

Small Code Snapshots Working Group Report

David Hovemeyer, York College of Pennsylvania, dhovemey@ycp.edu
Kelly Rivers, Carnegie Mellon University, krivers@andrew.cmu.edu

Our goal

Define a standard data representation for programming process data

- From programming exercise systems, IDEs, etc.

Make it (relatively) easy to analyze data produced in a variety of contexts

Allow development of common/reusable analysis tools

The process

Late 2017: process starts

February 2018: informal meetings at SIGCSE get the ball rolling; essential requirements are agreed upon

February 2018–November 2018: spec gradually takes shape, lots of discussion and collaboration

November 2018: first “implementable” draft, work on CSEDM workshop paper starts, work on prototype data exporters starts

Challenging issues

- Event ordering: Global, restricted, or no ordering? Mandate “physical” ordering?
- Timestamps: Client vs. server? Optional vs. required? Local times or include timezone? When are they trustworthy?
- Edits: How to tag types of edits?
- Code state representation: Easy/dumb (text blobs in CSV or files) vs. hard/smart (Git)

Principle: Let the producer of the data accurately describe what is known and unknown—this is why so many columns are optional

The essential difficulty of doing this

Given the vast diversity of data being collected, can a data representation be

1. General enough to capture data from many contexts
2. Descriptive enough that data can be recorded with high fidelity
3. Not so complicated that it is unnecessarily difficult to do analysis

Time will tell if we've struck the right balance.

Where we are (in slightly more detail than earlier)

Draft spec is implementable

Details will continue to evolve as we gain experience

Areas requiring more attention include:

- Event ordering
- Tagging edits
- Code state representation (especially the Git-based one)

What remains to be done

Completely working data exporters

Client libraries and tools to assist in implementation of analyses

- This is the fun part!

Real use cases

How you can be involved

This is a good time to get involved: a lot of the hard work has already been done

- Start by reading the spec:
 - <https://csssplice.github.io/progsnap2>
- Join the google group (email Dave or Kelly for an invite)
- Get your hands dirty