eReserve Student Engagement Analytics API

Introduction

The eReserve Student Engagement Analytics API will provide your analysts and development team with the ability to extract analytical data from the macro (e.g. for the school / department, course / paper / subject, reading list etc) to micro (e.g. the reading and the student). The analysts and development team can choose what level of analytical data they want to extract for ingesting into your institution's analytical applications / databases. Following is a detailed look at each of the resources (e.g. school, course, reading list etc) and what analytical data is available.

When using the eReserve Student Engagement Analytics API, in order to show the degree of student engagement and to identify at risk students and engagement with reading list materials, an institutions LMS using LTI standard will pass eReserve Plus limited amounts of student data. The student data includes the following:

- First name
- Last name
- email address
- LMS ID (an ID that is assigned to that student user of the LMS)
- student ID.

All data is securely stored and is handled with care, secure data handling procedures and in compliance with legal requirements and best practice security mechanisms.

Following are the key analytical information that can be accessed through the eReserve Plus Analytics API:

- School (aka department etc)
  - Basic metadata (i.e title / name)
  - Which course the school has
- Courses (aka course, subject, paper etc)
  - Basic metadata (i.e code, title / name)
  - Which reading lists the course has
  - Extraction of all other associated analytical data (e.g. reading access, student details etc) for a specific course
- Reading Lists
  - Duration of the reading list (i.e. start / end dates)
  - The reading list course (aka unit, subject, paper etc)
  - Identify how many readings are ready to be accessed by students and how many students are accessing those readings
- Readings
  - Basic metadata for the reading/s (e.g resource kind, source document title, authors, publication year etc)
  - The number of students accessing the readings on a reading list and how many times they have accessed it
  - Identify which students have been accessing readings and their individual accesses of readings on reading lists
- Teaching Sessions
  - Duration of the teaching session (i.e. start / end dates)
  - Reading lists that have been created using the teaching session
  - Extraction of all other associated analytical data (e.g. reading access, student details etc) for a specific teaching session
- Student Details
  - Basic metadata for identifying the student allowing for cross-referencing the eReserve Plus Analytics API with other institution systems
  - What readings a student has read along with how often and when they have read the readings

This API has been used by institutions to gain awareness of student engagement with resources on reading lists. The API can be used as part of data warehousing and to identify at risk students and to understand engagement with resources in detail.

In this document, you can also find a link and/or an image to a eReserve Student Engagement Analytics ERD which will describes the relationships between the API resources used in the API.
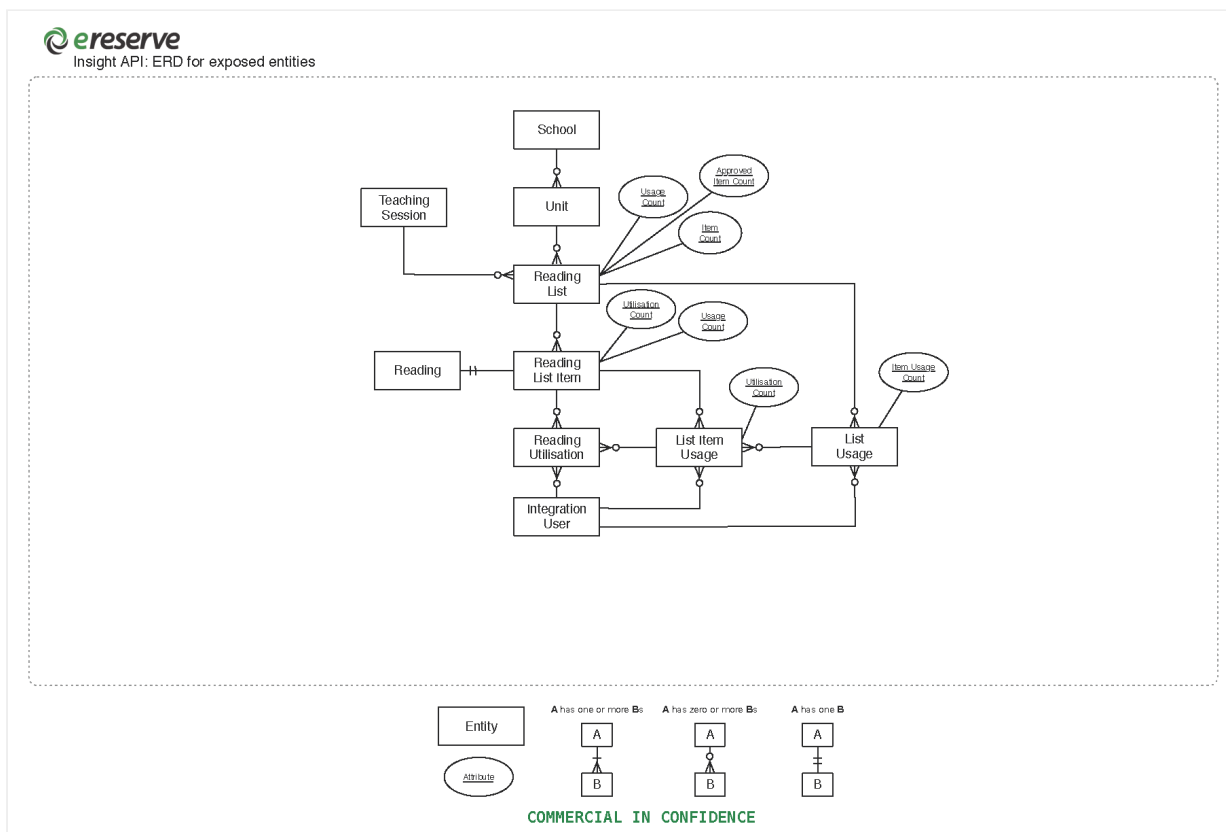

Figure 1: eReserve Plus Student Engagement Analytics API ERD

If you have any questions, please do not hesitate to contact our support team at support@ereserve.com.au

School

Through the `School` resource, the analysts and the development team can explore the schools (aka departments) maintained within eReserve Plus by library staff, and then use the resource records in subsequent API calls.

Course

Through the `Course` resource, the analysts and the development team can explore the courses (aka unit, paper, subject etc) maintained within eReserve Plus by library staff, and then use the resource records in subsequent API calls.

Reading Lists

Through the `ReadingList` resource the analysts and the development team can access macro level analytical details. This includes:

- when the reading list is available (i.e. start / end dates)
- how many readings are on the list
- the number of approved readings; and
- the number of readings that have been accessed by students.

If you are looking for more finer detail, the analysts and the development team can dig further, getting details about the following:

- Student Access Counts Through the related `ReadingListUsage` resource the analysts and the development team can identify how many readings a student has accessed on the reading list. This can be taken further to identify how many times a student has accessed a reading on the reading list through the `ReadingListItemUsage` resource.

- Reading List Readings
  Through the `ReadingListItem` resource, the analysts and the development team can get a list of the readings for the reading list and extract details such as the number of students that have accessed the reading and how many times the reading has been accessed.

Reading Student Accesses

Through the `ReadingUtilisation` resource, the analysts and the development team can get the detail for every reading accessed. This includes the time of access, the student accessing the reading and reference to the reading itself. Through the related resources more details can be extracted about the student and the reading.

- Student Details
  Through the related `IntegrationUser` resource, the analysts and the development team can get identifying metadata (e.g. First / Last name, email address, student number etc)

- Reading
  Through the related `ReadingListItem` resources, the analysts and the development team can retrieve the number of times a reading has been accessed on a reading list and how many students have accessed it. This also acts as a stepping stone to basic reading metadata (e.g. reading type, title, source document, authors, publication year etc) through the related `Reading` resource. This allows the analysts and the development team to create a picture of the activity around the reading.

Student

Through the `IntegrationUser` resource, the analysts and the development team can access identifying metadata including:

- the First and Last name
- email address
- student number; and
- when they were created.

Through the `IntegrationUser` relationships with the `ReadingUtilisation`, `ReadingListItemUsage` and `ReadingListUsage` resources, the analysts and the

development team can build a picture of what readings a student has accessed, how often and what reading lists they have accessed during their studies.

Teaching Session

Through the `TeachingSession` resource, the analysts and the development team can access basic teaching session metadata such as:

- the Name of the Teaching Session,
- the Start and End dates; and
- when it was Created & last Updated at.

Through its relationship with the `ReadingList` resource, the analysts and the development team can collect analytical information about reading lists and associated readings for the teaching session.

Deprecated fields

The eReserve Analytics API has been updated to include new fields and further refine the naming of existing fields. These fields are aimed at allowing your development team to streamline their interaction with the API.

The following fields are detailed below for your development team including the new fields and deprecated filters for resources on the API.

> NB: All deprecated filters will be removed as of 1st of February 20201.

Reading resource

- New field: source_document_created_at
- New field: source_document_updated_at
- New field: created_at

- New field: updated_at

ReadingList Resource

- New field: unit_id
- New field: teaching_session_id

ReadingListUsage

- New field: list_id
- New field: integration_user_id
- Deprecated Filter: filter[reading_list_id] (please use list_id)
- Deprecated Filter: filter[authenticated_user_id] (please use integration_user_id)

ReadingListItem

- New field: list_id
- Deprecated Filter: filter[reading_list_id] (please use list_id)

ReadingListItemUsage

- New field: item_id
- New field: list_usage_id
- New field: integration_user_id
- Deprecated Filter: filter[reading_list_id] (please use list_id)
- Deprecated Filter: filter[reading_list_item_id] (please use item_id)
- Deprecated Filter: filter[authenticated_user_id] (please use integration_user_id)

ReadingUtilisation

- New field: integration_user_id

- New field: item_id
- New field: item_usage_id
- Deprecated Filter: filter[reading_list_item_id] (please use item_id)
- Deprecated Filter: filter[authenticated_user_id] (please use integration_user_id)