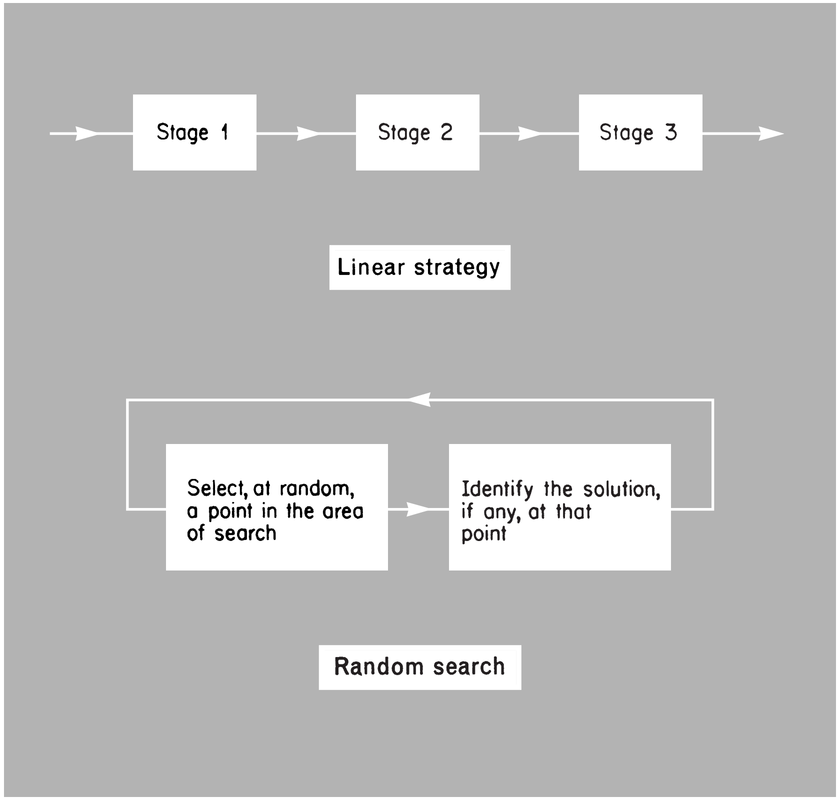

Figure 1.1 Examples of "block diagrams" from Jones 1970.

1. The masterplan

No, Watson, this was not done by accident, but by design.

1.1 Design methods

When the scientific study of design emerged after World War II, it began as an effort toward developing new procedures for designing. In the face of the increasingly complex tasks that designers were encountering, the pioneers of the field saw a need for improved ways of designing, as they thought the existing procedures were inadequate (Alexander 1964, 1971, Cross 1984, Jones 1970, Rittel 1972). Therefore, the early work almost exclusively sought to develop such new procedures, or *design methods*; and so, the field was appropriately called *design methodology*—the study of such methods. It was also known as "the design methods movement" (Cross 1984).

A design method is a normative scheme that specifies in detail a certain working procedure, the activities to perform, and also a specific order in which the activities should be carried out. It is usually very precise, and the designer is to follow it meticulously. It also covers the design process from beginning to end.

But the easiest way of describing design methods is through the boxes-and-arrows diagrams that always come with them (figure 1.1). The boxes and arrows are always there; it is the labels on the boxes and the connections between them that distinguish one method from another (Jones 1970, p. 61):

> Perhaps the most characteristic feature of the literature on design methods is the prevalence of block diagrams, matrices and networks of many kinds that resemble, to varying degrees, the diagrams and calculations that computer programmers use.

In the history of design methodology, there are two original works that tend to stand out from the rest. They are Alexander's *Notes on the Synthesis of Form* (1964) and *Design Methods* by Jones (1970). Together they epitomize the movement, for a number of reasons. First, they were both rather early and very influential. Earlier versions of the central ideas in both these books were presented at the first conference on design methods in 1962 (Alexander 1963, Jones 1963, Jones

& Thornley 1963). Secondly, even though these two texts and other pioneering work in the field have been fundamentally revaluated since, not least by these two authors themselves, these two works stand out in retrospect as the best exemplars and strongest representatives of the design methods era. Although the ideas in them may have come of age, and other works have long since faded into history, these two have been able to hold their own also as texts, because of the clarity with which they express the central ideas of their original context. Thirdly, because of their scope and depth, and because of the general coherence of the field, these two works together are sufficient to cover the central ideas of design methodology. Jones' book is also very much a compilation of other early work on design methods, ending with a catalogue of various design techniques that had been developed.

An additional work also deserves to be mentioned here. The edited volume *Developments in Design Methodology* (Cross 1984) manages to collect many of the most important papers from the first twenty years of the field, with good introductions to each section. It is therefore an invaluable source. The two previous monographs together with this compilation of classic papers leave little more to wish for as a comprehensive overview of the field.

Four unifying principles

The number of design methods (and accompanying diagrams) that have been published is immense. Probably no two authors have ever agreed on a method, so at least as many methods have been presented as there have been authors. But as people change their minds, the number is probably higher. Therefore, if you begin to review the field and the various methods, you quickly become bewildered by the plethora of variants, the different labels on the various boxes, and the directions of the arrows.

But when you examine a large enough number of variants, patterns begin to form: certain features are due to the specific content of a domain; architecture is different from information design, and so the methods differ. In many cases, different labels disguise the same ideas; and different authors emphasize different aspect of design, so the methods focus on different aspects of the design process. Other variation comes from whether a method is an entirely theoretical construction, or if it has actually been confronted with real design projects, and so forth. (Regarding comparisons between different methods, and their resemblances, see e.g. Jones 1970, p. 24, Lawson 1980, pp. 23–29.) To attempt a comprehensive survey here would thus also be a futile endeavor.

Rather, I will stick to a few prototypical models, and instead concentrate on the common patterns. This is a more viable route, since as much variation as there is on the surface, there is also an Aristotelian essence that the methods share, because they differ in their details rather than in fundamental concepts. To make this essence explicit, I will characterize it in terms of four fundamental principles, which are of particular interest from a cognitive point of view:

1. *separation*: The separation of the design process into distinct phases, with each individual activity being performed in isolation from the others.
2. *logical order*: The specification of an explicit order in which to perform these different activities.
3. *planning*: The pre-specification of an order in which to perform the activities within a phase.
4. *product–process symmetry*: The plan being organized so as to make the structure of the design process reflect the structure of the sub-components of the resulting design product.

These principles do not appear in any design methodologist's lexicon, but they make up the heart of design methods thinking, and give the various methods their family resemblance.

Separation

Out of the four principles, each consecutive one is an elaboration of those before it, drawing out their consequences and filling in their details. From this it follows that they are ordered, from the first being the most general and most fundamental one, to successively becoming more explicit and detailed. Although it may seem abstract and inconspicuous, *separation* is the most important principle, from which the remaining three follow as consequences. The most important separation is to divide the design process into three major phases: analyzing the problem, synthesizing a solution, and evaluating the outcome (Jones 1970, p. 63):

> One of the simplest and most common observations about designing, and one upon which many writers agree, is that it in-

cludes the three essential stages of analysis, synthesis and evaluation. These can be described in simple words as "breaking the problem into pieces", "putting the pieces together in a new way" and "testing to discover the consequences of putting the new arrangement into practice".

In this chapter, it is the separation of analysis and synthesis that is the most important one. It is the foundation of all design methods, and may well be the most consequential idea of design methodology as a whole. As Jones also indicates, this division was also widely accepted by design methodologists as a basic model of the design process (cf. Cross 1984). Design methods assign such a trivial role to evaluation that it becomes of marginal interest. As Jones here describes evaluation, for example, it seems to be called in only when the real job has already been completed.

Design methods normally make additional separations. In particular, the three major stages are often divided further into several smaller sub-activities.

The principle of separation says that different *functions* of the design process are performed as separate *activities*. With respect to analysis and synthesis, one can say that design activity must serve two functions: understanding the problem and producing a solution. Separation then means that each of these two functions is worked on in a separate phase of problem solving. It is for instance easy to imagine a situation where both of these aspects are worked on together.

Logical order

The second principle concerns the imposition of an *order* among the activities of a design method. Perhaps the distinction between the different activities that a design method is made up of may seem obvious, and the prescribed ordering among the activities may seem more significant. However, even though it might appear so, the working order is a necessity that follows directly from separation, whereas it is not obvious that they should be kept separated: If you do separate analysis from synthesis, then you must perform the analysis before the synthesis, as you have to have to understand the problem before you produce the solution. The same goes for evaluation, it requires that you have something to evaluate, and so must follow synthesis. And conversely, if you do not separate the process into distinct phases then there is nothing to order, so an ordering doesn't

make sense. This applies to all other separations that are made: the ordering among the activities is a logical consequence of the purpose that each serves. It is therefore the *logical order*.

Taken together, the first two principles, separation and logical order, generate a basic three-stage model of design; cf. figures 1.2 & 1.4.

Figure 1.2 The basic three-stage design method schema.

Planning

Whereas the logical order concerns the relation between different phases, the third principle aims to lay down the organization of the design activities in even greater detail, to include the activity *within* a phase. Because of the size and complexity of design problems, each of the three major phases is quite complex. Without an internal order, each phase would be a large, unstructured activity, left by the methodologist for the eventual designer to decide. *Planning* consists in setting up a strategy, a plan, for how a particular activity should be performed. The prototypical case is when a plan is set up as the final part of the analysis, and the course of action in the synthesis is thereby laid down before this activity begins.

Product–process symmetry

The fourth principle concerns the decomposition scheme used in the plan; the particular strategy that organizes activity inside the synthesis phase. There is not automatically any *logical* ordering within the phases. Therefore, a decomposition strategy needs to be chosen. This strategy could be *ad hoc*, but typically design methods try to do better than that.

There is however one strategy that is particularly obvious. This is the idea of using the division of the *product* into subcomponents for the decomposition of the *activity* as well: As also the design solution is bound to be complex, it too ought to be broken down into manageable parts. Hence, part of the analysis typically consists in finding such a suitable solution decomposition, usually a hierarchical one. And when you have this decomposition, it is not far-fetched to use it to structure the synthesis activity as well. In effect, the synthesis phase gets a hierarchical organization that mirrors the hierarchical structure of the final product. Hence the process and product are structured in the same way; the decompositon principle consists in a *product–process symmetry*. This lies particularly close at hand since the symmetry results in a natural one-to-one mapping between different parts of the synthesis and of the design product.

All four principles taken together yield a resulting schema that is more complex than the basic three-stage version. As the last two principles are elaborations of the first and second, the complex schema can be regarded as an "elaborated" version of the basic one.

Examples of the elaborated version are the classical "waterfall" model (Boehm 1975, cf. figure 1.3) from software engineering (also cf. Adelson & Soloway 1988, Jeffries *et al.* 1981, Parnas & Clements 1986), and Alexander's (1964) method, which centers on a technique for determining a suitable problem decomposition. These are known as "structured design methods": analysis creates the decomposition structure of the artifact, and which the synthesis is to follow as a "structured decomposition". Together, the basic and elaborated versions capture the central features of most design methods.

To make relations between different design methods stand out more clearly, such as between these basic and elaborated versions, I will hereafter use a "timeline" format which does not obscure these relations, as does the clutter in diagrams of the boxes-and-arrows kind (see figure 1.4).

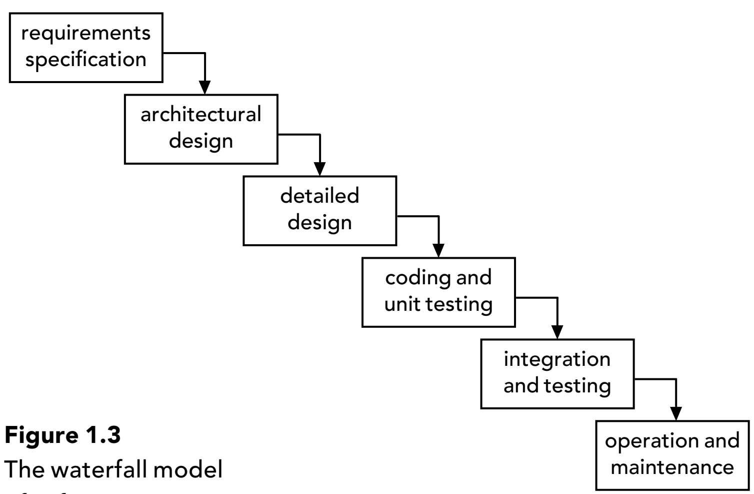

Figure 1.3
The waterfall model of software engineering.

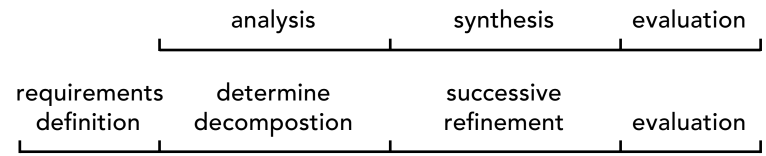

Figure 1.4 Timelines of the basic and elaborated design methods.

However, even with the timelines one problem remains: Different stage models can only be compared approximately, since the internal boundaries between their stages are inherently vague. For example, is requirements definition part of the analysis? Both yes and no are correct answers—mainly because they cannot be held apart in practice. In fact, the greatest weakness of stage models is the principle of separation, as I will argue in chapter 3. It is because such sharp divisions cannot be maintained that the separation of activities breaks down in practice. And for the same reason, one cannot strictly say what goes where within a model, and thus what it corresponds to in another model. For example, analysis in one model sometimes includes the "understand problem" phase of another, while it does not in a third one. Therefore some relations in the timeline diagrams will seem inconsistent.

1.2 The origin of design methods

Why do the design methods look like this? Where do they come from? What is the origin of these methods, the origin of the pattern that is reflected in the four principles? Compared to the motives behind the rise of the design methods movement, the authors of the field have been much less clear about the background of the methods themselves. Rittel (1972) mentions "the ways in which the large-scale NASA and military-type technological problems had been approached" as a major source of inspiration, and also elsewhere there is the occasional reference to general systems theory and operations research. But this does not lead to the answer we are looking for.

Alexander gives some minor, indirect clues when asked about the origins of his (1964) method (Alexander 1971):

> As you know, I studied mathematics for a long time. What I learned, among other things, was that if you want to specify something precisely, the only way to specify it and be sure that you

aren't kidding yourself is to specify a clearly defined step-by-step process which anyone can carry out, for constructing the thing you are trying to specify. In short, if you really understand what a fine piece of architecture is—really, thoroughly understand it—you will be able to specify a step-by-step process which will always lead to the creation of such a thing. ... So for me, the definition of a process, or a method, was just a way of being precise, a way of being sure I wasn't just waffling.

The step-by-step processes in mathematics that Alexander is referring to are formalized methods for mathematical proofs. It is however Parnas & Clements (1986) who give the most explicit clues—if not any lead to the source in itself:

> Ideally, we would like to derive our programs from a statement of requirements in the same sense that theorems are derived from axioms in a published proof. (p. 251)

Parnas & Clements do not make a big point out of this, or indicate any specific relation to mathematical proofs; these seem to have served more as a source of inspiration to design methodologists, because of their desire to have the same kind of solid foundations for their design choices as mathematicians have in their proofs. Neither did Alexander explicitly model his method after any specific mathematical procedure. His background in the field would rather have provided the "tools" to realize his method.

Hence, the origins of design methods are not well documented. Still, no introductory chapter with any pretensions should be without a reference to the ancient Greeks, and this is where the opportunity arises. Some shallow digging into the origins of logic and mathematical proofs shows a historical influence on design methods that is quite old, but nevertheless still clearly present in design methodology today: contemporary design methods have their roots in the pattern of classical Euclidean geometry proofs. What is more, it is somewhat surprising how many quite different roads in this chapter will be found to all lead back to this same Rome in the end.

Pappus

Or to be more correct, back to Alexandria, as the text in question was written by the Greek mathematician Pappus of Alexandria, probably around AD 300. (The original reference is the Latin translation in Hultsch 1876–77 vol. II, pp. 634–636, first English translation in

Heath 1921, see also Polya 1945 and Hintikka & Remes 1974.) In the seventh book of his *Collectio*, Pappus describes what he calls the *analyomenos*, which has been variously translated as "the Treasure of Analysis", "the art of solving problems" and "heuristics". The former is the conventional translation, the latter two are from Polya.

Pappus' text describes a method of analysis and synthesis, to be used for producing geometrical proofs such as those found in Euclid's *Elements.* The crucial elements of mathematical proofs and problem-solving procedures that are presented there stand essentially unaltered to this day. But what is more, the text also contains all the central ideas of design methods; in fact, these amount to the same four principles as those introduced above. The links between design methods, mathematical proofs, and Pappus' original account are remarkably strong, given the vast span of time between them. I will here include an extensive quotation of the original text for two reasons: first, because it has a central place in the argument that follows, and secondly to give enough material to show that the points are valid here even though they have been moved far from the original and very old context (The translation is taken from Hintikka & Remes 1974, pp. 8–9, my italics, and I will depart from the convention and leave out the original text in Greek):

> The so-called Treasury of Analysis is, in short, a special body of doctrines furnished for the use of those who, after going through the usual elements, wish to obtain the power of solving theoretical problems, which are set to them, and for this purpose only is it useful. It is the work of three men, Euclid the author of the *Elements*, Apollonius of Perga, and Aristaeus the Elder, and proceeds by the method of analysis and synthesis.

Analysis traces a path backward from the goal ("what is sought") until you reach the starting point. In geometry this is something given or something already known, e.g. an axiom or an existing proof:

> Now analysis is the way from what is sought—as if it were admitted—through its concomitants in order to something admitted in synthesis. For in analysis we suppose that which is sought to be already done, and we inquire from what it results, and again what is the antecedent of the latter, until we on our backward way light upon something already known and being first in order. *And we call such a method analysis, as being a solution backwards.*

Synthesis goes in the opposite direction from the start through the steps which were found in the analysis, and ends at the goal:

> In synthesis, on the other hand, we suppose that which was reached last in analysis to be already done, and arranging in their natural order as consequents the former antecedents and linking them one with another, we in the end arrive at the construction of the thing sought. And this we call synthesis.

Pappus distinguishes between two kinds of analysis, one for constructing a proof ("theoretical analysis"), and one for ordinary problem solving, i.e. finding and calculating a solution to a stated problem ("problematical analysis"). In the first kind, the proof is the reverse of the analysis:

> In the theoretical kind we suppose the thing sought as being true, and then we pass through its concomitants in order, as though they were true and existent by hypothesis, to something admitted; then, if that which is admitted be true, the thing sought is true, too, *and the proof will be the reverse of analysis*. ...

In the second kind, the steps necessary to reach ("synthesize") the solution consist of the steps of the analysis taken backward.

> In the problematical kind we suppose the desired thing to be known, and then we pass through its concomitants in order, as though they were true, up to something admitted. If the thing admitted is possible or can be done, that is, if it is what the mathematicians call given, the desired thing will also be possible. The proof will again be the reverse of analysis.

This concludes Pappus' original text.

Pappus in relation to design methods and the four principles

In relation to design methods, the first thing to note about Pappus' description is of course that he describes a process consisting of two parts having the same functions as in design: analyzing the problem and synthesizing a solution. The division of these two functions into *separate phases* is also there, as is their *relative order*. It is also clear that this order is appropriately regarded as natural or logical, as it is embodied naturally in what the two processes' functions are.

Also the principles of planning and product–process symmetry are present in Pappus' account, albeit more indirectly. This is because geometrical proofs are so very much simpler than design problems; as stated above, these two principles mainly aim to handle the complexity within a phase, a matter which is not as pressing in geometry. For this reason, Pappus provides only a very basic decomposition principle.

The central idea of planning is that of letting the work of the synthesis phase be determined in advance during analysis. As indicated above, this structure may follow different principles. The Pappan approach to this is that the proof consists of the path taken in the analysis, only in the reverse order. The proof is then the plan, as this reverse order is also what the synthesis is to follow.

The product–process symmetry is also present, although this too in a very simple form. The product of a geometrical proof problem is the proof itself: the sequence of steps from what is given to what is sought, which constitutes the demonstration. And in this case both phases, analysis and synthesis, have the structure of the the proof.

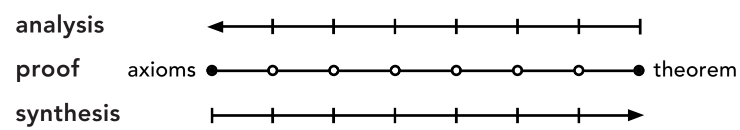

Figure 1.5 The relation between analysis, synthesis, and the proof. Compare with figure 1.8.

There is ample evidence that the method of analysis and synthesis was known even long before Pappus (Hintikka & Remes 1974, pp. 7, 85, 100). First, Pappus here refers to them as originating in Euclid and others. Secondly, there are other writers who refer to what appears to be the same things. Generally, the methods are held to have been known by Aristotle at least, and possibly invented by Plato; in these cases, we are back some 700 years earlier, in the fourth century BC. Pappus is accredited with giving the first comprehensive *description* of this as a method; like a methodologist describing a method already used by "practitioners", and for others to follow. This thereby comes very close to design methodology.

Polya

A much more recent commentary on Pappus is found in Polya's *How To Solve It* (1945), which devotes one section to a detailed account of

Pappus' method, including examples, further explanation, and drawing out certain consequences. Polya can also be regarded as a methodologist of mathematical problem solving, and he has served as the historical link from Pappus to modern-day methodology. First, he gives a "non-mathematical illustration" (p. 145):

> A primitive man wishes to cross a creek, but he cannot do so in the usual way because the water has risen overnight. Thus, the crossing becomes the object of a problem; "crossing the creek" is the *x* of this primitive problem. The man may recall that he has crossed some other creek by walking along a fallen tree. He looks around for a suitable fallen tree … He cannot find any suitable tree but there are plenty of trees standing along the creek: he wishes that one of them would fall. Could he make a tree fall across the creek?

This example makes a very good illustration of Pappus' method: the analysis makes a chain, from crossing the creek to walking on a fallen tree, to finding a suitable tree, to felling a tree on the bank of the river, etc. The synthesis carries out what the analysis has thought out, in the reverse order. Polya elaborates:

> This succession of ideas should be called analysis if we accept Pappus' terminology. … What will be the synthesis? Translation of ideas into actions. The finishing act of the synthesis is walking along a tree across the creek.
>
> The same objects fill the analysis and the synthesis; they exercise the mind of the man in the analysis and his muscles in the synthesis; The analysis consists in thoughts, the synthesis in acts. There is another difference; the order is reversed. Walking across the creek is the first desire from which the analysis starts and it is the last act with which the synthesis ends. …
>
> Analysis comes naturally first, synthesis afterwards. Analysis is invention, synthesis execution; *analysis is devising a plan, synthesis carrying through the plan*. (pp. 145–146)

Most important here is how Polya explicitly spells out the relation of *planning* to analysis & synthesis, which was only left implicit by Pappus. Polya himself writes that this passage "hints a little more distinctly than the original at the natural connection between analysis and synthesis". The final quoted sentence spells it out: analysis is making a plan, and synthesis is executing it.

He also describes the order between analysis and synthesis as "natural". Elsewhere gives further comment: "it is generaly useless to carry out details without having seen the big connection, or having made a sort of *plan*. … It is foolish to answer a question you do not understand." (p. 6)

The extent of Pappus' influence on Polya is the most striking in other parts of the book, where Pappus is not mentioned. For example, the idea of *heuristics* is central to Polya, but this concept too is usually attributed to Pappus. But more importantly, the influence is also quite clear in the parts that could be considered as the core of Polya's own contribution. The best example is in the very opening of the book, where he presents his general problem-solving schema, consisting of four parts (pp. xvi–xvii):

- Understanding the problem
- Devising a plan
  Find the connection between the data and the unknown.
  You should obtain eventually a *plan* of the solution.
- Carrying out the plan
- Looking back

This is Polya's overarching schema; it is a "method" of problem solving in every sense of the word. It is a mathematical equivalent to design methods. Furthermore, it is plain to see that this is an extended version of Pappus' original two-part scheme. A final stage of evaluation has been added for pedagogical purposes, and the analysis–synthesis schema has been refined with Polya's clarification regarding planning: The analysis has been elaborated into understanding the problem and devising a plan, and synthesis is called "carrying out the plan" (figure 1.6 overleaf).

With these ideas being so similar, is it likely that the design methodologists knew about these things; was there a link? Yes, it seems so. Although neither Alexander nor Jones gives any indication in that direction, there is good reason to believe that both were at least familiar with Polya's work. Given this suspicion, stemming from the unmistakable similarities between their methods and Polya's, one finds that Alexander (1964) cites later works by Polya, and also (Miller, Galanter & Pribram 1960) where Polya's method is presented in some detail. Jones' link is more indirect; however, the structure of (Jones 1970) is so similar to Polya (1945) that it could have been named "How

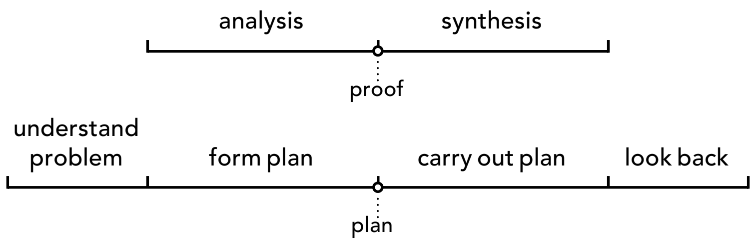

Figure 1.6 Polya's problem solving schema as an elaboration of Pappus' version. Here the proof and the plan are marked as entities passed on between the first and second halfs in both models.

To Design It": besides being based on the analysis– synthesis–evaluation pattern, both texts contain a catalog of heuristic techniques.

Descartes

Even though both of the principles of planning and product–process symmetry can indirectly be found already in Pappus, there is one point on which present-day design methods differ from the Greek original, and this is the technique of *hierarchical decomposition*. Its invention should probably be attributed to René Descartes, as part of the method he developed, a derivation of the Pappan original. This method was presented in his *Rules for the Direction of the Mind* and *Discourse on Method* (1628, 1637):

> We shall comply with it exactly, if we *resolve* involved and obscure data step by step *into those which are simpler*, and then *starting from* the intuition of those which are *simplest*, endeavour to *ascend* to the knowledge of all the others doing so by corresponding steps [taken in reverse order]. (*Rules*, rule v, my italics)

Loosely, hierarchical decomposition consists in the strategy of divide and conquer; in making a complex matter manageable by breaking it down into successively smaller parts, which taken together form a tree-structure of the problem. This is where the concepts of "top down" and "bottom up" come from. Here, analysis is the process which breaks the problem down into parts and creates the hierarchy, and synthesis is where the pieces are reassembled with the same structure to make up the solution (figure 1.7).

Like these chains, the hierarchical analysis and synthesis processes are also symmetrical to each other: they follow the same tree struc-

ture, only in opposite orders. Furthermore, they adhere to the principle of product–process symmetry: their tree structure is the same as the solution structure. This is a more powerful organizational principle than the strictly linear one that Pappus used, yet it can be regarded as refining rather than replacing the original.

Descartes developed his method with the objective of using it in his own effort to establish a systematic and absolutely certain foundation for science and all knowledge in general; the method would serve as the basis and rationale for this system.

> The aim of [the method] should be that of so guiding our mental powers that they are made capable of passing sound and true judgments *on all that presents itself to us*. (Rule I)

It was in this system that "cogito ergo sum" was made into the most certain and fundamental fact of all, and from which all other facts were to be systematically deduced.

As a system of this kind would be somewhat more complex than a geometrical proof, it called for a more powerful organizing scheme than Pappus had needed; this was presumably the reason why he developed the principle of hierarchical decomposition, something that hadn't been called for in geometry.

Descartes' definition of method

Descartes clearly stated mathematics as the source of inspiration (Rule II). Although admitting familiarity with the work of Pappus and other "ancient geometers", and stating that they had used a method of analysis, he claimed that they had hidden it from others:

> We have sufficient evidence that the ancient geometers made use of a certain "analysis" which they applied in the resolution of all

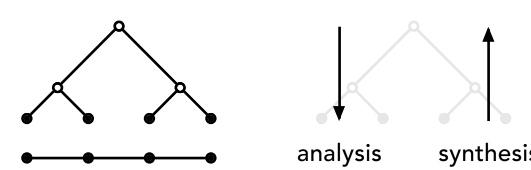

Figure 1.7 Descartes' hierarchical decomposition principle arranges the elements into a tree, whereas Pappus' version simply forms a linear chain (cf. figure 1.5).

> their problems, although, as we find, they grudged to their successors knowledge of this method.… Certain vestiges of this true mathematics I seem to find in Pappus and Diophantus… These writers, I am inclined to believe, by a certain baneful craftiness kept the secrets of this mathematics to themselves. (Rule IV)

They had withheld this knowledge, he claimed, so as not to take the luster off their own mathematical achievements—it would have made them look trivial, as nothing could be simpler than using this method (*ibid.*). He could therefore claim the method he was describing to be of his own invention. Judging from the philosophical literature on the topic, he seems to have had some success in this.

Descartes is also remembered for being the first to reason *about* methods; about their function, advantages, and so forth. Descartes also gave a definition of method:

> …Now by method I intend to signify rules which are certain and easy and such that whosoever will observe them accurately will never assume what is false as true, or uselessly waste his mental efforts, but gradually and steadily advancing in knowledge will attain to a true understanding of all those things which lie within his powers. (IV)

Method, rationality, and logic

Another point to recognize is his claim for the method to be of use far beyond his initial aim, and not to be restricted to any specific field. He claimed it to be general-purpose, to be used in all domains with the same results, as a means for attaining "universal Wisdom":

> For this discipline claims to contain the primary rudiments of human reason, and to extend to the eliciting of truths in every field whatsoever. (IV)

In making this claim, Descartes connects method with rationality. The generally recognized concept of rationality is itself inherently circular. It is defined as the "state of reasonableness", and "rational' is defined as "of the reason", from *ratio* which means *reason*, that is, the same thing. Consequently, rational thinking thus means "thinking that is of the mind". The concept of rationality hence says nothing about just *what* reason consists of, what is reasonable, and so on, so we need to add a theory that says what reason consists of, that determines what is agreeable to reason, and so forth.

What Descartes does is to propose that *method* is the foundation of

rationality. It is his method that "contains the primary rudiments of human reason", that can determine what is agreeable to reason, etc. Still today, arguably, the received view of rationality and reason is derived from method as well as logic; this holds for scientific theory as much as for lay views of rationality. Good thinking is thinking that follows a method; a particular procedure.

The best description of the relation between rationality and design methods, and methods in general, is given by Parnas & Clements:

> A perfectly rational person is one who always has a good reason for what he does. Each step taken can be shown to be the best way to get to a well defined goal. Most of us like to think of ourselves as rational professionals. However, to many observers, the usual process of designing software appears quite irrational. Programmers start without a clear statement of desired behavior and implementation constraints. They make a long sequence of design decisions with no clear statement of why they do things the way they do. Their rationale is rarely explained.
>
> Many of us are not satisfied with such a design process. That is why there is research in software design, programming methods, structured programming, and related topics. *Ideally, we would like to derive our programs from a statement of requirements in the same sense that theorems are derived from axioms in a published proof.* All of the methodologies that can be considered "top down" are the result of our desire to have a rational systematic way of designing software. (1986, p. 251, my italics)

The search for design methods is motivated by the desire for a rational design process, and the authors define rationality as "having good reason"; this is a good common-sense definition of rationality, having good reason for what you do, although the circularity is evident.

Here is also a rare mention of proofs as the ideal for design methods. This also points to logic, another important part of this picture. With method as the model for rational thinking, logic may be defined as method for thinking, and what is the discipline of logic if not the methodology of thinking and reasoning? With this in mind, and with the state of utter refinement that modern formal logic has reached, it is not hard to see why Parnas & Clements regard formal proofs as the model for rational design methods. And the history of logic follows a path that is similar to those of method and rationality, all the way back to Pappus and geometry.

This can also be seen in the format of modern formal proofs. The principal difference from the "classical pattern" described by Pappus is that the steps of the proof are only given once. From having in Pappus' days been presented first in a conceived order of discovery, then followed by the proof per se; today the first sequence is omitted, and only the proof proper is given. Hence, this "modern pattern" is equal to the second half of the classical pattern. As a consequence, a proof today begins with the axioms and from them goes through the steps that end with the proven theorem.
And as this is also the format used in mathematical proofs today, this modern format has taken over the role that the classical format (as

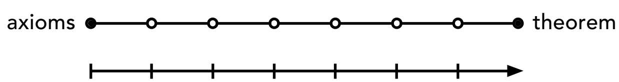

Figure 1.8 The structure of a modern proof. Cf. figure 1.5.

described by Pappus) was once invented for, in mathematical proofs. Hence, modern formal logic is also directly connected to the ancient geometry proofs. With respect to its application in mathematics, this seems not inappropriate at all; more so, however, when logic is held as some kind of model for human thinking, which is not so uncommon.

The domains of rationality, method, and logic are therefore tied to each other, both through their common historical origins and via principles and ideas that they share still today: Therefore (formal) logic, as the scientific study of rational thinking, may be seen as the purest form of methodology; and the principles of logic as the ideal for all other methods to aspire to, in design or otherwise.

1.3 Folk conceptions of thinking

With these ties in mind, it should come as no surprise that traces of the model of rationality can be found also in theories of cognition. More surprising, perhaps, is that the influence has been monumental, and continues to be so. But if we only looked for this influence in scientific approaches to cognition, even in the widest possible sense, then some of the most important aspects of its influence would elude us. The defining characteristics of this pattern can be found already in conceptions of the mental realm that lie well outside the domain of science. Our everyday conceptions of mind are so entrenched in these

basic notions that it is hard to envisage how they might be different. An alternative conception that is not based on the classical model of rationality is hard to even imagine—remember how the definitions of what reasoning and rationality *is* are largely made in its terms, in both lay and scientific language.

The folk model of cognition

The most basic of all conceptions of the mind and its workings is the distinction made between *perception, thinking,* and *action*. This is a good example of a notion that is hard to rethink or disregard: How could it be different; how might it not be this way? Perception refers to the functioning of the five senses; how could these and action *not* be separate from thinking? Hence, this blessed trinity is said to be very deeply rooted in "folk psychology"; a term that cognitive scientists use for the body of everyday, non-scientific psychological conceptions held by every one of us, independent of schooling.

An additional aspect of this trinity is its being arranged in a linear order from perception to thinking to action, based on an imagined "flow" from input to output: Information enters through the senses and via perception goes into the mind. Then, a decision is made which transforms this information via the motor system into action, and this is regarded as "output". Taken together, this yields a folk psychological three-stage model of perception, cognition, and action (cf. figure 1.9).

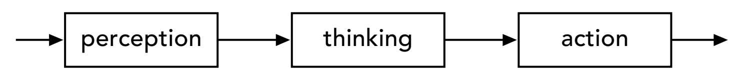

Figure 1.9 The folk-psychological schema of the linear relation between perception, thinking, and action.

A second core theme of folk psychology, only slightly less fundamental than the previous one, is the *intention–plan–action* triad (cf. figure 1.10). Behind it lies the intuition that thought controls our actions and behavior in general (springing from "free will", or the like). It is essentially a causal explanation of action and how action comes about, what controls it, etc. It consists of a three-part chain where the intention plays the role as cause, which via the mechanism of planning determines action. This is typically regarded as a folk-psychological "theory" of action, or rather of the relation of thought to action: a theory since it has the characteristics of a scientific theory; psychological since it is held to explain how people conceive of thinking, action and the relation between the two; and "folk" because of its simplicity and thereby poor standard as a theory.

The primary element in this explanatory schema is planning, since that is the mechanism that performs the translation from thought to action. It takes an intention as input, specifying what the individual wants to happen, and derives from it a plan that specifies a set of operations to perform. The plan is then passed on to the motor system which executes it, i.e. carries out the specified operations, and thereby brings about the intended outcome. As the origin of the intention is non-essential to the relation between thought and action, it can be left out of this schema.

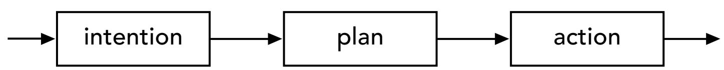

Figure 1.10 The folk-psychological intention–plan–action schema.

There is a distinct relation between the two presented schemas and the concepts therein. The first schema describes cognitive functioning on the highest and most general level of description. The second one is more specific, it concerns the relation between thinking and action, and treats a part of the general schema in greater detail. This elaboration consists firstly in introducing plans as the construct that connects thinking to action, and secondly in specifying thinking, the central part of the first schema, in greater detail: In order to realize the connection between thinking and action, a part of thinking must produce the plan that is passed on to action. The function doing this is of course planning, and it requires an intention as input. As the origin of the intention is non-essential to the relation between thought and action, it can be left out. Hence, the second schema is a refinement of the first (figure 1.11).

Connection to Pappus

How is this folk schema of cognition related to Pappus' method of analysis and synthesis? This is made clear in Polya's elaboration on Pappus, especially in the primitive-man example, which transfers Pappus' schema to a non-mathematical domain. Polya's point is that analysis corresponds to thinking, and synthesis to action.

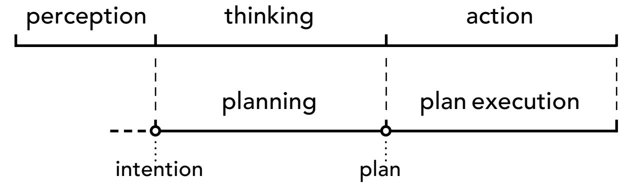

Figure 1.11 Timeline renditions of the folk psychological of perception–thinking–action and intention–plan–action, also showing how the second is an elaboration of the first.

First, the chain of steps that makes up analysis, going from the goal backward, is the train of thought that the man goes through in figuring out how to cross the river:

> The man might remember himself having crossed another river by walking on a fallen tree. He therefore looks around for a suitable fallen tree.... He doesn't find one but there are plenty of trees standing on the shore. He wishes one of them to fall. Could he himself make one fall across the river? ... *This succession of ideas should be called analysis* if we accept Pappus' terminology.

Secondly, synthesis consists in physically carrying out what thinking has conceived during analysis:

> What will be the synthesis? Translation of ideas into actions. The finishing act of the synthesis is walking along a tree across the creek. (Polya 1945, p. 145)

When Polya also relates them to planning and combines them into a whole, the relation between the cognitive schemas and Pappus' method becomes quite clear:

> The same objects fill the analysis and the synthesis; they exercise the mind of the man in the analysis and his muscles in the synthesis... Analysis comes naturally first, synthesis afterwards. Analysis is invention, synthesis execution; *Analysis is devising a plan, synthesis carrying through the plan*. (pp. 145–146)

Here, it is quite appropriate to equate the intention with the "goal" or "the thing sought", i.e. the theorem to be proven. Planning consists of analysis, which takes the intention as its starting point. Similarly, the intention is realized when the final step of the synthesis is performed. The result is a one-to-one match with Pappus' method,

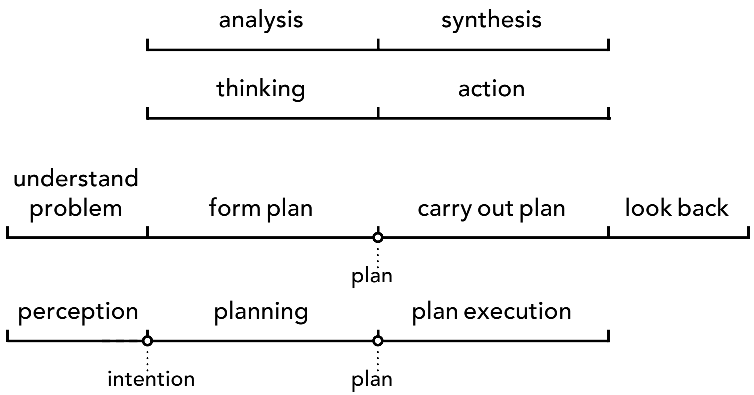

Figure 1.12 Comparisons between the basic and elaborated versions of folk psychology and Pappus' (Polya's) method.

between both the basic and elaborated versions, where the latter in both domains are yielded by adding planning (figure 1.12). This folk schema of cognition embodies the idea of intramental cognition. This view, which is entirely consistent with our intuitions, hinges on the separation of action and thinking, among other things.

Aristotle and the origin of folk psychology

So folk psychology follows very closely the ancient model of mathematical rationality. Or is it the other way around? Is this model based on everyday conceptions? There is a passage by Aristotle on deliberation (*bouleusis*) or "planning" compared to analysis, where he derives the equivalent of the intention–plan–action schema from the method of geometrical analysis (cf. Hintikka & Remes 1974, pp. 85f); it supplies us with evidence that contemporary folk psychology too is based on the ancient Greek method of analysis.

First it should be established that this account of "deliberation" really is concerned with planning, and not careful thinking in general as the term signifies today. Aristotle states that an individual's deliberation is concerned only with what she can do, what her own efforts can achieve. That is, with *action* (Aristotle, *Nichomachean Ethics* III:3, my italics):

> But we do not deliberate even about all human affairs; for instance, no Spartan deliberates about the best constitution for the Scythians. *For none of these things can be brought about by our own efforts.... We deliberate about things that are in our power and can be done*; and these are in fact what is left. For nature, necessity, and chance are thought to be causes, and also reason and everything that depends on man. Now every class of men *deliberates about the things that can be done by their own efforts*. And in the case of exact and self-contained sciences there is no deliberation, e.g. about the letters of the alphabet (for we have no doubt how they should be written); *but the things that are brought about by our own efforts, but not always in the same way, are the things about which we deliberate*, e.g. questions of medical treatment or of money-making.

Hence, deliberation is concerned with things we can make happen, but for which it is not obvious how to attain them; deliberation serves to figure out how to do this. It is therefore planning of action that Aristotle is discussing here.

Further, he states that ends, i.e. goals, purposes, aims, are not at issue in deliberation but are regarded as given. That is, the agent's intentions are not at stake, only how to attain them. This closely matches the view of planning as the mapping from intention to action:

> *We deliberate not about ends but about means*. For a doctor does not deliberate whether he shall heal, nor an orator whether he shall persuade, nor a statesman whether he shall produce law and order, nor does any one else deliberate about his end. *They assume the end and consider how and by what means it is to be attained*; and if it seems to be produced by several means they consider by which it is most easily and best produced, while if it is achieved by one only they consider how it will be achieved by this and by what means this will be achieved, *till they come to the first cause, which in the order of discovery is last. For the person who deliberates seems to investigate and analyse in the way described as though he were analysing a geometrical construction* (not all investigation appears to be deliberation—for instance mathematical investigations—but all deliberation is investigation), *and what is last in the order of analysis seems to be first in the order of becoming*.

Aristotle here provides conclusive evidence that his account of planning is directly based on the pattern of geometrical proofs. He even explicitly names geometrical analysis as the archetype for it, but even regardless of this mention the evidence is unequivocal: There is the assuming of the sought end, and then considering how it is reached; then there is the repetition of this "till they come to the first cause", which in turn is done in the reverse order in the "becoming" where the actions are carried out. Note that deliberation is compared to analysis only; not synthesis. Also, from the manner in which the final clause describes "becoming", it can only be understood as though action is held to follow planning in the same manner that synthesis follows analysis (not only in the temporal sense).

A closer analysis of the connection between Pappus' and Aristotle's accounts is made in Hintikka & Remes (1974, chs. 1 & 8), showing also extensive terminological similarities. However, they still hold that Pappus did not directly draw on the work by Aristotle.

The everyday meaning of "design"

With this in hand, one can analyze the conventional, everyday meaning of the term "design" itself. It is used both as noun and verb, where the verb, as in the process of design, is what the present book is concerned with. Keeping in mind that etymology is always a precarious venture, "design" comes from the Latin *designare*, to designate, which here means to *specify*, as in pointing out what to do. The modern sense of design is held to have originated in the Renaissance, when architect and builder functions came to be two separate functions. The architect would no longer always be present on site during building and therefore had to specify what to build, which previously hadn't been necessary (Herbert 1993).

Similarly, the noun "design" comes from *signum*, which is *not* so much in the modern sense of the root "sign" (as in symbol, mark; semantics, semiotics, etc.) as is sometimes claimed. It rather has the meaning of something that you follow, in the sense of the specifications passed on from architect to builder.

> Around the sixteenth century, there emerged in most of the European languages the term "design" or its equivalent. The emergence of the word coincided with the need to describe the occupation of designing. ... Above all, the term indicated that designing was to be separated from doing.
>
> (Cooley 1988, p. 197, quoted in Bødker et al. 1991)

In addition to this standard meaning of the noun, there are some peculiar senses which give indications toward the folk conception of how designing is done (definitions and etymology taken from the *American Heritage Dictionary*, synonyms from Roget, my italics):

> **de·sign** *n.* ... 7. A *plan*; a project. 8A. A *reasoned purpose*; an *intent*. B. *Deliberate intention*.

Here one somewhat unexpectedly finds evidence of the everyday view of design activity, given in the very definition of design. The given meanings include plan and intention, which both are part of the intention–plan–action schema. These folk conceptions are clearly expressed in the definition of the verb:

> **de·sign** *v.* 1A. *To conceive or fashion in the mind*; invent: design a good excuse for not attending the conference. B. To *formulate a plan* for; devise: designed a marketing strategy for the new product. 2. To plan out in systematic, usually graphic form: design a building; design a computer program. 3. To create or contrive for a particular purpose or effect: a game designed to appeal to all ages. 4. To have as a goal or purpose; *intend*. 5. To create or execute in an artistic or highly skilled manner.

The present topic is accurately captured by the sense listed as second—the illustrations are those of the dictionary—and sense number 5 is a variant thereof. The senses of the verb that remain capture the folk psychological view of design—"folk design theory" if you will: 1A states that design is an act of the mind; 1B (and 2) refers to planning, and number four mentions intent. Hence two items in the intention–plan–action triad are mentioned. When this connection is stated in the standard definitions, it strongly backs up a deeply rooted connection between the meanings of plan, intention, and design. The logic behind this becomes clear if these folk psychological terms are seen in relation to Pappus' schema (figure 1.13 below).

Here, action is the final step, whereas intention and plan both belong to the domain of thinking. Hence, in this view, design corresponds to transforming the goal into a plan for the implementation; a blueprint for how the product is to be executed. In other words, design corresponds to planning, or more roundly, to thinking. By extension, the design–implementation dichotomy corresponds to those of planning–execution, thought–action, and analysis–synthesis; a deceptively natural fit that surely has a large share in the appeal of this view. Note that this separation of design and implementation corresponds to the original division in Renaissance architecture.

Points 3 and 4 in the definition bear witness to the role of the intention in this scheme. At first sight they seem indistinguishable;

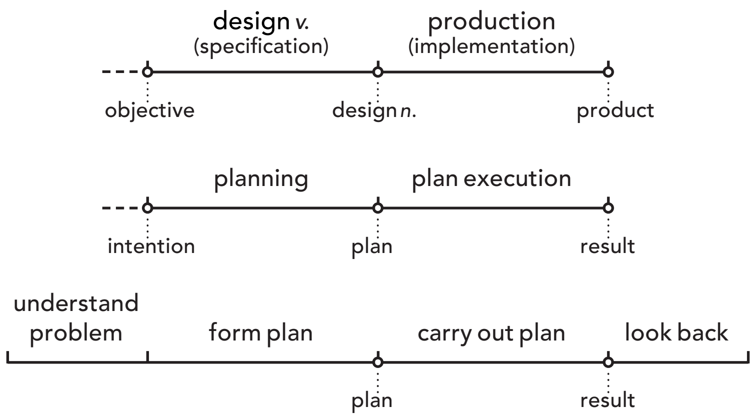

Figure 1.13 The folk-psychological notion of design, cf. the intention–plan–action schema (middle) and Pappus'/Polya's method (bottom).

however, the first refers to *creating* for a certain purpose; the second to *having as* purpose. In other words, having an aim vs. turning it into to a plan for its implementation. Folk psychology holds goals and purposes to always have an explicit mental entity, the intention, that directly corresponds to it. The intention is the cause that alone is responsible for starting the process of design, as well as determining its direction. The relation between intention, plan and design is verified by the definitions of the other two terms; even though being synonymous is not the same thing as being identical, this shows that all three terms are given their meaning in relation to the same intention-to-action schema:

in·tend *v. tr.* 1. To have in mind; *plan*. 2A. To *design* for a specific purpose. B. To have in mind for a particular use. 3. To signify or mean. *intr.* To have a *design* or purpose in mind.

plan *v.* 1. To formulate a scheme or program for the accomplishment, enactment, or attainment of. 2. To have as a specific aim or purpose; intend. 3. To draw or make a graphic representation of.

What is more, intention, plan and design are all three consistently listed as *synonyms*, even, and this in the relevant senses:

To have in mind as a goal or purpose: *intend*, aim, contemplate, *design*, envisage, envision, foresee, *plan* …

To form a plan/strategy for: *design*, *plan*, *think out*, prepare, outline, scheme, lay plans …

What one intends to do or achieve: purpose, aim, *design*, goal, intent, *intention*, meaning, *plan* …

So design and implementation eventually came to be assigned to separate parts of an individual's faculties, thought and action. In the very beginning, both design and building were done by one person, requiring her full physical and mental capacities. When design came to be considered as an activity of its own, it referred to one out of two responsibilities divided among two people, each of which still required one individual's full capacity.

In the folk sense it is implicitly assumed that one's thinking alone is sufficient for designing. This is the same effect that the Greek schema has had on our received conceptions of cognitive abilities in general: one's intellectual capacity is held by the mind alone, and intelligent performance is done strictly by thinking, such as in reasoning, problem solving, and so forth.

As the folk sense of design refers to the act of specification, which is only one aspect of a whole design process, I will use *specify* to refer to that particular aspect, in a theory-neutral manner. In relation to this, the topic of this book could be design as in *the whole process of design*.

How "folk" is folk psychology?

The passage from Aristotle shows that what today is considered as the folk-psychological model of action was in fact once upon a time derived directly from the method of geometrical analysis. In the same manner, "folk design theory" seems far too refined and consistent to be a reflection of mere intuitions. This casts some doubts on the view of folk psychology as a direct reflection of everyone's "intuitive" conceptions, which is customary in cognitive science.

This view is problematic by itself. Folk psychology is commonly thought of as a "folk" theory of psychology, as opposed to scientific psychological theory. That is, people are held to have "theories"; there are other examples, such as the "theories of mind" thought to be held by infants as well as primates, or "naive physics". However, it is questionable whether people's conceptions of these matters can be regarded as theories, and one may wonder whether scientists are not too self-indulgent when they attribute these tools of their own professional trade to whomever they study. As "folk" conceptions typically are neither consistent and coherent, nor well integrated and complete, the question is whether calling them theories is not going too far.

It is my belief that when scientists assemble people's conceptions and from them formulate a "folk theory", a coherent whole, they are in doing so attributing to these people a consistency, completeness, and so forth which only has come to exist as a product of the scientists' own research effort. The coherence required for this to be a theory is then something their own labor has had to produce, it is not a property of the material from which they built it.

Therefore, what is called folk psychology is not so very "folk", but rather a semi-scientific theory that is based on or derived from lay conceptions, although far from being a pure reflection of them. These intuitions are in themselves not at all as articulate, coherent, or consistent as this product, the "folk" theory. I believe the same is true for dictionary definitions. What person on the street would you expect to give a coherent and complete, eight items long definition of the noun "design"? Dictionary editors, however, typically do have scientific training.

What is more, these lay conceptions are often described as "intuitions", and the intuitions which the scientist used were typically her own. How representative can we consider these to be? How untainted by their professional experience and education? No ordinary folks would consider scientists to be ordinary folks. It often seems the veil of folk psychology is used to cover the fact that the ideas scientists are describing or defending are in fact their own intuitions, a fact which they do not quite owe up to (e.g. Fodor 1975).

Also, science and everyday life are not perfectly isolated domains. They continually influence each other through the culture they share. For instance, Freudian theory has had a tremendous impact on popular culture (an impact which remains strong, while its weight in science continues to fade). More contemporary are the information and computer metaphors, the impacts of which are beginning to follow the same pattern. Similar patterns are easy to imagine, going at least as far back as into antiquity as in the examples we have just seen, and the passing of ages tends to render them invisible.

All in all, folk psychology is arguably not so "folk" as is sometimes thought. More likely, models of this kind are highly refined derivatives from everyday conceptions, selectively chosen and subjected to scientific method to yield what resembles a coherent, plausible theory. Hence, the "folk" models of thought and action, of design, and so on that we have seen here, have therefore had plenty of opportunities for influence from modern science or ancient writings. In other words, in the good two thousand years that have passed, there has been plenty of time for the original Greek ideas to infiltrate contemporary semi-formalized accounts of these not so mundane topics. And given the *de facto* ties of these subjects with science and academia, it is not as unlikely that they retain a strong influence on even the most refined modern theories of design, problem solving, cognition, etc.

1.4 Cognitive planning theory

Plans and planning have gradually emerged as central concepts: Polya shows that the plan is what holds analysis and synthesis together, he holds analysis to consist in producing a plan, and the proof to be the same thing as the plan; Aristotle took geometrical analysis as the paradigm for planning in human thought; folk design theory holds design to correspond the planning stage in the folk model of cognition, and the "design" is regarded as a plan for building the product; the plan is regarded as the construct that connects thought to action; the third principle concerns planning, etc.

Why does planning keep recurring in this way? The reason why it turns out to be so central is that it is the mechanism by which thinking determines action. In the received, "folk" model of cognition, rational action is an extension to the model for rational thought. There is a model of rational thought, based on geometrical analysis, mathematical proofs, etc. *Acting* rationally means having good reasons for your actions; it is simply explained as a direct extension of rational thought: first you think rationally, then you act out your decision. This also follows the intuition that our minds control our actions.

In this scheme, planning is the construct that realizes this connection: as the last part of thinking, it produces a plan, which is passed on to be executed by the motor system.

Hence, planning theory and the intention–plan–action schema are in fact the received theory of (rational) action. It is based on the following principles: Given, to begin with, is the rationality of thought; add to this the separation of thought and action, and that thought precedes and determines action. Planning and the plan are required to connect the two that are separated.

From this it follows that planning is not a freely elected theoretical choice from a number of available alternatives. With thought and action being separated, which is done already in the basic perception–thinking–action schema, and free will or thought determining action, there is little option but first to place thought prior to action, secondly to introduce an entity needed to connect the separated parts, and thirdly to require the first separated part to produce this entity (i.e. to introduce a planning process into thinking).

In this way, planning theory comes to represent this relation in principle between thought and action, it becomes the concrete manifestation of this abstract principle. This is also why planning keeps coming up in so many different contexts.

What is more, since the planning concept has been directly and completely adopted by cognitive science, it has also become a cornerstone of the standard scientific theories, directly reifying the intuitive notion of thought or free will determining our actions. Planning theory has thereby become the standard scientific theory of action.

The first work to explicitly propose planning as the mechanism connecting intramental thought and action in cognitive theory was *Plans and the Structure of Behavior* (Miller *et al.* 1960). Their chapter on planning in problem solving also draws directly on the work of Polya (1945), with even a hint of its ancient origins. Hence, in this case the link from Pappus' method of analysis via Polya to cognitive planning theory was much more direct and explicit than via folk psychology only.

This is certainly why there has been such a great controversy over planning in cognitive science (e.g. Agre & Chapman 1990, Brooks 1990, 1991a, 1991b, Suchman 1987): this has not been a dispute over people's grocery shopping lists, or how they go about deciding how to spend their summer holidays. The row over planning has concerned the essential nature of human action; needless to say a fundamental issue also for cognitive science. What is more, such a great part of the work that has been done in this discipline hinges on this particular theory of action. If it doesn't hold, the work based on it stands under serious question. For this reason, planning is not a harmless, isolated little sub-aspect of interest only to a "planning community" of researchers, and the enormous heat that has been generated by this debate is understandable. This would not have happened over a

theory of shopping lists. What is interesting is that those explicitly concerned with planning research mainly seem to have taken the consequence of this critique, and have abandoned the received view to create a new field called "reactive planning". This remains to happen outside these circles.

Design as planning

In a paper that is arguably a classic, Jeffries *et al.* (1981) proposed a theory of design as planning, directly based on cognitive planning theory. The authors themselves however do not regard this as a *planning* theory of design as such, just as a cognitive theory. This also goes to show just how fundamental the concept of planning is to cognitive science.

This paper is also significant for another reason: the main method for studying design empirically is to collect protocols of designers thinking aloud while working on a design problem. This is also probably the first protocol study of design. It is also the first of three protocol studies of software design, the other two of which will be treated in the following chapters.

The paper ties together work from three disciplines: The first is design methodology, especially the part concerned with software design. The second is "automatic programming", i.e. attempts at design-by-computer, a branch of AI, and the third is cognitive research on "models of planning and design" (as notably the authors themselves call it). The last two represent the classical AI–cognitive psychology connection in cognitive science. How compatible all three disciplines are is shown by how smoothly their contributions are combined.

Already in the first paragraph, the description of the nature of design is entirely permeated with concepts from planning theory:

> The task of design involves a complex set of processes. Starting from a global statement of a problem, a designer must develop a precise plan for a solution that will be realized in some concrete way (e.g., as a building or as a computer program).

Here as often elsewhere, what is taken for granted in the paper is particularly evident in the very opening statements. Even when the authors here give their introductory description of what design is, they do so in terms of planning theory. This is not an analogy; it is not design *as* planning—design *is* planning. Design consists in developing a plan for the implementation, by translating the given goal into a specification of what should be done:

> Software design is the process of translating a set of task requirements (functional specifications) into a structured description of a computer program that will perform the task.

This definition completely parallels how the dictionary above defined design in terms of folk psychology and planning (cf. figure 1.13). The resulting design consists of a decomposition of the product into modules; the plan for the implementation process follows this decomposition structure:

> One can think of the original goal-oriented specifications as defining the properties that the solution must have. The design identifies the modules that can satisfy these properties. How these modules are to be implemented is a programming task, which follows the design task.

1.5 Problem solving theory

Problem Solving Theory (PST) and Information Processing Theory (IPT) are quite closely related; roughly, the latter is a generalized version in which the former has been made into a universal, domain-independent theory of cognition. Both theories originated in the work of Newell & Simon. The main presentation of these theories is considered to be their book *Human Problem Solving*, which was published in 1972. This was a good fifteen years after their research had begun, so this book must be considered as a mature presentation of their work, even though it may seem old today, or limited in its empirical material. The book, and the theory of problem solving, are based on the study of three problem solving tasks: logic, chess, and cryptarithmetic (i.e. substitute digits for the letters in DONALD + GERALD = ROBERT). And although problem solving is the specific scope of the book, it is also considered to be the major scientific work on information processing theory as a whole. This if anything is a testament to the relation between these theories, and to the status assigned to problem solving as an activity that is representative of human cognitive performance altogether. That is, to the relation between problem solving and cognition as a whole, as it is perceived by cognitive scientists.

These theories bear strong evidence of their origins. The problem with which Newell & Simon begun their research was to construct a computer program capable of proving theorems in formal logic. Both their choice of problem domain as logical proofs, and of computer implementation as their method, are clearly visible in the resulting theories, even though these are formulated in general terms.

Their first work was on a program known as the Logic Theorist (LT), which later was superseded by the General Problem Solver (GPS). The task of LT was to find a proof for a given theorem in formal logic, given the axioms to be used for the proof. For a program to be able to do this, everything needed first had to be encoded appropriately and given to it. Besides axioms and theorem, also the available rules of logic, their proper application, and how they are combined into deductive sequences, had to be encoded into the program in an appropriate form. Then, using various methods, the program was to assemble valid combinations of steps into a sequence leading from axioms to proof.

In order to turn this from a computer program into a theory, the various aspects of the program were expressed in mathematical form. The process was characterized as a search among the available rules, and their possible applications and combinations. This given information that had been fed into the computer was collected in what was called a “search space”, to match the analogy of the program’s function as a search. This space is a mathematical abstraction with no immediate counterpart in the computer program, but embracing various parts of it.

According to historical sources, Newell & Simon drew directly on the work by Polya (1945) in their initial work on LT, and followed his directions closely in developing their first algorithms. The most direct indication of this connection lies in that LT worked strictly backward from theorem to axioms in its search for proofs (Newell & Simon 1972). Additionally, given their oft mentioned concern with heuristics, they could hardly have missed Polya since it is his name that is most often associated with the study of heuristics in modern times. The one missing piece of evidence is that they themselves seem not to acknowledge this connection. (Still it would be unfair to imply that they were attempting to do a Descartes in this matter and deny the origins.) But using Polya’s recognized work on techniques for mathematical problem solving would of course be a wise thing to do, in attempting to develop a computer program with this ability.

When they extended their work to other tasks than logic, when the

Logic Theorist became the General Problem Solver and they formulated a universal theory of problem solving, the previous concepts were simply generalized: The initially given axioms and theorem became start and goal states, defined by the problem in question; the basic "search" view was transfered to problem solving, now taking place in a "problem space". The rules of logic were replaced by the rules of the particular problem, and so on. Hence, the basic pattern from logical deduction was preserved and became the backbone of the theory of problem solving.

At this point Newell & Simon were aiming toward greater psychological realism, and started to collect their own think-aloud protocols of actual subjects working on the problems, with the purpose of replicating such protocols in their programs. With this aim, and from the analysis of these protocols, they developed a new heuristic method called "means–ends analysis", to emulate in GPS what their subjects were doing (1972). Means–ends analysis allows for working alternatingly forward and backward, and it is therefore a more powerful method than working strictly backward (or forward). In conjunction with this, they also developed a heuristic called the "planning method". It consists in developing a simplified solution that abstracts from the specifics of the problem, and thereby can establish a general solution strategy, which then works as a plan that is used for the "implementation" of a solution that does deal with the details.

The first thing to note about these various "heuristic methods" is that they all correspond to the analysis part of Pappus' original schema. The original backward-working method of the Logic Theorist even exactly corresponds to Pappus' method of analysis for a proof problem: both start at the theorem to be proven, and move backward until they reach the axioms, and at this point the resulting steps are delivered as the proof. The only difference is that LT requires that also the axioms be given in advance. Means–ends analysis replaces that strategy with a generalized and more powerful method for finding the proof, but it *only* changes the method for finding it—the internal workings of the analysis stage—the givens are the same, axioms and theorem, and the result is the same, the proof.

The same holds also for the planning method; it changes only the method of analysis while preserving everything else. What is more, this method of analysis is identical to top-down decomposition in cognitive planning theory: it organizes the analysis phase in the same way that planning theory does. There are also great similarities between the planning method in problem-solving theory and Polya's "creating a plan". So these may be seen as three different "decomposition strategies", cf. Descartes' hierarchical principle.

The influence of folk psychology on current cognitive theory

But the most important point in this is that these are three variants of the analysis stage in Pappus (counting in strict backward movement). They are surely not equally powerful, but neither of them goes beyond analysis, which corresponds to thought in the folk-psychological schema, nor do they deviate from the rest of the schema on any point—the division into perception, thought, and action is preserved, as is planning as the mechanism connection thinking and action. And since the goal is considered as given, also the intention is left as is. They differ in the inner workings of the thinking box in the middle, but the rest is left intact as it has been in folk psychology all along and still is; they are different theories of thought, and of how thinking produces the plan for action (figure 1.14).

And this is what is striking about the current state of cognitive theory: while vast amounts of work have been spent on the details of the inner workings of the mind, the folk schema of cognition has been adopted by scientific theory on all other points, and stands unaltered even today. This holds for both planning theory and problem-solving/information-processing theory. The fundamental organization of the cognitive system has thereby been preserved more or less unchanged from Aristotle's treatise on planning. That is also why this section, even though serving to present cognitive theory, is dominat-

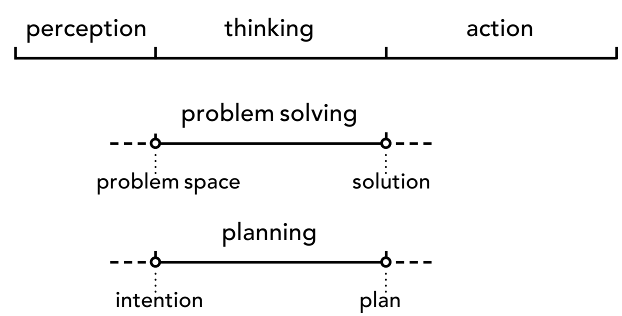

Figure 1.14 Problem-solving theory and planning theory as variants of the "thinking" step in the folk-psychological schema.

ed by the presentation of the pre-scientific, folk conceptions of cognition: These notions are where the majority of the concepts originate that are relevant for these theories. The continuous, unbroken line of development from folk psychology to modern-day cognitive science is especially clear in the case of planning and planning theory.

From this it is clear that today's cognitive theories also preserve the intramental view of cognition. One might say that these theories disagree over the intramental details, while they all preserve the basic patterns from folk psychology that are derived from Pappus, Aristotle, and so forth. When connectionism was up and coming, it was widely hoped that it would replace traditional information-processing theory, or what was then seen as the symbolic theories. It was seen as a question of symbolic vs. connectionist theory. It has since become clear that this replacement would not happen; the connectionists never really took on the higher-level issues. (Although Rumelhart *et al.* 1986 hinted at how this might be done.) Instead they took the route of the natural sciences, down into the biology, even the chemistry. That is why information processing theory and the work of Newell & Simon still stands as the best alternative; more or less unchallenged, even.

The reason why I will use "intramental" to refer to the traditional theories is also to a large extent just this: The fact that these are symbolic or information-processing theories, etc., is beside the point. What is significant is that they retain the view of cognition as intramental—a fact that connectionism hardly challenges, for example. This is the aspect of traditional theory that I will point out as the problem in what follows.

I am presenting PST here for several reasons. One is that it is the most influential theory of cognitive science. It was very early, pre-1960 even, and it has served as the basis for very influential, more recent theories: SOAR (Newell 1990) as well as GOMS (Card, Moran & Newell 1983). An additional but related reason is that PST definitely is the most developed cognitive theory when it comes to design specifically. This was explicitly named by Newell & Simon (1972) as a suitable domain for taking the theory to more realistic, more complex problems. This they have also done, in particular Simon, who has shown an interest in design (e.g. 1973, 1981). It has also been done by a number of others (e.g. Akin 1986, Goel 1995).

But the main reason for discussing problem-solving theory here is that I will use it as the basis for my critique. I have made this choice

because it is the *best* theory that cognitive science has to offer, with respect to the topics I will be discussing. That is to say, I have *not* chosen it as a bad example. Even though my argument will to a large extent be based on the flaws and problems with this theory, this should not be seen as much as a criticism against Newell & Simon. As scientists they are among the very best. And problem-solving theory is not a bad theory (it is wrong, however). In studying their work, even though I have not been on their side, I have come to regard them highly for what they did. For example, they did consider and work out the issues of how their theory would connect to action and the outside world, to an extent that no one else has done (Newell & Simon 1972, Simon 1981). Their theory is also coherent, and regarding these issues, they would not easily lend themselves to making *ad hoc* extensions that were not compatible with other parts of the theory, as is too often done.

Hence, the critique of problem solving theory should not be seen so much as a critique directed toward Newell and Simon, but rather toward the rest of cognitive science, which in the good 25 years since has hardly even come up with anything significantly different. The only real alternative that might be considered here, for example, is mental models theory (Johnson-Laird 1983). In all, it seems cognitive science has ever sincebeen little more than footnotes to Newell and Simon.

1.6 Summary: A common model of rational action

In the introduction I stated that this chapter would establish a connection between design methodology and cognitive theory. These are two disciplines that seem quite unrelated at first sight. Here I have shown the link to be a shared underlying pattern that is *a general model of rationality and rational action*. This model has been the foundation for such apparently diverse domains as classical design methods and structured development techniques in software engineering; with respect to cognition, folk psychology, cognitive planning theory, problem-solving theory, and information-processing theory; and lastly, to proof theory and formal logic, and the philosophical notions of method and rationality.

Although they vary in their details, for various reasons, these widely different domains all are founded on this underlying pattern. This pattern is what the four defining principles are meant to make visible, and they should be seen as my attempt to make this model of ra-

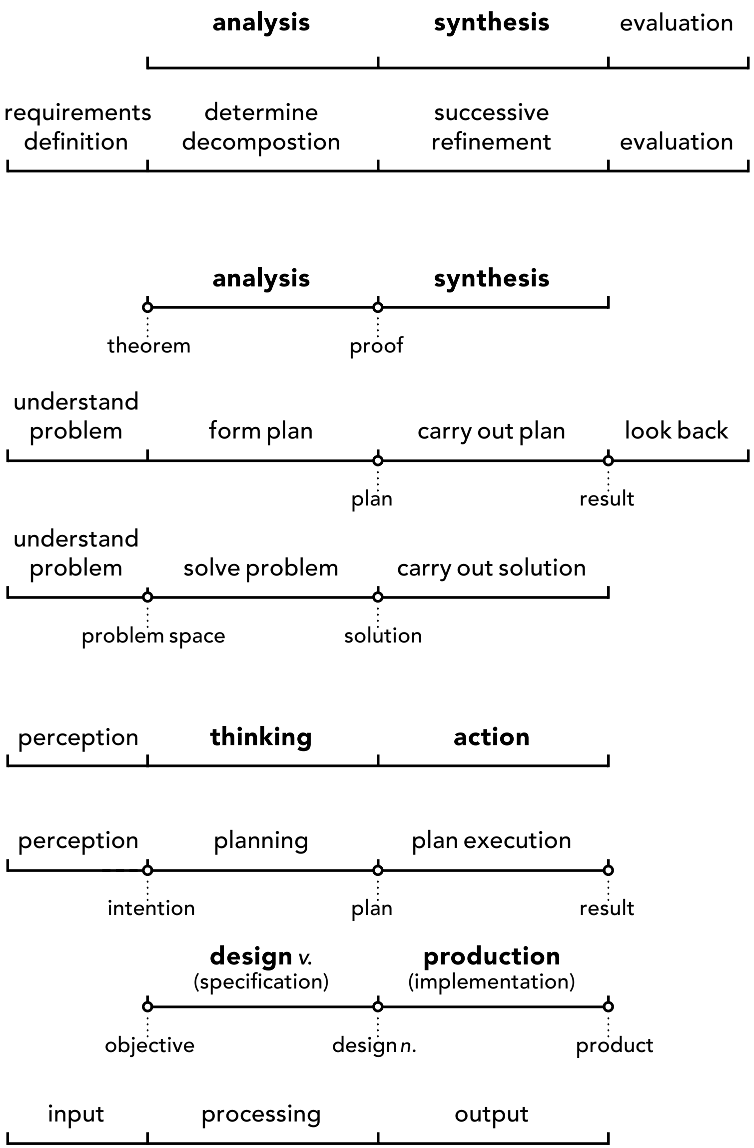

Figure 1.15 The Rosetta stone of rational action models. The models belong to three categories: design methodology, logic/problem solving, and folk psychology. The final line is the basic pattern of computer architectures.

tional action visible and explicit, and to articulate its essential underlying ideas.

By placing the various timeline diagrams beside each other, you get a Rosetta stone of rational action models, as in figure 1.15. There, the models can be compared and their corresponding elements can be read out directly. For example, the analysis–synthesis dichotomy lies behind those of thought–action, planning–plan execution, and design–implementation. These pairs make up the stem of the family tree, as it were. (The original Rosetta stone was found in the Nile delta, not far from Alexandria.)

An intramental model of rationality

This underlying model of rationality and rational action, which is based on ancient geometry proofs, stands as an archetype of good and desirable thinking, and it is the paradigm after which all the descendants have been modeled. The four principles capture the nature and essence of this archetype, showing that it is essentially an *intramental model of cognition and action*: The central idea is of rational *action* as an extension of rational *thinking*, and the elements of this idea is expressed in the four principles. Taking the rationality of thought as a premise, they make the following points, each in turn:

0. The rationality of thought.
1. *separation*: The separation of thought and action.
2. *logical order*: Thought preceding and determining action.
3. *planning*: Plans (and the intention-to-action schema) as the mechanism whereby thought pre-determines action
4. *product–process symmetry*: The idea that the *structure* of a *product* of action directly reflects structure of the *process* which produced it, and thereby also of the underlying *plan*.

In particular, the final point implies that since this structure was inherent also in the plan, it thereby had to be known before the start of the process which produced it.

As seen in the lower part of the Rosetta stone, the basic computer architecture of input–processing–output also follows the ancient pattern, with "processing" corresponding to analysis and thinking. This connection is likewise evident when Jones compares the designer to

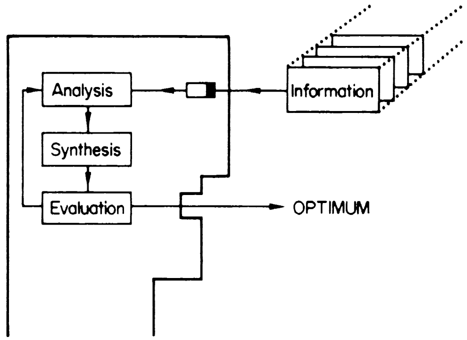

Figure 1.16 Original caption: "Designer as computer".

a computer—here also note the labels on the boxes inside the head in figure 1.16 (1970, p. 50):

> The picture of the rational, or systematic, designer is very much that of a human computer, a person who operates only on the information that is fed to him, and who follows through a planned sequence of analytical, synthetic, and evaluative steps and cycles until he recognizes the best of all possible solutions. This assumption is, of course, valid in the case of computer optimization of the variables within a known design situation, but it also underlies such systematic design methods as morphology and systems engineering which are intended for the human "computer" to use in solving unfamiliar design problems.

Here, Jones at once ties together the basic three-stage design schema, the input–processing–output schema, and thinking/problem solving. Jones also mentions "cycles"—notably in relation to computers—or *iteration* as is today's term; a ubiquitous feature of contemporary design methods; also notice the backward arrow from evaluation to analysis in the "flowchart" inside the head. The reader may have wondered why I have left this prominent aspect out of my analysis of design methods; a brief answer is that this is an *ad hoc* extension that goes quite counter to the principles of the underlying model of rationality, in effect proving it wrong; also cf. chapter 3.