The divergent histories of particle physics and computing

Jim Pivarski

Princeton University – IRIS-HEP

September 13, 2019

Prehistory

We can start anywhere (Antikythera mechanism? Ishango bone?),
but let's start with the Hollerith machine.

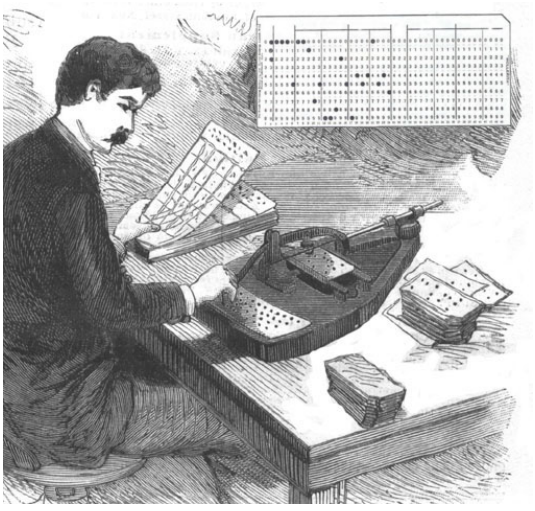
The U.S. Census's problem

The U.S. does a census every 10 years. The 1880 census took 8 years to process.

$\longrightarrow$ Big data problem!

Held a competition for a new method; winner was $10\times$ faster than the rest:

Census records on punch cards, which filtered electrical contacts



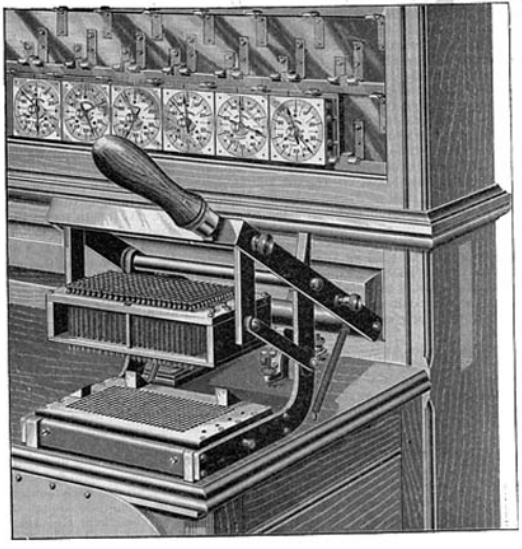


Fig. 8.—Circuit-Closing Press.

Hollerith's Electric Sorting and Tabulating Machine.

Wired to a machine that opens a door for each matching pattern

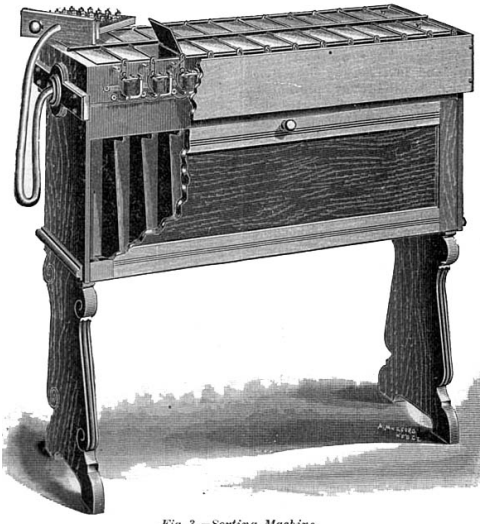
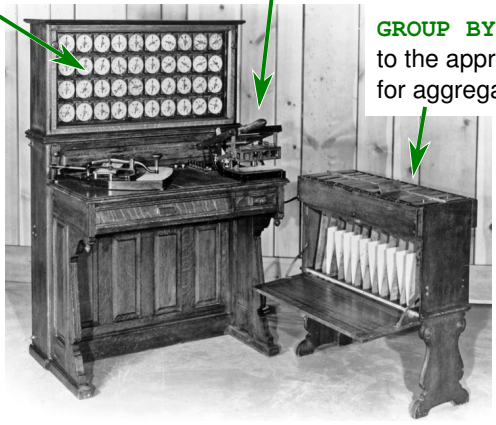


Fig. 3.—Sorting Machine.
Hollerith's Electric Sorting and Tabulating Machine.

It was an SQL machine: 3 basic clauses of most SQL queries




```
SELECT name WHERE literate GROUP BY marital_status
```

Origin of business computing

Herman Hollerith founded a company selling these machines, which, after a series of mergers, became International Business Machines.

https://www.officemuseum.com/data_processing_machines.htm

https://www.ibm.com/ibm/history/exhibits/builders/builders_hollerith.html

Physics interest in computing came later

Nuclear/particle physics was a tabletop science before the Manhattan Project (nuclear bomb).

Exceptions:

- Ernest Lawrence's group at Berkeley (invented accelerators) employed dozens—disparaged as "Berkeleitis." https://history.aip.org/history/exhibits/lawrence/epa.htm
- Cryogenics was big science in the early 20th century, but didn't require much number-crunching.

Physics and computing don't converge until the 1940's.

Physicists got into computers when they became general-purpose

1944: John Mauchly (physicist) and J. Presper Eckert (electrical engineer) designed ENIAC to replace mechanical computers for ballistics.

ENIAC was one of the first computers driven by machine code instructions, stored as a program in memory.

1945: John von Neumann learned of their work and suggested using it for nuclear simulations (H-bomb).

His internal memo describing ENIAC's stored programs was leaked; now known as "Von Neumann architecture."

Los Alamos group led by Nicholas Metropolis, developed Monte Carlo techniques for physics problems.

Computers got general-purpose when physicists got involved

1944: John Mauchly (physicist) and J. Presper Eckert (electrical engineer) designed ENIAC to replace mechanical computers for ballistics.

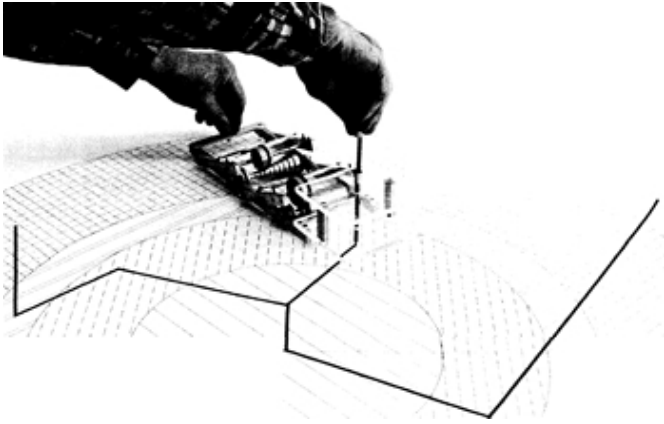
ENIAC was one of the first computers driven by machine code instructions, stored as a program in memory.

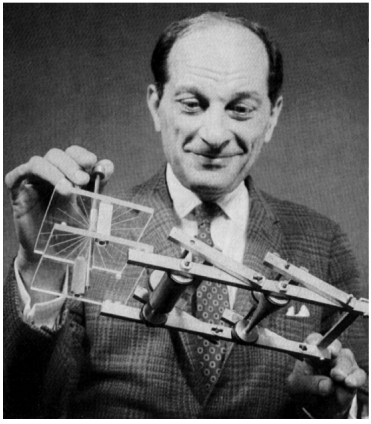
1945: John von Neumann learned of their work and suggested using it for nuclear simulations (H-bomb).

His internal memo describing ENIAC's stored programs was leaked; now known as "Von Neumann architecture."

Los Alamos group led by Nicholas Metropolis, developed Monte Carlo techniques for physics problems.

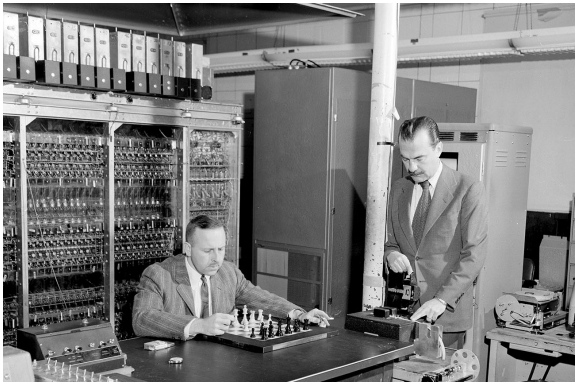
What were they doing before that?

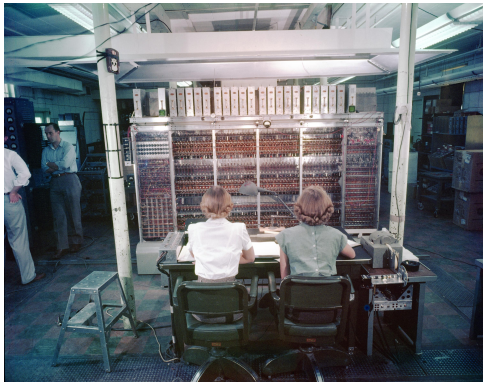




https://www.tandfonline.com/doi/abs/10.1080/23324309.2018.1514312

Metropolis And Neumann Invent Awful Contraption





https://www.atomicheritage.org/history/computing-and-manhattan-project

https://www.manhattanprojectvoices.org/oral-histories/nicholas-metropolis-interview

https://www.jstor.org/stable/20025423

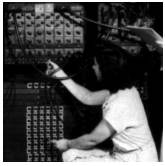
https://books.google.com/books?id=qB819m2ibUQC

The actual programming was performed by these six women

Kathleen McNulty

Frances Bilas

Betty Jean Jennings



Ruth Lichterman

Elizabeth Snyder

Marlyn Wescoff

Kathy Kleiman's research:

http://eniacprogrammers.org/eniac-programmers-project/

There are many secondary-source articles like this one:

http://mentalfloss.com/article/53160/meet-refrigerator-ladies-who-programmed-eniac

This is a fantastic article; an overview of women in computing until the 1980's:

https://www.nytimes.com/2019/02/13/magazine/women-coding-computer-programming.html

First assembly language invented by Kathleen Booth:

http://bobmackay.com/Booth/Booth.html
http://www.computinghistory.org.uk/det/32489/Kathleen-Booth/

Eckert-Mauchly Computer Corporation → Remington Rand

Mauchly and Eckert "went into industry" selling computers; the first one (UNIVAC) to the U.S. Census.

1950: Short Code, the first executable high-level language: a transliterated interpreter of mathematical formulas.

```
math: X3 =  (  X1 +  Y1 )  /  X1 * Y1
code: X3 03 09 X1 07 Y1 02 04 X1   Y1
```

50× slower than machine code because it was interpreted.

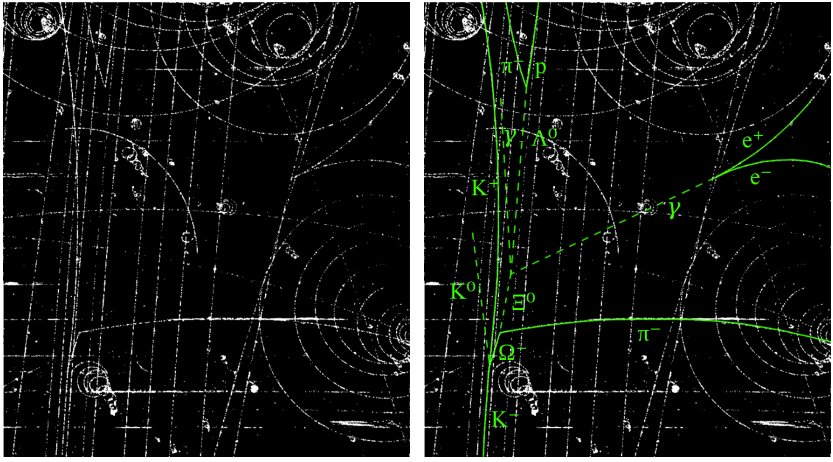
1952–1959: At Remington Rand, Grace Hopper developed a series of *compiled* languages, ultimately COBOL.

Meanwhile, IBM developed FORTRAN: 1954–1957.

http://www.historyofinformation.com/detail.php?entryid=839

The physics analysis problem

Particle collision/decay images are rich datasets. Analyzers need to infer "who decayed to whom" and measure the curvature of tracks, which yields momentum.



Fully manual → human/computer → fully automated

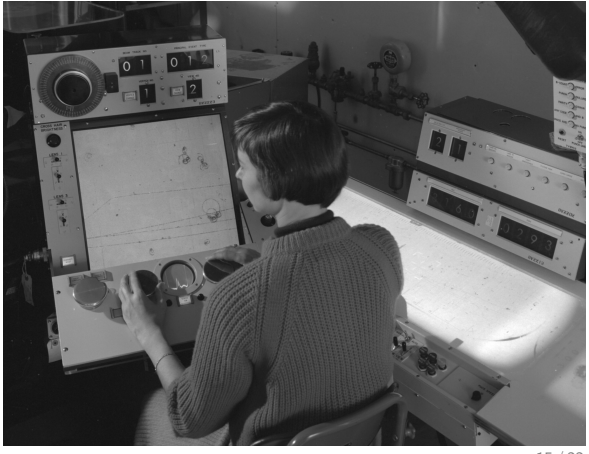
$2M bubble chamber, $0.2M IBM 650

Jack Franck's "Franckenstein" converted positions marked along tracks into numbers on punch cards for analysis.

https://www2.lbl.gov/Science-Articles/Research-Review/Magagine/1981/81fchp6.html

Madeleine Isenberg describes being a "scanner."

http://www.physics.ucla.edu/marty/HighEnergyPhysics.pdf



Processing events thousands of times faster

1959 paper on the software infrastructure: PANG, KICK, EXAMIN, DRIVEL

http://inspirehep.net/record/919917/files/HEACC59_575-583.pdf

Image and Logic: A Material Culture of Microphysics by Peter Galison:

https://books.google.com/books?id=6Gcu92U8CwYC&lpg=PA373&ots=8YV5cPePE6&dq=Franckenstein%20bubble&pg=PA373#v=onepage&q=Franckenstein&f=false

Discovering Alvarez: Selected Works of Luis W. Alvarez: that plot $\longrightarrow$

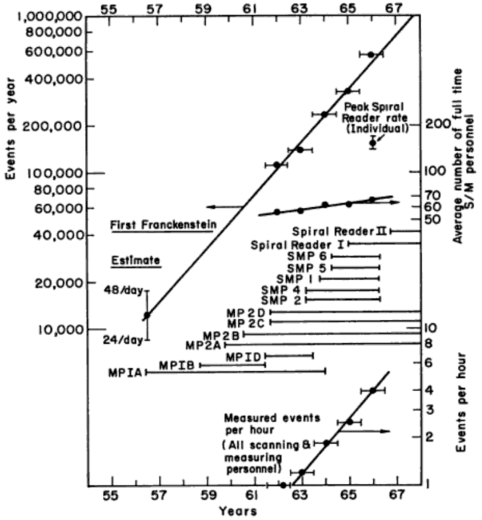
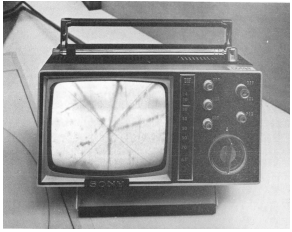


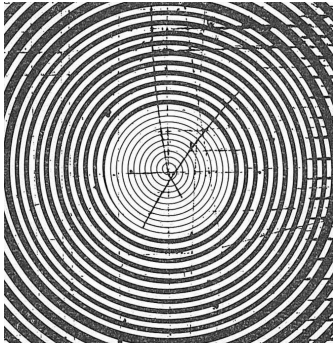
Fig. 9.
Measuring Rates.

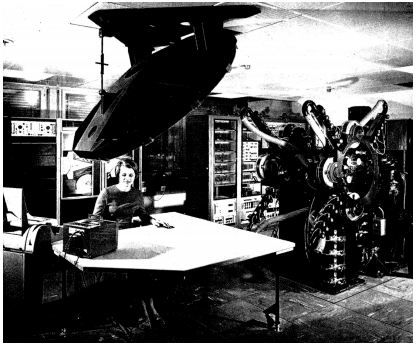
Getting more automated: the spiral reader

Known as the "LSD" (Lecteur à Spirale Digitisée) at CERN. The analyst only needs to center the spiral on the vertex, specify the number of tracks, and POOH finds them.

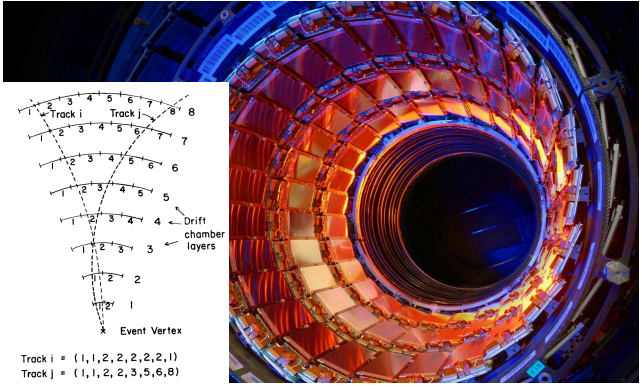
https://escholarship.org/content/qt7hf5r27c/qt7hf5r27c.pdf

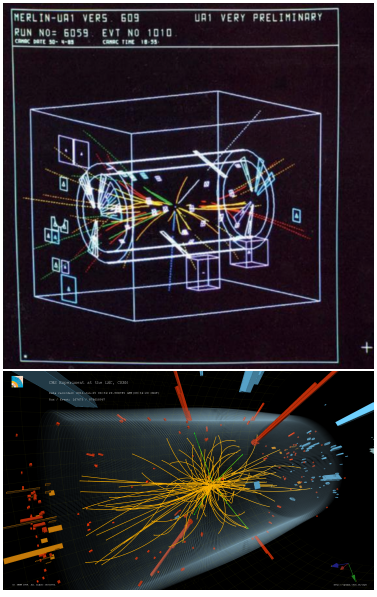






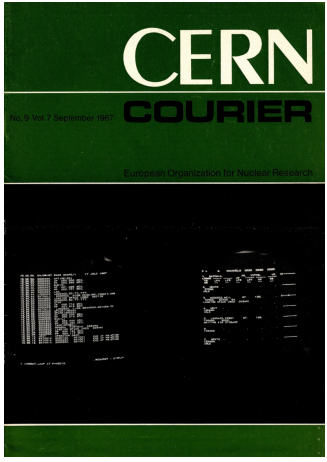
Detectors gets digitized; tracking algorithms get automated





CERN Courier special issues on computing

September 1967



https://cds.cern.ch/record/1728900?ln=en

March 1972

https://cds.cern.ch/record/1729464?ln=en

Divergence

inside a program. Logical relations between banks are expressed by including the address of one bank in the link-table of an other bank. For example, all tracks of a vertex-point are linked together by each track pointing to the next. Such a *data-structure* contains not only the numeric information but also logical information about the object it describes.

The program modularity is achieved by organizing the program into processors each having a well defined task. This task is *entirely* describable as a transformation applied to a data-structure in the dynamic store : some banks provide the input data to the processor and some contain the desired results. For a given application, a steering program is written to coordinate the operations of the processors needed. Any processor consists of at least one FORTRAN subroutine, its operation being invoked by transferring control to this subroutine. As a matter of internal organization, the processor may be divided up into the primary and several secondary subroutines. The programming of a processor has to observe certain conventions in order to be compatible with the HYDRA system. Precisely these conventions, which are the same through the whole program (indeed through all HYDRA programs) are responsible for the easy documentation and the good readability of the program.

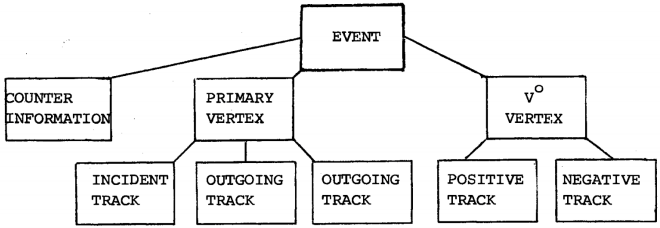
The processors are supported by the HYDRA system. Its services are requested with CALL statements much like the services of the FORTRAN system which are part of the definition of the basic language. In this sense, the HYDRA system is an extension of the FORTRAN language to provide - primarily - dynamic memory management facilities. Some languages contain these facilities in their basic definition, but the HYDRA-FORTRAN combination has two important advantages — the execution speed is that of a normal FORTRAN program, with very little overhead for the HYDRA system, and FORTRAN is a commonly accepted language. Because of the need for machine independence (so that the same programs can be used on a variety of computers) the processors for the new bubble chamber program, as well as the HYDRA system packages, have been written in ANSI FORTRAN which is the internationally accepted minimum requirement expected from anybody's compiler.

The bubble chamber programs of the HYDRA form will come into operation in 1972. They should help to tear down the walls that have sometimes threatened, on the data handling side, to separate physicists from computer specialists, or bubble chamber groups from each other and from physicists using other techniques.

Divergence

Example: High Energy Physics events are made up of vertices, every vertex has tracks associated to it. Also, to each event is associated a bank of information concerning electronic counter information to be used later. Assume the event to be a two-prong with an associated V°.

The pictorial graph for this event information is then



https://inis.iaea.org/collection/NCLCollectionStore/_Public/06/181/6181254.pdf

"Why physicists still use Fortran" (1983):
https://inis.iaea.org/collection/NCLCollectionStore/_Public/15/014/15014438.pdf

Reconvergence?