

My Life as a Programming Historian

José E. Igartua 14 April 2015

(revised and updated 26 December 2020)

My participation at McMaster's [#MACBIG 2015](#) event has brought memories of how I got to be a programming historian.

I started in 1967 as a Ph.D. student at Michigan State University: I learned FORTRAN on punch cards; what a drag! I thought I could use it in my dissertation, but was soon disabused for lack of data.

1974-1978 in the Social Science Computing Lab at Western: punch-card 10 statement FORTRAN on a Control Data CYBER mainframe; doing text processing on a DEC 10 mainframe on a teletype terminal; BASIC on DEC PDP 8s and 11s on a CRT terminal; SPSS on punch cards for the CYBER. I gave up trying to learn binary code programming. For the Landon Project, I wrote a data entry BASIC app on the PDP for inputting printed tabular data (published census data on South-Western Ontario) for storage on the mainframe. No Excel then!

1978-1981: At Université du Québec à Chicoutimi (UQAC), besides running SPSS jobs, I spent a week toying with APL: seriously elegant but required a special keyboard with Greek characters. And not easy to use on data sets, as I recall.

At UQAC I taught computing to History undergraduates and MA students.

1981: Moved to UQAM, which had a terrific research computing support team. I used SIR hierarchical database software (command language modelled on SPSS) for various research data. UQAM quickly switched to relational ORACLE databases, in which I stored the Alcan employee records and the Arvida assessment rolls. I became very proficient in SQL for my book on Arvida. I also used SPSS on that data. I dabbled in INGRES databases at Université du Québec à Chicoutimi, to access the Saguenay demographic data remotely. INGRES supported both the SQL query language and its own QUEL query language; I used the former and the Saguenay Project programmers preferred QUEL. We got the same results.

Nearly forty years ago, in December 1981, I bought my first IBM PC: I splurged for 48k of RAM and two floppy drives for \$5000; there was no hard drive available. The IBM PCs had better keyboards and screens than Apple II. I could program in BASIC at home! I first used email programs on an Amdahl (IBM 360 compatible) mainframe to communicate with colleagues in other universities. I programmed interfaces between PCs and the mainframe for data upload and editing. I made my PC a terminal to the mainframe. I could compute on the mainframe from home! WordPerfect was the first PC

word processor that handled footnotes. WYSIWYG too! No more text processing on UQAM's DEC 10.

1990s: UQAM moved from IBM mainframe to Unix servers for research data. All my ORACLE databases followed. I learned Unix OS. I even installed ORACLE on a PC. The dawn of the Web meant learning HTML and the use of Apache servers. With a stellar team of colleagues, I created Histoire-Hypermédia (www.h-h.ca), a training tool for history undergraduates learning to do library research. I learned to read PHP and do simple PHP code changes for the Histoire-Hypermédia project when the project had an intranet developed in PHP and MySQL to facilitate updating the project's bibliographic data. I used SQL on that project's MySQL databases.

In 1989, I launched the Canadian Historical Association's History and Computing committee in 1989 and chaired it to 1997. It was the Canadian branch of the International Association for History and Computing. I organized its 1995 Computing conference in Montreal. I developed an interest in computer-assisted history learning but discovered I was getting out of my depth (one had to absorb history education theory and practice as well as extensive web programming).

Late 1990s: I had a look at [Microcosm](#) hypertext mark-up software from Southampton University and Manfred Thaller's [Kleio](#) database system designed specifically for historical research: commands in Latin! I learned it is more practical to use commercial software.

2000s: I worked with professional Web historian-programmers at Edmonton's Chinook Multimedia, notably on the CHA's short-lived Canadian History Portal, which proposed a curated collection of web resources on Canadian History. The Portal was shut down for lack of sustained funding.

2005-2008: While its creator, Francis Chateauraynaud, was visiting the Centre interuniversitaire de recherche sur la science et la technologie at UQAM, I discovered [Prospéro](#), a sophisticated, chronological text analysis package produced by a group of French sociologists. I used it to explore the contents of textbooks for my 2008 article in the *[Journal of Canadian Studies / Revue d'études canadiennes](#)*, 42, 3 (Fall 2008): 106-132. See my brief description of Prospéro [here](#).

I had a look at Python, which reminded me of BASIC...

I used MapInfo's UQAM site licence in teaching computer tools for history M.A. students.

SPSS became available on PCs! A freeware clone, [PSPP](#), is available. Also available free for PCS is the PHP/MySQL developer environment, [EasyPHP](#), which includes Apache servers. You can install the EasyPHP environment on a PC or a Mac to develop applications which you then upload on a secure web server. But it is only "easy" if you have some familiarity with HTML, PHP, and SQL on relational databases.

Retired since 2008.

As an experiment, I once installed [Xubuntu](#) OS (a variant of Linux) on some elderly laptops, to explore the OS. All the app software is free, and the OS was faster than Windows XP on these machines. The GUI is very well done (I was using the French-language distribution). However, resizing disk partitions to enable booting either in Windows or in Xubuntu is not for the faint-hearted! Since most of the software I use regularly runs on Windows, I stayed with Windows for everyday computing.

“Take-away”: For historical research, computer software evolves, comes and goes, but basically, all computers do is execute algorithms that a scholar designs to answer a research question with data. It the QUESTIONS and the quality of the data that matter! But learning programming gives you an idea of what goes on beneath the nice-looking user interfaces.