Title

moocRP: Enabling Open Learning Analytics with an Open Source Platform for Data Distribution, Analysis, and Visualization

Journal

Technology, Knowledge and Learning, 21(1)

ISSN

2211-1662

Authors

Pardos, Zachary A
Whyte, Anthony
Kao, Kevin

Publication Date

2016-04-01

DOI

10.1007/s10758-015-9268-2

Peer reviewed

Tech Know Learn (2016) 21:75–98
DOI 10.1007/s10758-015-9268-2

moocRP: Enabling Open Learning Analytics with an Open Source Platform for Data Distribution, Analysis, and Visualization

Zachary A. Pardos[1] · **Anthony Whyte**[2] · **Kevin Kao**[1]

Published online: 22 January 2016

Abstract In this paper, we address issues of transparency, modularity, and privacy with the introduction of an open source, web-based data repository and analysis tool tailored to the Massive Open Online Course community. The tool integrates data request/authorization and distribution workflow features as well as provides a simple analytics module upload format to enable reuse and replication of analytics results among instructors and researchers. We survey the evolving landscape of competing established and emerging data models, all of which are accommodated in the platform. Data model descriptions are provided to analytics authors who choose, much like with smartphone app stores, to write for any number of data models depending on their needs and the proliferation of the particular data model. Two case study examples of analytics and responsive visualizations based on different data models are described in the paper. The result is a simple but effective approach to learning analytics immediately applicable to X consortium MOOCs and beyond.

Keywords Open learning analytics · Modularization · MOOC · Dashboards · Visualizations · Reproducible research

✉ Zachary A. Pardos
pardos@berkeley.edu

Anthony Whyte
arwhyte@umich.edu

Kevin Kao
kkao@berkeley.edu

[1] University of California at Berkeley, Berkeley, CA 94720, USA

[2] University of Michigan, Ann Arbor, MI, USA

1 Introduction

Matters of practicality and principle face the communities of educational data mining (EDM) and learning analytics (LA) with both matters coming into particular relief on the topic of Massive Open Online Course (MOOC) data collection, management, distribution, and analytics. We introduce a new tool called moocRP, an open-source[1] analytics platform for the greater MOOC community that situates itself at the intersection of these issues.

Let us start with issues of practicality. The non-profit higher education platform, edX, now has 44 partner universities offering or set to offer free courses online. The majority of these universities do not know how to handle the data they receive from edX, nor what to do with it once handled. Coursera and other MOOC providers put stakeholders in a similar state of disorientation with the nuanced distinction that they deliver data direct to the instructor of record while edX delivers data at the feet of the institution. The institution must then co-ordinate distribution to the appropriate parties. The learning analytics community can provide much needed guidance on the cutting edge of what is actionable in the data and how it can be leveraged by instructor and student (Verbert et al. 2014) towards the betterment of the learning experience. Our moocRP platform can help expand the impact of learning analytics by enabling its use among a quickly expanding new cohort of online learners and instructors. The platform and its built-in tools provide answers to the practical questions of how to prepare the data, grant users access to it securely, and scrutinize the data for instructionally actionable information.

Matters of principle relevant to education data analysis and distribution include (a) transparency—what data is being collected, how is it being represented, and what exactly the various technical components are doing in the background to manipulate these data behind the analytic dashboard? To address this, our tool is open source, as is the technical design of the pipeline that serves analytics to its users. New analytics and visualizations that are uploaded must also be open-source to function on the system. In order for a data model to be imported into the system, it must describe the data elements it exposes and this description is available both to the authors and consumers of the analytics. The next principle is (b) modularity—in order to foster an inclusive community of analytic contributors, analytics must be easily incorporated into the tool. We specify a simple but feature rich modular format that lowers the technical and time requirements to contribute to analytics in higher ed. Lastly, c) privacy is of particular concern in the current climate of big data, surveillance, and ethics surrounding the beneficent use of sensitive education information[2] (Daries et al. 2014). With moocRP, the analytic can be brought to the data instead of the other way around. Institutions needing to keep data close to the chest can control its distribution but still open the window to education researchers by allowing analytic module collaborations and researchers access to a level of aggregate analytics returned by the module. Similarly, individuals possessing data do not need to upload it to a centralized location to have analyses run but can instead spin up their own local instance of moocRP and import their desired analytic modules to run locally. These issues of transparency, openness, modularity, and privacy were outlined in the open learning analytics vision document written by leaders of the Society for Learning Analytics Research (SoLAR) (Siemens et al. 2011).

In the past, researchers have suggested how modern visualization techniques could aid instructors in understanding how student activity corresponded to course success (Xu et al.

[1] https://github.com/CAHLR/moocRP.

[2] The Asilomar Convention for Learning Research in Higher Education (http://asilomar-highered.info/asilomar-convention-20140612.pdf).

2014). Other work categorized visualizations into the following groups consisting of (1) quantitative information (assignment grades, demographics, age), (2) qualitative information (discussions, course surveys), and (3) "real-time" visualizations while the course is running. The moocRP platform enables the visualization of all three concepts through its analytic module system. In the case studies in Sect. 3.1, we discuss two proof of concept analytics that provide interactive visualizations that are at the confluence of the categories described above. These analytics, uploaded to moocRP, serve as examples of the types of modules an instructor could utilize through moocRP. Problem areas left unsolved by previous work are (1) duplication of analytics work across platforms, institutions, and individual instructors and their TAs, (2) the lack of replication of data intensive research results due to logistical challenges in adopting published techniques, and (3) the divide between published studies and practical application. The objective of this work, which builds on Pardos and Kao (2015), is to ameliorate these issues through the implementation of open and modular paradigms with functional affordances that significantly reduce logistical friction.

1.1 On the Communities Catalyzing Education Analytics Research

The educational data mining (Koedinger et al. 2015) and learning analytics (Berland et al. 2014) communities have been a driving force behind academic and industry efforts towards collecting, curating, and operationalizing education data in service to the learner. As summarized by Ifenthaler and Widanapathirana (2014), EDM is primarily concerned with the translation of large sets of education data into information while LA focuses more on the learning process (not to be confused with cognitive process). Adding to this definition, EDM refreshes and improves on the algorithms and statistical techniques aiding in the fidelity of this translation from data to information and explores the added predictive utility gained from nascent sources of data. Learning analytics stays closer to the practice of the educational enterprise, often integrating EDM techniques into interventions affecting decisions made by the student, teacher, or administration. There is often an abundance of logistical friction at the operational interface of these different communities; moocRP is designed to lower the barrier of communication between EDM, LA, and stakeholders.

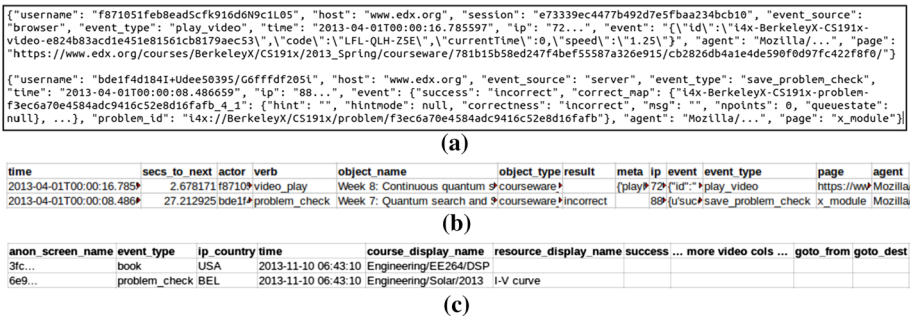
1.2 Related Work on Existing MOOC Data Models

The pedagogical developments in online education are still changing form and function and thus the model used to represent the data coming from these new pedagogical interfaces is likewise in flux. In this section, we provide an overview of a few of the most visible data models currently in play in the MOOC ecosystem. All of these models are compatible with moocRP, which takes a data model agnostic approach in its design.

1.2.1 edX Tracking Log and Database Model

The most primitive form of data that is obtained from edX by partner schools are the tracking and database logs[3] hosted by the Amazon S3 storage service. The raw edX data consists of a set of compressed and encrypted files that contain *event data* for all courses offered by an institution. Course data is distributed across multiple servers. As a result, to

[3] edX Research Guide. Data Delivered in Data Packages. http://edx.readthedocs.org/projects/devdata/en/latest/internal_data_formats/package.html, 2014.



{"username": "f871051feb8ead5cfk916d6N9c1L05", "host": "www.edx.org", "session": "e73339ec4477b492d7e5fbaa234bcb10", "event_source": "browser", "event_type": "play_video", "time": "2013-04-01T00:00:16.785597", "ip": "72...", "event": "{\"id\":\"i4x-BerkeleyX-CS191x-video-e824b83acd1e451e81561cb8179aec53\",\"code\":\"LFL-QLH-ZSE\",\"currentTime\":0,\"speed\":\"1.25\"}", "agent": "Mozilla/...", "page": "https://www.edx.org/courses/BerkeleyX/CS191x/2013_Spring/courseware/781b15b58ed247f4bef55587a326e915/cb2826db4a1e4de590f0d97fc422f8f0/"}

{"username": "bde1f4d184I+Udee50395/G6fffdf205i", "host": "www.edx.org", "event_source": "server", "event_type": "save_problem_check", "time": "2013-04-01T00:00:08.486659", "ip": "88...", "event": {"success": "incorrect", "correct_map": {"i4x-BerkeleyX-CS191x-problem-f3ec6a70e4584adc9416c52e8d16fafb_4_1": {"hint": "", "hintmode": null, "correctness": "incorrect", "msg": "", "npoints": 0, "queuestate": null}, ...}, "problem_id": "i4x://BerkeleyX/CS191x/problem/f3ec6a70e4584adc9416c52e8d16fafb"}, "agent": "Mozilla/...", "page": "x_module"}

(a)

time	secs_to_next	actor	verb	object_name	object_type	result	meta	ip	event	event_type	page	agent
2013-04-01T00:00:16.785	2.678171	f8710	video_play	Week 8: Continuous quantum s	courseware		{play	72	{"id":"	play_video	https://ww	Mozilla
2013-04-01T00:00:08.486	27.212925	bde1f	problem_check	Week 7: Quantum search and s	courseware	incorrect		88	{u'suc	save_problem_check	x_module	Mozilla

(b)

anon_screen_name	event_type	ip_country	time	course_display_name	resource_display_name	success	... more video cols ...	goto_from	goto_dest
3fc...	book	USA	2013-11-10 06:43:10	Engineering/EE264/DSP					
6e9...	problem_check	BEL	2013-11-10 06:43:10	Engineering/Solar/2013	I-V curve				

(c)

Fig. 1 Examples of event logs represented in different data models. **a** Example event logs in *raw edX event data* file. **b** Example event logs corresponding to the ones in **a** in *HarvardX event data* format. **c** Example event logs in VPOL format

obtain complete *event data* for one course, data must be first aggregated. While this is getting a bit into the weeds of data wrangling, it's a reality that can be overwhelming and keep an institution and its learners from reaping the benefits of analytics. The moocRP platform has built into it data cleaning and processing scripts that manage these processes from data ingestion, decryption, through to viewing the analytics on the web.

Figure 1a shows an example of event logs in an event data file. One *event data* file contains all event data for all courses within a one-day period in a JSON format. The course *database data* consists of a set of SQL and MongoDB database files which include information about students, demographics, and survey answers provided by students, as well as grades and certification statuses.

1.2.2 HarvardX Data Model

HarvardX Tool, developed by Jim Waldo[4], is the tool used by the HarvardX research team to package, analyze, and manipulate edX data. The tool converts *edX tracking log data* into more tangible csv files based loosely on the Experience API[5] (formerly Tin Can) by the Advance Distributed Learning Initiative (ADL). We call the output csv file, the *HarvardX event data*. Figure 1b shows the same event logs as in Fig. 1a in the *HarvardX event data* format.

1.2.3 Stanford VPOL Data Model

The Stanford Vice Provost Office for Online Learning (VPOL) provides MOOC data for researchers and instructors in a multi-file format[6]. It supports data derived from NovoEd, Coursera, and edX platforms. For edX data, the platform uses scripts developed by Andreas Paepcke[7] to transform raw edX event data and raw edX database data into relational tables. For example, the *EventXtract* table contains event data of all courses.

[4] Jim Waldo. HarvardX Tools. http://github.com/jimwaldo/HarvardX-Tools.

[5] ADL. http://github.com/adlnet/-xAPI-Spec/blob/master/xAPI.md.

[6] Stanford Vice Provost Office for Online Learning. How to Access the VPOL Online Learning Data. http://datastage.stanford.edu.

[7] Andreas Paepcke. json_to_relation. http://github.com/paepcke/json_to_relation.

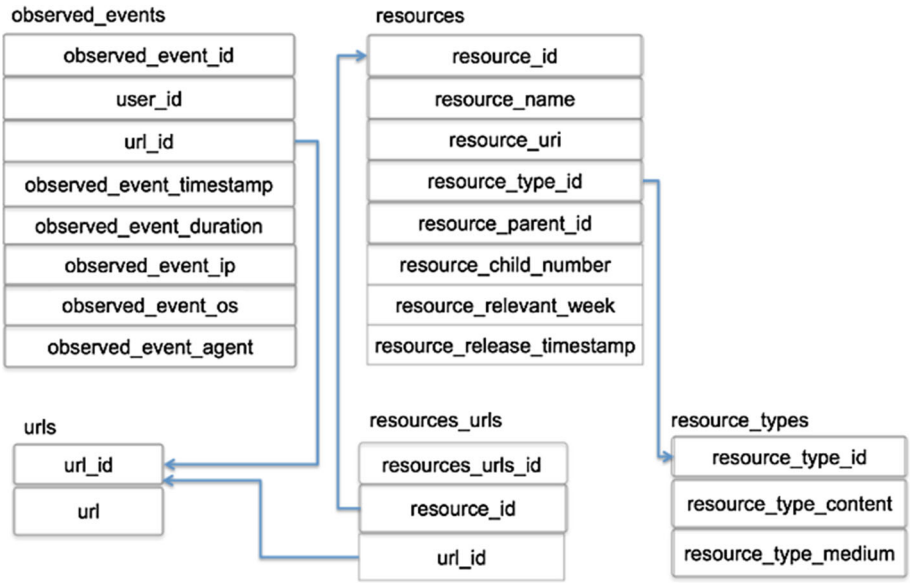


Fig. 2 MOOCdb's observing mode tables

Figure 1c shows an example of an *EventXtract* table. Observe that the two logs are from different courses. Other tables such as *AcitivityGrade, Demographics,* and *Forum post* are generated from edX database data.

1.2.4 MOOCdb

MOOCdb is a relational database schema for MOOCs[8] (Veeramachaneni et al. 2014). The schema organizes the data with respect to four different interaction modes: observing, submitting, collaborating, and feedback. Figure 2 shows the schema of observing mode tables.

1.2.5 LearnLab DataShop Model

In the space of data repositories, the Pittsburgh Science of Learning Center's DataShop (Koedinger et al. 2010) has hosted an abundance of both public and private datasets. While it is not currently a MOOC repository, it is widely used in the Intelligent Tutoring Systems community, and we would be remiss to not acknowledge its impact and use in the learning community. DataShop hosts datasets as a service to the community, largely from the Cognitive Tutor (Anderson et al. 1995), which was born out of the same lab, as well as from other tutoring systems such as the Andes Physics Tutor (Veeramachaneni et al. 2014), the Open Learning Initiative (OLI) (Lovett et al. 2008) for college statistics, a web-based mathematics platform, ASSISTments (Heffernan and Heffernan 2014), among others. These tutoring systems are based on mastery learning (Bloom 1968) and the effective practice of providing immediate feedback (Corbett and Anderson 2001) in achieving that

[8] MOOCdb. http://moocdb.csail.mit.edu/wiki/index.php?title=MOOCdb.

mastery. As such, these tutoring systems are problem-solving-centric with most of the tutors breaking problems down into a collection of steps to be answered with hints available as the primary pedagogical device. The singular data model underlying the DataShop was therefore designed to be problem-centric, with each row in the data format representing an answer to a step. In MOOCs, where problem solving takes a back seat to lecture videos and student time on task (Breslow et al. 2013), capturing a wider variety of behavior and interaction is desirable. The DataShop data model was not designed to incorporate this variety of information nor social network information from the many social contexts in which learning takes place (Ferguson and Shum 2012). This is not to say that the DataShop model falls short of serving its purpose, or that a globally compatible data model should be sought, but rather that a single data model is not sufficient to accommodate the numerous philosophies of learning represented.

1.3 Related Work on Emerging Data Models Supporting Interoperability

Two learning data specifications are of particular interest to moocRP users interested in analytics that span multiple data sources: ADL's *Experience API* (xAPI), mentioned briefly in sect. 1.2.2, and the IMS Global Learning Consortium's *Caliper Analytics*™. Both specifications are a response to the need for data interoperability in an increasingly federated digital ecosystem comprising diverse learning services that are expected to leverage data analytics in order to drive pedagogical innovation. For ADL changes in the technology landscape occasioned by the shift to cloud computing, mobile platforms and social learning "have stimulated a new market for ubiquitous learning" and a desire for "flexible tracking of learning activities and experiences".[9] In the case of IMS the advent of "highly scaled online curriculum delivery" coupled with a heightened demand for "accountability backed by measurability" requires creation of a "standardized measurement framework".[10] For both communities, early work has focused on designing an extensible information model, developing a RESTful architecture for mediating machine-to-machine data exchange and grappling with the challenge of defining controlled vocabularies that leverage semantic web principles and practices.

ADL's xAPI is a "learning-technology" specification and set of open-source RESTful APIs for describing, recording and transmitting securely "statements of experience" to a target learning record store (LRS) for subsequent analysis.[11] Introduced in 2012, an early proponent described xAPI as "the next generation of SCORM that allows e-learning to use modern technologies in an interoperable way… it lets us do all of the fun stuff we want to include in our learning programs but just can't because SCORM gets in the way".[12] Although ADL's SCORM specification facilitates development of reusable courseware and learner tracking, it assumes a browser-based, course-based, LMS-based mode of delivery. In contrast, xAPI is both content and system agnostic and champions the tracking

[9] Advanced Distributed Learning Initiative (ADL), U.S. Department of Defense. http://adlnet.gov/adl-research/performance-tracking-analysis/experience-api/xapi-background-history/.

[10] IMS Global Learning Consortium (IMS). "Learning Measurement for Analytics Whitepaper (2013). http://www.imsglobal.org/sites/default/files/caliper/IMSLearningAnalyticsWP.pdf.

[11] ADL. http://adlnet.gov/adl-research/performance-tracking-analysis/experience-api/xapi-technical-specifications/. See also http://github.com/adlnet/xAPI-Spec/blob/master/xAPI.md#roleofxapi.

[12] Andy Whitaker, "An Introduction to the Tin Can API", The Training Business (19 July 2012). http://www.thetrainingbusiness.com/softwaretools/tin-can-api/.

of learning activities irrespective of the context—formal, informal, social and experiential are all grist for the mill.[13]

IMS's Caliper Analytics™ specification was released in October 2015.[14] Like xAPI it provides an extensible information model and an API for instrumenting learning applications and services that log learning events. Unlike xAPI, Caliper's information model is based on a discrete set of learning activity profiles that make use of domain-specific controlled vocabularies. Activities described in the initial release of the specification include login/logout, page navigation, resource viewing and annotation, rich media interactions, assessment, and outcome scoring.

The xAPI spec models a learning experience in the guise of a "statement", a data construct consisting minimally of an actor, verb and object triple. Optionally, a statement may include details about the learning context (including custom extensions), the date and time when the activity occurred, a result or measured outcome, an array of attachments, as well as an assertion by a recognized authority regarding the veracity of the statement.[15] Statements are stored as JSON and all verbs, activity types and extension keys must be identified uniquely using an Internationalized Resource Identifier (IRI).

Likewise, the base Caliper event consists of an actor, action, object triple and a timestamp. Similar to xAPI the Caliper event model promotes use of richer data structures designed to capture and transmit information about the learning context within which an activity is situated. That said, a key differentiator between the two specifications involves their respective approach to controlled vocabularies and adoption of semantic web principles. The xAPI specification makes no attempt to prescribe use of controlled vocabularies. Choice of verbs, activity types, attachments and extensions are left to xAPI implementers to define themselves. Indeed, responsibility for identifying use cases and developing domain-specific controlled vocabularies is ceded to user groups recognized as xAPI "Communities of Practice" (CoP), nine of which have been chartered to date.[16] The recently formed Badges CoP has made a start at defining a controlled vocabulary for use with Mozilla's Open Badges.[17] So too have the Course, Social and Video CoPs.[18] ADL itself maintains its own controlled vocabulary of xAPI verbs and activity types. Each verb and activity type definition in the ADL vocabulary includes a reference to a matching entry in Princeton University's *WordNet* database as an example of linking to an authoritative lexical source.[19]

Caliper, in contrast, has embraced linked data principles in its adoption of JSON-LD and its profiles aspire to provide both structure and shared meaning when data is exchanged between different agents (both human and machine). Caliper's annotation profile, for example, models seventeen actions: attached, bookmarked, classified,

[13] ADL. http://adlnet.gov/adl-research/performance-tracking-analysis/experience-api/.

[14] IMS. http://www.imsglobal.org/article/ims-global-learning-consortium-announces-products-certified-newly-released-caliper.

[15] ADL. http://github.com/adlnet/xAPI-Spec/blob/master/xAPI.md#stmtprops.

[16] See http://adlnet.gov/adl-research/performance-tracking-analysis/experience-api/xapi-community-of-practice-cop/.

[17] xAPI Badges CoP. http://github.com/ht2/BadgesCoP/blob/master/earning/vocab.md.

[18] xAPI Course CoP. http://github.com/adlnet/xAPI-SCORM-Profile/blob/master/xapi-scorm-profile.md; xAPI Social CoP. http://docs.google.com/document/d/1RpFxEh0KdO6WGgK74LUctP5oM35nsWHk0Czk__syH1Q/edit; xAPI Video CoP. http://docs.google.com/spreadsheets/d/1jq2zrvv2LKsE6-vbSBCc6H-PCyn40dQA4P96bl3s6BI/edit-gid=0.

[19] ADL. http://w3id.org/xapi/adl/; WordNet, Princeton University. http://wordnet-rdf.princeton.edu/.

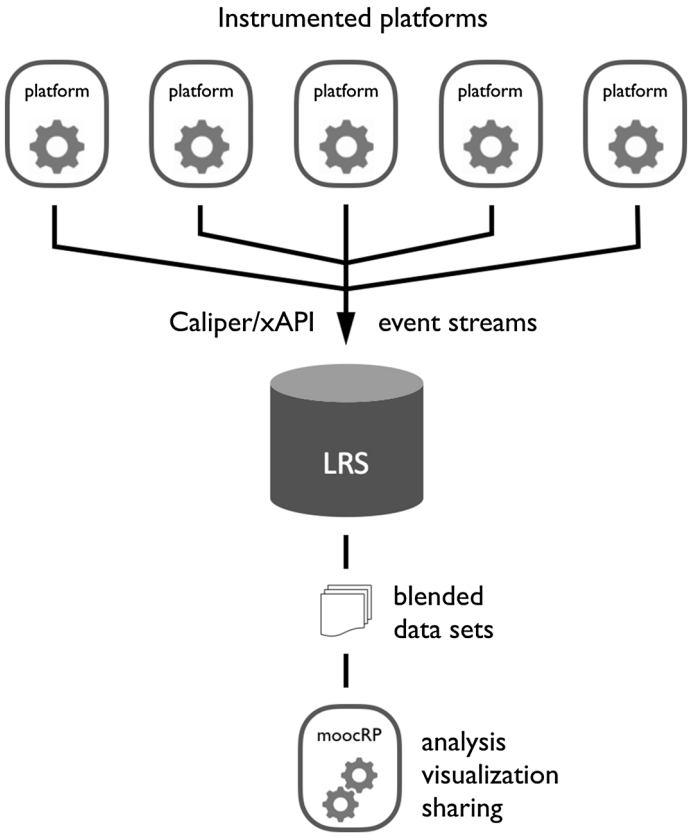


Fig. 3 Standardized data streams consumed by moocRP

commented, described, disliked, highlighted, identified, liked, linked, ranked, questioned, recommended, replied, shared, subscribed, and tagged. Annotation events are described using an expanded event model that includes the application context, group/organizational context, an actor's membership context (e.g. role, status), as well as objects that are the target of the activity and/or generated as a result of the interaction as in the case of a highlight or bookmark created as a result of a user annotating a digital resource. That said, much work remains to be done before either specification is in a position to support both data *and* semantic interoperability. The initial set of Caliper profiles is modest in scope and common learning activities such as content authoring and threaded discussions have yet to be modeled. Caliper actions are as yet poorly described and their IRIs are currently not resolvable, rendering semantic dereferencing currently impossible.[20] As for xAPI, the reliance on community-generated vocabularies *sans* guidance is not without its detractors. The ADL technical team issued a draft white paper in February 2015 lamenting the lack of progress in producing semantically consistent and reusable xAPI controlled vocabularies. "[T]he xAPI community is living in the 'Wild West' where vocabularies are everywhere, yet nowhere" it charged. The white paper called for the establishment of a dedicated group responsible for overseeing the

[20] A number of these issues are expected to be resolved in the upcoming Caliper 1.1 release.

```
Note:  properties with empty/null values excluded.
{
    "actor": {
        "mbox": "mailto:554433@example.edu",
        "name": "554433",
        "objectType": "Agent"
    },
    "verb": {
        "id": "http://adlnet.gov/expapi/verbs/resumed",
        "display": {
            "en-US": "resumed"
        }
    },
    "object": {
        "id": "https://youtu.be/hFz6uHztnMA?t=94",
        "definition": {
            "name": {
                "en-US": “Lec 2.1. Describing one quantitative variable"
            },
            "description": {
                "en-US": "Video"
            },
            "type": "http://adlnet.gov/expapi/activities/media"
        },
        "objectType": "Activity"
    },
    "context": {
        "platform": "https://www.edx.org/",
        "extensions": [
            {
                "id": "i4x-BerkeleyX-Stat2_1x-video-
8275cb3eb13e434d831bea15dbe3dfa1",
                "playback_speed": "1.0",
                "playback_position_secs": 94,
                "youtube_id": "hFz6uHztnMA"
            }
        ]
    },
    "contextActivities": {
        "parent": {
            "id":
"https://www.edx.org/courses/BerkeleyX/Stat2.1x/2013_Spring"
        }
    },
    "timestamp": "2013-02-21T13:09:58.122346Z"
    }
```


Fig. 4 xAPI JSON representation of an edX video resumed event

development of controlled vocabularies, leveraging linked data principles in order to “improve the quality and semantic interoperability of xAPI data”.[21] In response, an

[21] ADL Technical Team, http://docs.google.com/document/d/1zBPKryuF1tXHTI-AYjXd0ctdWoq4o4P-Uq9SAhJfus0/edit?pli=1#.

```
Note:  properties with empty/null values excluded.
{
    "sensor": "https://www.edx.org/caliperSensor/001",
    "sendTime": "2013-02-21T13:09:58.122349Z",
    "data": [
        {
            "@context": "http://purl.imsglobal.org/ctx/caliper/v1/Context",
            "@type": "http://purl.imsglobal.org/caliper/v1/MediaEvent",
            "actor": {
                "@id": "https://example.edu/user/554433",
                "@type": "http://purl.imsglobal.org/caliper/v1/lis/Person"
            },
            "action": "http://purl.imsglobal.org/vocab/caliper/v1/action#Resumed",
            "object": {
                "@id": "https://youtu.be/hFz6uHztnMA?t=94",
                "@type": "http://purl.imsglobal.org/caliper/v1/MediaLocation",
                "name": "Lec 2.1. Describing one quantitative variable",
                "isPartOf": {
                    "@id": "https://youtu.be/hFz6uHztnMA",
                    "@type": "http://purl.imsglobal.org/caliper/v1/VideoObject",
                    "name": "Week 1: Section 2a: The Histogram (Lec 2.1 - 2.3)"
                },
                "currentTime": 94,
                "extensions": [
                    {
                        "id": "i4x-BerkeleyX-Stat2_1x-video-
8275cb3eb13e434d831bea15dbe3dfa1",
                        "playback_speed": "1.0",
                        "playback_position_secs": 94,
                        "youtube_id": "hFz6uHztnMA"
                    }
                ]
            },
            "eventTime": "2013-02-21T13:09:58.122346Z",
            "edApp": {
                "@id": "https://www.edx.org/",
                "@type": "http://purl.imsglobal.org/caliper/v1/SoftwareApplication"
            },
            "group": {
                "@id":
"https://www.edx.org/courses/BerkeleyX/Stat2.1x/2013_Spring",
                "@type": "http://purl.imsglobal.org/caliper/v1/lis/CourseOffering",
                "courseNumber": "Stat2.1x",
                "academicSession": "Spring 2013"
            }

            }
        ]
    }
```


Fig. 5 Caliper JSON-LD representation of an edX video resumed event

xAPI Vocabulary and Semantic Interoperability Working Group was formed as a W3C Community Group in July 2015.[22] The group has since produced an *xAPI Controlled Vocabulary Ontology* "intended to complement the core xAPI specification". The ontology defines a schema and provides a list of terms drawn in part from existing ontologies such as the Simple Knowledge Organization System (Miles and Bechhofer 2009) and the PROV Ontology (Lebo et al. 2013) that is recommended for use by communities of practice when creating xAPI controlled vocabularies.[23]

Despite the challenges outlined above both xAPI and Caliper are gaining traction in the marketplace. Several vendors now offer learning record stores designed from the outset to store xAPI statements. The Caliper release announcement noted that a number of IMS member organizations had already certified their learning platforms as Caliper-compliant with more on the way while the Unizin Consortium has pledged to speed Caliper adoption among its 22 member institutions.[24] Other more ambitious initiatives are underway to build learning analytics infrastructures, both proprietary and open source, that are primed to both emit and consume xAPI and/or Caliper event data streams.[25] The Caliper team plans to equip its sensors with the ability to represent event data in either format, offering the tantalizing prospect of eventual convergence between the two specifications. We expect that in time MOOC providers such as edX will instrument their platforms and generate learning data in conformance with one or both of the specifications. Future research activities will involve leveraging the xAPI and Caliper specifications to produce extended datasets for moocRP to consume that blend data from MOOC providers, learning managements systems, and back-end student information systems (Fig. 3). In the meantime, we have begun working with both xAPI and Caliper, transforming edX event log data into xAPI statements and Caliper events for use with moocRP (see Figs. 4 and 5).

2 Functionality and Implementation of moocRP (Methods)

In this section we discuss the various practical functionality of the tool, such as the data request/authorization workflow, security details, technical implementation details, and multiple data model compatibility.

2.1 Instructor-Oriented Interface

A prominent design principle kept in mind during the design of this application is the creation of an instructor-oriented dashboard. Instructors have much to gain from inspecting various aspects of their courses through the research analyses and visualizations of other educational researchers, so the targeted use case is that of instructors running MOOCs extracting actionable analytics. To achieve this, we concentrated on three major areas: ease

[22] xAPI Vocabulary & Semantic Interoperability Group. http://www.w3.org/community/xapivocabulary/. See also http://github.com/adlnet/xapi-vocabulary.

[23] ADL. http://xapi.vocab.pub/ontology/index.html.

[24] Unizin Consortium. http://unizin.org/2015/11/unizin-consortium-partners-with-ims-global-learning-consortium-to-drive-caliper-analytics-adoption/.

[25] Unizin, the Apereo Foundation and the UK's JISC have all outlined plans to build standards-based learning analytics infrastructures in partnership with commercial vendors. See http://unizin.org/, http://www.apereo.org/communities/learning-analytics-initiative, http://analytics.jiscinvolve.org/wp/2015/06/15/jiscs-learning-analytics-architecture-whos-involved-what-are-the-products-and-when-will-it-be-available/.

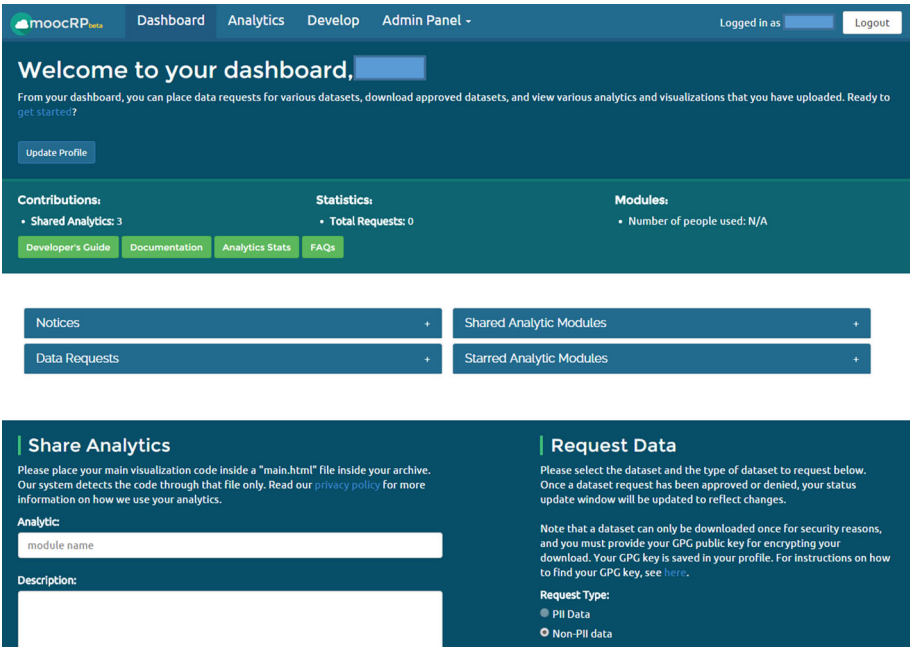


Fig. 6 The main interface for moocRP, providing easy access to data requests, analytic modules, and basic stats about various features

of access through an aggregated and simplistic interface, facilitation of data distribution, and security (Fig. 6).

2.1.1 Aggregated Interface

MoocRP provides a simple, aggregated interface to all of its basic functions. On its main page, moocRP has a series of four closed but expandable panels. The four panels provide easy access to (1) data requests, (2) data downloads, (3) shared analytics modules, and (4) starred analytics modules. Below, are two forms for requesting data and uploading analytic modules. The system for data requests and downloads is discussed further in the next section, so here, we focus on the interface for analytics. To facilitate an instructor's access to interesting modules, moocRP allows users to "star" analytic modules and visualizations, essentially creating a bookmark for quick access to the module (Fig. 7).

The analytics page provides a paginated list of approved visualizations and analytic modules uploaded by various users of the system. Each row representing a module also has a dropdown list of datasets for the module to be applied on. The datasets listed are (1) restricted to the datasets that have already been granted to the browsing user, and (2) limited to the compatible data models as specified by the uploading analytic author. For example, modules with a particular Coursera data model are most likely incompatible with an analytic module developed for a particular edX data model. A search feature is implemented to allow instructors to quickly scout out relevant modules to their problems.

A set of links is also provided for quick access to various significant moocRP developer guides, such as the documentation and data model guides for developing analytic modules.

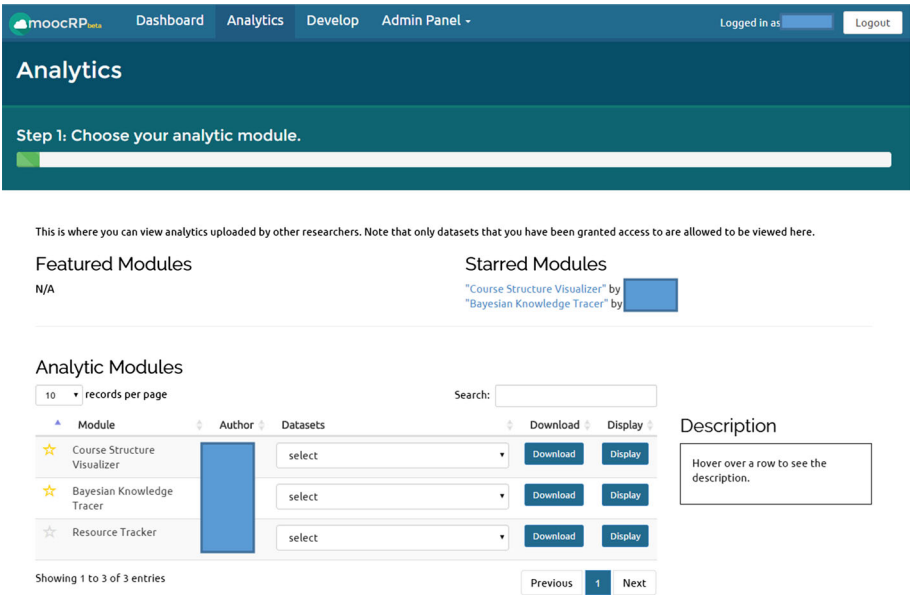


Fig. 7 A two column header provides access to popular analytics modules as well as the visualizations that the instructor has starred. Below is the list of available analytics modules, each with drop-down lists of compatible datasets

2.1.2 Data Distribution

A basic but essential feature of moocRP is its facilitation of data distribution. The processes to request and receive data at numerous institutions often involve a large number of email handoffs culminating in an exchange of a thumb drive with encrypted data in tow. This is just one example of a significantly inefficient process that could become more streamlined and secure. To solve this problem, moocRP provides a shorter data distribution pipeline. Upon registration, a user provides a GPG public keys (step-by-step instructions on how to do so are provided on the same page). The user can then login and view the "Request Data" form, where they simply select their choice of dataset and submit the form to initiate the data request. The data scrubbing scripts packaged with moocRP can produce personally identifiable information (PII) and non-PII versions of a dataset with separate authorization for each (Fig. 8).

After a data request is sent, any moocRP administrator can go to the administration panel and view a searchable table of data requests, which displays information about each data request (i.e. user's full name, type of dataset, dataset name, date requested). The administrator can either grant or deny the request, which would then reflect on the researcher's dashboard. The users can then easily download their approved datasets with a one-time download provided by the server. A more technical discussion of the implementation behind this pipeline follows in a later section of the paper (Fig. 9).

2.1.3 Security

With a web application that works with sensitive datasets containing personal information, security is a major concern. This concern is reflected in the development of moocRP where security comes first at every level (Fig. 10).

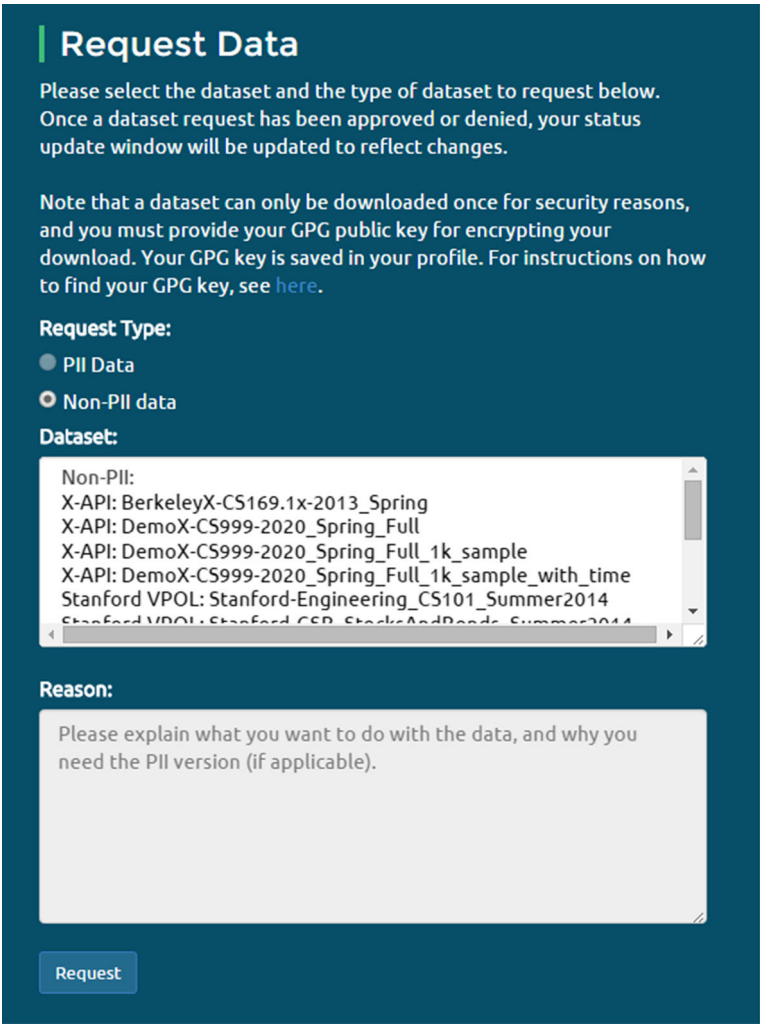


Fig. 8 A simple form to request various datasets

The first set of security measures comes directly from the user authentication system built into moocRP. The current release of moocRP uses the CAS protocol, otherwise known as the Central Authentication Service. Using CAS has two benefits: a secure way to manage users using a university's existing CAS setup and the ability to use one set of credentials to login to multiple services. Focusing on the security aspect of CAS, we can inspect the CAS authentication process: a user logs into moocRP, which redirects to the institution's CAS portal, where the user enters login credentials. The moocRP back-end server waits for a response from the CAS portal that has a "ticket"—moocRP then sends this ticket back to the CAS portal for validation (to prevent access through an expired session or fabricated ticket). If validation is successful, then the user is allowed access to moocRP. Future modularization of the authentication module will allow for substitutions of alternative authentication protocols into moocRP, based on each institution's needs.

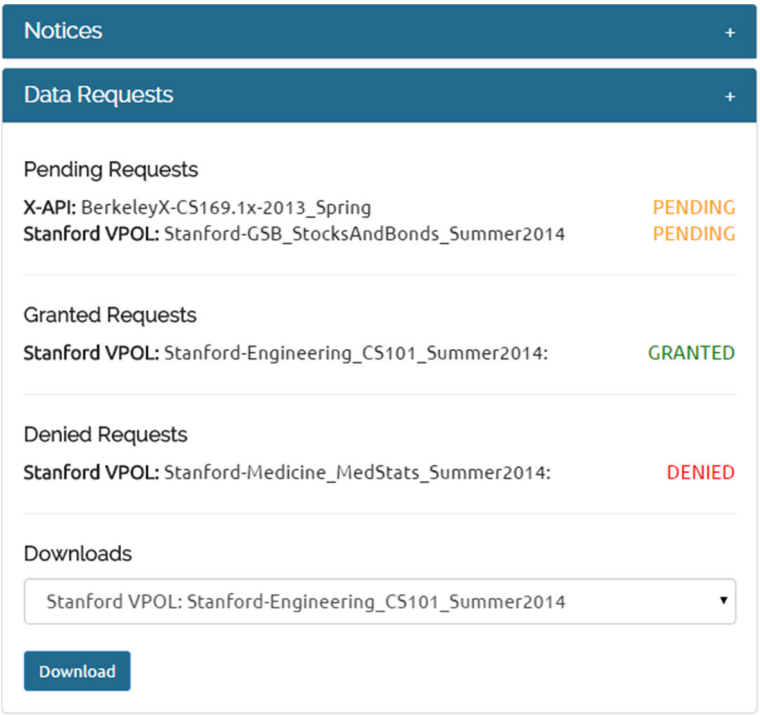


Fig. 9 A slide-out panel tracking a user's data requests for MOOC data distribution, showing granted, pending, and rejected requests as well as a dropdown for approved datasets available for download

Fig. 10 CAS authentication module is built into the platform

The next level of security targets the datasets moocRP distributes. Upon registration, the user inputs his/her PGP public key, allowing moocRP to utilize the power of encryption to secure datasets. moocRP automates dataset encryption when a dataset is approved for download for a particular user, encrypting the dataset with 2048-bit PGP encryption using the user's public key, allowing only the original owner of the private key to decrypt the dataset. Thus, even if a malicious attacker somehow intercepted the dataset and downloaded it, it would be near impossible for that person to decrypt and extract the actual data inside.

The moocRP web interface runs over the HTTPS protocol, with SSL built into the platform. This secures the connection between the client and the moocRP server, preventing spoofers and eavesdroppers from gaining unauthorized access to communications. The data distribution is finally secured with CSRF protection and a one-time download link, preventing someone with malicious intent from attempting to resubmit a data request form or re-download an approved dataset.

Some additional minor security features are found in the administration panel and logging. The administration panel provides features to manage users, including removal and editing of users, if needed. Uploaded analytics modules can also be removed if found to include malicious code. Logging keeps track of user actions within moocRP, which can be parsed to check for unusual activities.

The last major security measure is inherent to moocRP itself: moocRP in its present state is a relatively closed system, meaning that each moocRP instance will usually be used and managed by individual institutions, so only members or affiliates of the institution will be able to access moocRP. A *required level of trust* is necessary to re-inforce all of the security precautions that moocRP implements: users are only allowed into the system by each institution's moocRP administrators. Likewise, data requests and analytic modules must all go through an approval process overseen by the administrators. Datasets for analytic modules are only available to a user to view if the user has been granted access to the dataset by an administrator.

Despite the security measures put into place during the development of moocRP, no system is foolproof and flawless, and there remains work to be done for moocRP as security threats evolve and expand, but moocRP's multi-tiered security model provides a solid foundation against the majority of threats and is a promising start for a relatively tight community within each institution.

2.2 Technical Design

2.2.1 Technology Stack

The choice of technologies was selected with care. During the planning phase of moocRP development, there were two obvious choices of technologies: Ruby on Rails or Node.js with a web MVC framework. Note that Node.js should not be directly compared to Rails, since Node.js is a web server while Rails is a web application framework.

Rails is a popular framework but is facing competition from Node.js, a Javascript runtime environment for developing server-side web applications. A few notable differences and considerations when using Node.js with an MVC framework versus using Rails:

- Rails is built on Ruby. Node.js is built on JavaScript, which (often times) has comparable or faster benchmark times than Ruby.

- Rails is an opinionated framework, forcing a developer to adhere to its culture and stigmas. Node.js is the opposite, more allowing of a step-by-step build of various components.
- Rails has a steep learning curve, with numerous Ruby intricacies to consider. Learning JavaScript is rather quick with experience in any major language like Java or C.

With these observations in mind, we can weigh some advantages and disadvantages of each. For the purposes of moocRP, we heavily favored speed and customization, so we chose Node.js. To make up for what Rails provides out of the box, we used a popular Node.js MVC framework: Sails.js. This allowed us to quickly develop a web application with basic functionality in a Rails manner while still maintaining the ability to customize the application as we desired. Since our work began, moocRP has developed greater support for web sockets, better support of databases, and will soon extend integration of AngularJS, which will allow for a smoother presentation and transfer of data to the client.

2.2.2 Data Distribution Pipeline

The implementation of the data distribution pipeline is relatively simple. moocRP keeps a file watch on the directory `datasets/available`, which holds sub-directories for each data model and its available datasets. These datasets should be stored as archive.ZIP files, where all the files for a particular dataset are compressed into a single archive file. When new sub-directories and archives are created or added in, moocRP detects the change and updates the interface with new datasets that are available to request.

When an administrator grants a request, moocRP looks for the corresponding sub-directory and archive file, looks up the requesting user's public key, and encrypts the archive with 2048-bit encryption. The archive is then moved to a new folder, `datasets/encrypted`, to be available for download by the requesting user. Upon downloading and leaving the page, moocRP will expire the download link and prevent additional downloads.

Nearly all of the technical details behind providing datasets to users are abstracted away with a "drag-and-drop" approach, where new datasets can simply be archived and dropped into new folders in the directory being watched by moocRP. The system allows for as simple an approach as dropping new archives in or as complex an approach as adding cron jobs to automate the addition of new datasets on a schedule.

2.2.3 Analytics Pipeline

The analytics pipeline for sharing and visualizing modules created by users is central to moocRP. There are a few notable components of the analytics pipeline to discuss: the file watch, the syncing of datasets to clients, and the uploaded analytic modules.

Essentially, moocRP keeps a file watch on specific folders in the base directory of moocRP for datasets. These folders are associated with the datasets of each data model in moocRP. The folders are created when an administrator creates a new data model in the moocRP admin panel. For example, a sample structure of the data directories used for analytics modules:

```
- datasets/extracted
    - harvardx
        - berkeleyx_stat21
            - berkeleyx_stat21_out.csv
    - raw_edx
    - ...
```

When a new folder is detected inside these directories, moocRP will add new data models and datasets to be made available for requesting by users. When a user's request is granted, the user will also be allowed to select the dataset for relevant analytic modules.

The other component of the analytics pipeline involves the uploaded modules. As mentioned in the case studies, the analytic module is a package consisting of one HTML file, CSS files, and JS files. moocRP extracts these files into a temp folder and moves them into the correct directories to be served. The CSS and JS files are moved into the assets folder, while the HTML file is moved to a view folder used by the framework Sails.js to serve pages with additional features, e.g. template variables between the client and server. Once placed in the correct folders, moocRP will display them on the analytics page.

2.3 Support for Multiple Data Models

The moocRP tool allows an administrator to add a new data model easily. The process of adding a new data model consists of only a few steps. The administrator can navigate to the "Data Models" tab in the administration panel. The administrator then fills out a form that asks for the user-friendly name of the new data model and a system-friendly name for the data model. The system-friendly name is a name that moocRP can use to automatically create the appropriate directories on the server. Then, the administrator only needs to configure the tools necessary for transforming the data into the new data model formats in a way such that the tools output the datasets in the directories that moocRP created. Therefore, once a new data model is added in the administration panel, moocRP will be able to recognize the new data model and provide support for it for analytics modules and for data distribution (Fig. 11).

2.3.1 edX Data Scrubbing

On the administrator's side, moocRP conveniently provides an edX data scrubbing tool out of the box that helps the administrator

- download edX data from the Amazon servers
- decrypt the downloaded data
- organize event data logs across multiple servers into one event data log file for each course, and
- transform the aggregated event data log files into *HarvardX event data* format.

Our edX data scrubbing tool is built on top of the HarvardX Tools (see in Sect. 1.2.2). We modified several parts of the original scripts to make the tool compatible with the current format of raw edX data, and to speed up the data organizing and transforming time.

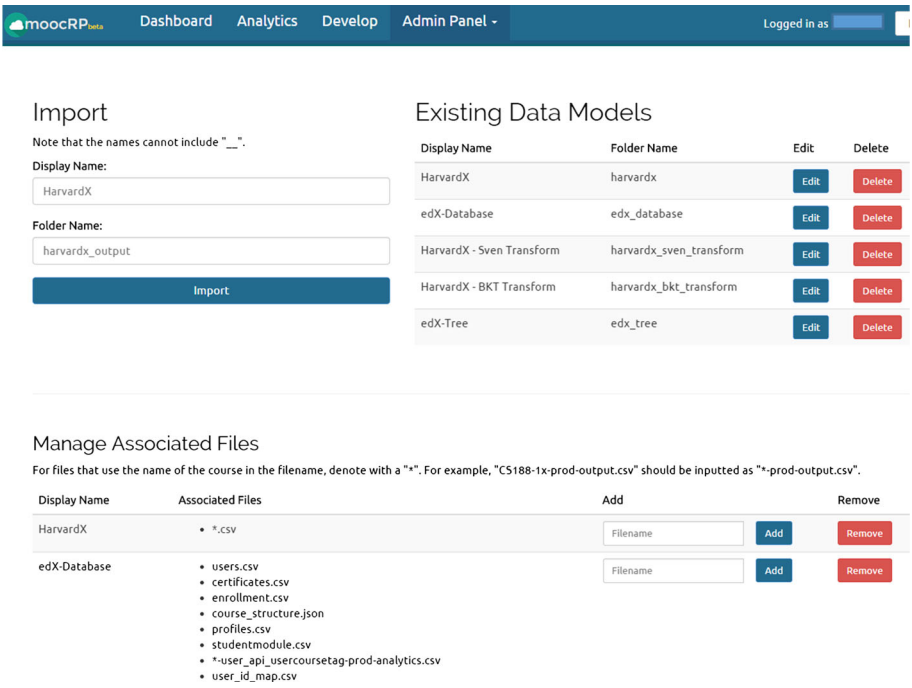


Fig. 11 The data models management page, where data models can be added or deleted in a few clicks. Associated files of each data model is also managed here

3 Learning Analytics Module Case Study (Results)

In this section, we will describe some proof of concept implementations of analytics modules developed as a result of the moocRP system.

Integration of analytics modules into moocRP is relatively straightforward and requires minor modifications in which the analytics source code is structured. The basic structure of an imported analytics package follows this package structure:

- `main.html`
- `/css`
- `/js`

The main display of the analytics module is placed in `main.html`, where the HTML will be rendered in a container on the analytics page of moocRP. Any script dependencies and styling dependencies must be declared in this file as well—this allows moocRP to have a single location to rewrite dependencies in a more compatible fashion. CSS files and JavaScript files are placed in the `/css` and `/js` folders, respectively.

Another minor change that needs to be made is how the data is read into the analytics module. The analytics module should access the dataset using "`<%= dataset %>`", which represents a dictionary that holds the various files associated with a data model.

To access each data model's files in an HTML file, we can use JavaScript and write the following code:

```
var dataFiles = {}
<% for (var dataFile in dataset) { %>
  <% if (dataset.hasOwnProperty(dataFile)) { %>
    dataFiles["<%= dataFile %>"] = <%- JSON.stringify(dataset[dataFile]) %>;
  <% } %>
<% } %>
```

Then, to access any of the data model's file contents, simply select the file using the name of the data model file as the key to the dictionary. Raw content will be stored in the dictionaries for parsing and manipulation by the analytics modules.

3.1 Bayesian Network Knowledge Analysis

In this first case, the Bayesian Knowledge Tracing algorithm is employed to assess student current and prior knowledge for each given problem in the course. The visualization is presented along with age and level of education information so that an instructor may inspect for which demographic students lacked the requisite prior knowledge. Additional attributes can be added to this visualization, including country, survey information, and outcomes for previous parts of the course (Fig. 12).

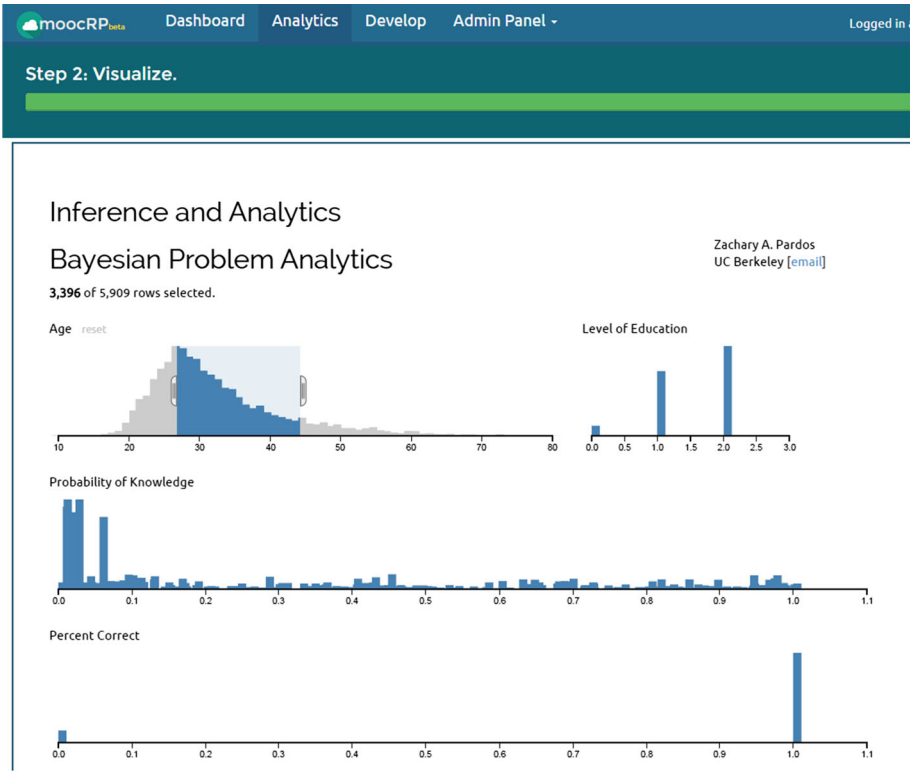


Fig. 12 This analytic module is based on Bayesian Knowledge Tracing

To integrate this module into moocRP, we simply had to move all JavaScript and CSS dependencies that were not in the main HTML file into their respective folders as defined earlier. In the original HTML file, `d3.csv(''filepath'', function)` was used to read the CSV contents of a file to perform the D3 visualizations. With moocRP, instead of directly reading the file from disk using a function like `d3.csv`, we can receive the CSV file contents from the data model by including the code previously mentioned with the addition of the snippet `var data = dataFiles[''output.csv'']`. Then, we can use the function `d3.csv.parse` on the `data` variable, which contains the raw contents of the file. With the modifications complete, we can then zip up the `main.html` file along with the `css/` and `js/` folders and upload the archive to moocRP to be shared and applied on other datasets.

Some short, sample code is shown below for a simple overview of how the data can be used with D3:

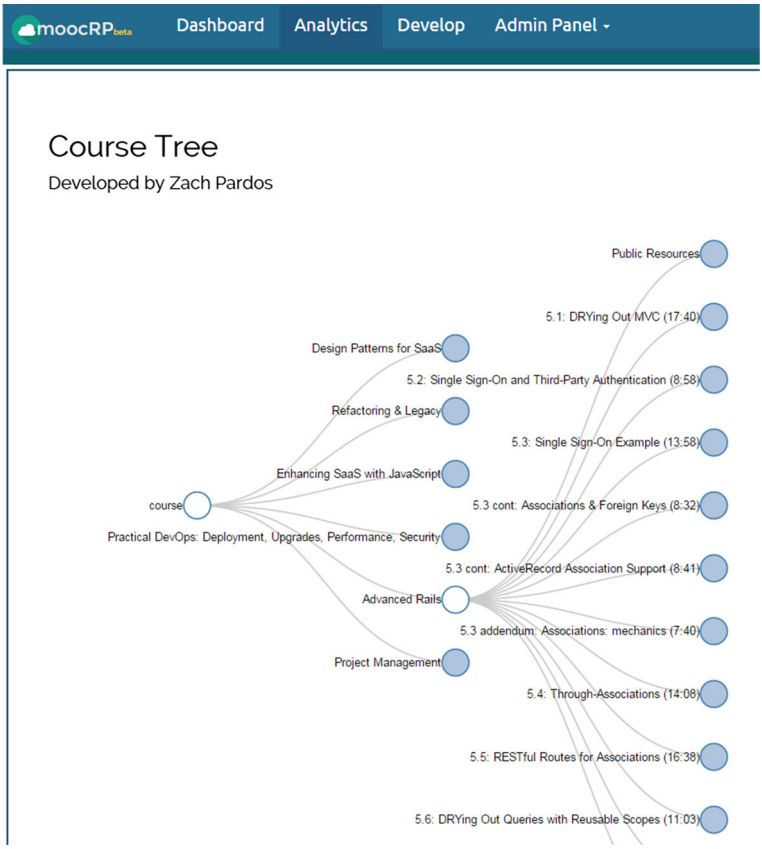


Fig. 13 An analytic module used to visualize the course components of CS169 Software Engineering from BerkeleyX

```
var raw_data = dataFiles["bnt.csv"]
                .replace(/\\n/g, "\n");
var flights = d3.csv.parse(raw_data);
flights.forEach(function(d, i) {
    // do something to data rows
}
var flight = crossfilter(flights), ...
var chart = d3.selectAll(".chart")
    .data(charts)
    .each(function(chart) { ... });
```

3.2 Course Structure Visualizer

In the second case study, we present a module for visually inspecting the flow and course structure of a course. The tree is created from parsing of the `course_structure.json` file and visualized using an expandable tree layout in D3. While this visualization shows only an interactive course tree structure, it can be used as the basis for navigating to analytics about different elements of the course, such as selecting the problem to receive the aforementioned Bayesian problem analytics (Fig. 13).

Integration of this module was similar to the first case study. This particular analytic module used the `course_structure.json` from the edX database model. moocRP can handle JSON files in an even more efficient manner, reading and passing the contents of the file as a JSON object to the HTML page instead of raw string contents.

For this module then, we took out the original `d3.json(''filename'', function)` and simply used `var data = dataFiles[''course_structure.json'']` to access the data, without any parsing necessary.

4 Tool Sustainability and Future Work

Centralized repositories, such as the LearnLab's DataShop, are not supported by an institution and rely on government funding to keep its services running. Instead of relying on a funding-dependent model, the moocRP model utilizes institutional self-interest to maintain its services. Institutions have an interest in facilitating research from its faculty and staff as well as leveraging value from its data in order to provide a more effective and efficient educational experience to its students. No particular platform buy-in or inter-institutional network is required; rather each institution may utilize the tool and benefit from the generalizable analytics created for it.

Much work remains to be done to improve moocRP to its full potential. Some important features to be implemented with community support and continued engineering work include:

- Automated analytic module security screening
- Alternative authentication protocols and modularization
- Integration of data pre-processing scripts (see Fig. 14)
- Support for scripts written in alternative languages, e.g. Matlab, Perl, Python, that could be used for machine learning analytics
- Additional statistics on data model and analytic usage information
- Various server and implementation optimizations

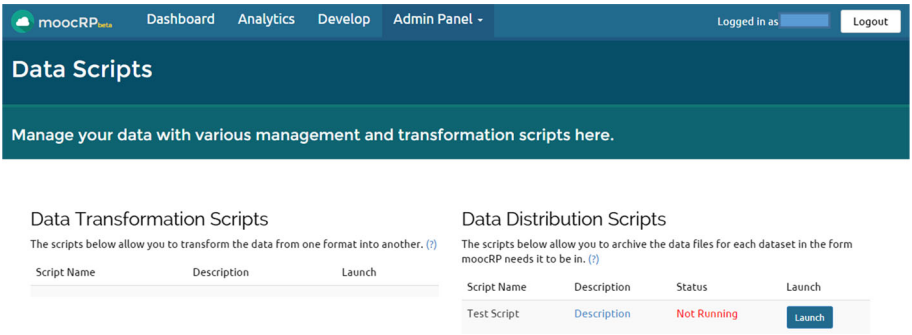


Fig. 14 In-progress work on integration data transformation and pre-processing scripts

We will also continue collaborations with interested universities and learning systems through webinars, etc. to gain feedback on the tool. The future of moocRP will only flourish as much as its community involves itself in the platform, whether through contribution of analytic modules, adoption of the moocRP system, or direct development on the moocRP codebase.

5 Conclusions

In this paper, we described the value of an open source analytics platform in the context of modern research on MOOC datasets. Open learning analytics requires a streamlined approach to distributing research findings for other researchers to apply and adapt to their datasets. Emergent learning data specifications like xAPI and Caliper offer a standardized approach to modeling learning activities that simplify the work of blending data sets from diverse sources. moocRP approaches this challenge with a solution that facilitates data analysis, visualization, distribution, and research analytics module reuse. The analytics modules, such as dashboards, are simply not as useful unless there is a distribution vehicle that can bring the modules beyond their creation point; analytics based purely on boutique individual projects are not beneficial to the learning analytics community.

Overall, moocRP is not a panacea to the problem of open analytics or reproducible research, but it is a major step in the right direction. Tools like moocRP are needed to realize the impact of each researcher's work in the learning analytics community and foster collaborations; these tools can help researchers build upon each other's work, and facilitating this should prove a productive and valuable area for future work. By developing these tools and open sourcing them to the community, we hope to encourage more contributions to this project leading to accelerated adoption of learning analytics.

References

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *The Journal of the Learning Sciences, 4*(2), 167–207.

Berland, M., Baker, R. S., & Blikstein, P. (2014). Educational data mining and learning analytics: Applications to constructionist research. *Technology, Knowledge and Learning, 19*(1–2), 205–220.

Bloom, B. S. (1968). Learning for mastery. Instruction and curriculum. Regional education laboratory for the Carolinas and Virginia, topical papers and reprints, Number 1. *Evaluation Comment, 1*(2), 1–12.

Breslow, L., Pritchard, D. E., DeBoer, J., Stump, G. S., Ho, A. D., & Seaton, D. T. (2013). Studying learning in the worldwide classroom: Research into edX's first MOOC. *Research & Practice in Assessment, 8*, 13–25.

Corbett, A. T., & Anderson, J. R. (2001). Locus of feedback control in computer-based tutoring: Impact on learning rate, achievement and attitudes. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 245–252), ACM.

Daries, J. P., Reich, J., Waldo, J., Young, E. M., Whittinghill, J., Ho, A. D., & Chuang, I. (2014). Privacy, anonymity, and big data in the social sciences. *Communications of the ACM, 57*(9), 56–63.

Ferguson, R., & Shum, S. B. (2012). Social learning analytics: Five approaches. In *Proceedings of the 2nd international conference on learning analytics and knowledge* (pp. 23–33), ACM.

Heffernan, N. T., & Heffernan, C. L. (2014). The ASSISTments Ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education, 24*(4), 470–497.

Ifenthaler, D., & Widanapathirana, C. (2014). Development and validation of a learning analytics framework: Two case studies using support vector machines. *Technology, Knowledge and Learning, 19*(1–2), 221–240.

Koedinger, K. R., Baker, R. S., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A data repository for the EDM community: The PSLC DataShop. *Handbook of Educational Data Mining*, 43–56.

Koedinger, K. R., D'Mello, S., McLaughlin, E. A., Pardos, Z. A., & Rosé, C. P. (2015). Data mining and education. *WIREs Cognitive Science, 6*, 333–353. doi:10.1002/wcs.1350.

Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., & et al. (2013). Prov-o: The prov ontology. *W3C Recommendation, 30*.

Lovett, M., Meyer, O., & Thille, C. (2008). The Open Learning Initiative: Measuring the effectiveness of the OLI statistics course in accelerating student learning. *Journal of Interactive Media in Education*, no. 14, JIME Special Issue: Researching Open Content in Education. http://jime.open.ac.uk/2008/14.

Miles, A., & Bechhofer, S. (2009). SKOS simple knowledge organization system reference. W3C recommendation, 18, W3C. http://www.w3.org/TR/2009/REC-skos-reference-20090818.

Pardos, Z. A., & Kao, K. (2015). moocRP: An open-source analytics platform. In *Proceedings of the Second (2015) ACM conference on learning@ scale* (pp. 103–110), ACM.

Siemens, G., Gasevic, D., Haythornthwaite, C., Dawson, S., Buckingham Shum, S., & Ferguson, R. (2011). *Open learning analytics: An integrated and modularized platform (Concept Paper)*. SOLAR.

Veeramachaneni, K., Halawa, S., Dernoncourt, F., O'Reilly, U. M., Taylor, C., & Do, C. (2014). *MOOCdb: Developing standards and systems to support mooc data science*. arXiv preprint arXiv:1406.2015.

Verbert, K., Govaerts, S., Duval, E., Santos, J. L., Van Assche, F., Parra, G., & Klerkx, J. (2014). Learning dashboards: An overview and future research opportunities. *Personal and Ubiquitous Computing, 18*(6), 1499–1514.

Xu, Z., Goldwasser, D., Bederson, B. B., & Lin, J. (2014). Visual analytics of MOOCs at maryland. In *Proceedings of the first ACM conference on learning@ scale conference* (pp. 195–196).