SWHAP:
the Software Heritage Acquisition Process
– Draft –

R. Di Cosmo, C. Montangero

April 4, 2019

Contents

Chapter 1

Preliminaries

1.1 Kick-off

Here is a recollection of the discussion we had on Friday, March the 1^{st} in the Department in Pisa, based on Roberto's "brain dump" drawn on the blackboard (see Figure 1.1).

Fondi vs Collezioni I found two terms in this respect: *fonds*[1] (The entire body of records of an organization, family, or individual that have been created and accumulated as the result of an organic process reflecting the functions of the creator[2]), and *artificial collection* (materials with different provenance assembled and organized to facilitate its management or use[3]).

The idea we have in mind here is that we should ensure *full traceability* of the recollection process, and for this we need to keep *the raw material* we come across, as we found it, that may be

- the content of a magnetic tape that contains a backup of a workstation, like the raw Doug Engelbart tapes put online by the Computer History Museum
- digital archives (.tar, .zip, etc.) that may contain binaries, sources, articles, documentation, tests etc.

Remark: in the following, we assume that this "raw material" is already in digital form, but of course we need also to handle the case where we need to recover the digital representation from printouts of a software system, blueprints of a system architecture, punched cards, floppy disks, etc.

Looking in detail at the definitions that Carlo found and reported above, it seems that this raw material may be called a "fonds" if it comes from a same source (e.g. the archives of the CS department, or the archives of CNR), or a "collection" if it is already the result of a selection/curation

[1] This is a French word, used in the archivist community, from the expression "'respecter les fonds"', a principle dating back to just prior the revolution of 1789: it prescribes that the original organization of the fonds should not be changed by the archivist.

[2] https://www2.archivists.org/glossary/terms/f/fonds

[3] https://www2.archivists.org/glossary/terms/a/artificial-collection

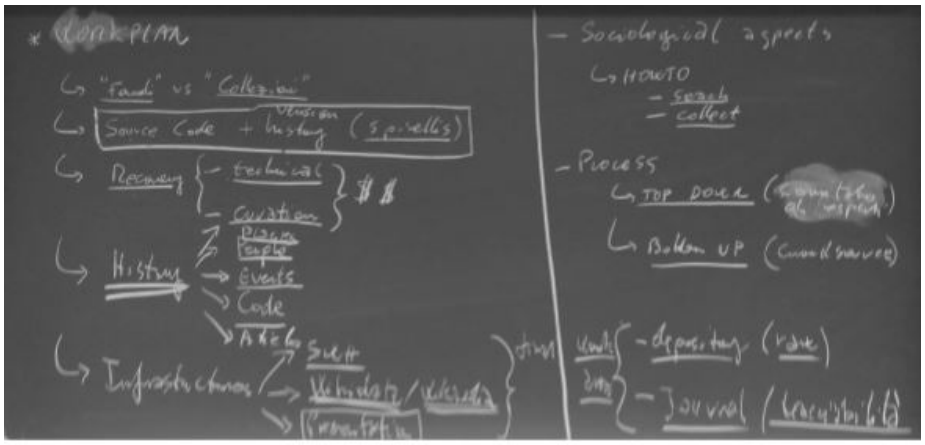

Figure 1.1: The Blackboard

effort (e.g. a selection of particular source codes preserved by a collegue for some specific reason, or a thematic archive of source codes), while what I (Roberto) had in mind was more of separating raw material from curated material.

Source Code + Version History This is one of the principal tenets of the SWH approach: *source code*, rather than executables, for its value in terms of readability and understanding of the operation of the system; *version history*, as an analogous of the *strata* of an archeological site, providing insights on the development process of the software.

Recovery There are two facets in the process:

1. The first one is technical, and is concerned with transferring the software from its original base to a location, that we can call "the depository"[4], where it is easily accessible in digital form. This process may be particularly hard and costly when the original software is not already accessible in the web:[5] indeed, it may involve the discovery of the coding scheme of the stored information and the recovery, may be even the building anew of the reading device, when the source code is stored on obsolete data supports.
2. The second facet is conceptual, and is concerned with understanding what's in the fonds, identifying and separating the software units/components/projects that are present, and organize them so that they can then be transfered in the SWH Archive. In particular this involves:
   (a) identifying the different versions of a software, with their release names/versions
   (b) identifying the authors of the different versions
   (c) import each version into a single git version control system that will represent all the history of the development of this particular piece of software

[4]See https://www2.archivists.org/glossary/terms/a/artificial-collection for a definition
[5]This is not to say that even if this is the case the process is always simple.

(d) deposit the result into the SWH Archive

- using a moderated portal like HAL[6], for software that has only one version
- using the "save code now" functionality if there is more than one version

For steps 2a to 2c, a good reading is the work of Diomidis Spinellis on the history of Unix

These two activities are two sequential phases of the recovery process, even when the software is available in digital form.

History Besides the source code made available in the SWH vault[7] many other information about the code are interesting and should be preserved in suitable infrastructures:

1. Places
2. People
3. Events
4. Documentation
5. Articles: I have seen cases in which articles are cited in the ReadMe in the vault. Copyright issues may interfere with the open access attitude.

I would like to add also

1. Usages, since they provide evidence of the impact of the software.

Part of these information may have been made available on line by the authors themselves or by previous researchers and/or fans of the software. Is there the need of a standard also for this recovery process? Surely, at least of the indication that open infrastructures should be used (see below). ???

Infrastructures Different supports are needed for different purposes:

1. SWH, to fulfill the primary goal of the project, that is, to preserve the software base
2. Wikidata/Wikipedia, to preserve the related historical information in a machine accessible way, for further generic dissemination of the knowledge embedded in the software. Wikidata has a normalised property for the SWH identifiers, so it will be easy to have links into Software Heritage from Wikidata.
3. Open Access repositories for articles, books, technical reports, etc.
4. Presentation tools/mechanisms, to provide the relevant information available in the previous archives in a way tailored to specific situations, like public events, lectures etc. A good example is the work that has been done for sciencestories.io.

[6] https://hal.archives-ouvertes.fr/
[7] https://www.softwareheritage.org

Process Besides the technical issues related to software recovery, there are managerial ones, related to the best usage of the available resources, i.e., to the discovery and prioritization of the software to be salvaged. There are at least two possibilities:

1. Top-down, when the identification of the software to be saved is made by experts, convened to cover a specific area of interest
2. Bottom-up, when the software is recovered as long as it is offered by interested partners

As it often happens, the winning approach may well be the *sandwich* one, which combines the two, with the top-down part organizing and monitoring the bottom-up part, as it happens in Crowd-source.

Verifiability We need to preserve the original materials and to allow checking a-posteriori the recovery and curation process, and eventually fix any information that was not correctly and faithfully handled. So one must foresee the existence of

1. a *Depository*, that is, a separate archive where the raw initial information is preserved, with at least the same safeguarding characteristics of the SWH Archive.
2. a "journal" (or "log") of the operations performed in the recovery process, for traceability. The journal may be preserved in the Depository itself.

Chapter 2

A first attempt á la IDEF-0

This section elaborates Roberto's schema of March 21^{st}:

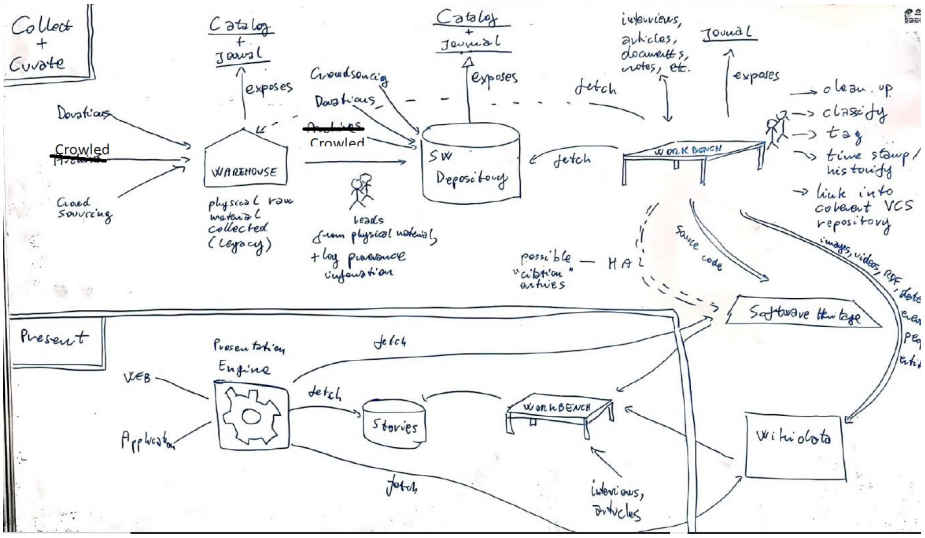

Figure 2.1: Roberto's schema

2.1 Notation

IDEF-0, una notazione molto antica ma efficace.

- Le scatole sono attività, che trasformano documenti da destra verso sinistra.
- Quindi, le frecce verso destra rappresentano gli input dell'attività, e le frecce da sinistra gli output. Spesso gli output diventano input.
- Le frecce dal basso rappresentano gli strumenti caratteristici dell'attività.
- Le frecce dall'alto rappresentano i *ruoli* (leggi: insiemi di competenze) coinvolti nell'attività: i ruoli sono assegnati a persone (almeno in prima

istanza, poi si potranno mettere automi più o meno intelligenti), e ovviamente la stessa persona può svolgere più ruoli, anche in attività diverse.

- Ho esteso la notazione per accogliere la nozione di classe astratta, denotata dall'avere il nome in corsivo (à la UML - sottolineato quando scritto a mano, secondo la tradizionale convenzione tipografica). Ad esempio, il processo Collect è astratto, come pure la Notice in input, lo strumento Warehouse e il ruolo Collector.

2.2 Process elements

There is a lot to recover from section 1.1!

AcSCnotice : Documento Questo documento (AcquirableSourecCode Notice) riflette l'assunzione che il processo di acquisizione parte con una "notizia di reato": sto assumendo che ogni processo di acquisizione parta da un documento - una lettera di un donatore (esempio: la lettera di Attardi relativa al suo garbage collector, poi utilizzato da Google per Java), una call per raccogliere una certa categoria di SW (tipo quella che stiamo preparando per "5 software per 5 decadi"), una qualche dichiarazione d'interesse per (una parte di) un archivio SW - che in qualche modo identifica il SW da salvare.

***Collect* : Attività** Questa si presenta concretamente in varie forme, una per tipo di SW da raccoglliere. La figura **??** dettaglia (che parolona!) quello per il SW archived (ho usato :: per indicare la superclasse).

Il fatto che ci sia già a questo stadio l'intervento del PresentationDesigner riflette la mia comprensione della freccia dal workbench (in alto) alla Warehouse: ci può essere già una prima visura del materiale, magari solo per valutarne l'interesse per una presentazione.

CollectArchivedSC : Attività Questa attività, specializzazione dell'attività Collect, si ha quando il software che si vuole raccogliere è già disponibile in un repository moderno, tipo Git Lab. L'attività ha inizio con una specifica Notice, che contiene le coordinate necessarie per reperire il software nel repository origine, da cui l'attività deve prelevare i documenti. La figura **??** specifica solo gli elementi astratti di figura 2.2: la notizia è relativa a un SW Archived, la Warehouse è un Git Lab, e il Collector deve avere le relative competenze.

Una conseguenza di questa scelta è che eventuali dati nonSW sono comunque memorizzati nel Git Lab, su indicazione del PresentationDesigner, a sua disposizione nei passi successivi.

Il prossimo passo è di avere una descrizione del processo CollectArchived, almeno a parole. Qui siamo a un livello di dettaglio per cui forse si può pensare a una notazione diversa (diagrammi di attività UML, Business Process Modeling Notation, BPMN,...).

***Collector* : Ruolo**

Curator : Ruolo

***SC* : Documento** Ciò che si vuol salvare in primis (Source Code). Si presenta in diverse forme, che riflettono l'avanzamento del processo:

OriginalSC Il codice sorgente che si trova nell'Origin (vedi); è conveniente pensarlo di diversi tipi, sia rispetto al tipo di acquisizione

pulled in cui l'acquisizione parte da un centro, più o meno legato a SWH, interessato a raccogliere una data classe di SC;

pushed in cui l'iniziativa parte dalla periferia, sia individuale sia di gruppo,

sia rispetto all'accessibilità originale

online

offline

Delle quattro combinazioni possibili dei due tipi, pulled/online è quella realizzata, come CrawledSC, da SWH finora.

DepositedSC Ciò che si ha a disposizione una volta terminata la pertinente attività Collect, conservato a futura memoria per analisi approfondite, anche sulle scelte fatte dal Curator in prima istanza.

CuratedSC Questo è l'obiettivo primario di SWH: il codice sorgente di cui alla relativa Notice, a disposizione del pubblico nel caveau, con la sua evoluzione ricostruita a partire dalle informazioni originali, sotto la responsabilità del Curator.

PresentedSC Questo è l'obiettivo secondario di SWH, realizzato per caratteristiche particolari del codice (importanza dello stesso nella storia del SW) o per pressioni particolari di un committente (salvataggio fatto in occasioni particolari, come il 50-nario del corso di laurea in Scienze dell'Informazione): al codice sorgente salvato nel caveau come CuratedSW, si associa una pagina web che raccoglie documenti storici e illustrativi, guide d'uso ecc.

Origin **: Strumento** Il sito dove è raccolto il materiale originale da importare, che può essere:

online

offline

PresentationDesigner: Ruolo

Warehouse : Strumento Un repository che viene usato come appoggio temporaneo per ogni tipo di documento. Può essere esterno alle risorse di SWH o introdotto esplicitamente in qualche fase del processo.

2.3 Model

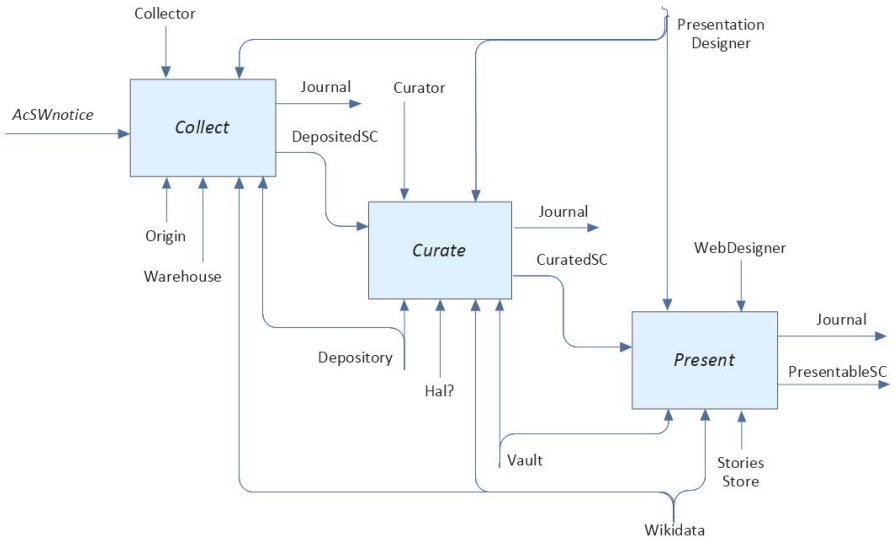

Figure 2.2: The overall process

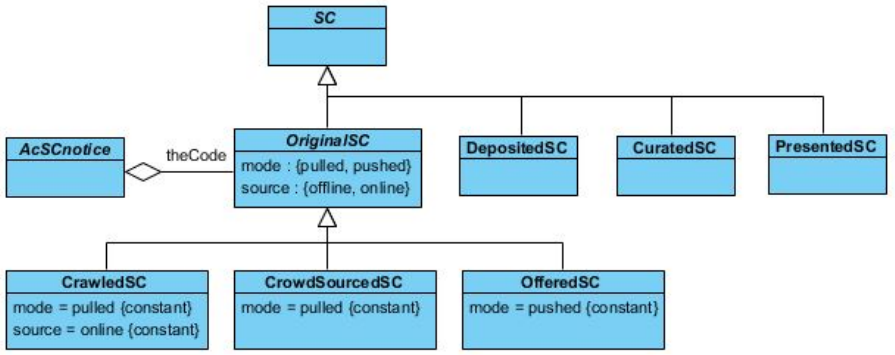

Figure 2.3: Class diagram - SC types

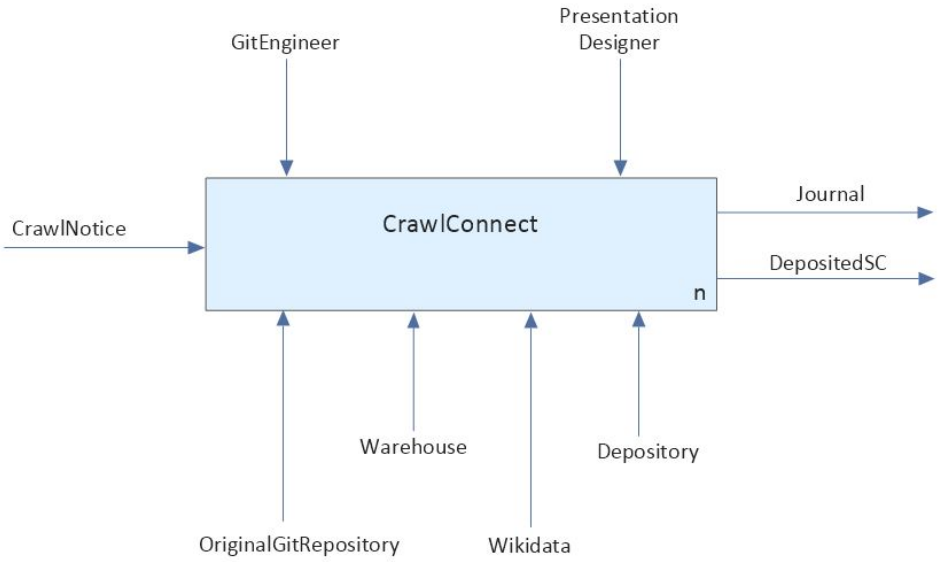

Figure 2.4: The CrawlCollect activity

List of Figures

List of Tables