A Framework for Constructive Design Rationale

Udo Kannengiesser[1] and John S Gero[2]
[1] *NICTA, Australia, and School of Computer Science and Engineering, University of New South Wales, Sydney, Australia*
[2] *Krasnow Institute for Advanced Study and Volgenau School of Information Technology and Engineering, George Mason University, USA, and University of Technology, Sydney, Australia*

This paper proposes a framework for describing design rationale as a constructive notion rather than a fixed record of design reasoning. The framework is based on two views: an instance-based view of design rationale as an ordered set of decisions, and a state-space view of design rationale as a space of solution alternatives. The two views are connected with each other using the function-behaviour-structure (FBS) ontology. Constructive design rationale is defined and categorised based on reformulations of the function, behaviour or structure of the rationale. The drivers of the different reformulations are represented in the situated FBS framework.

Introduction

In design a typical ontology is concerned with either the object being designed or the processes of designing and hence views designing a forward moving activity; it is forward looking. ("Design" is used to indicate the output and "designing" the process that produces a design.) Design rationale is concerned with tracking the decisions that were taken to reach the design during the process of designing [1, 2]. In this it is backward looking in the same sense that history is backward looking. Design rationale can be equated to a truth maintenance system in that it aims to represent the beliefs behind the decisions and their dependencies.

One view of design rationale expands the elementary description given above to include not only the basis of the decision as listed but allows for any basis that produces the same decision. For example, in a closed world

pp. xx-yy.

defined by a set of rules, this is conceptually similar to allowing for potential multiple paths leading to a goal, where the paths model the rationale for the decision. As a consequence even if just part of or even the entire path taken that leads to a goal is retracted it does not necessarily follow that the goal is incorrect as there may be other paths that would support the goal. The same idea applies in an open world. It is consistent with the common observation that when designers are asked for the rationale underpinning their designs, they rarely produce pre-conceived or pre-recorded explanations but construct them on the fly [3]. The rationale they construct is adapted to the specific question being asked in the specific situation, which emerges from factors such as the presumed expertise and goals of the recipient of the rationale and expectations of accuracy and relevance. We refer to this notion as *constructive* design rationale.

In contrast, most computational models assume design rationale to be a fixed record of the design process. One of the problems associated with this "record-and-replay" [3] paradigm is that fixed rationale needs to be recorded as designing unfolds. Recording design rationale is a time-consuming activity that causes significant overhead for designers [4]. Recent approaches aim towards reducing this effort by providing integrated design environments that can automatically record design rationale [5]. However, over the course of several design projects, and even within the same design process, the relevance and applicability of a recorded design rationale decreases. This is because many design projects are unique, and rationale instances are based on assumptions [6] that frequently change during designing.

This paper proposes a framework of design rationale as a constructive rather than a fixed representation of designing. It provides an ontological basis for developing new design support systems that can generate different rationales for different situations.

An Ontological Representation of Design Rationale

The FBS Ontology

Modelling design rationale is facilitated by using an ontological framework that provides a common terminology with agreed meanings for a domain of discourse. The function-behaviour-structure (FBS) ontology [7, 8] provides such a framework for the design domain.

- *Function* (F) of an artefact is its teleology ("what it is for"). An example is the function "to wake someone up" that humans generally ascribe to the behaviour of an alarm clock.
- *Behaviour* (B) of an artefact is the attributes that can be derived from its structure ("what it does"). An example of a physical artefact is "weight", which can be derived directly from the product's structure properties of material and geometry.
- *Structure* (S) of an artefact is its components and their relationships ("what it consists of"). For physical artefacts, it comprises geometry, topology and material.

Humans construct relationships between function, behaviour and structure through experience and through the development of causal models based on interactions with the artefact. Function is ascribed to behaviour by establishing a teleological connection between the human's goals and measurable effects of the artefact. Behaviour is causally related to structure, i.e. it can be derived from structure using physical laws or heuristics. This may require knowledge about external (or exogenous) effects and their interaction with the artefact's structure. There is no direct relationship between function and structure.

Instance-Based and State-Space Views of Design Rationale

Design rationale can be conceptualised in three ways [9]: (1) as a historical record of designing, (2) as a set of claims about the properties embodied by an artefact, or (3) as a space of possible design alternatives.

The first two views can be combined into an instance-based view of design rationale as a history of specific decisions, about artefact properties or about the design process. Decisions on the decision alternatives are not independent of one another. A variety of dependencies can exist between design decisions [10]. During designing only a subset of these dependencies is known or anticipated in advance. A great deal of designing is based on the designer's assumptions that may be incomplete and incorrect, or that may change later in the process. Yet, the dependencies assumed by the designer are used to proceed from one decision to the next, thus generating a path through a network of possible design decisions, Figure 1. The instance-based view of design rationale can be understood as comprising three elements:

- *Starting point*: is an antecedent design decision
- *End point*: is a specific decision alternative selected for a consequent decision
- *Path*: is a sequence of intermediate decisions connecting the starting point and the end point. *Elementary* paths include only one in-

intermediate decision that is concerned with establishing a basis for assessing and then selecting specific decision alternatives. *Complex* paths include multiple intermediate decisions that create supplementary knowledge needed for reaching the end point. This includes, for example, decisions related to information-seeking activities and decisions on other design issues affecting the issue under consideration.

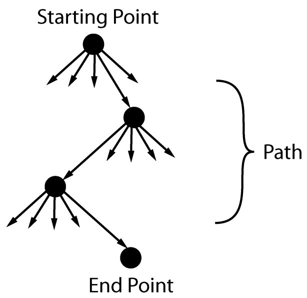


Fig1. A rationale instance described as a directed graph, with nodes representing decisions

In this paper, we refer to design rationale modelled in terms of these three elements as a *rationale instance*. Table 1 shows an example of a rationale instance in the context of a conceptual design process of a monitoring system for drilling tools. Here, the end point decision on using laser sensors (rather than other types of sensors) is reached from the starting point decision on using automatic monitoring technology, and following a (complex) path that includes the intermediate decisions on monitoring tool breakage for tool diameters that may be as small as 0.8mm.

Table 1 Example of a rationale instance for the conceptual design of a monitoring system for drilling tools. Selected decision alternatives are represented in italics.

Rationale element	Decision problem	Decision alternatives
Starting point	What level of automation?	*Automatic*, Semiautomatic, manual
Path	What data is monitored?	Tool wear, *Tool breakage*
	For what tool diameters?	> 10mm, > 5mm, > *0.8mm*
End point	What sensor type?	Force, Acoustic, *Laser*, Strain

A state-space view of design rationale as a set of possible design alternatives can be understood as capturing classes rather than instances of design decisions. These generic classes of design decisions can be modelled as decision problems or issues associated with sets of alternative solutions, as shown in Table 1. This is consistent with the most common representations of design rationale. For example, in the QOC [11], IBIS [12] and DRL [9] approaches, issues are called “Questions”, “Issues” and “Decision Problems”, respectively. Alternative solutions are called “Options” in QOC, “Positions” in IBIS, and “Alternatives” in DRL.

Computationally, design decisions can be modelled as a state space in terms of variables and their ranges of values, where the variables correspond to issues and the ranges of values to a set of alternative solutions. A simple example of a design decision, taken from MacLean et al. [11], is the issue “how wide is the scroll bar” and the associated alternative solutions of “wide” and “narrow”. Here, the variable is “width of the scroll bar”, and its range of (qualitative) values comprises “wide” and “narrow”. Design decisions can be more complex, consisting of multiple variables where every variable represents a decision problem on a finer level of granularity that can itself be represented in terms of variables and ranges of values.

An FBS View of Design Rationale

Individual design decisions may deal with the function, behaviour or structure of the artefact. However, what they all have in common is that they compose structured sets of decision variables. They can be viewed as abstract yet first-class artefacts, an idea that has found recent interest in the software design community [13]. We can apply the notion of structure (S) in the FBS ontology to describe design decisions as artefacts, and refer to the space of design decisions as the *structure state space* of design rationale.

Most approaches to representing design rationale include the notion of criteria that are used in design decisions as a basis for evaluating, comparing and selecting alternative solutions. In the scroll bar example, the criteria presented by MacLean et al. [11] include “screen compactness” and “ease of hitting with the mouse”. Different alternative solutions fulfil these criteria to different extents. Criteria may be more or less formally defined, and may be qualitative or quantitative. In all instances, they represent the performance of a design decision, which in the FBS ontology is captured by the notion of behaviour (B). We call the set of criteria associated with a design decision the *behaviour state space* of design rationale.

Some accounts of design rationale include the strategies and goals underlying particular design decisions. This allows reasoning about the process or plan of designing, and establishes a basis for deriving criteria in accordance with particular goals [14]. The overall goal of a design decision is to create knowledge for advancing the state of the design [15]. The knowledge created is often needed to support other design decisions. Generally, the goals of a design decision include creating knowledge for refining or realising prior decisions and for enabling or guiding subsequent decisions. For example, the decision on using a scroll bar may have the goal of refining the prior decision on using a graphical user interface, and the goal of enabling the subsequent decision on scroll-bar width. The notion of function (F) in the FBS ontology can be used to capture the goals associated with design decisions. The set of functions for a design decision then establishes the *function state space* of design rationale.

The union of the function state space, the behaviour state space and the structure state space of design rationale is termed the *rationale state space*. We can establish connections between the instance-based view and the FBS state-space view of design rationale:

- The *starting point* of a rationale instance is covered by the function state space. This is because functions relate a decision to other decisions, including those that occur prior to that decision. The notion of issues in starting point decisions is covered by function variables. The notion of solution alternatives in starting point decisions is covered by ranges of function values.
- The *end point* of a rationale instance is covered by the structure state space. This is because the end point is a specific, targeted decision that is a point in the structure state space. The notion of issues in end point decisions is covered by structure variables. The notion of solution alternatives in end point decisions is covered by ranges of structure values.
- The *path* of a rationale instance, if it is elementary, is covered by the behaviour state space. This is because behaviour provides a link between function and structure [16] by forming a basis for assessing different structures oriented to achieving given functions. Behaviour variables (and their ranges of values) can then be viewed as path variables (and their ranges of values). Variables of a complex path that correspond to a set of additional intermediate decisions are not covered in the FBS state-space view of the rationale instance. They can be mapped onto the function, behaviour and structure variables of those rationale instances that are associated with these intermediate decisions.

What is Constructive Design Rationale?

Design rationale is termed constructive if there are reformulations of any of the three subspaces of the rationale state space. The reformulations affect a state space in terms of its variables or their ranges of values. According to this definition, changes of values within the boundaries of an original state space are not considered constructive.

State spaces are constructed for a current problem in a particular situation. As a result, reformulating a state space can be as simple as changing expectations about the current problem, by taking into account existing knowledge about potential issues and potential solutions. This is akin to a recombination of known concepts. In other instances, reformulating the rationale state space may involve new knowledge that has not existed before, leading to what may be called innovative or creative design rationale. We can use notions from research in creativity to categorise these differences in meaning of the word “constructive”. Boden [17] draws a distinction between “historical” (or h-) creativity and “psychological” (or p-) creativity. H-creativity is the strongest form of creativity, where novelty is assessed in relation to the history of humankind. For example, the first steam engine was an h-creative design. P-creativity implies novelty with respect to the history of an individual. An architect designing a high-rise building using, for his or her first time, reflecting glass can be viewed as producing a p-creative design. H-creative designs must also involve p-creativity. This classification has been extended to include the notion of “situated” (or s-) creativity [18]. S-creativity is defined relative to the situation that pertains during the process of designing. A design or design feature is s-creative if it is the result of a change of the world within which designing operates. P-creativity must involve s-creativity.

The notion of constructive design rationale developed in this paper corresponds to the concept of s-creativity. By analogy, we may refer to it as “s-constructive” design rationale, and distinguish it from “p-constructive” and “h-constructive” design rationale. However, for reasons of simplicity in this paper, we will just use the term “constructive” and define it in the sense of “s-constructive”.

Constructive design rationale allows producing rationale instances that have at least one element that is constructed: the starting point, the end point, or the path.

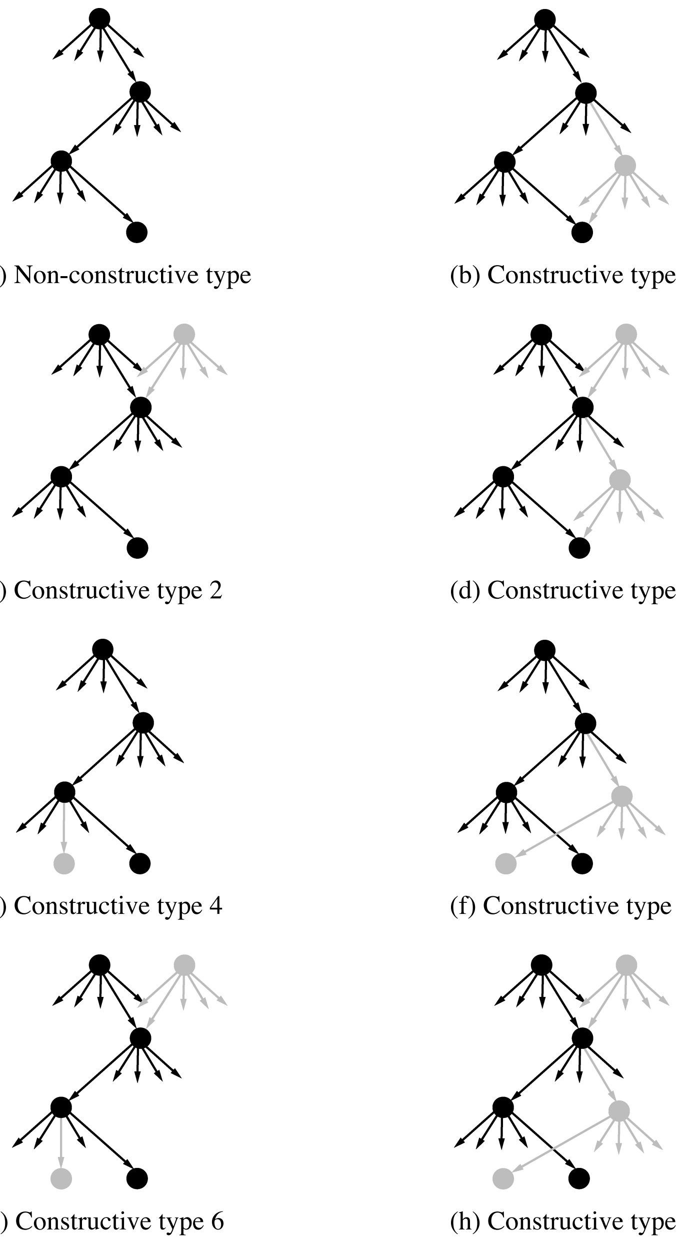
- *Constructed starting points* are based on novel issues or novel solution alternatives of antecedent decisions. They require reformulating the function state space in terms of its variables or ranges of values.

- *Constructed end points* are based on novel issues or novel solution alternatives of consequent decisions. They require reformulating the structure state space in terms of its variables or ranges of values.
- *Constructed paths* are based on novel issues or novel solution alternatives of intermediate decisions, and novel connections between intermediate decisions. For elementary paths, this requires reformulating the behaviour state space in terms of its variables or ranges of values. For complex paths, this may also require reformulating the individual rationale state spaces associated with intermediate decisions. In particular, reformulating the structure state space of an intermediate decision produces new intermediate decision variables or ranges of values. Reformulating the function state space of an intermediate decision produces new connections between intermediate decisions.

There are only seven combinations of constructed and non-constructed elements of constructive rationale instances, as shown in Table 2. Figure 2 shows graphically each of these combinations, Figures 2(b)-(h), contrasted with an instance of traditional, non-constructive rationale, Figure 2(a). Each of the seven combinations represents a type of constructive rationale. These types can be further elaborated, as non-constructed elements may be either fixed (i.e., their values remain unchanged) or variant (i.e., their values vary within the pre-defined ranges of the state space). Table 3 gives an overview of the 19 possible sub-types based on combinations of constructed, variant and fixed elements of constructive rationale instances.

Table 2 The seven possible types of constructive design rationale, based on different combinations of constructed and non-constructed elements

Type	End point	Starting point	Path
1	Non-constructed	Non-constructed	Constructed
2	Non-constructed	Constructed	Non-constructed
3	Non-constructed	Constructed	Constructed
4	Constructed	Non-constructed	Non-constructed
5	Constructed	Non-constructed	Constructed
6	Constructed	Constructed	Non-constructed
7	Constructed	Constructed	Constructed



(a) Non-constructive type
(b) Constructive type 1
(c) Constructive type 2
(d) Constructive type 3
(e) Constructive type 4
(f) Constructive type 5
(g) Constructive type 6
(h) Constructive type 7

Fig2. Graph-based representations of rationale instances, including the non-constructive type (a) and the seven constructive types listed in Table 2 (b to h). Constructed elements are in grey; non-constructed elements are in black.

Table 3 The nineteen possible sub-types of constructive design rationale, based on different combinations of constructed, variant and fixed elements

Type	Sub-Type	End point	Starting point	Path
1	1.1	Fixed	Fixed	Constructed
	1.2	Fixed	Variant	Constructed
	1.3	Variant	Fixed	Constructed
	1.4	Variant	Variant	Constructed
2	2.1	Fixed	Constructed	Fixed
	2.2	Fixed	Constructed	Variant
	2.3	Variant	Constructed	Fixed
	2.4	Variant	Constructed	Variant
3	3.1	Fixed	Constructed	Constructed
	3.2	Variant	Constructed	Constructed
4	4.1	Constructed	Fixed	Fixed
	4.2	Constructed	Fixed	Variant
	4.3	Constructed	Variant	Fixed
	4.4	Constructed	Variant	Variant
5	5.1	Constructed	Fixed	Constructed
	5.2	Constructed	Variant	Constructed
6	6.1	Constructed	Constructed	Fixed
	6.2	Constructed	Constructed	Variant
7	7.1	Constructed	Constructed	Constructed

Drivers of Constructive Design Rationale

This Section presents the drivers of constructive design rationale using the situated FBS framework [8].

The Situated FBS Framework of Designing

This Section provides a brief description of the situated FBS framework; for more information see Gero and Kannengiesser [8].

The basis for the situated FBS framework is a three-world model of designing interactions, Figure 3(a). The *external world* is composed of representations outside the designer or design agent. The *interpreted world* is built up inside the design agent in terms of sensory experiences, percepts and concepts. It is the internal representation of that part of the external world that the design agent interacts with. The *expected world* is the world imagined actions of the design agent will produce. It is the environment in which the effects of actions are predicted according to current goals and interpretations of the current state of the world.

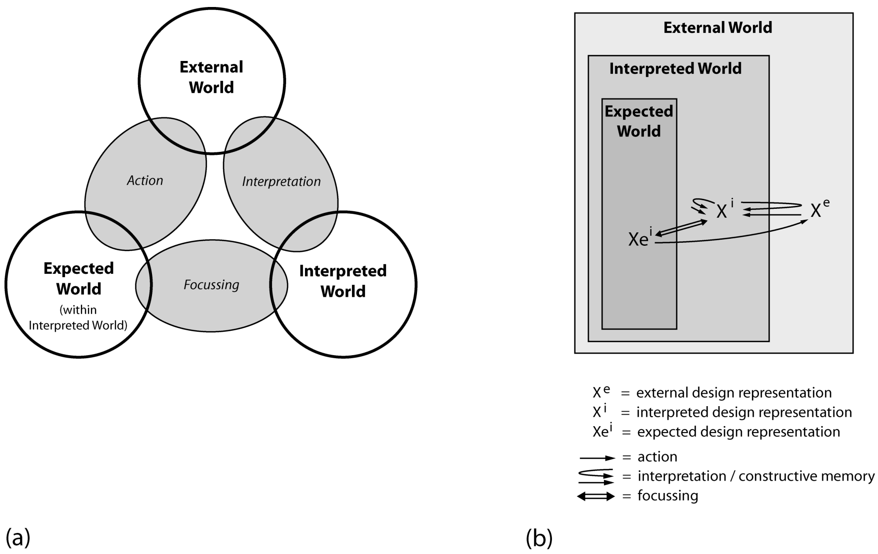


Fig3. Three interacting worlds: (a) general model, (b) specialised model for design representations

These three worlds are linked together by three classes of connections. *Interpretation* transforms variables which are sensed in the external world into the interpretations of sensory experiences, percepts and concepts that compose the interpreted world. *Focussing* takes some aspects of the interpreted world, and uses them as goals for the expected world that then become the basis for the suggestion of actions. *Action* is an effect which brings about a change in the external world according to the goals in the expected world.

Figure 3(b) specialises this model by nesting the three worlds and articulating general classes of design representations as well as the activity of reflection [19]. The set of expected design representations (Xe^i) corresponds to the notion of a design state space, i.e. the state space of all possible designs that satisfy the set of requirements. This state space can be modified during the process of designing by transferring new interpreted design representations (X^i) into the expected world and/or transferring some of the expected design representations (Xe^i) out of the expected world. This leads to changes in external design representations (X^e), which may then be used as a basis for re-interpretation changing the interpreted world. Novel interpreted design representations (X^i) may also be the result of *constructive memory*, which can be viewed as a process of interaction among design representations within the interpreted world rather than across the interpreted and the external world. Both interpretation and constructive memory are represented as "push-pull" activities [20]. This emphasises the role of individual experience in constructing the interpreted world, by "pulling" interpreted representations rather than just by "pushing" what is presented in the external world. It is the interaction of push and pull that may produce new representations that can be used to modify the design state space.

The situated FBS framework, Figure 4, combines the FBS ontology with the three-world model. Here, the variable X in Figure 3(b) is replaced with the more specific representations F, B and S. The situated FBS framework also uses explicit representations of external requirements given to the designer. Specifically, there are external requirements on function (FR^e), external requirements on behaviour (BR^e), and external requirements on structure (SR^e). However, we assume that there are no external requirements when applying the FBS ontology to modelling design decisions.

Drivers for Constructing Rationale Structure

Reformulation of structure (process 9 in Figure 4) covers constructing new end points of constructive design rationale. Two processes are potential drivers of this type of reformulation: the interpretation of an external structure (process 13), and the internal construction of an interpreted structure (process 6).

Interpretation of an end point decision (or external structure; process 13) is very common, as most rationale instances are generated based on a given decision that is available in the external world. Two of the most frequent scenarios that trigger the construction of a rationale instance based on a given end point include:

- Design justification: The design agent is given a particular decision (end point) and asked to communicate the reasoning that has led to this decision (starting point and/or path).
- Designing: The design agent interprets a potential decision on some aspect of a current design (end point) and reflects upon the reasoning that could lead to this or another decision (starting point and/or path).

In both scenarios, there is the potential for producing a different interpreted structure than originally intended by the design agent. This potential is enhanced by re-representations of the same external structure that may stimulate the emergence of new issues (or decision variables). Emergence is a process that makes implicit or unintentional design decisions explicit.

Emergent design decisions often include visual forms and their potential consequences, although they are not limited to visual forms. They are based on the fact that producing designs, by means of sketching or modelling, necessarily imposes decisions on the organization and details of the design, not all of which are specifically intended by the designer. For example, sketching components of a design on a piece of paper produces a set of lines that compose shapes with intended spatial relations. Other spatial relations emerge when the designer inspects the sketch at a later point in time.

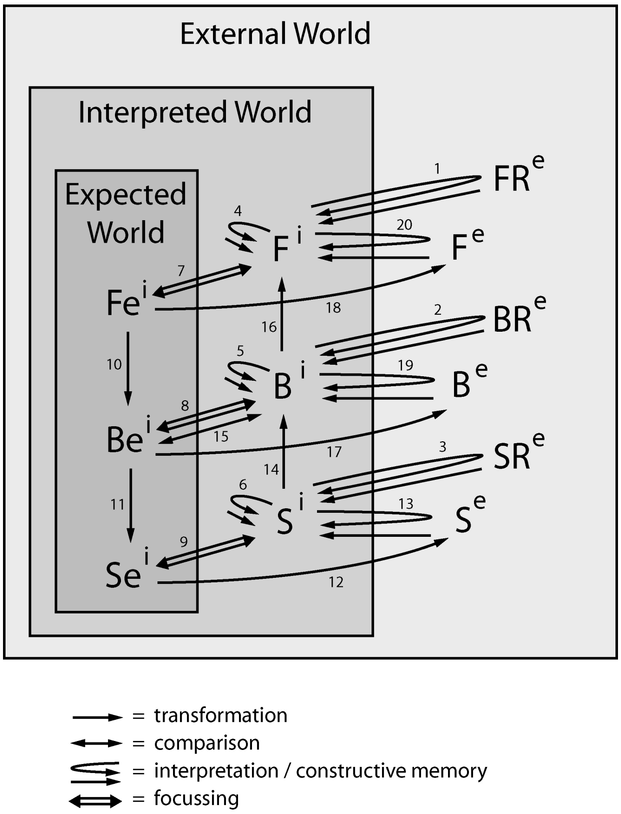
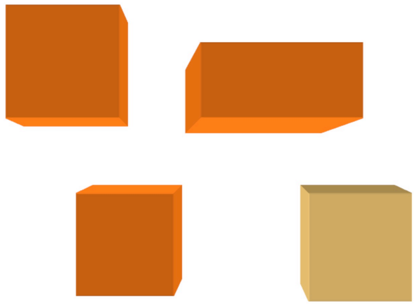
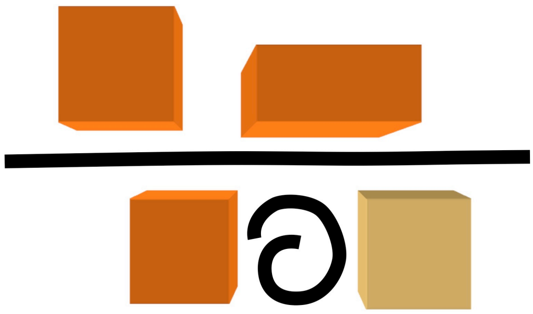


Fig4. The situated FBS framework

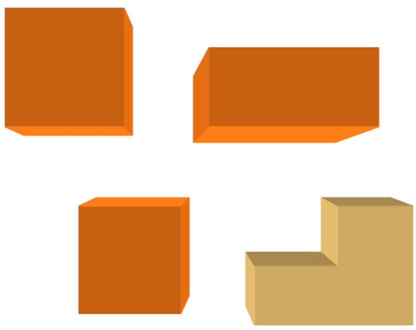
Take the layout of a set of buildings produced by an urban designer, shown in Figure 5(a). At the initial time of drawing the layout, the designer attends to the four buildings individually, leading to a set of independent decisions for each of them. Upon inspection of the layout, the designer becomes aware of a horizontal axis and an urban space between two buildings, as shown in Figure 5(b). These features are decisions on spatial relations that were implicit in the initial set of design decisions but are now made explicit. A rationale instance that comprises this constructed decision as an end point is of type 4 in Table 2.



(a)



(b)



(c)

Fig5. A sequence of sketches of a town layout: (a) the initial layout; (b) the same layout highlighting an emergent urban space and horizontal axis; and (c) subsequent change of the design as a consequence of the emergent urban space in (b)

Internal construction of new end point decisions (process 6) often occurs in the form of new solution alternatives that were not explicitly considered in the original decision-making process. This expands the ranges of values for decision variables. One benefit of this is that a design decision can be shown to remain valid or appropriate even when new decision alternatives, such as new technologies and competitors, come up later. And if one of these alternatives proves to be a better candidate decision, the design may be modified to incorporate it and to provide a closer fit with the design requirements.

Drivers for Constructing Rationale Behaviour

Reformulation of behaviour (process 8) covers constructing new elementary paths of constructive design rationale. New complex paths are covered by reformulation of the function and structure of intermediate decisions. Three processes are potential drivers of this type of reformulation: the interpretation of an external behaviour (process 19), the internal construction of an interpreted behaviour (process 5), and the derivation of interpreted behaviour from interpreted structure (process 14).

Interpretation can produce new interpreted behaviour (process 19) most commonly when a new external behaviour is also provided. The new behaviour may be in addition to or in conflict with previous behaviour. For example, the question "Does your decision on construction materials also consider recyclability besides strength?" represents a new, additional behaviour (i.e., recyclability). The question "Given we had to reduce our limit for material cost from $20 to $15 per unit, is our decision on suppliers still valid?" represents a new range of behaviour that is partially in conflict with the previous range of behaviour. The rationale instances resulting from additive and substitutive changes of behaviour are of type 1 in Table 2.

Internal construction of new behaviour (process 5) is often the consequence of inferring new interactions of end point decisions with exogenous effects. For example, let us assume an end point decision on using a particular part supplier. The path previously included cost per unit as a decision criterion. However, new government regulations (an exogenous effect) may require a minimum percentage of parts to be manufactured in a specific country, so the criterion of geographical location of the supplier must be constructed. This may add to or replace the previous cost criterion. The resulting rationale instance is of type 1.

Deriving new interpreted behaviour from interpreted structure (process 14) is usually the consequence of a reformulated structure. Returning to the example in Figure 5(b), the emerging decision on creating an urban space provides the basis for deriving decision criteria such as "support social interaction" and "provide a space for public events". As this corresponds to constructing a new path in addition to the new end point, this rationale instance is of type 5.

Reformulated behaviours may lead to subsequent refinements of end point decisions. For example, the urban designer may use the new criteria of "support social interaction" and "provide a space for public events" to more directly produce an urban space. Figure 5(c) shows how the emerged urban space is modified by changing the design of an individual building. This refined end point decision can be modelled in the FBS framework as the result of synthesis (processes 11 and 12), analysis (processes 13 and 14) and evaluation (process 15).

Drivers for Constructing Rationale Function

Reformulation of function (process 7) covers constructing new starting points of constructive design rationale. Three processes are potential drivers of this type of reformulation: the interpretation of an external function (process 20), the internal construction of an interpreted function (process 4), and the ascription of interpreted function to interpreted behaviour (process 16).

Interpretation can produce new interpreted function (process 20) most commonly when also a new external function is provided. This can occur when the robustness of an existing end point decision is assessed by relating that decision to a hypothetical starting point (e.g., during design review meetings): "What if we decide on using a different operating system; would we still use the same user commands?" In most cases, however, previous starting points become invalid because of changes in the external requirements on the product, the design process and the project. In non-constructive design rationale, this new starting point would invalidate all consequent decisions, including the original path and the original end point. In constructive design rationale, this need not be the case. An example is the decision to no longer outsource the manufacture of a physical part but to produce it in-house. The previous path included the decision on using a specific OEM parts catalogue; and the previous end point was a decision on a specific geometry of the part that is consistent with the catalogue. Based on considerations of maintainability and its associated principles of standardisation, the same path can be used leading to the same end point. This rationale instance is of type 2.

Internal construction of new function (process 4) can similarly produce new starting points, based on the designer's changed understanding of the design problem. Take the example from designing a distributed software system; an original starting point here is the decision to allow for extensibility of the system. The original path from this starting point includes a decision to use loose coupling, leading to an end point decision to use a Publish/Subscribe messaging model. Based on the designer developing a better understanding of the domain, the original starting point is modified to include a decision to allow for a high degree of system security. This leads to a modified path that includes a decision to use a mechanism that easily filters messages and then routes them according to their content. The Publish/Subscribe model still performs well under this additional criterion, and remains the chosen end point decision. This rationale instance is of type 3.

Ascribing new interpreted function to interpreted behaviour (process 16) is often triggered by reformulated behaviour. For example, the new behaviours associated with the decision on the urban space in Figures 3(b) and 3(c) may establish a basis for ascribing the new functions "to refine the decision to design for increased quality of urban life" (a new starting point) and "to guide the decision on what particular social activities should be supported in the urban space" (a new consequent set of decisions). The resulting rationale instance is of type 7, as all three elements of this instance are constructed.

Reformulated functions may lead to the formulation of new behaviours (via process 10), corresponding to constructing new elementary paths. The new behaviours can then be used to synthesise, analyse and evaluate refined end point decisions. New complex paths may need to be constructed by (re-) formulating the structure and/or function of intermediate decisions.

A new function and a new structure but with the same behaviour can occur, type 6 constructive rationale, when the designer's starting point has changed but leads to the same intermediate decision as previously existed and there is a change in the final decision of the rationale instance leading to a different structure than before. A new function, either as a result of an exogenous activity or as a result of emergence, changes the starting point of the rationale instance. The additional requirement that the artefact be collapsible may produce no change as that behaviour may already be embedded in its design. However, it will produce some different values for the variables that are propagated down the decision path. As a consequence it is possible that the final decision will therefore be different even though the same path as previously has been followed.

Conclusion

Design rationale can be understood either as a passive and fixed description of the history of designing, or as a dynamic act that constructs the assumptions underpinning the design decisions as they are needed in a current situation.

The first way of understanding rationale has benefits in supporting routine designing and activities such as auditing, learning and design maintenance. However, the remaining problems of capture and reuse of rationale in situations that are novel and dynamic, require a more subtle view of design rationale as a dynamic act that allows instances of design reasoning to be constructed on the fly. In the light of a new situation, a new line of reasoning can thus be generated that can provide new explanations for existing design decisions that may or may not lead to modifications of the design. When a set of antecedent decisions is invalidated or no longer available, a new set of decisions can be created without necessarily invalidating consequent decisions. In turn, new consequent decisions can be created without necessarily being in conflict with antecedent decisions. On the other hand, every new decision has the potential to affect other decisions, allowing for changes of both the design process and its outcomes to better adapt to different situations.

Our ontological framework can represent constructive design rationale as well as the drivers for constructing new decisions that form the starting points, the paths and the end points of rationale instances. The framework can be used for developing agent-based design rationale systems that not only capture and document rationale instances but also interpret them based on the agent's situation. This can produce different interpretations and thus different design rationale in different situations. Recent work on the relationship between design rationale and design creativity [21] can be supported by using constructive design rationale systems as testbeds for research hypotheses.

Future work includes validating our framework empirically. Studies need to capture initial rationale instances and their transformation as they are reconstructed by different designers or by the same designer at a later point in time. It would be interesting to establish which of the seven types and nineteen subtypes of constructive design rationale occur most frequently. The activities that drive the modification of rationale instances may be represented using an FBS-based coding scheme and then mapped onto the situated FBS framework.

Acknowledgments

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References

1. Moran T, Carroll J (eds) (1996) Design Rationale: Concepts, Techniques, and Use. Lawrence Erlbaum, Mahwah
2. Dutoit AH, McCall R, Mistrík I, Paech B (eds) (2006) Rationale Management in Software Engineering. Springer-Verlag, Berlin
3. Gruber TR, Russell DM (1996) Generative design rationale: Beyond the record and replay paradigm. In Moran T, Carroll J (eds) Design Rationale: Concepts, Techniques, and Use. Lawrence Erlbaum, Mahwah, pp. 323–349
4. Tang A, Babar MA, Gorton I, Han J (2008) A survey of architecture design rationale. The Journal of Systems and Software: 79:1792–1804
5. Burge JE, Brown DC (2008) Software engineering using RATionale. The Journal of Systems and Software: 81:395–413
6. Brown DC (2006) Assumptions in design and design rationale. In Burge JE, Bracewell R (eds) Workshop on Design Rationale: Problems and Progress. Design Computing and Cognition '06. Eindhoven, The Netherlands
7. Gero JS (1990) Design prototypes: A knowledge representation schema for design. AI Magazine: 11:26–36
8. Gero JS, Kannengiesser U (2004) The situated function-behaviour-structure framework. Design Studies: 25:373–391
9. Lee J, Lai K-Y (1991) What's in design rationale? Human-Computer Interaction: 6:251–280
10. Kruchten P (2004) An ontology of architectural design decisions. In 2nd Groningen Workshop on Software Variability Management. Rijksuniversiteit Groningen, The Netherlands
11. MacLean A, Young RM, Bellotti VME, Moran TP (1991) Questions, options, and criteria: Elements of design space analysis. Human-Computer Interaction: 6:201–250
12. Kunz W, Rittel H (1970) Issues as Elements of Information Systems. Working Paper 131. Institute of Urban and Regional Development, University of California, Berkeley, CA
13. Jansen A, Bosch J (2005) Software architecture as a set of architectural design decisions. In 5th Working IEEE/IFIP Conference on Software Architecture. Pittsburgh, PA, pp. 109–120
14. Lee J (1997) Design rationale systems: Understanding the issues. IEEE Expert: 12:78–85

15. Sim SK, Duffy AHB (2003) Towards an ontology of generic engineering design activities. Research in Engineering Design: 14:200–223
16. Qian L, Gero JS (1996) Function-behaviour-structure paths and their role in analogy-based design. Artificial Intelligence for Engineering Design, Analysis and Manufacturing: 10:289–312
17. Boden MA (1991) The Creative Mind: Myths and Mechanisms. Basic Books, New York
18. Suwa M, Gero JS, Purcell T (1999) Unexpected discoveries and s-inventions of design requirements: A key to creative designs. In Gero JS, Maher ML (eds) Computational Models of Creative Design IV. Key Centre of Design Computing and Cognition, University of Sydney, Australia, pp. 297–320
19. Schön DA (1983) The Reflective Practitioner: How Professionals Think in Action. Harper Collins, New York
20. Gero JS, Fujii H (2000) A computational framework for concept formation for a situated design agent. Knowledge-Based Systems: 13:361–368
21. Daughtry J, Burge J, Carroll JM, Potts C (2009) Creativity and rationale in software design. ACM SIGSOFT Software Engineering Notes: 34:27–29