**"Satisfying the need for determinate searching: Labs, APIs, and search engines" by Gabriel Egan**

As a project, the Text Creation Partnership or TCP is almost invisible. The majority of its users access TCP via the ProQuest Corporation's Early English Books Online (EEBO) website-database, for which it provides the full-text search feature. Without TCP's typed up transcriptions, EEBO is just a collection of pictures of pages from books and is searchable only by its metadata--author, title, publisher, and so on--and not searchable by the words on each page. Yet in essays based on searching EEBO-TCP, investigators who use its full-text searching often leave off the -TCP suffix and claim that they have searched within EEBO, which is untrue. At the New Oxford Shakespeare we made heavy use of EEBO-TCP in our new edition and we do not commit this error: we say "EEBO-TCP" when that is what we mean.

For many users, the integration of TCP within EEBO gives them everything they need. Many, but not all. Some investigators, for example those doing authorship attribution, need definitive answers to questions about whether certain words and phrases are in the corpus, or not. That 'not' is crucial, since you might fail to find something not because it isn't there but because you didn't look properly. For example, Brian Vickers thinks that particular phrases and collocations from *Macbeth* are not found anwhere in the canon of Thomas Middleton, and so he thinks Middleton did not adapt the play. But the only reason that Vickers fails to find these phrases in Middleton's work is that he searches in a private corpus that lacks most of what Middleton wrote. If you search for those phrases in a complete and publicly accessible corpus, like EEBO-TCP, they show up in Middleton's work. This fundamental error also vitiates Vickers's attributions of several works to Thomas Kyd.

If you want to be definitive about absence, you have to start with a complete corpus. You also have to be scrupulous about how you phrase your searches, which is difficult because unfortunately each of the tools we use--OED, EEBO-TCP, STC, Literature Online--has a different convention for expressing complex searches by specifying the desired proximity of the targets, the application of Boolean logic ('this AND/OR that'), and the use of wildcards to do grammatical stemming. A frequent cause of error is that users misremember the various conventions needed and enter their searches incorrectly. And even if you express your search correctly, you might still get back the wrong answer because the database lies to you. In June 2014, the ProQuest Corporation upgraded its server software and accidentally broke the advanced searching features of their flagship product Literature. As of yesterday when I last checked, Literature Online's advanced searching remains broken: the numbers it returns for complex queries are just wrong. Luckily, the New Oxford Shakespeare editors spotted this fault as soon as it occurred and made sure it didn't affect any of the edition's claims about who wrote what.

All these faults are fixable. We could all use one international standard for expressing searches; it exists, it's called *regular expressions* and you should learn it even if only to make Microsoft Word work better for you. We could choose to use only databases like TCP that are available to everyone and have known contents. And we could insist that the databases our universities pay for actually work as adverstised. We would then have definitiveness in our searching and that should be our aim.

But that would only be the beginning. A lot of interesting searches are much easier to state in words than to actually perform. If you have a list of one hundred phrases and want to know how various combination of them appear in books across the 16th century, that can entail entering into the TCP or EEBO-TCP website thousands of searches. It is possible to use software that robotically impersonates the researcher and tirelessly types a long series of queries into the website form one at a time, clicks 'Search', and then gathers the resulting output from the website in one file to show you. This is known as screen-scraping, and although it works after a fashion it is fragile: if the database provider changes the layout of the screen, the screen-scraper will start to press the wrong buttons and put your search terms in the wrong places.

There are two ways forward for the user who wants to perform computationally intensive searches of a digital corpus like TCP. One is to download the entire TCP corpus to your own machine and run the searches locally. With TCP this is quite feasible as the entire dataset of 25,000 books in Phase One is available for anyone to download and fits easily on a $5 USB memory stick. But it takes some technical knowledge to do the downloading and still more to develop the software needed to perform the searches on your machine. For most users the best solution is to leave the data on a webserver with a search engine, but provide something better than just a website form for inputting the queries. We need an Application Programming Interface or API. This is an agreed set of standards for how anyone may, over the Internet, send queries to the server holding the dataset and how the results that come back will be formatted. APIs are a widely used and well-tried technology. They are, for example, the technology that enables the easy embedding of Google maps into anyone's website.

Just over a decade ago the British government made ProQuest an offer they couldn't refuse and for a one-time payment bought the entire EEBO dataset of images and metadata. The government's computer department called Jisc set about building its own website front-end to rival ProQuest's EEBO. The Text Creation Partnership files were added to enable full-text searching, and then was added the whole of Gale Cengage's Eighteenth-Century Collections Online database (ECCO), which was also purchased outright. The resulting database, called Jisc Historical Texts, is sold as a subscription to British universities at a fraction (about one-twentieth) of the cost of an EEBO subscription.

Jisc Historical Texts allows you enter searches as regular expressions and later this year it will offer an API and a  'Labs' feature that enables the construction of long, computationally intensive searches that are automatically fed to the API and into the search engine. The intention is to allow users without advanced computer skills to perform queries that are easy to state--'search for all these terms in this way'--but are tedious to type one-by-one. Unfortunately, due the terms of the licence from ProQuest, no-one outside the UK may use Jisc Historical Texts. But the TCP Phase One dataset is free, so any enterprising institution may host a copy of it, add a search engine, and publish the necessary API. I hope to be able to show in a year or two some case studies of TCP research done in the UK via the Jisc Historical Texts API and 'Labs' service that will, I hope, convince fellow investigators here in the US that what we need now is a worldwide API for TCP.