Representing Augmented Reality Applications in Systems Modeling Language

Emi Aoki*, Bach Tran†, Vinh Tran*, Kimseng Soth*,
Charles Thompson*, Susan Tripathy‡, Kavitha Chandra* and Shivakumar Sastry†
*Department of Electrical & Computer Engineering
University of Massachusetts Lowell, Lowell, Massachusetts 01854
Email: emi_aoki@student.uml.edu
†Department of Electrical and Computer Engineering
University of Akron, Akron, Ohio, 44325
‡Department of Sociology
University of Massachusetts Lowell, Lowell, Massachusetts 01854

***Abstract*—Augmented Reality (AR) devices offer novel capabilities that can be exploited in AR systems to positively impact human-machine interactions in a variety of future-work and education contexts. This paper presents a systems model for a no-code AR systems framework that can be used to create AR applications that present just-in-time informatics to assist and guide users in the completion of complex task sequences while ensuring operator and environment safety. The salient structural and behavioral aspects of the system, and key use cases are modeled using the Systems Modeling Language (SysML). Representative examples of the model are presented using use case, block definition, internal block, activity, and state-machine diagrams. These models offer new insights into how AR capabilities can be integrated with a variety of engineered systems. In the future such SysML models can steer the design of new tools and an ontology to strengthen connections to domain knowledge.**

***Index Terms*—Augmented Reality, Model-Based Systems Engineering, SysML, Annotation, Navigation, Systems Framework**

I. Introduction

Augmented reality (AR) devices[1] that are capable of immersive experience have matured significantly in recent years. When incorporated effectively into systems, these devices can now address many of the challenges brought about by the digital transformation initiatives in manufacturing, health, and education arenas. AR devices can present super-imposed virtual informatics on objects and spaces in the physical environment. This ability to offer in-situ informatics in the physical work-spaces creates new opportunities for workforce training, that includes a transfer of knowledge from experts to trainees, rapid acquisition of new skills, and the safe execution of acquired skills. For students and educators, such systems offer a personalized learning portal, enabling new models of pedagogy in STEM education that emphasizes the engagement of students across traditional disciplinary boundaries. Thus, there is considerable interest to better understand the extent to which such technologies can support future workers, students, and educators to become more efficient in adopting the digital environment, and to play a role in improving its use.

[1]e.g., Microsoft Hololens, MagicLeap, or Varjo

There are several challenges one must confront in the design and deployment of effective AR systems. First, most AR systems are structured around a game engine, with their attendant idioms and interaction modalities. Education and training systems do not require complex game plays or challenges, even when the AR content must be delivered in a personalized manner. Second, much of the effort involves the identification, design, and delivery of effective AR content. This is perforce a multidisciplinary endeavor, and requires a reduction in technology barriers to enable such teams to collaborate effectively during the content creation process. Third, personalized delivery of AR informatics must cope with a variety of unanticipated situations that can occur either because of user's background or due to variabilities and uncertainties in the physical workspace. Fourth, the requirements-design-implementation-operation continuum in many complex environments is necessarily an iterative process. Effective modeling tools and approaches can mitigate the risk of inconsistencies across this continuum. Finally, given the evolution toward a future where user behaviors are likely to be tightly intertwined with technology advances, it is critical to integrate into the design of the AR systems platform, the best practices for engaging users in the research and design of the system. Participatory action research (PAR) for example is a research framework that prioritizes the active participation of all stakeholders to create an inclusive action process for problem-solving and constructive change. PAR is especially suited for examining and improving teamwork practices, because of its emphasis on empowering participants to have their voices heard, share authority in decision-making, change existing systems, and collaborate with others in a respectful and inclusive manner [1], [2]. AR systems in the future can better enable and support these practices if the content creation process is reasonably accessible to corresponding domain experts such as sociologists in the context of PAR.

To address the above challenges, this paper presents a systems framework to design and deploy AR systems for education and training without the need for extensive a priori knowledge on coding or game design. The salient structural and behavioral elements of this framework are presented using SysML, which is a well-known systems modeling language. The AR Systems Framework emphasizes the role of an AR system developer as one who develops the AR content, i.e., text, images, videos, audio clips, animations, or models to accurately represent the domain issues and in formats that are necessary for the underlying AR devices. This framework abstracts many aspects of the underlying game engine and considerably simplifies the development process. The SysML models emphasize the core structural and behavioral elements of the framework and offer new insights for extending the framework and its applications along several dimensions.

The remainder of this paper is organized as follows. Section II provides a brief background on approaches being taken to modularize AR application design. The current practice in AR application design is summarized in Section III. Section IV outlines the approach taken in this work that is referred to as a no-code AR Systems Framework. A subset of SysML models for the AR application is presented in Section V. Future extensions of this work and trade-offs involved are discussed in Section VI, and Section VII concludes the paper.

II. Background

The role of AR devices in several domains such as engineering [3]–[6], health [7], [8], and humanities [9] has been investigated for several decades. A recent technology forecast report also identifies the central role of AR in emerging applications [10]. In [4], the authors present a comprehensive classification of the AR landscape in industrial environments. They identify four main areas — manufacturing, logistics, maintenance, and training — as the primary areas of focus in industrial environments. The role of CAD as an enabler for AR and VR applications is recognized. Modeling abstraction of physical components as virtual products, referred to as digital twins, is expected to provide various benefits throughout product development and management [11], [12]. These models can be used for design specification, verification, and validation. It also allows for quick forecasting of product behaviors and structures at different scales and various scenarios. AR applications can play a role in the construction of digital twins.

Indoor navigation has remained a long-standing problem and has new importance in AR Applications. The authors in [13] discuss infrastructure-based and sensor-based approaches. The navigation problem has new dimensions that must be better addressed in future work scenarios.

The development of AR applications is often structured at source-code level. As a solution to this limitation, a number of research studies have been conducted to propose a framework or toolkit for the development of AR applications, that include component-based and model-based approaches. A model-based approach for designing new interactions between users and AR systems is presented in [14]. In this work, new interaction models are specified in a declarative manner using state charts. The specification helps to bridge the gap between authoring considerations and the code that is necessary for the AR system. The model-based approach used in [15] presents a toolkit which simplifies the development of AR applications. This approach can provide a run-time implementation of the AR system from the high-level models. The application developed in [16] provides AR-assisted production configuration by implementing a model-based approach. The architecture performs the product modeling by implementing features stored in meta-data which allows for considering user needs and environmental constraints.

Component-based software framework proposed in [17] is a new approach for building AR systems. This architecture consists of a set of reusable programs which solve key problems in an AR system. This also provides flexibility and scalability of potential applications. The AR system for the health-care training process in [7] is an example of this development approach. Although this system provides benefits in terms of audio-visual feedback and high fidelity simulation training, it is difficult to extend into new application contexts.

Some studies have proposed to build an AR framework to overcome the inflexibility of developing single-purposed software products. These frameworks are commonly data-driven and allow the developers to modify the system in either graphic-based or text-based manner. The work in [18] presented a graphical-based approach for designing and deploying AR applications in mobile platforms. This framework provides a set of programmable components, their properties, and programming rules. The behavior of the system can be modified by simply adding or deleting component blocks and selecting or deselecting properties of each block. It is applicable to many contexts, such as e-learning, health-care, and design. However, it still requires basic programming knowledge to understand the components and change the system accordingly. It also requires the involvement of programmers to adjust the system for a new operational context. The work in [19] is interesting because it squarely addresses the challenges of AR content development and the central role of CAD databases.

III. Current Practice in AR Application System Design

Currently, AR system designers must be well-versed in the intricacies of game engine platforms such as UNITY 3D or UNREAL [20]. In addition, they must have considerable experience in a software pipeline that includes tools for rendering, modeling, and animation. The designers are usually aware of a large spectrum of 3D models that are available to make their work more efficient.

The AR application development process is often focused on a particular AR device such as the Microsoft HoloLens. The approach typically involves using a game engine to create an immersive person-centered experience [21]. Game engines such as Unity3D support the developers to render 2D or 3D graphics, to detect collisions and respond, to present sound, scripts and animations, to network with other compute and storage devices, to stream data, and to track the position of the users in the environment. The software tools also help developers to create and deploy the AR content before deploying to the AR device as illustrated in Fig: 1.

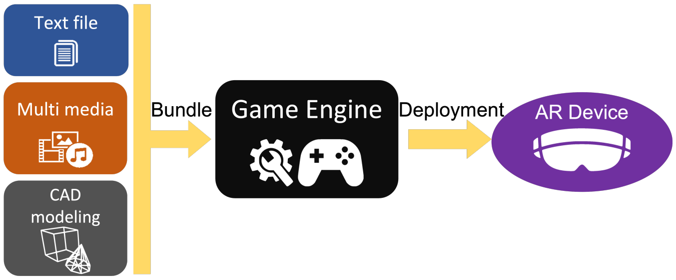

Fig. 1. A Game Engine is used to generate content and deploy AR systems.

An AR application that is developed using this approach often serves a single-purpose, and its AR contents cannot be changed after deployment. The pre-populated content is often based on the experience of a few training professionals. These systems are not personalized to the needs and preferences of individual users. The content is presented to the users when a set of defined event triggers are recognized. If the AR content relies on information in an underlying CAD model that was created when the physical environment was designed, such an approach is not likely to be useful especially when components and subsystems have changed or evolved. When changes such as a revision to the 3D models are necessary, the developer must re-work the entire development process and re-deploy the application. This approach is tedious and requires considerable programming maturity before the designers can be effective. The generation of designers who have good skills with such a set of tools often lack deep insight into the current systems for manufacturing or the current methods of pedagogy. Further, those with these skills are likely to operate at the level of programmers rather than at the level of architects.

Despite the above limitations, such AR systems that have pre-populated content have been used extensively in a variety of training and education scenarios.

IV. No-Code AR Systems Framework

The core idea in the AR Systems Framework described here is that of an *Immersive Action Unit* (IAU) [22]. An IAU is a unit of AR content that is presented to the user through the AR device. An IAU may contain any combination of text, video clip, audio clip, animation, or a model. The user is usually expected to perform an action when they are presented with an IAU. This framework abstracts the underlying game engine. Instead of the complex game plays and rich interaction experience, it offers a limited set of user-system interactions that are effective for education and training. The four principal user activities supported are:

- **Annotation:** *where the system presents just-in-time informatics about objects that are of interest to the user,*
- **Navigation:** *where the system presents navigational signs to help the user identify objects of interest or physically navigate a complex work environment,*
- **Guidance:** *where the system presents the user a sequence of IAUs to guide the completion of a task sequence,* and
- **Safety:** *where the system presents the status of a variety of safety parameters and alarms, as configured by the user.*

The structure of the AR Systems Framework is shown in Fig: 2. The IAU server is a repository of all IAUs that are stored in a standard relational database with an SQL front-end. The AR device interacts with the IAU Server using a request-response model. Typically, upon completing the action step associated with an IAU, the user would request the next IAU. The Operator and System Safety subsystem collects data from the physical environment and the operator safety equipment. All of these data are pushed to the AR device by the IAU server. The AR device can also initiate requests for annotations and navigation via the CAD server, again using a request-response model. Responses for valid requests are sent back to the AR device. A live stream from the AR device can be viewed by other staff or trainees at the monitoring station. The monitoring station can also be used to monitor progress of a maintenance mission by replaying the historical status captured in the IAU server. Further details of this framework are presented in Section V using SysML models.

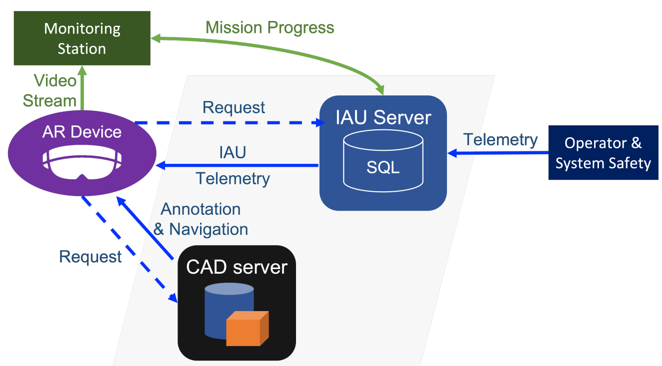

Fig. 2. System Architecture for the AR Framework.

AR system designers who use the above framework can now focus on creating the AR content and designing the flow, or sequences, in which these content items can be presented to users as IAUs. This fundamental change in the work pattern is illustrated in Fig: 3 and offers several advantages over the approach in Fig: 1. Notably, the content-centric approach of the framework serves as a conduit between content creators and content consumers, via the AR devices, using well-known and popular software applications that are illustrated in Fig: 4. These easy-to-use tools can be readily utilized in education and training or participatory action research settings to create, update, and deploy AR systems without the need to develop extensive code.

V. Modeling the AR Systems Framework

This section presents SysML models for the AR Systems Framework. They have been created using the CATIA-2021

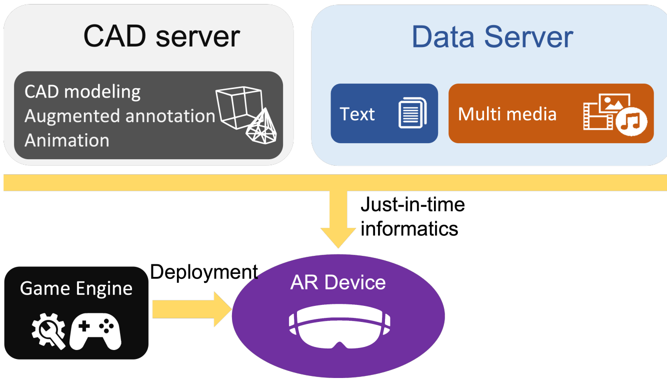

Fig. 3. The AR Systems Framework creates a fundamental change in the development pipeline.

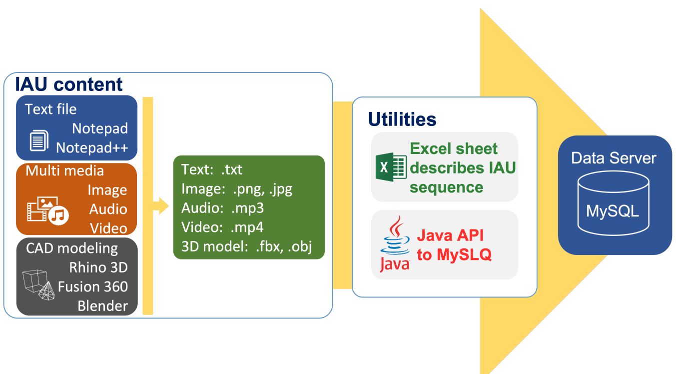

Fig. 4. Interdisciplinary teams can develop AR Content.

Magic Systems of Systems Architect toolset. This toolset offers support to capture the requirements, use cases, structure, and behavioral models that describe the operational features of the application.

Specifying the AR system in SysML has several benefits. First, such a framework can enable multi-person teams with varied expertise to contribute to activities in the requirement-operations continuum to fully realize the system. The consistency checks and simulation capabilities are valuable to ideate options and select from available options based on rigorous discussions. Second, in the future, these models can serve as a basis to programmatically generate the code necessary to operationalize the framework. Thus, the framework can be adapted to future work environments and contexts that may not be anticipated at this time. Third, the models can reveal a rich set of opportunities for AI-assisted tools that can both improve the user experience and improve the efficiency of the system operations. Finally, one can expect that these SysML models can be integrated to enhance the specification of a large spectrum of engineered systems with AR capabilities.

Systems design involves translating abstract concepts and stakeholder requirements into a concrete specification that can be implemented, verified, validated, deployed, and operated. In the AR Systems Framework, the inputs are streams of user position, gaze, gesture, and surrounding environment, and the outputs are IAUs.

The primary use cases supported in the AR Systems Framework are shown in Fig: 5. As illustrated, the user must power on and wear the AR device. The device may initiate a device-level calibration when a new user is detected. The registration step is required to map the virtual coordinates in the AR device to the corresponding physical coordinates in the workspace as described in [23]. The four use cases, Annotation, Navigation, Guidance, and Safety, correspond to the four primary activities described in Section IV. A use case describes how users can benefit from the system, representing their requirements at the highest level such as the one shown in Fig: 5. More detailed specifications of use cases can be presented using other SysML features such as activity diagrams, state machines, or sequence diagrams. These diagrams can capture some of the details related to the design and implementation of the AR Systems Framework.

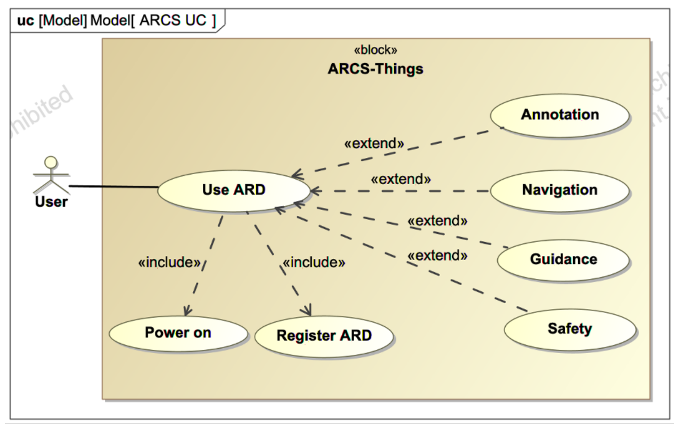

Fig. 5. Primary use case in the AR Systems Framework.

The top-level structural representation of the system that is needed to support the use cases in Fig: 5 is captured by a block definition diagram (BDD) that is shown in Fig: 6. As illustrated in the figure, the main blocks are consistent with the subsystems shown in Fig: 2. The SQL server in Fig: 6 corresponds to the IAU server in Fig: 2. The CAD server, the monitoring station, the safety subsystem, and AR device are illustrated.

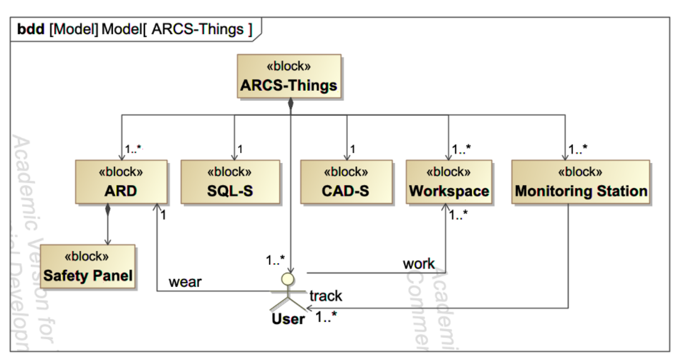

Fig. 6. Top-level Block Definition Diagram

The BDD shows parts of the system at the level of interest. Individual blocks can be further refined. For example, the BDD for the SQL Server is presented in Fig: 7. A solid line with an arrow shows a composition representing a parent-child relationship. A solid line without any arrowheads means the connected components have access to each other's data. As illustrated, the IAUs are stored in a relational database. In addition, there is a task sequence and a flow table that is used to store sequence information for a sequence of IAUs that must be presented in order. Each such sequence is called a mission, and the user can select a desired mission using the AR device.

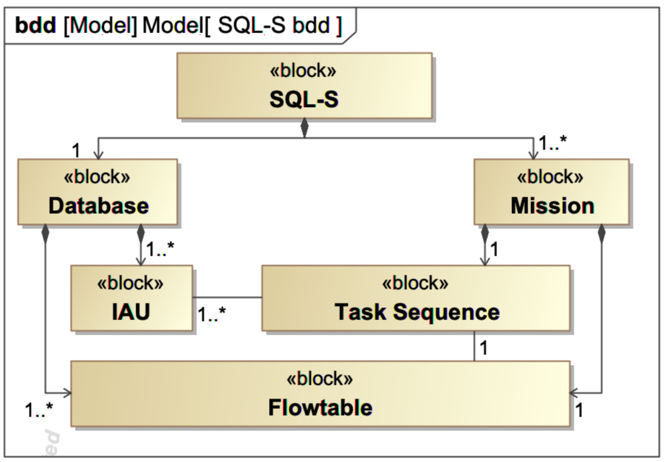

Fig. 7. SQL Server Block Definition Diagram

Internal block diagrams (IBD) depict how the components within a block interact in performing a particular action. Multiple IBDs can be configured for the required action set. The connections depict data/object flow among the components. The IBD for the SQL Server in Fig: 7 is presented in Fig: 8. These figures capture object flows between blocks and the block responsible for performing computation. The small squares attached to the blocks are ports that are used as an object receptor or sender. As illustrated, a request for a new IAU must specify a mission ID. The mission ID is used to acquire a task ID from the flowtable. The actual task step is extracted from the state of the task sequence. The task step is used as a key to acquire the AR content (IAU) from the database which is sent back to the AR device.

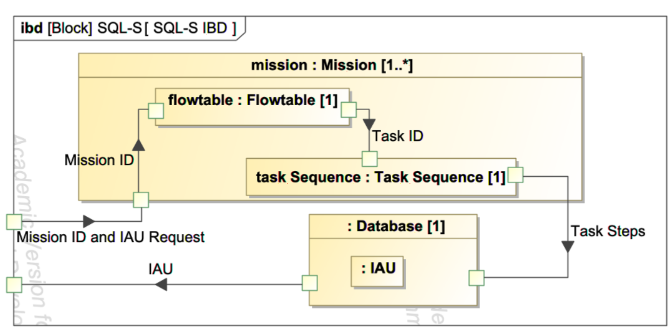

Fig. 8. SQL Server Internal Block Diagram

Fig: 9 presents the components of the CAD server. The CAD server abstracts a CAD model for the physical workspace, an octree structure constructed from the CAD model, and a path planner. The CAD model contains several artifacts of interest in the physical workspace. These artifacts are also represented in the octree at the appropriate level of depth [24]–[26]. Each artifact has a set of physical coordinates with respect to the origin of the CAD model. The octree stores the annotations associated with the artifact. A dashed line shows what item flows between two blocks. The path planner is used to support the navigation activity/use case.

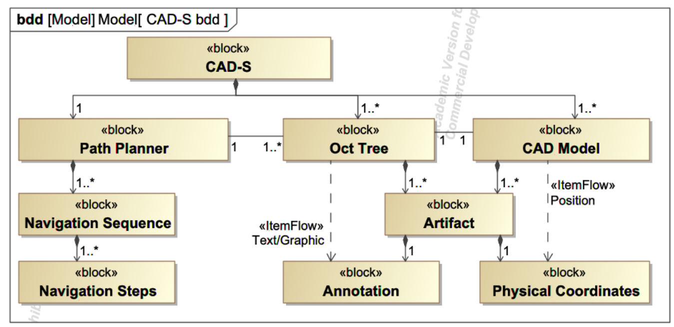

Fig. 9. CAD Server Block Definition Diagram

The connections between the parts defined in Fig: 9 are presented in the IBD's of the CAD block. Fig: 10 represents the annotation activity and Fig: 11 the navigation activity.

In the annotation activity shown in Fig: 10, the physical coordinates received are used by the octree to compute the first intersecting artifact with matching physical coordinates. When the artifact is identified, the corresponding annotation is retrieved and sent back to the AR device.

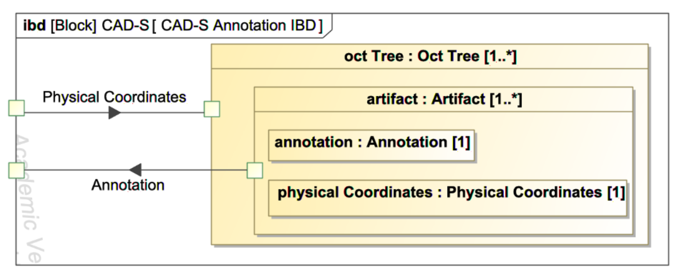

Fig. 10. CAD Server Internal Block Diagram for Annotation

In the navigation activity of Fig: 11, the user specifies the target artifact to which they desire navigational guidance. This target and the current position of the user are used by the path planner to compute a path in the physical workspace. If the path involves multiple steps, this sequence is stored, and the first step is returned to the AR device for display to the user.

While the internal block diagrams represent object flow and some level of interactions between components, system-wide behaviors must be represented using other models in SysML. For example, Fig: 12 shows the annotation use case from a system-wide perspective with an Activity diagram.

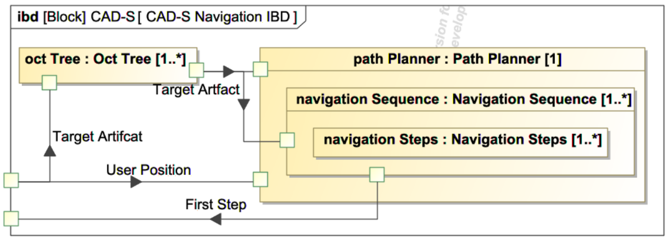

Fig. 11. CAD Server Internal Block Diagram for Navigation

Notice that the user must first wear the device and launch the annotation mission. When the mission is launched, it connects to the server and is ready to respond to annotation requests after the connection request is confirmed. When the user gaze rests on an artifact in the physical workspace for a short duration, or the user points to an artifact, the virtual coordinates of the artifact obtained from the AR device are sent to the server. These virtual coordinates are transformed to physical coordinates using the transforms computed during the registration step [23]. The physical coordinates are sent to the CAD server where the CAD server determines if there is a corresponding artifact with matching physical coordinates. If such an artifact is identified, the annotation that corresponds to the artifact is retrieved and sent to the AR device.

State machine diagrams are another class of behavior models that are applied in this model. As depicted in Fig: 12, a use case involves several user actions and system actions. These actions are events that cause the system state to change. A state machine diagram captures important states of components that can be observed and the events that cause these states to change. Fig: 13 shows the states of the CAD server and the events that trigger transitions among these states. A state is represented by a rectangle with rounded corners. Action performed upon an entry or while in that state is also identified for certain states. A solid circle indicates the initial state, while a circle with a dot inside is the final state. Solid lines connected between the states are called events, precisely user or system actions in the developed AR system.

The state machine for the AR device is illustrated in Fig: 14. Notice that depending on the activity that the user selects, a different part of the state machine describes the behavior. Events that are triggered by the system or user action are highlighted in the figure.

VI. Discussion

The AR Systems Framework presented in this paper offers a new approach to designing and deploying AR systems that emphasizes AR content. This approach is different from application-focused approaches that involve a significant level of program development with the Unity tool chain. The framework enables content designers to use popular tools to create the content consistent with the formats that are acceptable to AR devices. The compliance with these formats can be checked offline before deployment. In contrast to the general game plays that can be created by designers using game engines, this framework supports four common use cases that appear to be most relevant to the use of such AR systems in education and training scenarios.

In contrast with traditional AR apps, the content for this framework can be dynamically changed, without the need for compiling and re-deploying the app on the AR device. This flexibility offers tremendous advantages in high-value maintenance scenarios where design maintenance has remained a challenge for several decades.

The system architecture is flexible, robust, and scalable. The SQL platform is time-tested and well-understood. New missions can be added or deleted by changing the database. Notably, this capability enables end-users to create and own their content without relying on third-party software tools to curate and load their content into the AR devices.

The CAD server serves two important roles. First, it is an important gateway to authenticate the users before sharing proprietary content. Second, and perhaps more important, by registering the virtual view in the AR device with an underlying CAD model, we eliminate the need for expensive image processing techniques to identify artifacts or geometrical constraints in the physical workspaces. While this approach limits the applicability of our framework to the realm of engineered systems, where CAD models exist or can be readily created, the emerging needs for hands-on training in workforce reskilling and education far outweigh the limitations.

The SysML models, that are currently under development, offer a pathway both to refactor the design of the framework and to communicate its capabilities to a wider community. In the future, we expect to further develop the SysML models by also including models for the structure and dynamics of physical workspaces. When fully developed, we expect that these models will offer new insights into how AR capabilities can be best incorporated to enrich the human-machine interactions in engineered workspaces.

There are a few limitations and challenges that must be addressed. Currently, the annotation support can be provided only for artifacts that are static in the physical workspaces. It will be very interesting to explore how the parametric modeling capabilities of SysML can be leveraged to annotate artifacts that move in the workspaces.

VII. Conclusions

An AR Systems Framework and selected SysML models for the salient structural and behavioral aspects of this framework are presented in this paper. The framework enables interdisciplinary teams of AR content creators to design and deploy AR systems without the need for extensive coding. The application of the framework to provide just-in-time informatics in four use cases that enrich a user's interaction experience in a physical workspace has been presented. To annotate an artifact in the physical workspace, the user can gaze at the artifact or point to the artifact. The user can select a mission to receive guidance to complete a sequence of tasks. The system presents a sequence of IAUs and expects the user to complete a specific action in each step. When the

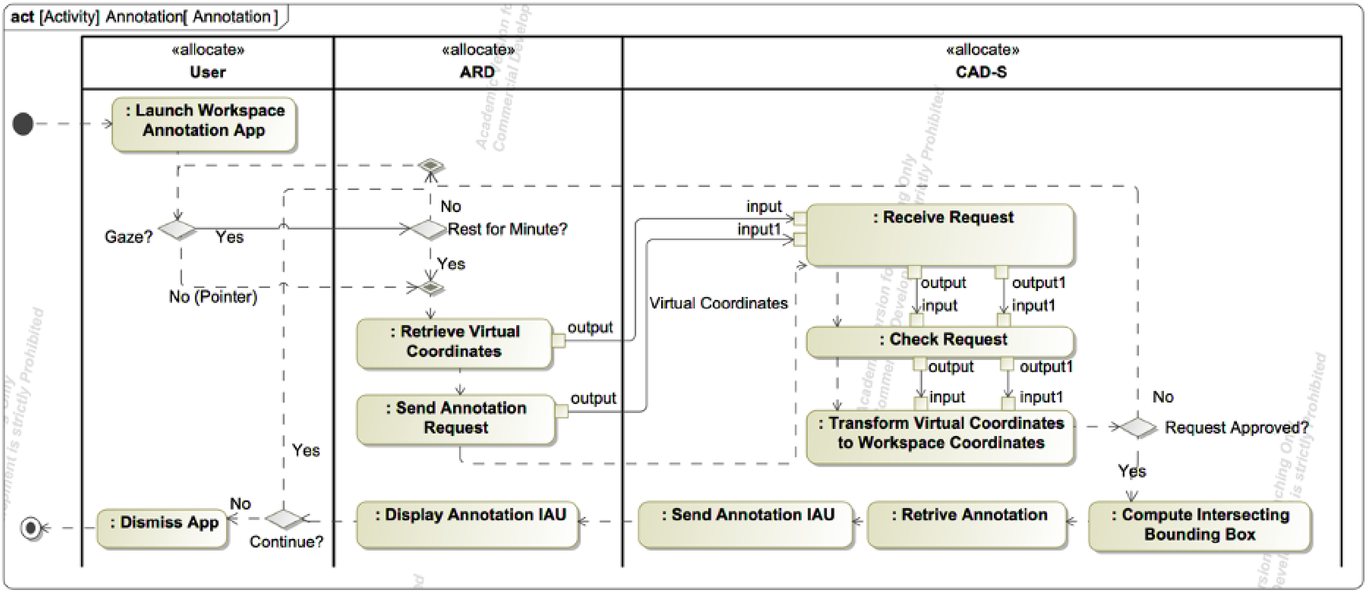

Fig. 12. Activity Diagram for the Annotation Use Case

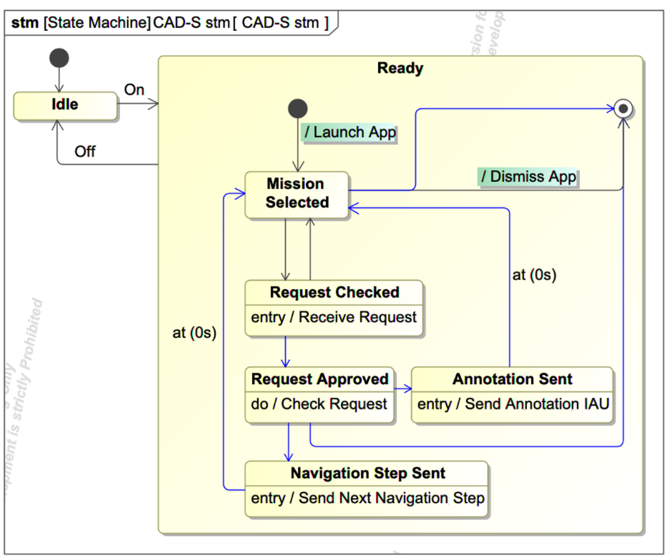

Fig. 13. CAD Server State Machine

user requests navigation guidance, they must identify a target artifact, and the system responds by presenting navigational signs to help the user to either recognize the artifact or to physically move to the location of the artifact in the physical workspace. The operator safety panel presents status of the physical workspace or the operator's safety equipment, as configured. The SysML models presented offer insights into the internal structure and behavior of the framework and how the system can be extended. In the future we expect that the SysML models will be useful to incorporate AR interfaces to a variety of engineered systems.

Acknowledgment

The authors acknowledge support of this research from the National Science Foundation Future of Work at the Human Technology Frontier program under Grant Nos. #2128749 and #2128950. Any opinions, findings, conclusions or recommendations expressed are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] E. Ericson-Lidman and G. Stranberg, "Learning to deal constructively with troubled conscience related to care providers' perceptions of deficient teamwork in residential care of older people - a participatory action research study." *Scandinavian Journal of Caring Sciences*, vol. 29, no. 2, pp. 215–224, 2014.

[2] M. Minkler, A. P. Garcia, V. Rubin, and N. Wallerstein, "Community-Based Participatory Research: A Strategy for Building Healthy Communities and Promoting Health through Policy Change," University of California, Berkeley School of Public Health, Tech. Rep., 2012. [Online]. Available: https://www.policylink.org/sites/default/files/CBPR.pdf

[3] N. Navab, "Developing killer apps for industrial augmented reality," *IEEE Computer Graphics and Appplications*, vol. 24, no. 3, pp. 16–20, 2004.

[4] S. Buttner, H. Mucha, M. Funk, T. Kosch, M. Aehnelt, S. Robert, and C. Rocker, "The design space of augmented and virtual reality applications for assistive environments in manufacturing: A visual approach," in *Proceedings of Pervasive Technologies Related to Assistive Environments*, 2017.

[5] M. Funk, T. Kosch, and A.Schmidt, "Interactive worker assistance: Comparing the effects of in-situ projection, head-mounted displays, tablet, and paper instructions," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 934–39.

[6] C. D. Fehling, A. Mueller, and M. Aehnelt, "Enhancing vocational training with augmented reality," in *Proceedings of the 16th International Conference on Knowledge Technologies and Data-driven Business*, 2016.

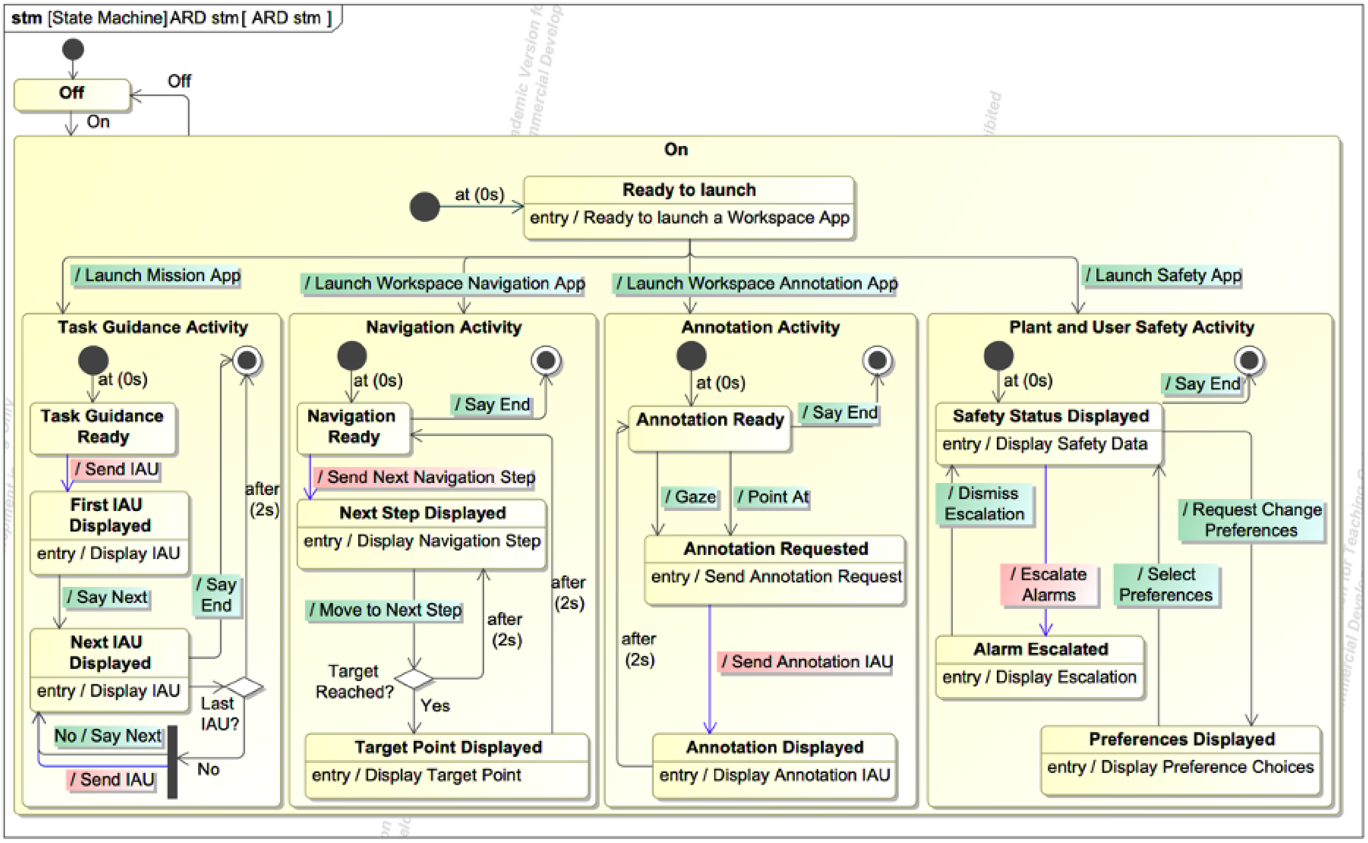

Fig. 14. State Machine for the AR Device is complicated because the device must support multiple activities.

[7] S. Balian, S. K. McGovern, B. S. Abella, A. L. Blewer, and M. Leary, "Feasibility of an augmented reality cardiopulmonary resuscitation training system for health care providers," *Heliyon*, vol. 5, no. 8, p. e02205, 2019.

[8] N. Arora, M. Soumalainen, M. Pouke, E. Center, K. Mimnaugh, A.P.Chambers, S. Pouke, and S. LaValle, "Augmenting immersive telepresence experience with a virtual body," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 5, pp. 2135–45, 2022.

[9] B. Serres, S. Letienne, O. Roussey, and G. Venturini, "Visit: A content management and exploration system for mobile augmented reality in the context of digital humanities and cultural heritage," *ACM Journal of Computation and Cultural Heritage*, 2023.

[10] e. A. Abadi, "2023 technology predictions," IEEE Computer Society, Tech. Rep., 2023.

[11] B. Schleich, N. Anwer, L. Mathieu, and S. Wartzack, "Shaping the digital twin for design and production engineering," *CIRP Annals - Manufacturing Technology*, vol. 66, pp. 141–144, 2017.

[12] A. Nee, S. Ong, G. Chryssolouris, and D. Mourtziss, "Augmented reality applications in design and manufacturing," *CIRP Annals Manufacturing Technology*, vol. 61, no. 2, pp. 657–679, 2012.

[13] H. Lynam, E. Folmer, and S. Dascalu, "Harin: Hololens augmented reality indoor navigation," in *ACIT, May*, 2022.

[14] S. Feuerstack, A. C. M. de Oliveira, M. dos Santos Anjo, R. B.Araujo, and E. B. Pizzolato, "Model-based design of multimodal interaction for augmented reality web applications," *Proceedings of the 20th International Conference on 3D Web Technology*, pp. 259–267, 2015.

[15] C. Wolfe, J. D. Smith, W. G. Phillips, and T. N. Graham, "Fiia: A Model-Based Approach to Engineering Collaborative Augmented Reality," *The Engineering of Mixed Reality Systems. Human-Computer Interaction Series.*, 2010.

[16] S. Gottschalk, E. Yigitbas, E. Schmidt, and G. Engels, "Model-Based Product Configuration in Augmented Reality Applications," *International Conference on Human-Centred Softwae Engineering*, vol. 12481, pp. 84–104, 2020.

[17] M. Bauer, B. Bruegge, G. Klinker, A. MacWilliams, T. Rceicher, S. Rib, C. Sandor, and M. Wagner, "Design of a Component-Based Augmented Reality Framework," *Proceedings IEEE and ACM International Symposium on Augmented Reality*, pp. 45–54, 2001.

[18] J. M. Mota, I. Ruiz-Rube, J. M. Dodero, and I. Arnedillo-Sánchez, "Augmented reality mobile app development for all," *Computers & Electrical Engineering*, vol. 65, pp. 250–260, 2018.

[19] M. Speicher, K. Tenhaft, S. Heinen, and H. Handorf, "Enabling industry 4.0 with holobuilder," in *Informatik*, 2015.

[20] M. Aimonetti, H. Ferrone, T. Sherer, Q. Wen, A. Buck, and V. Tieto, "Introduction to mixed reality development," https://learn.microsoft.com/en-us/windows/mixed-reality/develop/development, last updated: Sep. 20, 2022.

[21] H. Ferrone, V. Tieto, A. Buck, and Kurtis, "Unity tutorials," https://docs.microsoft.com/en-us/windows/mixed-reality/develop/unity/tutorials, last updated: Sept. 22, 2022.

[22] B. Tran, "Framework for Artificial Intelligence Assisted Augmented Reality Systems for Education and Training," Doctoral Dissertation, University of Akron, 2022.

[23] B. Tran and S. Sastry, "Mapping a Virtual View to the Physical World to Guide the Completion of Complex Task Sequences," in *2019 IEEE International Systems Conference (SysCon)*. IEEE, 2019, pp. 1–8.

[24] H. Samet, "The Quadtree and Related Hierarchical Data Structures," *ACM Computing Surveys*, vol. 16, no. 2, pp. 187–260, 1984.

[25] C. L. Jackins and S. L. Tanimoto, "Oct-trees and their use in representing three-dimensional objects," *Computer Graphics and Image Processing*, vol. 14, no. 3, pp. 249–270, 1980.

[26] C. Chien and J. Aggarwal, "Volume/surface octrees for the representation of three-dimensional objects," *Computer Vision, Graphics, and Image Processing*, vol. 36, no. 1, pp. 100–113, 1986.