Received: 22 March 2023 | Revised: 14 July 2023 | Accepted: 5 August 2023

DOI: 10.1002/spe.3254

RESEARCH ARTICLE

WILEY

A framework to support interoperable Game-based Assessments as a Service (GBAaaS): Design, development, and use cases

Manuel J. Gomez | **José A. Ruipérez-Valiente** | **Félix J. García Clemente**

Faculty of Computer Science, University of Murcia, Murcia, Spain

Correspondence
Manuel J. Gomez, Faculty of Computer Science, University of Murcia, Calle Campus Universitario 32, 30100, Murcia, Spain.
Email: manueljesus.gomezm@um.es

Funding information
Fundación Séneca, Grant/Award Number: 21795/FPI/22; Call for Projects to Generate New Scientific Leadership, Grant/Award Number: 21948/JLI/22

Abstract

During the last few years, there has been increasing attention paid to serious games (SGs), which are games used for non-entertainment purposes. SGs offer the potential for more valid and reliable assessments compared to traditional methods such as paper-and-pencil tests. However, the incorporation of assessment features into SGs is still in its early stages, requiring specific design efforts for each game and adding significant time to the design of Game-based Assessments (GBAs). In this research, we present a completely novel framework that aims to perform interoperable GBAs by: (a) integrating a common GBA ontology model to process RDF data; (b) developing in-game metrics to infer useful information and assess learners; (c) integrating a service API to provide an easy way to interact with the framework. We then validate our approach through performance evaluation and two use cases, demonstrating its effectiveness in real-world scenarios with large-scale datasets. Our results show that the developed framework achieves excellent performance, replicating metrics from previous literature. We anticipate that our work will help alleviate current limitations in the field and facilitate the deployment of GBAs as a Service.

KEYWORDS

big data, data mining, educational technology, Game-based Assessment, interoperability, ontologies

1 | INTRODUCTION

Video games have assumed an essential place in our lives, evolving into complex and diverse platforms that are enjoyed by people of all ages and backgrounds.[1,2] This has generated increasing interest in using games in various settings during the last decade.[3] The application of games with a non-entertainment primary purpose, known as serious games (SGs), can provide multiple benefits in environments where games were not traditionally used.[4] Education is one such domain,

Abbreviations: AI, artificial intelligence; API, application programming interface; GBAs, game-based assessments; GBAaaS, game-based assessment as a service; HDFS, hadoop distributed file system; ML, machine learning; OBDA, ontology-based data access; OWL, web ontology language; RDF, resource description framework; RDD, resilient distributed dataset; REST, representational state transfer; SANSA, scalable semantic analytics stack; SGs, serious games.

where traditional content still constitute the majority of learning materials, and there is little consensus on how to effectively integrate technology in the classrooms.[5] However, SGs are also used in many other domains,[6] including training, well-being, advertisement, and interpersonal communication, among others. SGs are also being explored as assessment tools, in particular for their potential to provide more valid assessments compared to traditional assessment approaches, providing more meaningful and authentic contexts for assessments through interactive and immersive environments.[7] In the context of Game-based Assessment (GBA), a key challenge is making valid inferences about what the student knows, believes, and can do without disrupting the game flow.[8]

Furthermore, GBA machinery, which includes metrics, dashboards, and other analytics,[9,10] is usually designed for each game, leading to increased costs, time, and effort. This development also requires the maintenance of a complete infrastructure, requiring dedicated engineers to address these tasks. Semantic web technologies, and particularly ontologies, can address these heterogeneity problems. Ontologies capture domain-specific knowledge and offer an explicit common conceptualization.[11] Nowadays, ontologies are used in various application areas involving artificial intelligence, natural language processing, data integration, and knowledge management.[12] Using ontologies to define a standard knowledge model in the context of Game-based Assessments (GBAs) could alleviate the time-consuming and costly step of creating GBAs.[13]

The growing use of GBAs has led to the creation of large data repositories, presenting new assessment opportunities.[14,15] A game (educational or not) can generate vast amounts of interaction data, even in a short game-play session. Data mining and visualization techniques applied to player interaction logs can provide valuable insights into how players engage with the game, leading to improvements in assessment methods and real-time feedback on activity progress.[16] However, data processing usually shows performance deficiencies when the dataset exceeds the memory size of a single machine, and distributed computing frameworks can be employed to address this limitation.[17,18] Developing an efficient system capable of processing large amounts of data and performing interoperable GBAs could significantly simplify the design process. In addition, utilizing this system as an external service could reduce costs and efforts, enabling the use of GBAs as a Service (GBAaaS).

In this research, we present a novel approach that combines the use of ontologies with big data technologies to create interoperable GBAs. To address the challenge of game interoperability and specific assessment machinery, we have developed a new framework that automatically computes in-game metrics using the provided data and the standard ontology model, where we re-use the existing Scalable Semantic Analytics Stack (SANSA).[19] Additionally, to solve the infrastructure development and maintenance issues, we integrate a service API into our framework, allowing it to function an external service and enabling the GBAaaS paradigm. Specifically, in this research, we present the following contributions:

- **Framework development**. Using SANSA-Stack as a baseline, we develop a framework that integrates our created GBA ontology using big data technologies. This facilitates game interoperability and enables the GBAaaS paradigm. The main novel features of our framework are:
  - **Ontology integration**. Our framework integrates the previously designed ontology into the SANSA framework to process large resource description framework (RDF) data.
  - **In-game metric development**. We develop and integrate a set of basic GBA metrics from the literature into the framework, enabling interoperable learners' assessment using RDF data. These metrics are also used to test the framework's performance when querying large datasets.
  - **Service API integration**. We develop and integrate a REST API into our ecosystem, providing an easy way to interact with the framework and allowing the use of our framework as an input/output service (GBAaaS).
- **Framework evaluation**. We have conducted a performance evaluation and a case study validation:
  - **Performance evaluation**. This involves two tasks aiming to validate our framework's input services: (1) data scalability, testing how our framework scales to larger datasets and what the improvement is concerning the number of workers in the cluster mode; and (2) flexibility, testing how the framework processes different metrics.
  - **Case study validation**. We present a case study with two use cases to demonstrate how our framework can be used as GBAaaS in various real applications of GBA in educational environments.

The rest of the article is structured as follows: Section 2 reviews background literature on SGs, GBA, and the use of ontologies in big data environments. Section 3 presents the framework proposal and the case studies performed to test its performance and capabilities. Then, we finalize the article with a discussion in Section 5, and conclusions and future work in Section 6.

2 | RELATED WORK

In this section, we present a review of the literature in the areas most closely related to our work: in Section 2.1, we review literature related to SGs and GBA, and Section 2.2 reviews the literature related to ontologies and their use in big data environments.

2.1 | Serious games and game-based assessment

The idea of playing a game dates to the ancient past and is considered an integral part of all societies.[6] In addition to the previously mentioned benefits, it is argued that SGs can also positively impact the players' development of several different skills.[20] SGs are currently being used in several contexts. For example, using games for formal education has become widely accepted as playing games has become an essential part of young people's lives worldwide.[21] Additionally, there is also an increasing interest in how games can be effectively applied in learning and training contexts, as well as in other areas such as healthcare,[22] rehabilitation,[23] and military training.[24] For instance, Albaladejo et al.[25] presented a multimodal system that could be used to improve cyberdefense capabilities by utilizing gamified platforms for training and testing individuals and organizations in cybersecurity practices and techniques.

However, SGs must be able to show that the necessary learning has occurred. An advantage of SGs as assessment tools is that they can be programmed to capture, store, and share massive amounts of user data over time.[7] This data can be used to perform reliable assessments and manifest this learning, enabling GBA. A common approach to perform this assessment is to use a set of metrics (or indicators) that transform raw data into meaningful information. For example, the authors in Reference 26 proposed a multidimensional measurement of engagement in a learning game ("The Radix Endeavor") across four dimensions: general activity, social, exploration, and quests. Similarly, researchers in Reference 27 explored the creation of engagement profiles based on log data, considering the different ways players engage with the game and highlighting patterns associated with active play. Many other studies have conducted research aiming to assess users' interaction with games, measuring factors such as persistence,[28-30] difficulty,[31,32] and level completion,[10,33] among other measures. Furthermore, we found examples of metrics developed in non-educational contexts. Authors in Reference 34 developed a GBA using multi-level functional tasks to assess instrumental activities of daily living in a sample of inpatients with chronic schizophrenia, measuring completion times and errors in each task. Furthermore, Jackson et al.[35] assessed a sample of 67 Reserve Officers' Training Corps (military sample) using the commercial game "Crysis 2,"[36] which simulates key features involved in combat situations. In this research, authors measured tasks such as the number of eliminations, shots accuracy, or damage per bullet, among others.

As we have seen, many SGs track their learners' interactions, but they usually use custom formats.[37] However, there are enough case studies that identify common interactions tracked by SGs to start defining a standardized model. Previous studies have proposed approaches to standardize analytics in games. For example, Serrano et al.[37] presented xAPI, an implementation of a standard model that sets a basis for performing analysis in SGs methodologically. Moreover, Said et al.[38] proposed an ontology to model player experience and its association with in-game personalization, also defining a set of reasoning rules to suggest tailored games for each player's assessment path and player experience. Authors in Reference 39 proposed an ontology that allows the description and representation of SGs that use resources from the Web of Data, introducing concepts such as "game structure," "game simulation," or "game rule." However, we did not find any study attempting to standardize the GBA area to build interoperable assessments.

In this research, we go beyond the existing literature by attempting to standardize the GBA area through the definition of a common knowledge model. With this purpose in mind, we combine ontologies, big data technologies, metrics, and API services to create a novel framework capable of analyzing and inferring new knowledge using data from different games. This information is analyzed and transformed using interoperable metrics, which are available for consultation in various output formats. Finally, the API service has been defined and integrated to facilitate interaction with our framework by different sources.

2.2 | Ontologies and big data architectures

Over the last 15 years, ontology-based applications have been spreading and maturing. One can now find ontology-based applications in diverse areas, including customer support and car engineering.[40] For instance, researchers in Reference 41 designed and developed an integrated ontology of software engineering approaches to support sustainable software development knowledge, awareness, and implementation. Furthermore, in Reference 42, authors conducted an analysis that explained a detailed approach to building an ontology that can be used across different e-learning platforms. Additionally, as we have seen in the previous section, several studies[37-39,43] presented ontologies in different SGs applications (e.g., collaborative learning, Web Data technologies). However, little research has been conducted in the direction on ontology development for GBA. To the best of our knowledge, no ontology has been proposed specifically for the GBA area, except for the one we presented in our previous research.[44] This ontology is part of the framework and acts as the intermediate semantic layer in this research.

Several tools for managing ontology and ontology-based data are available, such as Protégé. Previous research includes studies and frameworks that dealt with ontology-based data. Botoeva et al.[45] generalized ontology-based data access (OBDA) to allow querying arbitrary data sources using SparQL and compared implementing an OBDA system over MongoDB with a triple store. Moreover, authors in Reference 46 presented *Minerva*, a storage and inference system for large-scale OWL ontologies on top of relational databases. *Minerva* comprises four different modules: an import module for reading ontology data, an inference module, a storage module for storing original and inferred assertions, and a query module that uses SparQL.

However, storing ontology data in the computer's main memory is a problem for applications that manipulate a large amount of ontology-based data.[47] In recent years, several studies have proposed new approaches that use big data technologies to manage ontology-based data. For example, Abbes and Gargouri[48] proposed an approach based on MongoDB and modular ontologies. They made it possible by wrapping data sources to MongoDB databases, generating local ontologies, and finally composing the local ontologies to get a global one. Mountasser et al.[49] presented a semantic-based big data integration framework that relies on large-scale ontology matching and probabilistic-logical-based assessment strategies; they proved its efficiency in terms of accuracy, performance, and scalability. Moreover, Reyes-Álvarez et al.[50] presented a novel approach that enables the distributed storage of ontology-based data by exploiting the inherent distribution of NoSQL database nodes. Finally, authors in Reference 19 presented SANSA, a big data engine for scalable processing of large-scale RDF data using Spark and Flink. In our research, we use SANSA as a base for our framework. In particular, we take advantage of various SANSA functionalities, such as the "Read/Write RDF library," and the "inference library." Furthermore, we enhance SANSA's functionalities by adding custom rules and queries to infer new knowledge from existing data, support for various output formats, and a service API.

3 | FRAMEWORK PROPOSAL

3.1 | Framework requirements

Classic assessment has evolved over the past several years from traditional pen and paper-based tests to the use of technology, such as games, and continues to evolve.[51] However, implementing assessment features into games is only in its early stages because it adds a time-consuming step to the design process.[13] This is due to heterogeneity issues since assessment mechanisms are usually explicitly designed for each game. Moreover, with the challenges brought on by GBAs, including data analytics, the large amount of data now available for teachers is far too complex for conventional database software to store, manage, and process.[52] Finally, integrating GBAs in different environments generates a diverse range of data from various sources,[37] emphasizing the need for a unified and secure way to access GBA data. Therefore, we identified the following requirements:

Requirement 1—Semantic layer between the event data and a common knowledge model: As previous literature reported,[11,53,54] there are heterogeneity issues with the collected data. In fact, most previous studies did not report any specific format for the collected data. Therefore, we need to define a standard knowledge model that unifies the GBA area and can represent the necessary information for user assessment.

Requirement 2—Processing of large scale data: User interaction with games generate massive amounts of data to be analyzed. Although the authors in Reference 53 reported relatively low sample sizes in the studies, Gomez et al.[55] stated that GBA research would benefit from using larger datasets since nearly half of the studies in their review described limitations in data sampling. Thus, it is important to use larger data samples, and we need to be prepared to process large quantities of data to extract useful and reliable information from them.

Requirement 3—Game interoperability for GBA metrics and visualizations: GBA studies previously conducted normally used ad-hoc solutions that enable data gathering to perform some assessment based on users' data. In addition, some studies also enabled visualization dashboards for instructors and users.[10] These solutions had to be developed specifically for each game, severely limiting reusability. Therefore, to scale up the number of GBA implementations, we need to provide new interoperable approaches that can reduce the effort required to build new GBAs, using scalable and interoperable modules that can easily be added and reconfigured.

Requirement 4—Easy communication with external sources: Due to the integration of GBAs in different environments, numerous data sources can hold valuable information to assess user interactions with games. Since many different sources could use this computed information, we need to enable the system to be used as an input/output service to deposit data from different sources or query the interoperable assessments generated. This need could be met by integrating an API that can be used across applications, allowing the use of GBAaaS.

Requirement 5—Privacy, authentication, and authorization configurations: Users' privacy in web services is essential to address. GBA data contains valuable but also sensitive information, and sharing or analyzing these data introduces privacy risks for the data subjects, primarily students.[56] Among the many methods proposed in the literature, role-based access control (RBAC) has been widely accepted as the most promising model because of its simplicity, flexibility in capturing dynamic requirements, and support for the principle of least privilege and efficient privilege management.[57] Therefore, we need to provide a system with different roles and permissions to ensure that only the appropriate users can access specific analyses and data.

3.2 | Architecture

In this section, we describe each module of the framework's architecture, which can be seen in Figure 1. We divide our framework into five different modules: (1) Preprocessing module, (2) analytics, inference, and querying module, (3) metric output module, (4) authentication and authorization module, and (5) API module. As we can see, the first module aims to transform the raw data (CSV, TSV, JSON …) into RDF/XML files by using an ontology model. In the next step, we use the SANSA framework to process the ontology data, infer new information, and perform queries over the inferred data. In the metric output module, we aim to provide different output formats for the query results. Then, the authentication and authorization module manage the different roles and authorizations in the system, ensuring that clients making requests through our API module are correctly logged in and have the necessary permissions. Finally, the API module aims to facilitate easy integration with external sources by generating a web service that can be used across different applications.

3.2.1 | Preprocessing

The first step is to transform the raw data, which can be received in multiple formats (e.g., CSV, TSV, JSON), into a format understandable by the common knowledge model. In this research, we use the "GBA ontology," a previously developed ontology used as an intermediate knowledge model between the raw data and the metrics outputted by the framework. Since there was no existing ontology that met our requirements and could be expanded, the "GBA ontology" was built from scratch using methontology.[58] Methontology is a structured method designed to build ontologies from scratch or by re-engineering existing ones.

Our ontology aims to satisfy **R1**, creating an intermediate layer to transform log data (produced by users' interaction with games) into ontology data. This new format is used to analyze and infer new information for assessing users. The ontology includes core concepts required for user assessment given any data set, such as "game event," "attempt," "unit of play" (which is equivalent to a level), "user," or "user group." An overview of the ontology (including classes and relationships between them) can be seen in Figure 2. The reader can view the full ontology using the *Web-Protégé* web page link or the source file, both available in Reference 59. In addition, the full development process and a more detailed view of the ontology can be found in Reference 44.

Using this model, the raw data is transformed into ontology classes, annotations, and relationships that contain the same information. This information is stored using RDF/XML format. The Extensible Markup Language (XML) has become widely adopted, along with transformation languages like XSLT and XQuery, to translate data from one XML format into another.[60] However, RDF has become another popular data representation and exchange standard. It is a

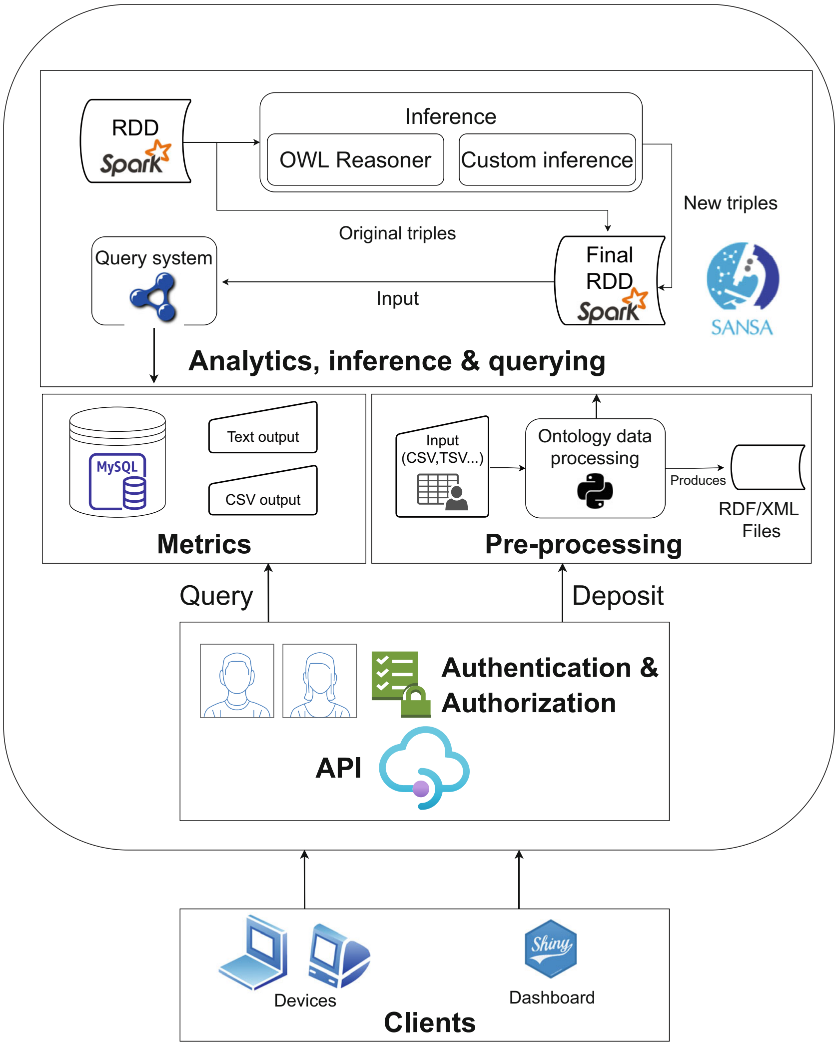


FIGURE 1 Framework's architecture.

general-purpose language for representing data and metadata on the web and is supported by its own query language, SparQL, which enables the extraction and transformation of RDF data. RDF has an XML syntax called RDF/XML, where the formal grammar for the syntax is annotated with actions generating triples of the RDF graph.[61]

3.2.2 | Analytics, inference, and querying

Once the data is stored in RDF/XML format, the next step is to process this data and infer new information. To fulfill **R2** and **R3**, we have decided to use the SANSA framework as the basis for performing our analyses. SANSA is an open-source structured data processing engine that enables distributed computation over large-scale RDF datasets. It provides data distribution, scalability, and fault tolerance for manipulating large RDF datasets. SANSA facilitates scalable analytics on the data by utilizing cluster-based big data processing engines, with Spark being the specific engine we employ.[19] An overview of SANSA architecture is shown in Figure 3. Specifically, SANSA includes:

- Specialized serialization mechanisms and partitioning schemas for RDF, using vertical partitioning strategies.
- A scalable query engine for large RDF datasets and different distributed representation formats for RDF.
- An adaptive reasoning engine that derives an efficient execution and evaluation plan from a given set of inference rules.
- Several distributed structured machine learning (ML) algorithms can be applied to large-scale RDF data.
- A framework with a unified API that aims to combine distributed in-memory computation technology with semantic technologies.

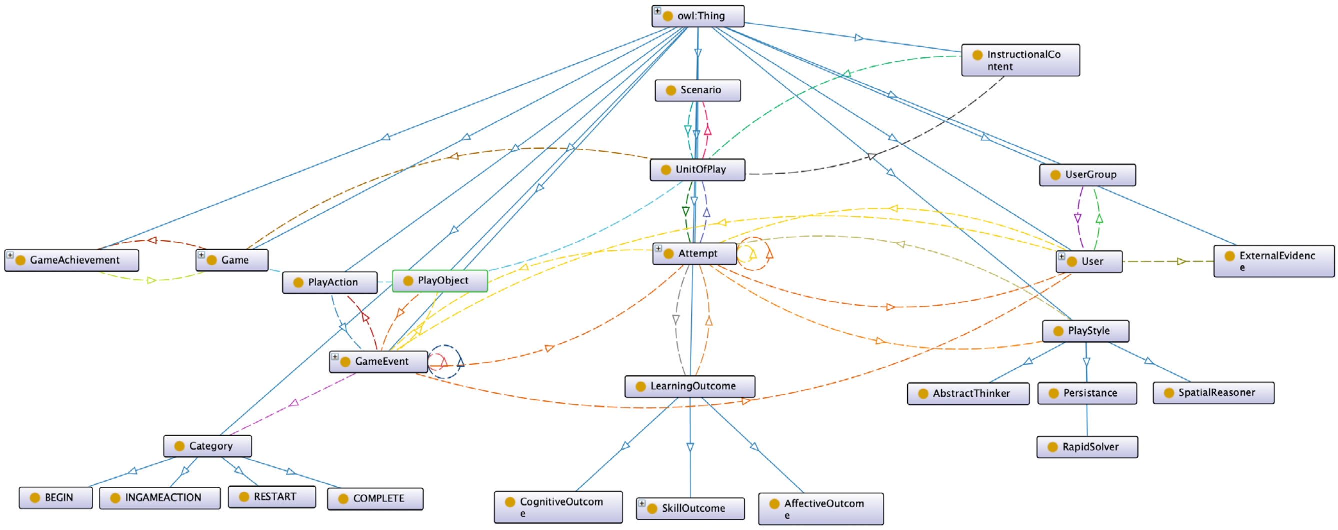


FIGURE 2 An overview of GBA ontology classes and relationships visualized via Protégé Ontograf.

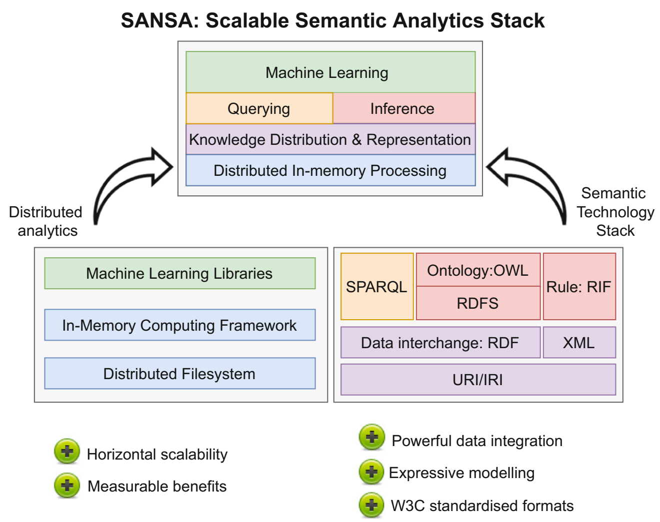


FIGURE 3 SANSA framework architecture.

In our framework, we leverage several SANSA's functionalities. We use the "Read/Write RDF library" to read the RDF data obtained from the raw data. This library allows us to read the RDF data from HDFS or the local drive file in the form of triples, and represent it in Spark's native distributed data structure, RDDs (resilient distributed datasets). To infer new information from the existing triples, we use an extended version of the "inference library" provided by SANSA. This library supports Jena and Web Ontology Language (OWL) API interfaces for processing RDF and OWL data, respectively. As both RDF and OWL contain schema information and links between different resources, applying rules enables us to infer new knowledge and expand upon the existing one. SANSA provides an adaptive rule engine that can utilize a given set of rules and derive an efficient execution plan. Finally, we use the "querying library," which provides methods for performing queries directly within programs instead of writing the code corresponding to those queries.[62] For querying the data, we use SparQL, which allows users to query RDF graphs by specifying "templates" against which to compare graph components. Data that matches or "satisfies" a template is returned from the query.[63]

Inference

As mentioned earlier, SANSA's inference library provides several sets of rules, which can be used to infer new knowledge from existing facts. In our implementation, we use the OWL-Horst reasoner, which contains a set of useful rules from the OWL language.[64] An example of a rule can be seen in Table 1. In this rule, if a *property*$_1$ is defined as "inverse" of

TABLE 1 OWL Horst rule example.

Condition	Consequence
property$_1$ owl:inverseOf *property*$_2$	*instance*$_2$ *prop*$_2$ *instance*$_1$
instance$_1$ *prop*$_1$ *instance*$_2$	

another *property*$_2$, and then we have an *instance*$_1$ connected through *property*$_1$ with an *instance*$_2$, the reasoner will create the relationship between *instance*$_2$ and *instance*$_1$ through *property*$_2$.

We can see an example of this property applied to our ontology: we have two relationships, "has" and "from," defined as inverse. If we have a connection between an instance of the "user" class and an instance of the "game session" class through property "has" (i.e., a user has a game session), the reasoner will create the relationship between "game session" and "user" through the property "from" (i.e., game session from a user). In addition to the predefined set of rules we use for the inference, we extend this inference by using the "Triple" class implementation provided by SANSA. Our implementation allows for custom inferences, iterating over the triples, and creating or deleting new ones if necessary. This way, we can create custom rules that extend the existing ones in SANSA, adapting the inference process to our specific needs.

Querying

Querying an RDF graph is the primary method for searching, exploring, and extracting information from the underlying RDF data.[19] For querying our data, we use SparQL, which is the standard language for querying RDF data. SparQL queries consist of three different parts: the pattern matching part, which includes various features of graph pattern matching, such as optional parts, the union of patterns, or nesting; the solution modifiers, which allow modifying the query results by applying operators like projection, distinct, or limit; and the output, which can be in different forms, such as yes/no or a selection of values.[65] To perform SparQL queries, SANSA implements *Sparklify*, a scalable software component for efficient evaluation of SparQL queries over distributed RDF datasets. It uses a SparQL-to-SQL rewriter to translate SparQL queries into Spark executable code.[66] Thus, our metrics are developed in the form of SparQL queries, which are executed over the RDF graph created. The use of our ontology, along with the SANSA framework and SparQL queries, enables metric interoperability, satisfying **R3**. To select the metrics to implement, we focused on replicating metrics from the current state of the art in GBA. For this purpose, we used the selection of papers from a previous systematic literature review in the GBA area,[55] carefully reviewing each paper and selecting the described metrics. We excluded calculations over data that involved ML, deep learning (DL), and similar models/algorithms, as our focus was on metrics described in the literature.

3.2.3 | Metrics output

When querying in SANSA, SparQL takes the description as a query and returns that information as a set of bindings or an RDF graph. In our framework, we extended this functionality, providing three different output formats:

- **Text formatting**: Query results are transformed into a text readable format, which can be stored as a text file or shown via the console output.
- **CSV formatting**: Query results are transformed into CSV format.
- **Database store**: Query results are saved into a MySQL database, which enables metric persistence for later retrieval from different applications.

3.2.4 | Authentication and authorization

We have implemented both authentication and authorization processes in our framework, addressing the issues presented in **R5**. Authentication is the process of identifying an entity (users) and is a prerequisite for authorization. Authorization, or access control, is the process of determining whether an entity (a device or a user) can access specific resources.[67] Specifically, we have implemented RBAC using the functionalities of the play framework.[68] This allows us to restrict access to specific resources based on the user's role. In our framework, each user can log in to the system using a username and password, and a user can have one of the following three roles:

- **Admin role**. Users with this role can access the entire system. They have the ability to add or remove new users, insert new GBA data, and query metrics from any game and group.
- **Instructor role**. Instructors can insert new GBA data and query metrics from games and groups in which they participate.
- **Learner role**. Learners are only allowed to query their own metric results.

This way, we restrict access to different groups and games data to ensure that only appropriate users can have access. To make that possible, the system keeps a record of the games and groups related to each user (which will be the ones to which the user has access).

3.2.5 | Service API

Our service API has been developed using play framework. This scala-based solution offers an HTTP-focused framework with numerous helpers to accelerate development, resulting in shorter iterations and faster deployments. The API supports two types of calls: retrieval calls, represented by HTTP GET methods (typically used to retrieve data from a server at the specified resource), and insertion calls, represented by HTTP POST methods (used to send data to the API server to create or update a resource).

Specifically, GET methods allow users to access metrics data. Generally, these methods have the following route pattern: `/api/metricName/game/group/user`. By specifying the name of the metric, game, group, or user, the corresponding data can be accessed. Before each call, the authorization and authentication module checks that the user authenticated has the appropriate permissions to access the requested resource. On the other hand, POST methods enable users to insert data related to new users (if the user has an admin role), games, and GBA data. All information should be sent in JSON format using the body of the HTTP POST method. Moreover, when inserting data, we have implemented two different possibilities (calls), each one thought for a different purpose:

- `/api/event/addAll`: This call has been designed to process whole datasets containing a large number of events. The system will process the dataset provided as soon as possible.
- `/api/event/add`: This call has been designed for streaming-oriented systems (e.g., students are playing the game in the classroom, and the system sends log data in real-time). Thus, the events will be sent individually; the system will save each event and process the whole dataset periodically.

To ensure that the system only processes new data, each time a dataset is received, the system checks the number of events associated with each user. Only users with new events will be considered for further processing.

APIs expose data and services that consumers want to use. An API should be designed with an interface the consumer can understand, and API documentation is key to the app developers comprehending the API. For documenting our API, we have used Swagger,[69] one of the most popular API documentation frameworks. It provides a standard, language-agnostic way of defining a REST API interface, allowing the client to understand the capabilities of the REST service without any prior access to the service implementation code or network inspection.[70] The complete API specification in Swagger can be found in Reference 59. This fulfills the requirements presented in **R4**.

4 | PERFORMANCE EVALUATION AND CASE STUDY VALIDATION

4.1 | GBA selected metrics

As previously mentioned, we thoroughly reviewed a previous systematic literature review on GBA[55] to identify and replicate metrics implemented in previous studies. After reviewing all the metrics, we selected the following six groups of metrics:

- **Levels of activity**: This metric is computed for each game, group, and user. It includes straightforward metrics to compute based on a feature engineering process, such as the active time, inactive time, number of events, and the number of distinct types of events.
- **Persistence indicators**: This metric is computed for each game, group, and user. It includes the total amount of time spent in units (levels), the number of units completed, and the maximum time spent in a single unit.
- **Action indicators**: This metric is computed for each game, group, and user. It includes the total amount of time spent in the game and the frequency of events (number of events/total time).
- **Event types**: This metric is computed for each user and game, and it includes the number of events of each user grouped by event type (e.g., "Complete," "Retry," or "Interaction"). In addition, this metric group also includes the interaction level, defined as interaction events divided by the sum of the rest of the events.
- **Funnel by user**: This metric is computed for each game, group, and user. It includes the percentage of units started, the percentage of units interacted with, and the percentage of units completed by the user. This funnel provides a quick overview of each user and the game's current status and progress.
- **User performance**: This metric is computed for each game, group, and user. It includes the percentage of success (defined as the number of units completed divided by the number of units started) and the maximum unit reached by the player.

These metrics have been implemented in our framework using SparQL queries. They are later used to test the system's performance and serve as an example in our use case validation.

4.2 | Performance evaluation

In our performance evaluation, we evaluate the impact of our framework computation and analyze our approach's scalability when the dataset size increases. Specifically, we focus on examining the flexibility (how quickly our approach processes different types of metrics) and scalability (how well our framework scales with larger RDF datasets). In the following subsections, we present the server configuration settings, the datasets used, and our findings.

4.2.1 | Experimental setup

For our experiments, we aimed to test our framework with real data. Therefore, we selected a diverse set of SGs from various knowledge domains to evaluate the interoperability of our approach. Field Day[71] is a research lab at the Wisconsin Center for Education Research, University of Wisconsin-Madison, that designs learning games and makes their game data publicly available. From this open game data, we selected five different SGs to use their data and test the capabilities of our framework.

As we see in Table 2, each one of these datasets contains a total of 2M game events derived from real players' interaction with the games. To test the system's scalability when increasing the size of the dataset, we partitioned each dataset into smaller parts to have 100k, 250k, 500k, and 1M events datasets. The number of triples in our experiments varies from 1.6 to 34.8M, depending on the game and the dataset size.

TABLE 2 Dataset sizes.

Game	Size (GB)	# of triples
Crystal	2.98	34,500,365
Balloon	2.87	32,665,037
Cycle carbon	2.93	33,973,329
Magnet	2.76	31,249,623
Waves	3.01	34,780,117

We implemented our approach using Python 3.8, Spark-3.0.1, Scala 2.12.11, Java 11, and all the data were stored on an HDFS cluster using Hadoop 2.10.2. All experiments were conducted on a cluster of six nodes: one master and five workers. The cluster had a total of 36 cores (six cores per worker), 112 GB RAM (32 GB for the server node, 16 GB for each worker), and 3 TB SSD storage with a speed of 12 GB/s.

4.2.2 | Performance results

We evaluate our approach using the experimental setup described in the previous section and the metrics described in Section 4.1. We assessed the runtime of our distributed framework throughout the entire pipeline, from processing the raw events to calculating the metrics using SparQL queries. We ran experiments on five different sizes to measure the performance of size-up scalability. Since the ontology data processing stage is run locally, we did not include this as part of the node scalability performance evaluation. The average execution time of this stage is shown in Table 3. Then, to measure the performance of node scalability, we run experiments using one to five worker nodes on each of the five dataset sizes. Since we selected data from five games, we executed each experiment using those five datasets. The average execution time is presented in Table 4 and Figure 4.

In Table 4, we highlight the best execution time for each dataset size and stage/query in green. As we can see, the inference time benefits from increasing the number of workers as the dataset size increases. In fact, we see that the best execution time for 100k events is given by using two workers; meanwhile, the best execution time for 2M events is given by using five workers. Regarding the different metrics, querying the RDF triples also benefits from increasing the number of workers. However, in most cases, the best performance is achieved using four workers, except for some specific metrics with 1 and 2M events, which show better performance using five workers. In addition, we also see that the one worker cluster fails to process the 2M events dataset due to working memory errors.

As we can observe in Figure 5, the execution time grows linearly when the size of the datasets increases, demonstrating the scalability of our approach when using three or more workers. Furthermore, the query execution time varies depending on the metric being computed. For instance, with a dataset of 2M events, the framework can compute the “action indicators” metric in an average of 13.6 s, while the “event types” metric takes an average of 203.8 s to calculate. This discrepancy is due to the different SparQL queries designed for each metric, as they involve different types of operations. Taking the same example as before, the “action indicators” metric uses simple filtering and aggregation operations. In contrast, the “event types” metric uses several join operations, significantly increasing the computational cost.

The total execution time for each experiment is shown in Figure 4. When using smaller datasets, we can see that the performance improvement when using more workers could be more remarkable. However, with four and five workers, performance slightly decreases. For instance, when computing a 100k events file, the average execution time increases from 180.6 s using two workers to 199.8 s using five workers. With larger datasets (1 and 2M events), there is a performance improvement when using three and four workers, but the impact of using five workers is not significant. For the 1M events experiments, we obtain an average execution time of 1047.8 s using four workers and an average of 1070.6 s using five workers. When computing 2M events, if we compare the two workers configuration and the four workers configuration, the average execution time is 69% lower using four workers. With these results, combined with the fact that most of the lowest query execution times were given by the four workers cluster, we can affirm that using this configuration is the best option to obtain better performance and save the resources required by an additional worker node.

TABLE 3 Ontology data processing execution time.

# Events	Ontology data processing time (s) (mean)
100k	80.2
250k	271
500k	762.4
1M	2112.8
2M	4017

TABLE 4 Performance analysis on large-scale GBA datasets.

# Events	# Workers	Runtime (s) (mean) Inference	Levels of activity	Persistence	Activity indicators	Event types	Funnel by user	User performance
100k	1	92	20	16.2	8.2	51.6	22.4	10.6
	2	83.4	15	10.6	6	41.6	16.8	7.2
	3	86.4	15	10.6	6	41.6	16.8	7.2
	4	97.4	13.8	8.4	5.2	37.4	16.2	5
	5	103.2	15.8	10.2	5.2	40.4	17.8	7.2
250k	1	287.4	27.4	20.8	11.6	71.8	34.6	14.8
	2	216.8	16.6	12.4	7.2	49.6	26	9.6
	3	202	15.2	9.8	6.4	44.4	23.6	7.6
	4	223.4	15	9	5.4	41.4	23.8	7.2
	5	248	17.6	12.6	7	44.6	29.8	8.6
500k	1	744.8	37.8	26	13.2	110	50	21.2
	2	503	27.6	15.8	9	70.6	34.8	11.6
	3	404.6	24.2	13.8	9.2	66.2	41.2	10
	4	437	23	10.4	9.6	54.2	38	8.6
	5	433.8	23.2	13.2	10.6	57.6	43	9.8
1M	1	1249.8	60	33.4	19.2	218	78.6	30.8
	2	902.2	38	20	14.2	151	47.8	16
	3	812.4	31.4	18.6	16.2	149.8	45.2	15.6
	4	788.8	33.4	17.2	11.6	144	41.6	11.2
	5	807	32.6	18.4	11.4	149.4	40.8	11
2M	1	FAIL						
	2	2803	63.8	33.2	16.4	231.6	140.6	25.6
	3	1940.6	55.8	28.2	14.8	218.6	132.2	22
	4	1532.8	51.8	24.2	13.6	203.8	123.2	16.8
	5	1522.6	56.6	29.6	14.6	215	119.4	19

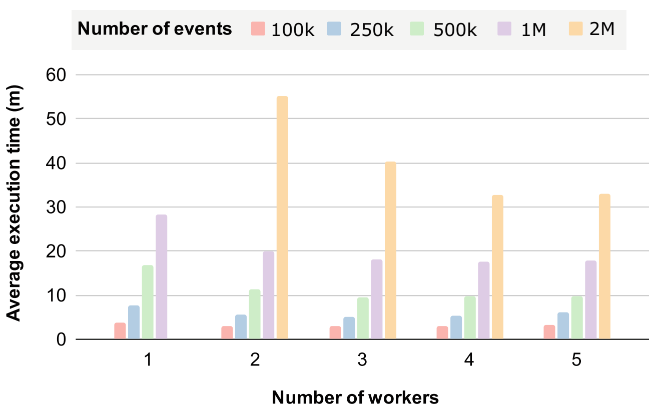
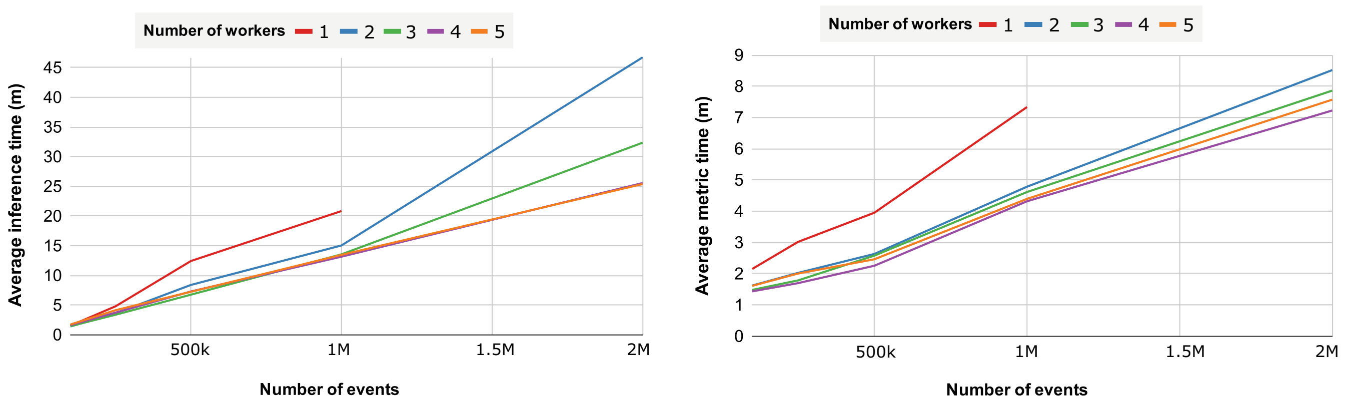


FIGURE 4 Size and worker scalability analysis.



(A) (B)

FIGURE 5 Scalability analysis. (A) Inference execution time per number of events and workers. (B) Metric execution time per number of events and workers.

4.3 | Case study validation

In this section, we present a case study with two use cases to exemplify how our framework and metrics can be applied in a real context. First, we conducted a use case by designing and implementing a dashboard that utilizes the analyzed and transformed data in the form of metrics, which can be consumed through visualizations. Second, we implemented a learner report system that enables instructors to easily track their learners' progress over time. The use cases performed show how, using the API, we provide a straightforward interface that can be used from almost any device, meeting **R4** and using the results from meeting the rest of **Requirements**.

4.3.1 | Use case: Dashboard

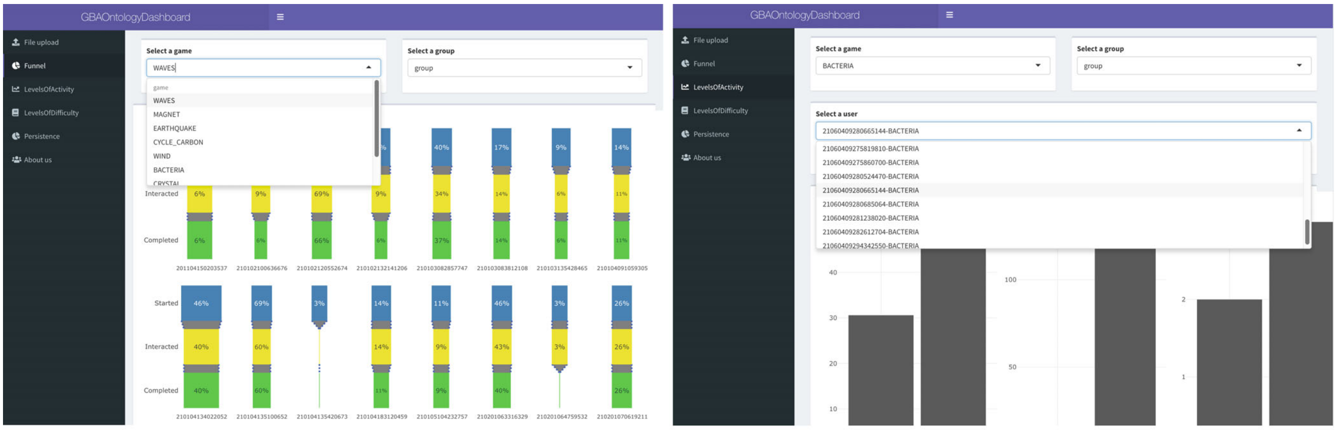
In this first use case, we introduce a visualization dashboard system that leverages the data analyzed and transformed into metrics by our framework. This dashboard utilizes specific API calls to input new data into the system or retrieve existing metrics from different games and groups. It enables (1) instructors to monitor learners' interactions with games, adapting their interventions based on these insights or using the metrics for formative evaluation, and (2) learners to track their own game-related activity. The dashboard aligns with the different roles defined within the framework. We developed the dashboard using the Shiny framework in R and deployed it on the ShinyApps web server.

In Figure 6, we can see the dashboard running live on the ShinyApps server. Users can log into the system using a username and a password. Each user will have different permissions and functionalities depending on the credentials used, complying with **R5**. For example, instructors and administrators can upload new GBA data in the "file upload" tab. Users can also navigate through the available tabs to upload new data or query the different metric results calculated.

Finally, we can see how the dashboard fully benefits the interoperability between games and metrics. The user can use selection boxes to choose between games and groups, and also between users depending on the granularity of the metric. That way, when a game is selected among the available options, the system loads the existing groups for that specific game in the corresponding selection box. When a group is selected, the system loads the existing users for that specific group. Once all the selection boxes for that metric tab are filled with a choice, the system queries the necessary information and represents it using interactive visualizations, as shown in Figure 6A,B.

4.3.2 | Use case: Reports

This second use case consists of a learner report summarizing the learners' progress using different games. The report is automatically generated using RMarkdown and outputs a PDF or HTML file that is sent to instructors periodically (the frequency in which the reports are generated can be adjusted in the system). Regarding the connection with the



(A)

(B)

FIGURE 6 Screenshots of the dashboard developed. (A) Game and group selection box in funnel by puzzle metric. (B) Game, group and user selection box in levels of activity metric.

framework, our report system uses specific query API calls, and the retrieved results are then employed to build the different parts of the report. For this specific example, we selected an instructor that has access to two different games: EARTHQUAKE and MAGNET, and specifically to the "MainGroup" data of each game. Some examples of information that these reports include are shown in Figure 7. Specifically, in Figure 7A we can see the first part of the report. Here, we see a group summary for each game, including some key metrics such as the total active time in seconds, the total number of units started, the total number of units completed, or the units that have been more problematic for learners. Then, in Figure 7B, we can see an individual report for each learner, also showing similar key metrics; and finally, in Figure 7C, a plot showing the learners' persistence. In this plot, each bubble represents a different learner, the x-axis represents the number of attempted units, and the size of each bubble represents the number of completed units. Moreover, the y-axis represents the average persistence percentile, so more persistent learners will be at the top of the plot. This report provides an easy way to monitor groups and learners while playing different games, allowing instructors to perform quick assessments based on different metrics calculated automatically using learners' data.

5 | DISCUSSION

SGs are considered practical tools in multiple domains. In particular, it is believed that its use for assessment (GBA) will be an increasing part of testing programs in future generations. This is due to their promising possibilities for more valid and reliable measurement of learners' skills compared to traditional assessment methods, such as paper-and-pencil tests.[72] However, the time and cost-intensive process of developing digital learning or assessment environments restricts the practical implementation of GBAs. Furthermore, the limited interoperability of assessment and tracking systems across different platforms presents a critical constraint in this area.[73] Our approach addresses these limitations by developing a framework that incorporates an intermediate semantic layer to enable interoperable GBAs. By utilizing an ontology as a common knowledge model, our framework can integrate log events from diverse games into a unified data model. This, combined with the use of interoperable RDF metrics, promotes standardization in the field and facilitates the utilization of numerous games, each designed for specific purposes, knowledge domains, and target participants.

Another known constraint in the area is the use of small data sample sizes. Generally, sample sizes used in GBA studies are pretty limited in size, resulting in low statistical power and a reduced chance of detecting actual effects.[53,74] Although collecting large samples of in-context data is a challenging task,[55] future research should use larger data samples in order to improve the results generalization and validity. This would also enable the use of more complex techniques, such as neural networks, which often require large amounts of data to outperform other models. Our contribution involves leveraging big data technologies to efficiently process large quantities of GBA data. Using a cluster of four worker nodes, our framework can process 2M events (including the computation of the six different metrics) in an average of 6434 s (107.2 min). We estimated that each user produces approximately 512 game events/hour based on fifteen different datasets from our experiments. Considering a classroom of 25 learners using a game for one hour/week, it would result in 51,200

GBA Ontology Framework Report

This report has been generated automatically for instructor with username: teacher1

This report includes MainGroup from the game EARTHQUAKE and MainGroup from the game MAGNET.

Group summary

This has been the performance of each group for the selected games:

Summary for each group.

Game	Group	Avg Active Time By Attempt	Total Active Time	Avg Events By Attempt	Total Events	Total Units Started	Total Units Completed	Most Difficult Unit
EARTHQUAKE	MainGroup	108.62055	21398.25	21.954315	4325	354	16	EARTHQUAKE-36
MAGNET	MainGroup	34.69733	157005.42	8.273149	37436	5935	3318	MAGNET-58

For each game, the units that could be problematic (abandoned percentage > 75%) are:

Game	Unit	Percentage Abandoned
EARTHQUAKE	EARTHQUAKE-18	100.0000
EARTHQUAKE	EARTHQUAKE-19	100.0000
EARTHQUAKE	EARTHQUAKE-36	76.4706
EARTHQUAKE	EARTHQUAKE-47	100.0000
EARTHQUAKE	EARTHQUAKE-6	100.0000
MAGNET	MAGNET-0	100.0000

(A)

General student report

This section provides a general overview of users' progress in selected games.

General performance

Summary for each user.

Game	User	Avg Active Time By Attempt	Total Active Time	Total Inactive Time	Avg Events By Attempt	Total Events	Total Units Started	Total Units Completed	Total Time
EARTHQUAKE	21050212144374524	127.9285	255.857	113.035	21.5	43	2	0	368.892
EARTHQUAKE	21050408083960244	79.2530	158.506	124.032	18.0	36	2	0	282.538
EARTHQUAKE	21050410373924384	61.8830	61.883	79.698	34.0	34	1	0	141.581
EARTHQUAKE	21060113472500124	10.4990	10.499	0.000	1.0	1	2	0	10.499
EARTHQUAKE	21070005462090520	151.8510	151.851	40.175	50.0	50	1	0	192.026
EARTHQUAKE	21070008000370016	8.3100	8.310	0.000	1.0	1	1	0	8.310
EARTHQUAKE	21070008572818384	101.0990	101.099	308.638	36.0	36	1	0	409.737
EARTHQUAKE	21070009344713380	77.0570	77.057	0.000	5.0	5	1	0	77.057
EARTHQUAKE	21070012163629812	63.7750	63.775	184.488	15.0	15	1	0	248.263
EARTHQUAKE	21070012553624696	50.6720	50.672	0.000	8.0	8	1	0	50.672

(B)

FIGURE 7 Learners' report screenshots. (A) Group summary. (B) General student report. (C) General learners report.

events per class and month. This implies that our framework can process data from approximately 39 full classrooms for an entire month in just 107.2 min. Additionally, our approach supports streaming data, as log events are received individually, allowing for real-time processing and just-in-time feedback. Since only new events are considered in each processing iteration, the data size processed is significantly reduced.

In-game metrics are necessary and essential, but we have to choose the most appropriate ones depending on each project. The most used metrics everywhere in any platform are performance metrics.[75] However, beyond performance, we can obtain further insights from the analysis of learner-generated information. Actions and behaviors should be convertible into metrics to identify learners' individual characteristics (including behaviors, performance, or skills) and learner-generated game data (e.g., time spent, goals, tasks completed).[52,76] After reviewing previous GBA literature, we found a set of commonly used metrics to replicate in our system. As mentioned earlier, one challenge is that these metrics and indicators are typically designed and developed specifically for each game. In this study, we have successfully replicated and integrated all of these previously established metrics into our framework, showing the interoperability between different games and excellent performance using large-scale datasets. For example, the "levels of activity" metric takes an average of 51.8 s to compute using 2M events, and the metric "user performance" takes an average of only 16.8 s using the same number of events. This achievement is made possible through the querying module, which translates SparQL code into Spark executable code, enabling the creation of interoperable metrics that measure not only performance-related characteristics but also other types of skills and behaviors. For example, we could apply clustering techniques to identify distinct student behaviors by utilizing SparQL code to collect student features, followed by employing the ML module provided by SANSA.

Typically, GBA systems develop their own interfaces to interact with external data sources. In our system, we have integrated a service API, which allows for easy insertion and retrieval of data, facilitating the interaction of various sources with our framework. One of the main advantages of our approach is its simplicity and ease of use. By integrating the API, users can easily build applications that connect to the framework and access its capabilities. This approach offers a number of benefits, including the ability to scale the service to meet the demands of a large number of users, the ability to easily update and maintain the service, and the ability to offer a seamless user experience across different devices. Overall, the integration of an API into a framework is a key enabler for the GBAaaS paradigm and offers a number of advantages for both GBA researchers and users. One potential limitation is the risk of external users accessing confidential information. To address this concern, we have developed an authorization and authentication module that controls access to each resource, ensuring maximum user data privacy.

This work also has some limitations: first, although we have defined an ontology with terms and concepts that almost any log data from the area should have, there is still a manual process of adapting the GBA data to our ontology to meet the input's requirements. Future researchers could take into account the ontology in the collected data design to skip this manual step. Second, the ontology processing data stage is run locally, which does not allow to take full advantage of the possibilities and performance that distributed-systems have. In addition, our framework (with the current configuration described in Section 4.2.1) cannot compute datasets with more than 3M events in a single batch due to working memory limitations. However, the system can solve this increasing resources or by splitting those files into smaller chunks and processing them sequentially. Finally, the system supports ML techniques but does not support more complex methods, such as knowledge inference or DL, which are also common in the GBA field.[52] Using these methods could help infer more helpful information from learners' data and improve the results' validity and reliability.

6 | CONCLUSIONS AND FUTURE WORK

This research aimed to create a robust novel framework for enabling GBAaaS using ontologies and big data technologies. Moreover, we demonstrated its capabilities by replicating existing metrics in GBA literature and conducting a case study with two use cases to show how external users can consume the system as a service. We also conducted a performance evaluation using different cluster configurations, concluding that using a cluster of one master node and four worker nodes was the best option in terms of resource management and performance. This cluster configuration was capable of processing 2M user events (approximately the size of 39 classrooms using a game for one hour/week for one month) in an average of 107.2 min.

As part of our future work, we want to validate our approach by conducting case studies in which the framework will be used in real-time, collecting data from learners and instructors, and validating the data streaming functionality implemented. Moreover, we would like to continue developing new GBA metrics that could use more advanced techniques, such as ML algorithms. Additionally, DL models could also be developed to test their predictive performance for inferring students' knowledge using existing data. Despite all the benefits that the application of ML and DL could have, most non-technical users perceive them as "black boxes." In this regard, future work should address the use of eXplainable Artificial Intelligence (XAI) approaches, enabling non-technical users (such as teachers) to interpret AI-generated insights and recommendations, empowering them to make informed decisions. Finally, we plan to integrate the ontology data processing stage (which is currently running locally) into the distributed environment to take full advantage of the cluster capabilities and obtain even better performance results. Future work could also address the deployment of the service in the cloud. Data and applications hosted on the cloud allow businesses to be more responsive and adaptable, becoming more efficient, strategic, and insight-driven.[77] Additionally, the use of fog computing could also be introduced in our platform, extending cloud computing due to its low latency, energy efficiency and the reduction in bandwidth required for data transport.[77]

This research contributes significantly to the current state of the art, including a completely novel framework that enables interoperable GBAs using large-scale data, privacy management, and easy interaction from external sources. We expect our contributions to solve current limitations regarding GBA interoperability, reducing the cost and effort that designing and performing specific GBAs have and allowing the deployment of GBAaaS.

AUTHOR CONTRIBUTIONS

Manuel J. Gomez: Conceptualization; methodology; software; validation; formal analysis; data curation; writing – original draft; visualization. **José A. Ruipérez-Valiente**: Conceptualization; writing – review and editing; supervision; project administration. **Félix J. García Clemente**: Conceptualization; writing – review and editing; supervision; project administration.

ACKNOWLEDGMENTS

This work was partially supported by (a) Grant 21795/FPI/22 - Séneca Foundation. Cofinanced by Innovatiio Global Educación. Region of Murcia (Spain), and (b) REASSESS project (Grant 21948/JLI/22), funded by the Call for Projects to Generate New Scientific Leadership, included in the Regional Program for the Promotion of Scientific and Technical Excellence Research (2022 Action Plan) of the Seneca Foundation, Science and Technology Agency of the Region of Murcia.

CONFLICT OF INTEREST STATEMENT

The authors declare no potential conflict of interests.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

ORCID

Manuel J. Gomez https://orcid.org/0000-0003-0571-2923
José A. Ruipérez-Valiente https://orcid.org/0000-0002-2304-6365
Félix J. García Clemente https://orcid.org/0000-0001-6181-5033

REFERENCES

1. Gros B. Digital games in education: the design of games-based learning environments. *J Res Technol Educ*. 2007;40(1):23-38.
2. Daniel M, Garry C. *Video Games as Culture: Considering the Role and Importance of Video Games in Contemporary Society*. Routledge; 2018.
3. Marklund BB, Backlund P, Engstrom H. The practicalities of educational games: challenges of taking games into formal educational settings. *2014 6th International Conference on Games and Virtual Worlds for Serious Applications*. IEEE; 2014:1-8.
4. Boyle EA, Hainey T, Connolly TM, et al. An update to the systematic literature review of empirical evidence of the impacts and outcomes of computer games and serious games. *Comput Educ*. 2016;94:178-192.
5. Alonso-Fernández C, Cano AR, Calvo-Morata A, Freire M, Martínez-Ortiz I, Fernández-Manjón B. Lessons learned applying learning analytics to assess serious games. *Comput Human Behav*. 2019;99:301-309.
6. Laamarti F, Eid M, El Saddik A. An overview of serious games. *Int J Comput Games Technol*. 2014;2014:358152.
7. Kato PM, Klerk S. Serious games for assessment: welcome to the jungle. *J Appl Test Technol*. 2017;18(S1):1-6.
8. Shute VJ. Stealth assessment in computer-based games to support learning. *Comput Games Instr*. 2011;55(2):503-524.
9. Kim YJ, Shute VJ. The interplay of game elements with psychometric qualities, learning, and enjoyment in game-based assessment. *Comput Educ*. 2015;87:340-356.
10. Ruipérez-Valiente JA, Gomez MJ, Martínez PA, Kim YJ. Ideating and developing a visualization dashboard to support teachers using educational games in the classroom. *IEEE Access*. 2021;9:83467-83481.
11. Fathy N, Gad W, Badr N. A unified access to heterogeneous big data through ontology-based semantic integration. *2019 Ninth International Conference on Intelligent Computing and Information Systems*. IEEE; 2019:387-392.
12. Gómez-Pérez A, Fernández-López M, Corcho O. *Ontological Engineering: with Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer Science & Business Media; 2006.
13. Ifenthaler D, Eseryel D, Ge X. *Assessment for Game-Based Learning*. Springer; 2012:1-8.
14. Freire M, Serrano-Laguna Á, Manero B, Martínez-Ortiz I, Moreno-Ger P, Fernández-Manjón B. Game learning analytics: learning analytics for serious games. In: Spector M, Lockee B, Childress M, eds. *Learning, Design, and Technology*. Springer; 2016:1-29.
15. Bienkowski M, Feng M, Means B. *Enhancing Teaching and Learning through Educational Data Mining and Learning Analytics: an Issue Brief*. US Department of Education: Office of Educational Technology; 2012.
16. Serrano Á, Marchiori EJ, Blanco Á, Torrente J, Fernández-Manjón B. A framework to improve evaluation in educational games. *Proceedings of the 2012 IEEE Global Engineering Education Conference*. IEEE; 2012:1-8.
17. Sejdiu G, Rula A, Lehmann J, Jabeen H. A scalable framework for quality assessment of RDF datasets. In: Ghidini C, Hartig O, Maleshkova M, et al., eds. *The Semantic Web – ISWC 2019*. Springer; 2019:261-276.
18. Mami MN, Graux D, Scerri S, Jabeen H, Auer S, Lehmann J. Squerall: virtual ontology-based access to heterogeneous and large data sources. In: Ghidini C, Hartig O, Maleshkova M, et al., eds. *The Semantic Web – ISWC 2019*. Springer; 2019:229-245.

19. Lehmann J, Sejdiu G, Bühmann L, et al. Distributed semantic analytics using the SANSA stack. *International Semantic Web Conference*. Springer; 2017:147-155.
20. Susi T, Johannesson M, Backlund P. Serious games: an overview. Technical Report. Institutionen för kommunikation och information; 2007.
21. Gomez MJ, Ruipérez-Valiente JA, Martínez PA, Kim YJ. Applying learning Analytics to detect sequences of actions and common errors in a geometry game. *Sensors*. 2021;21(4):1025.
22. Wang R, DeMaria S Jr, Goldberg A, Katz D. A systematic review of serious games in training health care professionals. *Simul Healthc*. 2016;11(1):41-51.
23. Proença JP, Quaresma C, Vieira P. Serious games for upper limb rehabilitation: a systematic review. *Disabil Rehabil Assist Technol*. 2018;13(1):95-100.
24. Samčović AB. Serious games in military applications. *Vojnoteh Glas*. 2018;66(3):597-613.
25. Albaladejo-González M, Strukova S, Ruipérez-Valiente JA, Gómez Mármol F. Exploring the affordances of multimodal data to improve cybersecurity training with cyber range environments. VI Jornadas Nacionales de Investigación en Ciberseguridad (JNIC 2021); 2021.
26. Ruiperez-Valiente JA, Gaydos M, Rosenheck L, Kim YJ, Klopfer E. Patterns of engagement in an educational massively multiplayer online game: a multidimensional view. *IEEE Trans Learn Technol*. 2020;13(4):648-661.
27. Harpstead E, Zimmermann T, Nagapan N, et al. What drives people: creating engagement profiles of players from game log data. *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play*. ACM; 2015:369-379.
28. DiCerbo KE. Game-based assessment of persistence. *J Educ Technol Soc*. 2014;17(1):17-28.
29. Ruipérez-Valiente JA, Kim YJ. Effects of solo vs. collaborative play in a digital learning game on geometry: results from a K12 experiment. *Comput Educ*. 2020;159:104008.
30. Kim YJ, Lin G, Ruipérez-Valiente JA. *Expanding Teacher Assessment Literacy with the Use of Data Visualizations in Game-Based Assessment*. Springer; 2021:399-419.
31. Jaffal Y, Wloka D. Employing game analytics techniques in the psychometric measurement of game-based assessments with dynamic content. *J e-Learn Knowl Soc*. 2015;11(3):101-115.
32. Dziedzic D, Włodarczyk W. Approaches to measuring the difficulty of games in dynamic difficulty adjustment systems. *Int J Hum–Comput Interact*. 2018;34(8):707-715.
33. Kiili K, Moeller K, Ninaus M. Evaluating the effectiveness of a game-based rational number training-in-game metrics as learning indicators. *Comput Educ*. 2018;120:13-28.
34. Lindenmayer JP, Goldring A, Borne S, et al. Assessing instrumental activities of daily living (iADL) with a game-based assessment for individuals with schizophrenia. *Schizophr Res*. 2020;223:166-172.
35. Jackson DJ, Kim S, Lee C, Choi Y, Song J. Simulating déjà vu: what happens to game performance when controlling for situational features? *Comput Human Behav*. 2016;55:796-803.
36. Crytek U. *Crysis 2*. Electronic Arts; 2011.
37. Serrano-Laguna Á, Martinez-Ortiz I, Haag J, Regan D, Johnson A, Fernández-Manjón B. Applying standards to systematize learning analytics in serious games. *Comput Stand Interfaces*. 2017;50:116-123.
38. Said B, Cheniti-Belcadhi L, El Khayat G. An ontology for personalization in serious games for assessment. *2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*. IEEE; 2019:148-154.
39. Rocha OR, Zucker CF. Ludo: an ontology to create linked data driven serious games. ISWC 2015—Workshop on LINKed EDucation; 2015.
40. Staab S, Studer R. *Handbook on Ontologies*. Springer Science & Business Media; 2010.
41. Zada I, Shahzad S, Ali S, Mehmood RM. OntoSuSD: Software engineering approaches integration ontology for sustainable software development. *Softw Pract Exp*. 2023;53(2):283-317.
42. Al-Chalabi HKM, Hussein AMA. Ontology applications in E-learning systems. *2020 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*. IEEE; 2020:1-6.
43. Andrews-Todd J, Kerr D. Application of ontologies for assessing collaborative problem solving skills. *Int J Test*. 2019;19(2):172-187.
44. Gomez MJ, Ruipérez-Valiente JA, Clemente FJG. Developing and Validating Interoperable Ontology-Driven Game-Based Assessments (In Review); 2022.
45. Botoeva E, Calvanese D, Cogrel B, Corman J, Xiao G. A generalized framework for ontology-based data access. In: Ghidini C, Magnini B, Passerini A, Traverso P, eds. *AI*IA 2018 – Advances in Artificial Intelligence*. Springer; 2018:166-180.
46. Zhou J, Ma L, Liu Q, Zhang L, Yu Y, Pan Y. Minerva: a scalable OWL ontology storage and inference system. In: Mizoguchi R, Shi Z, Giunchiglia F, eds. *The Semantic Web – ASWC 2006*. Springer; 2006:429-443.
47. Dehainsala H, Pierra G, Bellatreche L. Ontodb: an ontology-based database for data intensive applications. In: Kotagiri R, Krishna PR, Mohania M, Nantajeewarawat E, eds. *International Conference on Database Systems for Advanced Applications*. Springer; 2007:497-508.
48. Abbes H, Gargouri F. Big data integration: a MongoDB database and modular ontologies based approach. *Procedia Comput Sci*. 2016;96:446-455.
49. Mountasser I, Ouhbi B, Hdioud F, Frikh B. Semantic-based big data integration framework using scalable distributed ontology matching strategy. *Distrib Parallel Databases*. 2021;39(4):891-937.
50. Reyes-Álvarez L, Roldán-Garcia MM, Aldana-Montes JF. Tool for materializing OWL ontologies in a column-oriented database. *Softw Pract Exp*. 2019;49(1):100-119.
51. Verma V, Baron T, Bansal A, Amresh A. Emerging practices in game-based assessment. In: Ifenthaler D, Kim YJ, eds. *Game-Based Assessment Revisited*. Springer; 2019:327-346.

52. Kim YJ, Ifenthaler D. Game-based assessment: the past ten years and moving forward. In: Ifenthaler D, Kim YJ, eds. *Game-Based Assessment Revisited*. Springer; 2019:3-11.
53. Alonso-Fernandez C, Calvo-Morata A, Freire M, Martinez-Ortiz I, Fernández-Manjón B. Applications of data science to game learning analytics data: a systematic literature review. *Comput Educ*. 2019;141:103612.
54. Pérez-Berenguer D, Garcia-Molina J. A standard-based architecture to support learning interoperability: a practical experience in gamification. *Softw Pract Exp*. 2018;48(6):1238-1268.
55. Gomez MJ, Ruipérez-Valiente JA, Clemente FJG. A systematic literature review of game-based assessment studies: trends and challenges. *IEEE Trans Learn Technol*. 2022.
56. Gursoy ME, Inan A, Nergiz ME, Saygin Y. Privacy-preserving learning analytics: challenges and techniques. *IEEE Trans Learn Technol*. 2016;10(1):68-81.
57. Takabi H, Joshi JB, Ahn GJ. Security and privacy challenges in cloud computing environments. *IEEE Secur Priv*. 2010;8(6):24-31.
58. Fernández-López M, Gómez-Pérez A, Juristo N. Methontology: from ontological art towards ontological engineering. *Proceedings of the AAAI97 Spring Symposium*. AAAI Press; 1997:33-40.
59. Gomez MJ, Ruipérez-Valiente J, García Clemente FJ. Supplementary materials: a framework to support interoperable game-based assessments as a service (GBAaaS): design, development, and use cases; 2022. Accessed December 16, 2022. https://osf.io/ctb8p/?view_only=fe268264bd1346668ae05fa2e3048f8c
60. Bischof S, Decker S, Krennwallner T, Lopes N, Polleres A. Mapping between RDF and XML with XSPARQL. *J Data Semant*. 2012;1(3):147-185.
61. Gandon F, Bottollier V, Corby O, Durville P. RDF/XML source declaration; 2007.
62. Analytics SD. Sparklify; 2021.
63. Grobe M. RDF, Jena, SparQL and the 'semantic web'. *Proceedings of the 37th Annual ACM SIGUCCS Fall Conference: Communication and Collaboration*. Association for Computing Machinery; 2009:131-138.
64. Kim JM, Park YT. Scalable OWL-Horst ontology reasoning using SPARK. *2015 International Conference on Big Data and Smart Computing*. IEEE; 2015:79-86.
65. Pérez J, Arenas M, Gutierrez C. Semantics and complexity of SPARQL. *ACM Trans Database Syst*. 2009;34(3):1-45.
66. Stadler C, Sejdiu G, Graux D, Lehmann J. Sparklify: a scalable software component for efficient evaluation of SPARQL queries over distributed RDF datasets. In: Ghidini C, Hartig O, Maleshkova M, et al., eds. *The Semantic Web – ISWC 2019*. Springer; 2019:293-308.
67. Kim H, Lee EA. Authentication and authorization for the Internet of Things. *IT Prof*. 2017;19(5):27-33.
68. Hunt J. Play framework. *A Beginner's Guide to Scala, Object Orientation and Functional Programming*. Springer; 2018:431-446.
69. Software S. API development for everyone; 2022. Accessed April 28, 2022.
70. De B. *API Documentation*. Springer; 2017:59-80.
71. Day F. We're field day; 2022. Accessed April 29, 2022.
72. Klerk S, Kato PM. The future value of serious games for assessment: where do we go now? *J Appl Test Technol*. 2017;18(S1):32-37.
73. Hruska M, Long R, Amburn C, Kilcullen T, Poeppelman T. Experience API and team evaluation: evolving interoperable performance assessment. Proceedings of the Interservice/Industry Training, Simulation & Education Conference; 2014.
74. Petri G, Wangenheim CG. How games for computing education are evaluated? A systematic literature review. *Comput Educ*. 2017;107:68-90.
75. Junaidi J, Julianto A, Anwar N, Safrizal S, Warnars HLHS, Hashimoto K. Perfecting a video game with game metrics. *TELKOMNIKA*. 2018;16(3):1324-1331.
76. Loh CS, Sheng Y. Measuring expert performance for serious games analytics: from data to insights. *Serious Games Analytics*. Springer; 2015:101-134.
77. Gill SS, Xu M, Ottaviani C, et al. AI for next generation computing: emerging trends and future directions. *Internet Things*. 2022;19:100514.

How to cite this article: Gomez MJ, Ruipérez-Valiente JA, García Clemente FJ. A framework to support interoperable Game-based Assessments as a Service (GBAaaS): Design, development, and use cases. *Softw Pract Exper*. 2023;53(11):2222-2240. doi: 10.1002/spe.3254