To appear in: *The Encyclopedia of Computer Science and Technology*
A. Kent and J. G. Williams, (Eds.). New York: Marcel Dekker, Inc. (1996)

Design Argumentation as Design Rationale

Simon BUCKINGHAM SHUM

Knowledge Media Institute
The Open University
Milton Keynes
MK7 6AA, U.K.
Tel: +44 1908-655723
Fax: +44 1908-653169
S.Buckingham.Shum@open.ac.uk

Contents

I. The idea of design rationale 1

II. Design rationale as argumentation 2
- A. The vision, and a representation 2
- B. Developing the vision 3
- C. Argumentation-based DR notations 6
- D. Embedding argumentation in the design artifact 13

III. Argumentation-based DR in use 15
- A. Recording design argumentation can assist ongoing design work 15
- B. Access to argumentation about past decisions can be valuable 18
- C. Creating design argumentation can also impede ongoing design work 19
- D. Summary of empirical studies 20

IV. Lessons learnt, and the future agenda 21
- A. Negotiating the cost/benefit trade-off: notations and tools 21
- B. Who should use it, and how? 24
- C. The importance of training 25
- D. Towards model-based DR for user interfaces 26
- E. Building DR on existing organizational practices 26
- F. Politics and organizational culture 27

V. References 28

VI. Bibliography for further reading 32

Design Argumentation as Design Rationale

Abstract

A design rationale (DR) is a representation of the reasoning behind the design of an artifact. In recent years, the use of semiformal notations for organising arguments about design decisions has attracted much interest within the software engineering and human-computer interaction communities, leading to the development of a number of DR notations and tool environments. This article begins by reviewing the motivation for expressing DR as design argumentation, and then surveys evidence from design studies to show when and how it can be productive to construct explicit design argumentation during design. The article then discusses practical cognitive, organizational and technological factors which could facilitate the uptake of design rationale systems.

Keywords: design rationale, argumentation, process-oriented software engineering, organizational memory, computer-supported collaborative work

I. The idea of design rationale

A design rationale (DR) expresses elements of the reasoning which has been invested in the design of an artifact. A DR answers *Why...?* questions of different sorts, depending on the class of DR represented. Recently, interest in DR has been growing amongst the development and research communities in both *software engineering* [1-6], and *human-computer interaction* [7-12] .

It has been argued for some time [2,13] that at present the software development process is largely *artifact oriented*, that is, the emphasis is on generating and tracking design artifacts throughout the lifecycle—requirements, specifications, prototypes, user documentation—culminating in the final system itself. However, the *process* by which these artifacts assume their final form—in particular, the reasoning underlying the decisions which shape the products of design—often remains implicit (e.g. hidden in minutes of meetings, design notebooks, e-mail archives, or designers' memories), and is consequently difficult to recover, and even harder to reuse. Forward looking design organizations are increasingly recognising the value of finding ways to capture and reuse the effort and expertise which has been invested in development projects, in order to learn from past mistakes, avoid re-inventing the wheel when a similar problem has already been solved, and support the latter stages of the design lifecycle when questions about early design decisions often cannot be answered. (It has been estimated, for instance, that up to 70% of lifecycle costs are incurred in the maintenance phase, and that up to half of system maintainers' resources are taken up with reverse engineering designs in order to make appropriate changes [2,14]).

The challenge for DR research is to find the most helpful, accessible representations for expressing design reasoning, which whilst minimising the non-productive effort required to create it (encoding useful DR is bound to require some effort), maximises its utility for those designers creating it, and for others subsequently. I shall shortly describe how the expression of design rationale as *arguments* about *design issues* has been attracting interest, and shall refer to these approaches as *argumentation-based DR*. Proponents of approaches such as this argue that they have the potential to play several useful roles in design:

- structuring and analysing novel design problems
- keeping track of decisions

- communicating design reasoning within projects
- maintaining consistency in decision-making
- tracking progress in projects and identifying recurring and unresolved issues
- supporting the building of cumulative design knowledge, through reusing DR
- assisting the integration of perspectives from multiple stakeholders on decisions.

The remainder of this article is organised as follows. Firstly, historical and conceptual background is provided to show why argumentation, particularly in graphical, hypertext form, has come to be used for expressing design rationale. Secondly, a number of case studies are described, and critiqued, to show how argumentation-based approaches to DR have been used in software development. Finally, practical lessons from these studies are suggested, and key issues which need to be addressed are highlighted.

II. Design rationale as argumentation

In this section, three strong influences are identified which serve as historical and conceptual roots to the current interest in argumentative DR. Examples of argumentation-based DR notations are then introduced, to acquaint the reader with the representations and tools of concern.

A. The vision, and a representation

Possibly the earliest 'mission statement' which argumentation research today continues to pursue was set out by Englebart [15], in his seminal paper, *A Conceptual Framework for the Augmentation of Man's Intellect*. This foresaw the day when computers, far more powerful than those available at the time of writing, would enable people to overcome some of the limitations of their cognitive faculties by manipulating externalised "concept structures":

> A concept structure (...) is something that can be designed or modified, and a basic hypothesis of our study is that better concept structures can be developed—structures that when mapped into a human's mental structure will significantly improve his capability to comprehend and to find solutions within his complex-problem solving situations.
>
> *Englebart, 1963 (reprinted in [15], p. 54)*

This vision of designing better computer-supported notational structures with which to think marks the beginning of the trail which much of the work surveyed in this paper has continued to explore. We shall not discuss this work in any more detail, except to note that Englebart's framework[1] embraced not only cognitive tools, but notational and training issues, facets of DR research on which we focus in more detail towards the end of this paper.

The second 'root' concerns the representational form for arguments. *The Uses of Argument* by Stephen Toulmin [16] was originally written as a challenge to the dominance in philosophy of formal, Aristotelian logic. Toulmin's aim was to develop a view of logic which was grounded in the study of reasoning practice. Taking argumentation as the most common form of practical everyday reasoning, he posed the question, "what, then, is involved in establishing conclusions by the production of arguments?" His analysis of the logical structure of arguments led to a graphical format for laying out the structure of arguments, a representational approach reflected in much subsequent argumentation work.

[1] Englebart developed what he called the "H-LAM/T" framework (Humans using Language, Artifacts, and Methodology, in which they are Trained).

The notation consists of five components and four relationships (Figure 1). According to the analysis, whether or not it is made explicit, all arguments logically comprise a fact or observation (a *Datum*), which via a logical step (a *Warrant*), allows one to make a consequent assertion (a *Claim*). The Warrant can be supported by a *Backing* if necessary (why the assumed Warrant is valid), and the Claim qualified with a *Rebuttal* (specifying exceptions to the rule).

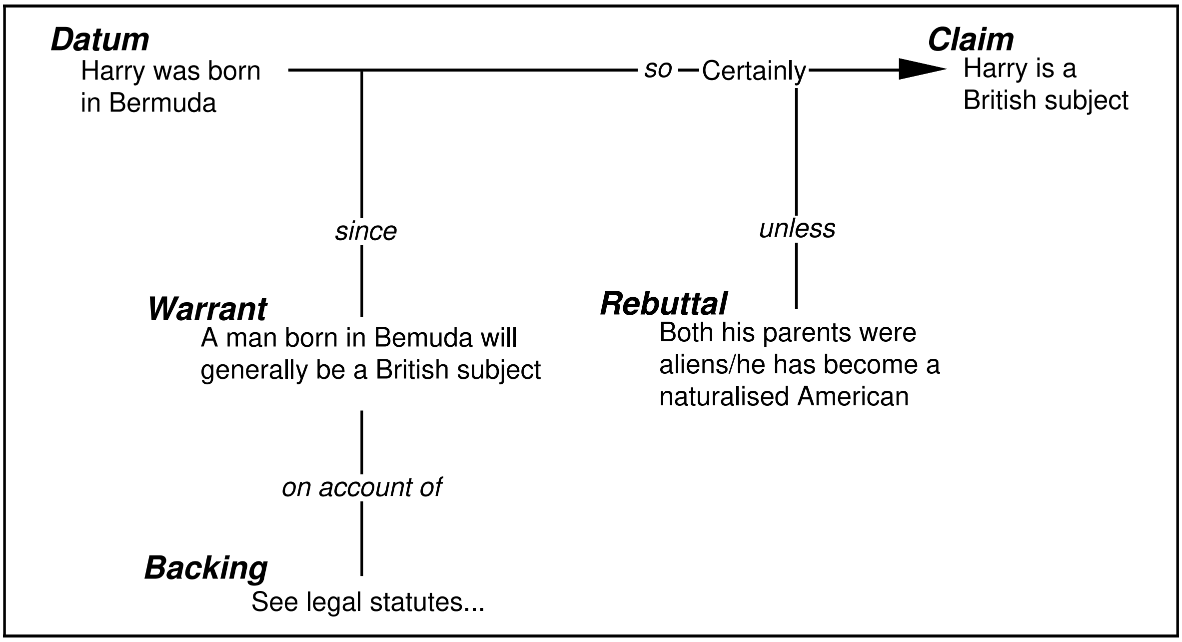

FIGURE 1: An example of Toulmin's graphical argument structure [16] .

The purpose of this review is to set the conceptual background for the use of argumentation-based DR, so the considerable amount of debate which Toulmin's model stimulated is not of immediate relevance. Further discussion can be found in Hair and Lewis [17], and Newman and Marshall [18] who set the model in context within the wider philosophical literature on argumentation. It is sufficient to note that Toulmin's work, through its use of a semiformal graphical representation, was the first step towards the argumentation schemes being used at present.

B. Developing the vision

Englebart's call helped set in process research efforts in numerous locations. One of the most influential of these was based at the Xerox Palo Alto Research Center, where research into computational support for idea structuring tasks like writing and argumentation was just beginning in the early 1980s. In a forward looking paper, Brown [19] focused on the potential value of representing the *process* by which ideas and artifacts develop, as well as the final product:

> *Process versus product*
>
> By focusing in the design of empowering environments on the product of a creative effort, we are missing the real source of power for computer-based tools: the computer can record and represent the process underlying the created product. By making explicitly available to the user the series of steps and missteps that leads to the creation of a particular object or result, we create a basis on which to build extraordinarily powerful editing, merging, undoing, and transforming tools. Tools designed to manipulate this "historical" information, or *audit trail*,

> can be used to carry out intellectual and creative tasks of great complexity. (p.182, original emphasis)
>
> (...) By maintaining an explicit audit trail of the steps that led to the created "object," not only can more powerful tools be constructed but, perhaps more importantly, the ability for one to understand what another has done is greatly enhanced. (p.183)

Complementary to this challenge, and perhaps as one possible solution, was a subsequent interest in the potential of tools for manipulating and filtering arguments [20]:

> Current communications tools and methods force the crafting of complex arguments into linear form for presentation, so that the web-like connections among ideas is hidden from view, making it difficult to see alternate interpretations and points of view. (...) As a result many of the underlying ideas, arguments and assumptions either remain implicit or are lost altogether. But consider the possibility of crafting new information tools to capture not just conclusions and the view of matters that supports them, but to allow the explicit representation of underlying assumptions and argument structures. (p. 484)

It was noted in particular that work was needed on developing *notations* with an appropriate vocabulary for the task domain:

> To accomplish these goals, we need a taxonomy of epistemological links for relating ideas, as well as link-related filters. That is, we must now think about giving users access to and utilisation of not just undifferentiated links, but links with appropriate kinds of labels. (p. 485)

Representation of these two key elements – deliberation process and arguments "with appropriate kinds of labels" – are precisely the concerns dominating DR research. In a subsequent project on computational support for meetings, Stefik *et al.* [21] explicitly set out to substantiate the first claim examined in this paper with a tool called *Argnoter*:

> [Design] is essentially a dialectic between goals and possibilities... in collaborative design tasks, this interaction and tension between goals and alternatives must play itself out in the communications among collaborators. ...A major theme of Argnoter's design is that alternatives be made explicit: Proposals themselves are explicit, as are assumptions and evaluation criteria. (p. 38)
>
> A major working hypothesis behind the design of Argnoter is that making the structure of arguments explicit facilitates consensus by reducing uncommunicated differences. (p. 40)

The motivation behind Argnoter was clearly that of a 'group DR' tool – a way to represent design arguments explicitly, but with the group process adding another dimension. It was hypothesised that the process of striving to agree on rankings and assumptions in Argnoter would help designers recognise where their differences lay (cf. Claim 1 in this paper). Although Argnoter was never built [22], we note this work in passing since it was one of the earliest specifications for a DR tool using modern technology.

To conclude this section, it is necessary to recognise the seminal work of Horst Rittel, which has strongly influenced the adoption of semiformal argumentation as a representational basis for DR [23,24][2]. Several DR research efforts (e.g. [9,26,27]) have described in some detail how their approaches derive from Rittel's concept of *argumentative design*, which was motivated, at least in the first instance, by different concerns from those of Englebart and Toulmin.

---

2 Other reviews of Rittel's philosophy and of IBIS can be found in [9,25].

Rittel *et al.* made two important contributions to our understanding of design and the role of argumentation. Firstly, they characterised many design problems as ‘wicked,’ in contrast to ‘tame’ or ‘benign’ problems which can be modelled computationally:

> “Wicked and incorrigible [problems]...defy efforts to delineate their boundaries and to identify their causes, and thus to expose their problematic nature” [24].

Such problems lack a single, agreed-upon formulation or well-developed plans of action, are unique, and have no well-defined stopping rule, because there are only ‘better’ or ‘worse’ (rather than right or wrong) solutions. Closure is often forced by pragmatic constraints (e.g. managerial or political) rather than ‘rational scientific’ principles. Subsequent research into the properties and defining features of ‘ill-structured’ design problems and the problem solving strategies required to solve them have since corroborated Rittel’s characterisation of such problems [28-30].

The second contribution made by Rittel *et al.,* was their response to the challenge of wicked problems. Such problems could not be solved by formal models or methodologies, classed by Rittel as the ‘first-generation’ design methodologies. Instead, an *argumentative* approach to such problems was proposed (a second-generation design method). The essence of this perspective was that an open-ended, dialectic process of collaboratively defining and debating issues is a powerful way of discovering the structure of wicked problems:

> “First generation methods seem to start once all the truly difficult questions have been dealt with already (...) The second generation deals with difficulties underlying what was taken as input for the methods of the first generation.”
>
> “[Second generation] methods are characterised by a number of traits, one of them being that the design process is not considered to be a sequence of activities that are pretty well defined and that are carried through one after the other, like ‘understand the problem, collect information, analyse information, synthesise, decide,’ and so on...”
>
> “My recommendation [for the future of design methodologies] would be to emphasise investigations into the understanding of designing as an argumentative process ... how to understand designing as a counterplay of raising issues and dealing with them, which in turn raises new issues, and so on...”
>
> “[Argumentative design] means that the statements are systematically challenged in order to expose them to the viewpoints of the different sides, and the structure of the process becomes one of alternating steps on the micro-level; that means the generation of solution specifications towards end statements, and subjecting them to discussion of their pros and cons.”
>
> *(from an interview with Rittel, 1972 [31])*

This perspective motivated the development of the Issue Based Information System (IBIS) as a medium to encourage the open deliberation of issues. The three key IBIS entities were Issues, Positions and Arguments, which could be linked by relationships such as *supports, objects-to, replaces, temporal-successor-of, more-general-than,* and their converses. Visualised as a graph, an IBIS grows into a network as more Issues are posted and debated (Figure 2).

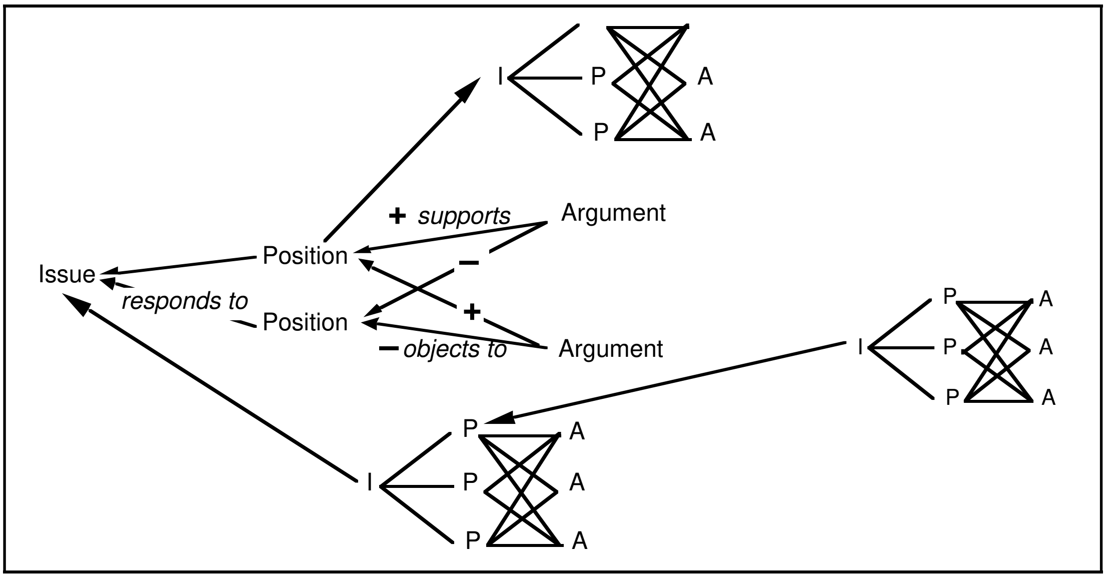

FIGURE 2: The basic IBIS structural unit of Issues, Positions and Arguments, developed to support Rittel's argumentative approach to wicked problem solving. (Reproduced with permission from [48], Academic Press Ltd.)

To summarise, in conjunction with Englebart, Toulmin, and Brown, Rittel contributes to the conceptual and historical context in which to understand current work on argumentation in design. Rittel's work established the most explicit bridge between design and argumentation. The argumentative approach to design elevated the importance of the *process* of understanding a problem from its minor status as a preliminary step to using first-generation design methods, to the central activity in tackling wicked design problems. Rittel *et al.,* argued that a particularly powerful way to tackle such problems is by an 'argumentative approach, leading to IBIS, the first explicit representation for argumentation in a design context.

C. Argumentation-based DR notations

The idea that DR should be represented as semiformal arguments can be traced back to wider research into the development of computational support for reasoning. Given that the formulation and critique of arguments is a central activity in many key areas (research, design, writing, management, government), a challenge with many potential benefits has been to develop representational schemes which enable computers to assist in authoring, retrieving, evaluating, and modifying arguments.

The main hypothesis behind research into supporting argumentation was that by making the structure of arguments explicit, they could be more rigorously constructed and communicated [32], and these are still key goals in DR research. Differences between notations lie in the constructs chosen to capture arguments, which, in turn, determine the kinds of argument which can be captured, and the level of detail which can be expressed (fewer, more general types, or a larger number of more specific types). Several approaches to argumentative DR are now outlined to provide a flavour of this approach to DR. For fuller descriptions, see [12,33].

Design Space Analysis (DSA) using the QOC notation (*Questions*, *Options*, and *Criteria*) has been proposed by MacLean *et al.* [34]. *Assessments* are the relationships between Options and Criteria (*supports* or *objects-to*), and *Arguments* are used to conduct debate about the status of the above entities and relationships. These elements are shown in their graphical form in

Figures 3a and 3b (a matrix of Options against Criteria can also be used). The DSA approach to making a design more transparent is to create "an explicit representation of a structured space of design alternatives and the considerations for choosing among them. Thus a particular artifact is understood in terms of its relationship to plausible alternative artifacts" [34], p.203). Thus, DSA is a process of identifying key problems (Questions), and raising and justifying (via Criteria) design alternatives (Options). A DSA-oriented DR emphasises the cumulation and reuse of knowledge through the articulation of coherent design spaces within which different solutions can be located (e.g. as implemented in other systems).

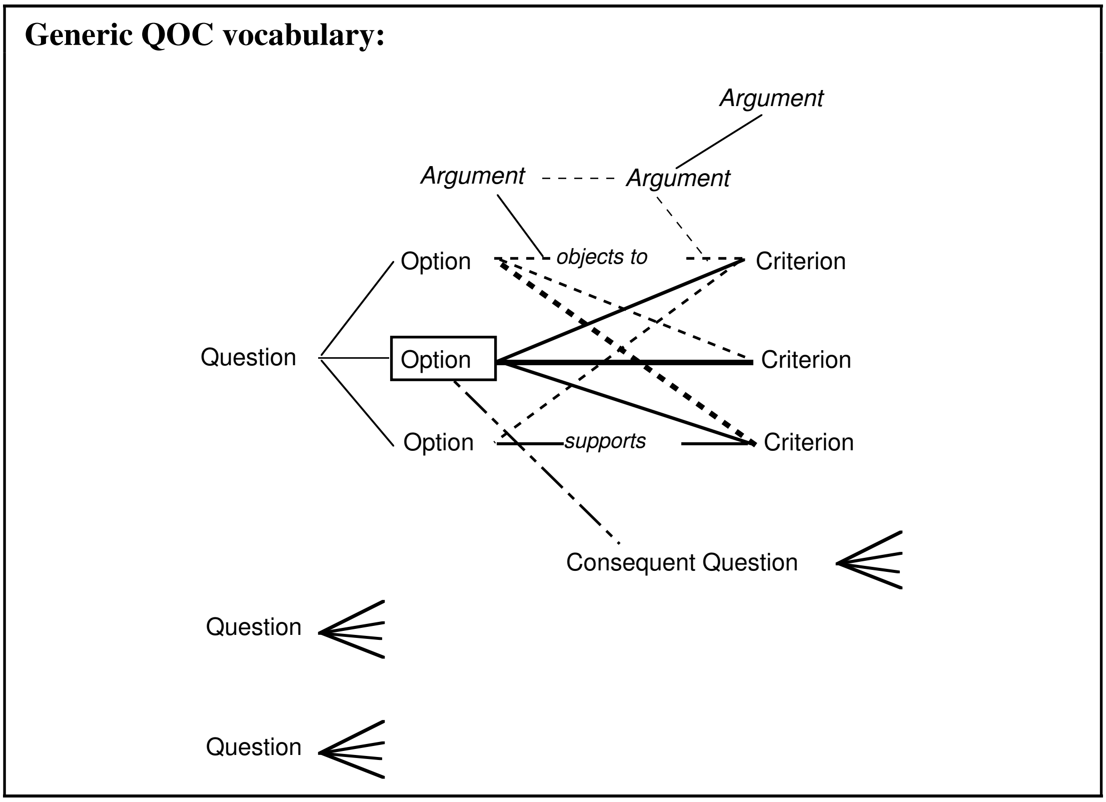

FIGURE 3a: The generic QOC notation as used in Design Space Analysis. Note the use of link-thickness to indicate relative weights of Assessment.

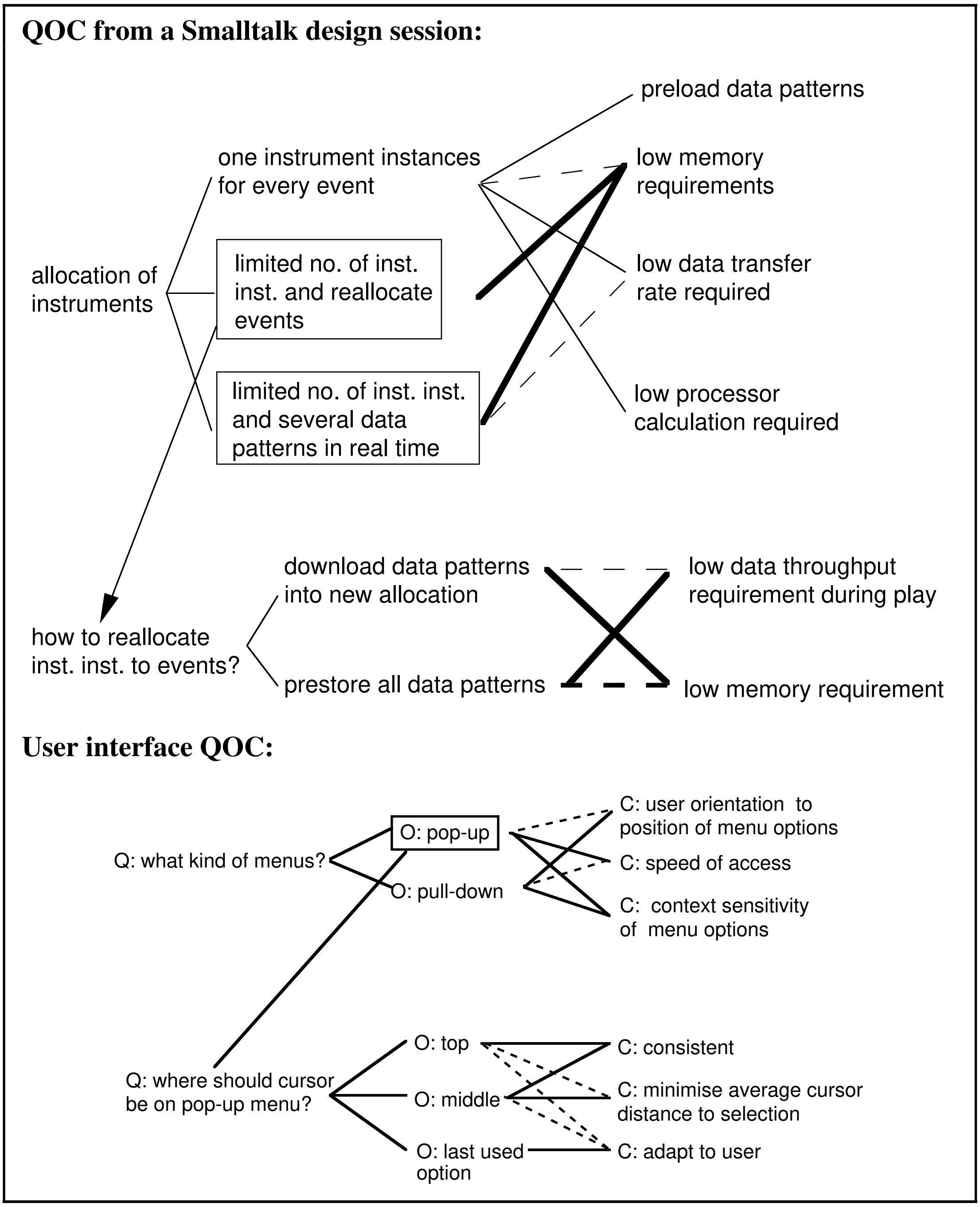

FIGURE 3b: Two examples of QOC representing DR for very different levels and domains of design. (Reproduced with permission from [48], Academic Press Ltd.)

The IBIS notation [24,31], extended for the graphical IBIS tool gIBIS [25,26] is similar superficially; the main difference lies in the IBIS emphasis on capturing more of the design *process* as it unfolds in meetings (as opposed to *retrospectively rationalising* the DR as in QOC/DSA), and for a *single* design (as opposed to clarify the dimensions defining the *space* in which the design sits in relation to other possible designs). The work on gIBIS has led to the development of the QuestMap hypertext groupware system, a commercial argumentation-based DR tool [35] (Figure 4).

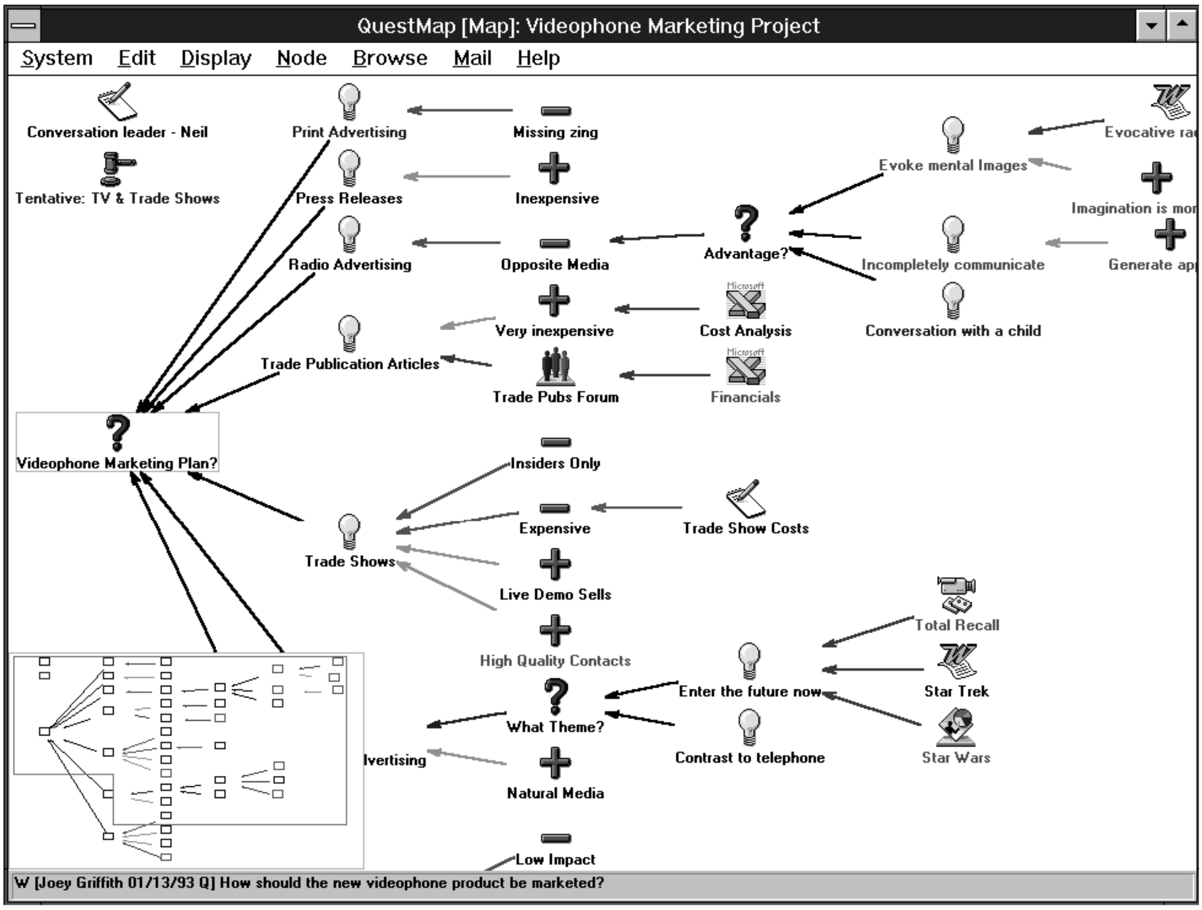

FIGURE 4: A screen from the QuestMap argumentation-based design rationale system [35]. Based on the gIBIS research tool [26], this is a commercial groupware hypertext system, providing design projects with a way to conduct extended, public debates about issues, make decisions, and capture the rationale behind them.

Lee and Lai [36] have proposed a notation which extends QOC and gIBIS. Decision Representation Language (DRL) was developed to explore the potential for computational support within the semiformal paradigm (Figure 5a).

SIBYL [10] (Figure 5b), the tool supporting DRL, provides a number of computational services, including dependency management (monitoring decisions which depend on each other), precedence management (other decisions sharing same Goals), viewpoint management (arguments sharing common assumptions), and plausibility management (the strength of supporting argumentation for an alternative).

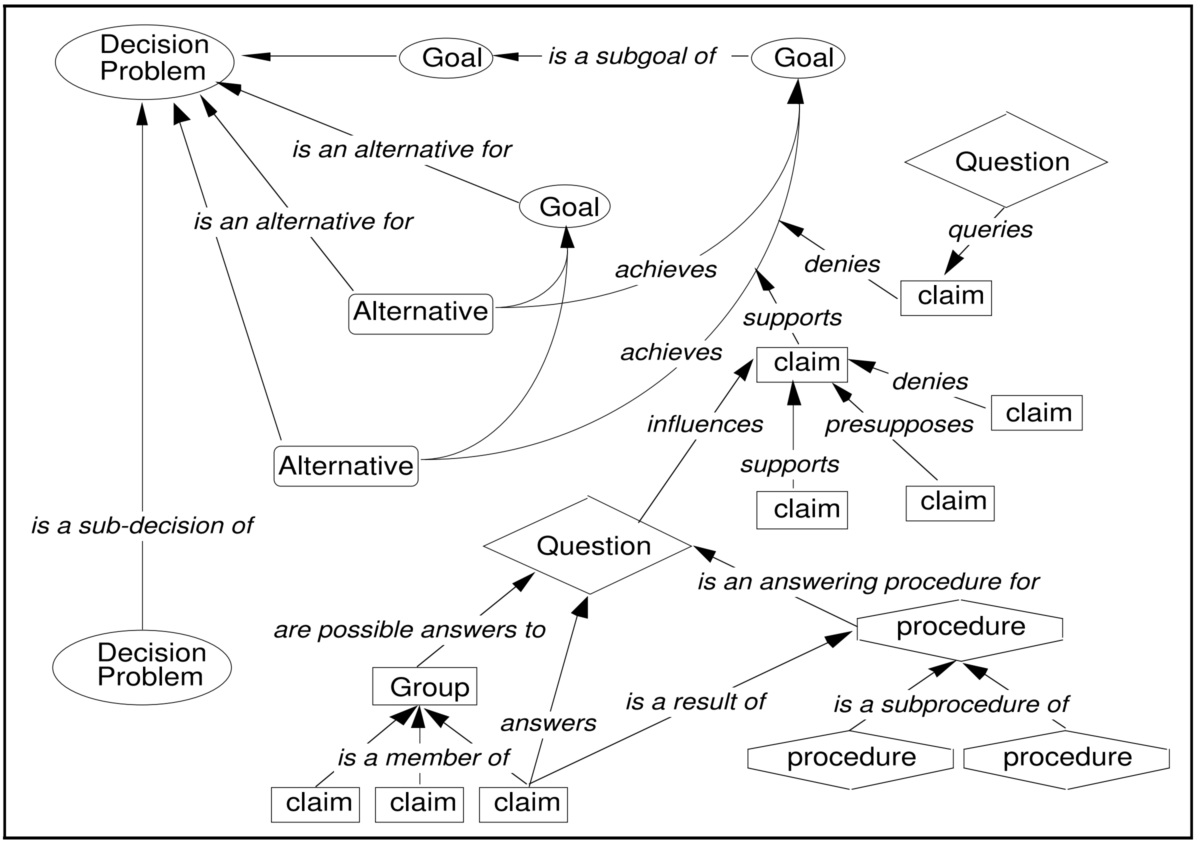

FIGURE 5a: The DRL notation (Reproduced with permission from [48], Academic Press Ltd.)

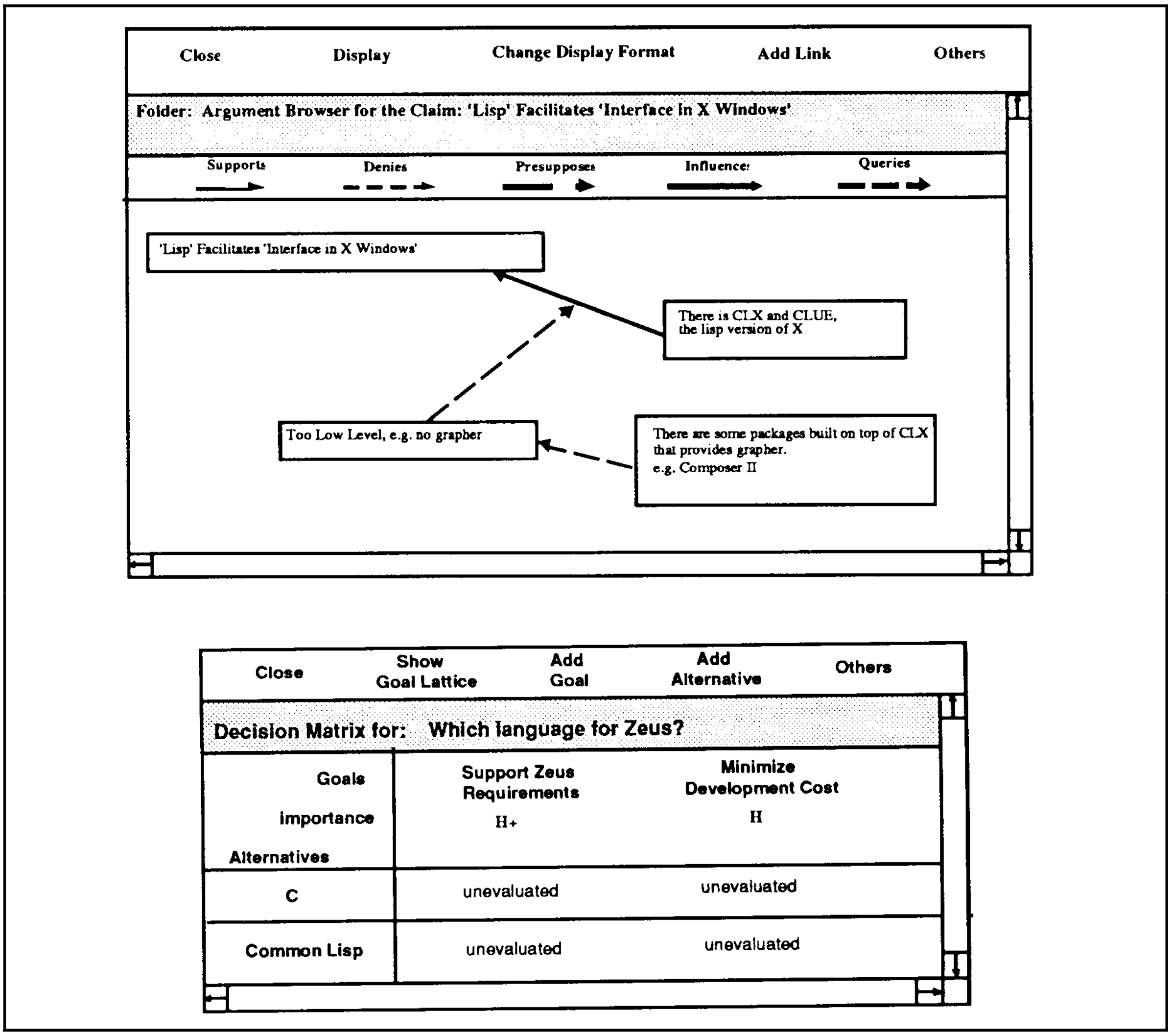

FIGURE 5b: Two views of a DRL network (an Argument browser and a Decision matrix) using SIBYL, its support tool (from [37]).

Potts and Bruns [38] presented a model for relating entities in a software engineering method to IBIS-based design deliberation. In this model, a design history is made up by the network of intermediate *artifacts* produced en route to a finished design, artifacts being specifications or design documents, which are *derived* from one another through *deliberation nodes* (represented as Issues, Alternatives, and Justifications). The particular artifacts depend on the software method being supported. Based on IBIS, Issues derive Alternatives, which derive justifications. The key difference from other DR representations, namely the integration with software engineering methods, is achieved through deriving Artifacts from Alternatives. The syntax of the notation is summarised in Figure 6a.

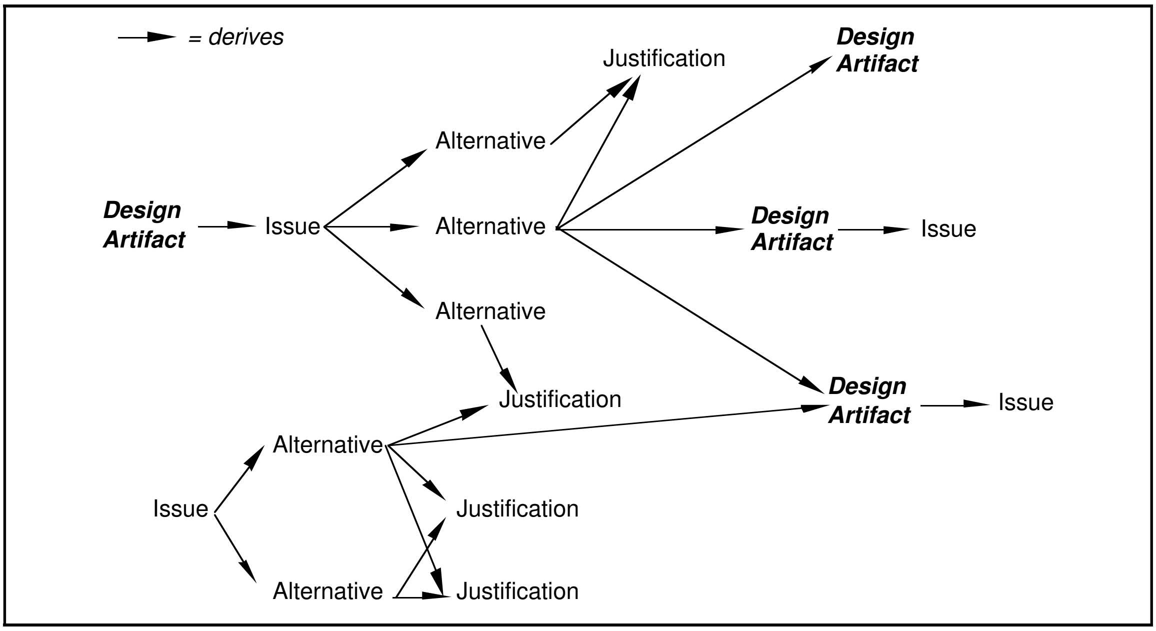

FIGURE 6a: Potts and Bruns representation for integrating design deliberation with design process artifacts.

Potts and Bruns proposed that this generic model should be tailored to support specific design methods, illustrating this with a worked example using the Liskov and Guttag method. A new entity specific to this method is the *Task*, anything performed by a procedural abstraction, which is incorporated into the IBIS method by raising a new Issue. Potts and Bruns' generic *Artifacts* are specialised into Liskov and Guttag's *procedural specifications*, and *data abstraction specifications* and *Issues* into *behavioural issues* and *encapsulation issues.* Figure 6b reproduces part of an analysis of a text formatter, to illustrate the adaptation to Liskov and Guttag. Understanding the model or design problem is not necessary; it should simply be understood that the generic model's entities are refined to accommodate a particular design method's vocabulary for deriving new artifacts. The transformations of Artifacts through deliberation was represented by Potts and Bruns using a hypertext system whose structure was verifiable by Prolog rules which could check the integrity of Liskov and Guttag syntax.

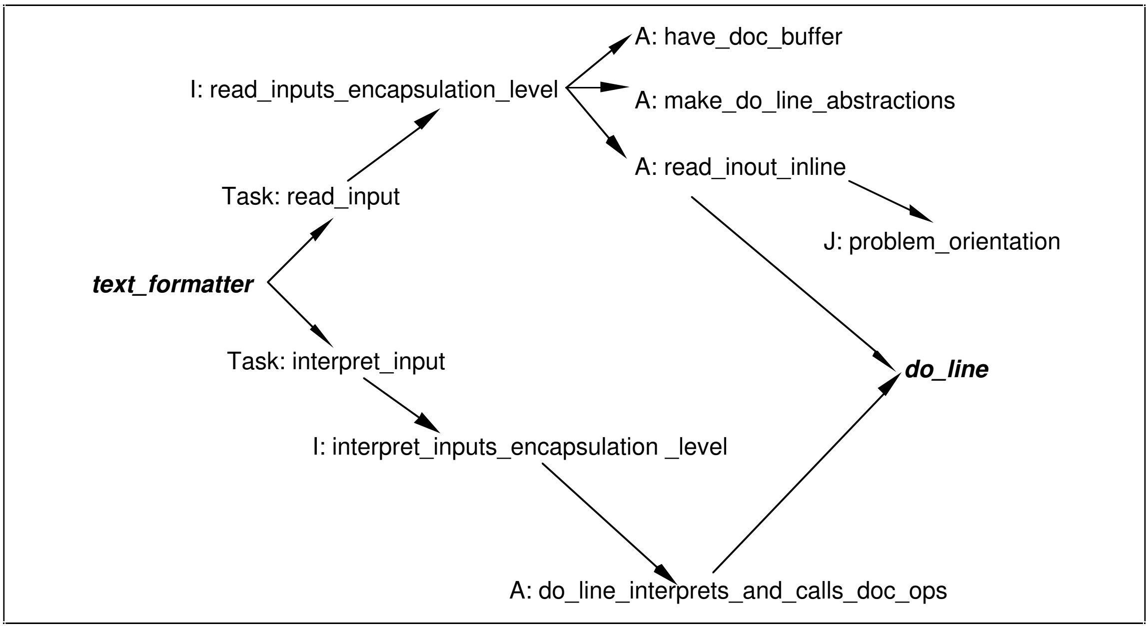

FIGURE 6b: Example of Potts and Bruns IBIS-based notation tailored to the Liskov and Guttag design method. Key: Artifacts in italics; I=Issue; A=Argument; J=Justification

The principle of showing the evolution of design artifacts through explicit argumentation has been realised most recently by Potts *et al.* in the form of Tuiquiao, a prototype requirements traceability tool [39].

D. Embedding argumentation in the design artifact

DR research is reflecting a growing recognition that argumentation *about* a design must be driven by implementational concerns at some level, or DR can remain too abstract or address problems of little consequence. As Fischer *et al.* [9] put it, "In a good design project, construction generates and regulates argumentation. Argumentation arises out of construction and is often tested by construction." (p.403). The artifact being 'constructed' may be any product of the design process, from initial requirements to code. The point is that being engaged in developing the artifact itself is different from reflecting on it.

This perspective is reflected in ongoing efforts to build software environments which integrate DR much more tightly with the artifact under development. For instance, in XNetwork [40] system network maintainers are able firstly to link informal rationale notes to network diagrams, but if an issue is judged to be important enough to capture more formally, then it can be added, with support from the system, to a semi-structured IBIS design rationale (Figure 7).

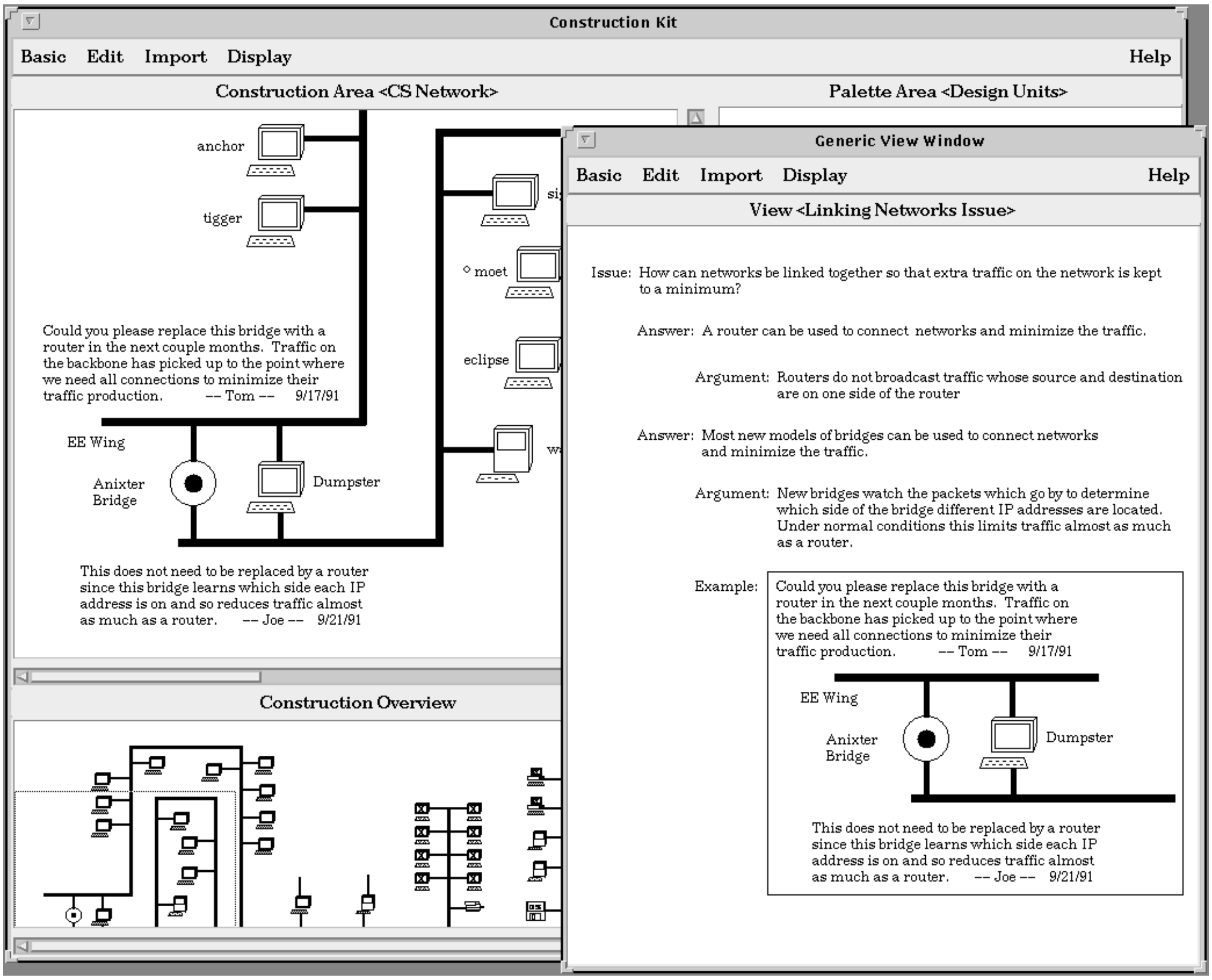

FIGURE 7: In XNETWORK [40], designers can communicate by attaching notes to design documents. As issues arise in the course of work (upper left figure) some may be worth capturing more formally (right hand window). The HyperObject Substrate [41] underlying XNETWORK can assist in this formalisation process by making suggestions as to how to structure informal texts.

Other approaches are exemplifed by Ryan [42] who has enriched the Smalltalk environment with QOC for class hierarchies, Carey and Crentsil [43] who embed QOC in user interface toolkit widgets, and Potts [44] who has considered different ways in which IBIS can be more closely linked to software engineering representations, in particular, providing rationale to track the evolution of design requirements [45]. Fischer *et al.* [9] and McCall *et al.* [46] have focused explicitly on the separation between Issue based deliberation and construction of the artifact, to the extent of modularising, but closely linking, these two components in their JANUS, and more recently, PHIDIAS design environments. Knowledge-based 'critics' monitor a CAD design, and when a guideline is violated, alert the designer, who can then display the PHI argumentation relevant to the problem, in order to assist reflective decision-making. Concepts developed from Schön's [47] analysis of expert designers have been a source of inspiration and rationalisation in these efforts to encourage reflection on the part of designers as they work.

III. Argumentation-based DR in use

In this section, a range of design case studies are considered in order to clarify the practical strengths and weaknesses of argumentation-based DR. This is not an exhaustive literature review; for a more detailed analysis see [48]. The design studies which are described are organised into three sections:

- evidence that recording design argumentation in a structured way can assist ongoing design activity;
- evidence that access to structured design argumentation about past decisions can be valuable;
- evidence of situations in which recording design argumentation in a structured way can also impede ongoing design activity.

Rather than merely reporting the claims that have been made for the efficacy of a particular DR approach, a critique is also made of the strength of evidence provided to support those claims. Conclusions from these studies are summarised at the end of this section.

A. Recording design argumentation can assist ongoing design work

Workstation case study. Conklin and Burgess Yakemovic [26,49] have described their use of an IBIS-based notation in a large industrial design project, developing software for a new workstation. In this project, they report that there was a positive impact on project team communication from using IBIS[3]. They propose that "a knowledge of the underlying form of the dialogue [i.e. rhetorical moves in IBIS] allowed the team to determine when the discussion had wandered from the intended topic" (Conklin & Burgess Yakemovic: p.375). That is, although discussion was not structured explicitly around the IBIS model, they suggest that familiarity with the IBIS model (recorded privately by a 'DR scribe'), helped maintain a shared awareness of the meeting's process. Designers did not generally make explicit reference to IBIS constructs like Issues and Positions when speaking, but could for instance decide that something should be recorded (by the scribe) for later attention, or query something which might be answered by reference to the DR.

In this study, it is not clear whether part of the DR scribe's role was to interject when discussion veered off-course – thus bringing IBIS to the explicit attention of the group – or whether the scribe played a passive role leaving the other members to direct discussion. If, during a meeting, the IBIS exists primarily as an implicit, private record in individuals' heads, it is of course very difficult to demonstrate precisely what effect this has on the meeting. To demonstrate that simply being aware of a rhetorical model can improve group process would require analysis of the dynamics of design meetings in greater detail than that presented. Anecdotal reports and examples may be the most we can expect in this respect. It is relevant to note, therefore, corroborating descriptions by Cassidy [50] of his informal use of QOC to facilitate meetings, suggesting that awareness of argumentation schemes may indeed assist deliberation.

Perhaps the most widely reported result of this case study was the discovery of eleven design flaws during the conversion from itIBIS to gIBIS. This result constitutes the only evidence which has been costed in any way (the time savings gained being estimated at between three and six times greater than the time cost of converting from itIBIS to gIBIS, and then using the

[3] A purely textual 'indented-text IBIS' (itIBIS) was used, with the Issues, Positions and Arguments indented in that order, like hierarchical levels of headings in an outlining tool.

gIBIS tool to review the DR). In reflecting on this result, Conklin and Burgess Yakemovic suggest that reviewing an issue base provides a different view of a design, and thus highlights different problems from more conventional design reviews. They propose more generally that "comparing and reviewing DR might actually reduce the cost of the design process by making it more rigorous and error free" (p.375).

A criticism which has been levelled at this finding is that the effort of converting from itIBIS to gIBIS was atypical of normal design activities. Moreover, some have asked, might not any equivalent effort have detected the problems? Whilst it may be true that *any* extra effort might have detected the problems, the crucial question is surely, would that effort have been invested normally? Explicit argumentation-based DR encourages the designer to 'make space' to reflect, which they might not otherwise do. Conklin and Burgess Yakemovic hint at the process which they believed operates when a DR is scanned: "The process of browsing the DR and looking for a specific entry seems to serendipitously trigger detection of problems or inconsistencies in the data" (p.375). More detailed analysis of how this may work in practice would greatly inform the design of DR notations and tools.

Hypermedia case study. McKerlie and MacLean [51,52] describe the use of QOC in a small hypermedia design team in which they were members. Firstly, they report that "using and creating QOC records (even roughly) during design discussions could help bring out assumptions which might otherwise have remained implicit." This is an often reported finding with semiformal argumentation schemes, since they encourage more systematic reflection and deliberation than one might otherwise engage in. Unfortunately, in this case study, no details are provided which explain how often, or by what process this occurred.

Secondly, QOC was used to integrate two strands of work on a problem into a common design space representation. It is reported that QOC helped to highlight the strengths of each strand, and their interrelationships, facilitating comparison between two sets of ideas by providing a common framework. This is an interesting example of the benefits of using a semi-structured representation, although again, further detail would have been of interest, describing for instance the process by which common Questions and Criteria were agreed, or how the fitting of the ideas into a common QOC space highlighted trade-offs and interrelationships between the two solutions.

Smalltalk-80 case study. In a annother series of studies, Buckingham Shum trained a designer working in Smalltalk-80 to use QOC, who was then studied over three monthly sessions [53,54]. He worked in a 'mock-up office' which he constructed for the purposes of the study to provide all the resources which he would normally have used (e.g. architectural diagrams; design notebook). In one session, the process of recording and restructuring QOC's Options and Criteria seemed to assist the designer in clarifying and developing several issues which had up until then remained clouded in design notes and in his mind. The main focus of this session was on the gradual refinement of Options.

In the process of redrawing, a number of insights were gained, as he sought to develop a coherent QOC structure, as shown schematically in Figure 8:

- another level of Option decomposition was added, by refining *prestore data elsewhere* (Option 1.2.2 → 1.2.2.1 and 1.2.2.2)
- it was realised that the third Option to the first Question, was really another refinement of the second Option (i.e. Option 1.3 became 1.2.3). This Option was therefore moved down a level in the Option hierarchy.

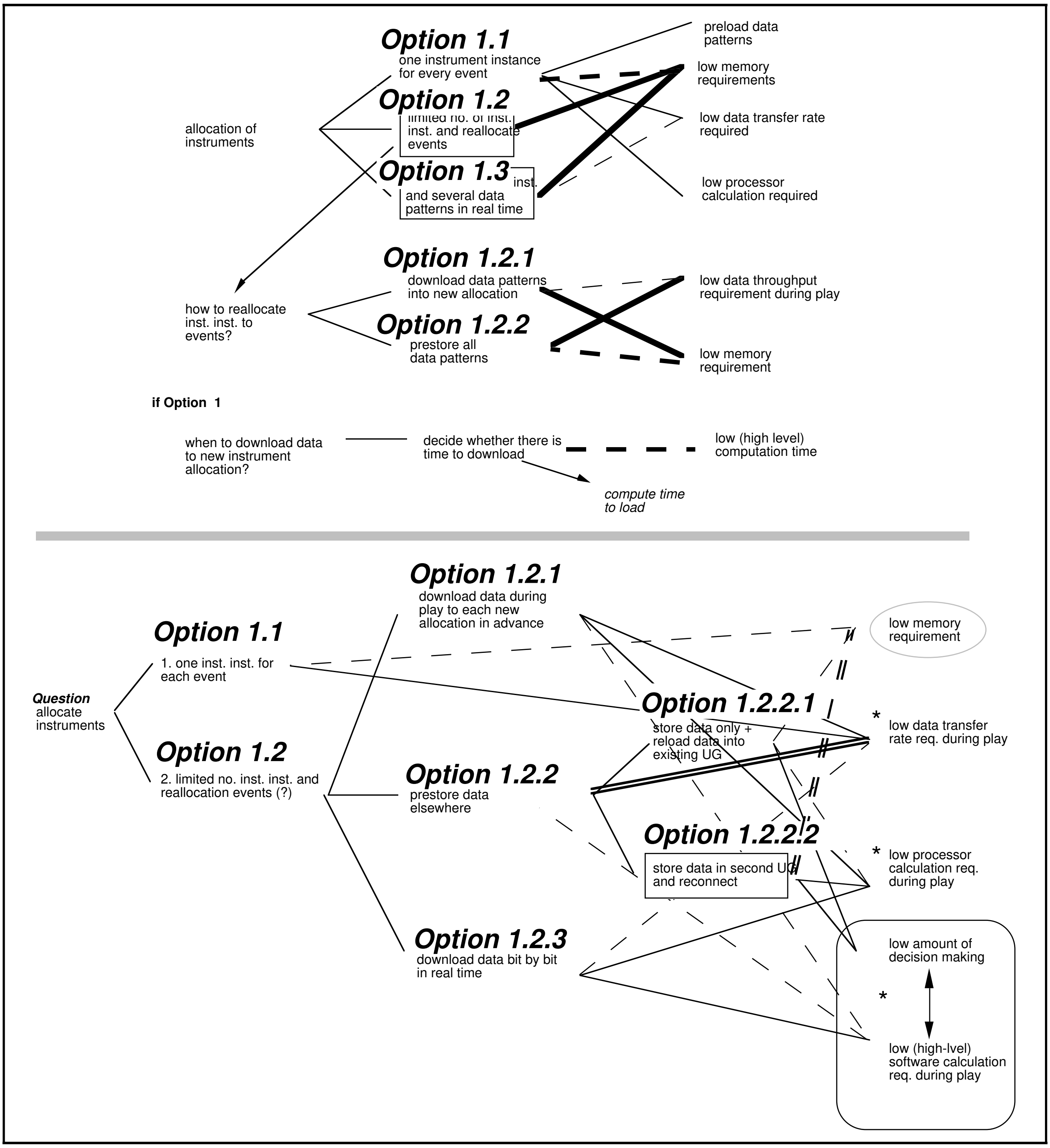

FIGURE 8: Overview of QOC analysis to show how it was restructured as the relationships between Options were better understood (Option numbers added to show the transition). The process of laying out competing Options as a hierarchy and then refining them (lower figure) prompted the designer to change Option 1.3 to Option 1.2.3, and to refine Option 1.2.2 one level further. As the designer was using pen and paper and renamed and restructured extensively, the original QOC was littered with changes which have been omitted for clarity. (Reproduced with permission from [48], Academic Press Ltd.)

Although it is possible that there was no causal relationship between using QOC and the insights gained (ideas do arise serendipitously 'out of the blue'), the video data and verbal transcript of the designer's use of the representations do appear to show that the requirement to articulate Options and Criteria forced him to clarify relationships and distinctions which had up until that point remained unaddressed. The designer declared that these were new insights, rather than old ones simply recorded as QOC.

Safety-argument tool case study. In this case study [55], QOC was used for about 9 months to document the evolution of design concepts and rationale, two years into a three year academic-industrial project. This was building a tool for constructing and verifying semiformal and formal arguments about the design of safety-critical systems. The goal of the QOC exercise was the production of a DR document which retrospectively structured the project's fragmented memory into a more accessible form, which could also support the incremental addition of ongoing decisions and rationale as the project progressed. QOC was used to organise what were termed *Sources* (design discussions from many contexts and in different media), through a process of *filtering* (reviewing relevance), *integrating* (pooling otherwise fragmented Sources), and *indexing* (providing links into the body of Sources for reference). All of the QOC work was carried out by a DR researcher (this author), although subsequently, QOC constructs came to be used by one designer as outline headings in project discussion documents.

The report gives examples of how Questions in particular proved to be a useful construct in the process of collating and rationalising the Source materials. As the disparate sources slowly revealed clusters of related discussion, Questions provided 'conceptual pigeonholes' into which related material could be posted; once sufficient Sources had been collated, the Question-based approach forced the DR researcher to tease out the key issues, and make the areas of common ground explicit. Sometimes it was possible to expand Questions into local design spaces in which Options could be evaluated against Criteria.

B. Access to argumentation about past decisions can be valuable

Workstation case study. In this case study, Conklin and Burgess Yakemovich reported that the IBIS record of previous discussions helped a meeting co-ordinator to identify seven interrelated Issues which needed to be resolved together. Summaries of these were then circulated to members with all the Positions and Arguments which had been discussed. This clear identification of the inter-relationships between issues, the decisions needing action and the relevant arguments pertaining to each enabled the different stakeholders to prepare fully for the meeting, and led subsequently to an elegant solution.

In this incident, it is as a concrete record of past discussions that IBIS seems to have been most valuable – both to the meeting planner in its raw IBIS form, and to the other members in the form of the agenda. Details are provided in the description of this incident about the context in which the Issues were identified: during conversion from indented-text IBIS to gIBIS, the meeting co-ordinator thought there were two Issues to resolve, but the itIBIS record showed him that there were five others which needed solving simultaneously. It would be of interest to know *how* the itIBIS highlighted this. For example, did the designer accidentally stumble across these five other Issues when browsing the IBIS, or were they explicitly linked to the first two Issues which the designer had initially regarded as relevant? Perhaps the designer was stimulated by the DR to think creatively which led him to the five other Issues which were distributed in the itIBIS record. With this level of analysis, it would be possible to more fully exploit the successes of this study, for instance, what representational properties should be preserved and given more support?

Hypermedia case study. McKerlie and MacLean describe how QOC was used to communicate to an absent designer what the team had discussed; they report that this "helped the originator of the ideas to understand how they had been interpreted by the rest of the team and therefore helped her contribute to the evolving design space." The report that QOC was useful to another party is not elaborated, although intuition would suggest that seeing a QOC will provide insight into someone's analysis of the problem (at least when the problem is familiar to both parties). More detailed analysis is required into the utility of argumentation-based DRs for communicating understandings and analyses, since there is at least one report [17] of argumentation schemes not assisting communication. A key factor is that both parties need to be familiar with the notation, since QOC and IBIS are undeniably novel ways to record discussions (in the workstation case study, Burgess Yakemovic and Conklin report that it was difficult to use IBIS to communicate with groups who were unfamiliar with it).

McKerlie and MacLean also report that "QOC records have been a valuable resource when it was necessary to revisit previous decisions either to alter them or to recollect the considerations which contributed to them." A quite detailed account is then presented, with accompanying QOC, of a decision reversal which led to other changes, whilst preserving the best points of the existing design. Specifically, articulating a new Question is shown to clarify the relationship between two existing Questions. It is concluded that, "QOC records' most important role for supporting the revisiting of decisions is to provide a broader context around the decisions of concern [...] which includes the relationships among Questions, the alternative Options considered, and the Criteria used to assess the Options."

This detailed example illustrates well the added leverage which the notation can provide in understanding the structure of a problem space. It would be of interest to know the frequency of such incidents, for instance, was this the only one all year, or one of a number? The case for using argumentation-based DR will be strengthened by demonstrating that such incidents are not so rare as to invalidate the effort.

Safety-argument tool case study. An earlier section described how QOC served as a useful framework for tracking and reorganising DR. However, a major goal was to see if a DR created by a relative outsider would prove useful to the design team whose ideas it represented. This is a variation on the more usual scenario of designers creating DR as a resource for other designers. However, at a meeting to evaluate the results of the DR work (approximately 50 pages indexing the design as QOC), designers' reactions to the DR document were generally negative, highlighting a number of obstacles to its acceptability. The main issues to arise were the *notational form* (benefits and clarity of QOC graphs and tables as opposed to less structured text), the *content* of the DR (the importance of capturing dominant themes and principles in the project), the *ease of DR navigation and retrieval* (such a large resource needs hypertext support), and *social/organizational issues* (coming late into the project, team members were highly pressured as deadlines approached, and it was difficult to get them to invest additional effort).

These studies represent initial investigations into what makes explicit design argumentation intelligible and useful as a resource. One of the most interesting issues yet to be studied is the extent to which a DR constructed by one community in a specific context (e.g. developers), proves either intelligible or useful to another community (e.g. maintenance; management; upgraders) with their own set of concerns and background.

C. Creating design argumentation can also impede ongoing design work

Let us now consider cases where using an argumentation scheme seemed to impede rather than facilitate work.

Use of PHI-IBIS. McCall [27,56] extended the IBIS notation by adding a quasi-hierarchical Issue-*serves*-Issue structure (Procedural Hierarchy of Issues–PHI). In principle, this should help focus deliberation on Issues which serve the stated aims of the design (i.e. the top Issue)—if an issue cannot be shown to serve part of the hierarchy, then it is likely to be irrelevant. McCall argued that, at least in principle, PHI improves IBIS as a stimulus to reflective reasoning.

Of interest, therefore, are the results of studies into PHI's use for architectural design (as described by Fisher *et al.* [9]). They report that using PHI on its own prolonged the reflective process and encouraged a lot of analysis of high level or peripheral issues, distracting designers from getting down to concrete detailed design issues. These studies were conducted using design students over several years, with analyses of PHI's use varying from assessment of student assignments, to detailed video protocol analysis of individual students. As such, these results can be considered as a reliable example of how an argumentation scheme can obstruct the design process.

Hypermedia case study. In the hypermedia case study, McKerlie and MacLean [51] identified an interesting phenomenon: as QOC began to be used, there was an early tendency towards "compulsive rationalising" – detailed QOC analysis of local spaces which were not of great interest, such as non-controversial or out of date issues. This is an instance of DR impeding design, insofar as it did not lead to insights, it distracted from more productive work, and was never utilised by the project. This report provides examples of situations in which deliberate QOC analysis proved of little use.

Smalltalk-80 case study. In this study, described earlier, the designer had found QOC very useful for exploring a range of alternatives. However, in two other sessions (reported in detail elsewhere [53]), the argumentative mode of reasoning required by QOC proved incompatible with the construction task at hand. The designer encountered severe difficulties in breaking a different problem into discrete Questions, Options and Criteria. Analysis of this session led to the conclusion that he was engaged in depth-first, evolutionary design, in effect refining a single, complex Option (a data structure), as opposed to the breadth-first, strategic reasoning required in an earlier session when QOC had been useful in clarifying the relationships between a set of competing alternatives (as shown in Figure 8).

D. Summary of empirical studies

A graphical, searchable record of design discussions organized by issues is a far richer resource than is normally available in the design process. The cost of course is that somebody must construct that record, and it must be worthwhile for them to do so, or it simply will not be done. The empirical studies surveyed in this article have been analyzed with a view to understanding when and why it may be worthwhile to use semiformal argumentation schemes to capture design rationale. It is only then that informed claims can be made when seeking to demonstrate to others the value of more structured deliberation.

The current picture might be summarised as follows. Ill-structured, poorly understood problems require an analytical, reflective mode of designing for which QOC, IBIS and similar argumentation schemes seem well suited as an aid to articulating the key issues, alternatives, and trade-offs. In contrast, for well-structured, more constrained problems, the key issues are already established, and the designer will often have a repertoire of solutions which can be either directly applied, or refined. The deliberative process tends to be more implicit, and often focused on the details of constructing the artifact in some form (requirements; sketches; code), rather than on making broader issues and trade-offs explicit. Articulating rationale in this mode of working is often extremely difficult, and a requirement to capture it explicitly very disruptive. Thus, the mode of design work in which one tries to use an argumentation-based notation plays a key role in determining whether it provokes useful structuring and reflection, or is intrusive and obstructs what really needs to be done. This account corroborates Rittel's initial observations about the unique nature of 'wicked' problems, which in turn led to the development of the IBIS notation, on which current argumentative DR approaches are based. Obvious as this may seem in retrospect, it clearly was not obvious to those who sought to use design argumentation in what turned out to be ineffective ways. Analysis of successes and failures in the use of design argumentation clarifies what the boundaries of useful application are.

IV. Lessons learnt, and the future agenda

In principal, computer-supported, collaboratively constructed design rationale offers many benefits, as listed at the start of this article. Not surprisingly, the challenge of turning making this reality has thrown into relief a range of sociotechnical issues which are now discussed.

There are at least five elements to consider in assessing the usability and utility of a DR capture and management approach: the *user* [U], the *notation* [N] used to express DR, the *technology* [T] used to represent the notation, the *domain* [D], which we shall define as *those classes of rationale which need to be expressed*, and *organizational factors* [O], which cover relevant aspects of the design context. The relationships between these elements define a space of issues which need to be considered, for instance:

Cognitive/group cost of encoding and retrieving DR [relation between U and (T+N)]:
- how natural is it to express ideas in the DR notation?
- does the effort to express structured DR facilitate individual/group design process?
- is the DR comprehensible to others (i.e. is it reusable?)

Computational leverage over the DR representation [relation between T and N]:
- how machine-interpretable is the DR? (e.g. to support DR checking, analysis and retrieval)

Expressiveness of the DR language [relation between (N+T) and D]:
- does the computer-supported DR allow designers to express the kinds of DR that they want to?

Organizational fit [relation between (N+T) and (U+O)]:
- how compatible with extant design methods and management practices is the approach?
- is there CAD/CASE tool compatibility, enabling DR functionality to be easily added and shared within the existing infrastructure?

The remainder of this section discusses some of the issues raised by considering the interactions between these different ‘stakeholders’ in the world of design rationale.

A. Negotiating the cost/benefit trade-off: notations and tools

An obstacle which any new representation must overcome is the overhead which it introduces. If more work is introduced over and above the tasks already tackled by users, a good case must be made to convince them to adopt the new approach. In addition, the more formal the notation (i.e. the more computer interpretable) in general, the higher the initial learning costs will be; as one moves from informal notes, to structured notes, to hypertext, to knowledge and device modelling, the more cognitively expensive the representation will be to learn in general.

The onus is clearly on those who support a particular representational approach to demonstrate that its strengths outweigh its weaknesses. Thus, sceptics are entitled to ask such questions as: are designers prepared to create informal DR, and can it be retrieved?; are semiformal approaches learnable and used?; are designers prepared to invest the effort in return for more powerful DR representations? As indicated in the survey of design studies, answers to these questions are now forthcoming, but more practical trials are required to demonstrate how this cost/benefit trade-off can be successfully negotiated in commercial design environments.

One common result from the studies reviewed was that often, the status of ideas or their inter-relationships is not instantly clear—the requirement to make these explicit as node and link types can be intrusive. The feeling of being prematurely forced to commit to a particular structure can be reduced by devising and computationally supporting more ‘relaxed’ notations, which can be gradually formalised as ideas become clearer. Such ‘rough DR’ acts as a bridge between early, fragmented ideas and coherent, rigorous argumentation.

An implication of these findings is that a flexible representational environment is required which can support the crucial process elements of the task of rationalising decisions. Since designers can rarely ‘pour’ well articulated, coherent arguments direct ‘from their heads’ into a semiformal notational structure, the process is invariably one of refinement, and when in a group context, negotiation. Within the hypertext and collaborative design research communities, it is therefore encouraging to see a recent emphasis on the importance of allowing designers to express themselves ambiguously, rather than precisely and explicitly. For instance, Marshall and Rogers [57] emphasise the importance of being able to work with unlinked, but meaningful clusters of nodes in the early stages of developing a representational schema in the Aquanet hypertext system [58]. The requirements for a knowledge structuring tool which have been specified for Aquanet could provide an environment which meets many of the likely requirements for a DR tool. The ability to work with spatial hypertext layouts, and mechanisms to support the transition from these to more formal structures, are currently being explored in the VIKI system [59,60], a successor to Aquanet.

In another project, Twidale *et al.* [61] describe the Designers’ Note-Pad (DNP) for supporting the earliest stages of recording and organising ideas. DNP has been used experimentally for early brainstorming in writing, concept mapping, requirements analysis and design rationale. Much effort has been devoted to minimising the intrusiveness of the tool, by providing flexible schemes for organising nodes and links, such as colour, shape, and spatial arrangement. The Inspiration brainstorming tool [62] is a commercially available example with similar functionality. Another promising effort in this direction is collaborative pen-based computing, such as the Tivoli electronic whiteboard [63]. Tivoli provides pen-based manipulation of groups of text and graphic objects in order to support common representational tasks in group work around a whiteboard.

Delaying commitment to explicit structure clearly does not come without cost: avoiding typing and linking in favour of supporting the gradual emergence of coherence from ill-structured material, trades off against the provision of powerful abstractions and the explicit articulation of relationships. These concerns have been succinctly expressed by Green [64], who identifies them as dimensions germane to any environment for structuring information (respectively: *premature commitment*, *role expressiveness*, *viscosity*, and *hidden-dependency*). A link between requirements such as these, and tools which might satisfy them was set out in [65], demonstrating how these dimensions could be operationalised with respect to DR. The trade-offs incurred as one moves from the informal to the semiformal are illustrated schematically in Figure 9.

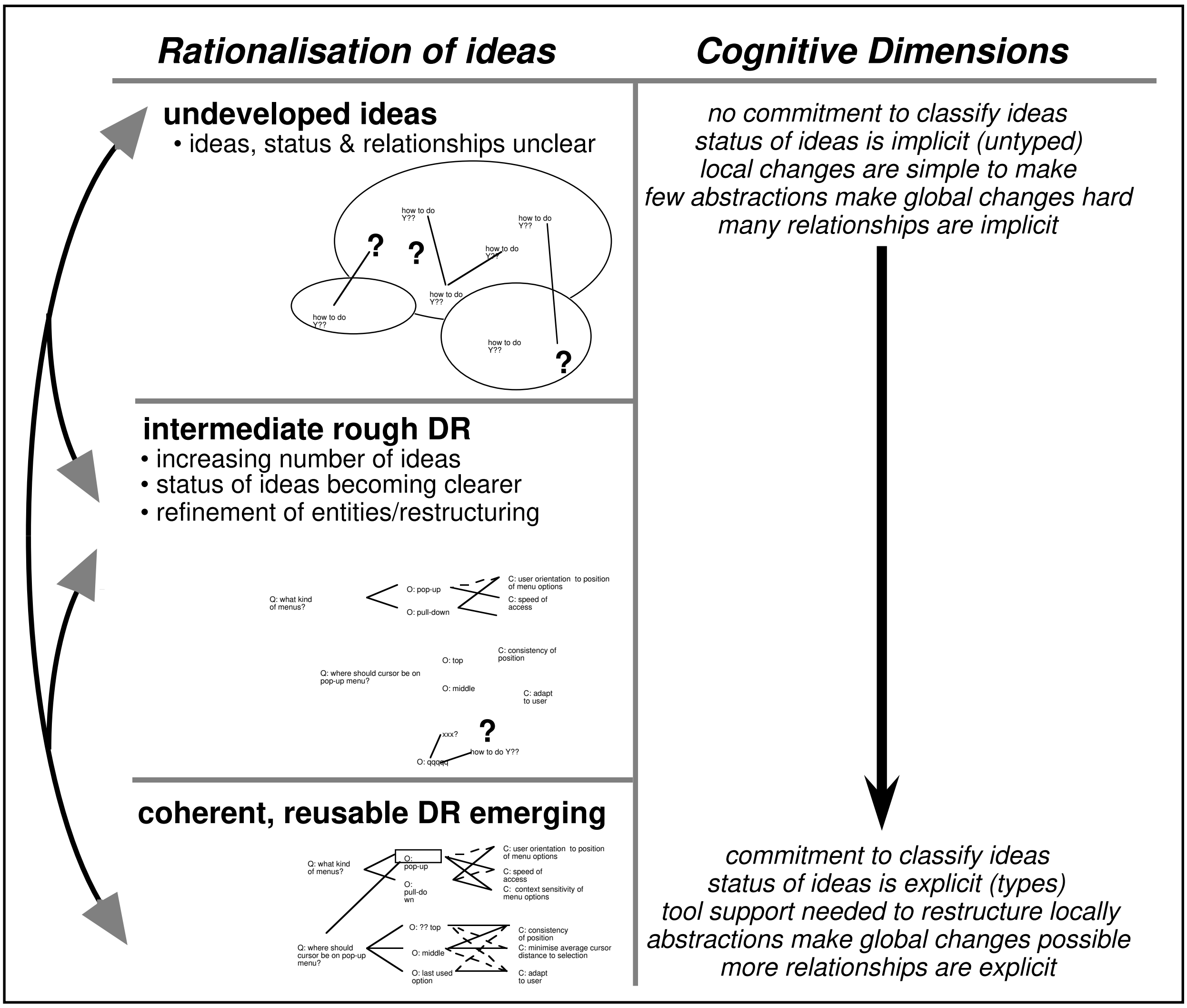

FIGURE 9: Trade-offs between 'cognitive dimensions' [65] at different stages in the process of rationalising argumentation. The curved arrows indicate that the process is non-linear: designers need to switch between representations opportunistically. (Reproduced with permission from [48], Academic Press Ltd.)

As representational tools become increasingly able to support the subtleties of the process outlined in Figure 9, the use of semiformal notations becomes a more tractable prospect.

Given the development of very low constraint representational tools such as DNP and VIKI, the next challenge is to support the designer in formalising the results of a brainstorming session into argumentative form (the transition shown in Figure 9). So for instance, how easy is it to select three unclassified DNP notes, declare them to be QOC Options, and link and position them neatly with another note which is declared to be a Question?; or to select a Criterion, and change it to a new Question, or convert two Criteria into one, with the associated link-management? These are examples of routine QOC restructuring tasks [53]. Without support for structural manipulations of this sort, the designer will find it easy to externalise initial ideas, but then face a tedious rewriting or copying task to convert them to more rigorous argumentation. With the addition of structural search facilities, automatic verification of certain sorts becomes possible. For example, one could check that no decisions have been made without considering negative assessments against Criteria (an indication of confirmation bias), or that there are no Questions which address more than one important issue. In summary, the challenge for tool developers is to make shifting between degrees of formality as smooth and transparent as possible.

Before concluding this section, a discussion of tool support for DR cannot ignore the developments of multimedia. Argumentation-based DR is without doubt dependent on technology, since the concept of editing and maintaining large network structures was unworkable before the emergence of hypertext and groupware to author, share and navigate rationale. However, the possibilities now exist to augment textual argumentation structures with audio and video-based rationale. Thus, it is becoming increasingly cheaper to link in to hypertext argument structures relevant design documents (e.g. CAD diagrams and rough sketches), and audio-visual extracts of users expressing requirements, or designers explaining decisions [52,66].

B. Who should use it, and how?

If DR is to be created, who is going to do it? The context in which a notation is used will determine the expertise of those present, and the time available to structure deliberation. These 'what, when, and how' factors cannot be ignored when designing a notation. Since design invariably involves meetings, we can ask what roles DR representations can play in such collaborative settings? From the literature, and the work which I and my colleagues have conducted with QOC in particular, it has become clear that a DR notation can shape design discussion – and hence, presumably, the design – to a greater or lesser extent depending on the way it is used. At one end of the continuum, the state of the DR plays a major role in driving discussion, for instance, the meeting agenda is to debate and resolve a set of IBIS Issues, and to record that process. In this case, the discussion will be largely conducted in terms of the notation's rhetorical moves, and the DR is owned by the whole group. The DR representation's influence can then be gradually weakened in the extent to which the recorded argumentation shapes subsequent deliberation, examples of which are shown in Figure 10.

Proactive role for DR

- Developing a coherent DR representation is a central, collaborative activity in the meeting; ideas are developed through informal DR, and then edited and restructured as ideas evolve, to create a succinct summary of the key arguments behind the main decisions.
- A DR-scribe records discussion, but other designers can see and use the emerging DR as a shared representation to monitor their progress and guide discussiony (a single author version of the one above).
- DR scribe privately records group discussion, which is then reviewed later by the team for erroneously recorded ideas, omissions, weaknesses in their reasoning, action items, and so forth; documentary role during meeting; provokes reflection on review.
- DR scribe privately records discussion, which is only referred to if information is later needed; no restructuring; purely documentary.

Reactive role for DR

FIGURE 10: Ways in which DR can be used in meetings, depending on the extent to which designers collaboratively construct the DR, and use it as a resource to focus discussion. (Adapted with permission from [48], Academic Press Ltd.)

Thinking about DR use in these terms has practical implications for two suggestions commonly made by designers and DR researchers as to how DR might be used in meetings. One suggestion has been is that a designated ‘scribe’ should record discussion as DR, whilst the other designers proceed with the ‘real’ design work, freed from the overhead of using DR; the second suggestion has been that decisions should be translated from notes into DR by someone after meetings. However, in both cases, DR plays no part in the synergistic process which is possible when a representation is jointly constructed and debated. The vision of a ‘living,’ up-to-date rationale requires shared ownership of the DR, in which domain experts and decision-makers record their own reasoning rather than relying on a scribe to capture it second-hand.

C. The importance of training

Technological intervention requires preparation ‘on the ground,’ in the form of communication with, and training of those it will affect. The importance of training in DR is now beginning to be recognised in DR research [67], and is for instance an integral part of the QuestMap IBIS-based groupware system [35].

Whilst most DR approaches are relatively easy to grasp in terms of simply understanding why a DR might be useful, and how rationale might be expressed using a particular notation, from our experiences in communicating our work to design teams, it has proven quite a different matter to actually use it in design. There is no substitute for hands-on practice. The development of a tutorial for the QOC Design Space Analysis approach is described in detail elsewhere [67]. The important point in the context of this paper is that it is based not only on in-house experience with QOC on the part of the developers of the approach, but on studies of its use. The challenge of communicating the approach to practitioners encouraged us to consider the practical implications of our research analyses, which were then codified in what we hoped would be appropriate forms. As the demands of conducting productive QOC analyses became clearer from empirical studies, the tutorial materials used to train designers in those studies were revised to sensitise the next set of designers to important features of QOC authoring. Consequently much greater emphasis was placed on practical hands-on tips.

To summarise, it is anticipated that the serious adoption of DR in the development process will have significant consequences at the individual, group, and organizational levels. Effective training facilitates an individual, project, or organization to change relevant structures more quickly, and highlights pitfalls to avoid at each level. Successful transfer from the research lab to commercial practice invariably depends on early pilot successes within client organizations, and it is here that good training has a critical role to play in setting appropriate expectations and equipping individuals with the awareness and skills they need to re-engineer established practices which may militate against new modes of working.

D. Towards model-based DR for user interfaces

In some engineering design domains, it is possible for DR tools to do much more than they can in domains like user interface design, because the domain is better understood. In electrical and mechanical engineering for instance, we find efforts (depending on the approach) to model design requirements, the artifact's operational structure, its interaction with the environment, and the design process [68-70]. As Gruber and Russell have argued [71], representations of the artifact are necessary in addition to representations of the argumentation about it, if DR systems are to provide answers to many classes of query about the artifact, and in particular, queries which have not been anticipated by the authors of the DR. In principle, with this additional knowledge, DR can be generated to:

- *explain the physical structure of artifacts:* what are the constituent components?
- *explain their operation:* how does this interact with other components/environment?
- *explain their function:* what is the purpose of this?
- *support evaluation/verification:* does the design satisfy the initial goals/requirements?
- *support modification:* what effect will the following changes have?
- *support dependency management:* is this decision consistent with the rest of our decisions?
- *support precedence management:* what decisions required similar design analyses to that required in the current problem? (i.e. how did we do this last time?)

As modelling techniques develop to formally describe users, systems, and their interaction [e.g. 72-76], descriptions of interactional properties may one day be sufficiently formalisable to provide substantial computational support for such queries.

E. Building DR on existing organizational practices

How is the DR message to penetrate current design practice? At a workshop [77] held at the ACM CHI'94 conference, researchers and industry practitioners agreed that the most fruitful strategy is likely to be through 'piggy-backing' on existing design practices and technology, that is, to identify areas in current practice where activities which involve rationale in various forms are already common-place, or where technology could be augmented in relatively simple ways in order to make DR capture relatively painless, with immediate benefit to the designer.

F. Politics and organizational culture

Like all formalisms, a design rationale notation has political dimensions [78]. Argumentation-based DR requires the adoption of a particularly open style of design and accountability which might be considered by some to be idealistic. DR faces the harsh realities of development environments where there is distrust between employees and managers, and where notions of meeting facilitation or making one's reasoning transparent for others to inspect will get short shrift. In such organizations, employees may well refuse to document who was responsible for a particular decision and why, for fear of recriminations in the event of a design flaw, or more senior designers will perceive DR schemes as undercutting their power, since their design arguments will be represented and treated on an equal footing with other team members' views.

Clearly, nobody is suggesting that these problems can be solved overnight. In the meantime, DR must get a little 'harder-nosed,' and develop its own case for those who think in terms of resource savings and productivity gains. What is the cost of not having readily accessible DR when organizations are constantly reshaping, and drawing increasingly on external consultants who often leave little trace of their rationale once a project has ended? How much time and energy is wasted in reconstructing old solutions, not being able to track down why past decisions were made, or endlessly revisiting the same old issues?

There is no doubt that an organizational culture shift towards greater corporate self-awareness is needed, coupled with the right training and tools, before DR becomes a viable prospect. Signs that DR tools are coming of age may be seen in the reports now beginning to emerge from industrial groups who have been successful in demonstrating to their employers the value of capturing and sharing DR [49,79], and the commercial release of an IBIS-based groupware system (see Figure 4) for personal computers which is finding use in several large organizations concerned with improving strategic decision-making. Intriguingly, it is possible that the organizational culture shift required is already underway at the management level, as we witness calls to develop and maintain 'organizational memory' [80,81], in conjunction with emerging models of 'intelligent, learning organizations.'

A design rationale is a design-specific organizational memory—a shared resource which allows distributed members of a design organization to share their expertise, in particular, important, context-specific 'nuggets' of knowledge which may otherwise be lost, or remain locked in particular individuals' memories. The same exciting prospects and practical problems apply to both design rationale and organizational memory systems. What is needed now are forward looking, process-aware design organizations who are prepared to pilot design rationale systems and track their effectiveness. This review has described the strengths and weaknesses of a particular approach to capturing design rationale which seeks to build on the conversations and debates which already underpin the design process, by making them more rigorous, sharable and reusable.

Acknowledgements

My thanks to Nick Hammond, Allan MacLean, Jean McKendree, Will Reader and the reviewers of earlier versions, whose detailed comments helped to strengthen this article. This analysis was conducted during projects funded by Rank Xerox EuroPARC, UK Science and Engineering Research Council (CASE Award 88504176), and ESPRIT Basic Research Actions 3066 and 7040 (AMODEUS Projects 1 & 2).

V. References

1. *A Tool Shell for Tracking Design Decisions.* Arango, G., Bruneau, L., Cloarec, J.-F. and Feroldi, A., *IEEE Software*, March: 75-83, 1991.
2. *Design Rationale and Maintainability.* Conklin, J., In *Proceedings 22nd Hawaii International Conference on System Science*, IEEE Computer Society Press, pp. 555-561, 1989.
3. *Design Rationale for Software Engineering: A Survey.* Jarczyk, A., Loffler, P. and Shipman, F., In *Proceedings of 25th Annual Hawaii International Conference on System Sciences (8-10 January, 1992)*, 1992.
4. *Extending the Potts and Bruns Model for Recording Design Rationale.* Lee, J., In *Proceedings 13th International Conference on Software Engineering*, Austin, Texas, 1991.
5. *Panel on Design Rationale.* Greenspan, S., In *RE'93: IEEE International Symposium on Requirements Engineering*, San Diego, CA, January, 1993, IEEE Society Press: New York, 1993.
6. *Presenting Ethnography in the Requirements Process.* Hughes, J., O'Brien, J., Rodden, T., Rouncefield, M. and Sommerville, I., In *Proc. RE'95: Second IEEE International Symposium on Requirements Engineering*, March, 1995 (York, UK), IEEE Computer Soc. Press, Los Alamitos, CA, pp. 27-34, 1993.
7. *Deliberated Evolution: Stalking The View Matcher in Design Space.* Carroll, J.M. and Rosson, M.B., *Human-Computer Interaction*, 6, 3&4: 281-318, 1991.
8. *gIBIS: A Hypertext Tool for Exploratory Policy Discussion.* Conklin, J. and Begeman, M.L., *ACM Transactions on Office Information Systems*, 6, 4: 303-331, 1988.
9. *Making Argumentation Serve Design.* Fischer, G., Lemke, A.C., McCall, R. and Morch, A.I., *Human-Computer Interaction*, 6, 3&4: 393-419, 1991.
10. *SIBYL: A Tool for Managing Group Design Rationale.* Lee, J., In *Computer Supported Cooperative Work*, Los Angeles, CA, ACM Press: New York, pp. 79-92, 1990.
11. *Design Rationale: The Argument Behind the Artifact.* MacLean, A., Young, R.M. and Moran, T., In *Proceedings of CHI'89: Human Factors in Computing Systems*, ACM: New York, pp. 247-252, 1989.
12. *Design Rationale: Concepts, Techniques, and Use.* Moran, T.P. and Carroll, J.M. (Eds.), Lawrence Erlbaum Associates: Hillsdale, NJ, in press.
13. *Outline of a Paradigm Change in Software Engineering.* Floyd, C., In *Computers and Democracy – A Scandinavian Challenge,* (Bjerknes, G., Ehn, P. and Kyng, M., ed.), Avebury: Aldershot: UK, 1987.
14. *Towards a Software Maintenance Support Environment.* Wild, C. and Maly, K., Department of Computer Science, Old Dominion University, Norfolk, Virginia, *Technical Report TR-88-04*, 1988.
15. *A Conceptual Framework for the Augmentation of Man's Intellect.* Englebart, D.C., In *Vistas in Information Handling*, Spartan Books: Washington, DC: London, pp. 1-29, 1963.
16. *The Uses of Argument.* Toulmin, S., Cambridge University Press: Cambridge, 1958.
17. *Are Argumentation Representation Schemes Useful?* Hair, D.C. and Lewis, C., Department of Computer Science, University of Colorado At Boulder, Boulder, Colorado 80309-1592, *Technical Report CU-CS-475-90*, 1990.
18. *Pushing Toulmin Too Far: Learning from an Argument Representation Scheme.* Newman, S. and Marshall, C., Xerox Palo Alto Research Center, *Technical Report*, 1991.
19. *Process Versus Product: A Perspective on Tools for Communal and Informal Electronic Learning.* Brown, J.S., *Report from The Learning Lab: Education in the Electronic Age, by the Educational Broadcasting Corporation; Reprinted in Journal of Educational Computing Research*, 1, 2: 179-202 (1985), 1983.
20. *From Cognitive Ergonomics to Social Ergonomics and Beyond.* Brown, J.S., In *User Centered System Design,* (Norman, D.A. and Draper, S.W., ed.), Lawrence Erlbaum Associates: Hillsdale, New Jersey, pp. 457-486, 1986.
21. *Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving in Meetings.* Stefik, M., Foster, G., Bobrow, D.G., Kahn, K., Lanning, S. and Suchman, L., *Communications of the ACM*, 30, 1: 32-47, 1987.
22. *The Colab Project Final Report.* Bobrow, D.G., Stefik, M., Foster, G., Halasz, F., Lanning, S. and Tatar, D., Xerox Palo Alto Research Center, *Technical Report SSL-90-45*, 1990.
23. *Issues as Elements of Information Systems.* Kunz, W. and Rittel, H.W.J., Institut für Gundlagen Der Planung I.A, Universität Stuttgart, Kleperstraße 11, 7000 Stuttgart 1, Germany, *Technical Report S-78-2*, 1970.
24. *Dilemmas in a General Theory of Planning.* Rittel, H.W.J. and Webber, M.M., *Policy Sciences*, 4: 155-169, 1973.

25. *gIBIS: A Tool for All Reasons.* Conklin, J. and Begeman, M.L., *Journal of the American Society for Information Science*, May: 200-213, 1989.
26. *A Process-Oriented Approach to Design Rationale.* Conklin, J. and Burgess Yakemovic, K.C., *Human-Computer Interaction*, 6, 3&4: 357-391, 1991.
27. *PHI: A Conceptual Foundation for Design Hypermedia.* McCall, R.J., *Design Studies*, 12, 1: 30-41, 1991.
28. *The Structure of Ill-Structured Problems.* Simon, H.A., *Artificial Intelligence*, 4, 181-201, 1973.
29. *Motivating the Notion of Generic Design Within Information-Processing Theory: The Design Problem Space.* Goel, V. and Pirolli, P., *AI Magazine*, , Spring: 18-36, 1989.
30. *The Structure of Design Problem Spaces.* Goel, V. and Pirolli, P., *Cognitive Science*, 16, 3: 395-429, 1991.
31. *Second Generation Design Methods.* Rittel, H.W.J., In *Developments in Design Methodology, (1984),* (Cross, N., ed.), J. Wiley & Sons: Chichester, pp. 317-327, 1972.
32. *Computer-Aided Reasoned Discourse Or, How to Argue with a Computer.* Smolensky, P., Fox, B., King, R. and Lewis, C., In *Cognitive Science and its Application to Human-Computer Interaction,* (Guindon, R., ed.), Ablex: Norwood, New Jersey, pp. 109-162, 1988.
33. *Special Issue on Design Rationale.* Human-Computer Interaction journal. Carroll, J.M. and Moran, T.P. (Eds.), LEA: New Jersey, 1991.
34. *Questions, Options, and Criteria: Elements of Design Space Analysis.* MacLean, A., Young, R.M., Bellotti, V. and Moran, T., *Human-Computer Interaction*, 6, 3 & 4: 201-250, 1991.
35. *QuestMap.* (Hypertext groupware for argumentation/organizational memory [Microsoft Windows™]). Corporate Memory Systems, Inc., 11824 Jollyville Road, Austin, TX 78759, USA, 1994.
36. *What's in Design Rationale?* Lee, J. and Lai, K., *Human-Computer Interaction*, 6, 3&4: 251-280, 1991.
37. *SIBYL: A Qualitative Decision Management System.* Lee, J., In *Artificial intelligence at MIT: Expanding Frontiers,* (Winston, P. and Shellard, S., ed.), MIT Press: Cambridge, Massachusetts, pp. 105-133, 1990.
38. *Recording the Reasons for Design Decisions.* Potts, C. and Bruns, G., In *Proceedings of 10th International Conference on Software Engineering*, pp. 418-427, 1988.
39. *An Active Hypertext Model for System Requirements.* Potts, C. and Takahashi, K., In *Proc. IEEE Workshop on Software Specification and Design*, IEEE Computer Soc. Press, Los Alamitos, CA, 1993.
40. *Supporting Communication Between Designers with Artifact-Centered Evolving Information Spaces.* Reeves, B. and Shipman, F., In *CSCW'92: Computer-Supported Cooperative Work*, Oct 31-Nov 4, 1992, Toronto, Canada, ACM Press: New York, pp. 394-401, 1992.
41. *Supporting Knowledge-Base Evolution with Incremental Formalization.* Shipman, F. and McCall, R., In *Proceedings of ACM CHI'94 Conference on Human Factors in Computing Systems*, Boston, Mass., Addison Wesley/ACM: New York, pp. 285-291, 1994.
42. *An Interactive Tool for the Construction, Modification and Browsing of Design Rationales of Smalltalk Programs.* Ryan, M. *Masters Thesis* Queen Mary and Westfield College, University of London,, 1992.
43. *Integrating Widget Design Knowledge with User Interface Toolkits.* Carey, T.T. and Crentsil, J., In *Proceedings IEEE CASE Workshop*, Montreal, July, 1992, Department of Computing and Information Science, University of Guelph, Canada, 1992.
44. *Supporting Software Design: Integrating Design Methods and Design Rationale.* Potts, C., To appear in: *Design Rationale: Concepts, Techniques and Use,* (Moran, T.P. and Carroll, J.M., ed.), Lawrence Erlbaum Associates: , in press.
45. *Inquiry-Based Requirements Analysis.* Potts, C., Takahashi, K. and Anton, A., *IEEE Software*, March'94: 21-32, 1994.
46. *PHIDIAS: A PHI-Based Design Environment Integrating CAD Graphics into Dynamic Hypertext.* McCall, R.J., Bennet, P., D'Oronzio, P., Ostwald, J., Shipman, F. and Wallace, N., In *Hypertext: Concepts, Systems and Applications,* (Rizk, A., Streitz, N. and André, J., ed.), Cambridge University Press: Cambridge, 1990.
47. *The Reflective Practitioner: How Professionals Think in Action.* Schön, D.A., Basic Books: New York, 1983.
48. *Argumentation-Based Design Rationale: What Use at What Cost?* Buckingham Shum, S. and Hammond, N., *International Journal of Human-Computer Studies*, 40, 4: 603-652, 1994.
49. *Report on a Development Project Use of an Issue-Based Information System.* Burgess Yakemovic, K.C. and Conklin, J., In *Proceedings of CSCW'90: Computer Supported Cooperative Work*, ACM: New York, pp. 105-118, 1990.
50. *Rationale in Practice: Templates for Capturing and Applying Design Experience.* Casaday, G., To appear in: *Design Rationale: Concepts, Techniques, and Use,* (Moran, T.P. and Carroll, J.M., ed.), Lawrence Erlbaum Associates: Hillsdale, NJ, in press.

51. *Reasoning with Design Rationale: Practical Experience with Design Space Analysis.* McKerlie, D. and MacLean, A., *Design Studies*, 15, 2: 214-226, 1994.
52. *QOC in Action: Using Design Rationale to Support Design.* McKerlie, D. and MacLean, A., In *Video Program, InterCHI'93: Human Factors in Computing Systems*, Amsterdam, Netherlands, ACM SIGGRAPH Video Review Series, 1993.
53. *Analyzing the Usability of a Design Rationale Notation.* Buckingham Shum, S., To appear in: *Design Rationale: Concepts, Techniques, and Use,* (Moran, T.P. and Carroll, J.M., ed.), Lawrence Erlbaum Associates: Hillsdale, NJ, in press.
54. *Learning to Use Argumentation-Based Design Rationale.* Buckingham Shum, S., MacLean, A., Bellotti, V.M.E. and Hammond, N., Amodeus-2 Project, *Working Paper TA/WP19*, 1993.
55. *Summarising the Evolution of Design Concepts Within a Design Rationale Framework.* Shum, S., MacLean, A., Forder, J. and Hammond, N.V., In *Adjunct Proceedings of InterCHI'93 (Short Papers)*, Amsterdam, pp. 43-44, 1993.
56. *Issue-Serve Systems: A Descriptive Theory of Design.* McCall, R.J., *Design Methods and Theories*, 20, 8: 443-458, 1986.
57. *Two Years before the Mist: Experiences with Aquanet.* Marshall, C.C. and Rogers, R.A., In *Proceedings of the Fourth ACM Conference on Hypertext*, pp. 53-62, 1992.
58. *Aquanet: A Hypertext Tool to Hold Your Knowledge in Place.* Marshall, C.C., Halasz, F.G., Rogers, R.A. and Janssen, W.C., In *Proceedings of Hypertext'91*, ACM: New York, 1991.
59. *Searching for the Missing Link: Discovering Implicit Structure in Spatial Hypertext.* Marshall, C.C. and Shipman, F.M., In *Proceedings of Hypertext'93: 5th ACM Conference on Hypertext*, Seattle, WA (Nov. 14-18, 1993), ACM: New York, 1993.
60. *VIKI: Spatial Hypertext Supporting Emergent Structure.* Marshall, C.C., Shipman, F.M. and Coombs, J.H., In *ECHT'94: Proceedings of the 6th ACM European Conference on Hypermedia Technology*, 18-23 Sept., Edinburgh, Scotland, 1994.
61. *The Designers' Notepad: Supporting and Understanding Cooperative Design.* Twidale, M., Rodden, T. and Sommerville, I., In *3rd European Conference on Computer-Supported Cooperative Work*, Milan, Italy (13-17 Sept.), Kluwer Academic Publishers, 1993.
62. *Inspiration.* (Brainstorming tool for Apple Macintosh). Inspiration Software Inc., 2920 S.W. Dolph Court, Suite 3, Portland, Oregon 97219, USA, 1992.
63. *Tivoli: An Electronic Whiteboard for Informal Workgroup Meetings.* Pedersen, E.R., McCall, K., Moran, T.P. and Halasz, F.G., In *Proceedings of InterCHI'93: Human Factors in Computing Systems*, Addison Wesley: , pp. 391-398, 1993.
64. *Cognitive Dimensions of Notations.* Green, T.R.G., In *People and Computers V,* (Sutcliffe, A. and Macaulay, L., ed.), Cambridge University Press: Cambridge, pp. 443-460, 1989.
65. *Cognitive Dimensions of Design Rationale.* Shum, S., In *People and Computers VI: Proceedings of HCI'91,* (Diaper, D. and Hammond, N.V., ed.), Cambridge University Press: Cambridge, pp. 331-344, 1991.
66. *Raison d'Etre: Capturing Design History and Rationale in Multimedia Narratives.* Carroll, J.M., Alpert, S.R., Karat, J., Deusen, M.S.V. and Rosson, M.B., In *Proceedings of ACM CHI'94 Conference on Human Factors in Computing Systems*, pp. 213, 1994.
67. *Developing the Design Space with Design Space Analysis.* MacLean, A., Bellotti, V. and Shum, S., In *Computers, Communication and Usability: Design Issues, Research and Methods for Integrated Services,* (Byerley, P.F., Barnard, P.J. and May, J., ed.), Elsevier: North Holland Series in Telecommunication.: Amsterdam, pp. 197-219, 1993.
68. *Recovering Rationale for Design Changes: A Knowledge-Based Approach.* Baudin, C., Sivard, C. and Zweben, M., In *IEEE Proceedings for Man, Machines and Cybernetics*, Los Angeles, CA, IEEE, 1990.
69. *Acquiring Design Knowledge Through Design Decision Justification.* Garcia, A.C.B. and Howard, H.C., *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 6, 1: 59-71, 1992.
70. *Capturing Design Rationale in Concurrent Engineering Teams.* Klein, M., *IEEE Computer*, January: 39-47, 1993.
71. *Generative Rationale: Beyond the Recall and Replay Paradigm.* Gruber, T.R. and Russell, D.M., To appear in: *Design Rationale: Concepts, Techniques, and Use,* (Moran, T.P. and Carroll, J.M., ed.), Lawrence Erlbaum Associates: Hillsdale, NJ, in press.
72. *Integrating Cognitive and System Models in Human-Computer Interaction.* Barnard, P. and Harrison, M., In *People and Computers V: Proceedings of HCI'89,* (Sutcliffe, A. and Macaulay, L., ed.), Cambridge University Press: Cambridge, pp. 87-103, 1989.

73. *Abstract Interaction Objects.* Duke, D. and Harrison, M., *Computer Graphics Forum*, 12: 25-36, 1993.

74. *Integrating User Requirements and System Specification.* Blandford, A., Harrison, M.D. and Barnard, P.J., In *Computers, Communication and Usability: Design Issues, Research and Methods for Integrated Services,* (Byerley, P.F., Barnard, P.J. and May, J., ed.), Elsevier: Amsterdam, pp. 165-196, 1993.

75. *Using Structural Descriptions of Interfaces to Automate the Modelling of User Cognition.* May, J., Barnard, P.J. and Blandford, A., *User Modelling and Adaptive User Interfaces*, 3: 27-64, 1993.

76. *Multidisciplinary Modelling In HCI Design... In Theory and In Practice.* Bellotti, V., Buckingham Shum, S., MacLean, A. and Hammond, N., In *Proc. ACM CHI'95: Human Factors in Computing Systems*, Denver, Colorado (May 7-11, 1995), ACM Press: New York, 1995, in press.

77. *Infiltrating Design Practice with Design Rationale: Report on the CHI'94 Workshop on Design Rationale.* MacLean, A., Carroll, J.M. and Sutcliffe, A., *To appear in: ACM SIGCHI Bulletin.*

78. *The Politics of Formalism.* Bowers, J., In *Contexts of Computer-Mediated Communication,* (Lea, M., ed.), Harvester Wheatsheaf: , pp. 232-261, 1991.

79. *From "Folklore" to "Living Design Memory".* Terveen, L.G., Selfridge, P.G. and Long, M.D., In *Proceedings of ACM/IFIP INTERCHI'93: Conference on Human Factors in Computing Systems*, pp. 15-22, 1993.

80. *Organizational Memory.* Walsh, J.P. and Ungson, G.R., *The Academy of Management Review*, 16, 1: 57-91, 1991.

81. *Definitional and Contextual Issues in Organizational and Group Memories.* Ackerman, M.S., In *Proceedings of HICSS'94: 27th Hawaii International Conference on System Sciences*, Hawaii, January, 1994.

VI. Bibliography for further reading

The following sources have been organised to assist further reading on different approaches to representing and maintaining design rationale.

Setting the scene:

Design Rationale for Software Engineering: A Survey: Jarczyk, A., Loffler, P. and Shipman, F., In *Proceedings of 25th Annual Hawaii International Conference on System Sciences (8-10 January, 1992)*, 1992

Introduction to this Special Issue on Design Rationale: Carroll, J.M. and Moran, T.P., *Human-Computer Interaction*, 6, 3 & 4: 197-200 (1991).

Outline of a Paradigm Change in Software Engineering: Floyd, C., In *Computers and Democracy – A Scandinavian Challenge,* (Bjerknes, G., Ehn, P. and Kyng, M., ed.), Avebury: Aldershot: UK, 1987.

Second Generation Design Methods: Rittel, H.W.J., In *Developments in Design Methodology, (1984),* (Cross, N., ed.), J. Wiley & Sons: Chichester, 1972, pp. 317-327.

What's in Design Rationale?: Lee, J. and Lai, K., *Human-Computer Interaction*, 6, 3&4: 251-280 (1991).

Compiled DR work:

Special Issue on Design Rationale: Carroll, J.M. and Moran, T.P. (Eds.), *Human-Computer Interaction*, 6, 3 & 4 (1991).

Design Rationale: Concepts, Techniques, and Use: Moran, T.P. and Carroll, J.M. (Eds.), Lawrence Erlbaum Associates: Hillsdale, NJ, 1995, in press.

Methodological perspectives on DR:

A Process-Oriented Approach to Design Rationale: Conklin, J. and Burgess Yakemovic, K.C., *Human-Computer Interaction*, 6, 3&4: 357-391 (1991).

Generative Rationale: Beyond the Recall and Replay Paradigm: Gruber, T.R. and Russell, D.M., To appear in: *Design Rationale: Concepts, Techniques, and Use,* (Moran, T.P. and Carroll, J.M., ed.), Lawrence Erlbaum Associates: Hillsdale, NJ, 1995, in press.

Information Systems Design as Argumentation: An Investigation into Design Rationale as a Conceptualization of Design: Goldkühl, G., In *14th IRIS Conference (Information System Research In Scandinavia)*, 1991

Making Argumentation Serve Design: Fischer, G., Lemke, A.C., McCall, R. and Morch, A.I., *Human-Computer Interaction*, 6, 3&4: 393-419 (1991).

Questions, Options, and Criteria: Elements of Design Space Analysis: MacLean, A., Young, R.M., Bellotti, V. and Moran, T., *Human-Computer Interaction*, 6, 3 & 4: 201-250 (1991).

Supporting Software Design: Integrating Design Methods and Design Rationale: Potts, C., To appear in: *Design Rationale: Concepts, Techniques and Use,* (Moran, T.P. and Carroll, J.M., ed.), Lawrence Erlbaum Associates , 1995, in press

The Task-Artifact Cycle: Carroll, J.M., Kellogg, W.A. and Rosson, M., In *Designing Interaction: Psychology At the Human-Computer Interface,* (Carroll, J.M., ed.), Cambridge University Press: New York, 1991.

Using Domain Concepts to Index Engineering Design Information: Baudin, C., Gevins, J., Baya, V. and Mabogunje, A., In *Proceedings of CogSci-92: 14th Annual Conference of the Cognitive Science Society*, 1992, Lawrence Erlbaum Associates: Hillsdale, NJ

Argumentation tool support:

Aquanet: A Hypertext Tool to Hold Your Knowledge in Place: Marshall, C.C., Halasz, F.G., Rogers, R.A. and Janssen, W.C., In *Proceedings of Hypertext'91*, 1991, ACM: New York

Author's Argumentation Assistant (AAA): A Hypertext-Based Authoring Tool for Argumentative Texts: Schuler, W. and Smith, J., In *Hypertext: Concepts, Systems and Applications,* (Rizk, A., Streitz, N. and André, J., ed.), Cambridge University Press: Cambridge, 1990.

A Tool Shell for Tracking Design Decisions: Arango, G., Bruneau, L., Cloarec, J.-F. and Feroldi, A., *IEEE Software*, March: 75-83 (1991).

Design Environments for Constructive and Argumentative Design: Fischer, G., Morch, A.I. and McCall, R., In *Proceedings of CHI'89: Human Factors in Computing Systems*, 1989, ACM: New York, pp. 269-275.

gIBIS: A Hypertext Tool for Exploratory Policy Discussion: Conklin, J. and Begeman, M.L., *ACM Transactions on Office Information Systems*, 6, 4: 303-331 (1988).

Integrating Widget Design Knowledge with User Interface Toolkits: Carey, T.T. and Crentsil, J., In *Proceedings IEEE CASE Workshop*, 1992, Department of Computing and Information Science, University of Guelph, Canada

QuestMap. Hypertext groupware for organizational memory [Microsoft Windows™]). Corporate Memory Systems, Inc., 11824 Jollyville Road, Austin, TX 78759, USA, 1994.

Raison d'Etre: Capturing Design History and Rationale in Multimedia Narratives: Carroll, J.M., Alpert, S.R., Karat, J., Deusen, M.S.V. and Rosson, M.B., In *Proceedings of ACM CHI'94 Conference on Human Factors in Computing Systems*, 1994, pp. 213.

rIBIS: A Real-Time Group Hypertext System: Rein, G.L. and Ellis, C.A., *International Journal of Man-Machine Studies*, 24, 3: 349-367 (1991).

SIBYL: A Tool for Managing Group Design Rationale: Lee, J., In *Computer Supported Cooperative Work*, 1990, ACM Press: New York, pp. 79-92.

Supporting Communication Between Designers with Artifact-Centered Evolving Information Spaces: Reeves, B. and Shipman, F., In *CSCW'92: Computer-Supported Cooperative Work*, 1992, ACM Press: New York, pp. 394-401.

Supporting Situated Interpretation: Stahl, G., In *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society*, 1993, Lawrence Erlbaum Associates: New Jersey, pp. 965-970.

The Designers' Notepad: Supporting and Understanding Cooperative Design: Twidale, M., Rodden, T. and Sommerville, I., In *3rd European Conference on Computer-Supported Cooperative Work*, 1993, Kluwer Academic Publishers

Case studies of DR use:

Deliberated Evolution: Stalking The View Matcher in Design Space: Carroll, J.M. and Rosson, M.B., *Human-Computer Interaction*, 6, 3&4: 281-318 (1991).

From "Folklore" to "Living Design Memory": Terveen, L.G., Selfridge, P.G. and Long, M.D., In *Proceedings of ACM INTERCHI'93 Conference on Human Factors in Computing Systems*, 1993, pp. 15-22.

Inside Multi-disciplinary Design in Medical Informatics: Experiences from the use of an Argumentative Design Method: Sjöberg, C. and Timpka, T., In *MEDINFO'95: Tri-annual World Conference in Medical Informatics*, 1995

QOC in Action: Using Design Rationale to Support Design: McKerlie, D. and MacLean, A., In *Video Program, InterCHI'93: Human Factors in Computing Systems*, 1993, ACM SIGGRAPH Video Review Series

Rationale in Practice: Templates for Capturing and Applying Design Experience: Casaday, G., To appear in: *Design Rationale: Concepts, Techniques, and Use,* (Moran, T.P. and Carroll, J.M., ed.), Lawrence Erlbaum Associates: Hillsdale, NJ, 1995, in press.

Reasoning with Design Rationale: Practical Experience with Design Space Analysis: McKerlie, D. and MacLean, A., *Design Studies*, 15, 2: 214-226 (1994).

Report on a Development Project Use of an Issue-Based Information System: Burgess Yakemovic, K.C. and Conklin, J., In *Proceedings of CSCW'90: Computer Supported Cooperative Work*, 1990, ACM: New York, pp. 105-118.

Summarising the Evolution of Design Concepts Within a Design Rationale Framework: Shum, S., MacLean, A., Forder, J. and Hammond, N.V., In *Adjunct Proceedings of InterCHI'93 (Short Papers)*, 1993, pp. 43-44.

Where Did You Put It? Issues in the Design and Use of a Group Memory: Berlin, L.M., Jeffries, R., O'Day, V.L., Paepcke, A. and Wharton, C., In *Proceedings of ACM/IFIP INTERCHI'93 Conference on Human Factors in Computing Systems*, 1993, pp. 23-30.

Organizational perspectives:

Organizational Memory. Walsh, J.P. and Ungson, G.R., *The Academy of Management Review*, 16, 1: 57-91, 1991.

Definitional and Contextual Issues in Organizational and Group Memories: Ackerman, M.S., In *Proceedings of HICSS'94: 27th Hawaii International Conference on System Sciences*, 1994

Evaluating Opportunities for Design Capture: Grudin, J., To appear in: *Design Rationale: Concepts, Techniques, and Use,* (Moran, T.P. and Carroll, J.M., ed.), Lawrence Erlbaum Associates: Hillsdale, NJ, 1995, in press.

Organizational Learning: A Review of Some Literatures: Dodgson, M., *Organization Studies*, 14, 3: 375-394 1993.