Ein generisches Datenmodell für Learning Analytics

Albrecht Fortenbacher, Marcus Klüsener, Sebastian Schwarzrock

Hochschule für Technik und Wirtschaft Berlin

Wilhelminenhofstraße 75A

12459 Berlin

{albrecht.fortenbacher, marcus.kluesener, sebstian.schwarzrock}@htw-berlin.de

Abstract: Bei der Wahl eines geeigneten Datenmodells für Learning Analytics sind folgende Fragen zu beantworten: welche Lernplattformen sollen unterstützt werden, welche Art von Analysen, wie viel (oder wenig) personenbezogene Daten sollen genutzt werden, welche Kontextinformationen, kann Plattformunabhängigkeit erreicht werden, können die Daten oder Analyseverfahren ausgetauscht werden? Und können Standards eingesetzt werden? Der vorgestellte Ansatz für ein Datenmodell kann nicht alle Fragen beantworten, bietet aber ein einfaches und klares Modell, welches für verschiedene Plattformen und Analyseanforderungen geeignet ist. Dies wird durch einen generischen Ansatz erreicht: Lernobjekte werden bzgl. ihrer Interaktionsform in drei Kategorien eingeteilt; konkrete Typen von Lernobjekten aus bestimmten Plattformen oder für bestimmte Analysen werden nicht im Datenmodell beschrieben, sondern erst für eine konkrete Analyse-Anwendung definiert. Das gleiche gilt für Attribute der Lernobjekte und für Attribute der Nutzer bzw. Lerner, welche bezüglich eines konkreten Lehr- und Lernkontextes definiert werden.

1 Einleitung

Learning Analytics kann als „... measurement, collection, analysis and report of data about learners and their contexts ...“ beschrieben werden [Si10]. Aktivitätsdaten von Usern werden durch die Loggingmöglichkeiten von LMS oder MOOC-Plattformen erhoben, aber auch auf Kollaborationsplattformen wie GoogleDrive oder auf Social-Media-Plattformen. Im sogenannten ETL-Prozess (Extract-Transform-Load) werden diese Daten gesammelt und in ein für die Analyse geeignetes Format überführt. Dieser Transformationsprozess kann deutlich aufwändiger sein als die eigentliche Analyse (vgl. [Ve13]. ETL ermöglicht eine Filterung von Daten, zum Zweck der Datenreduktion oder zur Anonymisierung, aber auch eine Anreicherung der „einfachen“ Log-Daten durch Informationen zur Lernsituation. Die Datentransformation kann vor jeder Analyse

durchgeführt werden, oder die Daten werden kontinuierlich aus einer Lernplattform in die Datenbank eines Analyse-Tools übernommen [EFM13].

Für die Definition des ETL-Prozesses wird ein Datenmodell der Analyse-Daten benötigt, unabhängig davon, ob die Daten persistent in einem Analyse-Tool gehalten werden. Die Datenmodelle der verschiedenen Analyse-Tools unterscheiden sich nach der Provenienz der Aktivitätsdaten (LMS, MOOC, Online-Enzyklopädie, Social Media), aber auch durch die Art der durchgeführten Analysen oder auf Grund des Personalisierungsgrads (Analysen eines Lernenden vs. Analyse von Lerntrends). Durch Untersuchung der Eigenschaften Generalisierbarkeit, Mächtigkeit und Aussagekraft von Student-Modeling-Systemen der letzten Jahrzehnte konnte eine Vielfalt an Anwendungen und Personalisierungsmöglichkeiten festgestellt werden. Unter Einbeziehung der technischen Entwicklungen und neuen Anforderungen wird es auch in Zukunft eine große Anzahl unterschiedlicher Modelle zum Student Modeling geben [Ko07]. Diese Vielzahl der Datenmodelle führt zu einer Inkompatibilität von Learning-Analytics-Tools, aber auch zu Standardisierungsbemühungen, von welchen die Open Learning Initiative (OLI), das MOOCdb-Projekt und das CAM-Schema hier vorgestellt werden (CAM steht für Contextualized Attention Metadata).

Vorgeschlagen wird ein generisches Datenmodell für Learning Analytics (gDMLA), welches durch eine einfache Struktur, und die generische Beschreibung von Lernobjekten und User-Attributen charakterisiert ist. Dadurch ist gDMLA für verschiedene Lernplattformen und Analyseformen geeignet. gDMLA wird in einer neuen Version des LeMo-Tools [EFM13] implementiert und ersetzt dabei das bestehende, stark an Moodle orientierte Datenmodell. Durch die einfache Anbindung an verschiedene Plattformen, und durch die Mächtigkeit bei der Darstellung analysespezifischer Attribute, könnte gDMLA auch zu einem Austauschformat für Learning-Analytics-Anwendungen werden.

2 Learning-Analytics-Daten

Lernanktivitäten werden durch Logging festgehalten. Beim Logging werden feinkörnige, umfangreiche Verlaufsdaten des Lernprozesses aufgezeichnet. Das Logging dient der Dokumentation der User-Interaktion / des Lernprozesses und speichert die Aktivitätsdaten in Form eines Protokolls. Elementare Bestandteile eines Log-Eintrags sind die User ID, die ID des Lernobjekts, sowie ein Zeitstempel. Weiterhin können eine ID des Kurses oder Aktionen wie Pause bei einem Video-Objekt oder Submit bei einem Assessment-Objekt aufgezeichnet werden. Über die ID des Users und des Lernobjekts, aber auch über andere Log-Einträge (bezüglich der zeitlichen Reihenfolge) können weitere Informationen zum Lernkontext erhalten werden.

Die Open Learning Initiative (OLI) bietet ein LMS mit externen Analysemöglichkeiten für Dozenten an. Um die Kompatibilität zu externen Systemen wie PSLC Datashop [PSLC] zu gewährleisten, wird ein Datenmodell vorgestellt, welches einen Datenaustausch auf Log-Ebene gewährleistet. Ein Logging-Service soll einen einfachen, verlässlichen und sicheren Datenaustausch ermöglichen.

User_id	Session_id	Source	Time	Time_zone	Action	Container	External_obj_id	Info_type	Info

Abbildung 1: Logzeile im OLI-Format

Für spezielle Analysen oder zu Analyseresultaten können weitere Metadaten hinzugefügt werden, zum Beispiel semantisch codierte Informationen über vermutete Kompetenzen oder den Wissenstand zur Zeit der Aktivität.

Contextualized attention metadata (CAM) ist eine Erweiterung von AttentionXML und eignet sich um Lernprozesse abzubilden [NDW06]. CAM beschreibt, wie Datenobjekte die Aufmerksamkeit der Nutzer gewinnen und welche Handlungen mit diesen Objekten in welchem Kontext durchgeführt werden. Mit CAM können kontextspezifische Nutzerprofile dargestellt werden. Es gibt unterschiedliche CAM-Schemata, die Daten über die Fokussierung der Aufmerksamkeit von Nutzern sammeln. Eine durch mehrere Iterationen herbeigeführte Vereinfachung ist in Abbildung 2 dargestellt.

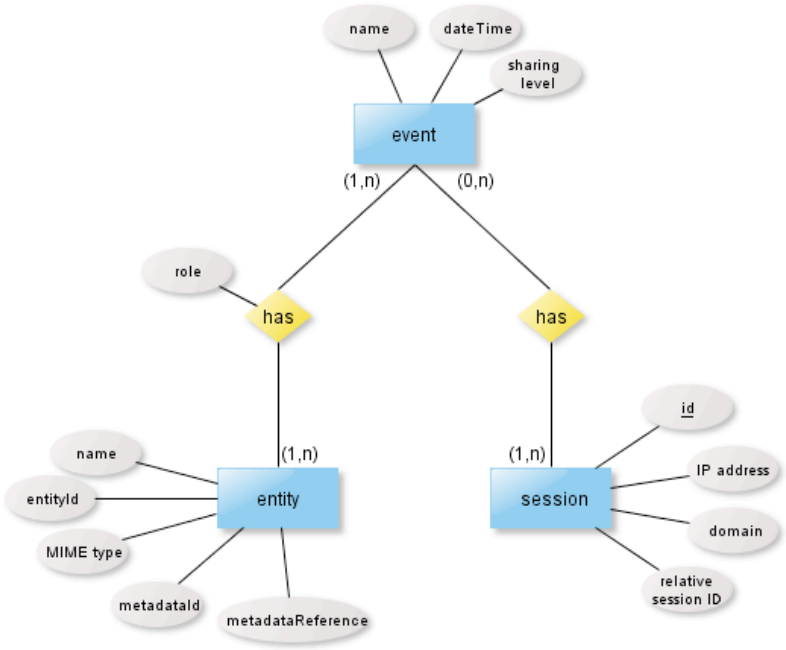

Abbildung 2: Von Google veröffentlichtes CAM-Modell [CAM]

Informationen über den Lernkontext können in einem Session-Objekt gespeichert werden, welches einen Zusammenhang zwischen Events beschreibt und ein Indikator für bestimmte Lernprozesse sein kann.

Die MOOCdb-Initiative stellt einen Ansatz für ein Datenbankschema vor, welches auf Analysen von MOOC-Daten ausgerichtet ist. Das Datenbankschema definiert eine Schnittstelle zu MOOC-Plattformen. Analysealgorithmen können plattformunabhängig

entwickelt werden, ohne die Notwendigkeit, Daten zwischen den Plattformen auszutauschen, was oft aus datenschutzrechtlichen Gründen nicht möglich ist.

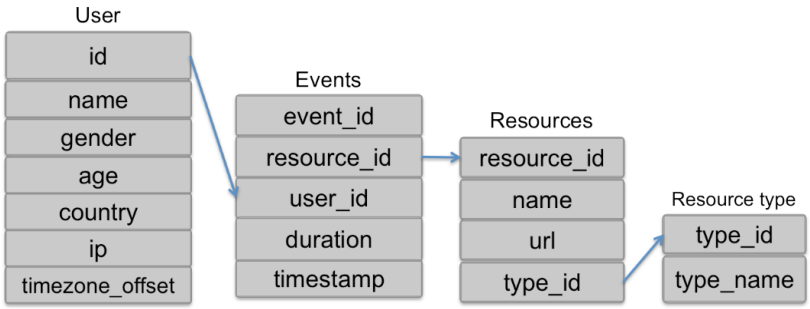

Abbildung 3: Datenbankschema für beobachtete Events [Ve13]

Ebenso wie in [Ko07] wird auch bei MOOCdb ein generischer Ansatz verfolgt, um verschiedene Plattformen unterstützen zu können. Anstatt einzelne Ressourcen (Lerninhalte) wie video oder document zu beschreiben, können über eine type_id auf jeder Plattform eigene Typen definiert werden. Lerninhalte und die dazugehörigen Aktivitäten (Events) werden nach Interaktionstypen unterschieden: Observing Mode, Collaboration Mode und Submitting Mode.

Das hier vorgestellte Datenmodell erweitert den generischen Ansatz von MOOCdb, um einerseits Aktivitäten genauer modellieren zu können, und um das MOOCdb -Modell auf auf weitere (nicht MOOC-spezifische) Daten und Analyseformen anwendbar zu machen. Der generische Ansatz vermeidet, wie in MOOCdb, die explizite Beschreibung einzelner Typen von Lernobjekten; statt dessen werden Lernobjekte nach den drei Interaktionstypen „Access“ (entspricht Observing Mode bei MOOCdb), „Collaboration“ und „Assessment“ kategorisiert. Entsprechend gibt es drei Typen von Aktivitätsobjekten mit unterschiedlichen Eigenschaften.

Der gegenüber MOOCdb weitergehende generische Ansatz bezieht sich auf die Eigenschaften der einzelnen Objekte. Dies betrifft die Aktivitäten (Log-Objekte), die Lernobjekte, welche durch eine konkrete Datenbankinstanz definiert werden, aber auch die User-Objekte. Attribute werden über Assoziationen hinzugefügt und können für jede gewählte Instanz verschieden sein. Die Implikationen dieses generischen Modells werden im übernächsten Abschnitt näher diskutiert.

3 gDMLA

Ein Hauptaugenmerk im Laufe des LeMo-Projekts [LEMO] lag auf der Abbildung von Daten verschiedener Lernplattformen für die Analyse in einer Datenbank. Das ursprünglich verwendete Datenmodell war an das Modell der Lernplattform Moodle [MO14] angelehnt, und die Abbildung der Daten anderer Plattformen auf das Modell gestaltete sich schwierig. Deshalb können in gDMLA, das wenige Standardattribute vorgibt, Ob-

jekte der Typen Kurs, Nutzer und Lerninhalt generisch um Attribute erweitert werden, um plattformspezifische Gegebenheiten zu berücksichtigen. Um ein möglich allgemeines Modell zu realisieren, das dennoch an eine Vielzahl von spezifischen Anforderungen angepasst werden kann, gibt es in gDMLA die Möglichkeit, die mit wenigen vordefinierte Attributen versehenden Inhaltsobjekte generisch um Attribute zu erweitern.

3.1 User und Kurse

Zur Abbildung von Daten der Plattformnutzer wird die generisch erweiterbare Tabelle User verwendet. Wie bei den anderen erweiterbaren Objekten des Datenmodells, soll somit die Möglichkeit gegeben sein, beliebig viele Attribute hinzuzufügen. Gründe hierfür sind die auf der Quellplattform vorhandenen und für die Analysen benötigten Daten, sowie die Möglichkeit der Einschränkung aufgrund von Datenschutz-Einschränkungen. Analyseergebnisse können durch weitere Benutzerattribute, wie zum Beispiel Geschlecht, Wohnort und Alter, verfeinert werden. Im LeMo-Projekt, welches sich vor allem an die Betreiber von Lernplattformen von Hochschulen richtete, war eines der Voraussetzungen die Gewährleistung des Datenschutzes. Aufgrund dessen wurden die in der Datenbank vorgehaltenen Daten bezüglich der Nutzer auf das absolute Minimum beschränkt. Abweichend davon wurde bei gDMLA die Erweiterung des Nutzerobjekts um personenbezogene Daten ermöglicht.

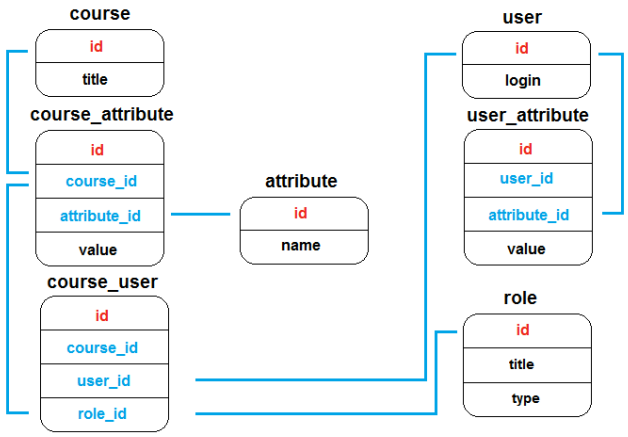

Abbildung 4: Schemaausschnitt User und Course

Das Attribut Login wird benötigt, um den Nutzer gegen das System zu authentifizieren.

Das Kursobjekt wird genutzt, um Lerninhalte zu gliedern. Ein Kursobjekt repräsentiert eine Ansammlung von Lerninhalten zu einem Thema. Es stellt die oberste Struktureinheit dar. Für den Kurs ist standardmäßig nur das Attribut Title vorgesehen, es können jedoch weitere Eigenschaften generisch hinzugefügt werden. Denkbar wären zum Beispiel Beschreibungstexte, Start- und Endtermine und Tags zur thematischen Einordnung. Über die Tabelle Course_User werden die Rollen der verschiedenen Benutzer innerhalb

der Kurse repräsentiert. Somit wird die Unterscheidung zwischen Studenten, Dozenten und Administratoren gewährleistet. Dies ist zum einen für die Bestimmung der Zugriffsrechte auf etwaige Analysen von Bedeutung und kann darüber hinaus auch verwendet werden, um in den Analysen Zugriffe von Studenten und Dozenten oder Administratoren zu unterscheiden.

3.2 Lernobjekte

Alle Lerninhalte werden in gDMLA mit Hilfe der Tabelle Learning_Obj abgebildet. Die Vielfalt der auf Lernplattformen vorkommenden Inhaltstypen und der möglichen Attribute einzelner Objekte ist einer der Hauptgründe für den generischen Ansatz des Datenmodells. Um einen Kurs inhaltlich zu gliedern, können spezielle Lernobjekte (Containerobjekte) über die Parent-Relation referenziert werden. Ein Anwendungsfall für eine derartige Unterteilung wäre zum Beispiel die Unterteilung eines Kurses in Kapitel und Lektionen, wie es auf MOOC-Plattformen üblich ist.

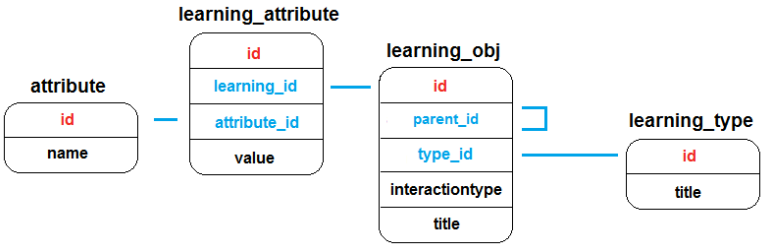

Abbildung 5: Schemaausschnitt Learning_Obj

Da die Zugriffsdaten der Lernobjekte, aufgrund ihrer Charakteristik, in diesem Modell auf drei verschiedene Tabellen verteilt werden, muss das Attribut interaction_type angeben, für welchen Interaktionstyp sich das jeweilige Objekt eignet. Es kann die Werte „Access“ (z.B. Videos, Dateien, Texte), „Collaboration“ (z.B. Foren, Chats, Wikis) und „Assessment“ (Tests, Assignments, Exams) annehmen. Ein hierarchischer Aufbau zwischen verschiedenen Typen, zum Beispiel zwischen Foren und Threads, kann durch die Parent_Id-Relation nachgebildet werden.

3.3 Logobjekte

Um Zugriffe der Nutzer auf Inhalte abzubilden werden die Tabellen Access_Log, Collaboration_Log und Assessment_Log verwendet. Die Aufteilung der Daten in verschiedene Logtypen ergibt sich dabei aus der Art der Interaktion. In unserem Modell findet sich, wenngleich die Benennung geändert wurde, die Unterteilung des MOOCdb-Projekts [Ve13] wieder. Alle drei Tabellen speichern den Nutzer (user_id), das Lernobjekt (learning_id), den Kurs (course_id) und den Zeitpunkt (timestamp) der Aktion. Darüber hinaus wird das Attribut „action“ bereitgestellt, um die Art des Zugriffs näher zu spezifizieren.

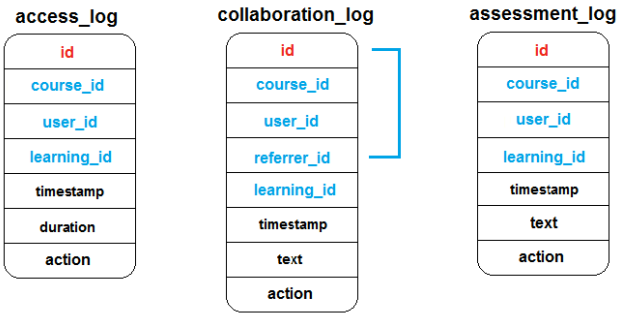

Abbildung 6: Schemaausschnitt Access_Log, Collaboration_Log und Assessment_Log

Die Tabelle Access_Log speichert Zugriffe auf Lerninhalte, die vom Nutzer nur „konsumiert“ werden, ohne Interaktion mit anderen Nutzern der. Im Gegensatz zu den anderen Logtabellen findet sich im Access_Log das Attribut „duration“. Dies soll genutzt werden, um die Verweildauer des Nutzers auf dem Lernobjekt zu erfassen. Typische Werte für das Attribut „action“ in der Tabelle Access_Log sind „play“ und „stop“ bei Videoelementen oder „view“ bei Texten.

Die Tabelle Collaboration_Log dient zum Abbilden von Interaktion mit anderen Nutzern. Die Tabelle unterscheidet sich von den anderen Logtabellen hauptsächlich dadurch, dass ein Attribut Referrer_id existiert, das ein anderes Collaboration_Log referenzieren kann. Durch diese Maßnahme soll es ermöglicht werden, inhaltliche Zusammenhänge zwischen einzelnen Nutzeraktionen darzustellen. Das beste Beispiel für die Funktionalität ist das Forum. Während das Forum selbst und der Thread durch ein Learning_Obj in der Datenbank abgebildet werden können, kann die Referrer_Id genutzt werden, um Zusammenhänge zwischen einzelnen Posts darzustellen. Beispiele für Werte für das Attribut „action“ in dieser Tabelle sind „Post“, „Comment“ und „View“.

Um Nutzerinteraktionen mit Lernobjekten zu erfassen, welche bewertet werden können, steht in gDMLA die Tabelle Assessment_Log zur Verfügung. Um effiziente Analysen von Bewertungen zu ermöglichen, wird das endgültige Ergebnis in der Tabelle User_Assessment gespeichert. Dadurch kann schneller erfasst werden, wer mit welchem Ergebnis an welcher Aufgabe teilgenommen hat. Das Feld „action“ kann in der Assessment_Log Tabelle beispielsweise Werte wie „submit“ oder „attempt“ annehmen.

Das vollständige Datenmodell kann auf der Webseite des LeMo-Projekts eingesehen werden [LEMO].

4 Fazit und Ausblick

Das Werkzeug LeMo, welches ein stark an Moodle orientiertes Datenmodell für Analysedaten besitzt, wurde auf gDMLA umgestellt, ohne durch die generische Struktur Daten für die Analysen zu „verlieren“. LeMo bietet insgesamt 14 Analysen an. 4 Nutzungsanalysen werden entweder als Aktivität über die Zeit (Funktionsdarstellung oder Heatmap)

oder als Aktivität pro Lerninhalt (Balkendiagramm oder Treemap) visualisiert. Die 5 Analysen zur Navigation der Studierenden werden in 2 Navigationsgraphen (mit unterschiedlichem Detaillierungsgrad), als Circle Graph oder als häufige Pfade dargestellt, welche mit zwei verschiedenen Algorithmen berechnet werden (siehe [EFM13]). Daneben gibt es eine Darstellung der Aktivitäten bezogen auf die Wochentage, sowie 4 Analysen zur erreichten Leistung in Tests (Quizzes). 10 dieser Analysen sind generisch in dem Sinne, dass Sie mit jeder gDMLA-Instanz durchgeführt werden können; lediglich die 4 Analysen zu Leistung hängen von der Definition eines Assessment-Lernobjekts ab. Auch die Interaktivität bei den Analysen mit LeMo, welche im Wesentlichen durch Filterung erreicht wird, kann durch gDMLA verbessert werden, da außer nach Lernobjekt-Typen auch nach dem Interaktionstyp (Access, Collaboration, Assessment) selektiert werden kann.

Auf Grund des generischen Charakters von gDMLA wird die Anbindung eines Analysetools an verschiedene Lernumgebungen stark vereinfacht. Zum einen ist das Datenmodell nicht an einer bestimmten Plattform (z.B. Moodle oder MOOC-Plattform) orientiert, was die Einschränkungen dieser Plattform überwindet. Dadurch können auch Lernumgebungen angebunden werden, welche ein mit Moodle oder der MOOC-Plattform inkompatibles Datenmodell besitzen. Interoperabilität mit verschiedenen Plattformen kann zusätzlich durch geeignete ETL-Tools, wie etwa das Framework Talend, verbessert werden. Durch die Umstellung auf gDMLA und Talend wird das Tool LeMo in Kürze auch für ILIAS, StudIP und für eine MOOC-Plattform verfügbar sein.

Literaturverzeichnis

[Ko07] A. Kobsa: Generic user modeling systems. In The adaptive web, pages 136-154. Springer, 2007.

[Si10] G. Siemens. Call for Papers of the 1st International Conference on Learning Analytics & Knowledge, 2010

[NDW06] Najjar, J., Duval, E., & Wolpers, M.: Attention metadata: Collection and management. In WWW2006 workshop on logging traces of web activity: the mechanics of data collection (pp. 1-4). (2006)

[Ve13] K. Veeramachaneni et Al.: MOOCdb: Developing Data Standards for MOOC Data Science" In Proceedings of the 1st Workshop on Massive Open Online Courses at the 16th Annual Conference on Artificial Intelligence in Education (2013). Memphis, TN.

[EFM13] M. Elkina, A. Fortenbacher, A. Merceron: The Learning Analytics Application LeMo – Rationals and First Results. International Journal of Computing, 12(3) 2013, pp 226-234.

Quellen

[CAM] Google: CAM-Scheme, URL: https://sites.google.com/site/camschema/ (Stand: 10.07.2014)

[LEMO] Datenmodell gDMLA , URL: http://www.lemo-projekt.de/gdmla (Stand: 10.07.2014)

[MO14] Moodle: Open-source learning platform, URL: https://moodle.org (Stand: 10.07.2014)

[PSLC] Data Shop, Guide to the Tutor Message format, URL: http://pslcdatashop.org/dtd/guide/ (Stand: 10.07.2014)