Automation in Construction 10 (2001) 687–699
AUTOMATION IN CONSTRUCTION
www.elsevier.com/locate/autcon

A hybrid representation of architectural precedents

Ulrich Flemming [a,*], Zeyno Aygen [b]

[a] *School of Architecture and Institute for Complex Engineered Systems (ICES), Carnegie Mellon University, Pittsburgh, PA 15217, USA*
[b] *Trilogy Software, Inc., Austin, TX 78730, USA*

Abstract

We present a hybrid representation of architectural precedents that separates precedent instances from the concepts they embody, where the concepts are defined in terms of multiple classification taxonomies. The representation allows us to combine the classical view of concept acquisition with aspects of the probabilistic and exemplar views and to organize the database into the equivalents of "episodic" and "semantic" memory. A first application and test context is provided by the Software Environment to Support Early Building Design (SEED), where precedents are used as prototypes and cases. But the representation is flexible enough to support the use of precedents in other application contexts.

Keywords: Architectural design; Precedents; Concept formation; Design databases; Design prototypes; Case-based design

1. Introduction

In the present paper, we deal with architectural precedents under a computational perspective. We specifically address the following question: How should we represent architectural precedents so that multiple uses can be made of this representation in design practice, research or education? Implicitly, we address a larger question: How should — or, at least, can — we think about precedents under a computational perspective?

Precedents have been collected and published long before the advent of computers. A well-known "precedent" is Durand's "parallele", a collection of "buildings of all types, ancient and modern, remarkable for their beauty, for their grandeur and for their uniqueness," drawn in standard projections and presented without comments [6]; that is, readers could use the collection as they pleased. More recent examples are those of Clark and Pause [5], who present precedents augmented by analytical diagrams clearly intended for educational use; or the floor plan atlas of housing precedents [17], a workbook for practitioners. Whatever their intended use, these paper-based collections have several features in common; for example, they can be easily browsed and — at least the last two — easily be carried around. Why then would we want to create a collection of precedents in a computable representation?

Suppose we have a database of architectural designs planned or executed in the past that contains, among many other examples, a description of H.H. Richardson's Allegheny Courthouse in Pittsburgh. The following is a selection of situations in which we may want to retrieve this description:

- as an instance of a courthouse when we are studying courthouses as a building type or when

* Corresponding author.
E-mail addresses: ujf@cmu.edu (U. Flemming), zeyno_aygen@trilogy.com (Z. Aygen).

0926-5805/01/$ - see front matter
PII: S0926-5805(00)00086-8

we have to plan a new courthouse and want to learn from the past;

- as an instance of a courtyard building, again to study courtyard buildings in general or when we are contemplating a courtyard scheme for a specific context;
- as a public building organized around a courtyard when we consider designing such a building (not necessarily a courthouse);
- as a large courthouse (containing, for example, at least *x* courtrooms) when we want to prescreen the hits in the database;
- as an H.H. Richardson building in Pittsburgh when we plan a trip to that city and are an H.H. Richardson fan.

In addition, we may want to retrieve parts of the building, for example, an individual courtroom cluster independently of the overall design. This can happen in at least two complementary situations: (i) when we want to find out how large a typical courtroom is or which ancillary spaces it requires; or (ii) when we know which ancillary spaces we need and are interested in precedents that organize these areas effectively. Note that this list could be extended; for example, a user of the database may discover that the courthouse is also a good example of a building constructed of load-bearing masonry walls; she may then add this classification to the instance. But note also that we cannot anticipate all possible situations under which the instance may be of interest or all categories under which users may want to classify it.

We observe that not all this could be done with a paper-based collection or could be done only with great difficulties. Take multiple classifications as an example. The floor plan atlas mentioned above introduces multiple classifications of apartment buildings (plan organization, overall access, urban context), each of which may provide a primary browsing focus for a reader; but in organizing the book, the editor had to decide on a dominant classification and group the precedents accordingly. She then used the other classification categories in a fixed order to form subcategories under the primary ones. Browsing becomes increasingly tedious as we use lower-level categories. Inferences based on subsumption are completely left to readers; for example, if they look for residential structures, they have to *know* that housing is a subclass of that building type and that the atlas may consequently be of interest. The atlas with its fixed and limited taxonomy also does not support searches for subparts under their own classification (like L-shaped or U-shaped kitchens). Readers have to construct such groupings on their own by perusing the collection, and once they are done, the collection cannot be augmented with new types and categories. Computer-based collections of precedents, on the other hand, can be designed to have none of these limitations. We can therefore envision benefits from a precedent database implemented on a computer, at least in principle.

How should then a precedent be represented in a computational database conceived as broadly as outlined here? We can immediately derive some requirements from our initial example: (1) precedents should be multiply classified under a classification scheme that can be extended or otherwise modified over time; (2) a classification category may belong to a classification hierarchy and the database has to support subsumption inferences (e.g., it has to know that a courthouse is a public building); (3) it should be possible to combine classifications with instance attributes when we query the database (note that the category "large" may be highly dependent on context and cannot be anticipated in an a-priori classification; note also that if we cannot find a complete match between desired attribute values, we may be interested in precedents that come closest); (4) it should be possible to query on attribute values alone (for example, there is no need to classify the courthouse in our example as an H.H. Richardson building if the architect's name is stored as an attribute in the first place); (5) it should be possible to query on the structure of the precedents, for example, its part-of relations; (6) all of the above queries should be available for an overall design and its parts; and (7) it should be possible to combine all of the available query categories in a flexible indexing-for-retrieval scheme.

We describe in Sections 3–5 a representation of precedents that satisfies all of the above requirements except one. The scheme is highly influenced by Tulving's [20] memory model (Section 2.1) and Smith and Medin's [18] survey of the literature on concept acquisition (Section 2.2). We also surveyed