DHQ: Digital Humanities Quarterly
2009
Volume 3 Number 2

Done: Finishing Projects in the Digital Humanities

Matthew G. Kirschenbaum <mgk_at_umd_dot_edu>, University of Maryland

Abstract

How do we know when we're done? This cluster of articles explores completion and incompletion in the digital humanities from a variety of perspectives.

Done. Way back when I worked as the project manager for the *William Blake Archive*, there was no more satisfying sequence of characters than those five keystrokes (including punctuation). *Record newly received transparencies in project log.* Done. *Parse files in holding directory.* Done. *Purchase new scanner.* Done. *Proofread documentation for image annotation applet.* Done. It was easy to measure progress as I ticked off tasks in email messages to the archive's editors. In time, when enough of these individual tasks were "done," the project might be finished. 1

Or would it? How do we *know* when we're done? What does it mean to "finish" a piece of digital work? As Bill Kretzschmar points out in his essay, the verb "to finish" can mean to *complete* or something more like to polish or perfect. What is the measure of "completeness" in a medium where the prevailing wisdom is to celebrate the incomplete, the open-ended, and the extensible? Digital humanities and the electronic publishing ventures associated with it are used to deriving considerable rhetorical mileage and the occasional moral high-ground by contrasting their radical flexibility and mutability with the glacial nature of scholarly communication in the fixed and frozen world of print-based publication. 2

It is also, let us admit, usually more exciting to begin something than to finish it. We partake in what Julia Flanders has aptly called the culture of the perpetual prototype: the demo, the proof-of-concept, the alpha and beta version. Building things is fun, for a lot of us it's why we got into this business; debugging and documenting them, tweaking and patching them is not. This tendency extends to our grant cycles as well. In the United States, we now have the great benefit of digital humanities start-up grants, but not dedicated awards for finishing or closing projects. Granting agencies reward innovation and originality, not steady, measured progress. No one writes a grant application to say, "With just two more years of funding we can finish the good work we've started. No, we're not going to be doing anything new. We just want finally to put this thing to bed." 3

It may be tempting to write off worn out tools, abandoned projects, and occasionally even outright vaporware as the victims of a rapacious technological obsolescence stalking all our endeavors. But this doesn't tell the whole story. Nor is the digital humanities unique in its uncertainty — arguably its timidity — over knowing when to call something done. As Silicon Valley journalist Scott Rosenberg suggests, software development has an important analog in a specific point of computational theory, the "halting problem" as defined by Alan Turing. Knowing how or when a particular version or release is finished and ready for public debut is an ongoing and often heated debate among coders. Rosenberg quotes from Alan Cooper's *The Inmates are Running the Asylum*: "Software development lacks one key element — an understanding of what it means to be 'Done' " [Rosenberg 2007, 337]. 4

But we cannot escape the pressure of milestones, deadlines, deliverables, and products (as opposed to projects) as the measure of academic success and the currency of professional promotion and reward. They are also what allow our work to escape a small circle of developers and enthusiasts and become part of a larger scholarly community, where it can be used in the consequential manner presumably always intended. 5

6 At Digital Humanities 2007 I organized a panel to begin addressing what it actually means to finish work that we've started. The essays that follow are significantly revised and expanded from the remarks the authors presented at the conference. William A. Kretzschmar's piece traces the local history of one particular project, grafting milestones and markers of what's been "done" to various versions and releases as his tools and technologies evolve. His remarks raise issues not only for how one assesses closure, but also digital preservation and version control. David Sewell addresses the questions I posed from the perspective of a publisher. His work at the University of Virginia Press's Rotunda digital imprint offers a different context, one where finding some measure of "done" is a professional and financial necessity. Finally, Susan Brown and her colleagues from the *Orlando* project offer the view from one of the largest and longest running undertakings in the digital humanities, at the moment when the project has gone to press. Brown discusses what got done and what it will take for it to stay that way.

Works Cited

Rosenberg 2007 Rosenberg, Scott. *Dreaming in Code*. New York: Crown, 2007.