On Teaching Programming Languages Using a Wiki*

Iliano Cervesato

Carnegie Mellon University — Qatar Campus
42866 Doha, Qatar
iliano@cmu.edu

This paper reports on an ongoing experiment with using a wiki as support material for CMU-CS 15-212, a sophomore-level course on advanced programming concepts and skills. We found that the use of a wiki promotes participatory learning and encourages the students to read upcoming material before class. Future developments include pairing up the wiki with an e-tutor and supporting delivery to small-screen devices such as PDAs and smartphones. A wiki is also a natural medium for a repository for essential programming language concepts and material for teaching such concepts.

1 Context and Opportunity

CMU-CS 15-212 — *Principles of Programming* — is a sophomore-level course which teaches students advanced programming concepts and skills, including a first exposure to programming language semantics. For the past couple of years, 15-212 has had little supporting material as textbooks used in the past (e.g., [HR99, Pau96]) have either gone out of print or become obsolete[1]: the students are expected to master the contents of the course on the sole basis of the notes they take in class, code posted on the class web page and handouts for a fraction of the lectures. This is a challenge for many of them.

As a remedy to this situation, we have embarked in the task of putting the whole course material on a wiki [CSe08]. This wiki contains detailed explanations of the material covered in class, with special emphasis on examples. It categorizes and cross-references this material. It also provides exercises, as well as pointers to further readings and advanced material.

Wiki editing is relatively simple and transparent, which is conducive to thinking about the big picture rather than detailed formatting issues. It also encourages segmenting the material into relatively small chunks and linking them together through dependencies. The possibility of embedding LaTeX code has proved very useful for the more theoretical parts of the course. We also made heavy use of a drawing extension to provide graphical explanations of many concepts.

The 15-212 wiki was started in November 2007 and it now contains about 25% of the material covered in the class. It is being expanded, by both instructors and students, as the

*Partially supported by the Qatar Foundation under grant number 930107.

[1] A recent manuscript written for 15-212, [Har05], is at too early a stage of development to act as a primary reference.

semester unfolds. It is expected to cover the entire set of topics covered in 15-212 by the summer of 2008 and to be further extended in years to come as new topics are explored. As with every wiki, there is no finish line, just a reasonable steady state.

2 Wiki-Based Instruction

To the extent of our knowledge, this endeavor differs from previous attempts to use wikis in a course (e.g., Luis von Ahn's pioneering work for CMU-CS 15-251) in that it is meant not so much as a substitute for a textbook (although this is definitely needed in 15-212) but as a comprehensive didactic tool centered around the notion of participatory learning.

Because it is a collaborative framework, a wiki allows the instructor to involve the student in the creation and especially improvement of contents.[2] In 15-212, we have encouraged the students to come up with explanations better than our own and put them on the wiki. Such volunteer edits (in the true spirit of a wiki) were rewarded with "participation points". We have also done this in a more structured way, by asking them to write specific pages as part of their homework (with good results). This empowers students and makes them active participants in the didactic process. It is also a way to get them to research topics and have them gain a deep understanding of a concept because they have to explain it to others. Indeed, the whole experiment was well-received by the students.

The wiki software we used (the open-source MediaWiki platform that powers Wikipedia) provides simple ways for an administrator to spot recent changes to an article and especially to revert to a previous version if these changes are found to introduce inaccuracies or be inappropriate in other ways. It is equally easy to monitor newly added articles. Since every edit is logged, it is also very easy to find out what each student did, and therefore to reward good work, penalize sloppy edits, and monitor the absence of any edit. It is indeed a good barometer of the health of the class.

The wiki logs every access, not just edits. This is a good way to make sure that the students have read the material before coming to class, which allows for a more interactive and focused in-class discussion. To support this, we had to disable the standard guest account which would allow anyone to view the contents of the wiki without logging in.

3 Future Developments

In the future, we plan to extend the reaches of this experiment in several directions, each of which may require the development of dedicated extensions to MediaWiki.

Because in 15-212 most programming concepts are demonstrated by providing actual code that implements them (currently in SML), it would be natural if the wiki provided an easy way to test short program, both as a way to run the given code on different inputs, and also for the student to experiment with other ways of solving problem. We plan to investigate the possibility of embedding a terminal frame within the 15-212 wiki, which, when opened, would launch a sandboxed interpreter for the implementation language in use.

[2] The fact that a wiki is a collaborative framework also enables instructors teachings courses similar to CMU-CS 15-212, their students, or any interested person to contribute to its contents. For temporary technical reasons, this is currently restricted to users in the *cmu.edu* domain, but this constraint will be lifted shortly.

We mentioned that the 15-212 wiki contains exercises. If a student finds difficulties, he or she must come see the teaching staff. A natural next step is therefore to extend the wiki with an e-tutoring system that could monitor the comprehension of the material and deliver it as appropriate for the level of the student.

Another interesting development would be to offer a version of the contents reformatted for viewing on personal devices such as cellphones and PDAs, which could complement more traditional delivery mechanisms. The same medium could also be used to deliver short video segments of the actual lecture. This would allow a student to use otherwise dead times (e.g., commute to school, time between classes, and time waiting for friends) to review a topic and possibly read about related topics, which is easy to do with hyperlinks.[3] This solution can be further developed by pairing up a web server providing these pages formatted for PDA-viewing with a system able to process student input. This can go from a simple web form that provides the student with multiple choice quizzes and exercises to an e-tutor that maintains a model of the students and presents him/her with exercises that maximize learning.

Looking beyond the particular use described above, the collaborative nature of what goes into a wiki makes it an ideal medium as a repository of the accepted concepts and approaches in a discipline. This is particularly true for the field of programming language foundations as modern wiki software provide support for a lot of what is needed to describe programming language concepts, in particular LaTeX, drawing capabilities, and much more.

References

[CSe08] Iliano Cervesato, Thierry Sans, and *other contributors*. The 15-212 wiki. `https://wiki.qatar.cmu.edu/cs/15212/wiki/`, 2008. Currently restricted to course attendees; please contact *iliano@cmu.edu* for access.

[Har05] Robert Harper. Programming in Standard ML. Working draft, 2005.

[HR99] Michael R. Hansen and Hans Rischel. *Introduction to Programming Using SML*. Addison-Wesley, 1999. Out of print.

[Pau96] Lawrence C. Paulson. *ML for the Working Programmer*. Cambridge University Press, 2nd edition, 1996. Out of print.

[3] The hypothesis behind this idea is that the traditional way the lecture material is organized, in linear segments of an hour or more matching traditional class duration (80 minutes at CMU), is not necessary and is probably not efficient. Smaller segments, centered around individual topics (or subtopics) require a much shorter study time (typically 5 to 20 minutes). Organizing the material along these lines allows the student to use otherwise dead time to go over a segment in its entirety. In these situations, a mobile solution is often preferable to opening books or waiting for a laptop to boot. Providing the readings in small bites also makes the bulk of the material appear less intimidating.

This requires the instructor lays down the structure of the material very accurately so that the student can easily follow the dependencies (this piecemeal approach breaks the sequentiality of the traditional material presentation). A basic approach is to put the material in specially designed web pages formatted for small-screen viewing. This provides mobility and does not force sequentiality of topics as the instructor can use hyperlinks to define dependencies between topics.