Corporate Memory in Action

Renate Fruchter\(^1\) and Peter Demian\(^2\)

Abstract

Managing and reusing knowledge in architecture, engineering and construction (A/E/C) firms can lead to greater efficiency, productivity, and improved designs. However reuse often fails, since knowledge is not captured, it is captured out of context rendering it not reusable, or there are no formal mechanisms for finding and retrieving reusable knowledge. This paper presents ongoing research on design knowledge reuse that introduces the notion of knowledge in context. We argue that in order for knowledge to be reusable, the user needs to see the context in which this knowledge was created and interact with this rich content. We call a repository of such knowledge in context the corporate memory. We describe empirical observations of designers reusing knowledge from their personal design experiences. Based on these observations, we formalize two key activities that must be supported in the process of knowledge reuse from a corporate repository: finding reusable items, and understanding these items in context. We illustrate the use of a prototype system, called CoMem, that supports these activities with an industry case study. CoMem is distinguished from the document-centric state-of-practice solutions by its approach of “overview first, filter, and then details-on-demand” accomplished through the implementation of three interaction metaphors: corporate map, fisheye lens, and storyteller.

\(^1\) ASCE Member, Director of Project Based Learning Laboratory, Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305-4020, fruchter@ce.stanford.edu
\(^2\) Ph.D. Candidate, Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305-4020, pdemian@stanford.edu
Introduction

The average designer, whether consciously or subconsciously, draws from a vast well of previous design experience. This can be experience acquired by the individual or by his/her mentors or professional community. We refer to this activity as design knowledge reuse. Specifically, we define design knowledge reuse as the reuse of previously designed artifacts or components, as well as the knowledge and expertise ingrained in these previous designs.

Designers in the architecture, engineering, and construction (A/E/C) industry frequently reuse design knowledge from previous (completed or dormant) projects in their current projects. For example, structural designers reuse typical or standard building details, design tools such as spreadsheets, or abstract design ideas such as king post trusses. We distinguish between two types of reuse:

Internal knowledge reuse: a designer reusing knowledge from his/her own personal experiences (internal memory). For example, a structural designer might remember that the last time she designed a floor slab for a hotel ballroom it was too thin, which resulted in vibration problems. The next time she is faced with a similar design situation, she designs the floor slab deeper.

External knowledge reuse: a designer reusing knowledge from an external knowledge repository (external memory). For example, the same structural designer might look for floor slab designs in her company’s standard components database. She retrieves a floor slab design that comes with a spreadsheet for calculating the correct slab thickness. This spreadsheet takes into account the company’s previous experiences with bouncy floor slabs and increases the depth beyond the minimum required by the building code.

Whereas internal knowledge reuse is effective, external knowledge reuse often fails. This failure occurs for numerous reasons, including:

- Designers do not appreciate the importance of knowledge capture because of the additional overhead required to document their process and rationale. Consequently, knowledge is often not captured.
- When knowledge capture does take place, it is limited to formal knowledge (e.g. documents). Contextual or informal knowledge, such as the rationale behind design decisions, or the interaction between team members, is often lost, rendering the captured knowledge not reusable, as is often the case in current industry documentation practices.
- There are no mechanisms from both the information technology and organizational viewpoints for finding, and retrieving reusable knowledge from large corporate archives.

Our empirical observations of designers at work show that internal knowledge reuse is effective since:

- The designer can quickly find (mentally) reusable items.
- The designer can remember the context of each item, and can therefore understand it and reuse more effectively.
Internal knowledge reuse is effective but it limits the pool of knowledge available to be reused to only those components in whose design the individual was involved. Enabling the designer to reuse from a repository that contains the work of the entire corporation or design firm would drastically increase the knowledge available to be reused. We use the above observations of the effectiveness of internal knowledge reuse as the basis for improving external knowledge reuse.

We introduce the notion of knowledge in context. Knowledge in context is design knowledge as it occurs in a designer’s personal memory: rich, detailed, and contextual. This context includes design evolution (from sketches and back-of-the-envelope calculations to detailed 3D CAD, analysis, and simulations), design rationale, and relationships between different perspectives within cross-disciplinary design teams. We define the corporate memory as a repository of knowledge in context; in other words, it is an external knowledge repository containing the corporation’s past projects that attempts to emulate the characteristics of an internal memory, i.e. rich, detailed, and contextual. The corporate memory grows as the design firm works on more projects.

The motivation behind the development of external knowledge reuse systems is that the capture and reuse of knowledge is less costly than its recreation. In many A/E/C firms today, knowledge capture and reuse is limited to dealing with paper archives. Even when the archives are digital, they are usually in the form of electronic files (documents) arranged in folders which are difficult to explore and navigate. A typical query might be, “how did we design previous cooling tower support structures in hotel building projects?” In many cases, the user of such systems is overloaded with information, but with very little context to help him/her decide if and what to reuse.

This paper addresses the following central questions:

- What are the key characteristics of the knowledge reuse process?
- How can the design knowledge reuse process in the AEC industry be supported by a computer system?
- What are natural idioms that can be modeled into a computer system to provide an effective knowledge reuse experience to a designer?

The objective is to assist the designer and to support the process of design knowledge reuse rather than to automate it.

**Related Research**

Related research studies on design knowledge reuse focus either on the cognitive aspects or on the computational aspects.

Research into the cognitive aspects of reuse has helped to identify the information needed by designers. Kuffner and Ullman (1990) found that the majority of
information requested by mechanical engineers was concerning the operation or purpose of a designed object, information that is not typically captured in standard design documents (drawings and specifications). Finger (1998) observed that designers rarely use CAD tools to help them organize and retrieve design information. This research extends these findings by formalizing the requirements for contextual information when reusing items from previous projects. On the computational side, research into design knowledge reuse focuses on design knowledge representation and reasoning. Knowledge representation ranges from informal classification systems for standard components to more structured design rationale approaches (Hu et. al. 2000 gives an overview). There is a tradeoff in design rationale systems between the overhead for recording design activities and the structure of the knowledge captured. Highly structured representations of design knowledge can be used for reasoning. However, these approaches usually require manual pre or post processing, structuring and indexing of design knowledge. For example, ARCHIE is a case-based reasoning tool for aiding architects during conceptual design (Domeshek and Kolodner 1993). CASECAD enables designers to retrieve previous design cases based on formal specifications of new design problems (Maher 1997). IDEAL is a model-based reasoning tool that uses both general domain knowledge as well as knowledge from specific cases (Bhatta et. al. 1994). These tools enable knowledge retrieval and reuse based on a priori set representations that are specific to narrowly defined domains and media types.

This research brings together the cognitive and computational approaches. We consider reuse to be a combined effort involving both the human and the computer. We therefore address the issue of design knowledge reuse as a human-computer interaction (HCI) problem, and we take a user-centered approach to designing this interaction. In our approach, capture and indexing take place in real time, with the least possible intrusion on the design process. Knowledge is captured by supporting the typical communication and coordination activities that occur during collaborative design (Fruchter et. al. 1998).

**Knowledge Reuse Process**

We conducted an ethnographic study at a structural design office in Northern California to observe how design reuse occurs in current engineering practice. We identified a number of scenarios engaging different stakeholders such as experts and novice or junior engineers. This paper presents a detailed example of one of the scenarios of *internal knowledge reuse* in which a novice asked an expert engineer questions. We observed that the expert always referred to his work on previous projects when answering these questions.

In developing the CoMem (Corporate Memory) prototype that supports knowledge reuse, we adopted a scenario-based method to the design of human-computer interaction (Rosson and Carroll 2001). The premise behind scenario-
based methods is that descriptions of people using technology are essential in analyzing how technology is reshaping their activities. The scenario-based design process begins with an analysis of current practice using problem scenarios. The following is an example of a problem scenario that was developed based on our empirical observations.

An expert structural designer, Eric, and a novice, Nick, both work for a structural design office in Northern California. The office is part of the “X Inc” Structural Engineering Firm. They are working on a ten-story hotel that has a large cooling tower unit. Nick must design the frame that will support this cooling tower. Nick gets stuck and asks Eric for advice. Eric recalls several other hotel projects that were designed by “X Inc”. He tells Nick to look at the drawings from the Bay Saint Louis project, a hotel project that “X Inc” designed a couple of years ago. Nick spends over an hour looking for the Bay Saint Louis drawings in the “X Inc” paper archive. He eventually finds the drawing sheet with the Bay Saint Louis cooling tower frame. He shows it to Eric. The drawing shows the cooling tower frame as it was finally built. It is a steel frame. Eric realizes that what he had in mind for Nick to reuse is an earlier version that had a steel part and a concrete part. He is not sure if this earlier version is documented somewhere in the archive. Rather than go through the paper archive again, Eric simply sketches the design for Nick. Eric’s sketch also shows the load path concept much more clearly than the CAD drawing would have, which helps Nick to understand the design. Eric explains to Nick how and why the design evolved. Given the current project they are working on, it would be more appropriate to reuse the earlier composite version. Eric recalls that the specifications of the cooling tower unit itself, which were provided by the HVAC subcontractor, had a large impact on the design. Nick now feels confident enough to design the new cooling tower frame by reusing the same concepts as the Bay Saint Louis cooling tower frame, as well as some of the standard details.

During our study, the expert’s internal knowledge reuse process was observed to be very effective. He was always able to recall directly related past experiences and apply them to the situation at hand. Two key observations in particular characterize the effectiveness of internal knowledge reuse:

1. Even though the expert’s internal memory was very large (over 20 years experience), he was always able to find relevant designs or experiences to reuse.
2. For each specific design or part of a design he was reusing, he was able to retrieve a lot of contextual knowledge. This helped him to understand this design and apply it to the situation at hand. When describing contextual knowledge to the novice, the expert explored two contextual dimensions: the project context and the evolution history.
CoMem Modules

CoMem is a prototype system that supports external knowledge reuse by supporting the three steps observed during internal knowledge reuse: find, explore project context, and explore evolution history. The CoMem human-computer interaction experience is based on the principle of “*overview first, zoom and filter, and then details-on-demand*” (Shneiderman 1999). This principle is used to design a user experience that is based on our empirical observations of internal knowledge reuse (Table 1).

**Table 1. Transformation of observed reuse steps into user interactions**

<table>
<thead>
<tr>
<th>Reuse step</th>
<th>User interaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find reusable item</td>
<td>“overview first, zoom and filter, and then details-on-demand”</td>
</tr>
<tr>
<td>Explore item’s evolution history</td>
<td>Evolution history explorer</td>
</tr>
<tr>
<td>Explore item’s project context</td>
<td>Project context explorer</td>
</tr>
</tbody>
</table>

The input to CoMem is a corporate archive of *project memories* (Fruchter et al. 1998, Reiner and Fruchter 2000). A project memory consists of a hierarchy of semantic objects: a *project object* containing multiple *discipline objects*, each discipline object containing multiple *component objects*. Each semantic object can be linked to individual graphic objects from a shared 3D CAD model. In this framework, a graphic object can have multiple interpretations in different contexts; for example, the same graphic object can be linked to the “Shear Wall” component in the “Engineering” discipline and the “Partition Wall” component in the “Architecture” discipline (Fruchter 1996). Each object (whether graphic or semantic) is individually versioned each time a change is made to the project, and so the evolution of the project is captured at three levels of granularity: project, discipline subsystem, and component. It is also possible to attach electronic documents (such as sketches, analysis models, spreadsheets, online vendor catalog information, etc.) to any of the semantic objects.

The objective of CoMem is to enable the user to interact with such a potentially huge archive of design knowledge from previous projects. Specifically, CoMem must support the three activities in the reuse process: *find, explore project context, and explore evolution history*. CoMem comprises three modules that support each of those three activities: *overview, project context explorer, and evolution history explorer*.

The CoMem *overview* prototype module supports the designer in finding reusable items. Assuming that the designer does not know a priori where in the corporate memory reusable items can be found, the overview should initially show all items. The overview needs to provide a succinct at-a-glance view of the entire corporate memory. CoMem uses a map metaphor for the overview. The hierarchy consisting of projects, disciplines, and components is visualized as a series of
nested rectangles using the squarified treemap (Bruls et. al. 1999) technique. The area on the map allocated to each item is based on a measure of how much knowledge this item encapsulates, i.e. how richly annotated it is, how many times it is versioned, how much external data is linked to it. Each item on the map is color-coded by a measure of relevance to the designer’s current task, and so the overview gives the designer an indication of which “regions” of the corporate memory contain potentially reusable items. Currently, this relevance measure is based on latent semantic analysis (Landauer and Dumais 1995) of the textual data in the corporate memory. CoMem allows the user to filter out items from the overview using dynamic querying. In a dynamic querying environment, search results are instantly updated as the user adjusts sliders or selects buttons to query a database (Shneiderman 1994). The designer can filter based on relevance, date, person, or keyword. Items that are filtered out can either appear grayed out, or are not drawn at all, leaving more space for the remaining items. The filtering tools are intended to prevent information overload.

Once the user has selected an item from the overview, the CoMem project context explorer prototype module supports the designer in exploring this item’s project context. This shows the project and discipline to which this item belongs, as well as related components, disciplines and projects. The item selected from the overview becomes the focal point of the project context explorer. CoMem uses a fisheye lens metaphor for the project context explorer. A fisheye lens balances local detail with global context. Given a user-specified focal point, CoMem uses the fisheye formulation (Furnas 1981) to assign a degree of interest to every item in the corporate memory. Items with a higher degree of interest are displayed more prominently in the project context explorer. In the CoMem project context explorer, each object is positioned in the vertical axis according to its level of granularity, and in the horizontal axis according to its degree of interest. This visualization emphasizes structural relationships in the hierarchy that are obscured in the map, and facilitates effective exploration of the focal node’s context.

Finally, in the CoMem evolution history explorer prototype module the designer can explore the evolution history of any item selected from the overview. This view tells the story of how this item evolved from an abstract idea to a fully designed and detailed physical artifact or component. CoMem uses a storytelling metaphor for the evolution history explorer. This is based on our observation that the most striking means of transmitting knowledge from experts to novices in A/E/C design offices is through the recounting of experiences from past projects. Stories convey great amounts of knowledge and information in relatively few words (Gershon and Page 2001). Each version is represented by a circle that is color-coded to indicate this version’s level of importance (low, conflict, or milestone) and level of sharing (private, public, or consensus). The levels of importance and sharing are provided by the original designers working on the project. Next to this circle, thumbnails appear for any CAD objects, sketches, documents, or notes linked to this version. The designer is able to click on any of
the thumbnails for a larger view of this content. If the display is too dense because the evolution history contains a large number of versions, the designer is able to filter out versions based on the levels of importance and sharing to help prevent information overload.

Interaction Scenario

The following is an example of an interaction scenario that was developed from the presented problem scenario. It illustrates the use of CoMem in the design of the hotel cooling tower support structure.

As before, Eric and Nick are working on a ten-story hotel that has a large cooling tower unit and Nick is assigned the task to design the frame that will support this cooling tower. They use the ProMem system (Figure 1) to capture the current project evolution. Nick gets stuck, but Eric is not around to help. Nick clicks on the Reuse button in ProMem, which brings up CoMem. CoMem displays a map of the entire corporate memory (Figure 2). Items on the map are color-coded according to how relevant they are to his current project. Nick uses sliders to filter out irrelevant projects, disciplines, and components from the map. Most of the rectangles in the map are now grayed out (Figure 3). Of the few items that remain highlighted, Nick notices the Bay Saint Louis project. It has a relevant Engineering discipline, and several relevant components within that discipline. He clicks on the component labeled Cooling Tower Frame.

The project context and evolution history of the Bay Saint Louis cooling tower frame appear in two separate displays (Figure 4, top right and bottom left corners). Nick examines the evolution of the frame. He chooses to see only milestone versions of the evolution (Figure 5). He sees that it started as a composite steel-concrete frame but was later changed into a steel frame. He sees several notes that were exchanged between the architect and engineer that help to explain this change. Nick clicks on one of the versions, and a detailed view of this version appears (Figure 6). He finds a useful early sketch of the composite frame, which he saves to his local hard drive.

Next, Nick begins to explore the project context of the Bay Saint Louis frame. He clicks on the Engineering discipline object in the project context explorer and sees that the Bay Saint Louis structural design criteria are similar to those in his current project (Figure 7). He notices a related component under the HVAC discipline: it is labeled Cooling Tower. This is the air conditioning unit that is supported by the frame. Nick finds a specifications sheet attached to this component (Figure 8). It gives him an idea of the loads for which he must now design his cooling tower.
Figure 1: Nick is working in the ProMem system when he gets stuck. He presses the *Reuse* button.

Figure 2: CoMem pops up and displays the corporate map overview.

Figure 3: Nick filters out some items from the map using the sliders. He notices the cooling tower frame and clicks on it.

Figure 4: The project context and evolution history of the cooling tower are displayed.

Figure 5: Nick filters out unimportant versions from the cooling tower evolution using the slider and enlarges several thumbnails by clicking on them.

Figure 6: Nick clicks on a particular version of the cooling tower from the evolution history explorer. The details of this version appear in the display area.
Conclusions

Reusing designs and design knowledge from an external repository of knowledge from previous projects is an important process that often fails. We attribute this failure to the fact that state-of-practice archiving systems (such as archives of paper drawings or electronic files arranged in folders) do not support the designer in finding reusable items and understanding these items in context in order to be able to reuse them. We argue that the designer must be able to explore the project context and evolution history of an item in order to understand it and reuse it. We present a prototype system, CoMem, which supports finding, project context exploration, and evolution history exploration in a large corporate memory.

Using traditional user interfaces such as Windows Explorer to find an item in a corporate archive, the user has to choose between a top level overview of the folder tree and the fully expanded folder tree that would require scrolling through project files and folders while maintaining a partial view of the tree on screen. CoMem proposes interactions that are radically different than those found in current practice. CoMem supports the finding of reusable items using the corporate map, which provides a succinct at-a-glance overview of the entire corporate memory and provides filtering tools to prevent information overload. Once a potentially reusable item is found, the designer can explore the context of this item in order to understand its design and assess its reusability. The designer can explore the item’s project context using the project context explorer that uses a fisheye lens metaphor to support systematic exploration of related items in the corporate memory. The designer can explore the item’s evolution history using the evolution history explorer based on a storytelling metaphor to visualize the evolution of this item, the rationale, and team interactions driving this evolution.
Ongoing work focuses on usability tests to assess the extent to which the CoMem interface offers an improvement in the reuse process compared to traditional interfaces to support different kinds of tasks, e.g., knowledge retrieval vs. exploration, different types of users, e.g., expert vs. novice, and different sizes of repositories.

Acknowledgements

This research is supported by the UPS Endowment Fund at Stanford University. The authors thank Dr Gregory Luth, Professor Helmut Krawinkler, and Professor Terry Winograd. CoMem uses the Treemap Java Library by Christophe Bouthier, and the Latent Semantic Indexing software by Telcordia Technologies.

References


