Saving the Software Heritage - the process -

Laura Bussi[1,2]

Roberto Di Cosmo[2]

Carlo Montangero[1]

Guido Scatena[1]

[1]Department of Computer Science, University of Pisa

[2]Software Heritage

Roadmap

- Prologue
- SWH: The Software Heritage initiative
- SWHAP: The SWH Acquisition Process
- SWHAPPE: The SWHAP Pisa Enactor
- Epilogue

Roadmap

- Prologue
  - where we frame our work in the larger picture of software history
- SWH: Software Heritage
- SWHAP: The SWH Acquisition Process
- SWHAPPE: Concrete support to the acquisition
- Epilogue

Which are the sources?

- Ideally [Mahoney, 2008]: running software
- "historians of technology must tinker with the things to discover the ideas which […] informed them" and
- historians of technology must "experience the software as users experienced it and hence analyze that experience critically".

- Actually, for *legacy* software: source code
- Hence, our work

Why is software history hard?

"Just as

the design of software

begins with

an analysis of

the activity to be automated...

...the history of software

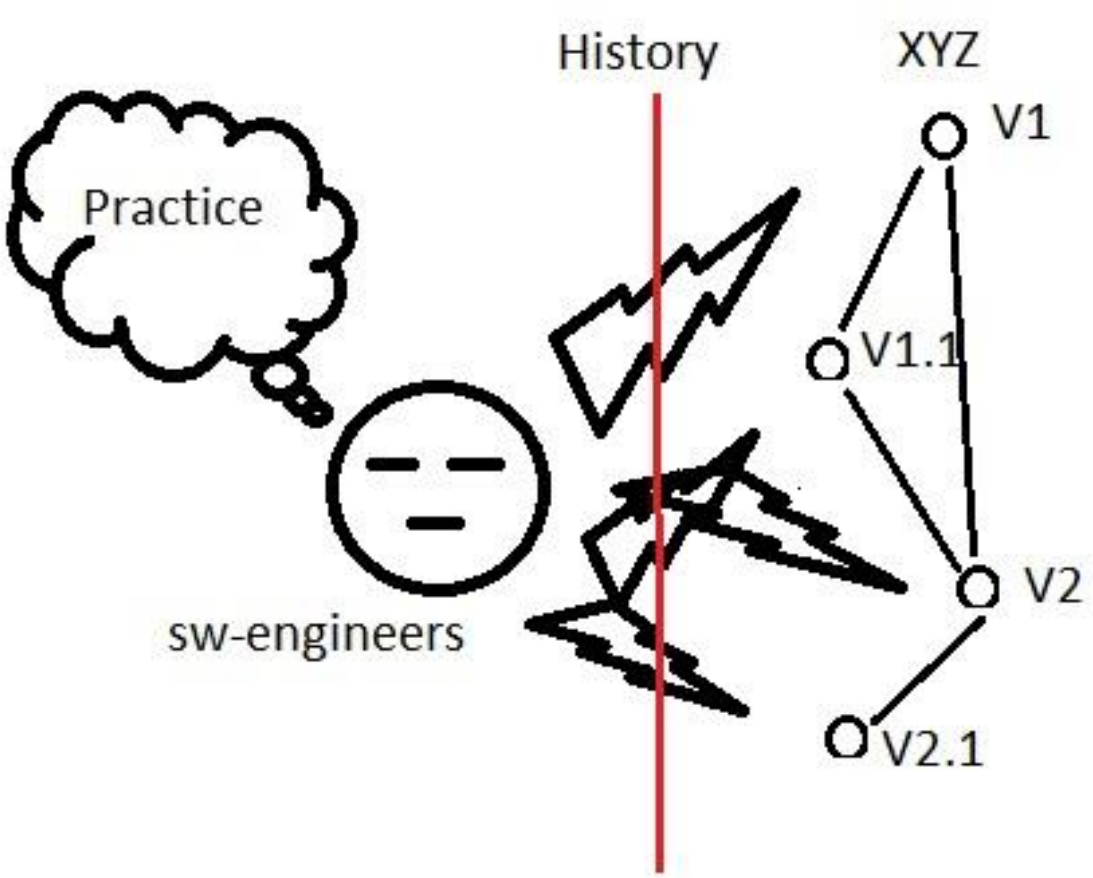
begins with

the *history* of what was done to understand how

the practice (of that activity) was translated into a computational model."

[Mahoney, 2008]

Recover first the version history



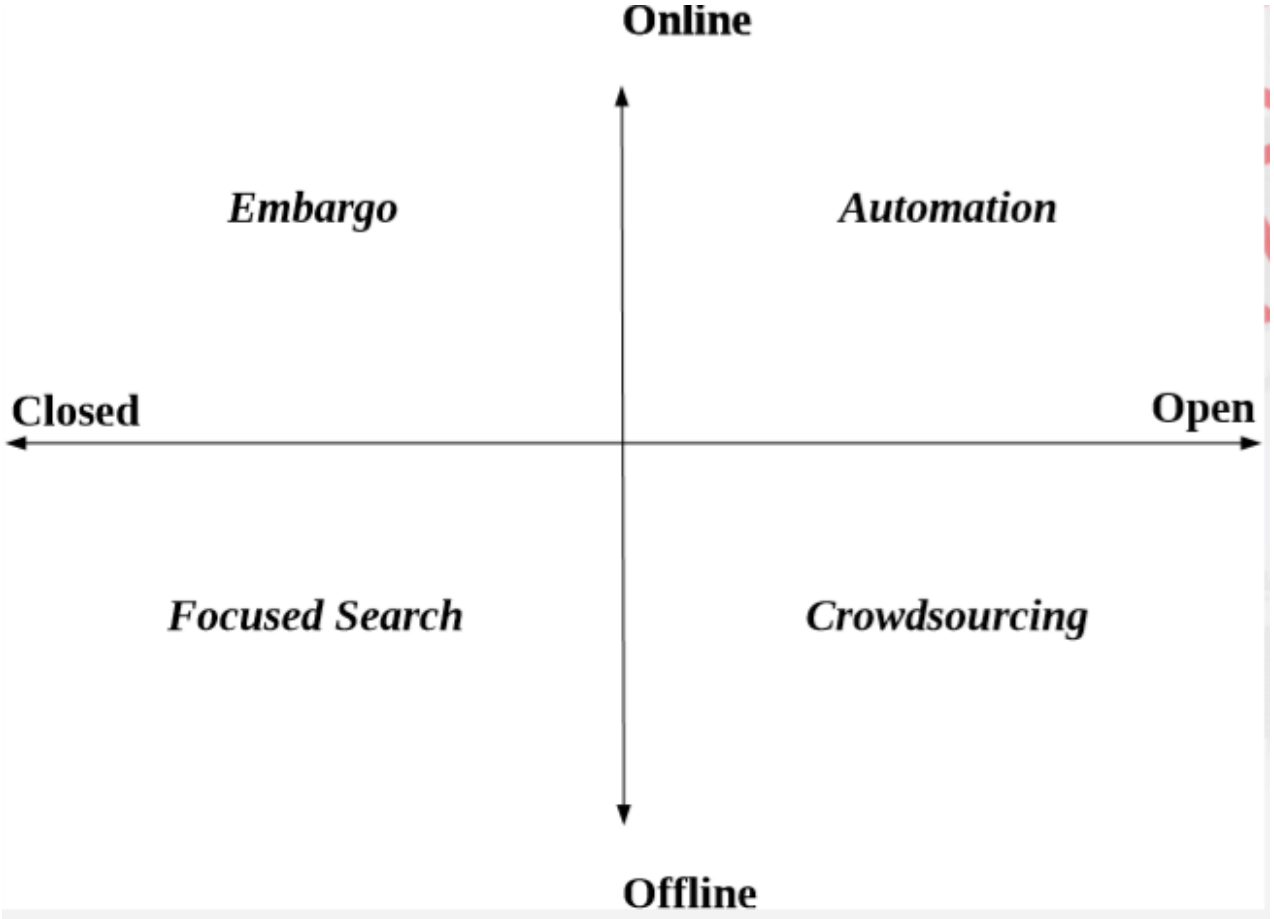
Roadmap

- Prologue
- The Software Heritage initiative
  - Where we frame our work in its own context
- SWHAP: The SWH Acquisition Process
- SWHAPPE: Concrete support to the acquisition
- Epilogue

Software Heritage

- Mission: build an infrastructure to collect, preserve and share the source code of all *available* software on the long term
- Requirements: ensure
  - Availability
    - *Open* architecture, software, and collaboration
  - Traceability
    - Unique intrinsic identifiers, directly computed from the source code
  - Uniformity
    - access through the same uniform API/web interface

Dimensions of source recovery



Software Heritage, as of Oct. 2019

- Harvested code
  - 90,860,137 projects
  - 6,317,723,261 source files
  - 1,394,141,708 *commits*
- Infrastructure
  - Main code repository at INRIA in Paris
  - Mirror in ENEA in Bologna – announced on Oct. 24
- Partnership with UNESCO
- Sponsored by Intel, Microsoft, Google, GitHub, ...

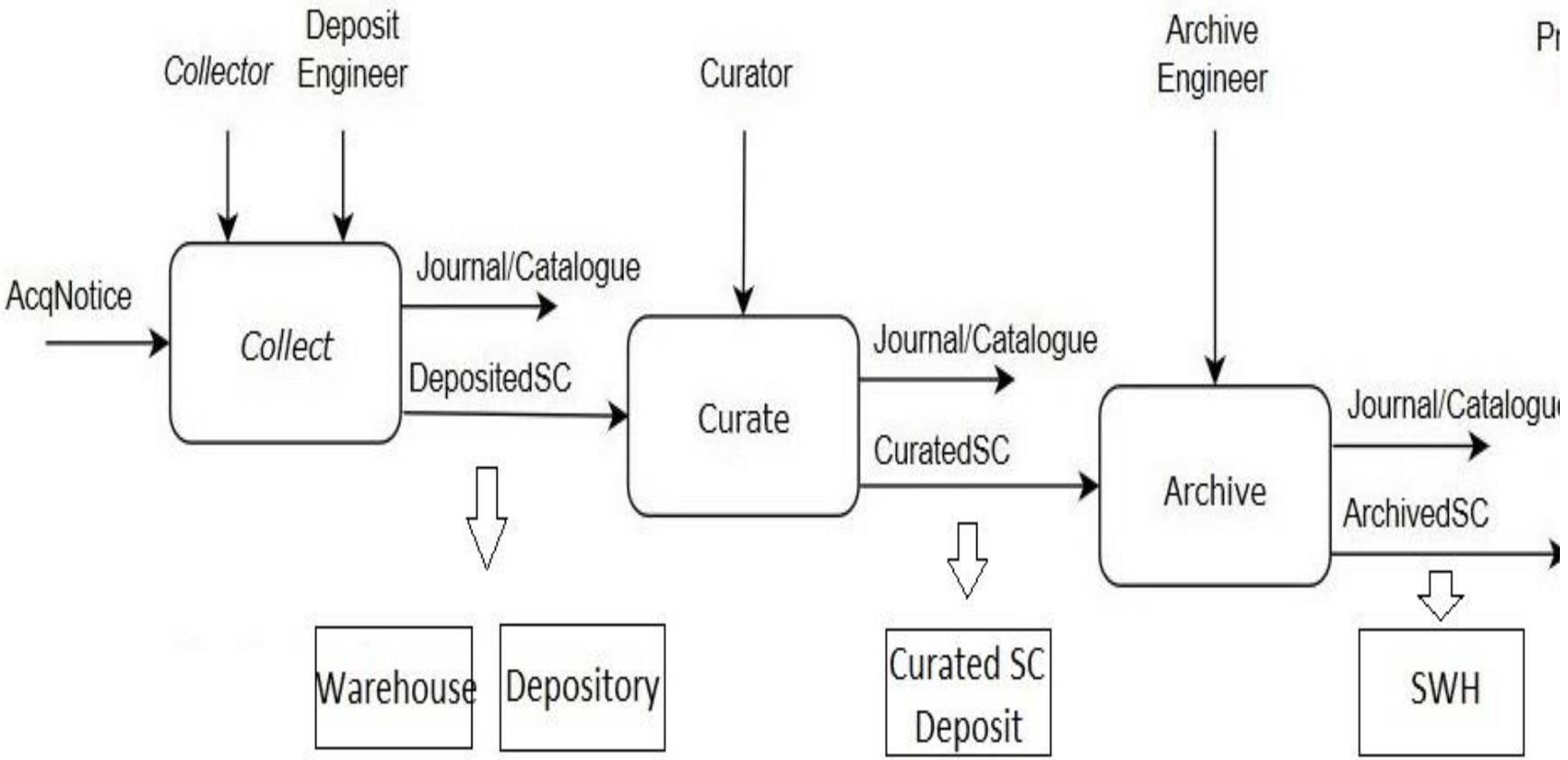
Roadmap

- Prologue
- The context: Software Heritage
- SWHAP: The SWH Acquisition Process
  - Where we sketch our proposal for software archaeology
- SWHAPPE: Concrete support to the acquisition
- Epilogue

A naïf view of archeologists' work

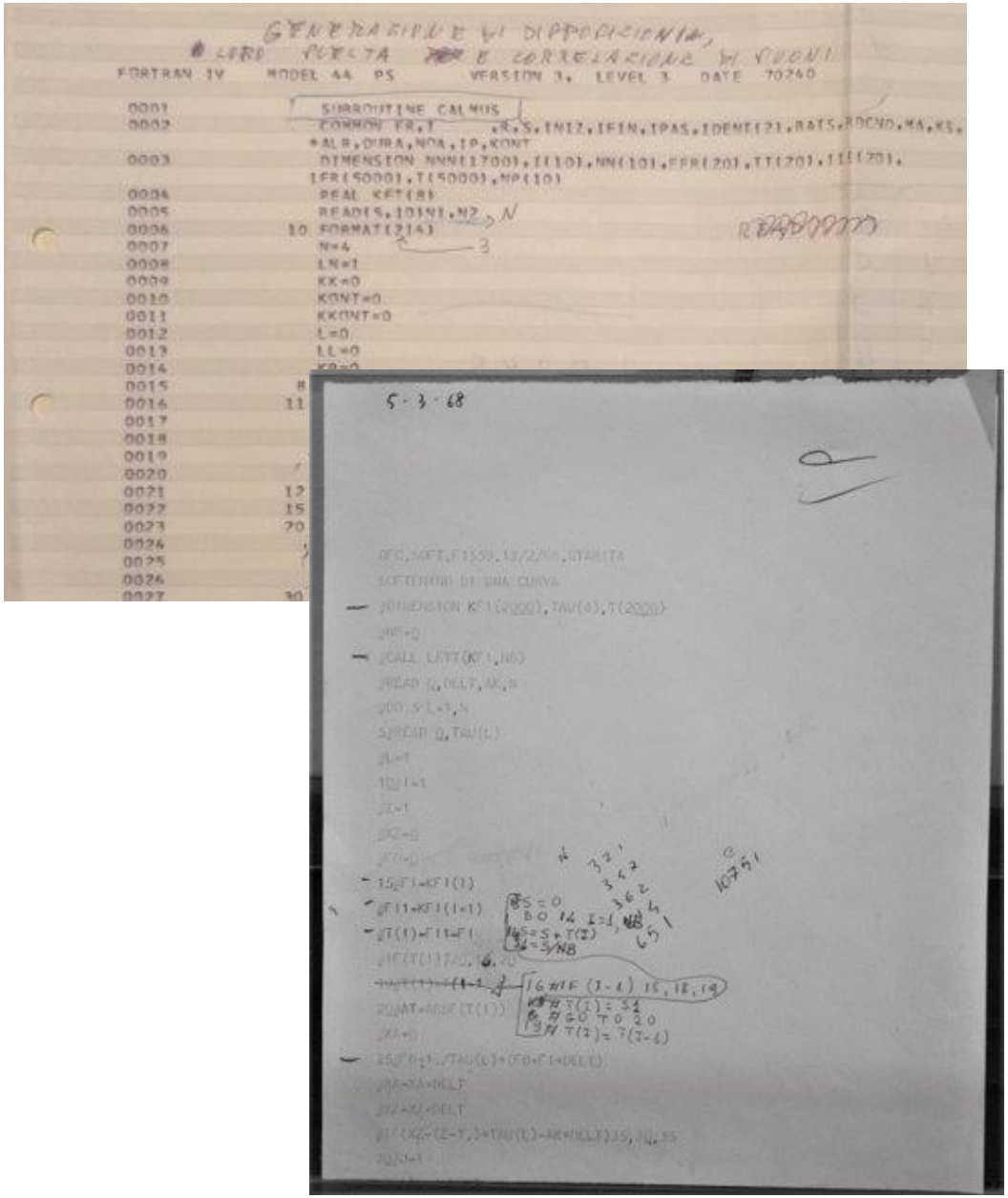
- First, on site, they collect and identify the finds.
- Then, in the museum, they safely store, curate, and exhibit them.
- Often, they come back on site for a new campaign.

SWHAP: an overview



The deposited harvest, so far

- Softi, a small numerical exercise, CEP Fortran (1968)
- TAUmus, TAU2 controller, IBM Fortran (70's)
- CCM, customizable memory manager, C++ (1994)
- OrbFit, astronomy library, FORTRAN (current)



Roadmap

- Prologue
- The context: Software Heritage
- SWHAP: The SWH Acquisition Process
- SWHAPPE: Concrete support to the acquisition
  - Where we talk about the SWHAP Pisa Enactor
- Epilogue

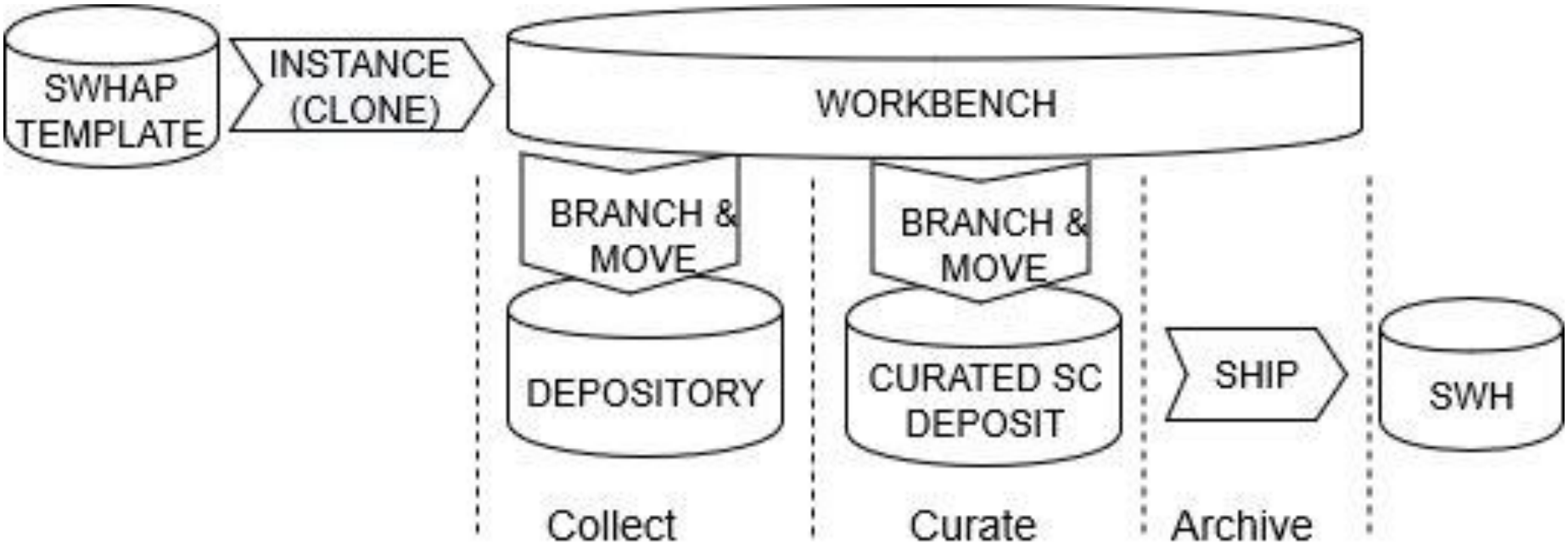
SWHAPPE: requirements

- Long term availability
- Historical accuracy
- Traceability
- Openness
- Interoperability

SWHAPPE: design choices

- The same tool all over the process
  - to reduce the learning effort and to streamline the process
- Git as the *revision control system*, to manage the source code history
  - Git supports *traceability* and *historical accuracy*, distinguishing between *author* and *committer*
- GitHub as the *collaborative* platform,
  - to host the virtual stores and working areas
  - to offer a web interface to access the saved information
- GitHub is archived in SWH, hence *long term availability* is guaranteed
- Both Git and GitHub are *open*
- Not the only choice, but very popular and active, and supported by Unipi

SWHAPPE in practice



- Infrastructure at https://github.com/Unipisa/SWHAP-TEMPLATE guide at https://github.com/SoftwareHeritage/swhapguide

SWHAP-SWHAPPE correspondence

- Warehouse: in the MSC in Pisa
  - most similar to archeology, we need to learn
- Virtual areas: repositories
  - In the ‘Unipisa organization’ space on GitHub.com
  - For the acquisition of code XXX:
    - XXX-Depository, to save the original finds
    - XXX, to save the curated source for SWH
    - XXX-Workbench, to support the process activities

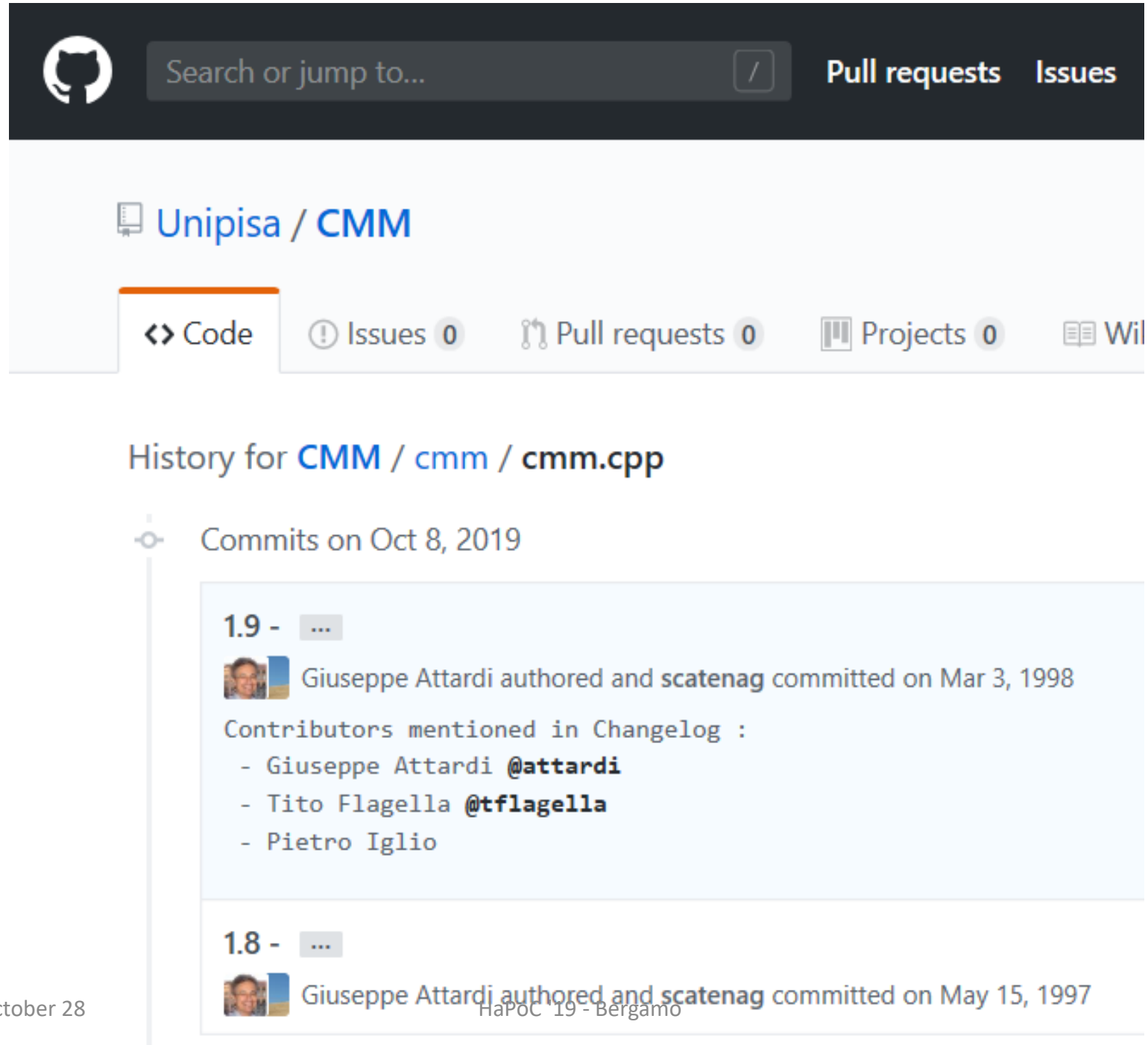
Some details: recovering the story

- For *each* version of the software ascertain
  - the *main contributing author*,
  - the *exact date* of the release of this particular version
- store these data in a dedicated metadata file
  - `version_history.csv`

Some details: recording the story

- Either manually
  - Committing the versions in the right order,
    - Using the info in the .cvs file
- Or automatically
  - Feeding the code and the .csv to
    - DT2SG: Directory Tree to Synthetic Git

      a SWHAPPE tool developed by Guido S.
- In either case you get historical accuracy


Search or jump to...
/
Pull requests
Issues
Unipisa / CMM
Code
Issues 0
Pull requests 0
Projects 0
Wil
History for CMM / cmm / cmm.cpp
Commits on Oct 8, 2019
1.9 - ...
Giuseppe Attardi authored and scatenag committed on Mar 3, 1998
Contributors mentioned in Changelog :
- Giuseppe Attardi @attardi
- Tito Flagella @tflagella
- Pietro Iglio
1.8 - ...
Giuseppe Attardi authored and scatenag committed on May 15, 1997

Roadmap

- Prologue
- The context: Software Heritage
- SWHAP: The SWH Acquisition Process
- SWHAPPE: Concrete support to the acquisition
- Epilogue
  - Where we draw some conclusions, and look at some open issues for future work

Conclusions

- SWH: a cooperative venture to
  - recover the past to preserve our heritage
  - share the knowledge to prepare the future
    - to guarantee scientific reproducibility
    - to make research software more valuable
    - to support research on software
- SWHAP: guidelines to this end
- SWHAPPE: a supporting infrastructure

A new *library of Alexandria* of source code

Open issues

- In the short term:
  - Increase the level of automation of the SWHAPPE support
- In the long term:
  - Acquire and internalize the procedures to store the physical finds, like listings, etc.
  - Acquire the means to streamline the transformation into digital form of the same
  - Critical review of the process
  - Porting of the process on other platforms

=> we are looking for cooperation and strategies to create a community

References

- M.S. Mahoney. *What Makes the History of Software Hard and Why It Matters*. Annals of the History of Computing 30,3 (2008).
- D. Spinellis. Unix History Repository. https://github.com/dspinellis/unix-history-repo (2017).
- UNESCO. Paris call - Software Source Code as Heritage for Sustainable Development. https://en.unesco.org/foss/paris-call-software-source-code (2018).
- Software Heritage. Home page. https://www.softwareheritage.org/ (2019).
- J.-F. Abramatic, R. Di Cosmo, S. Zacchiroli. *Building the Universal Archive of Source Code.* Comm. ACM (Oct. 2018).

Useful pointers

- The Software Heritage home page is at
  https://www.softwareheritage.org/
- The SWHAP guide, call to contribution, and mailing list can be found at
  https://www.softwareheritage.org/swhap/
- The SWHAPPE home page is at
  https://github.com/Unipisa/SWHAPPE
- The SWHAP acquisition catalogue is being updated at
  https://github.com/Unipisa/SWHAPPE