Chapter 5

Representing Complex Problems: A Representational Epistemic Approach

Peter Cheng
R. Barone
University of Sussex

Understanding how to represent problems is a multidisciplinary endeavor. Cognitive science has shown the substantial impact that alternative forms of representation can have on the difficulty of solving problems (e.g., Kaplan & Simon, 1990; Kotovsky, Hayes, & Simon, 1985; Zhang, 1997). It has also provided explanations why certain forms of representation, such as diagrams, may confer substantial advantages on problem solvers (e.g., Cheng, 2004; Larkin & Simon, 1987). In computer science, the areas of information visualization and scientific visualization (e.g., Card, MacKinlay, & Shneiderman, 1999; Ware, 1999) have provided approaches to the design of displays that support the comprehension of large databases of information. Artificial intelligence has created systems that reason using diagrams, which gives theoretical insights into the nature of representations for problem solving (e.g., Glasgow, Narayanan, & Chandrasekaran, 1995). Studies in the area of human–computer interaction and human factors have provided guidelines for analysis and design of good notional systems and usable computer interfaces (e.g., Burns & Hajdukiewicz, 2004; Green, 1989). Psychology speaks to the design of effective representations for problem solving by informing us about the nature of underpinning human perceptual processes and cognitive processes (e.g., Kosslyn, 1989; Pinker, 1990).

Substantial advances have been made with respect to the representation of problems and the presentation of information within complex domains. However, our understanding of how to represent complex problems is still limited. The work on problem representations has tended to focus on relatively narrow domains, such as numeration systems (e.g., Zhang & Norman, 1994), or toy problems such as the mutilated checker board (e.g., Kaplan & Simon, 1990). Where work has focused on complex domains, it has tended to focus on the visualization of rich datasets to find hidden relations or patterns and often has little emphasis on problem solving more broadly conceived (e.g., Card et al., 1999; Ware, 1999).

The purpose of this chapter is to present the Representational Epistemic (REEP) approach to understanding and designing representations for complex problem domains. The approach was initially developed from studies on the role of alternative representations in scientific discovery (Cheng & Simon, 1995) and on the invention of novel diagrammatic systems to enhance conceptual learning in science and mathematics (Cheng, 2002; Cheng, 2003; Cheng & Shipstone, 2003). REEP has also been applied to the design of representations for event scheduling and personnel rostering (Barone & Cheng, 2004; Cheng, Barone, Cowling, & Ahmadi, 2002). We use the terms "representational" and "epistemic" for the approach because it focuses on the nature of representational systems to preserve the conceptual structure, or inherent system of knowledge, of the problem domain. A refinement to the approach is presented that involves theoretical advances that provide better definitions of key notions and a more coherent formulation of the REEP design principles.

Bakery production scheduling will be used as an example of a complex problem-solving domain throughout the chapter, to illustrate the key ideas underpinning the REEP approach and to explain how the design principles are applied. A problem-solving domain is defined as the problem-solving environment (e.g., bakeries) plus the range of tasks that are under consideration (e.g., production planning and scheduling). The application of the REEP approach to this domain is part of the ROLLOUT project, which has been found to improve bakery scheduling practices and training. The project involves a consortium of 10 commercial partners including a membership-based research organization, plant (factory scale) bakery companies, supermarket chains with in-store bakeries, and bakery equipment manufacturers. The project has conducted detailed studies of bakery scheduling knowledge and problem-solving strategies. The REEP approach has been used to design a novel graphical representation for bakery planning and scheduling—the ROLLOUT diagram. The ROLLOUT diagram has been used as the interface for a software tool that allows schedules to be modified through the interactive manipulation of the representation. Figure 5.1 shows a screen snapshot of the tool setup to model an in-store bakery of a

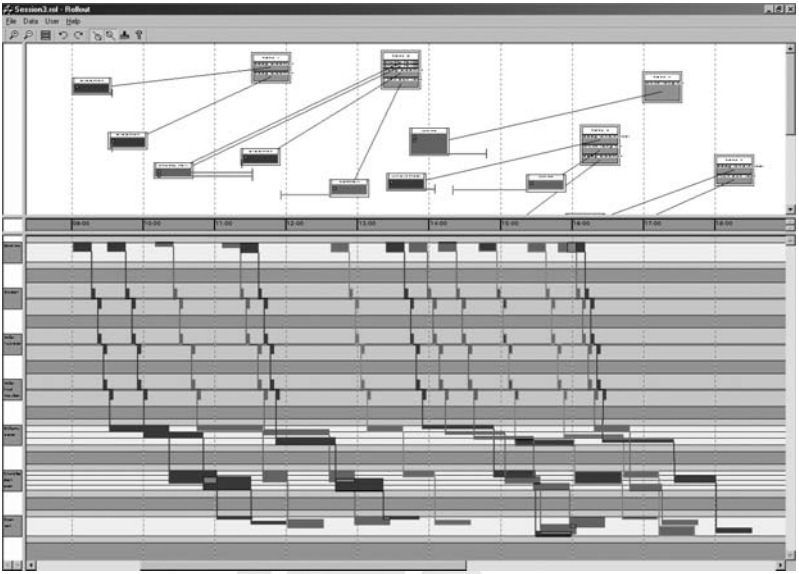

Figure 5.1. The ROLLOUT diagram for bakery scheduling.

supermarket. The effectiveness of ROLLOUT has been established using empirical evaluation, with favorable comparisons to familiar tabular representations and spreadsheet tools. The project studies real bakers in a range of trials, including the following: laboratory experiments using realistic schedules and a range of problem-solving tasks, evaluations in a training bakery configured as a simulation of a real bakery requiring real-time rescheduling of production, and trials of ROLLOUT in a real working bakery. Some of the outcomes are summarized by Cheng et al. (2006) and full details are reported elsewhere. The outcomes of all the trials demonstrate that ROLLOUT is at least as effective as conventional approaches, and usually superior. Bakery schedulers can quickly learn to use ROLLOUT successfully with minimal training. The success of ROLLOUT then in turn provides further evidence of the utility and benefits of the REEP approach to representation design.

The next section of this chapter considers the challenges to the design of effective representations for complex problems by discussing the various ways in which a problem domain may be complex. This provides a number of challenges that representations for complex domains must satisfy to be effective for complex problem solving. Next the REEP approach is described to explain the key concepts and how the design principles can be applied, with bakery scheduling and ROLLOUT providing examples. The penultimate section discusses the advantages of representations designed using the REEP approach and specifically addresses how these benefits can be used to tackle the challenges of complex problem solving. The final section is a concluding summary.

WHAT MAKES PROBLEMS COMPLEX?

First, there are sources that arise from the inherent nature of the domain in which problem solving is occurring. Second, the process of problem solving can itself be complex in various ways; and third, the individual doing the problem solving brings with him or her a set of issues related to the variability of knowledge and experience. Each of these issues is considered in turn while drawing on examples from our adopted bakery scheduling problem domain.

Complexity of Information and Concepts

Solving problems in a domain typically requires the problem solver to have declarative or conceptual knowledge of that domain. That knowledge may be complex in many ways that directly impacts on how it is possible to represent the domain.

A domain can be complex because it involves many different types of entities, each with numerous properties or attributes, and each of those in turn can possess many different possible values. Products and equipment are two obvious examples. The product attributes include the following: type (e.g., loaf, stick, bun, roll, bloomer, doughnut), dough or mix recipe (e.g., white, wholemeal, brown, whole grain), weight (e.g., 800 g, 400 g), decoration or coating (e.g., split, poppy seeds, flour dusted, “Tiger”), proof duration, baking duration, and so forth. Equipment attributes include the following: type (e.g., mixer, divider, prover, oven), configuration (e.g., rack oven, deck oven), capacity (e.g., maximum number of items of each product type), and process type (e.g., bulk versus piece-wise or conveyor-based).

At a more abstract level, potential complexity arises when many concepts are essential to understanding a domain. In bakery scheduling, many of these concepts are common notions found in all scheduling tasks, such as deadline, start time, process duration, delay, and capacity. Others have more specific interpretations. Bakery process stages are particular types of baking operations such as mixing, dividing, molding, proving, baking, and cooling. A process step is the occurrence of one process stage for one product. A batch is the production of a single product and a run is the production of a sequence of batches. An order is a group of products with a common deadline, perhaps destined for a single customer.

Large amounts of data are another source of complexity. A bakery may manufacture hundreds of products in a day. Each product will have multiple process steps and each step will have several parameters. Hence, many thousands of pieces of information are potentially relevant. To solve problems, the relations among such pieces of information must be considered, which brings into consideration combinatoric issues when dealing with all potentially relevant relations.

Another sense in which a domain may be complex is in relation to the different classes of concepts, or ontologies, that must be used to reason about a domain. Each of these perspectives there contain hierarchies of concepts and relations, but problem solving will require interrelations among the perspectives to be maintained. Hence, there is complexity due to the existence of tangled networks of concepts to be navigated and manipulated during problem solving. For example, in bakery scheduling, it is critical to relate temporal information (such as process durations, start times, and deadlines) with assignment information about what products are being processed in which equipment.

Conceptual complexity also arises when a domain involves different levels of abstraction and various levels of granularity, or scale. Problem solving is more difficult when one has to successfully apply general laws or principles to specific concrete cases, in addition to reasoning about just one level or the other. Similarly, when there are different scales at which things must be considered, this makes problem solving harder because things at different levels of granularity must be interrelated. In bakery scheduling, considerations range from the small scale, such as individual process steps for a single product, through to the large scale level comprising sequences of runs of multiple products.

There are many different manifestations of informational and conceptual complexity. The design of a representation for problem solving in a complex domain will need to support the amount, variety, and interrelatedness of the information and concepts of the domain. Deciding what information and knowledge to make explicit and how to coherently structure the presentation are major challenges of the designer of representations.

Complexity of the Task and the Solution Process

To solve problems in complex domains requires the means to process the information of the domain. Newell and Simon's (1972) classical account of well-structured problem solving provides an initial basis for considering the ways in which information processing may be complex. The established theory that problem solving is a process of heuristic search through a problem space identifies ways in which problem solving may vary in complexity. The elementary operators to transform one meaningful problem state to the next may be elaborate. Problem solving may be more complex if the average branching factor of the problem space is large, which in turn may be due to the availability of a wide range of operators or multiple ways in which a given set of operators can be applied to a state. Similarly, problem solving will be more complex when, on average, long sequences of operators are required to move from the initial to the goal state. In addition to the complexity due to breadth and depth of the problem space, complexity arises from the heterogeneity of the search, in terms of the tortuousness of the goal hierarchy needed for problem solving.

Many of the tasks of bakery scheduling have these characteristics of complex problems. Schedulers use many different criteria to assess the quality of a schedule, for example: the meeting of deadlines or matching of customer demand profile, the overall production time used, how efficiently equipment has been used, the number of changes of dough type or equipment settings, sufficient inclusion of staff breaks. The problem space is often broad and deep, because there are typically many options for when a product can be scheduled and many consequences follow from each option. For example, if an extra production run is to be inserted into an existing schedule, this can be done by displacing planned mixes. However, if any of the subsequent runs is near their deadline, they will in turn have to be brought forward, which in turn requires further rescheduling.

Moving beyond the classical theory of well-structured problem solving, there are other ways in which problem solving can be considered complex. Ill-structured problems (Simon, 1973) are more complex than well-structured problems, because some part of the definition or processing of the problem space is absent. For instance, the lack of a mechanistic means to test whether a goal state has been found means that it would be difficult to know whether problem solving has been successful and so can stop. Ill-structured problems will require a superordinate problem space to be invoked to fill in the absent aspects before a solution in the given problem space can be found (Simon, 1973). An example in bakery scheduling is knowing when a satisfactory schedule has been produced. This requires considerations at a higher level than the schedule itself and involves considerations of what criteria to use to make such judgments. Some problems cannot adequately be described in terms of the search of a single problem space, but are more coherently modeled as the search of multiple spaces, with the outcomes of the searches in each space mutually constraining the search in the other spaces (e.g., Klahr & Dunbar, 1988; Simon & Lea, 1974). At its most general, bakery scheduling may be considered as complementary searches of two spaces: (a) the space of possible sequences of runs assembled from similar products from different orders, and (b) the space of possible assignment of products to items of equipment at specific times. Decisions about what products can be put together into runs will be constrained by how much free capacity has been left from previous assignments, which in turn will depend on previously assembled and assigned runs.

One of the goals for the designer of representations is to mollify the impact of these forms of complexity. Different approaches may be taken and representations may be created that aim to support the process of searching the breadth and depth of the problem spaces, by providing additional representational tools that record and make explicit the problems state actually examined. When there are multiple problem spaces, the designer may provide automatic links between the information in multiple representations to aid mapping between the spaces (e.g., Cheng, 1999; Scaife & Rogers, 1996).

The Human Factor

Focusing on the people doing the problem solving brings a further set of complicating factors. The level of expertise of a problem solver will differentially impact on their ability to solve a problem in different circumstances, which in turn raises a further set of issues for the designer of representations to consider. Novices and experts will have different requirements from a representation because of the amount of knowledge and the way it is encoded in memory. In addition to being limited, the novices' knowledge will be more declarative and will require conscious effort to apply it to specific problem cases. In contrast, experts' knowledge will be broad and deep. Some of their knowledge is in the form of rich perceptual chunks and schemas, with associated actions, which allows the swift recognition of problem states and applications of operations to improve those states. Other knowledge may be in the form of sophisticated mental models that capture intricate structural constraints concerning aspects of the domain, that allow them to mentally simulate problem scenarios and to read-off and interpret consequences.

Novice bakery schedulers will require support in recognizing significant problem situations and need help in inferring what actions to perform and the potential consequences. Expert bakery schedulers have well-rehearsed rules for creating and revising schedules, but they also use heuristics to manage the complexity of the problem; for example, by reducing variability of processing times across products and by building in gaps between runs to serve as contingency buffers in case problems occur. This suggests that their mental models are rather crude and are not suited to detailed reasoning about the interactions between sequences of production runs.

Other sources of complexity related to the problem solvers themselves are negative transfer and psychological biases. Negative transfer occurs when knowledge from one area of expertise is incorrectly applied in a different problem-solving domain (VanLehn, 1989). An example is our everyday knowledge of scheduling that is typically concerned with the scheduling of events or travel, which is not applicable to the scheduling of conveyor-based processes in bakeries. Psychological or cognitive biases are revealed when people's reasoning about simple situations fail to match the ideal of logic and mathematics (e.g., Evans, 1989). From our studies of bakery schedulers, it appears that they prefer to schedule forward rather backward in time. To populate a schedule, the common strategy is to allocate batches from a given point working forward to the future and to see how much time is left before the deadline. Only in special cases is the strategy of allocating backward from the deadline used. This may be due to a temporal cognitive bias that favors the making of assignments forward in time, which is perhaps more natural than thinking about things ordered backward in time.

Novice–expert differences are sometimes considered in the design of interfaces and displays, with the presumption that simplified versions should be provided for novices. The issues of avoiding negative transfer and preventing cognitive biases are not well addressed by the literature on the design of representations. Creating an effective representation for the basic requirements of problem solving is usually such a challenge that these secondary issues are not tackled. Nevertheless, a complete approach to the design of representations needs to be able to deal with these aspects of user complexity.

Clearly there are many things that make the representation of complex problems a challenge, including informational and conceptual complexity, task and solution process complexity, and complexity arising from the nature of users. The next section introduces an approach to the design of representations that aims to address difficulties for problem solving that arise from these sources of complexity by designing effective representations.

REPRESENTATIONAL EPISTEMIC APPROACH TO REPRESENTATION DESIGN

The central idea of the REEP approach is to design representations that preserve the conceptual structure of the domain in the design of the structure of the representational system. It is claimed that representations created to directly encode the system of knowledge that underpins the problem-solving domain will provide a whole range of small- and large-scale benefits for cognition and complex problem solving. The benefits are discussed in the section following this one. This section presents a revision of the REEP approach, which clarifies and extends our previous theoretical formulation. Previously, the design guidelines were couched in terms of semantic transparency and syntactic plasticity characteristics, which were posed as desirable properties that a representation should possess (Cheng, 2002, 2003; Cheng et al., 2002). However, that account was problematic, because the definition of these characteristics lacked precision, and being characteristics, it was not clear how they could be fully operationalized for the process of design.

The revision of the REEP approach has, in part, been driven by the need to deal with the complexities of the bakery scheduling domain. The approach is now centered on the notion of conceptual dimensions. The elaboration of the nature of conceptual dimensions and the interrelations among them provides an analysis of the system of knowledge that is to be preserved by the design of the representation. In turn, the principles for designing a representation are specified in terms of intrarelations among parts of each conceptual dimension and in terms of interrelations across sets of representational dimensions. The idea of conceptual dimensions is first introduced. Then the analysis of conceptual structures in terms of conceptual dimensions is discussed. Finally, the design principles are presented and explained using the ROLLOUT diagram as an example.

Conceptual Dimensions

The conceptual structure of a problem domain should be analyzed in terms of its constituent conceptual dimensions. A conceptual dimension is a subdivision of a given system of knowledge and comprises similar types of ideas or notions that are taken from the same perspective within the domain. A conceptual dimension may be considered, in general terms, as an ontology that defines the ways of describing associated facets of a domain. A conceptual dimension may be heterogeneous, possessing different levels or aspects to which its concepts may belong. The nature of these levels and aspects will be specific to the conceptual dimension concerned.

From the studies of bakery scheduling (Cheng et al., 2006) and previous work, seven conceptual dimensions have been identified as general perspectives that have been important in characterizing the nature of knowledge in the domains studied. Table 5.1 lists them and identifies the levels or aspects belonging to each. Each is considered in turn.

The entity-taxonomic conceptual dimension concerns basic objects, things, and types of things that exist in the domain. Under this dimension, two levels are distinguished in terms of particular instances of objects and classes of objects of the same kind. For example, in the bakery scheduling domain, some of the main classes of entities include the following: types of bakery (e.g., plant versus in-store), types of products (bread or loaves, buns or rolls, biscuits), types of processing stage (mixing, dividing, proving, baking, cooling), types of equipment (e.g., mixer, deck oven, rack oven), and types of dough (white, brown, wholemeal, granary). Each type of object may consist of subtypes, in the form of a taxonomic hierarchy: for instance, a bakery will have different dough recipes for each type of dough, which will be used for different sets of product types.

TABLE 5.1

Conceptual Dimensions

Conceptual Dimension	*Levels–Aspects and Examples*
Entity-taxonomic	Existing things: entities, objects; Classifications: category, subcategory, group
Property	Measure: Nominal, ordinal, cardinal, interval, ratio; Type: intrinsic property, extrinsic property
Temporal	Perspectives: point (specific times, deadlines, start, end); interval (duration, delays, lead and lags); relational (before, after, tomorrow)
Structural	Aspects: spatiality (position, orientation); angularity (obtuse, orthogonal, acute); divisions (component, partition, department, region); arrangement (containment, alignment, abutment, alignment, framework, chain); connection (intersection, overlap, link, bridge, crossing); association (group, pair, central, peripheral); paths (line, branch, network, lattice); Levels: granularity scales (local, global)
Functional	Aspects: Process (assemble, [de]compose, partitioning, group, maximize, control, target, tune, randomize, interact, supply, demand); change (transitions, development, evolution); motion (traverse, flow, navigate, follow); organizing (arrange, distribute, assign, allocate, align); sorting (prioritize, rank, order); repetition (cycle, loop, iterate, recursing, hyseresis); copy (reproduce, inherit, duplicate); cause and effect (drive, force, affect, outcome, result); rules (conditions and actions); differentiate (options, contingencies, alternatives); variability (fluctuate, constancy, noisy); Levels: generality (concrete, abstract)
Formal relational models	Aspects: set theory (disjunction, conjunction, conditionality); arithmetic (addition, subtraction, division, multiplication); algebra (distributive, associative, commutative); differential calculus (rate of change, integration); other mathematical (correlation, specific functions); Levels: abstraction with variables; concrete with actual values
Evaluative	Aspects: evaluation (cost, benefit, efficiency, importance, salience); purposes (goals, focus, rewards, violation avoidance); constraints (requirements, limits, minima, constancy)

The property conceptual dimension concerns the observable and measurable attributes or properties of things in a domain. Concepts within this dimension can be classified in terms of ideas about the nature of quantities. One attribute of a type of entity is the number of actual entities (e.g., the number of batches in a production run). The properties of entities can be distinguished in terms of the nature of the scale on which it may be measured: nominal, ordinal, interval, and ratio (e.g., Ellis, 1968). The difference between types of dough is nominal. The sequence of process stages is ordinal. The duration of the production steps are interval measures of time. The amount of product and capacity of equipment are ratio quantities. Another distinction that may be useful to make under this dimension is whether a property is intrinsic or extrinsic (directly measurable or derived); for example, the mass versus density, respectively, of a loaf of bread. For the purpose of bakery scheduling, the quantity percentage capacity of equipment is a useful derived measure, as this allows considerations of the use of equipment without the need to deal with the actual physical sizes of products and equipment.

The temporal conceptual dimension includes various perspectives or measures of time. Time can be specified as points, intervals, or as relations (Shahar, 1997). Start and end times of batches or process steps, and also deadlines, are examples of time points. Examples of time intervals include production durations, lead or lag times between process steps, and spare time or delays with respect to deadlines. Time relations include notions such as before and after, or binary temporal relations, such as "produce white bread before wholemeal."

The structural conceptual dimension is concerned with the static form, shape, or organization of things in a domain. Concepts in this dimension include spatial, geometrical, anatomical, architectural, and topological notions. Table 5.1 identifies a sample of different aspects of this dimension (with some examples of concepts). For a given domain, these concepts may be applicable at different levels of granularity, ranging from overarching ideas about the organization of a domain through to low level, local configurations of entities. In terms of bakery scheduling, one of the major structural concepts is the organization of the process of baking into distinct process stages, typically mixing, proving, dividing, final proof, baking, and cooling. Some of these stages have substages. The composition of orders is also a structural concept as the products to be made are components of an order. The same general notion also applies to configurations of equipment, such as the simple subdivision of shelves in deck ovens. The lowest level considered in this conceptual dimension is, perhaps, the arrangement of individual items of product on trays.

The functional conceptual dimension concerns activities and processes. Concepts considered under this dimension include physiological, operational, behavioral, and other ideas relating to dynamic processes. The concepts can be considered as different levels of generality. Bakery scheduling at the most general level is concerned with the abstract notions of satisfying demands using given resources. The demands are quantities of products to be produced to certain deadlines using particular processing stages, each with certain physical and spatial requirements. Resources are the capacity of equipment and its temporal availability. At a more specific level, there will be considerations of concepts such as the sequencing of processing stages, the flow and continuity of production, the decomposition of production orders into runs, and how manufactured product is aggregated to fulfill orders. At an even lower level, there will be functional consideration of the process of distribution or alignment of products in relation to the available equipment capacity.

The formal relations conceptual dimension has the role of modeling aspects of the domain using formal systems such as logic, set theory, arithmetic, algebra, and so forth. The aspects modeled under this dimension may be specific relations that pertain to one fragment of a domain. Alternatively, the concepts may be fundamental relations of the domain, its underpinning relations, universal invariants, conserved higher order quantities, symmetries, axioms, or laws. When such fundamental relations exist, they are particularly important for the design of representations of the domain. There are similarities between aspects of the structural dimension and the formal relations dimension. An aspect of a domain will belong to the formal relational models dimension when it is necessary to consider the aspect precisely and in depth. For example, when simple collections of things are being considered and there are no complex relations (e.g., exclusive-or), then the structural or functional dimension is more appropriate for the target concepts. However, the precise classification of concepts under particular dimensions is less critical to the approach than to successfully identify concepts relevant to the domain.

In bakery scheduling, there are no fundamental laws, but set theoretic and arithmetic relations have important roles. Orders may be considered as nonintersecting sets comprised of disjoint collections of product. Similar notions apply with the packing of batches of product into a piece of equipment, but arithmetic relations govern the quantities under consideration; for example, *free capacity = total capacity – Σ individual batch size.* In many domains, the more specific cases are such obvious relations that they are not explicitly considered, but for the purpose of designing an effective representation they are important to address. For any formal relation it is necessary to consider different levels of abstraction. On the one hand, there is the concrete level, in which particular values are assigned to variables for specific cases (e.g., in terms of the previous equation: free capacity, 22% = 100% – [33% + 45%]). On the other hand, the abstract level concerns general relations expressed using variables, which are applicable to a wide range of pertinent cases.

Finally, the evaluative conceptual dimension encompasses concepts to do with the assessment or judgment of the domain in different ways, often in relation to particular purposes. Such evaluations may range from local assessment of particular parts of the problem domain to more global assessment of the domain as a whole. Local evaluations concern things such as proximity to limits or whether particular constraints have been violated. Combinations or more complex relations of local evaluation may constitute global evaluations, which assess the overall characteristics of a domain, using concepts such as efficiency or cost–benefit ratio. In terms of bakery scheduling, there are a multitude of local and global evaluative characteristics. Deadlines for specific orders, capacity limits of pieces of equipment, and no lags between process steps are just a few examples of evaluative concepts at a local level. At an intermediate level, evaluations may, for example, concern how efficiently a piece of equipment is used, or how easy production is with regard to the number of dough changes, or whether there are many severe peaks and toughs in staff work load. The overall performance of a bakery is typically considered in relation to how well production deadlines and quantities are satisfied.

These seven conceptual dimensions have been identified for the REEP approach. Not all of them will be applicable in every domain and their degree of relevance will depend on the particular knowledge system of a domain. No claim is made that this is an exclusive list, or that the dimensions are formally distinct categories of knowledge, such that every concept belongs uniquely to only one dimension. The notion of constraint is included under the evaluative dimension, but it may warrant being considered as a conceptual dimension in its own right. One might also argue that the temporal dimension is the application of the structural dimension to the domain of time, so it should be subsumed under the structural dimension. In the REEP approach, it is treated as a distinct dimension, because temporal concepts are often considered independently of other structural concepts and we have found the distinction to be useful in creating our previous work. The purpose of conceptual dimensions is to provide a sound basis for the analysis of the conceptual structure domain, which the next subsection now considers.

Conceptual Structure

The key idea of the REEP approach is to preserve the conceptual structure of the problem-solving domain in the representational structures of the interface. It is assumed that detailed descriptions of the target domain have been obtained by normal knowledge acquisition methods such as interviews, problem walk-through, verbal protocol analysis, task analysis, and so forth. These methods will have been applied across the range of environments and tasks that define the target problem-solving domain.

The conceptual dimensions allow the conceptual structure to be derived from the descriptions of a domain in a systematic fashion at two levels. The first level concerns the structure of relations within each of the conceptual dimensions, which deal with aspects and levels specific to each conceptual dimension. The previous subsection discussed aspects and levels for each conceptual dimension. Table 5.1 may be used as a list of queries to identify those aspects and levels present in each conceptual dimension. The example from the bakery scheduling domain shows the richness of conceptual relations that may occur within each of the individual conceptual dimensions of the domain.

The second level of a conceptual structure is the interrelation between conceptual dimensions. Some combination of dimensions will be primary, because they are central to a domain; others will be more peripheral secondary dimensions. Primary conceptual dimensions will possess the greatest number, most complex, and pervasive concepts in the domain, and they will most often be considered simultaneously with many other conceptual dimensions. Secondary dimensions will involve concepts that are considered in a relatively narrow spectrum of cases and often in relative isolation from other conceptual dimensions. Alternative conceptual dimensions will be primary in different types of domains. In previous work on instruction domains for science and mathematics, the formal relational models and property dimensions were particularly important as those domains were focused on promoting an understanding of the underpinning theoretical principles or laws of the domain (Cheng, 2002, 2003; Cheng & Shipstone, 2003). In other work on event and personnel scheduling, the temporal, structural, and functional dimension were the primary dimensions. (Barone & Cheng, 2004; Cheng at al., 2002).

Determining what are the most effective methods for distinguishing primary and secondary conceptual dimensions is the subject of current work. One approach is to select and analyze a representative sample of problem-solving protocols and analyze the frequency of occurrence of the aspects and levels of each dimension and the frequency of simultaneous consideration of particular combinations of conceptual dimensions. Another approach is to obtain a list of the most important domain concepts and examine which conceptual dimensions are implicated in the description of those concepts. The frequency of references to a concept in descriptions or protocols of problem solving, or the judgment of domain experts, can be used to assess the relative importance of the concepts.

Table 5.2 shows such an analysis for common bakery scheduling domain concepts. The entries in the cells of the table are the levels or aspects of the

TABLE 5.2

Analysis of Domain Concepts in Terms of Conceptual Dimensions

	Conceptual Dimension						
Domain Concept	Entity	Structural	Functional	Temporal	Property	Evaluative	Formal Relational
Product type	Thing Types	Association	Organize		Nominal		
Process stage	Thing Types	Path Association					
Equipment	Thing Types	Divisions Arrangement			Nominal Ratio	Constraint	
Order	Thing	Association		Point Interval Relation		Constraint Evaluation	Set theory Arithmetic
Dough	Thing Types		Rule		Nominal		
Batch	Thing		Organize Move	Point Interval Relation		Constraint Evaluation	
Run	Thing	Association	Organize Sort	Point Interval Relation			
Process step	Thing Types	Division	Organize Move	Point Interval Relation	Ratio Nominal		Arithmetic
Bulk process	Thing		Change	Point Interval			
Conveyor process	Thing	Divisions	Change Sorting Move	Point Interval Relation			

conceptual dimensions that are germane to the concept. Although the number of entries (in a column) for each conceptual dimension should not be taken too literally, it does indicate the importance of that dimension across the domain concepts. Similarly, the co-occurrence of entries (in a row) for each domain concept gives some indication of which conceptual dimensions often co-occur in considerations of the domain. Of the chosen domain concepts, all can be considered as distinct facets in the domain with many possessing subcategories.

For the bakery scheduling the entity-taxonomic, temporal, structural, and functional domains are the primary dimensions. Time provides a context for thinking about most of the concepts, with the majority being considered in terms of all three temporal perspectives. Product, process stage, and equipment are a-temporal concepts, because they are essentially invariant in time. The structural dimension is important, because scheduling must take into account the spatial, associative, and componential forms that permeate the baking process (stages), bakeries themselves (equipment), and orders. Not surprisingly, the functional dimension is a primary dimension as scheduling is an activity that involves notions of organizing and sorting products, batches, runs and process steps, in time and space. Further, the idea of a product moving through the bakery, or individual items through a conveyor process, gives additional weight to the relevance of functional dimension.

The property, formal relational, and evaluative dimensions are secondary dimensions as they have a minority role in common domain concepts. Given the importance of the evaluative dimension in the scheduling-related domains previously studied (Barone & Cheng, 2004; Cheng at al., 2002), it is perhaps unexpected that this dimension in not a primary one. There are two reasons for this. First, the bakery scheduling domain is more complex with respect to the greater range of things to be assigned and to which they can be assigned, so the relative importance of the role of evaluative dimension is reduced. Second, the number of different types of constraints and the entities to which they apply is less in the bakery scheduling domain, at least compared to examination timetabling. However, the clear implication is that the evaluative and other dimensions can take a subsidiary role in the design of a representation for the domain.

Principles of Representation Design

How can the knowledge of the conceptual dimension and conceptual structures of a domain be used to design an effective representation? In the REEP approach, the fundamental idea is to support the interrelation between meaningfully connected concepts while differentiating those that are unconnected. This must be done within and between the conceptual dimensions. Four principles are proposed for use as heuristics in the process of design to structure a representation so that it satisfies these requirements. The principles specify the manner in which a representation should be structured to preserve the conceptual structure; how representational schemes should be used to encode conceptual dimensions and their interrelations and intrarelations.

First, it is necessary to clarify the notion of representational schemes, or formats. These are particular techniques used to encode information by making a correspondence or mapping between a concept and some component or property of the representation. Engelhardt (2002), Bertin (1983), Zhang (1996), Burns and Hajdukiewicz (2004), Card et al. (1999), and others, provided common examples of how such mappings may be achieved using various representational techniques,. These include the following: location in metric horizontal space (x-axis) or vertical space (y-axis); spatial properties of graphical objects, such as shape, size, and orientation; visual properties of objects, such as color, shading, or outlining; regions or subdivisions of space, such as embedded panels, separate windows, and layers; alphanumeric labeling; and relations among objects, such as spatial association, aligning in space, superposition, containment, or linking.

The four representational design principles are as follows:

1. Global interpretive framework—A global interpretive framework should be devised that coherently interrelates all of the primary conceptual dimensions of a domain at the highest level in the representation. The aim is to provide an overarching interpretative framework that can be used to interpret, contextualize, and interrelate concepts across any combination of the primary conceptual dimensions. The selection and organization of representational dimensions should be such that they encode the overarching interrelations among the primary conceptual dimensions. A global interpretive framework is achieved when single expressions in the representation can stand for complex concepts that involve all of the primary conceptual dimensions.
2. Differentiate primary conceptual dimensions—The global interpretive framework should clearly differentiate the primary conceptual dimensions, so that identifying and thinking about concepts from each dimension can be done in isolation as required. This is done by selecting different representational schemes (or unique combinations of schemes) for each of the primary conceptual dimensions. This principle is satisfied when an expression for a concept from a single conceptual can be interpreted using only its interpretive scheme without reference to, or interference from, schemes from any other dimension.
3. Integrate secondary conceptual dimensions—The global representational framework should be augmented with representational schemes

for each secondary conceptual dimension in a manner that allows the relations with other relevant dimensions to be made. Expanding schemes for existing primary conceptual dimensions or adding new representational schemes that interlock with relevant schemes can be used to achieve this.

4. Coherent schemes for conceptual dimensions—For each conceptual dimension, a representational scheme should be provided that interrelates the different levels and aspects of that dimension but that also allows the levels and aspects to be clearly differentiated. The aim of this principle is to support the unambiguous interpretation and contextualization of concepts within a dimension. This is done by selecting a representational scheme (or a unique combination of schemes) whose structural characteristics match the structural characteristics of the conceptual dimension.

The four principles should all be applied together in the design of a representation. They were developed by generalizing over various representations that we have designed and successfully evaluated. They also echo previous theoretical claims about what constitutes an effective mapping between structures in the representing world and the represented world; in particular, claims about the importance of homomorphic and analogous representations (Barwise & Etchemendy, 1995; Kosslyn, 1989; Palmer, 1978; Sloman, 1985; Stenning & Oberlander, 1995; Zhang, 1996). One important difference is that the principles explicitly acknowledge the heterogeneity of conceptual systems and specify how conceptual structure within and between conceptual dimensions should be represented. Previous accounts may be interpreted as addressing relations within a single conceptual dimension.

Design of ROLLOUT

Figure 5.1 shows an example of the ROLLOUT for bakery scheduling. The design of ROLLOUT is described with reference to the four principles. The primary conceptual dimensions for the domain are the entity-taxonomic, temporal, structural, and functional dimensions. The individual representational schemes used for each are first presented before their roles in the global interpretive framework are discussed.

Figure 5.2 shows how different types of icons represent different classes of entities and how individual instances are identified using labels. Figure 5.2a shows a product and Figure 5.2b is an order that is comprised of a number of such products. Figure 5.2c represents a product allocated to the schedule, which we call a *batch*. Figures 5.2d and 5.2e are icons representing bulk and conveyor or piecewise processing steps. A staircase-like strip of

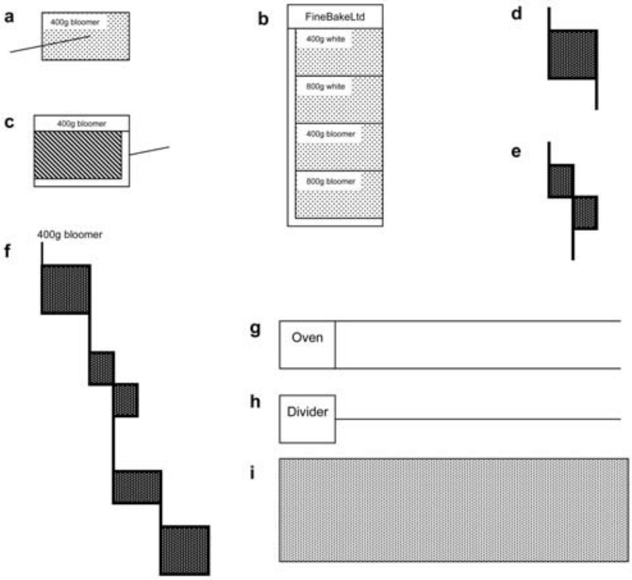

Figure 5.2. Types of icons for different classes of entities.

icons represents the production of a batch consisting of a set of process steps (see Fig. 5.2f). Figures 5.2g and 5.2h represent the two types of equipment, bulk equipment or conveyor equipment, respectively. Figure 5.2i is the generic representation of a process stage that could be performed by alternative pieces of equipment. The use of labeled icons for most entities in the bakery domain, with distinct types of icons for different types of entity, satisfies principle D that concerns the use of a coherent interpretive scheme for a conceptual dimension. An exception to this is the representation of types of dough, or mix recipes, which are shown by the color of product and process step icons. This particular scheme is used because each type of dough can make several different products; a one-to-many correspondence that cannot be adequately captured using more types of icons.

The temporal dimension is encoded by horizontal space (the x-axis) in ROLLOUT, with a linear timeline for reference running from left to right. Points in time are given by the location of graphical objects, as shown in Figure 5.3. Durations are distances between graphical objects or their parts. General temporal relations are encoded by the horizontal ordering of objects, with later things toward the right. This provides a coherent representational scheme for this conceptual dimension (principle D), because the different perspectives of time are encoded using the one representational scheme, but each perspective is differentiated by the use of distinct graphical elements (points versus distance versus order).

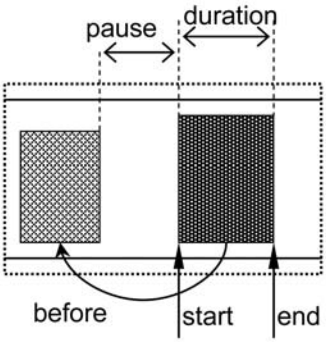

Figure 5.3. Aspects of the temporal conceptual dimension.

The structural conceptual dimension is encoded by the organization of graphical objects in vertical spatial dimension (y-axis). The representation of a group of products that makes up an order is achieved by arranging them together as a vertical stack, as shown in Figure 5.2b. Such groupings appear in the upper half of ROLLOUT (the planner diagram). The sequence of bakery processing stages is represented by the vertical sequence of process stage bars, their descending arrangement indicating their order in baking. The items of equipment that can be used for each process stage, which makes up the overall bakery configuration, are represented by equipment bars at the same level as the appropriate process stage bar. For example, Figure 5.4 shows the first two process stages in a bakery with one mixer and two dividers in the stages, respectively. These process stage and equipment configurations are placed in the bottom half of ROLLOUT (the scheduling diagram). The coherent interpretive scheme for each conceptual dimension principle (D) is satisfied by the use of positioning and grouping in vertical space, plus the differentiation of simple grouping (of products) and sequential configuration (of process stages and steps) into two separate panels (top and bottom).

The functional dimension concerns the dynamic activities of scheduling, which comprises three main aspects. First, there is the decomposition of orders and their collation into runs of the same type of product. This is achieved by associating together the icons for allocated batches, with lines connecting them to their respective orders, as shown in Figure 5.5. Second, the process of assigning a given process step to a particular piece of equipment is represented by the superposition of the process step icon on the bar for each piece of equipment. Figure 5.4 shows the result of assigning a batch to the mixer and then to the first two available dividers. Third, a product moves through the bakery as it is being processed and this sense of transition is captured by the descending chain of process step icons linked to-

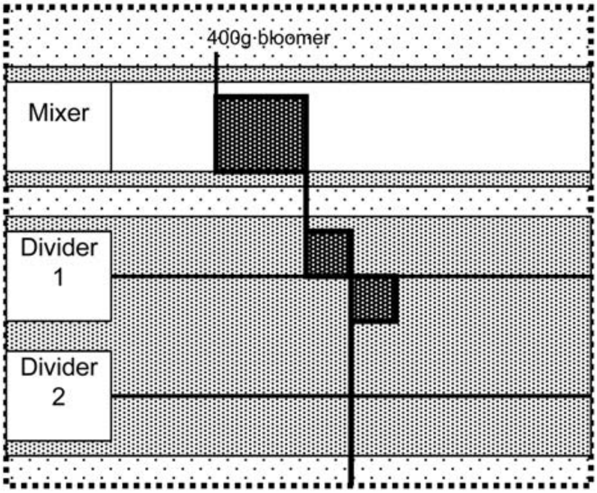

Figure 5.4. Structure of processing stages and assignment or processing steps.

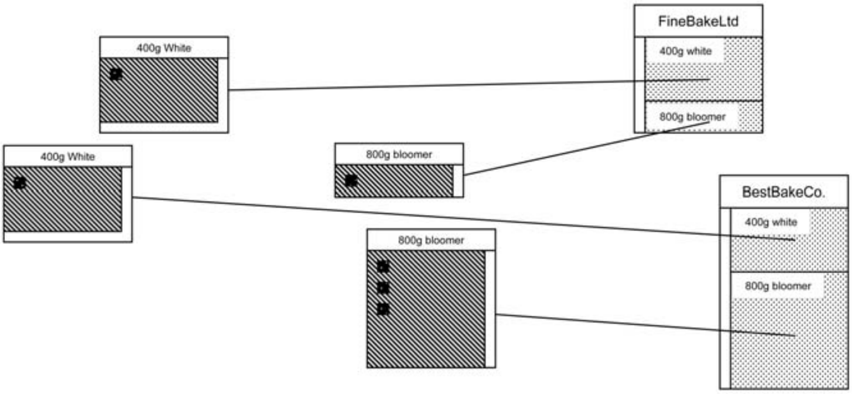

Figure 5.5. Decomposition of orders and assembly of runs: orders to the right and groupings of batches into run on the right.

gether by connecting lines, as shown in Figure 5.2f. The representational scheme for this conceptual dimension is that of relations among graphical objects and the use of different types of relations are used for the different aspects. Hence, principle D is satisfied for this dimension.

These are the pairings of conceptual dimensions and representational schemes: entity dimension to graphical object shape, temporal dimension to horizontal space, structural dimension to vertical space, function dimensions to relations among graphical objects. Hence, the design of the ROLLOUT diagram satisfies the principle B, which requires that each primary conceptual dimension be encoded using a different representational scheme. This means that changes in any one conceptual dimension are independent of the concepts in the other dimensions. For example, moving a process step icon to the right in ROLLOUT will give a delay to the start time, but its assignment to a process stage and a piece of equipment will remain the same (functional dimension), as will its processing order with respect to other steps (structural dimension). Similarly, functionally reassigning a process step from one piece of equipment to another within a process stage, for instance by moving the icon in Divider 1 to Divider 2 in Figure 5.4, does not change its timing or duration, or its structural relation to other process steps.

The ROLLOUT diagram satisfies principle A. The representational schemes work together to provide an integrated interpretive framework because they all share the same space. Figure 5.6 shows schematically how they are coordinated in the ROLLOUT diagram. Most domain concepts that span more than one conceptual dimension can usually be read as a sin-

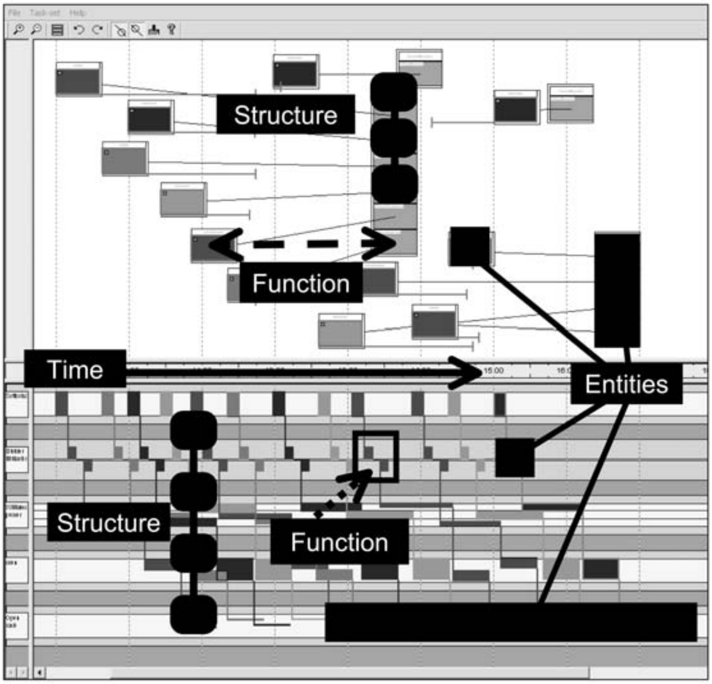

Figure 5.6. ROLLOUT global interpretive framework.

gle expression. For any given entity, information from each dimension is co-located with the icon representing the entity: its class is represented by the type of icon, temporal information by the icon's position and width, structural information by vertical position, functional information by its links to other icons or overlaying other icons. Further, the temporal dimension does not just apply to particular entities but also to relations between entities, such as the distance between two process steps standing for a pause in production in a piece of equipment, as shown in Figure 5.3. More complex relations across the dimension are also expressed. Consider Figure 5.5, which has two pairs of runs for similar products, with 400 g white loaves to the left and bloomers in the middle of the diagram. Their deadlines are given by the horizontal positions of the right-hand edges of their order icons and the start times for the allocated batches are given by the left sides of their icons. Hence, it is clear that the order of production of the white loaves is temporally consistent with the order deadlines, but not so for the bloomers; but this may not matter as the white loaves are being made ahead of time compared to the bloomers.

ROLLOUT is divided into two halves, with the planner panel (top) and the scheduling panel (bottom), which reflect the particular structural and functional aspects of the domain. The planner panel possesses the representational scheme for structural concepts dealing with groups of orders and products, and hence the functional scheme for representing the decomposition of orders and the assembly of runs is naturally part of that top panel. The scheduling panel possesses the schemes for structural concepts that deal with the configuration of process stages and bakery equipment, and hence the functional scheme for the assignment of process steps to equipment is naturally part of the bottom panel.

An important distinction in the domain is the difference between bulk process stages and conveyor stages, which process dough in a piece-wise fashion. Bulk process stages are represented by simple rectangular icons, which combine the entity (rectangular icon), temporal (width), and property dimensions (height—discussed later). The representation of conveyor stages additionally requires the combination of structural, functional, and more complex temporal concepts. Figure 5.7 shows the various components of a conveyor processing stage, including the distinct sequencing of items, the operations on those items, and the timings associated with each part of this processing step. This shows how, at a finer scale, the interpretive framework supports the expression of concepts that combine components from multiple conceptual dimensions.

The secondary conceptual dimensions are the property, formal relational models, and evaluative conceptual dimensions. The overall interpretive framework is augmented to encode these secondary conceptual dimensions to satisfy principle C. The main aspect of the property dimen-

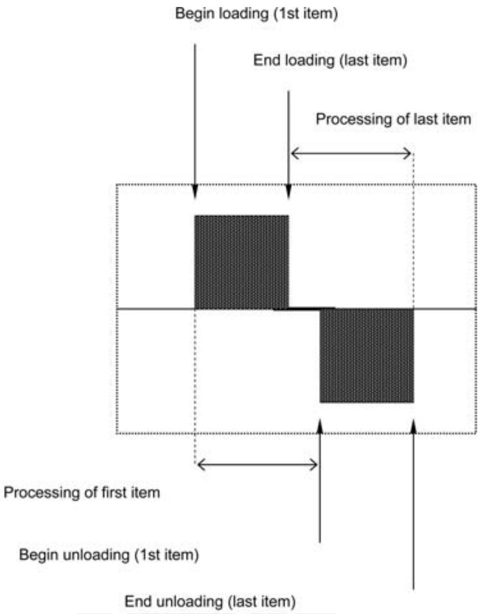

Figure 5.7. Interpretation of conveyor processing stage.

sion concerns the capacity of equipment and sizes of batches relative to the capacity of a particular piece of equipment. These quantities are represented by the height of the bars for the equipment and the height of process step rectangles, respectively, as shown in Figure 5.8. This method of representing (interval) quantities means the arithmetic rules—formal relation models dimension—that are used to compute capacity consumption (or spare capacity) are graphically encoded as the total height of a stack of process steps (or the unfilled space in an equipment bar). In the planner diagram, the height of the product icons in order groupings and the corresponding batch allocation icons, may also be used to represent the absolute size of batches (ratio quantities).

There are two aspects to the evaluative conceptual dimension: the general evaluation of schedule quality and the identification of violations of particular constraints. Particular graphical expressions in ROLLOUT can be used to reason about overall quality and to find specific violations. For instance, how effectively a piece of equipment is used can be visually assessed by inspecting how densely its bar is packed within process step icons. At a higher level, the quality of a schedule can be assessed by considering factors such as the packing density or uniformity of batches, or the sequence of different types of dough (block of icons of the same color). Similarly, particular graphical relations show violations. A missed deadline is shown by the end of a production strip in the schedule diagram being located to the right of its product grouping. A process step icon flowing over the boundaries of an equipment bar shows that the capacity has been exceeded, as shown in Figure 5.9a. Similarly, overlapping process step icons show clashes of product in a piece of equipment (see Fig. 5.9b). As violations of the constraints in a schedule correspond physically to impossible states of affairs or the breaking of important limits, these graphical relations are highlighted, augmenting the overall framework for a particular aspect of a conceptual dimension (principle C). In Figures 5.9a and 5.9b, the overflow and overlap are highlighted in red (black in the figure). For batches that miss deadlines, the whole process is highlighted by outlining all the process step icons and the links between them in red. Figure 5.9c shows one highlighted process step.

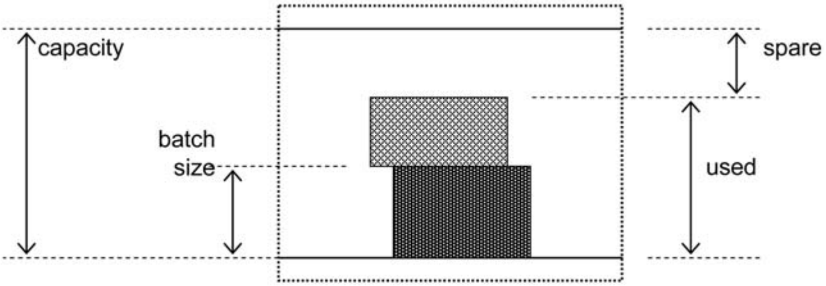

Figure 5.8. Representing property and formal relations conceptual dimensions.

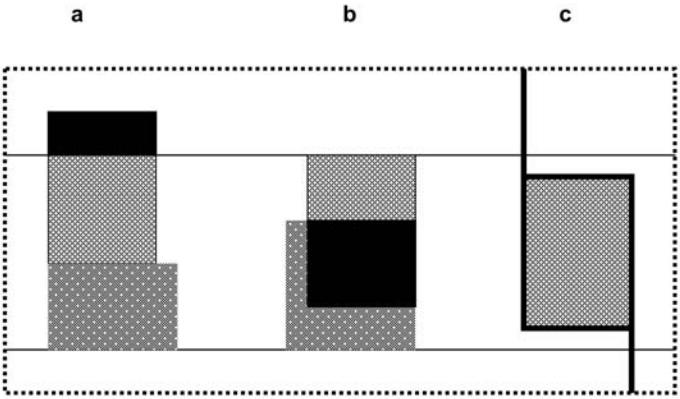

Figure 5.9. Various violations of constraints.

This section has shown how the REEP approach can be used to design representations that preserved the conceptual structure of a domain using appropriate representational structures. Conceptual dimensions are identified and used to explicate the conceptual structure of the domain. The design principles are then used to create a representation with a global interpretive framework, consisting of distinct and coherent conceptual schemes for each conceptual dimension. Such design is effortful, but the representations created may confer substantial benefits for complex problem solving and learning. These benefits are discussed in the next section.

BENEFITS OF REPRESENTATIONS DESIGNED USING REEP

The REEP approach has been successfully used to design representations for instructional domains and complex problem-solving domains. The latest is the ROLLOUT diagram, which has been shown to effectively support bakery planning and scheduling. Generalizing over these successful cases, various beneficial characteristics appear to be shared by them all: semantic and syntactic traits that enhance cognitive support at a number of different levels in such REEP representations. These characteristics are presented in order of increasing cognitive sophistication, and how they address the generic issues of complex problem solving, identified earlier, is now discussed.

Comprehension and Mental Models

The REEP approach seems to produce designs that support the comprehension of the problem domain and that help users to maintain good mental models of the problem. This appears to work in three interrelated ways (Cheng, 2002). First, the concepts tend to be represented by easily remembered and recognized patterns of graphical objects. Such patterns as graphical expressions may stand as composite icons for those concepts. Second, the encoding of concepts within and between conceptual dimensions using particular representational schemes and a global interpretive framework has the desired consequence that meaningful interpretive contexts are naturally provided for all graphical expressions. Third, the differentiation of the aspects and levels within conceptual dimensions, and the differentiation of conceptual dimensions themselves, further supports comprehension by reducing the likelihood of confusion between graphical expressions. Problem states are readily perceptible, which aids users in their acquisition of mental models of the problem environment, and also means that the cognitive cost of maintaining an accurate and detail mental model is relatively low.

An additional way in which representations designed using the REEP approach manage to support comprehension is through meaningful graphical expressions that happen naturally to emerge from the structure of the representation, which would not otherwise have been deliberately represented in a conventional design. Figure 5.10 shows such an example from ROLLOUT. It is necessary to identify the processing steps belonging to a given production batch. This is done with lines linking the bottom right and top left corners of successive steps, as in Figure 5.10a. These links can then be read as icons representing the transfer of the product between different process stages. The orientation of the links is an emergent property that encodes the information about the nature of the transfer. In Figure 5.10a, the transfer is immediate, whereas in Figure 5.10c, the downward slope means there is a delay. In Figure 5.10b, the upward slope represents a scheduling error, with a process step being set backward in time. These status, or warning, signals emerged from ROLLOUT without having been deliberately designed into the representations of the relations between process steps. When the basic semantics of process steps has been understood, users readily and correctly spot and then interpret the meaning of the slopes for themselves.

The REEP approach embraces the informational and conceptual complexity of a domain. The principles specify how such complexity can be addressed by using alternative representation schemes for different conceptual dimensions, but integrate the representations so that concepts and relations spanning multiple conceptual dimensions can be readily accessed. ROLLOUT and other REEP representations tend to be visually complex, as they bring together information from many conceptual dimensions within a single representation. However, users of the representations do not find this to be a particular difficulty, because they are able to selectively attend to those parts of the representations that are relevant to the particular goal and task context they have in mind, while ignoring other aspects of the representation that are irrelevant at the time. This form of se-

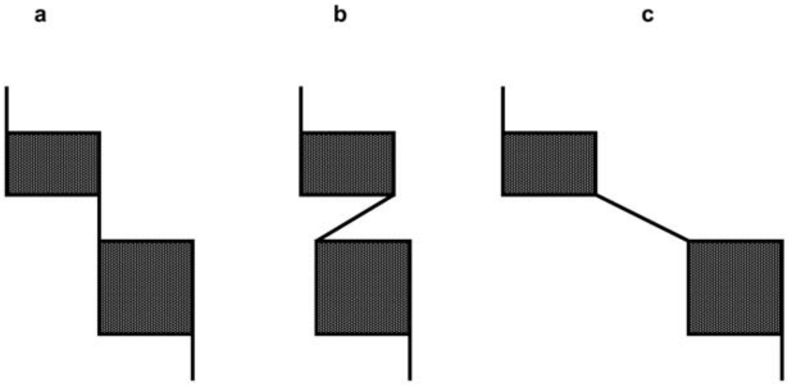

Figure 5.10. Transfer between process steps.

lective visual adaptation of the representation to current local problem requirements occurs easily without specific training and appears to be grounded in our natural ability to shift our focus of attention to different parts of complex environments depending upon task demands.

Supporting Problem Solving

REEP representations designed so far have all been diagrammatic. As such they accrue the well-understood benefits that most diagrams have when used for problem solving, such as reducing the effort needed for search and recognition (Larkin & Simon, 1987), and permitting the use of powerful schemas (Koedinger & Anderson, 1990). It also appears that REEP representations can address task and solution process complexities by providing more powerful problem operators and simpler solution procedures. A problem operator recognizes a problem state, or expression, and its action modifies that state in some way. The operators tend to be more powerful as more complex graphical expressions can be recognized and more complex transformations applied to them, because the expressions can encompass multiple aspects of a conceptual dimension or multiple conceptual dimensions. For example, to the bottom right in Figure 5.1, there is a cluster of three process steps that clash with each other. When attempting to resolve this, a user is not only aware of the clashes but also of the local spatial and temporal context, and also the broader context of the relations between the process steps upstream. The action to resolve the clash can be more sophisticated than just changing the start time of a process step, but can simultaneously involve moving a whole batch and an individual step within a piece of equipment. Problem-solving procedures are comprised of sequences of operator applications. REEP representations may improve these in various ways. The number of operators that are seen to be applicable to a problem state may be fewer and the selection of meaningful sequences of operators easier. For example, in ROLLOUT, all process steps can potentially be moved forward or backward in time, but the availability of the rich interpretive context means that operations that do not cause overcapacity clashes are readily apparent. An alternative way to describe these benefits is in terms of how accessible REEP representations make the many and varied constraints in a problem task. There are limitations or restrictions on the permissible relations under each conceptual dimension but these constraints, and how they interact, can often be read directly from the representation. More powerful operators, as mentioned earlier, may also mean a reduction in the number of operators needed in a procedure to achieve a goal. In other words, the branching factor and depth of problem state spaces given by REEP representations are likely to be smaller than for conventional representations (Cheng, 2002).

Problem-solving procedures are well supported by REEP representations, with users often adopting more sophisticated solution strategies than with conventional representations. One strategy involves looking ahead at the consequence of particular problem steps to plan sequences of actions that may temporally make the problem state worse, but that subsequently result in much greater improvements. Such strategies are more effective than the trial and error incremental approaches found with conventional representations (Barone & Cheng, 2004; Cheng at al., 2002). For example, ROLLOUT users may move the start time of a batch to one that is better but simultaneously create predictable clashes with other batches, which are then immediately resolved.

Complexity due to ill-structured problems and problems requiring the coordination of multiple problem state spaces may be alleviated by REEP representations. The provision of a coherent interpretive framework may allow the user to more easily jump from an ill-structured problem space to an overarching problem space and to find constraints to resolve the ill-structure parts for the first space. Different problem spaces may be associated with alternative combinations of conceptual dimensions, but as all such combinations are integrated within a single interpretive framework the coordinated search of the spaces is supported. For example, bakery scheduling requires the coordinated search of at least two spaces, the decomposing of orders into suitable runs and the allocation of batches to meet deadlines. The switching between the two spaces is facilitated by ROLLOUT as the planning and scheduling components are closely coordinated by the complementarity of planning and scheduling diagrams.

Studies on ROLLOUT and other REEP representations show that they have a dual character, allowing users to solve problems in knowledge-lean and knowledge-rich fashions, that are suited to users who are novices or experts in bakery scheduling, respectively. As the constraints of a domain are directly encoded in the representation, at one extreme, it is possible to solve problems as purely abstract geometry puzzles, in which graphical objects are arranged into diagrammatic configurations that satisfy given graphical constraints (which capture the requirements of the goal). At the other extreme, problems can be solved by knowledgeable users who only interpret patterns in the representation as meaningful states of the domain and reason just at that level. More typically, users do a mixture of both forms of reasoning, with their preference depending on their knowledge of the domain. This inherent flexibility means the complexity of problem solving due to the individual differences of the knowledge of users is naturally addressed.

Enhancing Learning

Some of the ways that REEP representations can effectively support learning build directly on the problem-solving benefits. As problem solving is easier, more episodes of problem solution will be experienced for a given amount of practice time, and more of the solutions found will be correct (Cheng, 2002). Also, fewer classes of operators and shorter, less complex solution paths mean that learning problem-solving procedures will be easier.

The representations can better support the development of expertise because they, in effect, capture the knowledge of the domain at multiple levels, which are then readily accessible as and when the learner is ready for them. As a learner acquires an understanding of the meaning of patterns of graphical objects, which are perceptual chunks for particular concepts, relations among them will be considered at higher level and the learner will naturally begin to acquire more complex conceptual relations. A possible hypothesis is that REEP representations provide learners with a self-adapting Vygotskian Zone of Proximal Development (e.g., Wood, 1988), which constantly changes as the learners becomes more knowledgeable. This parallels, at a higher level, the manner in which users appear to selectively adapt the representation to their current task goals and context. Possessing large meaningful patterns for use in problem solving is one of the characteristics of expertise (Chi, Glaser, & Farr, 1989), which the acquisition of REEP representations may be able bootstrap.

Supporting System Development

The development of ROLLOUT progressed through several cycles of prototyping, evaluation, and feedback from the users. Through the cycles, it was found that the underlying interpretive framework provided a rational basis for incorporating new developments. For example, bakery production was initially conceptualized as a sequence of bulk processes, but it became apparent that continuous, piece-by-piece processes, were just as important. Fortunately, the overall interpretive scheme provided appropriate constraints to devise a specific representation that incorporated the temporal, structural, and functional complexities of such conveyor processes. Figure 5.7 shows the representation, where the upper and lower rectangles represent the incremental loading and unloading of the equipment, respectively, and the offset between them represents the duration of the process.

Another way in which the REEP approach supports system development is through the emergence of useful novel concepts and relations, that were previously unavailable to the users. For example, in the ROLLOUT diagram, the horizontal distance between the points corresponding to the end of the production of a batch and the deadline of its order indicates the amount of spare time in production; something that was not readily available to the bakery schedulers. They considered this information to be so useful that they requested us to provide an even easier way of comprehending the spare time. This was achieved by adding an indicator bar to the icon for allocated batches in the planner window, as shown in Figure 5.11. The length of the line from the start to the bar (head of the rotated "T") gives the spare time. If the whole of the batch icon is to the right of the ideal start bar, then the batch will breach the deadline.

CONCLUSION

Designing representations for complex problems is important, because most real-world problems are complex. Research on the nature and design of representations has shied away from complex problems, because of the major challenges that complex problem pose. This chapter described some of these challenges and then presented an approach to the design of representations for such problems. The REEP approach contends that the conceptual structure of a domain and its underpinning system of knowledge should be preserved in the structure of the representation using an appropriate interpretive framework and representational schemes. The notion of conceptual dimensions was introduced as a basis for analyzing the conceptual structure of a domain. Seven such dimensions were described and applied to the bakery scheduling domain. The result was the ROLLOUT diagram, which effectively supports complex problem solving in that domain. This in turn adds further evidence to the claim that the REEP approach is an effective method for creating representations for complex problems.

ACKNOWLEDGMENTS

This research was support by the UK ESRC/EPSRC/DTI through PACCIT research program (RES-328–25-001). Our thanks go to present and past

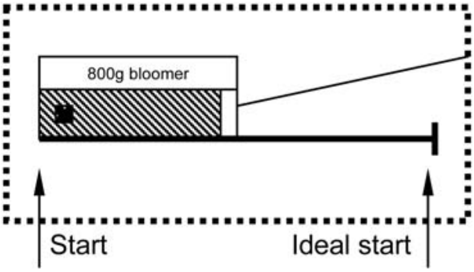

Figure 5.11. Allocated batch icon augmented with ideal start time bar.

members of the project team: Mark Auckland, John Wilson, Nikoleta Pappa, and Bill Dawson. We would also like to express our gratitude to the members of the ROLLOUT consortium for sharing their knowledge and insights into the fascinating world of the bakery industry and for their support of project, in particular Linda Young and Stan Cauvain of the CCFRA and BakeTran. Finally, we thank Jo Cheng for proof reading the manuscript.

REFERENCES

Barone, R., & Cheng, P. C.-H. (2004). Representations for problem solving: On the benefits of integrated structure. In E. Banissi, K. Börner, C. Chen, M. Dastbaz, G. Clapwortht, F. A., E. Izquierdo, C. Maple, J. Roberts, C. Moore, A. Ursyn, & J. J. Zhang (Eds.), *Proceedings of the 8th international conference on information visualisation* (pp. 575–580). Los Alamitos, CA: IEEE.

Barwise, J., & Etchemendy, J. (1995). Heterogenous logic. In J. Glasgow, N. H. Narayanan, & B. Chandrasekaran (Eds.), *Diagrammatic reasoning: Cognitive and computational perspectives* (pp. 211–234). Menlo Park, CA: AAAI Press.

Bertin, J. (1983). *Seminology of graphics: Diagrams, networks, maps.* XXXX: University of Wisconsin Press.

Burns, C. M., & Hajdukiewicz, J. R. (2004). *Ecological interface design.* Boca Raton, FL: CRC Press.

Card, S., MacKinlay, J., & Shneiderman, B. (Eds.). (1999). *Readings in information visualization: Using vision to think.* Mahwah, NJ: Morgan Kaufmann.

Cheng, P. C.-H. (1999). Interactive law encoding diagrams for learning and instruction. *Learning and Instruction, 9,* 309–326.

Cheng, P. C.-H. (2002). Electrifying diagrams for learning: Principles for effective representational systems. *Cognitive Science, 26,* 685–736.

Cheng, P. C. H. (2003). Diagrammatic re-codification of probability theory: A representational epistemological study. In R. Alterman & D. Kirsh (Eds.), *Proceedings of the twenty fifth annual conference of the cognitive science society* (pp. 234–239). Boston: Cognitive Science Society.

Cheng, P. C.-H. (2004). Why diagrams are (sometimes) six times easier than words: Benefits beyond locational indexing. In A. Blackwell, K. Marriot, & A. Shimojima (Eds.), *Diagrammatic representation and inference: Third international conference, diagrams 2004* (pp. 242–254). Berlin, Germany: Springer-Verlag.

Cheng, P. C.-H., Barone, R., Cowling, P. I., & Ahmadi, S. (2002). Opening the information bottleneck in complex scheduling problems with a novel representation: Stark diagrams. In M. Hegarty, B. Meyer, & N. H. Narayanan (Eds.), *Diagrammatic representations and inference: Second international conference, diagrams 2002* (pp. 264–278). Berlin, Germany: Springer-Verlag.

Cheng, P. C.-H., Barone, R., Pappa, N., Wilson, J. R., Cauvain, S. P., & Young, L. S. (2006). Understanding bakery scheduling: Diverse methods for convergent constraints in uder-centred design. In P. D. Bust (Ed.), *Contemporary ergonomics* (pp. XXX–XXX). London: Taylor & Francis.

Cheng, P. C.-H., & Shipstone, D. M. (2003). Supporting learning and promoting conceptual change with box and avow diagrams. Part 1: Representational design and instructional approaches. *International Journal of Science Education, 25,* 193–204.

Cheng, P. C.-H., & Simon, H. A. (1995). Scientific discovery and creative reasoning with diagrams. In S. Smith, T. Ward, & R. Finke (Eds.), *The creative cognition approach* (pp. 205–228). Cambridge, MA: MIT Press.

Chi, M. T. H., Glaser, R., & Farr, M. J. (Eds.). (1988). *The nature of expertise.* Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Ellis, B. (1968). *Basic concepts of measurement.* Cambridge, England: Cambridge University Press.

Engelhardt, J. (2002). *The language of graphics.* Amsterdam: ILLC, University of Amsterdam.

Evans, J. (1989). *Bias in human reasoning: Causes and consequences.* Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Glasgow, J., Narayanan, N. H., & Chandrasekaran, B. (Eds.). (1995). *Diagrammatic reasoning: Cognitive and computational perspectives.* Menlo Park, CA: AAAI Press.

Green, T. R. G. (1989). Cognitive dimensions of notations. In A. Sutcliffe & L. Maclaulay (Eds.), *People and computers v.* Cambridge, England: Cambridge University Press.

Kaplan, C. A., & Simon, H. A. (1990). In search of insight. *Cognitive Psychology, 22,* X.

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science, 12,* 1–48.

Koedinger, K. R., & Anderson, J. R. (1990). Abstract planning and perceptual chunks: Elements of expertise in geometry. *Cognitive Science, 14,* 511–550.

Kosslyn, S. M. (1989). Understanding charts and graphs. *Applied Cognitive Psychology, 3,* 185–226.

Kotovsky, K., Hayes, J. R., & Simon, H. A. (1985). Why are some problems hard? *Cognitive Psychology, 17,* 248–294.

Larkin, J. H., & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science, 11,* 65–99.

Newell, A., & Simon, H. A. (1972). *Human problem solving.* Englewood Cliffs, NJ: Prentice Hall.

Palmer, S. E. (1978). Fundamental aspects of cognitive representation. In E. Rosch & B. B. Lloyd (Eds.), *Cognition and categorization.* (pp. 259–303). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Pinker, S. (1990). A theory of graph comprehension. In R. Freedle (Ed.), *Artificial intelligence and the future of testing* (pp. 73–126). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Scaife, M., & Rogers, Y. (1996). External cognition: How do graphical representations work? *International Journal of Human–Computer Studies, 45,* 185–213.

Shahar, Y. (1997). A framework for knowledge-based temporal abstraction. *Artificial Intelligence, 90,* 79–133.

Sloman, A. (1985). Why we need many knowledge representation formalisms. In M. Bramer (Ed.), *Research and developments in expert systems* (pp. XXX–XXX). Cambridge, England: Cambridge University Press.

Simon, H. A. (1973). The structure of ill-structured problems. *Artificial Intelligence, 4,* 181–201.

Simon, H. A., & Lea, G. (1974). Problem solving and rule induction: A unified view. In L. W. Gregg (Ed.), *Knowledge and cognition* (pp. 105–127). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Stenning, K., & Oberlander, J. (1995). A cognitive theory of graphical and linguistic reasoning: Logic and implementation. *Cognitive Science, 19,* 97–140.

Ware, C. (1999). *Information visualisation: Perception for design.* San Francisco: Kaufmann.

Wood, D. J. (1988). *How children think and learn.* Oxford, England: Blackwell.

VanLehn, K. (1989). Problem solving and cognitive skill acquisition. In M. Posner (Ed.), *Foundations of cognitive science* (pp. 527—580). Cambridge, MA.: Bradford/MIT Press.

Zhang, J. (1996). A representational analysis of relational information displays. *International Journal of Human–Computer Studies, 45,* 59–74.

Zhang, J. (1997). The nature of external representations in problem solving. *Cognitive Science, 21,* 179–217.

Zhang, J., & Norman, D. A. (1994). A representational analysis of numeration systems. *Cognition, 57,* 271–295.