INTEGRATING DESIGN RATIONALE WITH A PROCESS MODEL

J. E. BURGE, D. C. BROWN
AI in Design Research Group
Department of Computer Science
WPI, 100 Institute Road
Worcester, MA 01609, USA

Abstract. One goal for having a process model is to guide the decisions made while designing. It is also useful to be able to describe the reasons behind the decisions, the design rationale. There are different types of rationale that may be involved in designing: rationale about the process, about the type of item or system being designed, and about the resulting design itself. Each type has different uses, some during design and some after designing is complete. We describe these types of rationale, first in the context of the DSPL hierarchical plan selection and refinement approach and then in the context of the software development process the Unified Process.

1. Introduction

The *design process* is the set of steps, or activities, that take place in achieving the design goals, or objectives [Lossack, 2000; Tate and Norlund, 1996]. Models of the design process are used in order to either describe the activities of the design process or prescribe how the designing should be done [Tate & Norlund, 1996; Cross, 1994]. Many decisions need to be made while designing. A process model can assist in guiding what decisions should be made when, and, if the model describes the design of a specific artifact, can even provide the knowledge to be used to make the decisions.

One aspect to designing that is not always captured as part of the design is its rationale. The design rationale captures not only the results of the design decisions (which would normally be an output of the design process) but also the reasons behind them and the alternatives considered. The explicit capture of the design alternatives and their rationale could be integrated into a design process model. This would be beneficial to the designer by encouraging the consideration of alternatives. Providing information about why design decisions were made would be useful during any future revision or reuse of the design.

The process model could also be annotated with the rationale behind the process. That could be used to explain the process to designers who are learning about it and to determine how the process could be tailored to meet the needs of a specific design project.

Section 2 of this paper gives a brief discussion of related work in process modeling in design rationale, section 3 describes the expected impact of design rationale integration with a process model, section 4 gives an example based on the hierarchical plan selection and refinement model of design, section 5 discusses possible integration with the Unified Process, a more general process for software design, and section 6 gives the summary and conclusions.

2. Related Work

There are many different types of process models proposed for many different types of design and design domains. Some are aimed at a specific type of design or type of product. An example of this would be the hierarchical plan selection and refinement model used by the AIR-CYL system that performs routine parametric design of Air Cylinders [Brown & Chandrasekaran, 1989]. Others are intended to be general models used by the designer to come up with a more specific implementation. An example of this is Tate and Nordlund's [1996] design process roadmap.

Finger and Dixon [1989] provide a survey of design models of the mechanical design process. They break them into three types: descriptive, prescriptive, and computer-based.

There have been some models of design that specifically address capturing the rationale along with the design. Blessing's Design Matrix [2000] models design as a series of decisions that involve generating design alternatives, evaluating the alternatives, and selecting alternatives. Ganeshan, et. al. [1994] used a model of the design process as the means for both guiding the design itself and capturing the rationale (or intent) behind the design decisions. This is an interesting way to capture the rationale since the capture of the rationale is tightly integrated with the activity of designing and, therefore, ensures that rationale is captured while not placing an additional burden on the designer.

3. Impact of Process Model Integration with DR

For some types of design, the design process model has been used to build a *design system* that uses that process, driven by customer requirements, to produce designs. Figure 1 shows the relationships between the type of design, the design process model, the design system, and the design itself. It consists of the following parts:

- *Design Type* – this indicates the type of design being performed. This could refer to a general domain (mechanical vs. electrical) or it could refer to a class of design (parametric vs. configuration).
- *Design Problem Type* – this indicates a type of "thing" being designed. Examples would be a car, gear shaft, or text editor.
- *Design Problem* – this is a specific design problem, i.e. set of initial requirements, which describe what the designer needs to create.
- *Design Process Model* – this is a model of a design process that could be followed to produce designs of the design type.
- *Design System* – this is a system that uses the design process model to produce designs given customer requirements. This would be an intelligent software system or, if no automated system available, a human designer or designers.
- *Design Trace* – this indicates all the steps taken by the design system in producing a design to address the design problem.
- *Design Artifact* – this is the design produced by the design system.

Figure 1 also shows four types of rationale:

- *Design Process Rationale (R_{dp})* – this is the rationale behind the design process model itself. This would include the rationale behind the process steps and their ordering.
- *Design System Rationale (R_{ds})* – this is the rationale embedded in the design system. This gives the decisions that can be made by the system in creating a design and the reasons that the system would select one decision versus another.
- *Design Trace Rationale (R_{dt})* – this is the rationale containing the reasons for the specific design decisions made by the design system. It is a subset of the design system rationale but only contains the reasons for the selected design alternatives. The decisions and their rationale will be given in temporal order as they occurred during designing.
- *Design Description Rationale (R_{dd})* – this is the rationale for the design itself. This may include ingredients from any of the other types of rationale, but it is organized/indexed by the artifact that was designed. In addition, it can be structured according to the

argumentation that occurred during the design, as opposed to being strictly linear.

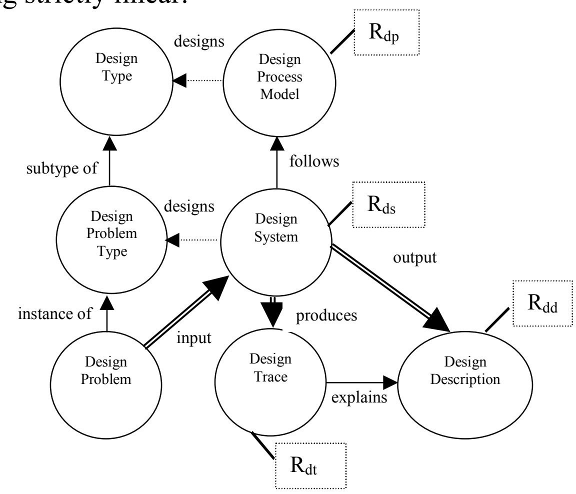

Figure 1. Problems, Processes & Rationale

3.1 MODEL IMPACT ON DR CONTENT

Lee's [1997] definition of rationale states: "design rationales include not only the reasons behind a design decision but also the justification for it, the other alternatives considered, the tradeoffs evaluated, and the argumentation that led to the decision." Generating rationale using integration with a process model will create a richer rationale than described by Lee's definition.

The rationale for the design process model (R_{dp}) describes why certain types of decisions were made and why they were made in a specific order. This includes reasons for proceeding to the next step of the design process.

The ability to capture a trace produced by the design system means that the design history could be saved as part of the rationale. This is not a type of rationale that is unique to process-based collection but it is rationale not explicitly mentioned in Lee's definition. This will give more than just a set of decisions—it will also give the order in which they are made.

Much of the design knowledge contained in the design process (which will vary depending on both the type of model and the item being designed) will become embedded in R_{ds}. For example, the different stages of the design process may contain sequencing and control knowledge that will drive the next steps of the design. This results in much, if not all, of the design system itself being stored as part of the rationale—this would allow for more powerful uses for the rationale (described in the next section) since knowledge about both the designing process (captured in R_{dp}) and the artifact being designed will be available. It also means that the structure of the rationale will be similar to that of the system. For example, if a design process followed by the system is hierarchical, the rationale will also be hierarchical; if the design process proceeds in stages, there will be rationale for the stages.

The standard view of rationale, the decisions, alternatives, and justifications, will be captured by a combination of R_{dt} (giving which alternatives were selected and why) and R_{ds} (giving the other alternatives that were considered).

3.2 THE MODEL'S IMPACT ON DR USE

Ganeshan, et. al. [1994] suggest several uses of the design rationale. One is for traceability. Another is to support modification of the design by using the model to determine what would be affected by a design modification. In some cases it may be possible to replay the entire design process to create a new design from a specification different from, but similar to, the original. It may also be possible to use the rationale along with the model to see the impact of changing one of the decisions.

The model also suggests what DR is useful at various points in the design process. This can be used in assisting DR capture by suggesting which rationale should be captured when.

Using the process model based approach will still allow the system to provide retrieval-based services such as allowing the designer to ask why a particular decision was made or not made. One disadvantage of a model-based approach is that if the model produces the rationale, the answers to these queries may not be in natural language unless the model contains comments.

4. Example: Design Rationale and Hierarchical Plan Selection and Refinement (HPS&R)

The Design Structures and Plans Language (DSPL) is used in systems, such as AIR-CYL [Brown & Chandrasekaran, 1989], that use the hierarchical plan selection and refinement approach. The type of design supported involves selecting from pre-determined sets of design alternatives. DSPL is used to encode both the design plans and their relationships, and to encode the problem solving and control knowledge for this design process [Brown, 1992]. The design is performed by a series of "specialists" where each specialist selects from a set of pre-defined plans that are followed to perform the part of the design that the specialist is responsible for. Figure 2 shows the relationships between the DSPL process (hierarchical plan selection and refinement), the DSPL system, and the artifact designed (the Air Cylinder).

4.1 RATIONALE CONTENT

There are several types of decisions made during the plan selection and refinement process. One is to decide which set of plans the specialist should select. In DSPL, a plan Selector uses evaluations provided by a Sponsor for each plan to select which plan should be pursued [Brown & Chandrasekaran, 1989]. For this decision, the alternatives are the plans themselves and the rationale for the plan selection consists of the evaluation of each plan provided by the plan Sponsor, the criteria used by the Sponsor in making that evaluation, and the reasons for the final Selection (in some cases there may be only one plan that is acceptable).

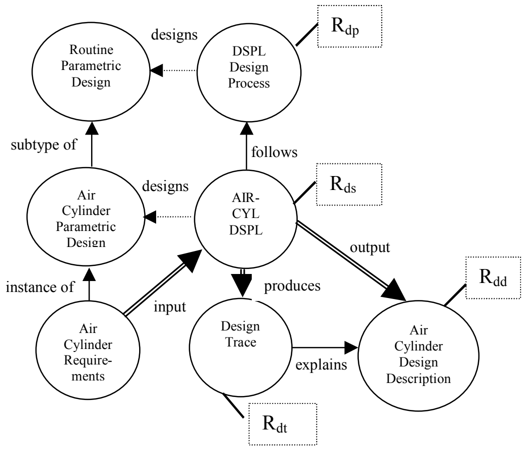

Figure 2. Air Cylinder Design

Plans consist of either calls to tasks or to more specialists, which in turn select plans (this is the hierarchical refinement part of the process). Tasks consist of Steps where each step defines a value for a design attribute. The steps also need to make decisions through calculations, choices, and constraints [Brown & Chandrasekaran, 1989]. The alternatives consist of the different attribute values considered and/or attempted and the reasons for choosing or rejecting them are the calculations and constraints encoded as part of the step. These different alternatives would be captured as R_{ds}.

At some point in the design, failure may occur. When this happens, the designer will need to decide how to “fix” the design in order to proceed with the designing process. The rationale will consist of the reasons for the failure (the symptoms), the process taken to find the cause of the failure, what that cause was, and the steps taken to make the correction. This information would be captured as R_{dt} and R_{dd}.

4.2 RATIONALE USES

Some rationale, reasons for and against the various plans, is used during the initial designing of the artifact. There are reasons embedded into the design model that, when used with the specific requirements for this design, indicate which plans should be followed next.

Rationale is also used during failure handling by indicating where there were alternative ways of proceeding with the design that could have been used. Storing information on which plans had been tried and failed previously is also useful if failures continue to occur. It may be possible to keep track of common failures and feed that information back into the model–if the same problems keep occurring then maybe the model itself needs modification. Brown and Chandrasekaran [1989] refer to this as pruning of suggestions.

Another use of rationale/history generated from a process is that it could be used to replay the design on a different set of requirements. The same decisions made earlier could be made on the new design. Re-using a design for a new one with a similar specification is one of the uses proposed by Ganeshan, et. al. [1994].

There are also some “standard” rationale uses that could be supported by this approach. These include being able to ask the design why certain decisions were made, such as why a particular value was assigned for a parameter. The usefulness of the answer will depend on how much information was encoded into the model to describe choices. It will also be possible to ask what types of failures occurred, and why. It may also be possible to use the rationale and the model to assess the impact of changing a parameter value by replaying the design to the point where the value was assigned and then continued to see what happens if a different value is chosen.

5. Design Rationale and the Unified Process

HPS&R is a very specific model, corresponding to a well-studied design problem. How could rationale be used for more general design situations such as generic software design? In particular, we are interested in the rationale generated during the Unified Process for software development [Jacobson, et. al., 1999]. Unlike the AIR-CYL problem, there are no automated systems available that can use the Unified Process to transform user requirements into a completed software system as depicted in Figure 3.

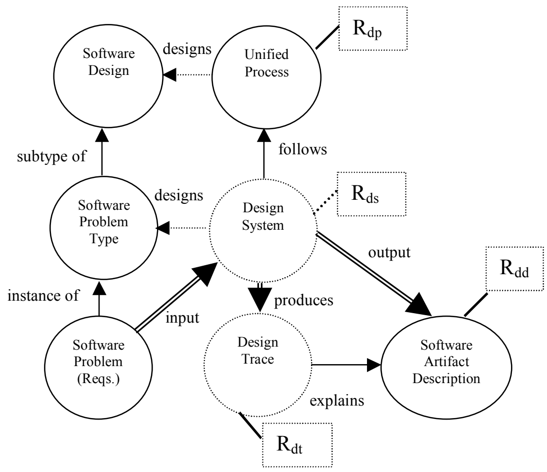

Figure 3. Software Design

What should be possible, however, is to obtain the rationale for the Unified Process (R_{dp}) and store it as part of a process model. The rationale for the process can be used in tailoring the process for the needs of the specific design problem. The Unified Process is large and complicated and not all of it is appropriate for every design task. The rationale behind the process could be used to help developers decide which portions of the process are most applicable to their design problem. The rationale would also be helpful in teaching developers about the Unified Process.

It is also possible to determine what some alternatives might be for decisions required by the process for the software problem type and store their rationale as R_{ds}. In this case, the system shown in Figure 3 would be the human designer. There are many decisions that may be required in most design projects that would be assisted by such rationale. It would encourage the developers to consider multiple alternatives. If the information is readily available for many options then this will help avoid the tendency to fixate on and choose the first option considered. This will also provide some initial rationale and reduce the amount of effort in rationale capture.

Figure 4 (adapted from Jacobson, et. al. [1999]) shows the workflow (i.e. the process model) for the Design Phase of the Unified Process. There is some iteration in the transitions between the phases as the design artifacts are refined.

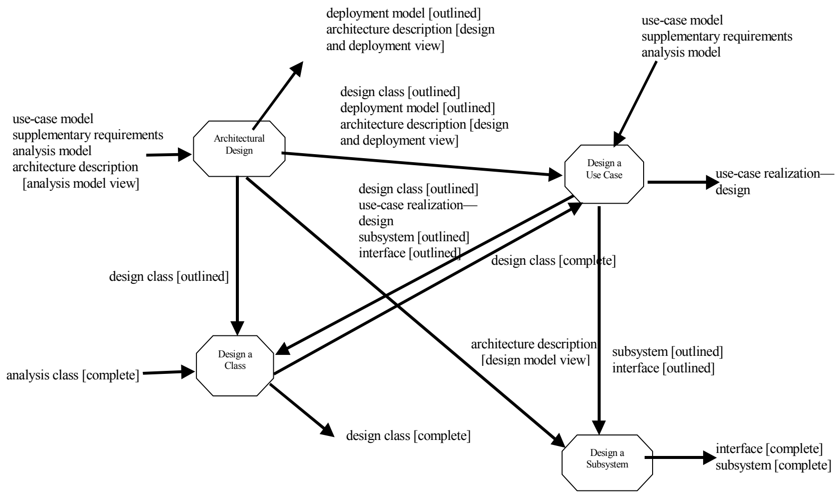

Figure 4. Design Workflow for the Unified Process

Each process within this workflow has certain tasks it must carry out in order to produce the artifacts shown in this diagram. One example is the Architectural Design process. This consists of the following tasks [Jacobson, et. al., 1999]:

- Identifying network nodes and their configuration;
- Identifying subsystems and their interfaces;
- Identifying architecturally significant design classes;
- Identifying generic design mechanisms.

Rationale can be generated for why these tasks are necessary, and (if applicable) in what order they should be performed. For example, the first task in the list will only be necessary for distributed systems. This condition should be stated as rationale (R_{dp}) for this task. If node identification is applicable, rationale for what steps should be taken and what information is needed to complete the task should also be stored as R_{ds}.

Rationale can also be generated for frequently considered alternatives for some design decisions. For example, there are a number of different, but well studied, network configurations. These configurations and their rationale could be the R_{ds} for this part of the development process.

6. Summary and Conclusions

In this paper, we defined different types of rationale that occur during design. These are the rationale behind the process used (R_{dp}), rationale in the system doing the designing (R_{ds}), rationale generated in the design trace created by the system (R_{dt}), and rationale for the specific design developed for the customer (R_{dd}). Each type of rationale has its own role during the design process.

The routine parametric design example shows how rationale can be generated and used for a well-studied specific type of design. Software is a more general case since the types of software, designs, and processes are unbounded. The Unified Process is one widely known software development process that could benefit from modeling with rationale attached. We described one aspect of this model, the architectural design task within the design phase, which could be annotated with both rationale about the process and rationale for design of specific types (e.g., distributed) of software systems. The system type rationale would contain alternatives frequently considered for this type of design and the reasons for and against each one. There are many other tasks within this process that could be annotated in a similar fashion. The Unified Process should be studied further to determine where rationale could be used to assist in software development and process definition.

References

Blessing, L.: 2000, The Design Matrix, *Workshop on Design Project Support Using Process Models, Artificial Intelligence in Design '00*, Worcester MA.

Brown, D.C.: 1992, The Reusability of DSPL Systems, *Workshop on Reusable Design Systems*, *Artificial Intelligence in Design '92*, Pittsburgh, PA.

Brown, D. C., Chandrasekaran, B.: 1989, *Design Problem Solving: Knowledge Structures and Control Strategies*, California: Morgan Kaufmann.

Cross, N.: 1994, *Engineering Design Methods*, John Wiley.

Finger, S., Dixon, J.R.: 1989, A Review of Research in Mechanical Engineering Design, Part I: Descriptive, Prescriptive, and Computer-Based Models of Design Processes, *Research in Engineering Design*, Vol. 1, pp. 51-67.

Ganeshan R., Garrett J., Finger, S.: 1994, A framework for representing design intent, *Design Studies Journal*, V15 No. 1, January, pp. 59-84.

Jacobson, I., Booch, G., Rumbaugh, J.: 1999, *The Unified Software Development Process*, Addison-Wesley Publishing Co.

Lee, J.: 1997, Design Rationale Systems: Understanding the Issues, *IEEE Expert*, Vol. 12, No. 3, pp. 78-85.

Lossack, R.: 2000, A Design Process Model, *Workshop on Design Project Support Using Process Models, Artificial Intelligence in Design '00*, Worcester MA.

Tate, D., Nordlund, M.: 1996, A Design Process Roadmap as a General Tool for Structuring and Supporting Design Activities, *Proc. of the 2nd World Conf. on Integrated Design and Process Technology*, Society for Design and Process Science, Austin, TX, pp. 97-104.