Diagrams as Tools in the Design of Information Systems

Jeffrey V. Nickerson,
Stevens Institute of Technology, USA

James E. Corter, Barbara Tversky,
Columbia University, USA

Doris Zahner,
Stevens Institute of Technology, USA

and **Yun Jin Rho**
Columbia University, USA

Design typically relies on diagrams to offload memory and information processing and to promote discovery and inferences. Design of information systems, in contrast to design of buildings and products, depends on topological connectivity rather than Euclidean distance. Understanding graph topology and manipulating graphs are essential skills in the design of information systems, because graph manipulation facilitates the refinement of designs and the generation of alternative designs. Here, we found that students of systems design have difficulties interpreting diagrams, revealing two biases, a sequential bias and a reading order bias. The results have implications for teaching as well as diagram design.

Design Computing and Cognition DCC'08. J.S. Gero and A. Goel (eds), pp. 103-122. © Springer 2008

Introduction

Design entails arranging and rearranging real or virtual objects and parts and evaluating the resulting configurations. Although the mind seems to have almost unlimited space to passively store information, its space for actively manipulating information is highly limited. When the mind runs out of mental space, it often turns to external space, using fingers and hands, salt and pepper shakers, the proverbial napkin, and, especially, paper. Sketches, diagrams, charts, models, and other externalizations of the workings of the mind serve many roles in thinking. They support memory, information processing, inferences, and discovery. They structure, reflect, and express ideas, for oneself and for others. They use elements and spatial relations in external space to represent the elements and relations of thought, literally or metaphorically (e. g. [1] and [2]). No wonder that they are so widely used.

One arena where diagrams and sketches have proven to be especially useful is design, particularly architectural design, documented as far back as ancient Egypt, where one of the temples in Luxor has a plan inscribed on one of its walls. For architectural design, sketches can simply reduce physical scale, demanding relatively simple and straightforward transformations for comprehension and inference. However, architectural sketches typically reduce not only scale but also detail. Ideally, they take out irrelevant detail and retain and emphasize the detail pertinent to design. Effective sketches, then, simplify the information to essentials.

Diagrams and sketches are typically richer and more complex than simple Euclidean shrinking and information reduction. In many cases, they are meant to represent more than just physical structure. Extending diagrams from representing structural information to representing functional or abstract information often requires the addition of symbolic conventions: lines, boxes, arrows, words, and more (e. g. [3]). This information can be ambiguous; arrows, for example, can indicate causation, sequence, or flow, among many other meanings (cf. [4]).

Sketches and diagrams, then, must be interpreted in order to be used. The Euclidean character and the metric properties of diagrams—distances, angles, sizes, shapes and their proximity — are difficult to ignore, even when irrelevant, and can encourage false inferences. Although diagrams and sketches present information in parallel and do not privilege any direction or location over any other, the mind does not process them in parallel; rather, they are interpreted sequentially. When there is a natural beginning, diagrams are "read" from there, but when there is not a natural beginning, diagrams tend to be scanned in reading order, in Western languages from left to right and top to bottom (e. g. [5]). The richness and complexity of diagrams render them more useful but also more problematic at the same time. Although those experienced in using and interpreting diagrams often think they are obvious because the information is "there," novices often need to learn to use and interpret the information that is "there" (cf. [6]).

Information systems infrastructure consists of physical components such as computers and routers connected together by and cables or radios. Because information systems depend on complex relations between large numbers of components, their design lends itself to sketches and diagrams. In contrast to the design of buildings and products, the design of information systems typically happens at a high level of abstraction. Buildings and products are structures visible in space. But, although an information system includes visible objects such as computers and cables, those physical objects are not the crucial components of design. At the core of an information system is a set of instructions. The set is sometimes subdivided, and sometimes copied, into multiple bundles of instructions – the programs. These programs are then installed on computers that are often distant from each other. The most important, that is, functional, aspects of the system are not visible. By looking at these instructions expressed as text, we cannot at first glance tell much about the way the system was constructed – not as clearly as we can understand a building by looking at its beams and columns.

There is structure to an information system, but it cannot be fully expressed in terms of Euclidean spatial relationships. Instead, it is about electronic connections. At the most basic level, the physical network is set of wires and computers connected together. Since electronic signals travel at the speed of light, the difference in communication performance between systems sitting next to each other and systems separated by miles is often negligible. Distance matters little. What matters more is the number of hops a message takes. The hops are the transfer points – much like the transfer points in a subway system. Because the connectivity aspects of an information network can be represented with various types of abstract graphs, we speak of *network topologies* [7].

The designer of an information system is expected to understand such network topologies. This understanding presupposes an ability to read and generate systems diagrams – diagrams that document the topology. These diagrams are conventional – there is an agreed-upon shorthand for representing networks.

Systems diagrams are used to plan the flow of information, much as city maps are used to plan the flow of people. But the constraints of physical and information systems are different, and this is reflected in systems diagrams. For example, ordinal and interval relations in the representations of systems components are often irrelevant. What is important are the links, and they are hierarchically organized in subtle ways. For example, at the infrastructure level, network bridges and routers are used to partition networks into subnetworks so that performance or security can be controlled in a fine-grain fashion.

In order to successfully create and interpret systems diagrams, students and practitioners must learn to suppress conscious or unconscious inferences based on Euclidean properties of diagrams, such as the planar distances among nodes, and learn to rely primarily on graphs: drawings of nodes and edges. How well do novices and experts understand these conventions and the formal structure underlying them? How able are they to interpret and generate the paths that a topology implies?

Here, these questions are addressed through design problems given students in a Master's level course on systems design at the beginning and at the end of the class. This approach is inspired by Simon's path-breaking work on *science of design* [8], especially his observations that diagrams are crucial to design and that experts use diagrams differently from novices [1], [9]. For example, it has been established that experts tend to use higher-level knowledge units compared to novices [9], and that experts tend to organize problems according to their underlying structure rather than based on surface similarity [10]. Similarly, experts are better able to make functional inferences from diagrams than novices [11].

To summarize, diagrams are commonly used to expand the mind by putting some of its contents into the world where they can be more easily examined, interpreted, and manipulated. Design is one area where sketches are particularly apt and broadly used. Diagrams are essential to systems design, so studying how they are produced and understood is important in its own right. Although information systems are instantiated by physical objects, the array of physical objects does not adequately capture the structure of information flow. Flow is conveyed through conventional use of lines, rather than Euclidean properties such as proximity. Thus, students in systems design must overcome habitual spatial interpretation practices and learn new graphical ones. This is the main concern of the present paper. Because systems diagrams are representative of diagrams in other disciplines, the results should have broader implications as well. For example, topological diagrams are used in the design of electrical plants, transportation systems, supply chains, and systems biology. Therefore, ways of better explaining or teaching these diagrams may assist those who come into contact with these fields. In addition, the analysis techniques discussed here might be used to evaluate diagram understanding of many sorts.

Related Work in Information Systems Design

There are good reasons to study the design of information systems. Recently, there has been a call to reinvigorate the science of design [12], [13]. One reason for this call is dissatisfaction with the progress of software design. While computer hardware has essentially doubled in complexity every 18 months for the past few decades, software gains have been much more modest. Many large software projects are never completed, and those that are completed are often bug-ridden [14].

In the computing disciplines, diagram use is common [15]. Practicing systems designers tend to use the diagrams that are defined as a standard in the Unified Modeling Language (UML) [16]. These diagrams are all topological – meaning they are all variations on graphs. Studies have looked at how these graphs are used (e.g. [17]). But to our knowledge, no one has looked at how well these graphs are understood.

There is reason to study this. Just as manipulating the structure of walls changes the design of a building, manipulating the structure of a network changes the design of an information system. Many of the goals of an information system – for example, reliability, performance, security, adaptability – are directly affected by the structure of the underlying network. We know that an ability to transform figures through rotation is important for architects and engineers. But information systems are built in a topological space, not a Euclidean one. Thus, it is possible that the ability to transform figures through *topological* operations is especially important for information systems designers. Moreover, it is possible that these skills, mental transformation of geometric structures and mental transformation of topological structure, are related.

Study 1: Understanding and Producing Network Topologies

In order to understand how expert and novice students produce and understand systems diagrams, we presented design problems to students in a Master's level class in the design of systems at the beginning of the semester (Study 1) and at the end of the semester (Study 2). These students represented a wide range of initial experience – some were full time students with no work experience, while others were part-time students who worked in the information technology departments of corporations. The course asked students to engage in a series of design exercises in which they produce systems diagrams and participate in critiques of the diagrams: for example, students are asked to design a personal information system and to present diagrams representing the temporal behavior and structure of the system [18]. Students had been exposed to the commonly used diagrams of information systems during introductory courses.

The general expectation was that students' productions and comprehension of diagrams would exhibit two biases: a sequential bias and a reading order bias. Because diagrams are presented in Euclidean space, students would be biased toward Euclidean interpretations of diagrams based on proximity and thus have difficulties comprehending the topological relations in systems diagrams, indicated by lines. Specifically, they would experience difficulties making interpretations based on connectivity rather than proximity. In particular, we expected students to have difficulties comprehending and producing a logical *bus*. A bus is a sub-cluster of components that are mutually interconnected. Most local area networks are organized in such a way. By convention, busses are indicated by a line with satellite lines as in Fig. 1c. Within a bus, all nodes are interconnected to all other nodes, even though those connections are not directly shown. In contrast, Fig. 1a directly shows the connections. Fig. 1b shows a hub and spokes model, which is in how modern buildings are wired – lines run from each computer to a hub; the hub insures that everyone can easily connect to everyone else.

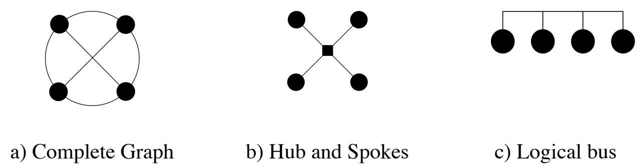
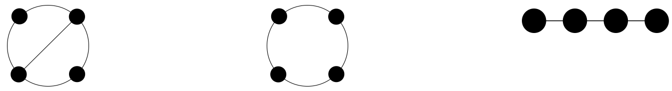


Fig. 1. Alternative LAN representations

All three of these diagrams represent the same logical topology, in that all nodes can directly connect to all other nodes.

Presented with the bus diagram (Fig. 1c), a student without knowledge of the bus diagram conventions might wrongly infer that a path between the far left node and the far right node must pass through the middle nodes. In fact, the two extreme nodes can connect directly. On the other hand, a student who misunderstands modern local area network technology may portray it using one of the diagrams shown in Fig. 2. These diagrams are inappropriate or incorrect representations.



a) Incomplete Graph   b) Ring   c) Chain

Fig. 2. Obsolete or incorrect LAN representations

A second expectation was that students would ordinarily inspect and interpret diagrams in reading order, which would bias them to see paths more compatible with reading order and to miss paths less compatible with reading order.

Methods, design, and predictions

Sixty-eight students from four different sections of the same course were presented with a set of four design problems to be answered in class (see Fig. 3). For problems 1 through 3, they were presented with a diagram of the configuration of a system and asked to generate all the shortest paths of information through the system. This is an important type of inference in systems design because it is a check that information flows according to system constraints. For problem 4, they were presented with a description of the configuration of a system and asked to produce a diagram of it then generate the set of shortest paths.

Problems 1, 2, and 4 had the same system configuration – the same topology - but with a different embedding on the plane. That is, the diagrams in 1 and 2 look different on the surface, but have isomorphic topologies. Thus, if students are able to ignore the spatial array of components and respond only to the connectivity relations, their answers to the two questions should be identical. Problem 4 was also isomorphic to problems 1 and 2. Problems 1, 2, and 4 describe two different networks joined by a bridging node (M, S, and B respectively). One network contains 4 nodes, and the other overlapping network contains 2 nodes (C and M, X and S, E and B). In networks, the bridging function is important, as it allows for the partitioning of networks into modular, manageable entities – indeed, the Internet can be thought of as such a hierarchical partitioning on a large scale. Students were expected to realize that all shortest paths with C as a terminal in problem 1 need to go through M – but that the shortest paths to Y from B, M, and C do not go through R. This presupposes they understand the diagramming convention, shortest paths, and bridges.

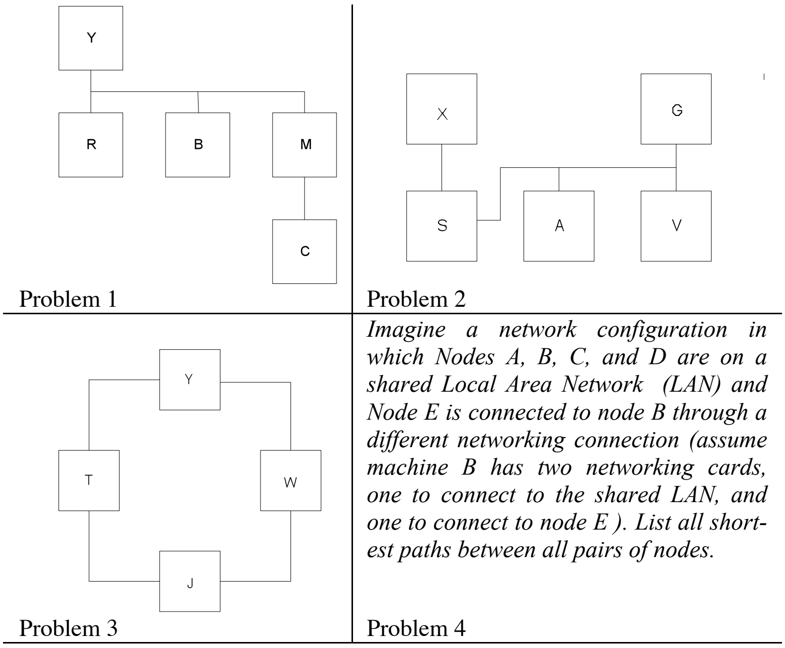


Fig. 3. The topology test questions

Students were given an example, and then asked to enumerate all shortest paths between the nodes in the graph. For problems 1, 2, and 4, there are twenty paths (Problem 1: YR, YB, YM, YMC, RY, BY, MY, CMY, RB, RM, RMC, BR, BY, CMR, BM, BMC, MB, CMB, MC, CM). For problem 3, there are sixteen paths (YT, YTJ, YW, YWJ, WY, WJ, WYT, WJT, JT, JW, JTY, JWY, TJ, TY, TYW, TJW). Students were instructed that all shortest paths should be enumerated.

For the problems requiring generation of shortest paths from pre-existing diagrams - problems 1, 2, and 3 - only two types of errors could occur. Students could list paths that were not shortest paths; these are errors of commission. Commission errors are a consequence of not understanding the topology represented by the bus convention. For example, in problem 1, listing YRBMC as a path is a commission error – the shortest path between Y and C is YMC. Thus, any commission errors are a consequence of failing to understand the essential concepts taught in the class. An error such YRBMC would suggest the students are reading the diagrams sequentially – and superficially - rather than responding to the deep structure of the diagram. This error is understandable, in that the spatial arrangement on the diagram – B is between R and M – may suggest to some that paths will include intervening nodes.

The second type of error is an omission error, failing to list one or more shortest paths. If students generate paths in reading order, starting from the upper left and proceeding left to right and top to bottom, then they should be more likely to omit backwards paths than forwards paths, where forwards means starting upper left and backwards means starting lower right. For problem 3, reading-order expectation is more counter-clockwise omissions than clock-wise.

In the case of problem 4, where students are asked to produce a diagram for a system containing a bus, and then enumerate the paths, the diagram itself should be diagnostic. The production of chains or rings suggests a sequential bias. Organizing the elements from left to right in the diagrams according to order of mention in the statement of the problem constitutes evidence for the reading order bias. For the enumeration of paths, the predictions here are the same as for the first three problems. In addition, because the diagram-generation task provides two opportunities for error, in translating the text to diagram and in interpreting the diagram, more shortest path errors (commission errors) were expected in problem 4 than in the equivalent problems 1 and 2. Specifically, for problem 4, using the appropriate diagrammatic representation, as shown in Fig. 4, should lead to better results.

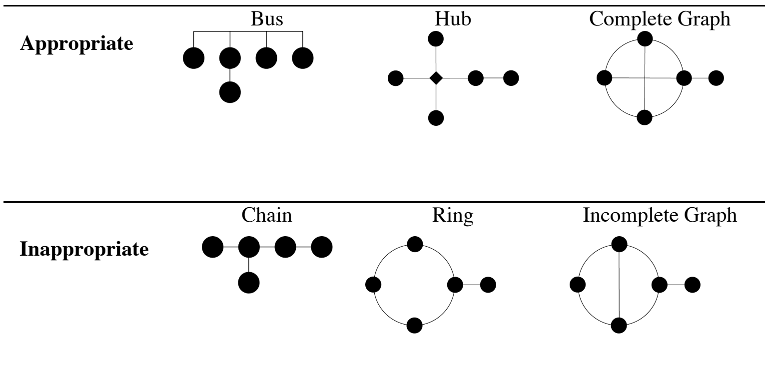


Fig. 4. Several possible diagram types for problem 4

Results

Problem solutions were coded for type of errors (omissions, commissions), and solution strategy. For problem 4 only, the type of diagram as shown in Fig. 4 was also coded.

The *reading order bias* predicts that students should list more paths starting from the upper left than starting from the lower right for problems 1 and 2; this translates into more forward paths than backwards paths. Some students, however, may have presupposed reversibility of paths; that is, they may have intentionally only listed forward paths and presupposed that each of them could be reversed to constitute a backwards path. Therefore, students who listed only forward paths for a question were eliminated from analyses of the reading order bias (this eliminated 2, 5, 4 and 1 student, respectively, for Problems 1-4). A dependent-groups t test revealed that there were more backwards omissions than forward omissions for both problem 1 ($\bar{X} = 1.64$, $s = 2.24$ for forward omissions, $\bar{X} = 2.11$, $s = 2.71$ for backwards omissions; $t(65) = -2.98$, $p = .004$), and problem 2 ($\bar{X} = 1.60$, $s = 2.49$ for forward omissions, $\bar{X} = 2.02$, $s = 2.87$ for backwards omissions; $t(62) = -2.54$, $p = .014$).

For problem 3, the reading order bias predicts the listing of more clockwise paths than counterclockwise paths. Although there were more counterclockwise omissions ($\bar{X}=1.08$, $s=1.61$) than clockwise omissions ($\bar{X} = .95$, $s=1.73$), this effect did not reach significance ($t(63) = .668$; $p = .506$). For problem 4, a reading order bias could be tested for those students who produced their own diagrams, by coding directionality of paths using the same left-right, top-bottom precedence rule. For problem 4, there were again more backwards omissions than forward omissions ($\bar{X} = 2.06$, $s = 2.49$ for forward omissions, $\bar{X} = 2.84$, $s = 2.99$ for backwards omissions; $t(64) = -3.38$, $p = .001$).

The *sequential bias* predicts that students should make commission errors which introduce extraneous nodes. That is, they should list paths that are either not necessary (for example, listing YRBMC) or not possible, for example listing YC, when YMC is correct because the bridge node M needs to be included. They could combine these errors, for example listing YRC. The first type of error, the introduction of an extraneous node, accounted for 93.7% of combined 298 commission errors in questions 1 and 2. Thus, the vast majority of commission errors are consistent with a sequential bias. The second type of error, the omission of the bridge node in a path crossing the bridge, accounted for only 2% of the errors, and omission of the bridge node combined with an extraneous node accounted for another 1.7% of the time. There were other paths that fell into no obvious category – such as the inclusion of a node from the previous diagram, or single node paths – and these occurred 2.7% of the time. Thus, bridges were only omitted 3.7% of the time. For the most part, students did understand that information had to travel through the bridge node. Students could choose nodes in any order in the commission errors. But the paths in general proceeded either forwards or backwards, confirming a sequential bias. Of all the commission errors in questions 1 and 2, only 14, or 4.7%, involved a change of direction – for example, BMY. Fig. 5 shows the ten most common forward direction errors and their frequencies in problem 1.

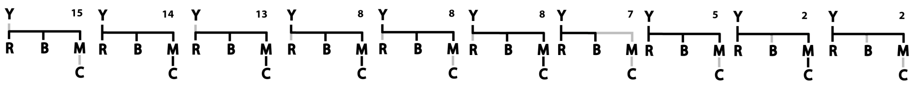


Fig. 5. Commission errors. Dark lines indicate the path; RBM, YRBMC, etc.

Because problem 4 presented two opportunities for error, in translating the text to a diagram and in generating the shortest paths (presumably from the diagram), there should be more errors on problem 4 than problems 1 and 2, even though all problems are identical in structure. Indeed, more omissions were observed for problem 4 ($\bar{X}$ = 5.45, s = 5.46) than for problem 1 ($\bar{X}$ = 4.0, s = 5.03) and problem 2 ($\bar{X}$ = 4.47, s = 5.94), and this difference is significant (F(1,66) = 7.58, p = .008). However, there were not significantly more commission errors in problem 4 ($\bar{X}$ = 2.51, s = 4.15) than for problem 1 ($\bar{X}$ = 2.26, s = 4.09) and problem 2 ($\bar{X}$ = 2.21, s = 4.21, F(1,66) = 0.45, p = .504). Problem 4 elicited a range of diagrams from students (Fig. 6 and Table 1). Five students produced no diagram while attempting to answer problem 4.

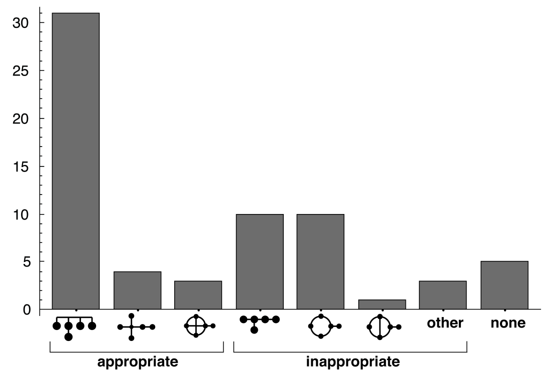


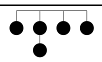
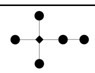
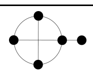
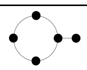
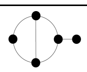
Fig. 6. Frequency of production of different diagram types for problem 4

Effectiveness of the diagram types was calculated based on number of path errors (Fig. 7). The mean number of omission errors for the use of a self-created appropriate diagram type ($\bar{X}$ = 2.84, s= 4.30) was compared to the number of omissions for use of a self-created incorrect diagram type

($\bar{X}$ = 8.71, s = 4.56) and to no use of a diagram ($\bar{X}$ = 9.60, s = 7.13). A between-subjects ANOVA showed that the use of a correct diagram resulted in significantly fewer omission errors compared to using an inappropriate diagram or no diagram ($F(1, 64) = 21.47$; $p < .001$). The two ineffective strategies - using an inappropriate diagram type and no diagram - did not differ in the number of omission errors ($F(1,64) = 0.15$, $p = .696$).

For commission errors, the mean number of errors for the use of a self-created appropriate diagram type ($\bar{X}$ = .47, s = 1.55) was compared to the use of a self-created inappropriate diagram type ($\bar{X}$ = 5.54, s = 5.19) and to no use of a diagram ($\bar{X}$ = 3.40, s = 3.28). An ANOVA showed that the use of a correct diagram resulted in significantly fewer commission errors compared to using an inappropriate diagram or no diagram ($F(1, 64) = 15.67$; $p < .001$). The two ineffective strategies - using an inappropriate diagram type and using no diagram - did not differ in the number of commission errors ($F(1,64) = 1.61$, $p = .209$).

Table 1. Frequency of production of different diagram types for problem 4

freq	type of diagram	classification	Freq	% use	mean (sd) omissions	mean (sd) commissions
31		appropriate diagram Type	38	57	2.8 (4.3)	0.5 (1.6)
4						
3						
10		inappropriate diagram Type	24	36	8.7 (4.6)	5.5 (5.2)
10						
1						
3	other					
5	none	None	5	8	9.6 (7.1)	3.4 (3.3)
67	(Total)	(Total)	67	100	5.5 (5.5)	2.5 (4.1)

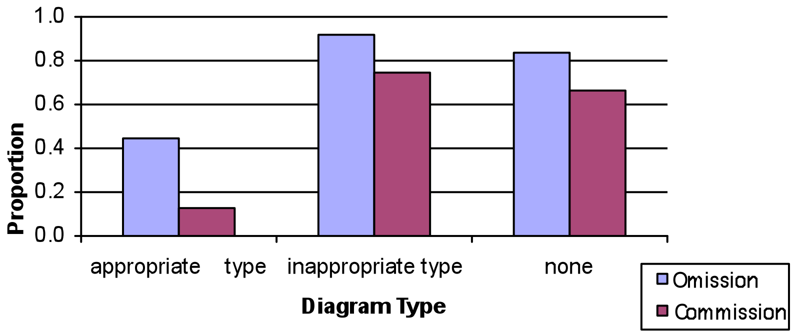


Fig. 7. Proportion of students making omission or commission errors, by appropriateness of diagram type

Discussion

Students in an introductory course in systems design were asked to solve four design problems. Three involved making inferences from a supplied diagram; a fourth entailed creating a diagram and making inferences from the created diagram. Two of the diagram problems (1 and 2) and the text-to-diagram problem (4) required understanding network buses and network bridging – that is, understanding the topological structure of the problem and the conventions used in diagrams to represent such structures. Students were expected to have difficulties interpreting and creating these kinds of diagrams. Specifically, they were expected to exhibit two biases, a sequential bias and a reading-order bias.

The sequential bias predicts the introduction of extraneous nodes in paths. Because B is between A and C on the diagram, students include B in the listed path – even though the bus convention is meant to convey that in the network topology, A connects directly to C. This bias showed up strongly in our data set. The sequential bias also predicts that only commission errors introducing extraneous nodes in their surface sequential order will occur: e.g. RBMC but not BRMC in problem 1.

What might explain these types of errors? Because we live in a Euclidean world, we may tend to make Euclidean inferences, based on the proximity of objects. More specifically, we may read the diagram imagining that we are traversing the lines of the diagrams as if they were paths.

Because diagrams are too complex to be comprehended as wholes, they must be examined in sequence. The default sequence is reading-order; this bias predicts more omissions for inferences that don't correspond to reading order. This prediction, too, was borne out by the data. When students were asked to generate a diagram as well as generating the set of shortest paths, they used their diagram to generate the paths. That is, the type of diagram students generated predicted the errors they made.

The results indicate that diagrams are useful and actually used in making inferences in systems design. The data further indicate that students have difficulties interpreting diagrams, especially when the diagrams portray a topological space that does not exactly correspond to Euclidean intuitions. Can classroom instruction improve performance? The second study addresses this question.

Study 2: Posttest Generating Network Topologies

Methods

Late in the course, 35 Master's level students from two sections of the design course were asked to solve Problem 4 a second time (the "posttest"). Thus, the posttest data was available from two out of the four sections that participated in the pretest. The posttest was coded identically to the pretest version of problem 4, and the results were compared. The expectation was improved diagrams and improved inferences as a consequence of the classroom instruction and exercises.

Results

Although students did make fewer omission errors in the posttest ($\bar{X}$ = 3.85, s = 5.02) than in the pretest ($\bar{X}$ = 4.48, s = 4.90), indicating decreased reading order bias, this difference did not reach significance (t(32) = .73, p = .470). Subjects made fewer commission errors, indicating decreased sequential bias, in the posttest ($\bar{X}$ = .97, s = 2.44) than in the pretest ($\bar{X}$ = 2.52, s = 3.80) and this difference was significant (t(32) = 2.11, p = .042). Furthermore, of the 33 students who participated in both tests, 12 students (36%) gave a fully correct answer in the pretest, while 18 (55%) gave a fully correct answer in the posttest. The reduction in commission errors demonstrates increased understanding of network topology.

Table 2 shows the frequency of producing bus, other, and no diagrams on the pretest and posttest. Those who used a bus on the pretest either used it again on the posttest or used no diagram on the posttest. Impressively, 13 of the 18 students who failed to use a bus on the pretest used a bus at post-test. This increase in use of the bus is marginally significant (χ^2 (1)= 3.02; p = 0.082).

Table 2. Diagram Type Constructed for Problem 4

		Posttest			
		bus	**inappropriate**	**none**	**Total**
Pretest	Bus	11	0	4	15
	other appropriate	4	1	1	6
	inappropriate	9	3	0	12
	None	0	0	0	0
	Total	24	4	5	N = 33

Do students who produce better diagrams also produce better solutions? Table 3 shows the effectiveness of the diagram types for promoting correct inferences. Fourteen out of the 26 students who drew the conventional bus diagram got the problem correct, while only one out of four students who tried to use another type of diagram did so. A chi-square test of independence for type of diagram used (bus versus other graph versus no graph) and solution correctness was conducted. The results were marginally significant ($\chi^2(2)=5.55$; $p = 0.062$). These results suggest that learning to choose the right diagram convention is key to solving the problem.

Table 3. Posttest Problem 4: Use of appropriate (bus) and inappropriate diagram types, with solution correctness.

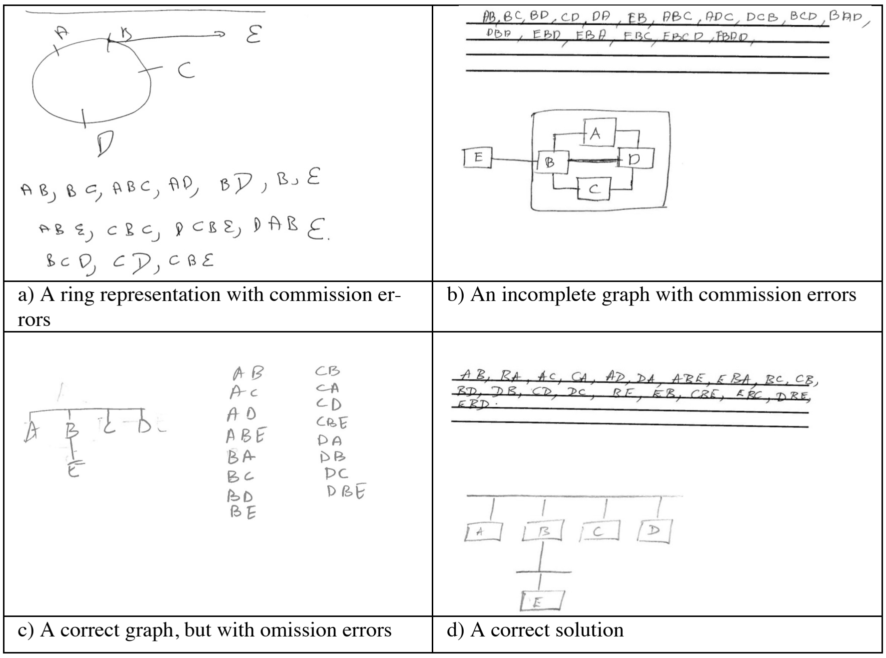
Diagram type	Correct answer	Incorrect answer	Proportion Correct	N
Bus	14	12	.54	26
inappropriate	1	3	.25	4
none	5	0	1.00	5
(Total)	20	15	.57	35

In sharp contrast to the pretest, in the posttest, the five students who did not draw any diagram all got the problem correct; all five had produced correct diagrams on the pretest. This finding is intriguing, but not unprecedented. Often, it is novices who benefit or need diagrams, while experts can solve the problems without that support (e. g. [19], [20]). A chi-square test of independence for an association between the use (or no use) of a diagram and solution correctness on the post-test was significant ($\chi^2(1)= 4.375$; $p = 0.036$). These students had entered the course with a high level of proficiency, and their proficiency appears to have increased to the point that they no longer needed the diagrams.

Students' diagrams provide feedback to instructors

As we have seen, diagrams can be useful to students in problem solving. Student-produced diagrams also give valuable feedback to instructors. Producing a diagram encourages extracting the essence of a problem and representing it completely. Conceptions and misconceptions may be more evident in students' diagrams than in their verbal responses.

For example, Fig. 8a shows a drawing of the network as a ring rather than as a bus. Using this configuration, the student commits commission errors – in a bus-based LAN, there is no need for a path such as DCBE.



a) A ring representation with commission errors

b) An incomplete graph with commission errors

c) A correct graph, but with omission errors

d) A correct solution

Fig. 8. Examples of post-test diagrams

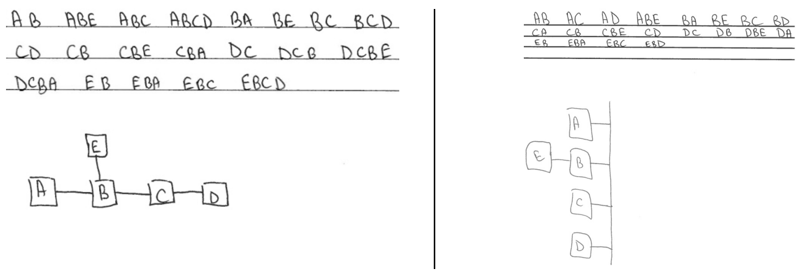


Fig. 9. A student's pretest and posttest, showing improvement.

Fig. 8b would have been topologically correct if there had been a link from A to C. Because of the missing line, commission errors are made – ADC rather than AC, for example. Fig. 8c has no commission errors. However, there are omission errors – the student has forgotten links back from node E. This was the most common type of error on the posttest.

Fig. 9 shows one student's pre- and posttest solutions. The pretest had an incorrect diagram and errors; the posttest diagram was fully correct. While this degree of improvement did not happen as often as we had hoped, students got much better at drawing the diagrams. This led to fewer commission errors overall. But the omission errors persisted.

Discussion

Students in a systems design class were asked to generate diagrams to solve design problems that included a logical bus topology early in the semester and late in the semester. Compared to the pretest, in the posttest more students produced diagrams that represented a bus, and more students were able to correctly produce all shortest paths and fewer paths that were not shortest. On the whole, the students who produced more satisfactory diagrams also produced better solutions. A subset of students was able to produce the correct and only the correct paths without a diagram at post-test, though these students all produced a diagram earlier in the semester, evidence that experts often no longer need an external representation to solve a problem. Presumably, experts have mentally unitized the problem so that it requires less working memory capacity, hence less need of an external representation.

Nevertheless, even at the end of the semester, there was evidence for both biases: the sequential bias, indicated by commission errors that exhibit a lack of understanding of topological space, and the reading order bias, indicated by omission errors, especially backwards omission errors. If students fully understood the structure of the system, they would not have made either error. They would not make commission errors because they would understand that all nodes on a bus are directly accessible, and they would not make omission errors because they would know how to generate and check a complete set of paths. Commission errors decreased - suggesting that most students did master topological concepts. Omission errors remained fairly constant – suggesting that the reading order bias needs to be counteracted with a different form of instruction. In the bigger picture, overcoming commission errors is more important – topological errors may lead to a drastic misestimation of a system's performance.

Conclusions

Sketches and diagrams are an essential component of design of information systems. Systems are often large, so they overload limited capacity working memory, a problem solved by externalizing the structure (and perhaps function) of a system by committing it to paper. An external representation serves as a basis for inferences and a basis for generating new designs. Sketches and diagrams are abstractions, and the successful ones select and emphasize the correct information while omitting information that is distracting. Diagrams and sketches facilitate inferences by capitalizing on their physical features, such as proximity, angle, and connectivity. They foster creativity by enabling alternatives, expansions, reductions, revisions. Sketches and diagrams are especially appropriate for design, as they can capture complex relations among parts and wholes.

All these virtues and more depend on successful reading and interpretation of diagrams and sketches, skills that depend on expertise. Even "realistic" undoctored photographs carry information that novices may not readily detect; examples include surveillance photographs and X-rays. Because diagrams and sketches are such common artifacts, the need for expertise in their use is not always recognized. Reading and interpreting diagrams are affected by habits and biases from reading and interpreting the visual world and other common external representations, such as maps. These habitual ways of interacting can lead to failures and to errors.

Here, we studied sketches produced by students of systems design in the service of problem solving, making inferences from the sketches. The task given students, generating the set of shortest information paths from a specific configuration, is in some ways similar to finding routes on a map. It differs crucially from a map in that the systems contain a logical bus, a set of links that are mutually connected. From any node on the bus there is a direct connection to any other node on the bus. The graphic convention for the bus shows elements attached to a line, and the convention causes difficulties for students. In the present experiments, students exhibited the difficulties by generating paths that include unneeded nodes – errors of commission. Because these extra nodes are virtually always listed in the order that they appear along the path connecting the endpoints, we call this bias a *sequential* bias. The bias is a tendency to assume that all nodes passed in the scanning sequence must join the path.

The second bias exhibited by students was a *reading order* bias. Students tended to generate paths in the canonical reading order for European languages, from upper left to lower right. They often failed to generate all the correct paths (errors of omission), and there were more backwards omissions than forward omissions.

Diagrams can be used to overcome the sequential bias; in a previous study a well-crafted diagram helped students understand that the two middle processes in a four-process system could be done in either order [21]. For the more complex problems used here, instruction over the course of a semester helped many students to master the concept of a logical bus and then to both diagram it appropriately and use it to make correct inferences.

While the logical bus is the convention used most often in industry, it causes confusion among students. In all fields, communications get abbreviated with use, and the abbreviations simplify and even distort some of the information. This can cause difficulties for the uninitiated, as it did here. Perhaps students need to work through simpler forms of connectivity and clearer forms of diagrams before they understand the current conventions in a field (the bus in this case). Indeed, physical devices corresponding to chains and rings have been used in network design and construction, but their usage has been eclipsed by fully connected topologies, which offer greater flexibility and reliability. Each of these five representations has been used in systems design in the past, and each was produced by at least one student in the present course. So the range of student performance exhibited here spans the development of design conventions, an instance of ontogeny recapitulating phylogeny, common in other domains (e. g. [2]). For these reasons, it may make sense to teach students these concepts, their instantiations, and their equivalences in roughly the order in which they have evolved, leaving the most densely coded conventions for last. Because diagrams in other domains are vulnerable to similar difficulties and misconceptions, this practice, of scaffolding changing diagrams and changing conceptual structures on each other, may have wide applicability.

Acknowledgements

The authors are grateful for the support of the National Science Foundation under grant IIS-0725223 as well as to NSF REC-0440103 and the Stanford Regional Visualization and Analysis Center.

References

1. Larkin JH, Simon H (1987) Why a diagram is (sometimes) worth ten thousand words? Cognitive Science 11: 65–99

2. Tversky B (2001) Spatial schemas in depictions. In M Gattis (ed), Spatial schemas and abstract thought, MIT Press, Cambridge, MA
3. Tversky B, Zacks J, Lee PU, Heiser J (2000) Lines, blobs, crosses, and arrows: Diagrammatic communication with schematic figures. In M Anderson, P Cheng, V Haarslev (eds), Theory and application of diagrams, Springer, Berlin
4. Nickerson JV (2005) The meaning of arrows: Diagrams and other facets in system sciences literature. 38th Hawaii Int. Conference on System Sciences
5. Taylor HA, Tversky B (1992) Descriptions and depictions of environments. Memory and Cognition 20: 483-496
6. Committee on Support for Thinking Spatially (2006) Learning to think spatially. The National Academies Press, Washington, DC
7. Peterson LL, Davie BS (2007) Computer networks: A systems approach. Morgan Kaufmann, Amsterdam
8. Simon HA (1969) The sciences of the artificial. MIT Press, Cambridge
9. Larkin JH, McDermott J, Simon DP, Simon HA (1980) Expert and novice performance in solving physics problems. Science 208: 1335-1342
10. Chi M, Feltovich P, Glaser R (1981) Categorization and representation of physics problems by experts and novices. Cognitive Science 8: 121-152
11. Suwa M, Tversky B (1997) What do architects and students perceive in their design sketches? A protocol analysis. Design Studies 18: 385-403
12. Hevner AR, March ST, Park J, Ram S (2004) Design science in information research. MIS Quarterly 28: 75-105
13. March ST, Smith GF (1995) Design and natural science research on information technology. Decision Support Systems 15: 251-266
14. Charete RN (2005) Why software fails. IEEE Spectrum 42: 42-49
15. Harel D (1988) On visual formalisms, CACM 31: 514-530
16. Fowler M (2004) UML distilled: A brief guide to the standard object modeling language. Addison-Wesley, Reading, MA
17. Lange CFJ, Chaudron MRV, Muskens J (2006) In practice: UML software architecture and design description. Software IEEE 23: 40-46
18. Nickerson JV (2006) Teaching the integration of information systems technologies. IEEE Transactions on Education 49: 1-7
19. Corter JE, Zahner DC (2007) Use of external visual representations in probability problem solving. Statistics Education Research Journal 6: 22–50
20. Heiser J, Tversky B, Silverman M (2004) Sketches for and from collaboration,. In JS Gero, B Tversky, T Knight (eds), Visual and spatial reasoning in design III, Key Centre of Design Computing and Cognition, Sydney
21. Glenberg AM, Langston WE (1992) Comprehension of illustrated text: Pictures help to build mental models. J. of Memory and Language 31: 129–151