

The Books of Software History

A prospective SCC subproject

PSN 4/26/06

Preliminaries

- I volunteered to compare current CHM holdings with the ~600 publications suggested by ACM members as classics/favorites.
- Wrote a perl script comparing the two lists, obtaining an initial result (mostly right but needs a bit more work) [..findACM.html](http://findACM.html)
- Then Bernard asked to consider the question of “collecting” the more significant books in an augmented sense:
 - creating annotated entries with author bios, context of writing, significance....
- And then Paula Jabloner asked for a write-up of the activity.

The result follows.

- Why “collect” the books?
(in the augmented sense)
- What is involved?
- How can it be done? (partial)

Why collect the books of software history?

- The SCC “collects” software for:
 - the software research collection
 - as a source for future software exhibits
- “Collecting” software means:
 - Preserving (and sometimes reviving) the bits
 - Assembling contextual information about the systems and their place in the history
 - Related documents, accounts of the developers,
- but
 - the story of software cannot be told only in terms of individual systems
 - “Collecting” the significant publications and contextual information is also necessary.....

Why is collecting significant publications and contextual information also necessary?

- A useful history of any type of tool must convey the history of the underlying ideas.
- For physical tools, including computers, the history of ideas emerges naturally from landmark/typical examples.
- But, for many software areas, major publications are themselves part of the history.

Examples of Software Areas

- Theory of Computation
 - All artifacts are pubs (except algorithm libraries)
- Software Engineering
 - Possible beginning: “The Mythical Man Month”
 - Prescriptive pubs on process not realized in code
- Data Bases
 - Ted Codd’s 1969-1970 RDB papers are critical pubs
 - No realizations until several years later
- Compilers
 - History = landmark systems +
bridging pubs (abstract, extend, disseminate the ideas)
 - e.g., “Programming Languages and their Compilers“, &
“the dragon book” (Still in print and still a bible)

What does “collecting” the books involve?

- **Selecting significant publications:**
 - Build initial list for some areas, list other areas.
 - Focus on older books, because obtaining contextual information for them will become increasingly difficult
 - Circulate initial list, solicit comments & additions
- **Obtaining a context entry for each →**
- Helping to obtain their editions for the CHM
- Providing online versions
 - when copyright arrangements can be made

Context Entry

Content

Sources

Author bio Authoring background & context	Authors, author home pages, colleagues, oral histories, IEEE Annals
Content summary Significance in software history Discussion of similar publications	Authors, and (probably older) experts in field
Reference list	anyone
Links to material on other pubs by same author	??

How obtain entries?

- First, establish entry guidelines,
 - draft a few and solicit comments & suggestions
- Then collect entries by “division of labor”:
 - Ask academics, and other reliable area experts to “adopt a book” or two or three
 - professors can solicit student help!!

Comments? Questions? Suggestions?