S. Bødker, M. Kyng, and K. Schmidt (eds.). *Proceedings of the Sixth European Conference on Computer-Supported Cooperative Work, 12-16 September 1999, Copenhagen, Denmark*
©1999 Kluwer Academic Publishers. Printed in the Netherlands

Moving document collections online: The evolution of a shared repository

Randall H. Trigg, Jeanette Blomberg, Lucy Suchman
Xerox Palo Alto Research Center
{trigg, blomberg, suchman}@parc.xerox.com

Abstract. This paper reports on a work-oriented design project concerned with the question of how to migrate shared, workgroup document collections currently kept on paper online. Based in a civil engineering work group, the focus of our project is a document collection called the "project files," a heterogeneous mix of documents that serve as an ongoing resource for the group during a project's course as well as an archival record at its completion. We describe the dynamics of the standardized classification scheme in use for the project files, existing practices of document filing including routine troubles, and the prototype developed to move the project files online. The latter includes a configuration of hardware and software along with associated practices of document scanning, coding and search. We conclude with some reflections on the difficulties of maintaining alignment across paper and digital media in the migration to online document collections, and with a summary of the questions posed and answers provided by our project.

Introduction

The organization and management of documents is viewed as a persistent problem for office workers of all kinds. Researchers have noted that people's offices and work areas frequently appear (to them and to their colleagues as well as to researchers) cluttered with documents. These documents often claim most of a work area's horizontal surfaces, and are typically arranged in more and less meaningful piles (Malone, 1983, Nardi & Barreau, 1997). While the potential of computational technologies to ease the burden of office filing and facilitate later document retrieval seems obvious, it largely remains to be realized (Celentano et al., 1991). In this paper we report on a project to explore the requirements for creating online repositories of scanned documents within a workgroup.

Our project is focused on what we call *working document collections* that are maintained jointly within a project team and serve as shared resources for the group's work (see also Blomberg et al., 1996). Working collections occupy a niche between active documents currently in use at any given time and those stored in an archive, with the file cabinet being the typical example. Some have suggested that the documents stored in file cabinets or scattered throughout offices have limited value as they are seldom retrieved for later use (see for example Kidd, 1994). The argument goes that once a document's immediate use has passed (e.g. to convey information, to denote that an action has occurred, etc.), its value is significantly reduced. In such cases the retention of the document is motivated more by difficulty in deciding to discard it or by the archival requirements of the organization than by the anticipated value of the document in some future transaction. Our research suggests, however, that in many organizations collections of documents are deliberately retained for their potential value in the day-to-day operations of the organization. Far from losing their value over time, these documents are critical to the effectiveness of workers and work groups. In such organizations the problem that confronts workers is how to organize document collections so that on those occasions when a document is needed, it can be found.

Our current project is part of a larger research program in *work-oriented design* (Ehn, 1988; Greenbaum and Kyng, 1991; Kyng & Mathiassen, 1997). Our approach comprises workplace studies closely integrated with the cooperative development of prototype systems, meant to exemplify new and useful work practice and technology configurations. We combine in situ interviewing, workplace observations and video analysis with design interventions that engage a range of representational artifacts (from paper mock-ups to working prototypes). Throughout the course of a project we move back and forth between our field studies and our design activities looking for opportunities to introduce design ideas and artifacts into the work environment.

At present we are engaged in a collaborative research project with a team of engineers employed by a state Department of Transportation in the U.S. (called here "the Department").[1] The team is involved in a bridge replacement project undertaken as part of a larger effort to bring the area's existing roads and bridges up to earthquake safety standards. Although our research has looked at various aspects of the work of these engineers (Suchman, 1998; Suchman, 1999; Suchman et al., 1998), our design efforts have focused on the work of filing and retrieving a collection of workgroup documents referred to as the *project files.*

Our prototype system for the engineering design team had its genesis in an earlier case-based prototyping project at a Silicon Valley law firm (Blomberg et al 1996).

[1] The engineering team was initially composed of six civil engineers but has grown to more than 20 The prototype system we developed is intended to support the work of the entire team

The previous prototype also acted as an online, scanned repository for a working document collection, and we learned several lessons from that experience that have informed our current design:

- *The problems in requiring extensive coding as a prerequisite for the addition of a document to the collection.* The only classification information that we had about the attorneys' documents was the name of the file folder in which the physical document would be placed. Attorneys made it clear that they had no time to input additional metadata.
- *The value of page images for document browsing and retrieval.* We learned from the users of our earlier prototype that a range of scaled reductions of page images were valuable for different purposes: thumbnails for browsing the results of a search, intermediate reductions for browsing the pages of a document, and larger, readable reductions for viewing single pages.
- *The importance of hybrid search.* Though our searches were primarily restricted to the OCR text of the scanned documents, we saw the potential value in providing multiple, alternative search strategies in a single interface.
- *The need to reproduce the current physical organization of the document collection in the online interface.* As one view among others, a representation of the filing scheme used in the paper collection serves both as a familiar, transitional rendering from paper to digital media, and as a useful option for filing and search in its own right.

Our focus at the law firm was on document search and browsing, rather than on practicalities of document scanning and coding. In addition, the prototype that we developed was never fully integrated with the firm's infrastructure, making it impossible to access the collection from multiple workstations. Our primary goals in redesigning and reimplementing the law firm prototype in our current project were to move to a WWW-based interface that would allow workgroup access from diverse platforms, and to develop fully the practices and infrastructure necessary for scanning and printing as well as search. Indeed, we think of the current prototype as more than a document repository with an online interface; it is a collection of artifacts (workstation, scanner, in-box, etc.) and associated practices of scanning, coding, browsing, retrieval, and printing (see Figure 1).

The creation and maintenance of project files is a basic requirement for every engineering project at the Department. The *Project Development Procedure Manual* explains that these documents represent a partial record of the activities of the group, "... that document project decisions, and that would be useful (or required) to develop a subsequent project." Project engineers make reference to these documents throughout the project's life. Initiated by the design team, when a project moves from design to construction some number of the documents are copied and made available to the construction team. At the conclusion of all stages of a project, a select group of documents from the project files are assembled for the Department's permanent archives.

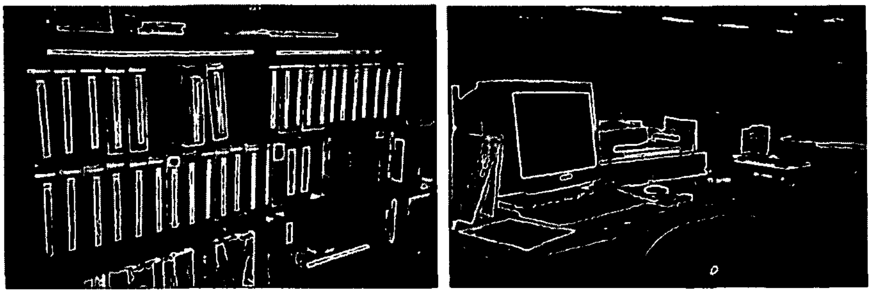


Figure 1. Project file binders and prototype scanning station.

Project file documents are a heterogeneous collection of letters, memos, reports, minutes of meetings, working design plans, and the like that originate both from inside the Department and from outside sources such as citizen groups, consulting firms, and other governmental agencies. The project file acts as a shared resource for team members and on occasion is referenced by other groups within the Department needing access to project file documents.

All project file documents are currently kept in hardcopy and stored in three-ring binders within the project team's work area. Our collaborative project with the bridge replacement team is to develop an online project file, accessible through the World Wide Web.

The basic design goals for the prototype are threefold:

- To require minimal overhead for entering documents into the online repository.
- To provide for incremental, modifiable coding of documents at any time.
- To provide multiple resources for document search and viewing.

In the remainder of the paper we describe the project and the lessons learned to date. We begin with a discussion of the work of document coding, including the dynamics of document classification and current practices of document filing. With that background we describe the development of an online project files prototype, including the design of working practices for scanning, document coding and document search as well as the configuration and development of associated hardware and software. We conclude with some reflections on the difficulties of maintaining alignment across paper and digital media in the migration to online document collections. Finally, we offer a summary of the questions posed and answers provided by our project.

The Uniform File System

Engineering teams are responsible for the maintenance of project files and are instructed to file documents according to a standard, organization-wide filing system called the Uniform File System or UFS:

> The Uniform File System is to be used by all Caltrans projects – regardless of size or type of project. The originating unit should start the file system as soon as preliminary studies can identify the project (*Project Development Procedure Manual*).

The UFS is a three-level hierarchical classification system in which an exclusive numerical code is assigned to each document, which is then placed in the corresponding location in the binders. This apparently clear structure masks, however, what is in fact a diverse, cross-cutting and sometimes conflicting set of interests in the documents to be filed. The latter include the type of document (correspondence, reports, maps and the like), the source of the document, to whom it was sent, the project stage to which the document relates (e.g. project approval, design, construction), and the main subject matter covered by the document. To address this problem the Department's procedure manual states that the source of the document should be the primary determiner of its UFS code, on the argument that "[m]any letters and reports cover more than one project issue. Consequently, items will be classified and filed according to the source that generated them, rather than by subject." The standard UFS framework has been relatively stable, with organization-wide changes occurring on the order of every five years. It is recognized, however, that minor modifications and elaborations to the UFS will be required by project teams in response to the particular requirements of a given engineering project.

The Project Development Procedure Manual suggests that "[t]he PE (Project Engineer) should use personal discretion when creating sub-categories for filing purposes." The changes to the UFS framework that we have observed include elaborating existing categories by naming particular entities, creating subcategories where none existed, and adding new, high order categories.

Elaborating existing categories: As an example, the UFS sets aside the 300 category for General Correspondence. Within that category, code 330 denotes Correspondence with Federal Agencies. The bridge replacement team has elaborated the UFS scheme by specifically naming the six federal agencies with whom they correspond, first as hand annotations and later as typed entries in a revised version of the UFS for their project. Similarly code 351, Correspondence with Cities, has been specified to include the five cities within or adjacent to the project area.

Creating new subcategories: The UFS may also be modified to reflect changes in the situation of an ongoing engineering project. For example, when the bridge replacement project first began, the Bay Area Transportation Authority (BATA) did not have jurisdiction over this particular project. BATA's authority was later extended to include oversight of funds set aside for the bridge replacement project. The engineering team then found it useful to add a new category, 134, specifically for

BATA resolutions. At the same time they elaborated code 353, Correspondence with Areawide Agencies, to include correspondence with BATA (but not resolutions).

Adding high order categories: Perhaps the biggest change that we have observed is the addition of a new category where none existed before. The 700 category was thought to be necessary because much of the project work in this case was being done by outside consultants. This level of contracting-out is a relatively new phenomenon at the Department, and the organization-wide UFS has not yet reflected this change. The bridge project team initially added the 700 category for all documents relating to their work with consultants. Later in the project, however, the 700 category was redesigned to cover only task order agreements with the consultants. All other correspondence with consultants was moved to a new category, 314, under the General Correspondence category.

Each of these changes to the UFS has had implications both for current filing practices and for the design of our prototype.

Filing practices using the Uniform File System

Our interest in understanding what would be required to move the project files online has led us to look closely at the bridge team's current filing practices. On one occasion we arranged to sit with Dave, the Senior Project Engineer, while he assigned UFS codes to documents destined for the project files. In the course of filing documents Dave commented to us about particular difficulties he was having in deciding which UFS code to assign to particular documents:

> D: One thing I'm noticing as I'm picking out where I want to store these things you know like for instance this one right here, it's our letter to the FHWA [Federal Highway Administration] regarding consultation for the endangered species act. So there's a permit involved, environmental is involved, the federal, FHWA is involved, external agencies. So there's all these categories it could conceivably go under and I have to pick one. Then I have to go back and maybe search for, because maybe I wasn't thinking, the next time when I'm looking back. So that's why it would be really cool if you could enter these things like you said, you could have a date, or a title, or a subject or keyword or whatever. So that's why I think it would be really handy, because I'm sitting here and I'm going, well, correspondence to federal agencies, yea, that's the one I think it is But it could easily be thrown underneath permit, and certainly my assessment may be different than the guy next aisle over.

Dave here is experiencing common troubles of filing, involved in deciding to which of several possible categories a document should be assigned. Several alternative UFS codes seem equally plausible, but insofar as the document in hand will go in only one location in the binders, he must pick only one code. He also recognizes that his choice will have consequences for his ability to find the document later on. Finally Dave is concerned that his choice of UFS may differ from "the guy the next aisle over," adding to the uncertainty that he or others in the workgroup will be able to find documents once they have been filed.

Dave also struggles in assigning codes with the conflicting logics of the Uniform File System. As we mentioned earlier, the UFS orders documents variously according to, among other things, the stage of a project and the topic covered in the document:

> D: (looking through UFS documentation) OK now I don't see what I thought I was looking for. So uhm, I guess I would stick it under uh Floodplain Evaluations. What was the other spot? Drainage is usually done during the design phase and we're not there yet. So that's why I would pick uhm, but see 231 is Draft Environmental Document which is pretty vague. So I'll never find it. It's just not going to happen. I'd probably be more inclined to stick it under Drainage even though that's not where it belongs? So that's what I'm going to do. I'm probably not doing it right but that's what I would do. So this is why your system would be nice.

In this case Dave finds it difficult simply to locate the UFS category he is looking for within the scheme itself, as he has to flip back and forth through the pages of the UFS documentation to find the appropriate code. Dave is also confronted with the problem of a misalignment between the normative chronological order of engineering projects and the topical concerns of the document in hand. While the bridge project is still in the environmental assessment stage at this point, the document is about "drainage." As Dave explains, the UFS code that deals specifically with drainage is found later on in the scheme, under a category that concerns the design stage of a project. As a result he is faced with the question of filing the document "where it belongs," or alternatively where he is most likely to look for it later on.

Extending the options for document coding

At this point in our project we could see clearly that putting the project files on line would ease the bridge replacement team's reliance on the UFS classification scheme as their primary means of document retrieval. Our observations and discussions with members of the team convinced us as well of the potential value of a system that would allow them to assign multiple, heterogeneous property values to project file documents. We first explored the possibility and practicality of assigning new metadata classes to project file documents by mocking-up a paper coding form with fields for UFS code, date, keywords, and document type.[2] We again sat with Dave as he used our coding form to code project file documents, asking him to add keywords and document types as he saw fit. This provided us with a tentative set of keywords and document type categories to include in an online coding form.

To explore the feasibility of using our online coding form, we asked Andrea, an engineer working on the bridge project to code documents using the form. As with our original paper coding form, we allowed new keywords and document types to be added as needed. Andrea's experience using the coding form led us to rethink the

[2] We originally hoped to support the automatic processing of the paper coding form but so far have not been able to do so with the technologies available to us.

form's design. In particular, we observed that Andrea had difficulty in locating relevant keywords for a given document. She would peruse the list of keywords, and failing to find what she was looking for she would instead add a new keyword. This despite the fact that the keyword for which she was looking, or at least a reasonable synonym, might on closer inspection already be on the list. We realized that the list of keywords would soon grow to be unmanageable (if it was not already) without some way of further structuring it.

Our approach was twofold. First, we added structure to the list of keywords by creating separate properties for the source and recipient of a document. Many of the keywords in the original list had been the names of organizations and groups with whom the bridge team corresponded. We pulled these names out of the keyword list and grouped them by type of entity (e.g. federal, state, local, other). It was now much easier to find the source/recipient values and the remaining list of keywords, now called topics, became much smaller.

An added benefit of creating a source/recipient property is that people with little knowledge of engineering practice or of a specific engineering project are able to assign source/recipient values for a large percentage of the documents in the project files. This information is often directly available from letterhead, memo fields, or in the text of the document. In the case of the bridge replacement team, a student intern has been hired to help with project file management. He has now assumed the responsibility for scanning documents and adding them to the online repository. He is able to assign sources and recipients, as well as dates, topics and document types to many of the documents. In keeping with the design goals mentioned earlier, there is no requirement that property values be assigned to documents at the time that they are added to the online repository. The intern does as much as he can as he scans, and engineers can assign additional topic values to documents, or modify those already assigned, at any time.

Along with adding further structure to the property lists, we addressed the problem of the proliferation of keywords by adding a free text field on the coding form, in which useful information about a document that was not easily indicated by the assignment of values from the existing list could be noted. This free text field could later be searched when attempting to retrieve a document. At the same time, we wanted to support users in adding new items to the coding form as needed. Relevant topics and associated organizations change over the course of an engineering project (which can last upwards of ten years), particularly as it moves from project approval through to design and construction. To provide for modifications we added a link to another online form where new property values can be entered. Our reasoning was that the separation of document coding from property value modification, combined with the possibility of keying relevant information into the free text field, would reduce the frequency with which new values would be added to the metadata fields.

At the suggestion of the engineers we also added two new binary-valued properties; "RE doc" for documents to be made available to the Resident Engineer on the construction team, and "Project History" for documents to be included in the permanent archives of the Department. The existing paper-based practice requires that engineers go through the project files pulling relevant documents when the project moves to construction and again at the completion of the project. Because this is a time consuming task, the engineers thought that it would be useful to mark such documents in the online repository in advance. It has turned out to be difficult to know at the time that the repository is created, however, which documents should become part of the resident engineer's file and which should go to the archives. As a consequence, these properties are now thought of as potentially useful when the engineers are actually faced with the task of pulling RE and project history documents. Engineers could first go through the online repository and mark the documents. They could then more easily be assembled either for printing or for access within another online repository.

As we were developing the online coding form, one of the engineers asked if we could provide a hardcopy version of the form, that he and other engineers could fill out in advance of scanning documents. His question opened up further discussion about divisions of labor across the work of scanning and coding project file documents. The paper coding forms meant that those with engineering expertise can code documents without having to become directly involved in scanning. They can simply drop a document with code sheet attached into a project files in-box. It is then scanned by the intern assigned to the project files, who uses the online coding form to enter the codes designated by the engineer. This practice has now been added to that of simply writing UFS designations on the upper right hand corner of the document, again placing it in the in-box for scanning and coding by the intern. The separation of document coding from scanning, both in time and space, provides flexibility in the working division of labor that accommodates differences in the responsibilities and expertise of the project team.

Organization of the prototype interface

In contrast to the binders for the paper project files, the Project Files home page is the user's entry into the online repository. From there one can add new documents to the repository, view the existing collection along various dimensions, search the repository, and perform various miscellaneous ("administrative") tasks.

Figure 2 shows how the different operations are ordered and interconnected in the interface. Each box in the diagram corresponds to a single interactive web page. As indicated schematically, there are separate web pages for each of the repository views depending on the form of metadata chosen.

Altogether, the interface comprises sixteen pages all of which, apart from the home page and the third-party pdf viewer used to print, are generated dynamically by cgi scripts. (Not shown are various help pages containing textual documentation.)

As there is not sufficient space to describe each of these interfaces in detail, we will briefly describe those that are the most frequently used.

Our efforts to date have focussed on building the online repository, which as of this writing includes approximately 1600 documents or five times that number of pages. The online coding interface is used both to upload a scanned document in the form of a multi-page tiff file, and to upload metadata, usually a subset of the properties available on the coding form. The repository is currently maintained at PARC, where PARC technologies are used to process the image file, creating scaled gif images, ascii text from the OCR, and pdf files suitable for printing. After indexing the text and metadata, we add the new documents to the repository. At that point, they are available for browsing and retrieval from the Department.[3]

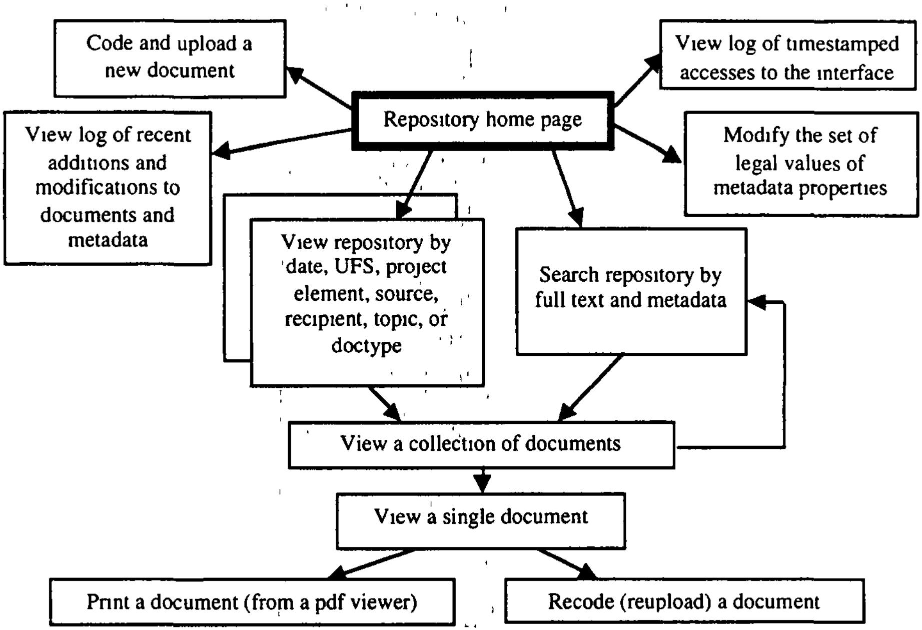
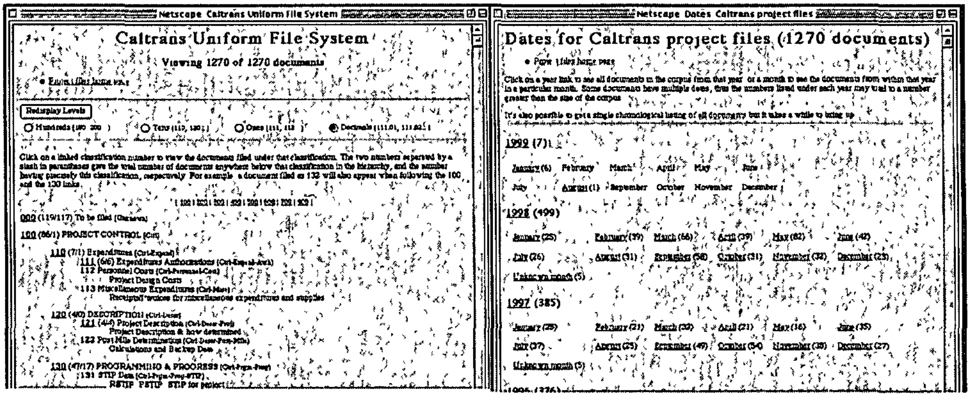

Code and upload a new document
View log of timestamped accesses to the interface
Repository home page
View log of recent additions and modifications to documents and metadata
Modify the set of legal values of metadata properties
View repository by date, UFS, project element, source, recipient, topic, or doctype
Search repository by full text and metadata
View a collection of documents
View a single document
Print a document (from a pdf viewer)
Recode (reupload) a document

Figure 2. Primary flow of control through the prototype's web interface. Boxes represent interactive web pages each generated and controlled by a CGI script. To reduce complexity, the links that connect pages to each other apart from the main flow are not shown.

Documents in the repository are browsed using a variety of *views*. Each repository view offers an overview of the entire corpus with links to subcollections along the given dimension. Figure 3 shows two of these views: by UFS, and by date. There are also viewers for several of the other metadata properties. For example, the view by Source provides a listing of all legal Source entities and the numbers of documents coded as having that entity as source. Following the link from a given Source entity

[3] We expect to move the management of the repository onto the Department's intranet in the near future

brings up a collection browser over the documents having that source.



Netscape Caltrans Uniform File System

Caltrans Uniform File System

Viewing 1270 of 1270 documents

- Elib files home page

Redisplay Levels

○ Hundreds (100 200) ○ Tens (110, 120) ○ Ones (111, 112) ● Decimals (111.01, 111.02)

Click on a linked classification number to view the documents filed under that classification. The two numbers seperated by a slash in parantheses give the total number of documents anywhere below that classification in the hierarchy, and the number having precisely this classification, respectively. For example, a document filed as 132 will also appear when following the 100 and the 130 links.

| 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 |

000 (119/117) To be filed (General)

100 (86/1) PROJECT CONTROL (Ctrl)

110 (7/1) Expenditures (Ctrl-Expend)

111 (6/6) Expenditures Authorizations (Ctrl-Expend-Auth)

112 Personnel Costs (Ctrl-Personnel-Cost)

Project Design Costs

113 Miscellaneous Expenditures (Ctrl-Misc)

Receipts/invoices for miscellaneous expenditures and supplies

120 (4/0) DESCRIPTION (Ctrl-Descr)

121 (4/4) Project Description (Ctrl-Descr-Proj)

Project Description & how determined

122 Post Mile Determination (Ctrl-Descr-Post-Mile)

Calculations and Backup Data

130 (47/17) PROGRAMMING & PROGRESS (Ctrl-Prgm-Prog)

131 STIP Data (Ctrl-Prgm-Prog-STIP)

RSTIP PSTIP STIP for project

Netscape Dates Caltrans project files

Dates for Caltrans project files (1270 documents)

- Elib files home page

Click on a year link to see all documents in the corpus from that year or a month to see the documents from within that year in a particular month. Some documents have multiple dates, thus the numbers listed under each year may total to a number greater than the size of the corpus

It's also possible to get a single chronological listing of all documents but it takes a while to bring up

1999 (7)

January (6) February March April May June

July August (1) September October November December

1998 (499)

January (25) February (39) March (66) April (39) May (62) June (42)

July (26) August (31) September (58) October (31) November (32) December (23)

Unknown month (5)

1997 (385)

January (28) February (21) March (32) April (21) May (16) June (35)

July (37) August (25) September (49) October (50) November (30) December (27)

Unknown month (5)

1996 (376)

Figure 3: Two views over all the documents in the corpus, one organized hierarchically by UFS category, and one chronologically by date. A link from a non-empty category or a year or a month, brings up a browser over the documents having that category, or having dates in the given year or month

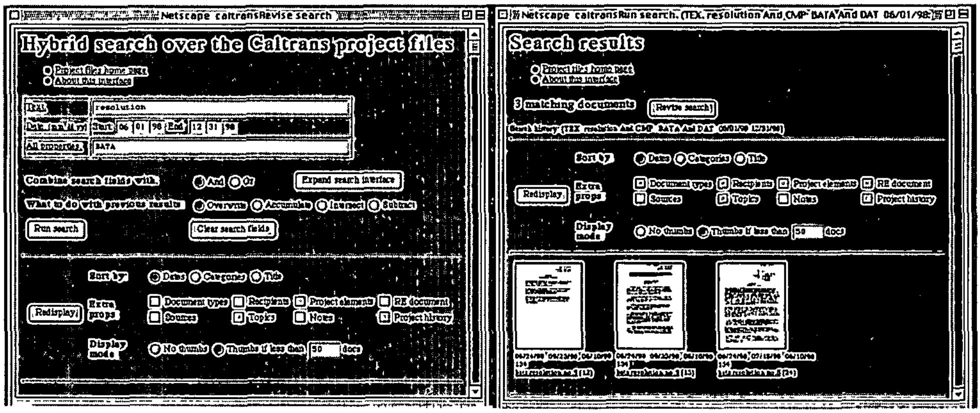
The search interface retrieves documents by means of queries that combine multiple criteria. For example, Figure 4 shows a search for documents that have the word "resolution" in the full text, were dated sometime in the latter half of 1998, and for which the acronym "BATA" appears in any metadata property. Below the query fields is a button panel that lets the current search be combined with the results of a previous search.[4]

The results are displayed in a separate interface shown to the right in Figure 4. The search query can be refined by clicking on the "Revise search" button. This returns to the search interface retaining the current settings of all parameters. The "Revise search" functionality is especially handy after following a link from one of the repository views. For example, following the 1997 link from the Dates view brings up the results interface showing the 385 documents coded as having a date in 1997. Clicking on "Revise search" moves to the search query interface with the start and end date set to the beginning and end dates of 1997. This date range could be limited to portions of 1997, or other query fields could be combined with date to narrow the search. In this way, browsing from repository views can smoothly segue into more refined searches.

We also offer an expanded search interface that lets the user specify particular values of properties from pull-down menus as search terms. The expanded search interface includes fields for text search across the document title and notes properties.

[4] The Chabot system (Ogle and Stonebraker, 1995) uses multiple means of retrieval including text, metadata and image attributes for a corpus of photographic images To date, we have seen less work on hybrid search across scanned document collections

Below the "Revise search" button is a textual description of the search query as it currently stands, including any previous searches that were combined. Beneath that is a panel for controlling the display of search results. Each result is displayed either using a thumbnail image, or as a row in a table. In either case the metadata properties, dates, categories, and titles are always shown. Other properties can be included by clicking on the appropriate checkboxes. Because of the time required to download a large number of images, we default the display to thumbnails only if there are fewer than 50 results. This threshold can be adjusted by the user.[5]



Netscape caltransRevise search

Hybrid search over the Caltrans project files

Text: resolution

Date (mm dd yy): Start 06 01 98 End 12 31 98

All properties: BATA

Combine search fields with: ● And ○ Or

What to do with previous results: ● Overwrite ○ Accumulate ○ Intersect ○ Subtract

Sort by: ● Dates ○ Categories ○ Title

Extra props: □ Document types □ Recipients ☒ Project elements □ RE document □ Sources ☒ Topics □ Notes ☒ Project history

Display mode: ○ No thumbs ● Thumbs if less than 50 docs

Netscape caltransRun search. (TEX. resolution And CMP BATA And DAT 06/01/98

Search results

3 matching documents

Search history (TEX resolution And CMP BATA And DAT 06/01/98 12/31/98)

Sort by: ● Dates ○ Categories ○ Title

Extra props: ☒ Document types ☒ Recipients ☒ Project elements ☒ RE document □ Sources ☒ Topics □ Notes ☒ Project history

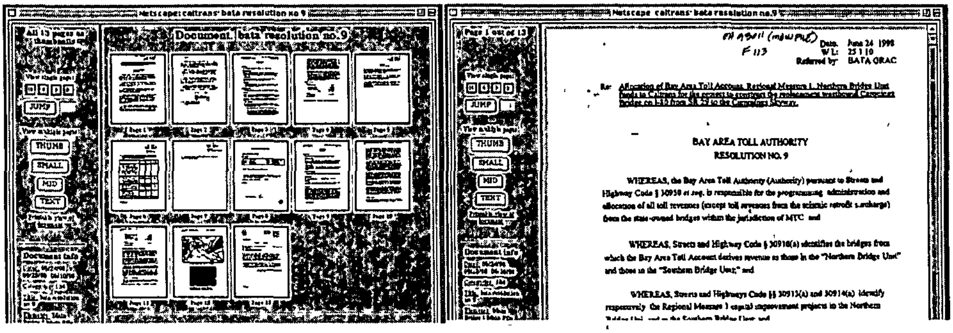
Display mode: ○ No thumbs ● Thumbs if less than 50 docs

Figure 4. A search form with three query fields specified, and the web page showing the three matching documents; namely, those containing the word "resolution", dated in the latter half of 1998, and coded with "BATA" in one or more metadata properties.

Each result, whether thumbnail or title, is linked to the document displayer interface, shown in Figure 5. Here, one can browse the pages of the document at various sizes, view the OCR text, and print the pdf rendition. The metadata is displayed along with a link that leads to a coding form with which to modify the metadata values. This coding form is automatically filled in with the current values, and allows the document image file to be re-uploaded in case the paper document has been re-scanned.

Other parts of the interface include an access log that lists anonymous timestamped entries for each time any of the web page cgi scripts is invoked, and an interface that allows users to change the names of property values and add new ones. More significant reorganizations of the metadata scheme cannot be formulated in this "Modify Props" interface. To date users have tended not to use this interface for even simple additions, preferring to convey their requests to us either directly or through the intern.

[5] Our use of thumbnails to represent the results of a search and to display document pages derives from the Protofoil project (Rao et al, 1994) and is motivated in Blomberg et al (1996)


Netscape: caltrans: bata resolution no.9
Document: bata resolution no. 9
JUMP
THUMB
SMALL
MID
TEXT
Document info

FH 4 9411 (mjw PUC)
F 113

Date. June 24 1998
W.I.: 25 1 10
Referred by: BATA ORAC

Re: Allocation of Bay Area Toll Account, Regional Measure 1, Northern Bridge Unit funds to Caltrans for the project to construct the replacement westbound Carquinez bridge on I-80 from SR 29 to the Carquinez Skyway.

BAY AREA TOLL AUTHORITY
RESOLUTION NO. 9

WHEREAS, the Bay Area Toll Authority (Authority) pursuant to Streets and Highway Code § 30950 et seq. is responsible for the programming, administration and allocation of all toll revenues (except toll revenues from the seismic retrofit surcharge) from the state-owned bridges within the jurisdiction of MTC; and

WHEREAS, Streets and Highway Code § 30910(a) identifies the bridges from which the Bay Area Toll Account derives revenue as those in the "Northern Bridge Unit" and those in the "Southern Bridge Unit;" and

WHEREAS, Streets and Highways Code §§ 30913(a) and 30914(a) identify respectively the Regional Measure 1 capital improvement projects in the Northern Bridge Unit and in the Southern Bridge Unit; and

Figure 5: The document displayer interface is used to browse the pages of a document either as thumbnails as shown on the left, or in larger reductions up to a readable size for a single page as shown on the right The ascii text from OCR can also be displayed. The metadata settings are shown on the left of the interface The interface includes links to display and print the pdf rendition of the document as well as to modify the document's metadata

Relations between online and paper documents

Keeping the renderings of online documents aligned with the paper files is crucial to the success of our project.[6] In this section, we describe some of the practices of maintaining this alignment and discuss issues relevant to repository design. We first, however, describe a significant event in the life of the project files that had implications for the alignment problem.

Grouping the project files by subproject

As the bridge replacement project has moved out of the environmental impact phase and into design, it has also grown in size and scope. Recently the project was split into a dozen or so subprojects comprising independent budget centers, named according to "Expenditure Authorization" (EA) number. Because each subproject (and its project files) moves into construction according to its own time schedule, the engineering team decided to subdivide the project files according to EA number. The result is a collection of mini-project files, each named by EA number and sorted by UFS and date.

In response to this reorganization of the project files, we added an "EA" property to our metadata. Assigning an EA number was mainly an issue for more recent documents, as documents created before the split were given the original EA number. Engineers were encouraged to pencil in an EA number along with a UFS code on the

[6] The paper project files continue to be maintained in binders alongside the online collection, because of the experimental status of the prototype and the fact that some documents will always be kept in hardcopy

top corner of the document's first page, to help the intern in doing the scanning and coding.

The new scheme seemed to be working well until it became clear that the EA numbers were not a stable category. New ones were being formed and old ones merged and even reassigned uncomfortably often. This posed a problem for the binders, which were relabeled to reflect the new designations. One of the engineers then proposed abstracting from specific EA numbers to a short designation corresponding to the part of the project (e.g. MAIN for main bridge, INT for interchange, etc.). Each of these "project elements" was expected to cover several possibly shifting EA numbers, while remaining relatively stable over the remainder of the project. Again the binders were relabeled to reflect the new designations, and our metadata scheme was changed as well.

The problem of alignment

For the work of aligning the online and physical files, three of the metadata fields are most significant: project elements, UFS categories, and dates.[7] These are grouped uppermost on our coding form. They are also the three values that govern where in the binders a document is filed: first by subproject, then UFS, and finally by date. For subprojects with few documents, some levels of the UFS hierarchy (say, below the tens digit) are not allocated separate sections, but rather are simply sorted by date.

Of the three properties, the date is most easily read off of the document, whereas project element and UFS are harder to code. For this reason, the engineers pencil in those values on the document. The three properties are thus represented in three ways in the overall filing system:

- On the paper document itself: Dates appear in the document text, while project element and UFS code are penciled onto the first page.
- In the document's location: Binders are organized according to project element and UFS code; within these, documents are filed in chronological order, most recent first.
- In the online metadata: For each document, our system stores up to three dates, three UFS codes, and any number of project elements.[8]

We should note that there are good reasons for the redundancy in coding. The value of the codes for document filing and for online search are perhaps obvious. But what about the penciled values on the physical document? Here, the primary goal is mobility; the document should carry its codes when it is found out of the binders. This happens before the document is scanned, when the penciled codes serve as

[7] Though each of these fields can have multiple values, the paper documents are filed and ordered according to the first project element, the first UFS, and the first date

[8] As described above, our system stores many more metadata properties than these three, but the others raise no alignment problems, since they are neither used in organizing the paper files, nor written in a modifiable way on the document itself.

instructions to the intern, as well as after filing, if the document is pulled from the binders for inspection or copying.

Ideally, a change to one of the three representations should lead to immediate updates of the other two. But this is not always easy. For example, an engineer changing the values of codes at her workstation might be reluctant to take the time to walk over to the project files, find the binder, move the documents to their new home, and change the penciled in codes. Likewise, someone working with the binders might choose a more appropriate UFS code for a document. In such a situation, refiling the physical document and writing down its new code is likely to be much easier than getting onto the system and searching for the document. At present we see engineers pulling documents from the binders, crossing out their current UFS codes and writing in new ones. They then place these documents in a stack for the intern to revise on the system and refile in the binders.

There is a further complication to the three-way redundant coding of documents. As we mentioned, at scan time most of the documents have penciled on them a project element value and UFS code. These penciled values are legible on the system when browsing the page images online. Though we can perhaps expect the workgroup to maintain the alignment of the three representations described above, it is certainly too much to expect them to rescan the first page and upload it into the system after changing the penciled marks. We could address this misalignment by giving precedence to the online metadata over any codes visible on the online page images. Unfortunately, this misses the problem of printing. One of the advantages of retaining a printable rendition of the scanned document online is that users can obtain hardcopy without having to recover the document from the binders (say, if they are working remotely). Once this rendition is printed, however, its out of date codes may lead to subsequent confusion.

Techniques for maintaining alignment

At any given time the shelves near the project files include stacks of documents in need of coding or recoding, recoded documents whose metadata needs modification online, scanned documents in need of filing, and the like. Sometimes these stacks are labeled as to their status, sometimes not. Amidst the confusion of piles of paper near the project files, it can help to have at least one record of the history of the processing of the document easily readable. In particular, when encountering a physical document a question arises as to whether it has been scanned and is thus findable online. The work of scanning a document now includes, therefore, the practice of affixing a self-sticking blue dot to the document's upper right corner.

Another aid to recording the status of the document's processing history is the engineers' choice to cross out rather than erase the penciled UFS codes when changing them. This can help with the task of aligning online metadata with the codes on the physical document. If the online code matches the one crossed out on paper, one can usually assume that the new code on the paper reflects the most recent change.

Technological possibilities

Online metadata include "standard" filing classifications relevant to the bureaucracy of project management (Marshall, 1998) or to schemes like the UFS maintained across the enterprise. In some future world perhaps all relevant workgroups (e.g. design, construction, archives, legal) could be assured access to the online project files, which they could assume held the "truth" - the trusted values. But in a transitional world like the one we encounter at the Department, where some of the players depend on having versions on paper or where paper is saved for legal signatures and the like, the problems of paper and digital alignment will persist.

One simple improvement to the technology of alignment could involve the automatic maintenance of a comprehensive version history. In particular, one could obtain the time of any change to any property value in the repository. This record of changes could be matched against the penciled values on the physical document to resolve cases of misalignment between online and offline codes (Dourish et.al., 1998).

A more radical change to both technology and practice would involve inscribing machine-readable codes on each project file document (Bloomberg, 1997; Barrett & Maglio, 1998). Ideally, the code would be human-readable as well as machine-readable so as to allow changes to be marked up on the physical document. To input these changes to the system, one could "swipe" the bar code or rescan that portion of the image to prompt the system to bring up the online rendition for recoding. If desired, the document could then be reprinted with a new machine-readable code.

Toward flexible document management

Several issues, we believe, should be considered by those who contemplate putting document collections online in support of group work. We present the issues here as a set of informed questions that together can be thought of as a kind of "due diligence" for analyzing and enabling flexibility in workgroup document handling both online and off.[9]

Who scans; who codes; who searches?:

Allowing flexibility with respect to who scans and who codes documents permits

[9] Drawing on an ethnographic study, Marshall (1998) analyzes metadata along several dimensions that include location, source, access, scope and temporality. We see our analysis as complementing hers, focusing on the day-to-day practical and technological contingencies that enable and constrain the emergence of document coding practices.

reconfigurable divisions of labor within workgroups. In our project, scanning has been done almost exclusively by a part-time student intern, but we imagine that other groups within the Department and elsewhere might make other choices. Coding on the other hand requires the involvement of engineers on the project. While the intern is able to assign date, source and recipient codes, project element and UFS codes require engineering expertise and familiarity with the project's history.

Our web interface supports distributed access to the project files from individual desktops. Nonetheless, most of the online searches to date have been done by the intern on behalf of the engineers. The intern has become familiar with both the coding and search interfaces through his experience in scanning documents, and therefore is viewed as an expert when it comes to the particulars of our prototype. We believe, however, that others will begin to conduct searches on their own as they gain familiarity with the interface.

When are documents coded?

The experiences of the bridge replacement team in developing a working practice around document coding suggest that the ability to code documents at various times is desirable. Specifically, document coding in this case occurred at three different times: before the document is scanned, usually by an engineer penciling in codes on the first page or filling out a coding form; at the time the document is uploaded just after scanning; or later upon encountering the document after a search.

Allowing coding and recoding at any of these times means that the system is able to meet the organizational contingencies of filing and search. These can include the need or desire to rush hardcopy documents into the system without taking the time required for coding, or the need to recode documents following changes to the metadata scheme or on discovering errors or omissions in a document's current coding.

How much coding is necessary/appropriate?

An initial goal of our prototyping work was to allow documents to be added to the system with a minimum of required coding. Early on, documents were entered into the system with little more than a date. In order to enable physical filing, however, documents must be coded with at least a UFS and a project element. Other codes may be added at scan time or following a search.

The question of how much to code is an issue that is under debate within the workgroup itself with some engineers taking the position that only minimal coding is required, relying on full-text search and UFS codes, while others support the idea of coding as much information as possible. Our approach has been to build a technology that accommodates either view, not casting any single perspective into stone through our design.

How to keep online and hardcopy documents aligned?

A continuing concern for us has been the relation between online and hardcopy documents. Here we see difficult tradeoffs between flexibility and alignment. In the case described here we attempt to align three representations of metadata codes: location in the physical files, codes inscribed on the document pages, and online representations of metadata. While these may well generalize across a variety of cases of working document collections, significant variation is also possible.

For example, in some cases alignment may not be that important. An organization might be willing to tolerate the occasional inconsistency if it was always clear which rendition of the coding to trust. In other cases keeping some record of changes to metadata, both online and on the document's pages, could address some alignment problems. But even such lightweight versioning might not be worth the trouble for still other organizations. In part the degree of investment placed on maintaining alignment may be a function of an organization's need to move quickly from paper to online and vice versa.

How does a metadata scheme evolve?

Our project has made visible some of the effort required to support evolving metadata, both for locally customizable, enterprise-wide standards like the UFS, and for locally designed and implemented schemes like a list of source and recipient entities. Our commitment to flexibility requires that the online and physical filing system keep up with changing agendas as projects progress from stage to stage.

At the same time, changing metadata schemes incur costs (Bentley & Dourish, 1995; Trigg & Bodker 1994). While some changes expand a scheme without affecting the coding of existing documents, other changes can cause a collection of documents to become "legacy"; that is, out of synchronization with newly scanned and coded documents. At times the change to a scheme comes with instructions on how to handle affected documents. On occasion, these instructions can even be automated with scripts ("Recode all documents in category 710 as 314"). In other cases, the documents may have to be visited one at a time by someone who is an expert in the content matter.

Conclusion

The activities reported in this paper make visible once again the dynamic character of the CSCW domain, where the latter encompasses both our own work as system builders and the work of those we aim to support. Our design project on working document collections is positioned within the larger project of cooperative design with users, as an approach to the creation of more useful and useable computer artifacts. It is the combination of envisioning, building and use that affords the power of cooperative design, as we work our way through successive rounds of trial and discovery regarding all of the ways in which the world is different than we had imagined it to be. We hope here to have provided a view onto the practicalities involved in moving working document collections online, as well as the interesting research questions that such a project poses for us and how we have attempted to answer them. This unfolding horizon of design questions and at least tentative answers is one sense of what we mean by the shared repository's evolution.

Another, less obvious aspect of CSCW as an evolving practice emerges from our experience as well. This aspect is tied to the fact that our prototyping project runs alongside and in relation to the bridge replacement project itself. Far from a situation in which we as designers enter a stable arena engaged only in maintaining itself, we find ourselves joining in with ongoing processes of development and change. In particular, systems of classification like that used in the management of critical documents mediate the interests of organization-wide consistency, manifest here in the basic template of the Uniform File System, with the requirements of specification and customization evident in the elaborations, modifications and re-workings of that scheme by the local workgroup. More generally, the project of designing a bridge itself constitutes a trajectory of shifting concerns, accountabilities and actors. All of this makes up the dynamic field of action both for us and for our collaborators. In this sense, co-development of CSCW technologies like a shared document repository means more than engaging prospective users in the design of new computer systems to support their work. It requires that we as designers engage in the unfolding performance of their work as well, co-developing a complex alignment among organizational concerns, unfolding trajectories of action, and new technological possibilities.

Acknowledgments

We would like to thank the members of the bridge engineering team for the time they graciously devoted to this project and for their insights on just what is required to design really useful online document repositories. Without their willing participation this project would not have been possible. We are also grateful to four anonymous reviewers for helpful comments.

References

Blomberg, J., Suchman, L., & Trigg, R. (1996): Reflections on a Work-Oriented Design Project. *Human-Computer Interaction, 11*(3), 237-265.

Barrett, R. and Maglio, P. P. (1988) "Informative Things: How to attach Information to the Real World", *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, pp. 81-88.

Bloomberg, D. (1997) "Embedding digital data on paper in iconic text.", In *Proceedings of IS&T/SPIE EI'97, Conference 3027: Document Recognition IV.* San Jose, CA, Feb 12-13, pp. 67-81.

Bentley, R. and Dourısh, P. (1995) "Medıum versus Mechanıshm: Supportıng Collaboratıon through Customızatıon", *Proceedings of the Fourth European Conference on Computer-Supported Cooperatıve Work ECSCW'95* (Stockholm, Sweden) Dordrecht. Kluwer.

Celentano, A , Fugını, M. G.,. Pozzı. S. (1991) "Classıfication and Retrıeval of Documents Usıng Offıce Organızatıon Knowledge", *Proceedıngs of the Conference on Organızatıonal Computıng Systemss*, pp. 159-164.

Dourısh, P , Edwards, K , LaMarca, A. and Salısbury, M (1998) "Presto. An Experımental Architecture for Fluid Interactıve Document Spaces." Xerox PSRC report (submıtted to ACM Transactıons on Computer-Human Interaction) Palo Alto: Xerox Corporatıon

Ehn, P. (1988) *Work-Orıented Desıgn of Computer Artifacts* Stockholm: Arbetslıvscentrum.

Greenbaum, J. & Kyng, M. (Eds.), (1991) Desıgn at Work:, Cooperatıve Desıgn of Computer Systems. Hıllsdale, New Jersey: Lawrence Erlbaum Associates.

Kidd, A (1994) "The Marks are on the Knowledge Worker Accessıng and Usıng Stored Documents", *Proceedıngs of ACM CHI'94 Conference on Human Factors in Computing Systems* , vol 1 pp 186-191.

Kyng, M , & Mathiassen, L. (Eds.). (1997) Computers and Desıgn ın Context. Cambridge, MA. MIT Press.

Malone, T. W. (1983) "How Do People Organized Their Desks? Implıcatıons for the Desıgn of Office Informatıon Systems", *ACM Transactıons on Office Information Systems,* vol. 1, no 1, 1983, pp 99-112.

Marshall, C. C (1998) Makıng Metadata: A study of metadata creation for a mıxed physıcal-dıgıtal collectıon. In I Witten, R. Akscyn, & I. Frank M. Shıpman (Eds), *Proceedıngs of Dıgıtal Libraries '98*. Pittsburgh, PA, June 23-26. ACM Press, 162-171.

Nardı, B., Barreau, D. (1997) Fındıng and Remındıng" Revisited: Approprıate Metaphors for Fıle Organızatıon at the Desktop / ACM SIGCHI Bulletın v.29 n.1 p 76-78.

Ogle, V E , & Stonebraker, M. (1995). Chabot· Retrıeval from a Relatıonal Database of Images. *IEEE Computer, 28*(9).

Rao, R., Card, S K., Johnson, W , Klotz, L., & Trigg, R. (1994). Protofoıl: Storıng and Findıng the Informatıon Worker's Paper Documents ın an Electronic File Cabınet In B. Adelson, S. Dumaıs, & J. Olson (Eds), *Proceedings of ACM CHI '94 Conference on Human Factors ın Computing Systems*. Boston, Massachusetts, April 24-28. ACM Press, 180-185.

Rose, D E , Mander, R., Oren, T., Ponceleon, D. B., Salomon, G., Wong, Y. Y (1993) "Content Awareness ın a Fıle System Interface: Implementıng the 'Pıle' Metaphor for Organızıng Informatıon", *Proceedings of the Sixteenth Annual Internatıonal ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. .260-269.

Suchman, L. (1998) Organızing Alıgnment: The case of brıdge buıldıng", Symposıum on Sıtuated Learnıng, Local Knowledge and Action: Social Approaches to the Study of Knowıng ın Organızatıons, Academy of Management Meetıng, August 9, 1998 SanDiego, CA.

Suchman, L (1999) "Embodıed Practices of Engineerıng Work", To appear ın Mind, Culture and Activity.

Suchman, L., Trıgg, R. and Blomberg, J. (1998) "Workıng Artıfacts. Ethnomethods of the prototype", Presented at the 1988 Amerıcan Socıologıcal Assocıatıon Meetıngs, August 22, San Francısco.

Trıgg, R. H , & Bødker, S. (1994). From ımplementation to desıgn: Taılorıng and the emergence of systematizatıon ın CSCW. In R Furuta & C Neuwirth (Eds.), *Proceedıngs of the Conference on Computer Supported Cooperative Work (CSCW'94)*. New York· ACM Press, 45-54.