A Multidisciplinary Approach To Designing Immersive Gameplay Elements for Learning Standard-Based Educational Content

Vinson Luo
vluo@smu.edu
SMU Guildhall
Dallas, Texas, USA

Lawrence J. Klinkert
jklinkert@smu.edu
SMU Guildhall
Dallas, Texas, USA

Paul Foster
pdfoster@smu.edu
Simmons School of Education
Dallas, Texas, USA

Ching-Yu Tseng
etseng@smu.edu
Simmons School of Education
Dallas, Texas, USA

Elizabeth Adams
eladams@smu.edu
Simmons School of Education
Dallas, Texas, USA

Leanne Ketterlin-Geller
lkgeller@smu.edu
Simmons School of Education
Dallas, Texas, USA

Eric C. Larson
eclarson@smu.edu
Department of Computer Science
Dallas, Texas, USA

Corey Clark
coreyc@smu.edu
SMU Guildhall and Department of Computer Science
Dallas, Texas, USA

ABSTRACT

Immersive, educational games may increase engagement and offer new affordances in assessment. Many aspects of instruction may be amenable to gameplay—but the degree of success often depends on student (and instructor) engagement. Commercial games tend to have vibrant engagement from school-aged children, but when transformed for instructional purposes, often break critical immersive gameplay elements or the established protocols of instruction. Moreover, educational game design is limited by the relevance these games have toward existing content standards and the degree to which core gameplay remains intact when educational elements are added. To this end, we present a methodology, called DeCoAD—Decomposition, Connection, and Activity Design, for learning scientists and game developers to design and evaluate immersive game mechanics for specific content standards. The three phases of the methodology guide collaborators in (1) decomposing a learning standard and an existing commercial video game into their basic elements, and (2) identifying the game's potential capabilities to facilitate learning opportunities related to the learning standard. Furthermore, we provide examples from our ongoing experience in creating educational game mechanics for Minecraft, with evaluation from an advisory panel of critical stakeholders.

ACM Reference Format:
Vinson Luo, Lawrence J. Klinkert, Paul Foster, Ching-Yu Tseng, Elizabeth Adams, Leanne Ketterlin-Geller, Eric C. Larson, and Corey Clark. 2021. A Multidisciplinary Approach To Designing Immersive Gameplay Elements for Learning Standard-Based Educational Content. In *CHI PLAY'21: Annual Symposium on Computer-Human Interaction in Play, October, 2021.*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3450337.3483467

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
CHI PLAY' 21, October, 2021, Virtual
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8356-1/21/10.
https://doi.org/10.1145/3450337.3483467

1 INTRODUCTION

A strategy to motivate engagement in learning for students who are digital-natives is employing stealth learning through non-traditional teaching tools, such as video games [19]. There are approximately 2.8 billion video game players worldwide [2] spending over 6 hours, on average, playing video games per week [4]. Harnessing game playing for education is the main aim of digital game-based learning (DGBL) [21]. In creating new DGBL lessons, educators often collaborate with game developers to build a game based on the educators' design [21]. However, many of these educational games struggle to create and sustain engagement [9] because they simply overlay teaching materials with gameplay as an extrinsic reward [13, 20] (sometimes referred to as "chocolate-covered broccoli" [14]). A study showed that without relating the gameplay to the learning content, dissonance occurs when students switch between disconnected contexts of the gameplay and the learning [14], which breaks the long-term state of full immersion and focus, known as flow [10]. Thus, an alternative to developing new educational games is to use existing commercial video games in the classroom [21]. However, it is not immediately apparent how to properly derive gameplay from existing platforms to support learning.

To this end, we propose the DeCoAD methodology, a systematic way to extract learning standard-based educational content from existing commercial video games, and provide students with learning opportunities related to a learning standard. The methodology consists of three phases - Decomposition, Connection, and Activity Design. The Decomposition phase guides the breakdown of a learning standard into observable learning behaviors, and an existing commercial video game into gameplay actions and game functionalities. The Connection phase helps identify overlaps between the learning behaviors and gameplay actions, which informs strategies to facilitate and assess the learning behavior through the game. The final phase, Activity Design, provides guidelines on designing game-based learning activities using the game and companion tools, and integrating them into instructors' curriculum to facilitate a DGBL environment. We hypothesize that using the inherent teaching capabilities and deep engagement techniques of existing commercial video games may dramatically increase student learning engagement and academic proficiency.

We evaluate our approach with a multidisciplinary team of collaborators with expertise in learning science and game design to execute this process. Learning scientists are needed to help design learning opportunities by identifying the cognitive and social practices that students can engage in to build deep conceptual understanding and connections [15]. Likewise, game designers with knowledge of user experience and technical implementation are needed to help create compelling game experiences by considering combinations of systems and dynamics to produce the desired player responses and create long-term engagement [11][16]. This preliminary evaluation covers how the framework was used to identify gameplay and functionalities of *Minecraft* to provide learning opportunities related to a computer science standard. The evaluation also explains how the framework informed the development of a node-based graphical planning application, Minecraft Factory Planner, which aided in addressing curriculum gaps not addressed in Minecraft gameplay.

2 DEFINING THE DECOAD METHODOLOGY

The DeCoAD methodology provides a three-phase process to identify the capabilities of an existing commercial video game to provide learning opportunities related to a selected learning standard, and guidance on using those capabilities to create a DGBL environment. The process uses learning standards to identify the desired learning outcomes, as the standards outline what the students should know and be able to perform related to the standards [8]. This process takes inspiration from the Backward Design Framework [6], where collaborators first identify the desired outcome at the end of the lesson plan, then determine how to assess said outcomes, and finally, plan instruction to achieve said outcomes. Guidelines and recommended artifacts for each phase are provided. Note that while we believe that video games are inherently educational, we do not propose that every educational concept can be extracted from any commercial video game or its elements using the DeCoAD methodology.

2.1 DeCoAD Phase 1: Decomposition Guidelines

The Decomposition phase reflects the first step of the Backward Design Framework to identify the desired learning outcome [6]. In this phase, collaborators identify the learning objectives from the selected learning standard, and the functional components of a video game that potentially provide the learning opportunities related to the standard.

2.1.1 Learning Standard Decomposition. The learning standard decomposition method builds from our previous work and is, therefore, described succinctly here [5]. It consists of several decompositions: (1) **Steps** of a learning standard, or iterative sequence for cognitive processes that students might engage in related to the overall standard. (2) **Importance or Objectives (OIs)** to learning each step of the standard. (3) **Pre-Knowledge, skills, and abilities (pre-KSAs)** that students need prior to interacting with the KSAs so as to engage in learning each step of the standard. (4) **Knowledge, skills, and abilities (KSAs)** that students should develop during the learning process for each step of the standard. (5) **Evidence of Learning (EoLs)**, defined by this research as behaviors that students display to demonstrate that they are learning and progressing towards mastery of each step of the standard, or to demonstrate development of pre-KSAs. (6) **Evidence of Mastery (EoMs)**, defined by this research as behaviors that students display to demonstrate that they have mastered each step of the standard, or to demonstrate development of KSAs.

One suggested approach to this decomposition process is to mirror the Backward Design Framework and identify the evidence related to the learning standard first. For each step, learning scientists first determine the EoMs and the related KSAs, and then the EoLs and the related pre-KSAs.

2.1.2 Video Game Decomposition. For a selected commercial video game, game designers study the game's design, gameplay, and functionalities that may provide learning opportunities related to the learning standard. Collaborators may perform the learning standard and video game decomposition processes concurrently if collaborators have already chosen a game for their selected standards. If a video game has not been selected, the learning standard decomposition may help collaborators find existing commercial video games that have the potential to provide learning opportunities related to the standard components.

The process requires the following functional components of the video game to be identified: (1) **Core gameplay loop**: a set of gameplay actions that players repeatedly perform throughout the game [11][7]. (2) **Secondary gameplay loop**: a set of activities, goals, or challenges for the players to complete by executing the core gameplay loop [11][7]. The secondary gameplay loop provides the circumstances to keep the core gameplay loop compelling [11]. (3) **Tertiary gameplay loop**: a set of activities, goals, or challenges for the players to complete on a long-term, game-wide scale by executing the core and secondary loop [7]. (4) **Mechanics**: the procedures and rules of the game that shape how the players play [11][12]. In a digital game, the mechanics are instructed by its source code to produce specific responses to player input or behavior [11][12]. (5) **Game Stats**: the data or metrics associated with a gameplay or mechanic. The ability for collaborators to proceed with the video game to the following DeCoAD phases is contingent on the collaborators' ability to access its game stats. Public sources of such information include online documentation, forums, and application programming interfaces (APIs). If the developers of the game are also collaborators of this process, the game stats may also be requested from them.

Moreover, it is also suggested to identify the following game-specific elements during the decomposition process. (1) **Pillars**: high-level, action-centric concepts or goals that detail the game's specific functionality and facilitated experiences [11]. In development of the game, its pillars act as guidelines to clarify the game's functional direction [11]. Note that the game pillars focus on the actions that the players take in the game; this provides developers a clear direction to implement functionalities that enable the specific actions [11]. (2) **Themes**: the central ideas of the game as the main takeaways for players, beyond the gameplay [11][16]. Themes serve as a guide to design decisions in development, ensuring that elements thematically fit into the game and reinforce each other [11][16]. (3) **Aesthetics**: the game's expression (visuals, tone, story, etc.) that evokes desired emotions from players when they perceive the game [11][16][12]. To retain the game's integrity and inherent engagement value in later DeCoAD phases, the video game's components must remain mostly unaltered. Additionally, the design components serve to prevent introduction of contradictory elements into designing DGBL activities to avoid disengaging or confusing students.

2.2 DeCoAD Phase 2: Connection Guidelines

This phase reflects the second step of the Backward Design Framework to determine how to integrate the given decomposition lists and, potentially, assess learning progression [6]. In this phase, collaborators identify video game components that may provide learning opportunities related to the learning standard components.

2.2.1 Identifying Overlaps. Learning scientists and game designers collaborate to identify overlaps between the observable behaviors exhibited in learning and mastery of the standard, and behaviors exhibited in the gameplay loops of the video game. For example, collaborators may identify the gameplay action that mirrors or facilitates a learning behavior related to a learning standard component. Then, collaborators form a pairing, which we define as a "learning-facilitating gameplay activity" (LFGA). Collaborators then identify potential mechanics that enable the gameplay action of the LFGA, and the game stats associated with said mechanic. The game stats may exist as stored information, generated output, or data otherwise related to the gameplay mechanic. The connection formed between the components, as shown in Figure 1, informs collaborators how the learning behavior may be quantified by the game stats related to the gameplay that facilitates the behavior. The learning behaviors exhibited in the learning standard components may be mapped to various layers of gameplay loops- core, secondary, and tertiary, depending on task complexity, time scale, and abstraction of the learning behaviors and gameplay loop actions. Collaborators may adopt any methodology to discuss how the components connect.

Figure 1: LFGA mapping between learning behavior exhibited in learning standard components to video game components, and LFGA mapping between unpaired learning standard components to companion tool.

2.2.2 Supplementing for Gaps. After integration, there may be learning behaviors associated with learning standard components that are not paired with a gameplay action because the game cannot naturally facilitate them. Note that the game does not need to be altered to address these gaps. Rather, these unpaired learning components inform the development of tools supplementary to the game to address these unpaired learning components, thereby more fully addressing the learning standard. As shown in Figure 1, collaborators may identify new LFGAs by defining the functionalities to fill the gaps, the features that support said functionalities, and the associated tool data that may be used to quantify the unpaired learning components. When designing the companion tools, collaborators need to consider the video game's pillars, themes, and aesthetics to make the tools' interaction loop compatible with the game. If collaborators struggle to identify overlaps between the learning standard and the video game, and also struggle to address the gaps, the game may not be appropriate for the selected learning standard. Collaborators should consider another video game and return to the Decomposition and Connection phases.

2.3 DeCoAD Phase 3: Activity Design Guidelines

This aligns with the third and final step of the Backward Design Framework to plan instructions that will achieve the desired outcomes [6]. In this step of the process, collaborators design gameplay activities to integrate into instructors' curriculum that will provide learning opportunities related to the learning standard. We do not propose any lesson format that will replace the instructors' lessons or restructure their curriculum. Rather, we suggest that collaborators communicate to instructors how the video game and the companion tools may be integrated into their curriculum in ways that suit the instructors' needs. We suggest the following outline: (1) **Objectives** of the activity plan. (2) **Materials** for the activity plan. (3) **Game-based Activities**: relating to the learning standard that students will perform. (4) **Assessment Opportunities**: for instructors to monitor student learning progress and differentiate instruction. Collaborators may expound the outline above with any detail that suits their needs. We suggest using the Decomposition and Connection phase findings, as follows: (1) Use the **OIs** of the learning standard to describe the Objectives of the activity plan. (2) Describe the video game and the companion tools, as well as the game's **mechanics** and the companion tool's features, as Materials of the activity plan. (3) For Activities, describe the **gameplay** activity or the interaction with the companion tools. The activities should be arranged in an order of execution appropriate for both learning and game progression. (4) Provide **game stats** related to the Activities above for Assessment Opportunities. Create instructor access to game stats related to students' progress toward mastery of the learning standard based on evidence collected during gameplay. Add descriptions to help instructors interpret the game stats.

3 EXAMPLE IMPLEMENTATION WITH MINECRAFT

For the project on Teaching Computer Science and Computational Thinking with Gaming, a multidisciplinary team was formed to facilitate learning of computational thinking in a *Minecraft* environment [3]. The team consisted of learning scientists, computer scientists, game designers, and game developers [5]. We present an ongoing evaluation of our method with an educator advisory panel (EAP) of five educators representing four public school districts in northern Texas. Three educators identified as teachers, one identified as an instructional coach, and one identified as an instructional technology specialist. At each stage the EAP provided guided feedback on (1) the decomposition of the standards, (2) the realism of implementing the game in an instructional format, and (3) the opportunities and challenges of assessment. The team also partnered with Mineplex's developers for porting *Lumberjack Tycoon* to *Minecraft* EDU servers, and for consultation on the mod. To further facilitate integration, the team built a backend communication pipeline using AWS Lambda to connect *Minecraft* to a learning management system. We choose computer science for middle school as our initial standards because these tend to be understood by both game developers and learning scientists. Though in our experience, these groups have a different set of assumptions and associated vocabulary. In particular, we focus on standard 2-DA-08 of the Computer Science Teachers Association [17, 18]: "Collect data using computational tools and transform the data to make it more useful and reliable." This standard is part of a larger set of teaching standards that also include its impact in computing topics such as problem decomposition and equity. All human subjects protocols and verbal consent procedures were approved by the University IRB, protocol H19064LARE.

3.1 Applying DeCoAD Phase 1: Decomposition

The team's learning scientists first decomposed 2-DA-08 into high-level steps- Collect, Clean, Organize, and Explain data. Each step was broken down into their respective tasks, which were compiled into a spreadsheet as the standard decomposition table, as shown in Figure 2 (top two rows). Simultaneously, a game designer from the team compiled a game decomposition document for *Lumberjack Tycoon*, containing a breakdown of the game's key features. Since *Lumberjack Tycoon* is a *Minecraft* mod, the information was collected from the Official Minecraft Wiki, a community-driven online encyclopedia that covers *Minecraft* in detail, including its gameplay loops, mechanics, and game stats. This decomposition document was then distributed to all collaborators for review before integration in Phase 2.

3.2 Applying DeCoAD Phase 2: Connection

With the decomposition of the learning standard, identification of critical game loops, and brainstormed game mechanics, the aim shifts to identifying overlaps or gaps between decompositions. To this end, collaborators constructed the KSA versus gameplay mechanic matrix in Figure 2 (top matrix), whereby learning scientists and game developers evaluated to what degree the initial game decomposition addressed each standard decomposition. Collaborators were asked to score each game element as "fully addressed," "partially addressed," or "not addressed" for each standard decomposition. This process revealed that only two of the game mechanics fully addressed two KSAs. Gameplay elements with scores of "partially addressed" or lower were either discarded or slated for updating. Moreover, KSAs not fully addressed were given priority in the next brainstorming session.

In this first pass of identifying overlaps (Figure 2, top matrix), the team realized that *Lumberjack Tycoon* could not naturally facilitate all of the learning behaviors related to 2-DA-08 - for example, creating graphs such as bar charts, which was an EoL for the "Organize data" step. While players in *Lumberjack Tycoon* could create a structure that resembles a bar chart by stacking blocks at various heights, this is not a gameplay that fits the game's theme of exploration and crafting for efficiency. The gaps informed the team that there is a deficiency in *Lumberjack Tycoon*'s ability to facilitate thematically consistent activities related to graphing for data visualization.

This led to the development of a companion node-based graphical planning tool called Minecraft Factory Planner [1]. It can access players' *Lumberjack Tycoon* data and game stats, such as events, inventory, tools, and buildings through the AWS Lambda backend communication pipeline. To reach a target quota on an item in a given time frame, players can use Minecraft Factory Planner prior to playing the game to create node maps, as shown in Figure 3, that display the number of resources they need to collect in a time frame to reach the desired production rate of the item. The team also implemented a graphing feature to address the aforementioned gap related to creating graphs in *Lumberjack Tycoon*; players can display data such as resource collection rates, item production rates, and production output in the forms of line, bar, and pie charts. Finally learning scientists and game developers met a further time to identify new game mechanics afforded by the Minecraft Factory Planner, that more fully addressed the learning standard decomposition. Various ways to integrate the companion app were ideated, as shown in Figure 2 (lower matrix row titles).

3.2.1 *Evaluating Integration with Advisory Panel.*

With the new set of game mechanics, we presented the game mechanics to our EAP, soliciting their opinion about the degree to which each KSA of the standard was addressed. This gave the EAP a second pass at reviewing overlap and gaps. To facilitate this, the game developers created mock-ups of the game and companion app (Figure 3) and demonstrated these mechanics to the EAP. Virtual polls and EAP comments were used to understand the relevance and appropriateness of the game mechanics for each KSA. Each comment from an EAP member was coded according to whether it supported, refuted, or was neutral towards a game mechanic addressing a KSA. In particular, Figure 2 (lower matrix) shows the output of this coding: Cells are highlighted green if 100% of the EAP agreed a game mechanic supported a KSA; yellow if 75-99% agreed; orange if 50-74%; red if less than 49%. Dark green indicates that members of the EAP commented in favor of the game mechanic, but a survey response did not directly address this area. At the end of our analysis, gaps remained in "identifying trustworthy source," "identifying anomalies," and "understanding a model." When panelists were asked how appropriate each game mechanic was for their grades level, four panelists rated the mechanics as "Completely Appropriate" and one as "Somewhat Appropriate." Subsequent discussion revealed this panelist was concerned students might not be familiar enough with gaming on a keyboard or with Minecraft gameplay. After this evaluation, we felt comfortable moving forward in the design of the *Lumberjack Tycoon* and Minecraft Factory Planner into an appropriate activity.

Knowledge, Skills, and Abilities

2-DA-08: Collect data using computational tools and transform the data to make it more useful and reliable	Collect the Data		Clean the Data					Organize the Data				Explain the Data					EAP Notes
	Decide data needed, how to collect and store	Execute the plan	Remove irrelevant data	Transform the data to remove errors	Identify trustworthy sources	Identify anomalies in graphs	Note / fix missing, miscoded, duplicate data	Abstract and summarize data representations	View, group, and classify the data	Depict data in useful and reliable ways	Create system/methods to represent data	Identify patterns to support hypotheses	Interpret visualizations	Understand what a model is	Represent information based on usage	Researcher Notes	
Design Element: Phase I – Initial Ideas from Brainstorming Sessions																	
Students chop down trees as fast as they can and observe amount of materials in their inventory.	Partially Addressed	Likely Addressed	Partially Addressed	Partially Addressed					Partially Addressed	Partially Addressed						Fully Addresses One or More KSA. Keep Element.	Recommended students have opportunities to share with others.
Inventory that the students collect (i.e., number of trees in the world, etc.) is fed to outside source as "data" via sensors or measuring blocks.	Partially Addressed			Partially Addressed				Partially Addressed		Partially Addressed					Partially Addressed	Only Partially Addresses KSAs. Pivot Idea: All data in game exposed in companion app?	
Use pre-made modular blocks in Minecraft that allow students to start with instructor created designs.										Partially Addressed							Can we limit the guessability of answers?
Determine types of trees that should be harvested to produce the largest amount of planks/wood.	Likely Addressed											Partially Addressed				Fully Addresses One or More KSA. Keep Element.	Teachers would like the ability to ask assessment questions in the companion app.
Allow creation of student modules within Minecraft, adding these systems to the collective student "Library."											Partially Addressed					Minimal KSAs partially addressed. Consider discarding or significantly revising element.	
Allow non-playable characters to be a resource for the world, automating mining processes.											Partially Addressed						
Gaps:	Likley No Gap	Likely No Gap	Significant Gap Found	Slight Gap Found	Complete Gap Found	Complete Gap Found	Complete Gap Found	Significant Gap Found	Significant Gap Found	Slight Gap Found	Significant Gap Found	Significant Gap Found	Complete Gap Found	Complete Gap Found	Significant Gap Found		
	Analysis: Large gaps exist in cleaning and organizing data that have downstream influence on hypothesis testing. **Main Pivot**: Creation of an Emulator/Companion App that allows students to decompose problems, create designs, and update through iterations to solve posed problems with gameplay.																
Design Element: Phase II – After Further Integration with Advisory Panels																	
Students enter companion app to see automatically generated graphs of resources they have collected in Minecraft. Compare performance to other groups.	(5/5) EAP members agree is relevant to this KSA.		(5/5) EAP members agree is relevant to KSA.					EAP commented that companion application facilitated data representation		(5/5) EAP members agree is relevant to KSA.		EAP commented that graphs help students see patterns among variables, but might not connect to hypothesis	(5/5) EAP members agree is relevant to KSA.		EAP commented that KSA was well integrated	Fully Addresses One or More KSA. Keep Element.	One teacher confirmed that checking in app supported assessment.
Students use companion app simulator to visualize item creation. They forecast trees that should be harvested to produce the largest amount item.	(5/5) EAP members agree difficulty is appropriate	(5/5) EAP members agree they could measure learning/ mastery							EAP commented that companion application allowed students to view and interpret graphs							Fully Addresses One or More KSA. Keep Element.	One teacher suggested using [closed] questions with definitive answers to check for understanding
After auto-collecting resources, return to Companion App to view actual relevant and irrelevant material collected vs what they predicted they would collect.											EAP commented that app allowed creation of custom representation		3/5 EAP agree they could measure learning/ mastery			Game Mechanic Needs to Be slightly Refactored to Address KSAs more fully.	
Student use Companion App to determine best resources for Lumber Mill location. Then they explore Minecraft to identify the best site in game to build Lumber Mill	(5/5) EAP members agree is relevant to KSAs.	(5/5) EAP members agree difficulty is appropriate		(5/5) EAP members agree they could measure learning/ mastery		(5/5) EAP members agree is relevant to KSA.				(5/5) EAP members agree is relevant to KSA.		EAP suggested students compare and contrast to other's choices for hypothesis support	(5/5) EAP members agree is relevant to KSA.			Fully Addresses One or More KSA. Keep Element.	EAP suggested process measures (e.g., attempts, key strokes) to determine engagement
	Already Addressed.		Addressed	Addressed	Gap	Addressed	Gap	Likely Addressed	Likely Addressed	Addressed	Likely Addressed	Likely Addressed	Addressed	Gap	Likely Addressed		
	Analysis: Most gaps filled. **Main Pivot**: Potential need to address the coding and identifying of duplicate data. Alternatively, might be better instructed using supplementary resources. Need to also address modeling and trustworthy sources. **Next Step**: Assess these gameplay elements for immersion: are these elements true to the existing Minecraft game loop? Can these KSA's be translated to the EoL/EoM and assessed?																

Figure 2: The proposed KSA versus Gameplay Mechanic matrix. The top of the matrix summarizes gaps found from the original brainstorming session. The bottom matrix summarizes an evaluation of additional gameplay elements designed to help fill the gaps identified. Evaluation of gaps was assessed by our educator advisory panel.

Figure 3: Screenshot taken in Minecraft Factory Planner

3.3 Applying DeCoAD Phase 3: Activity Design

After selecting proper overlaps between the game and learning standards, collaborators designed a high-level DGBL activity plan to facilitate learning opportunities related to 2-DA-08 and the sequence of those activities for the appropriate game and learning progression, using the identified game elements from Figure 2. Collaborators first determined the *Lumberjack Tycoon* gameplay equivalent of each step of 2-DA-08. Collaborators first developed a question related to the "Explain data" step in the context of *Lumberjack Tycoon*, "what biome should a lumber harvester be built in to have the highest production rate of wood blocks needed to most efficiently manufacture for a craft order, such as bookshelves?" After identifying the end goal, collaborators worked backward to elaborate the steps to achieve it—players first receive a craft order, determine the resources needed, build systems to collect the data, then process and visualize the collected data to assist in explaining their strategies and choices. Collaborators also planned tutorial activities for *Lumberjack Tycoon* and Minecraft Factory Planner to onboard students and provide them with the necessary knowledge and skills to engage in more advanced activities. Collaborators created an activity plan shown in Table 1.

4 DISCUSSION AND CONCLUSION

The DeCoAD methodology combines learning science and game design expertise to extract learning standard-based educational content from an existing commercial video game. It uses overlaps between the gameplay actions of a video game and learning behaviors exhibited in a learning standard to design activities that instructors may integrate into their curriculum to facilitate a DGBL environment. The project on Teaching Computer Science and Computational Thinking with Gaming employed the DeCoAD methodology to reveal a number of gameplay mechanics in *Lumberjack Tycoon* that facilitate learning of 2-DA-08. The process also informed

Activity	Objective	Material	Assessment
Student introduction to collecting resources and crafting in *LT*	Identify and collect relevant data to solve a problem	*LT*: Harvesting mechanic	Students demonstrate understanding of relevant data by collecting resources related to the problem. (Compare the namespaced IDs of collected material.)
Student introduction to *MFP*	Transfer data to a format that is easy to access for a computer and human	*MFP*: access game data, data visualization	Students demonstrate an understanding of formatting data as a graph by selecting an appropriate graph to display information. (Compare graph type.)
Students creating a crafting plan node map in *MFP*	Organize objects by comparing them	*MFP*: node-based mapping	Students demonstrate an understanding of organizing objects by comparing crafting plans with classmates based on the structure of the design.
Students validating their crafting plan in *LT*	Identify and collect relevant data to solve a problem. Find patterns and relationships in data.	*LT*: Harvesting mechanic, Inventory	Students demonstrate an understanding of identifying and collecting relevant data by visiting biomes that contain more of the needed materials and collecting them.
Students receive an advanced crafting order to complete in a limited time as a mastery exercise	Link explanations based on the data to the why or purpose	*LT*: Automatic harvesting buildings mechanic *MFP*: access game data, data visualization	Students demonstrate an understanding of linking explanations based on data to the purpose by explaining why they selected a location for their lumbermill based on data such as production rates. (Access students' buildings to see if placed in correct biomes with high yield.)

Table 1: Simplified DGBL Activity Plan. *LT* stands for *Lumberjack Tycoon* and *MFP* for Minecraft Factory Planner

development of the Minecraft Factory Planner, a companion tool to *Lumberjack Tycoon* with functionalities and features that help the game more fully address 2-DA-08. The team then designed a DGBL activity plan for 2-DA-08 using *Lumberjack Tycoon* and the companion tool Minecraft Factory Planner to be integrated into K-12 curriculum. While this paper provides suggested approaches to the process, as well as an example application to one learning standard and video game, collaborators may incorporate any methodology into the process for other standards and games to suit their needs. A limitation of our approach is that it does not specify how an appropriate video game might be chosen for use in the DeCoAD method, nor do we address the costs associated with the method in terms of time and effort. While the method successfully facilitated a collaboration of learning science and game design expertise, the time associated may be prohibitive for many applications. We thank our EAP, collaborators at Mineplex, Microsoft, and support from NSF grant #1933848.

REFERENCES

[1] 2021. Minecraft Web App. (2021). https://gitlab.com/humin-game-lab/minecraft-web-app.git [Online; accessed 2021-03-31].

[2] 2021. Newzoo's Key Numbers | Games, Esports, Mobile. (2021). https://newzoo.com/key-numbers/ [Online; accessed 2021-02-11].

[3] 2021. Teaching Computer Science and Computational Thinking with Community Gaming. (2021). https://humingamelab.com/Research/Projects/research/projects/Teaching-Computer-Science-and-Computational-Thinking-with-Community-Gaming.html [Online; accessed 2021-02-11].

[4] 2021. The State of Online Gaming – 2020. (2021). https://www.limelight.com/resources/white-paper/state-of-online-gaming-2020/ [Online; accessed 2021-02-11].

[5] E. L. Adams, C. Tseng, P. Foster, V. Luo, L. R. Ketterlin-Geller, E. C. Larson, and C. Clark. 2021. A standard decomposition process to inform the development of game-based learning environments focused on computational thinking. *International Conference on Computational Thinking and STEM Education* (2021).

[6] A. Boozer. 2014. Planning Backwards to Go Forward: Examining Pre-service Teachers' Use of Backward Design to Plan and Deliver Instruction. (2014).

[7] C. Clark, I. Greenberg, and M. Ouellette. 2018. A Model for Integrating Human Computing into Commercial Video Games. *2018 IEEE 6th International Conference on Serious Games and Applications for Health (SeGAH)* (2018), 1–8. https://doi.org/10.1109/SeGAH.2018.8401316

[8] National Research Council. 2014. *Knowing What Students Know: The Science and Design of Educational Assessment.* The National Academies Press.

[9] R W. Crues, N. Bosch, M. Perry, L. Angrave, N. Shaik, and S. Bhat. 2018. Refocusing the lens on engagement in MOOCs. In *Proceedings of the fifth annual ACM conference on learning at scale.* 1–10.

[10] M. Csikszentmihalyi. 1990. *Flow : the psychology of optimal experience.* New York : Harper and Row.

[11] W. Despain, K. Acosta, L. Canacari-Rose, M. Deneen, Z. Hiwiller, J. Howard, C. Kadinger, C. Keeling, and C. Kuczik. 2012. *100 Principles of Game Design.* New Riders.

[12] R. Hunicke, M. LeBlanc, and R. Zubek. 2004. MDA: A formal approach to game design and game research. 4, 1 (2004), 1722.

[13] B. Marklund and A. Taylor. 2016. Educational Games in Practice: The challenges involved in conducting a game-based curriculum. *Electronic Journal of e-Learning* 14, 2 (2016), pp122–135.

[14] European Union News. 2014. Kids Aren't Fooled By "Chocolate-Covered Broccoli". *Gale General OneFile* (2014). [Online; accessed 2021-02-11].

[15] R. Keith Sawyer. The New Science of Learning. In *The Cambridge Handbook of The Learning Sciences*. Cambridge University Press; 1st edition.
[16] J. Schell. 2014. *The Art of Game Design: A Book of Lenses.* CRC Press.
[17] D. Seehorn and L. Clayborn. 2017. CSTA K-12 CS Standards for All. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. 730–730.
[18] D. W. Seehorn, S. Carey, B. Fuschetto, I. Lee, D. Moix, D. O'Grady-Cunniff, B. Owens, C. Stephenson, and A. Verno. 2011. CSTA K–12 Computer Science Standards: Revised 2011.
[19] L. A. Sharp. 2012. Stealth Learning: Unexpected Learning Opportunities through Games. *Journal of Instructional Research* 1 (2012), 42–48.
[20] V. Shute, S. Rahimi, and X. Lu. 2019. Supporting learning in educational games: Promises and challenges. In *Learning in a Digital World*. Springer, 59–81.
[21] R. Van Eck. 2006. Digital game-based learning: It's not just the digital natives who are restless. *EDUCAUSE review* 41, 2 (2006), 16.