Introduction to This Special Issue on Design Rationale

John M. Carroll
IBM T. J. Watson Research Center

Thomas P. Moran
Xerox Palo Alto Research Center

Design research has traditionally been concerned with systematizing the design process—its tools, techniques, methods, and management—for artifacts and their specifications. This research has changed dramatically over the past 25 years. The design methods of the 1960s sought to provide general representations and formal decomposition and clustering techniques but failed in the end to be applicable to real problems (see Jones, 1970). In the 1970s and 1980s, there was a growing recognition that design is not just the solving of difficult problems but a kind of problem solving with distinctive properties: problem stages do not exist as such (or completely overlap, depending on how you want to look at it), there are never singularly correct solutions, the range of potentially relevant considerations is nearly always unbounded, side effects and interactions deflect even the most considered design moves, and so forth (see Cross, 1984).

This led to what Rittel (cited in Cross, 1984, p. 318) called “second-generation” design methods: methods that assume distributed expertise, the need for discovery, and the centrality of argument and multiple perspectives in all design work. *Design rationale* is a second-generation method that has

Authors' present addresses: John M. Carroll, User Interface Institute, IBM T. J. Watson Research Center, P.O. Box 704 Yorktown Heights, NY 10598; Thomas P. Moran, Xerox Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304.
recently become a focus of design research in human–computer interaction. It goes beyond merely accurate descriptions of artifacts, such as specifications, and articulates and represents the reasons and the reasoning processes behind the design and specification of artifacts. In this special issue, we are interested in a particular domain of artifacts and a particular orientation to their design, namely, computer and information systems that are explicitly designed with use and users in mind.

There are several motivations for constructing explicit design rationale: (a) to support reasoning processes in design, (b) to facilitate communication among the various players in the design process (designers, implementers, maintainers, users, etc.), and (c) to further the cumulation and development of design knowledge across design projects and products. Design rationale research addresses the issues involved in capturing, articulating, representing, and using explicit rationale for these various purposes. The goal of this research is to provide improved concepts, methods, and tools for design.

Much of the current work on design rationale in human–computer interaction is built on the seminal issue-based information system (IBIS) methodology developed by Rittel (e.g., Rittel & Webber, 1973). This method seeks to capture the issues that arise in the course of design deliberation, along with the various positions (or alternatives) that are raised in response to issues and the arguments for and against the positions. Keeping track of such a network of local deliberation produces a structured record of the design process that supports the recall of decisions and their rationale. This makes the decisions more understandable to designers at a later time and exposes them to reflection and reconsideration. Because of the broad influence of his work, we are dedicating this special issue to Horst Rittel, who died July 8, 1990.

The current research on design rationale pushes in several different directions and explores different viewpoints about the nature and use of design rationale. This special issue contains a representative sample of such work. Most of the current research is concerned with the representation of rationale, both its form and content. Although there is a shared goal of codifying rationale, the approaches being explored are different. Some emphasize the capture of rationale as a by-product of the design process, whereas some stress that the rationale itself must be constructed. Some emphasize the variety of considerations that should be incorporated into rationale, whereas others are concerned with deriving rationale from a scientific base (more rigorous, but narrower in scope). Other research is concerned with how design rationale relates to design practice. This includes explorations of tools for capturing rationale and design tools for using it, as well as empirical studies of the use of design rationale in real design situations.

The first article in this special issue, by MacLean, Young, Bellotti, and Moran, is perhaps the most tutorial. It describes the QOC (Questions,
Options, Criteria) notation for representing design rationale and presents small examples. The authors characterize their approach as “design space analysis,” that is, expanding the space of possible designs and the rationale for choosing within this space. In contrast to the IBIS notation, which is geared to capturing deliberation as it happens, the authors stress that QOC representations must be constructed by the designer as an act of reflection on the state of the design process. The article presents an empirical study of a design session and, from this, further analyzes the kinds of justification found in design.

The article by Lee and Lai explores in detail the requirements for the expressive adequacy of representations for design rationale by laying out the various aspects of knowledge surrounding rationale in design. They present DRL (Decision Representation Language) and show how it represents the various aspects of rationale knowledge. By accurately representing rationale, they show we can begin to provide useful computations for designers.

Carroll and Rosson approach design rationale from a different concern—to embed psychological theory into the evolutionary development of systems. They use the theory to extract implicit claims about the ways various features of artifacts support users' tasks or impact on their experience. By analyzing these claims and their limitations, they are able iteratively to improve the design of the artifact in a more principled way. They illustrate this approach with a case study of the design of integrated browsing facilities in Smalltalk for learning the system and for reusing code.

Lewis, Rieman, and Bell present a challenge to the notion that design rationale should be conceptualized as abstract issues, criteria, or principles. They propose a focus on representative concrete problems that the system is intended to address. These problems are used to guide exploration of the design space and evaluation of alternative designs. They illustrate this approach with a case study of designing a graphical programming system.

The last two articles build on the IBIS notation and explore design tools. Conklin and Yakemovic focus on capturing design rationale without disrupting the design process by developing tools for representing and managing an issue base, an indented text IBIS (itIBIS), and a graphical IBIS (gIBIS). They present a field trial of a real industrial design project in which a considerable body of rationale was captured over many months. They showed not only that this could be done, but also that the rationale had considerable value in catching and managing several design errors.

Fischer, Lemke, McCall, and Morch built a suite of exploratory design tools in which design rationale plays a central role. A large IBIS issue base serves as a knowledge base of generic design knowledge for their experimental domain of kitchen design. The rationale is integrated with a construction “kit” (called JANUS) in which users can lay out and explore kitchen designs; the rationale is brought forth when the user needs to reflect on the state of the
design. The authors discuss the role of argumentation in design and describe an architecture for an integrated design environment and the place of design rationale in it.

We are still in the early stages of design rationale research. For example, most of the approaches have only been applied in one or a small number of design domains, which confounds domain properties with properties of the approaches. Very little has been done to study and assess the use of design rationale in real design projects, that is, large projects that do not involve the researchers themselves as designers (although the Conklin & Yakemovic article is an important first step in this direction). Indeed, there are still many fundamental issues to be understood: the relative utility of unobtrusively capturing rationale versus using the construction of rationale as an explicit design activity, how rationale representations are treated in organizational settings, and so on. The articles in this special issue represent the beginnings of a broad and significant area of human–computer interaction research and development.

REFERENCES


ARTICLES IN THIS SPECIAL ISSUE
