of interaction over long periods of time, a concept that can be applied to avatars as virtual advisors⁵. Levin et al. (2002) suggests that if a user can develop a strong tie (or synonymously 'bond') with their virtual advisor, it would stand to reason that their level of trust goes up thereby increasing the effectiveness of the virtual advisor to pass knowledge on to the user. There are ways to build and strengthen bonds between a user and their virtual advisor, some enabled by technology, and others based on social and psychological principles. Increasing any of the four components (time, emotional intensity, intimacy and reciprocity) will then strengthen the bond.

The initial formation of a bond between a user and virtual advisor may be as simple as creating and customizing the virtual advisor. Recently, researchers at Penn State and the University of Michigan presented their findings that users' identities were mixed with that of their avatars when the user was allowed to customize their avatar. In the words of one of the researchers, "you are your avatar when it is customized" (Swayne, 2013). This bond likely would exist between a user-created virtual advisor as well though would not likely qualify as a strong bond. It would, however, serve as a foundation which can be built upon.

Likability and Familiarity

To build on a foundation of benevolence-based trust, virtual advisor design criteria could target likability and familiarity. Aside from designing a virtual advisor similar in physical appearance to the user, there are more subtle ways to create likability and familiarity, much of it centered around work conducted by Professor Jeremy Bailenson, the director of the Virtual Human Interaction Lab (VHIL) at Stanford University. Bailenson, in conjunction with colleagues, has done extensive work on creating more intimate interactions (as defined by Granovetter above) within virtual environments. His work on Transformed Social Interaction (TSI) discusses the use of mimicry by an avatar to copy the movement of its user. This mimicry showed an increase in more likable interactions between a user and an avatar (Bailenson, Beall, Blascovich, Loomis, & Turk, 2004). He also points out that morphing an avatar to change the outward appearance of the avatar (usually to add some of the user's physical attributes to the avatar) has the same

⁴ For more information on computers (and by proxy computer generated content) as social actors, see Dr B.J. Fogg's work on Captology.

⁵ Dr. B.J. Fogg also directly addresses reciprocity and computers in his book on Captology.

effect (Yee & Bailenson, 2007). While the concept and effects of morphing may seem contradictory to the idea of allowing a user to customize their avatar, it warrants further investigation to see if morphing and mimicry, when used in limited capacity and complimentary to user design, would further benefit the development of a strong tie between a user and their virtual advisor.

Competence-based trust can also be built, and like benevolence-based trust, laying the foundation may be a “simple” task – as simple as merely using a computer. Dr. B.J. Fogg, in his book titled *Computers as Persuasive Technologies: Using Computers to Change What We Think and Do*, details many ways that computers can be used to change our attitudes and behaviors in the physical world. He states that, among other things, people see computers as intelligent and infallible owing only to the fact that computers are highly complex machines (Fogg, 2003). This has the potential to lay a foundation for competence-based trust, as the virtual advisor (computer) would have instant credibility in the eyes of the user. That instant credibility might likely be fragile and would have to be enhanced though. Over time, and with successive correct answers, the instant credibility can be replaced with a robust, competence-based trust that would strengthen the tie between a user and their virtual advisor.

xAPI ENABLED VIRTUAL ADVISOR

An underlying requirement for the introductory design considerations listed above is access to data. From the perspective of a learner and with the virtual advisor playing the role of teacher/mentor, data pertinent to a user's learning history, their learning style, and any other data that would be useful in describing **how** a user learns would be necessary. It is worth pointing out that all of the data that would be required to ensure the success of a virtual advisor isn't necessarily going to be generated by xAPI-enabled learning environments alone. However, because the xAPI has the ability to collect data in the activity stream format, and the LRS is capable of storing much more than pass/fail information, the likelihood of the non-formal learning data being available increases, particularly if content creators specifically capture such data.

A Process Defined

At this early stage of implementation of the xAPI, researchers have the opportunity to begin planning what types of data will be relevant. There is also an opportunity to develop a standardized approach to collecting and using traditional and non-traditional learning data. Figure 3 depicts such a potential process and outlines activities to create and deploy virtual advisors. Depicted as seven activities (*Collect, Collate, Analyze, Inform, Design, Deploy* and *Feedback*) and loosely organized into four phases (gather, organize, use and feedback), the process is a starting point for discussions as well as for defining potential follow-on research and development.

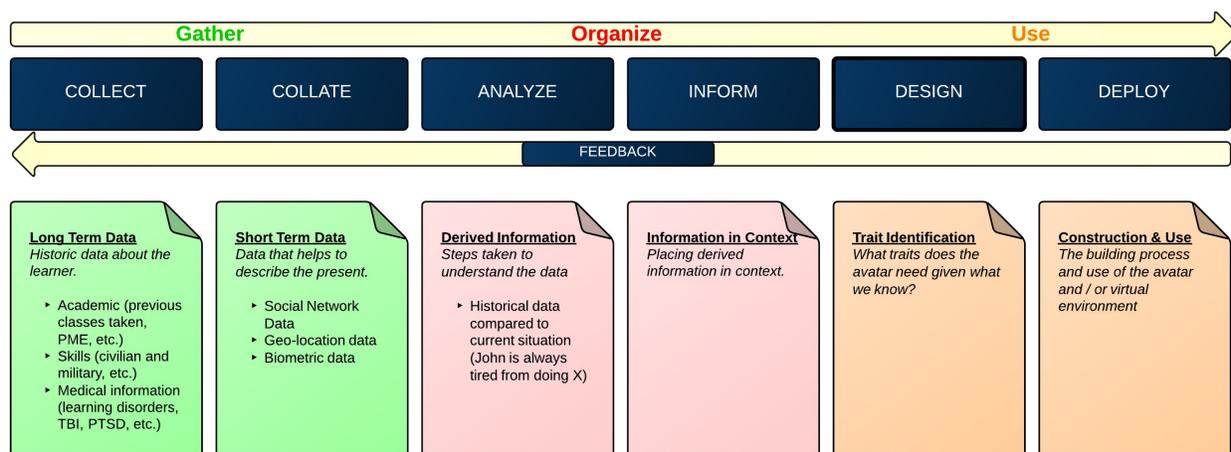


Figure 3. Virtual Advisor process

Although depicted linearly here, this representation is not meant to connote a serial process nor a finite one, but instead is conceived to be flexible enough to cover both initial and dynamic creation and deployment activities. These initial activities will be applicable the first time the virtual advisor is turned on. The *Feedback* activity allows

the virtual advisor to continue activities and respond to external inputs while still running. Because the virtual advisor is centered on the data about and for the user, the advisor is essentially persistent, though initial process activities would be necessary to bring the virtual advisor “to life”.

During the *Collect* activity, the virtual advisor would retrieve data from long-term data stores. Similar to the school-house threat of a permanent record, these highly persistent data stores are meant to house historical data about the user and their life. Example data stores include those used to house medical records for recording information about learning disorders, medical conditions, etc.; LRSs and LMSs containing academic information; and databases containing user skill data like academic, professional and military qualifications.

The *Collate* activity is similar to the *Collect* activity except that the data brought in by the virtual advisor is more temporal in nature. This data could be used to describe the current situation and may include data categorically similar to that in the *Collect* activity as well as additional data that may not be contained in long-term data stores. Examples include data about a user's personal and professional social network (who is around you), geolocation data (where are you) and even biometric data (how are you).

The *Analyze* activity is used to derive information based on the data from the *Collect and Collate* activities. In essence, it is comparing historical data to current data and is the first step for the virtual advisor in “knowing” about the user. The *Inform* activity is the second step in turning data into information. The purpose of the *Inform* activity is to take analyzed data and provide context to it by applying, for example, geolocation data. This data is then used in the *Design* activity.

The *Design* activity is used to determine the modality of the virtual advisor, whether it appears to the user in a visual, aural, textual or other form. The *Deploy* activity is responsible for the implementation of the virtual advisor. The *Feedback* activity completes the cycle by providing data back to each of the other activities in order to ensure smooth operation and maximize the effectiveness of the advisor. This data would include metrics detailing the efficiency of knowledge transfer. It would range in type from network data to determine the strength of a network connection to changes in location or activity and even environmental changes. Knowledge transfer could be measured by how well a user is accomplishing a task or test or what their attention span is (potentially measured by eye gaze or how long passes between clicks of a mouse, etc.). Standard network connection metrics could provide clues that a data intensive modality is no longer viable, creating an adjustment in the *Design* and *Deploy* activities or that certain data stores are no longer accessible, limiting effectiveness. Location data received by the advisor may also necessitate a modality change from visual to textual if the user moves from a computer lab to a quiet study space. The *Feedback* activity is likely the most active of the activities and phases and is critical to the success of the virtual advisor.

xAPI Data Collection

As previously discussed, in order to ensure the success of the virtual advisor, access to more data with a higher level of fidelity is required. With the introduction of activity streams, the xAPI can assist in collecting this additional data, including non-traditional learning data. Researchers and content developers may want to consider collecting data such as that detailed in the process outlined in Figure 3. To begin the discussion, the previously introduced design criteria of trust can be used as a starting point.

Collecting data to inform the design of a virtual advisor based on trust, or more specifically likeness, implies that a designer knows what a user sees as a likeness. Physical appearance is one way to instill likeness, and that can be relatively easily addressed by allowing a user to design and create a virtual advisor that incorporates some of their own features. But what if a user decides not to create their virtual advisor in their likeness, or what if the user's view of their likeness is skewed and they do not accurately represent themselves during the design phase? Through the collection of data that leads to understanding a user's likes and dislikes, it can be possible for the virtual advisor to take these inputs and, over time, morph itself to match a user's evolving sense of likeness. If likeness is synonymous with attractiveness⁶, the concept of determining what a user finds attractive and then subtly incorporating those characteristics could be equally as powerful as creating an advisor in self-defined likeness (Scharlat, 2007). Due to

⁶ Here, the assumption is made that attractiveness is very closely related to likeness. After all, it is generally accepted that people tend to find other people that look similar to themselves to be attractive.

the flexibility of the xAPI, those types of data could be collected by recording such data as who the user has in their social circles as well as with whom the user shares strong ties. For all cases, the design criteria data could be collected by tracking a user's activities.

Consider the following example of a learner, Bob, getting ready to undertake the arduous task of completing mandatory annual Information Assurance (IA) training. Once Bob has completed the ritualistic act of procrastinating, his virtual advisor reminds him that it is time to take the course. As it does so, the virtual advisor is already hard at work setting Bob up for success by pulling the current course information including the course content, the learning objectives and the evaluation questions. Comparing it to the course from last year, the virtual advisor notes new content to highlight. Bob's virtual advisor also recognizes what topic areas he didn't do well in last year and adds those content areas to a list of topics to highlight this year. As Bob enters the IVE and the course of instruction begins, the virtual world around him fades into the background, light music begins playing and his virtual advisor chooses a male avatar as his modality. The virtual advisor begins teaching the course to Bob, providing emphasis on those learning objectives that are either new or were prior areas of difficulty. The virtual advisor occasionally interrupts the course of instruction with jokes or funny anecdotes when he notices Bob's attention span diminishing or just to keep the instruction periods and content manageable.

All of the actions taken by the virtual advisor are done for a specific reason, based on data about Bob, and with a specific desired effect in mind. The virtual advisor chose an avatar as its modality because Bob was attempting to complete the course of instruction in an IVE. The virtual advisor further elected to draw himself as a male because he knows that Bob has hearing loss in one ear at the same frequency as higher pitched voices, even though he also knows that the female voice is a more attractive quality for getting the attention of male users. The virtual advisor's height was drawn relative to the height of Bob's avatar as a function of likeness and could dynamically morph depending on whether the advisor wanted to be more or less assertive with Bob. As the course of instruction began, the virtual advisor, unbeknownst to the user, faded the visual background and added light background music, because over the course of spending time with Bob, data the virtual advisor collected suggested that those are ideal conditions for Bob to concentrate on complex and tedious content. During the course of instruction, the virtual advisor collected more data using the xAPI, not only on Bob's progress, but created a digital library of everything Bob learned, and also collected data to increase the degree to which the virtual advisor "understands" how Bob learns. Some of the data collected was used in real-time to adjust the virtual advisor's appearance and approach. All of the data, whether used immediately or not, was stored for future use, and included data relating to how well the advisor's emphasis techniques worked on the new and historically poor performing topics. At the end of the course of instruction, the virtual advisor collected the exam results, remediated any questions required for Bob to satisfactorily pass the course then stored it all away in any variety of LRSs and other associated data stores.

CONCLUSION

The xAPI represents the next generation model for the collection of data about learning experiences. It is expected to provide flexibility beyond the SCORM model due in part to the introduction of activity streams to the learning community. With this format, practically anything that can be "done" can be tracked, which means the community now has the ability to collect data beyond that traditionally required for formal learning environments. Such non-traditional learning data would be useful in expanding our understanding of how people learn and ultimately may provide more personalized learning environments. One such opportunity is the use of a virtual advisor. Taking advantage of the inherently flexible nature of computer generated IVEs, research can provide virtual advisors that are informed by data collected through the implementation of the xAPI.

The research presented in this paper represents the foundation for this concept. Further work would be required to better understand what traits truly influence a user's ability to learn. One previous inhibitor to gaining this information was the lack of available data. The xAPI specification addresses this and allows developers to work from a larger data set, providing better insight into what makes a learning experience beneficial.

There are a few "easy" next steps that can be taken to help move this concept forward. First, content developers need to begin designing content using the xAPI specification. That will open up the ability to collect larger amounts of data and make it available for use. Next, by implementing a standard dictionary for activity streams dedicated to

virtual advisors, content designers and creators would be able to collect data in a consistent manner and that data would have meaning across all xAPI-enabled learning environments looking to implement virtual advisors. Concurrently, work can be started on further defining the activities of *Collect, Collate, Analyze, Inform, Design, Deploy* and *Feedback* so that as data does become available, a virtual advisor will be ready for implementation.

REFERENCES

- Advanced Distributed Learning Initiative: Experience API (xAPI). (n.d.-a). Retrieved April 30, 2013, from <http://www.adlnet.gov/tla/experience-api>
- Advanced Distributed Learning Initiative: Introducing the Training and Learning Architecture (TLA). (n.d.-b). Retrieved April 30, 2013, from <http://www.adlnet.gov/introducing-the-training-and-learning-architecture-tla>
- Advanced Distributed Learning Initiative: SCORM. (n.d.-c). Retrieved June 5, 2013, from <http://www.adlnet.gov/scorm>
- A serious Second Life for business schools. (n.d.). *International Society for Presence Research*. Retrieved June 11, 2013, from <http://ispr.info/2013/05/29/a-serious-second-life-for-business-schools/>
- Bailenson, J. N., Beall, A. C., Blascovich, J., Loomis, J., & Turk, M. (2004). Transformed Social Interaction: Decoupling Representation from Behavior and Form in Collaborative Virtual Environments. *Presence*, 13(4), 428–441.
- Bailenson, J., & Yee, N. (2005). Digital Chameleons: Automatic Assimilation of Nonverbal Gestures in Immersive Virtual Environments. *American Psychological Society*, 16(10), 814–819.
- Bloom, B. (1984). The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. *Educational Researcher*, 13(6), 4–16.
- Fogg, B. J. (2003). *Persuasive Technology: Using Computers to Change What We Think and Do*. Morgan Kaufman Publishing.
- Granovetter, M. (1973). The Strength of Weak Ties. *American Journal of Sociology*, 78(6), 1360–1380.
- Kling, A. (2012, September 12). Many-to-One vs. One-to-Many: An Opinionated Guide to Educational Technology. *The American*. Retrieved from <http://www.american.com/archive/2012/september/many-to-one-vs-one-to-many-an-opinionated-guide-to-educational-technology>
- Knowledge of Student Characteristics. (n.d.). *intime: integrating new technologies into the methods of education*. Retrieved May 25, 2013, from <http://www.intime.uni.edu/model/teacher/teac1summary.html>
- Kramer, S. H. (2010). *Teaching in virtual worlds: a qualitative case study*. Walden University.
- Levin, D., Cross, R., & Abrams, L. (2002). The Strength of Weak Ties You Can Trust: The Mediating Role of Trust in Effective Knowledge Transfer. In *Academy of Management Proceedings* (pp. D1–D6). Retrieved from <http://www.levin.rutgers.edu/research/trust-paper-shortened.pdf>
- Linden Labs. (2013). *Second Life*. Retrieved May 25, 2013, from <http://lindenlab.com/products/second-life>
- Military Open Simulator Enterprise Strategy (MOSES). (n.d.). *Home*. Retrieved June 3, 2013, from <http://brokentablet.arl.army.mil/index.html>
- Morie, J. (2008). The Performance of the Self and Its Effect on Presence in Virtual Worlds. In A. Spagnolli & L. Gamberini (Eds.), *Presence 2008: proceedings of the 11th International Workshop on Presence ; Padova, Italy, 16 - 18 October 2008* (pp. 265–269). Presented at the 11th International Workshop on Presence, Padova: CLEUP.
- O'Donovan, J., & Smyth, B. (2005). Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interfaces* (pp. 167–174). New York, NY, USA: ACM. doi:10.1145/1040830.1040870
- Scharlat, J. (2007, December). *Intelligent Virtual Environment Agents (IVEAs): Conducting Information Operations in Virtual Environments*. Naval Postgraduate School, Monterey, CA.
- Second Life Education Directory. (2013, June). Retrieved June 1, 2013, from wiki.secondlife.com/wiki/Second_Life_Education_Directory
- Smart, J. M., Cascio, J., & Paffendorf, J. (2007). Metaverse Roadmap Overview. Retrieved from <http://www.metaverseroadmap.org/overview/>
- Stephenson, N. (2000). *Snow Crash*. New York: Bantam Books.

- Swayne, M. (2013, May 2). Bonding with your virtual self may alter your actual perceptions. *Penn State News*. Retrieved June 3, 2013, from <http://news.psu.edu/story/275626/2013/05/02/research/bonding-your-virtual-self-may-alter-your-actual-perceptions>
- Tin Can API. (n.d.). *Overview*. Retrieved June 3, 2013, from <http://tincanapi.com/overview/>
- Virtual World Framework. (n.d.). *About*. Retrieved June 3, 2013, from <http://www.virtualworldframework.com/web/about.html>
- xAPI-Spec. (2013, May 29). *GitHub*. Retrieved June 3, 2013, from <https://github.com/adlnet/xAPI-Spec>
- Yee, N., & Bailenson, J. (2007). The Proteus Effect: The Effect of Transformed Self-Representation on Behavior. *Human Communication Research*, (33), 271–290. doi:10.1111/j.1468-2958.2007.0229.x