

Proposed Update to Submissions API

Rational: It is often useful to access submission data for all assignments for an entire class. This data can be used in a custom gradebook, a course dashboard, early detection of struggling students, a leaderboard, etc.

The current Submissions API returns ALL submissions that fit the requested parameters. In large classes with a large number of assignments, this can quickly become a very large number of submissions. My current class reaches over 10,000! In addition, the Canvas API only returns 100 submissions per page. This means that large submission requests are parsed into as many as 100 API requests or more for EACH page load! The Canvas API paging system does not allow asynchronous requests, so each page must be delivered and then the next requested, sequentially. Also, in large classes, these kinds of requests occur many times a day and I suspect that so many page requests from the same course results in API throttling which further slows down the requests.

The result is that API requests for all submissions in large classes can take 10s of minutes! Often the request fails to complete and/or returns corrupt data. This makes it nearly impossible to build a custom gradebook, course dashboards for teachers and students, an early detection system for struggling students, a leaderboard, etc. for large classes, which stand the most to benefit from these kinds of tools.

The traditional solution for improving performance of API data is to store it locally. However, this approach is only useful for data that does not change often. Submissions data are very dynamic and change frequently, also there is no way to know if new data is available. As a result, fresh data, the entire data set, must be requested each time a user wants to see the current status. Again, this can take 10s of minutes and may be a significant drag on Canvas servers. Another approach is to use the Live Events API to collect data locally as it is generated. This approach is also not very useful in this use case. First, the Live Events API returns data on an account level, not a course level. So, to get data for a class, a developer must request data for their ENTIRE university, a solution that is way overkill and has all kinds of security issues. Second, the Live Events API is still in beta and is not available.

Solution: A simple solution for this problem is outlined below. By adding `submitted_after[]` and `graded_after[]` parameters to the “list submissions for multiple assignments” API, developers can request only submission data that are new since the last request was made, and store updates locally. In a large course, data will be requested multiple times a day, and thus local data is updated multiple times a day, reducing the size of any individual submissions API request and dramatically reducing the amount of data requested from Canvas servers for a course. This will lower page load times from 10s of minutes to seconds and make a variety of innovative components possible! Coding this change to the API will likely be trivial, simply requiring a couple of ‘if statements’ and WHERE clauses on the DB request.

List submissions for multiple assignments

SubmissionsApiController#for_students

```
GET /api/v1/courses/:course_id/students/submissions
```

```
GET /api/v1/sections/:section_id/students/submissions
```

Get all existing submissions for a given set of students and assignments

Request Parameters:

Parameter	Type	Description
<code>submitted_after[]</code>	Date/Time	Return only/all submissions with <code>submitted_at</code> dates greater than a specified date / time
<code>graded_after[]</code>	Date/Time	Return only/all submissions with <code>graded_at</code> dates greater than a specified date / time