Closing the feedback loop for clicker questions

Simon Bates* and
Keith Brunton
School of Physics and Astronomy
University of Edinburgh
Edinburgh
EH9 3JZ
***s.p.bates@ed.ac.uk**

Abstract

We describe the output from a recently-funded JISC Learning and Teaching Innovation Grant: Electronic Voting Analysis and Feedback for all (EVAF4All). We have created a web-based software tool (EVAF) that allows electronic voting system data captured at the point of delivery in lectures, to be fed back to students, thus providing valuable formative feedback of their progress over what can be a large number of such questions. In institutions where 'loanership' models of handset distribution are used (typically, when students keep the same handset for a whole course or year) this is particularly powerful as it can supply students with their own data as well as the aggregate data from the rest of the cohort. Academic staff can use the tool to evaluate the effectiveness of their clicker questions as an aide to course monitoring or development processes. We briefly cover the technical aspects of the system we have built and also present a case study of its use in an introductory Physics course taught at the University of Edinburgh.

EVS background and motivation

The use of electronic voting systems (EVS) in taught classes, as an aid to interactive engagement and rapid feedback to students is now well-established within many disciplines in Higher Education. Comprehensive resources provide detailed information on the technology, pedagogy and various use scenarios of these handsets[1-3]. In the UK, the original JISCmail mailing list for EVS adopters has broadened and expanded to an online community exploring all forms of classroom technology to engage students[4]. Textbook publishers now routinely provide extensive electronic bundles of ‘concept tests’ with new titles or revised editions, a name for EVS questions coined in Eric Mazur’s 1997 seminal text on peer instruction[5].

We have created a web-based software tool (EVAF) that allows electronic voting system data captured at the point of delivery in lectures, to be fed back to students, thus providing valuable formative feedback of their progress over what can be a large number of such questions.

We have been using EVS in introductory Physics courses in Edinburgh for the past 6 years, developing our questions and methodology of use during this time. The technique has been widely adopted throughout Schools in Science and Engineering and deployment of handsets to practically all first year students is organised by our library. Students receive a handset at the start of the Semester and retain this for the duration of their course, returning it at the end of the Semester or academic year to the library, just as they would with borrowed textbooks. This mapping, matching a given student to a particular handset, is a key feature that we have exploited in the design of the online system described in this communication.

Our motivation for this project was to try and make use of the huge amount of data that was collected at the point of question delivery, usually in lectures, over a given course. In our first year Physics course we typically pose between 50 and 80 questions during the Semester-long programme of 30 lectures. In a class of 300 students, this represents an enormous volume of data that usually languishes on the laptops of the various staff who lecture on the course. This data would have value to both students and staff if it could be presented in a readily-digestible format, without unreasonable effort by busy academic staff. The data could provide students with detailed personalised feedback on their own performance on the range of EVS questions, together with the aggregate whole-class data. Staff would be able to use anonymous aggregated class data for question development, course monitoring and review. EVS software provided by handset vendors certainly does permit *some* of this functionality, but it tends to be focussed on staff evaluation of questions rather than provision of feedback to students. Whilst some of this software does integrate with ‘gradebook’ functionality in major Virtual Learning Environment (VLE) systems the maintenance of class lists and rosters within the EVS software is by-hand and time consuming for large classes.

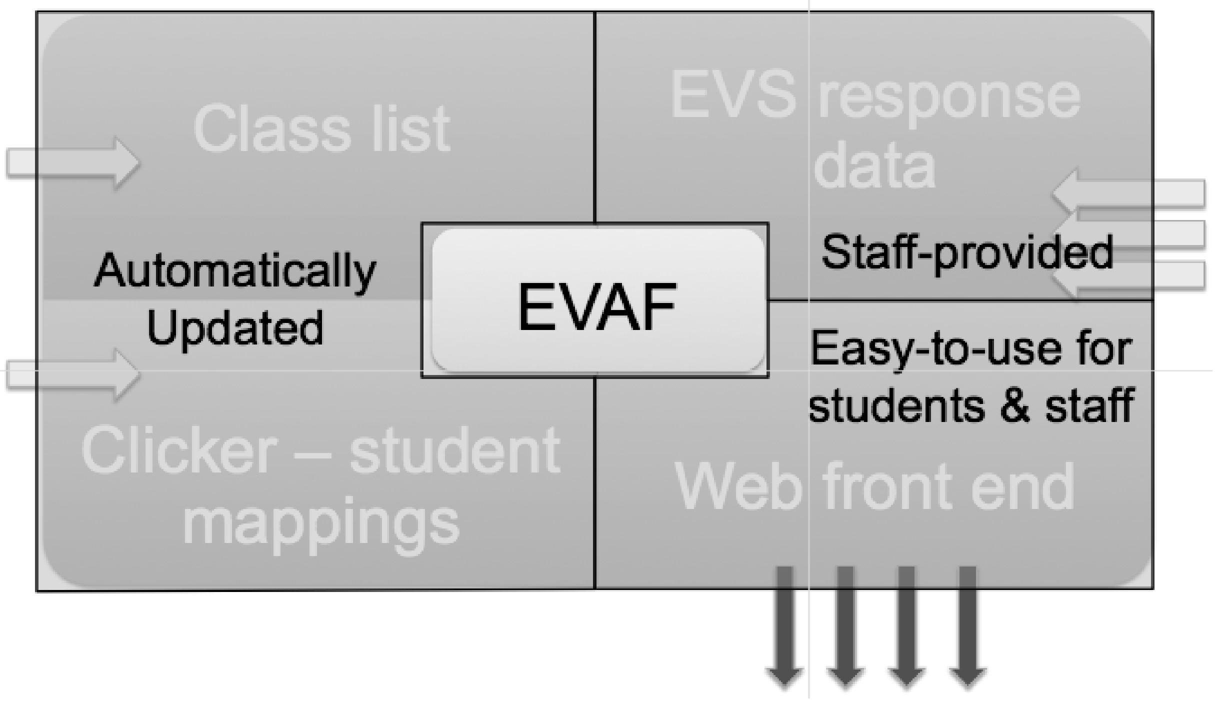


Figure 1: Schematic map of components and functionality in the EVAF software system.

We bid to the JISC 'Learning and Teaching Innovation Grant' scheme in 2008[6], to build an online system to make it easy to return this data to staff and students which was capable of working at other institutions by interfacing with their authentication systems and VLEs. The project was funded and ran for one year during 2009. This communication presents a brief overview of the system architecture and how it has been used in our first year Physics course in the 2009-10 academic session. Much more detailed documentation and access to the software developed during the project is available from the project homepage[7].

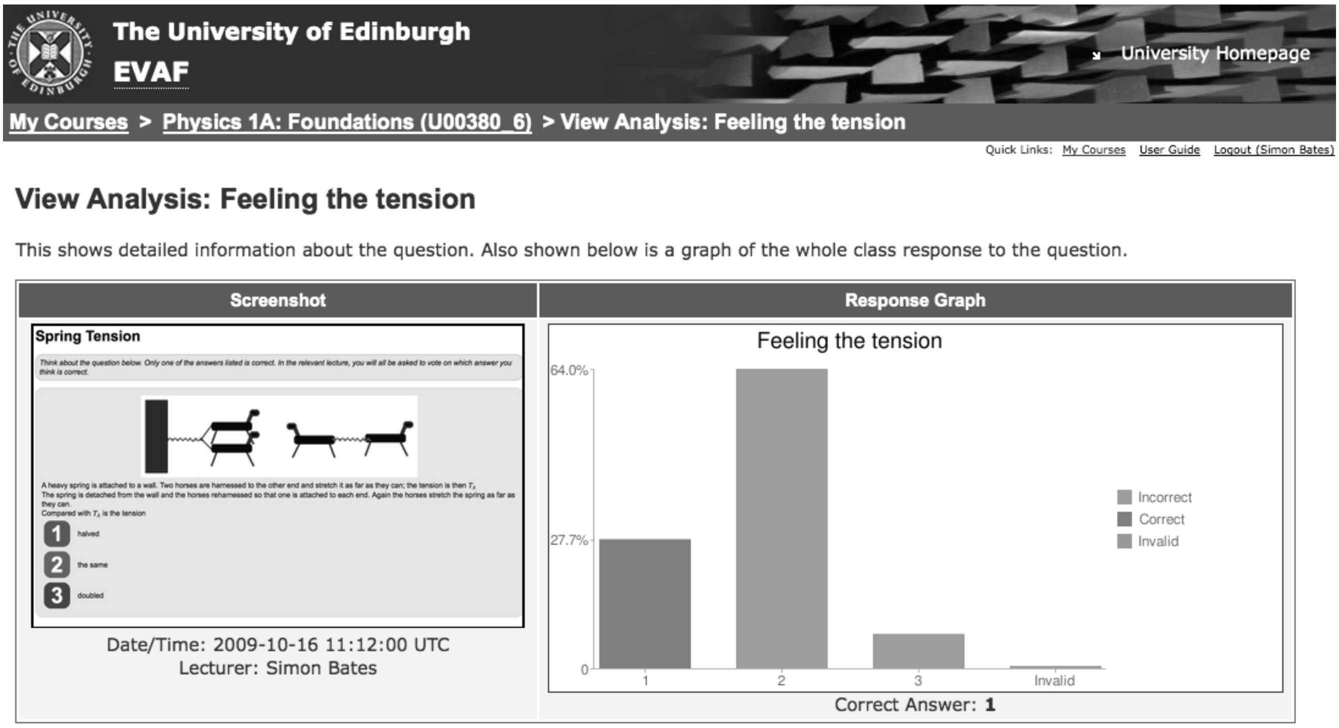
Architecture of the system

There are three components to the EVAF software: the most visible front-end software functionality, the 'connector' functionality and deployment/configuration aspects that allow the software to be installed and integrated to another institution. Here, we focus on the first two of these and Figure 1 schematically illustrates these components in the software. EVAF necessarily requires other behind-the-scenes information sources besides the voting data uploads: this is illustrated on the two left hand blocks of Figure 1. The class list of students on a particular course and clicker-student mappings form the basis of the information needed to identify individual students on a course with a given handset. These can be automatically updated from the VLE and loanership scheme records, respectively, or can be managed via manual uploading of .csv files. An automated feed of class list data is particularly useful at the start of large courses, as there is inevitably traffic into and out of the course after it has commenced. Within the year's duration of the project, we successfully implemented connectivity modules to one of the main EVS software export formats (e-Instruction) and the WebCT VLE platform. The system's modular software design is such that it can be extended to interact with different VLEs (though this is problematic for some VLEs) and import EVS data from different software. Additionally, EVAF has the capacity to use local authentication mechanisms for login of staff and students.

The other main data ingredient needed by EVAF is illustrated in the upper left hand block of Figure 1: the EVS response data provided by staff after collection during teaching sessions. It is a simple form-based upload screen, with the system trying to 'guess' sensible values for various fields based on what is read in from the files. It was neither practical nor desirable to try and automate this step completely; some staff even within a given course may prefer to use their own laptop or the lecture theatre fixed PC to collect data. To build in a specific data upload location would have been overly prescriptive here.

The most important part of the system is the web interface through which students and staff of an institution can review questions posed and votes that were cast during lectures using EVS. Staff across multiple courses can upload their saved voting data so that students can view, analyse and reflect on how they voted in lecture in the context of the rest of the class. Staff can view aggregated data, review efficacy of questions and add screenshots or comments to questions. EVAF has granular use and course management allowing staff to work in teams and encouraging colleagues to share findings and good practice. Context sensitive help is also provided, tailored according to role (student, staff, administrator etc).

Figure 2 shows the staff view of a typical question stored within EVAF. The screenshot on the left is a clickable link which expands to provide the viewer with a clear view of the question that was posed. (For staff who utilise EVS software



The University of Edinburgh

EVAF

My Courses > Physics 1A: Foundations (U00380_6) > View Analysis: Feeling the tension

Quick Links: My Courses User Guide Logout (Simon Bates)

View Analysis: Feeling the tension

This shows detailed information about the question. Also shown below is a graph of the whole class response to the question.

Screenshot	Response Graph
Spring Tension Think about the question below. Only one of the answers listed is correct. In the relevant lecture, you will all be asked to vote on which answer you think is correct. A heavy spring is attached to a wall. Two horses are harnessed to the other end and stretch it as far as they can; the tension is then T_A The spring is detached from the wall and the horses reharnessed so that one is attached to each end. Again the horses stretch the spring as far as they can. Compared with T_A is the tension 1 halved 2 the same 3 doubled Date/Time: 2009-10-16 11:12:00 UTC Lecturer: Simon Bates	Feeling the tension Correct Answer: 1

Figure 2: Screenshot of the staff view of an EVS question and response set stored within EVAF.

integrated with PowerPoint, these question screenshots are captured automatically at the point of question delivery). The right hand panel shows the whole class response data. The student view of this page is similar, with the addition of an arrow to indicate the response chosen by that particular student (together with a tick or cross if it is correct or incorrect).

Other useful features of the software include a full-text search facility, a ‘leaderboard’ option to show the clicker ids (not student ids!) of those with the most questions correct and a question filter feature. This last option comprises built-in filters for questions which are deemed ‘easy’ (>75% of the class answering correctly), ‘hard’ (<40% correct) and ‘tricky’ (most popular answer is not correct). Rather than present endless screenshots, we have prepared a short screencast that highlights many of the features of the system[9].
EVAF is written primarily in the Ruby language and the popular Ruby-on-Rails platform[8] which allowed for the agile development of features and functionality. The EVAF front-end does not require the use of any browser plugins. Other technologies used include SQL, XML (Web Services) and some freely available software & JavaScript APIs for components such as graphs (Google Chart API) and image manipulation (ImageMagick). Relevant Web standards have been kept in mind during development.

Implementation and use in Physics 1A
'Physics 1A: Foundations' is a first-year first-Semester course in classical Physics, taken by approximately 300 undergraduate students each year. It is compulsory for all students reading towards Physics degrees (these typically comprise half the cohort) and also available as an elective option for students on other degree programmes (chiefly, but not exclusively science degrees). The course enrols students with a wide range of previous study of the subject, ranging from one year's study post Standard Grade (Scottish Higher) to A-levels and an increasing number of International Baccalaureate (IB) candidates. This range of diverse background (in terms of both aptitude and prior study) coupled with the nature of the material (familiar, yet often counter-intuitive aspects of Newtonian mechanics) makes for its own particular set of challenges instructing a very heterogeneous class cohort.

The course has a long history of a blended learning approach, with an e-learning component designed to supplement (not supplant) the face to face teaching activities, which goes back over a decade. The course has been using the pedagogy associated with EVS handsets for approximately the same length of time. Initially, these questions were introduced and voted on with coloured cards with the switch to voting handsets taking place 6 years ago. There are three lecturers teaching on the course, aided by workshop demonstrators, technical and administrative support staff.

EVAF was not introduced to the students until around week 4 of an 11 week teaching semester. This is the first semester of (most) students' first year at University and there is plenty of information that students are bombarded with at the start of the course (and doubtless in other courses they are taking too). Experience shows that things have settled down after approximately 3 weeks of the course, and additionally a useful number of EVS questions have already been covered in lectures. At the end of the 3rd week of lectures, EVAF is shown to the students at the start of a lecture as 'the place where all the clicker data goes' and students are shown how to access the tool from the course homepage on the VLE. At the start of the 4th week of lectures, the course lecturer demonstrates how to view real data captured in the previous lecture and explains the sorts of things that students can do with it.

The tool is linked not only from the course homepage on the VLE, but also linked through the online content sections that are the EVS questions. (It is possible to link *directly* to individual question datasets in EVAF, however the option we have used is to include a 'top level' link to the course page for EVAF that can then be drilled down into individual questions). Course staff have already made use of EVAF to evaluate student performance on various questions, with a view to development of future questions or replacement of questions.

Google analytics was used to track site visitors to EVAF from the Physics 1A course. There were 650 visits to the site during the course, averaging between 10 and 35 visits most weekdays. Usage tended to tail off towards the end of the course, but persisted beyond the cessation of teaching and into the examination period in December. Visitors spent on average a few minutes on the site during each visit and approximately half the traffic on the site was returning visitors. One aspect that seemed to capture the imagination of the students was the introduction of a 'leaderboard' feature on the site: there was a competition run with a small prize for the student who answered the highest percentage of correct answers. This was clearly quite motivating for the stronger students in the class. The commenting functionality of the site (where students could comment on particular questions) rather like a discussion board thread was not used.

In the light of this modest but encouraging uptake, the course team are considering future plans for the use of the system in the course next year. At present, EVS questions are used in a single mode of delivery: asking questions in lectures. We are experimenting with and considering other methods of use, such as building 'consolidation' or review exercises on the EVS questions into the tutorial workshop sessions for the course, and also the use of student generated content (EVS questions) within the course.

Acknowledgements
Project funding for the development of EVAF was provided by JISC under their Teaching and Learning Innovation Grants programme.

References

1. Bates, S.P. Howie, K. and Murphy A. StJ. (2007). *The use of electronic voting systems in large group lectures: challenges and opportunities*. New Directions in the teaching of Physical Sciences, **2**, 2006, 1-8.
2. Draper, S. (2010). Electronic voting systems and interactive lectures [online]
<www.psy.gla.ac.uk/~steve/ilig>
[accessed 3rd May 2010]
3. Carl Wieman Science Education Initiative, UBC (2010). CWSEI – Clicker Resources [online]
<www.cwsei.ubc.ca/resources/clickers.htm>
[accessed 2nd May 2010]
4. ESTICT – Engaging students through in-class technology (2010) [online]
<estict.ning.com/>
[accessed 3rd May 2010]
5. Mazur, E. (1997) Peer Instruction, Prentice Hall New Jersey
6. JISC Learning and Teaching Innovation Grants (2010) [online]
<www.jisc.ac.uk/whatwedo/programmes/elearning/ltig.aspx>
[accessed 29th November 2009]
7. Bates, S.P. and Brunton, K. (2010) EVAF – Electronic Voting Analysis and Feedback (2010) [online]
<www2.ph.ed.ac.uk/elearning/projects/evaf/>
[accessed 26th February 2010]
8. Ruby on Rails (2010) [online]
<rubyonrails.org/>
[accessed 6th May 2010]
9. Bates, S.P. (2010) EVAF features and functionality [online]
<tinyurl.com/EVAFfeatures>
[accessed 7th May 2010]

One aspect that seemed to capture the imagination of the students was the introduction of a 'leaderboard' feature on the site: there was a competition run with a small prize for the student who answered the highest percentage of correct answers.