Open Game Data: A Technical Infrastructure for Open Science with Educational Games

David J. Gagnon(✉) and Luke Swanson

University of Wisconsin – Madison, Madison, WI 53706, USA
{djgagnon,lwswanson2}@wisc.edu

Abstract. In this paper we describe a technical infrastructure, entitled Open Game Data, for conducting educational game research using open science, educational data mining and learning engineering approaches. We describe a modular data pipeline which begins with telemetry events from gameplay and ends with real time APIs and automated archival exports that support research. We demonstrate the usefulness of this infrastructure by summarizing several game research projects that have utilized and contributed back to Open Game Data. We then conclude with current efforts to expand the infrastructure.

Keywords: Learning Games · Assessment Games · Learning Engineering · Game Data · Educational Data Mining · Open Science · Research Infrastructure

1 Introduction

Effective human collaboration and coordination may well be the most powerful force in the universe. In this paper, and the project it describes, we contribute motivations and tools for collaborating and coordinating educational game research. This project is a direct response to both the promise of learning games for teaching, learning and research, as well as the challenges faced by game researchers.

One of the most obvious challenges is made clear in the 10 min after every talk the authors have given, when inspired graduate students come up to us and ask, "How do I get involved?" We are not satisfied by the most obvious answer to this question, which is to relocate oneself to one of the handful of large research institutions that have an in-house game studio, secured distribution, built research infrastructure and have an established research team with expertise in data science. What is the alternative? For many, it is to develop small versions of all these components once again, leading to educational game projects that are designed by novices, ad hoc infrastructure, small sample sizes and narrow studies. But what if there were a way to collaborate? What if there was a way to build from ongoing, well designed game projects with existing audiences, datasets and even sample analysis code so that the researcher could focus on their experiment and not all the apparatus required? What if a studio could adopt data driven design practices and better understand their players without having to develop all their own technology and in house expertise?

M. Haahr et al. (Eds.): JCSG 2023, LNCS 14309, pp. 3–19, 2023.
https://doi.org/10.1007/978-3-031-44751-8_1

This is the pursuit of Open Game Data: to open educational games research to a wider audience than we see today. We are looking to decouple learning game studios and research teams, so they don't need to be at the same organization. We are looking to modularize technical infrastructure so that a given studio or researcher can leverage useful portions of a data collection and analysis pipeline that serve the unique needs of their project, getting 99% of what they require for free and only having to contribute the final 1%. In the end, we believe that this collaboration will accelerate discovery in fields such as human computer interaction, machine learning and learning science, and in the end, produce better, more effective learning experiences for students.

The approach is not just an idea, it is happening today. Currently, a shared infrastructure is already collecting millions of rows of research data daily, from numerous learning games, and making those data and related analysis approaches available online for free. Open-source libraries are available for studios to integrate into their game. Hundreds of datasets are available for download, and many have code samples to explore, analyze and visualize the data. It's time to collaborate and coordinate!

2 Background

2.1 Games for Learning

While there are certainly critiques of digital games for learning [28], the overwhelming evidence favors their usefulness when applied appropriately. Educational games have been empirically shown to enhance students' understanding of targeted concepts, as well as their information processing skills [4, 5]. Games and situated virtual simulations have also been linked to increased motivation and interest [20, 22].

Scientifically based simulations and games provide contexts for representing and interacting with the world. When properly designed, they allow players to affect, experiment, and observe consequences in the simulated world in ways that are representative of the real world. Thus, simulations and games can produce situated learning experiences tied to disciplinary-specific concepts and language [9, 37]. James Gee [9] posits that well-designed games can situate players perceptually, narratively, and socially in a way that leads to empathetic embodiment for complex systems, thus prompting players to develop new ways of thinking, knowing, and being in the world, what Shaffer [32] calls an "epistemic frame". Because players' actions within simulations are mediated, reducing danger and cost while expanding or compressing time, perspective, etc., games provide a useful means of teaching complex concepts, particularly ones that cannot be experienced directly [13]. A challenge, then, is to produce designed experiences that align and balance context, goals, challenges, and feedback with desired student behaviors and learning outcomes [18].

2.2 Videogames as Instruments for Skills and Knowledge

There is increasing interest in using games as instruments and assessments [15]. This interest in grounded in the facts that: (1) games can engage players in authentic situations that resemble what they will encounter in real-world situations and (2) the telemetry of digital games can allow for collecting rich data that can be used to build computational models that reconstruct the entire problem-solving process, instead of only looking at the final outcomes of the problem.

Frameworks such as evidence-centered design (ECD) [21] formalize a method for designing games as psychometrically valid evaluations, leading to what Shute et al. [34] calls stealth assessment. This approach has been successfully used to assess conceptual understanding of academic topics, as well as persistence, and problem-solving skills using game play interaction data [42].

2.3 Analytics and Educational Data Mining

An alternate, but compatible approach to evidence centered design leans into the affordances created by the amount of data produced by games and the use of machine learning to extract insights. This approach, commonly referred to as educational data mining [35] uses fine-grained interaction data from a digital system to support educationally relevant conclusions. The method does not necessarily require the game to be designed in any particular way, but “mines” the data that is available to build models and draw inferences.

Supervised approaches utilized “labeled” data to train models. These labels can be internal to the game, such as a “quit” event where the player leaves a challenge before it is completed. The labels can also be provided by a researcher based on observation of the player or game play. In either case, models are trained using numerical or categorical “features” that describe some segment of gameplay along with the labels. If successful, the model can take a new set of features and accurately predict the outcome label. Recent work has demonstrated the use of supervised educational data mining approaches with game log data to effectively predict quitting, off-task behavior [2], wheel-spinning [12, 26], detect the learner’s goal [6], as well as assess the learners science content knowledge [8, 30].

Unsupervised approaches provide insights into players without the use of labeled data for training. For example, clustering analysis has been used to develop a taxonomy of play “styles” [38], and latent class analysis has been used to develop learner typologies in relationship to learners’ prior knowledge and domain interest [36].

2.4 Open Science for Education Research

Open Science is a movement that seeks to leverage new practices and digital technologies to increase transparency and increase access in scholarly research. It can be summarized as a form of scientific enterprise in which every step of the scientific endeavor is made available for others to inspect, critique and reproduce. Open Science is a response to a series of issues that have been raised with the way science is conducted and the latent structures that exist within scientific practice. These problems include high failure rates of replication studies [19], publication bias [29], high rates of false-positives [11], and cost barriers to accessing scientific research [23].

Open science for education encourages sharing artifacts at each stage of research [17, 40]. While specific terms may differ slightly, the following components of open science are common (0) Preregistration wherein researchers document their study protocol before executing the study [41]; (1) Open access to make research results available after publication; (2) open data as a way to publish the raw data before analysis; (3) Open Analysis to share the process by which the final results were obtained from the raw data; (4) Open Materials ensure that the tools and instruments used as an intervention or to collect data are available. Researchers are thereby encouraged to be as transparent as possible during the research endeavor and widely share artifacts outside of their immediate research team.

The transparency of Open Science not only combats some of the problems mentioned above by making every part of the research available for critique, but also may serve as a catalyst to accelerate inquiry and diversify those involved. Research that adopts any of the open practices also decouples the expertise (and cost) required to participate in that phase of research from the others. The availability of Open Data allows researchers to repurpose data in pursuit of a novel research agenda from what was originally in mind when the data was collected, saving significant cost for the secondary use. Similarly, Open Materials allow researchers to reuse interventions and instruments to be repurposed for new audiences, contexts or modified to support new research without having to develop these materials from scratch. Open Analysis allows potentially complex analysis procedures to be leveraged with other data, bootstrapping the efforts of the new research project.

2.5 Open Game Data as a Platform for Open Science with Game Data

Open Game Data was initially developed at the Field Day Lab at University of Wisconsin-Madison to explore data being generated by their own games. Following years of informal collaboration with researchers at Carnegie Mellon and University of Pennsylvania, a few larger research projects required game datasets to be able to be moved quickly between institutions to support the collaboration. Open Game Data was developed to serve those immediate needs while beginning to tun attention to expanding the network of who was collaborating.

Open Game Data is an online infrastructure designed to facilitate several components of Open Science, namely Open Data, Open Analysis, and Open Materials. The technical system is designed to capture, store, transform and disseminate player interaction data from learning games. A website provides links to game source code, publications based on these data and analysis code utilized by those investigations. This infrastructure has already evolved through several iterations and it's own progress has been documented and disseminated openly at https://github.com/opengamedata.

Open Game Data builds on the insights of previous work. It borrows the ideas of modularity between data providers, data stores and data consumers as well as the definition of a cross-project logging schema from xAPI [1], but is not well suited for high-frequency data created by games due to the verbose JSON structure. Open Game Data is also inspired by Data Shop [16] to create a public repository of learner datasets that are used by researchers beyond the initial authors but is most appropriate for instructional tutor data. Finally, this work builds upon the goal of a reusable system for analyzing educational game data that was demonstrated in ADAGE [25], but has since been abandoned.

3 Open Game Data Technical Infrastructure

We now outline the technical elements of the Open Game Data infrastructure. This includes a general pipeline for event data processing, a set of design decisions to ensure the data and pipeline work well across game genres and levels of granularity of research, and a description of our implementation of this design.

3.1 Levels of Granularity in Game Data Analysis

The goal of this infrastructure is to enable analyses that provide an understanding of game player experiences. We support such research by generating data suitable for a diverse range of analytical approaches. This may include anything from simple statistical summaries of a group of players to full in-engine replay of an individual gameplay session.

These approaches generally require the ability to track interactions between a player and a game, potentially across multiple sessions or repeated playthroughs. Further, different approaches often require different levels of granularity. The Open Game Data pipeline directly supports four levels of granularity: event, session, player, and population. This is not a comprehensive list of all possible granularities but represent the most common use cases we have encountered.

- **Event data:** An event represents a single interaction between player and game. This could be an action taken by a player, such as clicking a button; a reaction by the game system, such as a change in a score display; or a marker of progression, such as the player completing a level [24]. We further identify two classes of events:
  - *Game events*: These events represent objective moments of gameplay, encoded in the game system itself. These are the most common kind of event data, and include click and navigation events, changes in the game state, and movement of the player through a game's progression system.
  - *Calculated events:* These are derived events that are not explicitly a part of the game. These events can be generated during analysis of the game events. This category includes events that are based on arbitrary thresholds, or whose definitions might change over time. For example, an "idle player" would be best implemented as a calculated event.
- **Session data:** This level of granularity comes from aggregating events across a gameplay session, in order to extract some understanding of the session as a whole. We define a session as the time from a player opening the game application to the time the application is closed. It is generally assumed that a session represents only one player's interaction with the game; however, it is possible multiple players could take turns in a single session.
- **Player data:** In the same way session data is generated by aggregating events across one session of gameplay, player-level data is generated by aggregating events across all of a player's sessions. For some games, session and player data have a one-to-one relationship; that is, the game does not keep track of a player across sessions, assuming instead that each session is a unique, one-time player.

- **Population data:** In this case, events are aggregated to generate data about a population of players. Practically, this data tends to be based on aggregations of player-level data, rather than raw aggregation of events. For example, where a player-level feature might be the total score on a puzzle, an equivalent population-level feature may be the average player score.

3.2 A Pipeline for Event Data Processing

In order to provide data for analysis at each level of granularity, we have developed a pipeline for the collection, processing, and distribution of game event logs. Our pipeline begins with the collection and storage of raw game events, and includes a series of data transformations, yielding calculated events and feature data. It ends with distribution of data suitable for games researchers and data scientists to perform analysis, model-building, or data visualization.

This pipeline is agnostic to game genre and allows for complete customization of the set of calculated events and features for each game of interest. Further, our pipeline is independent of any specific data storage solution or game platform. We achieve this flexibility by enforcing a specific schema for all events, creating a common structure for inputs to the pipeline. So long as a mapping can be defined to place existing event data into the form defined by the schema, our pipeline provides a uniform interface for processing events into the granularities discussed in Sect. 3.1 (Fig. 1).

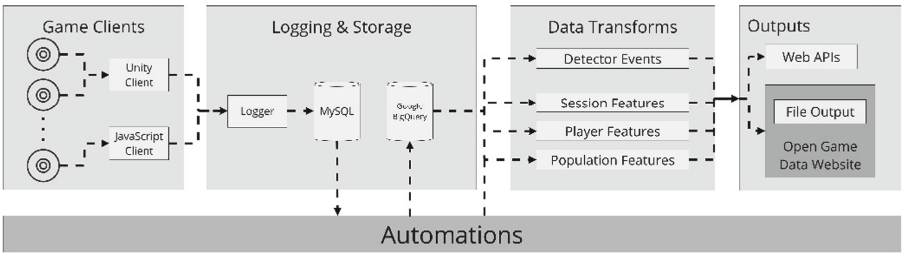

Fig. 1. The Open Game Data pipeline consists of four segments, namely game clients, logging and storage, data transforms, and data sharing.

The Open Game Data Event Schema
There are two goals in designing a single standard for event logs. The first, as discussed previously, is to allow for a single interface to process event log data. The second is to ensure that gameplay is logged in sufficient detail to allow a full reconstruction of the session. Any given event log should answer a few questions, including when did the event occur? what happened? who was the player? where was the player in the space of possible game states? which version of the game was being played? Our schema is designed to ensure these questions are answered by each event. The full list of schema elements is included in Table 1. These elements can be grouped into the following categories:

- **Identifiers:** We include three identifiers with each event, namely an *app ID*, a *session ID*, and a *player ID*. The app ID indicates which game logged the event. Session IDs are uniquely generated each time a game application is started and are included with each event until the application is closed. Lastly, the player ID is an optional identifier, which may be used by games that implement a player code or similar mechanism to track users across multiple sessions.
- **Timestamps:** The *timestamp* category includes an eponymous UTC timecode indicating when an event occurred in absolute time, as well as an *event sequence index*, sometimes called "session index". The first event within a session is given a value of 1, the next 2, and so forth, ensuring that the precise order of events can be reconstructed. This addresses a potential issue where some storage systems do not with sufficient granularity to uniquely order the events in a session.
- **Versioning:** Games regularly change over time as developers add new content or tune existing content to create a better play experience. Thus, it is necessary to include versioning information with event logs. The schema includes three "versions." *App version* indicates the sequential version of the game itself. An optional *app branch* (sometimes called "app flavor") may be included. It is often useful to distinguish between official release versioning and temporary experimental versions, as in A-B testing, where multiple variations of a game are deployed in parallel. Lastly, *log version* indicates a version of the logging schema and internal structure of the *event data* or *game state* elements for a game. These may change in response to changes to the game itself, or to address shortcomings of the earlier logging code.
- **User Data:** Beyond merely identifying the player, it may be helpful for some games to track general information about a user. This may vary from game to game, so our schema defines a *user data* element as a dictionary, possibly empty, that maps sub-element names to values. The choice of sub-elements is completely left to the implementer, but all chosen sub-elements should be included with every event. That is, regardless of the specific details of an event, the user data should always have the same structure.
- **Game State:** It is often useful to understand the game-specific context in which an event occurred. The schema thus includes a *game state* element. *Game state* is a dictionary mapping sub-element names to values. These could include the current level, the player's current score, or the position of the player character on the game map. Again, the specific structure of the game state dictionary is decided on a game-by-game basis, but within a given game the structure should not vary from event to event.
- **Event Data:** This category includes an *event name*, an *event source*, and *event data*. The *event name* is a string indicating the specific type of event that occurred. Examples might include text_click, level_complete, or display_image. We recommend a convention using a noun_verb format (e.g. text_click) to name player action and progression events, and a verb_noun format (e.g. display_image) for system feedback events. *Event source* simply indicates whether the event was generated by the game, or created in post-processing. Finally, *event data* is a name-value dictionary, like *game state* and *user data*. Unlike those other elements, the internal structure of *event data* will vary across events. For example, a text_click might have only a single sub-element containing the text that was clicked. On the other hand, a display_image

event might include sub-elements for the name and dimensions of the image file being displayed. Note that all other elements of the schema have been common across all events in a game. Regardless of a game's details, every event includes identifiers, a time, and version information. Based on the details of a specific game, we can define data that describes a user or a game state, which again has a common structure across all of that game's events. In this category of schema element, we finally address specific details that vary by event type. This is a conscious design choice; by isolating event-specific details to a few elements, we are able to maintain a generally uniform interface for the pipeline to interact with event data.

Game Clients

Processing of event data necessarily begins with the collection of event logs from the game itself. Logging should be loosely coupled with the game, such that it does not impact the player's experience in any meaningful way. To this end, our infrastructure provides client libraries (currently available for Unity3d and JavaScript development environments) that may be included as modules in a game project. These reduce the need for ad-hoc implementations, shortening development time.

To integrate a client library, the game developer should first play through their game, identifying each distinct type of event, and specifying the appropriate *event data* elements for each event type. Logging can then be implemented via calls to a logger function in the programming interface of the appropriate client library, which is designed to encourage best practices and will structure data according to the event schema described in the previous section.

Game Log Storage

As events are sent from instances of a game, they must be received and stored for later analysis. A key concern here is scalability. For example, over the last three years, we have collected approximately 1.5 billion individual events across 17 games. This staggering amount of research data needs to be stored inexpensively and accessed rapidly.

In order to decouple the receipt of data from any one storage vendor or technology, Open Game Data provides a simple "catcher" PHP script as an intermediate between game client and storage. While many vendors provide APIs to directly interact with storage, this intermediate ensures already-deployed games will not need to be modified in the event of migration to a new storage system. The "catcher" is designed to only insert into storage, never modifying or removing existing data. This simplicity helps to minimize maintenance costs, which typically outweigh storage costs.

We utilize a two-part data storage solution, consisting of "short-term" and "long-term" databases. Data is initially written into the short-term database, and a daily automation script copies all new events to long-term storage. Events remain in short-term storage for one week, leaving a time buffer for detection and fixing of any copy errors, before data is lost. The short-term database, then, maintains a relatively small and stable size, growing only as new games are added to the system. This makes it suitable as a data source for real-time tools, which need fast access to only the most recent events. The long-term database, on the other hand, grows daily. In order to maintain performance, this database should be segmented by time so that it does not need to search through years of data to find a given session or segment of the population.

In our implementation, we use a locally-hosted MySQL database for short-term storage. This system is relatively slow and expensive at scale, but fast for small datasets. For long-term archival storage, we turn to 3rd-party vendors, in this case Google's BigQuery. By using a 3rd party product, we further reduce maintenance costs, and the cost for raw storage at the time of writing was approximately $10/year for each year of data stored.

Data Transformations

The most complex stage of the pipeline is data transformation. This phase can be viewed as a sub-pipeline, with steps for generation of new events; extraction of session, player, and population features; and training and evaluation of models. These transforms are performed nondestructively, such that old data is never directly modified. In addition, the transforms use a "one-pass" approach that ensures each piece of data need only be retrieved once from storage, saving data transfer times that often plague ad-hoc feature engineering.

The first transform adds new, calculated events to the overall event dataset. These are produced by "detectors", and we refer to this as the detector phase. As discussed previously, the game should only log "objective" events that are accounted for by the game system itself. This transform handles the "other" events that may be based on shifting or subjective definitions.

The next transform is the "feature" phase, in which "extractors" convert the event stream to a set of summary descriptions of sessions, players, and/or populations. Each of these three levels of granularity may be calculated in parallel during this phase. A feature may be a simple summary statistic, such as a count of button clicks, or a compound data structure, such as a list of levels completed.

The third transformation takes feature and event data and produces trained machine learning models of player behavior. Implementation of this stage of the data transformation sub-pipeline is a matter of future work; we discuss it further in Sect. 5.

All data transforms are implemented in Python, the primary programming language of data science. To maintain flexibility and modularity, each event detector and feature extractor is written as an individual Python class. Each of these classes implements a filter for types of events to analyze, an update rule that executes on each accepted event, and a generation rule that produces either a new event (in the case of a detector) or a value (in the case of a feature). This makes it easy to add or remove detectors and features, allowing specific configurations for different analyses.

3.3 Data Sharing and Automation

The primary unit of data sharing in Open Game Data is the dataset file. The goal here is to provide easily-shareable files in a consistent, automated way. We also provide APIs for live, dynamic access to data. Together, these not only enable the use of different toolsets for game data analysis, but provide resources to support our goal of creating a community for open science.

File Production. File outputs from the Open Game Data pipeline are meant to be easily-shareable, both in terms of ease-of-access and ease-of-use. Files are given in a popular, vendor-agnostic tabular form, tab-separated value (TSV) files.

Monthly file generation is automated with the Actions feature of the GitHub platform. These export scripts generate files for each game with the prior month's raw events. For games that have some event detectors or feature extractors, files are created with combined raw and generated events, session-level features, player-level features, and/or population-level features. These are automatically uploaded to the Open Game Data website, discussed in a later section.

APIs. The Open Game Data RESTful APIs provide real-time access to data for web-based data tools. We provide API endpoints for each of the levels of granularity previously discussed. To maximize compatibility with the existing Python implementation of the data transformation stage, the API endpoints are developed in the Flask framework. This framework is itself written in Python and designed to run Python pto service API requests, so the same code can be used for the APIs as file generation.

Website. The Open Game Data website is central to the overall mission of the project, serving as the primary end-user interface for discovering and accessing open data, open analysis, open materials and open access research products. The front page of the site describes the project, encourages participation from various stakeholder groups, and most importantly, links to the individual game pages (Fig. 2 left). An API call populates the page on load, cataloging the games that are currently available and a count of the number of gameplay sessions available for download.

Clicking each of the games triggers the load from another API that retrieves everything available for that game (Fig. 2 right). Each month of data available for the game is displayed as a graph and clicking on a month loads the data pipeline. Currently, the pipeline provides links to downloadable files for the raw events, calculated events and calculated features at the session, player and population level. Also in the pipeline are links to the analysis code that was used during each transformation. For example, clicking on the detectors link in the pipeline will link to the GitHub repository and subfolder where each detector's python code is stored.

Each game page displays published research artifacts that utilize the game's data. The publications link to the openly available document itself, typically a PDF file, as well as to a GitHub repository for any analysis code that was used to write that paper.

Finally, each game page provides a link to the game's source code itself. In line with the Open materials component of open science, this provides the research community transparent documentation of exactly how each game is designed and how it produces data. It also provides a significant starting asset for modification in service of a novel but related research endeavor.

4 Use Cases

4.1 Visualizing Game Play and Student Change

Cool It was a game designed to teach design principles of cryogenic engineering, an advanced topic within mechanical engineering. Pfotenhauer et al. [27] describe how game telemetry data, captured by the technology that would eventually become Open Game Data, allowed research designers to develop visualizations of player actions and

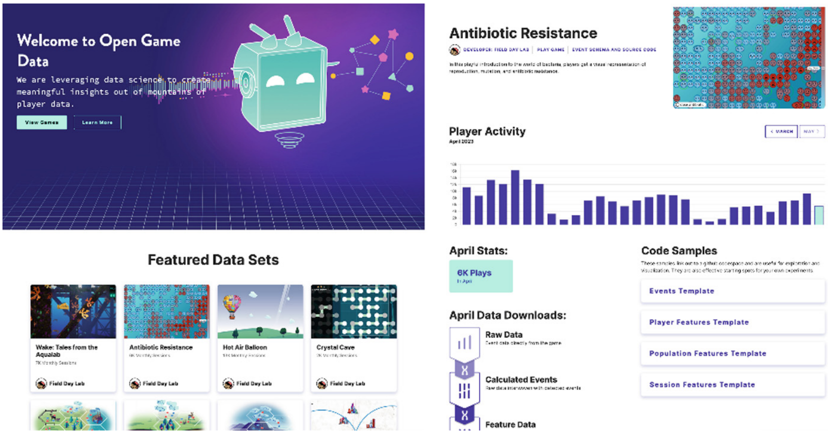

Fig. 2. The homepage (left) and a single game page (right) on the Open Game Data website

differentiate between expert and novice behavior. At the end of this work, the team was able to visualize a player model that described their understanding between each independent and dependent variable in the game's simulation and watch a player develop those understandings over time.

Lakeland is a game designed to teach about the systems of farming and phosphorus runoff that cause algae blooms in lakes. The game targets grade 7–9 students learning in earth systems and life sciences. In 2021, Scianna et al. [31] used open game data captured over the month of December 2019 in combination with a method known as epistemic network analysis (ENA) [33] to visualize and contrast player's initial and second attempts playing Lakeland (see Fig. 3). The researchers found that the more expert players (the blue network) responded to gameplay events differently than the novice players (the red network). ENA analysis of game logs provided a way to see that novice players received more negative feedback from the game system and that feedback was connected with players' development of short term responses. In contrast, experienced players utilized more long-term strategies and did more of their planning in relation to the availability of new resources.

4.2 Deriving and Visualization Player Typologies

Educational game designers will always be interested in ways to understand their audiences and how those audiences are interacting with the game. Swanson et al. [38] used data from Lakeland, in combination with the unsupervised K-means clustering algorithm to mathematically summarize 30,000 unique player sessions into a small number of categories that could be qualitatively described. The researchers describe their process of feature selection, dimension reduction and algorithm tuning to sort each play session into clusters, then describe these clusters in relationship to each other and key features of the game's design. They performed this operation three times to explore clusters of player actions (i.e. strategy), clusters of game feedback and clusters of player progression in the game. The results were visualized with radar plots and qualitatively

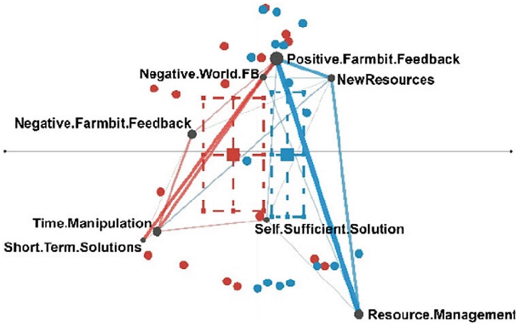

Fig. 3. An ENA visualization of new (red) vs. returning (blue) players of *Lakeland* (Color figure online)

interpreted (see Fig. 4 for an example). The analysis revealed the comparative size of different experiences (i.e. how many players used each strategy or progressed in a similar way) and suggested specific changes to the design to better align with the game's goals.

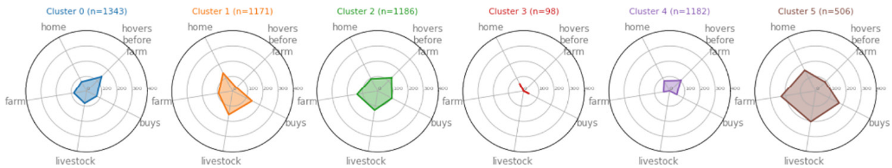

Fig. 4. Player action clusters for *Lakeland*

4.3 Predicting Outcomes on External Learning Measures

Many would argue that any learning game acts as an assessment of the players' skill and knowledge [9, 37], but assessment experts would caution that any use of a game to assess must first undergo a validity assessment. In Gagnon et al. [8], researchers embedded traditional multiple choice assessment items into the game to measure student understanding of key concepts within each of the games. The study utilized two games, Crystal Cave and Wave Combinator. In Crystal Cave, players are tasked with arranging different molecules to maximize the stability of the resulting structure. In Wave Combinator, players are presented visually with a wave and must adjust two signal generators to produce a resulting wave that closely matches the original. For both games, researchers found that a logistic regression model trained on features of approximately 5,000 gameplay sessions was able to aid the prediction of how students would perform on the embedded assessments.

In a more complex example, Kim et al. [14] studied Wake: Tales from the Aqualab, a game designed to teach the scientific practices of experimentation, modeling and argumentation in the context of aquatic life science topics such as food webs and photosynthesis. For this project, learners played the game for 5–7 class periods over two weeks, then were given a performance assessment, developed from existing measures, to score their post-game understanding of each science practice. Following the work in progress paper, the team will be identifying key performance metrics of the gameplay data and using them to build models that predict scores on the performance assessment.

4.4 Learning Engineering to Empirically Validate Design Theory

Educational video games are as much art as they are science. Learning theories certainly (should) inform the design process, but many decisions from art style, character representation, user interface and even music are made by the creative team. As Clarke et al. [4] conclude in their game efficacy meta-analysis, video game studies have moved from empirical studies assessing if games are useful for education and into a phase of attempting to understand how different components of their design support learning, and for whom. We believe that Learning Engineering may be the methodological approach to answering this call.

Learning Engineering is an emerging method for iteratively leveraging empirically based experiments to improve learning outcomes, specifically with digital learning interventions [3]. Common practices include A/B testing, educational data mining, and dataset generation. Applied to game research, learning engineering allows a design theory to be tested in terms of specific learning outcomes, producing evidence for or against the validity of a given theory.

Gagnon et al. [7] utilized learning engineering practices to study the design of Jo Wilder and the Capitol Case, a grade 3–5 history practices game. As part of the research, the team developed four distinct versions of the script, embedded questionnaire items into the game, then randomly deployed the different versions to over 11,000 students. The results surprised the designers, showing that none of the designs led to higher completion rates, and that some of the design choices were a poor fit for the age group.

Another study by Slater et al. [36] utilized the same Jo Wilder and the Capitol Case dataset but with different analysis approaches. This team began by developing a five group taxonomy of player attitudes using Latent Class Analysis, then studied these groups' reactions to each of the four scripts. The research highlighted the effect of different scripts for each audiences.

4.5 Developing Real Time Analytics to Support Instruction

Inspired by work such as Holstein et al. [10] with instructional tutors, Swanson et al. [39] utilized the real time components of Open Game Data to develop a web-based tool that would provide educators with insights into their students' learning experience while they were playing the game. This project utilized Lakeland, partially due to the complex systems involved in a real time strategy game and challenge for educators to assess student progress simply by looking at their screen at a moment in time. The project explored the key analysis that educators sought through a series of co-design activities, then tested a prototype of a real time tool in authentic learning contexts, concluding that much more effort should be put in developing these sorts of educator support tools for learning games.

5 Future Work and Extensions to Infrastructure

Several new capacities are planned or currently being developed to extend Open Game Data capabilities. All of these have been piloted at least once but haven't yet been developed as repeatable infrastructure.

Prototyping is underway to develop replay capabilities for games that were created using the Unity3d game engine. This new system allows the raw telemetry events generated by a player to be fed back into a version of the game, creating a time accurate replay of the entire experience, akin to a video recording of the experience, but much smaller and more flexible. This capability will then be combined with an annotation interface, allowing qualitative researchers to assign codes to segments of game play. These codes can be used to train detectors and automatically code new data without human intervention.

A system to support experiments and facilitate A/B testing is currently in planning. While the infrastructure to log data from multiple versions of a game is currently in place, a system to describe the changes between the versions of the game and automate assignment of a player to each version is not. Optimally this system would allow an experiment to be proposed completely online, defining rules for how the audience is assigned (randomly or based on some criteria) and parameters for the custom configuration in each game version, all without recompiling or deploying the game.

The addition of models, model training and model evaluation into the data pipeline is currently being explored. This entails integrating the capacity for common models, such as regressions, decision trees and neural networks to be added directly into the data pipeline. They would be continually trained on incoming data and their outputs available via API as well as in the automated file exports.

6 Conclusion

The paper describes the background and promise of educational games for teaching and assessment, especially when combined with educational data mining approaches. We then explore the notion of Open Science as a movement to improve scientific rigor while accelerating its progress and expanding participation. Open Game Data is a natural extension of these ideas and has already been utilized for several educational game research projects. This paper focuses on infrastructure and details a data pipeline that begins with telemetry events being generated by a learning game and ends with a website that makes datasets, analysis and research products available openly.

This paper outlines our thinking at an important point in time for the project, as it transitions from a tool used by a few researchers into a shared infrastructure intended to be informed by and support the research of a much wider audience. In the next year, many games are scheduled for integration, new analysis and visualization features are being developed and workshops are being facilitated to jumpstart new research projects. If successful, novel learning game scholarship will flourish by audiences that previously would not have been able to participate, due to cost or access.

Acknowledgements. Open Game Data would not exist without the regular and thoughtful contributions of Erik Harpstead, Jennifer Scianna and Stefan Slater over the last years, as well as the many staff and interns who have worked on the project. We also acknowledge the insights brought by Ryan Baker, Matthew Berland, YJ Kim, Elizabeth Owen, and Dennis Ramirez at key moments that have guided the project.

This material is based upon work supported by the 2022 Learning Agency Tools Competition, the Wisconsin Department of Public Instruction and the National Science Foundation under Grant No. (1907384, 2116046, 2142103, 2243668).

References

1. Advanced Distributed Learning: Experience API (xAPI) Standard (n.d). https://adlnet.gov/projects/xapi. Accessed 30 July 2023
2. Baker, R.: Modeling and understanding students' off-task behavior in intelligent tutoring systems. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI 2007, pp. 1059–1068 (2007). https://doi.org/10.1145/1240624.1240785
3. Baker, R.S., Boser, U.: High-leverage opportunities for learning engineering, p. 48. Penn Center for Learning Analytics (2022). https://learninganalytics.upenn.edu/Learning_Engineering_recommendations.pdf
4. Clark, D.B., Tanner-Smith, E.E., Killingsworth, S.S.: Digital games, design, and learning: a systematic review and meta-analysis. Rev. Educ. Res. **86**(1), 79–122 (2016). https://doi.org/10.3102/0034654315582065
5. D'Angelo, C., Rutstein, D., Harris, C., Haertel, G., Bernard, R., Borokhovski, E.: Simulations for STEM learning: systematic review and meta-analysis. SRI Int. **5**, 1–5 (2014)
6. DiCerbo, K.E., Kidwai, K.: Detecting player goals from game log files. In: Proceedings of the 6th International Conference on Educational Data Mining (EDM 2013), p. 2 (2013)
7. Gagnon, D.J., et al.: Exploring players' experience of humor and snark in a grade 3–6 history practices game. In: GLS 13.0 Conference Proceedings. Games, Learning Society, Irving, CA (2022)
8. Gagnon, D.J., Harpstead, E., Slater, S.: Comparison of off the shelf data mining methodologies in educational game analytics. In: Joint Proceedings of the Workshops of the 12th International Conference on Educational Data Mining Co-Located with the 12th International Conference on Educational Data Mining (EDM 2019), pp. 38–43 (2019)
9. Gee, J. P.: What Video Games Have to Teach us About Learning and Literacy (1. Paperback ed). Palgrave Macmillan (2004)
10. Holstein, K., McLaren, B.M., Aleven, V.: Co-designing a real-time classroom orchestration tool to support teacher–AI complementarity. J. Learn. Anal. **6**(2), 27–52 (2019). https://doi.org/10.18608/jla.2019.62.3
11. Ioannidis, J.P.A.: Why most published research findings are false. PLoS Med. **2**(8), e124 (2005). https://doi.org/10.1371/journal.pmed.0020124
12. Kai, S., Almeda, M.V., Baker, R.S., Heffernan, C., Heffernan, N.: Decision tree modeling of wheel-spinning and productive persistence in skill builders. JEDM | J. Educ. Data Min. **10**(1), 36–71 (2018)
13. Kamarainen, A.M., et al.: EcoMOBILE: integrating augmented reality and probeware with environmental education field trips. Comput. Educ. **68**, 1–12 (2013). https://doi.org/10.1016/j.compedu.2013.02.018
14. Kim, Y.J., Metcalf, S.J., Scianna, J., Perez, G., Gagnon, D.J.: AquaLab: establishing validity of an adventure game for middle school science practices. In: Iyer, S. (ed.) Proceedings of the 30th International Conference on Computers in Education (2022)

15. Kim, Y.J., Shute, V.J.: The interplay of game elements with psychometric qualities, learning, and enjoyment in game-based assessment. Comput. Educ. **87**, 340–356 (2015). https://doi.org/10.1016/j.compedu.2015.07.009
16. Koedinger, K.R., Baker, R.S.J.D., Cunningham, K., Skogsholm, A., Leber, B., Stamper, J.: A data repository for the EDM community: the PSLC DataShop. In: Romero, C., Ventura, S., Pechenizkiy, M., Baker, R.S. (eds.) Handbook of Educational Data Mining. CRC Press, Boca Raton (2010)
17. Kraker, P., Leony, D., Reinhardt, W., Gü, N.A., Beham, N.: The case for an open science in technology enhanced learning. Int. J. Technol. Enhanc. Learn. **3**(6), 643 (2011). https://doi.org/10.1504/IJTEL.2011.045454
18. Lomas, D.: Optimizing challenge in an educational game using large-scale design experiments, pp. 1–10 (2013)
19. Makel, M.C., Plucker, J.A.: Facts are more important than novelty: replication in the education sciences. Educ. Res. **43**(6), 304–316 (2014). https://doi.org/10.3102/0013189X14545513
20. Metcalf, S.J., Reilly, J.M., Kamarainen, A.M., King, J., Grotzer, T.A., Dede, C.: Supports for deeper learning of inquiry-based ecosystem science in virtual environments—comparing virtual and physical concept mapping. Comput. Hum. Behav. **87**, 459–469 (2018). https://doi.org/10.1016/j.chb.2018.03.018
21. Mislevy, R.J., Almond, R.G., Lukas, J.F.: A brief introduction to evidence-centered design. National Center for Research on Evaluation, Standards, and Student Testing (CRESST) Center for the Study of Evaluation (CSE), Graduate School of Education & Information Studies. (2004). https://files.eric.ed.gov/fulltext/ED483399.pdf
22. National Research Council: Successful K-12 STEM Education: Identifying Effective Approaches in Science, Technology, Engineering, and Mathematics. The National Academies Press (2011). https://doi.org/10.17226/13158
23. Noorden, R.V.: Researchers can get visibility and connections by putting their data online—if they go about it in the right way. Nature **500**, 243–245 (2013). https://doi.org/10.1038/nj7461-243a
24. Owen, V.E., Baker, R.S.: Fueling prediction of player decisions: foundations of feature engineering for optimized behavior modeling in serious games. Technol. Knowl. Learn. **25**(2), 225–250 (2020). https://doi.org/10.1007/s10758-018-9393-9
25. Owen, V.E., Ramirez, D., Salmon, A., Halverson, R.: Capturing learner trajectories in educational games through ADAGE (assessment data aggregator for game environments): a click-stream data framework for assessment of learning in play. In: American Educational Research Association Annual Meeting, Philadelphia (2014)
26. Owen, V.E., Roy, M.-H., Thai, K.P., Burnett, V., Jacobs, D., Baker, R.S.: Detecting wheel-spinning and productive persistence in educational games, p. 6 (2019)
27. Pfotenhauer, J.M., Gagnon, D.J., Litzkow, M.J., Blakesley, C.C.: Designing and using an on-line game to teach engineering. In: FIE2009, pp. 1–5 (2009)
28. Mayer, R.: Computer Games for Learning: An Evidence-Based Approach. MIT Press, Cambridge (2014)
29. Rosenthal, R.: The file drawer problem and tolerance for null results. Psychol. Bull. **86**(3), 638–641 (1979). https://doi.org/10.1037/0033-2909.86.3.638
30. Rowe, E., et al.: Assessing implicit science learning in digital games. Comput. Hum. Behav. **76**, 617–630 (2017). https://doi.org/10.1016/j.chb.2017.03.043
31. Scianna, J., Gagnon, D., Knowles, B.: Counting the game: visualizing changes in play by incorporating game events. In: Ruis, A.R., Lee, S.B. (eds.) ICQE 2021. CCIS, vol. 1312, pp. 218–231. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-67788-6_15
32. Shaffer, D.W.: Epistemic frames for epistemic games. Comput. Educ. **46**(3), 223–234 (2006). https://doi.org/10.1016/j.compedu.2005.11.003

33. Shaffer, D.W., et al.: Epistemic network analysis: a prototype for 21st-century assessment of learning. Int. J. Learn. Media **1**(2), 33–53 (2009). https://doi.org/10.1162/ijlm.2009.0013
34. Shute, V.J., Shute, V.J., Kim, Y.J.: Formative and stealth assessment. In: Spector, J., Merrill, M., Elen, J., Bishop, M. (eds.) Handbook of Research on Educational Communications and Technology, vol. 1, pp. 311–321. Springer, New York (2013). https://doi.org/10.1007/978-1-4614-3185-5_25
35. Siemens, G., Baker, R.S.: Learning analytics and educational data mining: towards communication and collaboration. In: Proceedings of the 2nd International Conference on Learning Analytics and Knowledge - LAK 2012, p. 252 (2012). https://doi.org/10.1145/2330601.2330661
36. Slater, S., Baker, R.S., Gagnon, D.J.: Changing students' perceptions of a history exploration game using different scripts. In: Proceedings of the 30th International Conference on Computers in Education. International Conference on Computers in Education, Kuala Lumpur, Malaysia (2022)
37. Squire, K.D.: From content to context: videogames as designed experience. Educ. Res. **35**(8), 19–29 (2006)
38. Swanson, L., et al.: Leveraging cluster analysis to understand educational game player styles and support design. In: GLS 13.0 Conference Proceedings. Games, Learning, Society, Irving, CA (2022)
39. Swanson, L., Gagnon, D.J., Scianna, J.: A pilot study on teacher-facing real-time classroom game dashboards. In: International Conference on Meaningful Play, East Lansing, MI (2022)
40. Van Der Zee, T., Reich, J.: Open education science. AERA Open **4**(3) (2018). https://doi.org/10.1177/2332858418787466
41. Van 'T Veer, A.E., Giner-Sorolla, R.: Pre-registration in social psychology—a discussion and suggested template. J. Exp. Soc. Psychol. **67**, 2–12. (2016). https://doi.org/10.1016/j.jesp.2016.03.004
42. Ventura, M., Shute, V.: The validity of a game-based assessment of persistence. Comput. Hum. Behav. **29**(6), 2568–2572 (2013). https://doi.org/10.1016/j.chb.2013.06.033