A Ubiquitous Learning Analytics Architecture for a Service-Oriented MOOC Platform

Tobias Rohloff(✉), Jan Renz, Gerardo Navarro Suarez, and Christoph Meinel

Hasso Plattner Institute, Potsdam, Germany
{tobias.rohloff,jan.renz,christoph.meinel}@hpi.de

Abstract. As Massive Open Online Courses (MOOCs) generate a huge amount of learning activity data through its thousands of users, great potential is provided to use this data to understand and optimize the learning experience and outcome, which is the goal of Learning Analytics. But first, the data needs to be collected, processed, analyzed and reported in order to gain actionable insights. Technical concepts and implementations are rarely accessible and therefore this work presents an architecture how Learning Analytics can be implemented in a service-oriented MOOC platform. To achieve that, a service based on extensible schema-agnostic processing pipelines is introduced for the HPI MOOC platform. The approach was evaluated regarding its scalability, extensibility, and versatility with real-world use cases. Also, data privacy was taken into account. Based on five years of running the service in production on several platform deployments, six design recommendations are presented which can be utilized as best practices for platform vendors and researchers when implementing Learning Analytics in MOOCs.

Keywords: MOOCs · Learning Analytics · Service-Oriented Architecture

1 Introduction

Since the peak of the Massive Open Online Course (MOOC) hype in 2012 with "The Year of the MOOC" [5] and the subsequent natural disillusionment that followed, the global phenomenon is slowly reaching the *plateau of productivity* according to Gartner's hype cycle [2]. Many higher education institutions and companies make extensive use of MOOCs, which has resulted in numerous different platforms on the market [12]. As MOOCs are used by thousands of learners, a huge amount of learning activity data is generated. With methods from the research field of Learning Analytics, this data can be utilized to understand and optimize the learning and the environments in which it occurs [13]. In order to leverage the tremendous research potential, platform providers and vendors have to establish the means and tools for collecting, processing, analyzing and accessing the produced data. However, technical concepts and insights

M. Calise et al. (Eds.): EMOOCs 2019, LNCS 11475, pp. 162–171, 2019.
https://doi.org/10.1007/978-3-030-19875-6_19

are rarely published especially for modern Microservice-based application architectures. Therefore, this work examines the following research question: How can Learning Analytics be implemented in a service-oriented and multi-client MOOC platform?

To investigate this question, the contribution of this work is twofold. First, we present a technical architecture to implement Learning Analytics into a service-oriented large-scale online learning environment using the example of the MOOC platform from Hasso Plattner Institute (HPI)[1] (Sect. 3), based on the previously explained requirements in Sect. 2. Second, we evaluate this work technically and practically (Sect. 4), by introducing real-world Learning Analytics use cases and features which are realized with this approach. This allows platform vendors and researchers to utilize our insights and best practices to support decision making when implementing Learning Analytics in MOOCs and similar learning platforms. Section 5 concludes the paper.

2 Platform and Requirements

This section presents the technical foundation of the HPI MOOC platform, its architecture, and design decisions. This conceptual understanding is utilized to define the requirements to implement Learning Analytics in such a context.

2.1 From LMS to SOA

The initial version of the HPI MOOC platform was based on an open-source Learning Management System (LMS), in order to quickly experiment and test the platform with first courses in 2012, which was a pioneering work in Europe [4]. Based on these first insights, a custom-tailored platform was developed from scratch which fits better to the paradigm of MOOCs, with thousands of learners in a single course and social activity, as well as a better scalability and performance. Therefore, the current platform was developed based on the principles of a Service-Oriented Architecture (SOA) with logically separated functionality in individual services [3]. For example, the *account service* is responsible for managing users accounts and the *course service* manages all information regarding courses and course enrollments. The services can communicate with each other synchronously through RESTful HTTP interfaces, or asynchronously by publishing events on a shared message queue. Currently, there are three clients available for the platform: a web client served by the *web service* and two native mobile clients for Android and iOS, which use the platform's API.

2.2 Learning Analytics Implementation Requirements

An SOA leads to a distributed data landscape because every service manages its own data persistence layer, and these layers are eventually distributed across different physical machines and rely on different database technologies. This makes

[1] https://open.hpi.de/.

it inconvenient when performing analytical tasks. Each service has to offer different analytics endpoints, which can cause heavy load on the overall system and block incoming requests, especially when the data is calculated on-demand. This is due to the fact that Microservices are designed to support an operational Online Transaction Processing (OLTP) model. However, the support for Online Analytical Processing (OLAP) is required. In order to overcome this issue, an independent service is mandatory which provides analytics and statistics on separate data stores. Thereby, it must be *extensible* to cover different Learning Analytics use cases of different stakeholders, *flexible* to gather data from different system components and clients, *avoid high system load* and performance impact when gathering and processing the data, allow *instant data availability* and *ensure data privacy*.

3 Learning Analytics Architecture

To implement and fulfill the previously introduced requirements, this section explains the concept and architecture of the realized Learning Analytics service. A complete architecture overview of all system components including the Learning Analytics service can be seen in Fig. 1. The service was realized by following the approach of an Extract, Transform, Load (ETL) process, as introduced in [7]. This process is implemented as extensible processing pipelines. Every pipeline consists of an extraction, multiple transforming, and a loading step. The extraction step processes the raw data into a container format. Afterward, the transformation steps process the data and map them to the desired data schema. At last, the loading step persists them in different analytics stores. These steps are explained in detail in the following subsections.

3.1 Event-Driven Data Collection

The data collection and extraction is implemented by taking advantage of the publish-subscribe message queue. This enables an asynchronous event-driven inter-process communication. Every service can publish events on the message queue. Here, two types of events are used. First, general model changes, like when a model record was created, updated or deleted. Second, explicit analytics events. The Learning Analytics service subscribed itself for all analytics events, as well as certain model changes. The queue then notifies and passes all corresponding events to the Learning Analytics service. In this way, the asynchronous non-blocking communication avoids performance impacts on the overall system.

The data structure of the analytics events is inspired by the xAPI[2]: «Actor» does «Verb» on «Object», with «Result» in «Context» at «Timestamp». In the context of the platform, the *Actor* is called *User* and the *Object* is called *Resource*. The *User* is the person who triggered the event, the *Verb* is the action that is being done by the *User*, the *Resource* is the entity the action was done

[2] https://xapi.com/overview/.

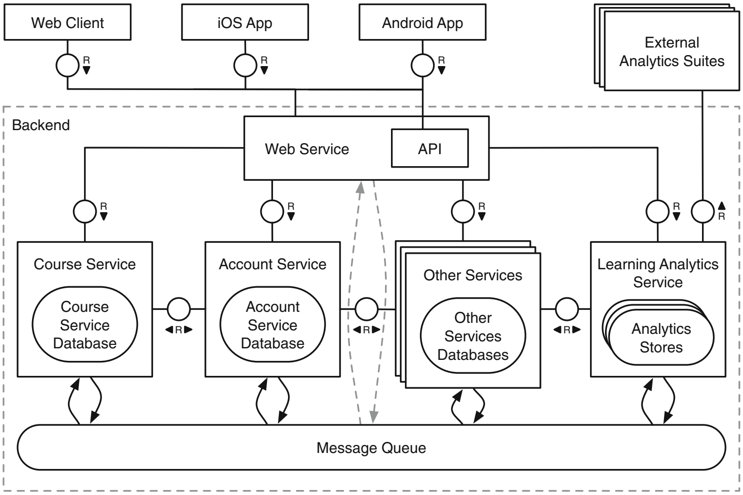

Fig. 1. The Platform's architecture with Learning Analytics service

at and the *Result* is the outcome of the action. The *Context* contains additional information which the action is related with and the *Timestamp* is the moment of the action.

3.2 Data Transformation with Processing Pipelines

The transformation steps process, enrich and clean the data. The first step processes the user-agent if the event was sent by the web client, to identify the user's operating system and browser. The next step determines a coarse location from the request's IP address to assess the country and city. The third step removes the user-agent and IP address from the event since all crucial information is already extracted from these attributes. They are classified as sensitive personal information, which makes it rather easy to identify a user when anonymized events with hashed user IDs are examined. The last step transforms the data into the appropriate schema of the targeted data storage.

3.3 Data Loading into Analytics Stores

The Learning Analytics service provides the possibility to host different data sources as analytics stores. This provides the advantage to store the same data redundantly – or different data – in various database technologies to optimize query performance. Each data source is configured with its own processing pipeline, whereby the extraction and transformation steps can be reused. The specific loading step stores the data at the end. The general concept of the service and its pipelines is shown in Fig. 2.

Currently, four different pipelines are used. User interaction events are stored redundantly in an SQL-based data source (PostgreSQL) and in a NoSQL-based data source (Elasticsearch). As an experiment also external analytics suites were tested as data stores. Therefore, a whitelisted subset of interaction events was anonymized and send to Google Analytics, which then serves as an analytics store [11]. At last, another pipeline is used to enable referrer tracking, which uses the Elasticsearch analytics store as well.

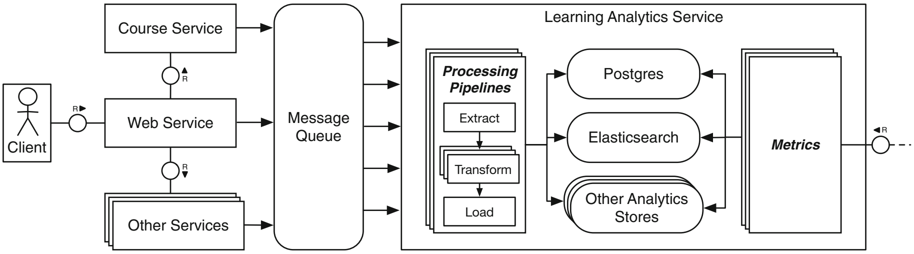

Fig. 2. The concept of the Learning Analytics service

3.4 Data Analysis with Metrics

Having the data stored in different analytics stores, it allows us to query the data, process them and expose insights as metrics within the platform. Every metric specifies its data source, optional and required parameters, and a short description with a custom Domain-Specific Language (DSL). This enables us to provide a self-documented endpoint for platform developers and researchers, introduce a standardized way to implement new metrics and support the discoverability of available metrics to increase the usage of data-driven insights, either for platform features or research studies. The calculation of a metric provides the possibility of pre- and post-processing of the data, as well as requesting the data source with its native query language.

3.5 Learning Analytics for Ubiquitous Learning

The broad availability of mobile devices has enabled Mobile Learning for online education like MOOCs [15]. By taking the user's context – like time and location – into account as well, the term Ubiquitous Learning arose. Therefore, the retrieval, analysis, and reporting of the data of mobile learners along with their contextual information is called Ubiquitous Learning Analytics [1].

To support this, two architecture components were enhanced [9]. First, a context model was defined and implemented. The contextual data is captured on the client-side and transferred to the Learning Analytics service as explained previously. The service applies additional extraction and cleaning transformation steps as part of its processing pipeline. Second, the client-side tracking capabilities were improved by supporting offline-usage and network interruptions. Therefore, all captured user interaction events are saved on a local database before transferred over the network once the device was connected again.

4 Evaluation

This section evaluates the implemented architecture based on the defined requirements. Therefore, the scalability, extensibility, and versatility are examined. Afterward, the data privacy mechanisms are reviewed. The section is closed by a presentation of compiled design recommendations and best practices.

4.1 Scalability

Since the implemented approach is used in a real-world MOOC platform with thousands of learners, it must be able to process the incoming data load and provide instant data availability. This means that a user always gets the latest data when requesting a certain metric, which is defined as a processing time for each event of at most one second. To evaluate the data load and availability, we examined a sample period of one year on the largest deployment of the platform[3]. The deployment consists of four web service nodes and four nodes with all other services, which means that the Learning Analytics service is also deployed four times redundantly for load balancing. The message queue used to publish events is hosted as a single instance, as well as the PostgreSQL database – which is one of the two analytics data stores. The other data store based on Elasticsearch is operated as a cluster with two nodes.

In the analyzed period from January 1, 2018, to December 31, 2018, a total number of 126,180,673 analytics events from 328,507 users were captured, which results in about four events per second on average. Although this number may seem low at first glance, it should be noted that the general activity on MOOC platforms varies significantly, depending on the time of day, course dates and deadlines. This results in periods of very high and low activity that must be considered separately. Therefore we examined the number of events waiting to be processed in the message queue per hour for the whole year. During the entire period, 67.6% of the time not a single event was waiting in the queue, which means every event was processed right away. In 31.1% of the captured hour intervals up to 14,400 events were waiting for a free consumer. This number was chosen since the four Learning Analytics service consumers are then theoretically stressed with one event per second on average, which is still considered as instant data availability. Based on this approximation, we achieved a total instant data availability in 98.7% of the time. The higher loads during the rest of the time are probably caused by infrastructure issues and not by activity peaks. To prevent data loss in such outages, all events are stored and kept as unacknowledged in message queue as long as the analytics stores are unavailable. All in all, we consider our architecture approach as proven to be suitable for the scale of a real-world MOOC platform.

[3] https://open.sap.com/.

4.2 Extensibility

An important requirement of the Learning Analytics service is to provide a flexible architectural design. It should be avoided to rebuild the whole architecture to include a new schema or data source. Thus, extensibility is ensured with the implemented processing pipeline design. New data which should be tracked can be published by other components through the message queue. Then, the Learning Analytics service can extract the data within its first pipeline step. A new data source can be added by providing a new load step, which maps the generic event schema to the specific database schema and executes the queries to persist the data. The modularity of the processing pipeline is the most valuable advantage. It can be easily extended or new pipelines can be created by providing additional transform or load steps. Also, every step can be reused by all pipelines.

4.3 Versatility

In this subsection, different use cases and features are explained that were implemented based on the presented Learning Analytics architecture. This is utilized to assess the versatility of the general approach.

As a typical use case, a *Teacher Dashboard* was implemented that visualizes various Learning Analytics metrics to give an overview of a course [11]. It includes enrollment numbers, active users and forum activity over time, as well as statistics about learning item visits, quiz performances, geographical learner locations, age distributions, used devices and learning times. Among other things, it supports teaching teams to identify anomalies and patterns in their courses, like too difficult learning content. Additionally, a *Learner Dashboard* was implemented and tested that gives students insights and feedback about their own learning behavior. It a based on concept to better support self-regulated learning, by providing personalized learning objectives a student can choose [10]. The dashboard should help to achieve that objective by enabling self-evaluation.

Unexperienced teaching teams or limited production times can lead to qualitative weaknesses in MOOCs. Therefore, it is valuable to assist with an *Automated Quality Assurance*, which Learning Analytics can enable [8]. Such a concept was implemented by translating best practices into machine-executable rules. These rules are checked periodically and a warning is issued if they are violated, whereby every warning is prioritized and linked with a recommendation for action. Two examples of such rules are too difficult quizzes or anomalies in student's video watching behavior, like too many rewinds. Another implemented feature enabled by Learning Analytics is the *Cluster Viewer*, which supports teachers to interactively explore meaningful subgroups of students by their learning activity to take informed action and measure the effect of performed interventions [14]. At last, the platform supports *A/B Testing*. With that, researchers can examine new features and compare the learning behavior and outcome of different test groups. This evaluation is based on Learning Analytics metrics, which are visualized and compared by their statistical differences and effect sizes [6].

The different presented use cases confirm the versatility of the implemented architecture. It allows to realize a broad range of techniques, ranging from simpler statistics and visualizations to more complex topics like data mining with clustering. Also, various stakeholder take advantage of the Learning Analytics capabilities, like teachers, learners, and researchers. This promises to be able to implement further requirements and use cases in the future as well.

4.4 Data Privacy

As the platform is developed and hosted in Germany, the European Union's General Data Protection Regulation (GDPR) is the law in force for governing processing of personal data. Since the Learning Analytic capabilities are exclusively used to improve the learning experience and optimizing the platform and its features, the data processing is considered as a legitimate interest. Therefore, no explicit consent is required from the user – as it would be for marketing purposes for example. Additionally, anonymization techniques are applied to further improve the data privacy of the tracked interaction data. Some attributes are omitted which are classified as personally identifiable information, e.g. the user's IP address and the browser's user-agent. No profile data is captured, like the user's name, email or date of birth. If some data is exported from the platform, the user IDs are additionally obfuscated. To ensure data reduction and data economy only relevant interaction events are captured, instead of tracking every single click on the platform.

4.5 Design Recommendations and Best Practices

Based on the experiences and insights we gathered in five years of running the Learning Analytics service in production on several platform deployments, we compiled a number of design recommendations for platform vendors and researchers. These best practices can support their decision making when implementing Learning Analytics capabilities into MOOC platforms.

Concurrent Data Collection and Processing Analytics, in general, can be seen as an extension to the main application. Thus, the performance impact on the overall application, caused by additional analytics tasks, should be kept to a minimum. A common technique is to execute such tasks concurrently. We realized this by utilizing an asynchronous message queue for event collection, to not block the sending components. The data processing is done by a separate service running independently from other system components.

Schema-Agnostic Pipelining Different data schemas and query requirements fit more or less well to different storage technologies. Therefore, various analytics data will eventually be stored in multiple databases. Hence, we recommend a pipeline processing architecture. By utilizing an ETL process for this, all data can be processed based on a generic data schema. Only the last load step converts the data into the database-specific format. This enables a schema-agnostic data processing and minimizes technology and vendor lock-ins.

Reusable Pipeline Components By utilizing the proposed schema-agnostic pipeline architecture, all transformation processing steps become reusable. For example, this allowed us to apply the same anonymization step to all of our analytics pipelines. This reduces implementation and maintenance efforts by applying the *don't repeat yourself* principle.

Central Interface for Data-Driven Insights Instead of having each application component providing its own analytics interface, it makes sense to have a central interface for data-driven insights. We realized this with an index of all available metrics within our Learning Analytics service. This also abstracts the underlying database technology.

Embrace Open Standards Interoperability with other applications and systems can best be achieved through the use of open standards. In the domain of Learning Analytics, the xAPI format has been accepted widely. This standard also defines the Learning Record Store. Thus, an implementation of such an analytics store could be used right away without further data transformations.

Data Protection by Design By taking data protection into account at every project stage, privacy risks can be reduced and trust increased. Users must stay in control of their data and the benefits of capturing and processing personal data should be communicated beforehand. It should also be ensured at an early stage that legal requirements like GDPR are complied with.

5 Conclusion

This work presented an architecture how Learning Analytics can be implemented in a service-oriented and multi-client MOOC platform. Based on the elaborated requirements, an ETL process was proposed to implement extensible processing pipelines within an independent Learning Analytics service. This approach utilizes an event-driven asynchronous data collection, a schema-agnostic data processing with reusable steps, and different analytics stores for optimized query performance. It was implemented for the HPI MOOC platform and deployed for real-world usage. User interaction events are captured with contextual data by different client applications, like the web client and mobile apps. This serves as the data foundation for Ubiquitous Learning Analytics, to generate data-driven insights about the learning behavior and create platform features to improve the learning experience and success.

Afterward, the architecture was evaluated to examine its scalability, extensibility, and versatility by discussing various implemented Learning Analytics use cases for different stakeholders like teachers and learners. Then, data privacy issues and mechanisms were presented, which also took the EU GDPR requirements into account. At last, six design recommendations – about concurrent data collection and processing, schema-agnostic pipelining, reusable pipeline components, centralized data-driven insights, open standards, and data protection – were introduced. These should serve as best practices for platform vendors and researchers, to support them during the implementation of Learning Analytics capabilities in MOOCs.

References

1. Aljohani, N.R., Davis, H.C.: Learning analytics in mobile and ubiquitous learning environments. In: 11th World Conference on Mobile and Contextual Learning (mLearn 2012), October 2012
2. Bozkurt, A., Keskin, N.O., de Waard, I.: Research trends in massive open online course (MOOC) theses and dissertations: surfing the tsunami wave. Open Prax. **8**(3), 203–221 (2016)
3. Meinel, C., Totschnig, M., Willems, C.: openHPI: evolution of a MOOC platform from LMS to SOA. In: Proceedings of the 5th International Conference on Computer Supported Education (CSEDU) (2013)
4. Meinel, C., Willems, C.: openHPI: the MOOC offer at Hasso Plattner Institute. Universitätsverlag Potsdam, Technical report (2013)
5. Pappano, L.: The Year of the MOOC (2012). http://www.nytimes.com/2012/11/04/education/edlife/massive-open-online-courses-are-multiplyingat-a-rapid-pace.html
6. Renz, J., Hoffmann, D., Staubitz, T., Meinel, C.: Using A/B testing in MOOC environments. In: Proceedings of the Sixth International Conference on Learning Analytics & Knowledge, pp. 304–313. ACM (2016). https://doi.org/10.1145/2883851.2883876
7. Renz, J., Navarro-Suarez, G., Sathi, R., Staubitz, T., Meinel, C.: Enabling schema agnostic learning analytics in a service-oriented MOOC platform. In: Proceedings of the Third (2016) ACM Conference on Learning @ Scale, L@S 2016, pp. 137–140. ACM (2016). https://doi.org/10.1145/2876034.2893389
8. Renz, J., Rohloff, T., Meinel, C.: Automated quality assurance in MOOCs through learning analytics. In: Joint Proceedings of the Pre-Conference Workshops of DeLFI and GMW 2017. CEUR-WS.org (2017)
9. Rohloff, T., Bothe, M., Renz, J., Meinel, C.: Towards a better understanding of mobile learning in MOOCs. In: Learning with MOOCs 2018 (LWMOOCS V), September 2018. https://doi.org/10.1109/LWMOOCS.2018.8534685
10. Rohloff, T., Meinel, C.: Towards personalized learning objectives in MOOCs. In: Pammer-Schindler, V., Pérez-Sanagustín, M., Drachsler, H., Elferink, R., Scheffel, M. (eds.) EC-TEL 2018. LNCS, vol. 11082, pp. 202–215. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98572-5_16
11. Rohloff, T., Oldag, S., Renz, J., Meinel, C.: Utilizing web analytics in the context of learning analytics for large-scale online learning. In: 2019 IEEE Global Engineering Education Conference (EDUCON) (2019, in press)
12. Shah, D.: Massive List of MOOC Providers Around The World (2017). https://www.class-central.com/report/mooc-providers-list/
13. Siemens, G.: Call for Papers of the 1st International Conference on Learning Analytics & Knowledge (LAK 2011) (2010). https://tekri.athabascau.ca/analytics/
14. Teusner, R., Rollmann, K.-A., Renz, J.: Taking informed action on student activity in MOOCs. In: Proceedings of the Fourth ACM Conference on Learning @ Scale, pp. 149–152. ACM (2017). https://doi.org/10.1145/3051457.3053971
15. Thüs, H., et al.: Mobile learning in context. Int. J. Technol. Enhanc. Learn. **4**(5/6), 332–344 (2012). https://doi.org/10.1504/IJTEL.2012.051818