23. Intra-Medium and Inter-Media Constraints

Stephen M. Ervin

Assistant Professor
Department of Landscape Architecture
Harvard Graduate School of Design
Cambridge, Massachusetts USA 02138

Designers work with multiple representations in a variety of media to express and explore different kinds of knowledge. The advantages of multi-media in design are well-known, and exemplified by the current interest in 'hyper-media' approaches to knowledge exploration. A principal activity in working between views in one medium (e.g. plan, section and perspective drawings), or between different representations (diagrams, maps, graphs, pictures, e.g.) is extrapolating decisions made in one view or medium over to others, so that some consistency is maintained, and implications can be explored. The former kind of consistency maintenance (intra-medium) is beginning to be well understood - techniques for constraint expression, satisfaction and propagation are starting to appear in 'smart CAD' systems. The latter kind of consistency maintenance (inter-media) is different, less well understood, and will require new mechanisms for constraint management and exploration. Experiments, hypotheses, and solutions in this direction will be central to any effort that seeks to explain, emulate or assist the integrative, synthetic reasoning that characterizes environmental design and planning. This paper examines some of the characteristics and advantages of intra- and inter-media constraint exploration, describes a prototype 'designers' workstation' and some experiments in the context of landscape planning and design, and lays out some directions for development of these ideas in future computer aided design systems.

Introduction -- Constraints among Multiple Representations

Designers work in a variety of media to express and explore different kinds of knowledge. A principal activity in working between views in one medium (e.g. plan, section and perspective drawings), or between different media (diagrams, maps, graphs, pictures, e.g.) is extrapolating decisions made in one view or medium over to others, so that consistency of representation is maintained, and the many implications of any decision are explored. In practice, this consistency cannot be absolute, and the exploration of implications cannot be exhaustive. Designers depend on the conventions and organization of the design-build community enterprise to identify errors and inconsistencies that make it through the design stage. Similarly, designers are often surprised (pleasantly or not) by some unanticipated effect in a built environment. No effort in computational support or artificial intelligence applied to design will likely overcome the complexity and consistency problem of any real-world design problem, especially at the scale of, say landscape planning or urban design. Nor should any such effort seek to limit the possibility for fortuitous delight and discovery. We might well want, however, to improve the chances of error detection and extend the range of possible design explorations. Computer assisted design in its proper sense, seeks to assist, not replace, designers. To that end, we are concerned with understanding the processes of designing, and the forms and dimensions of support required.

In this regard, the general domain of environmental design offers a rich field for exploration and discovery. Problems are often large and complex, solutions are varied, some rules are clear and constant, others as ephemeral as the tracing paper on which they are expressed. One obvious characteristic of such design, whether in practice, education or research, is the heavy reliance on multiple forms of representation, in multiple media. Designers and planners use text, reports, charts, diagrams, maps, plans, details, sketches,photographs, movies, computer simulations and animations, and more, to explore their problems and tell their stories. This paper is based on the simple observation that designers and planners use multiple representations, and explore constraints among them, all the time. Why not provide computational support for these processes?

Graphic Knowledge Representation: Diagrams, Maps, Graphs and Pictures

Multi-media technologies for designers are increasingly available, and their appeal is exemplified by the current interest in 'hyper-media' approaches across disciplines [Delisle, Jonassen, McAleese, Nielsen]. The hyper-media movement, which builds on the original ideas of hyper-text [Nelson 1984], multi-media technology [Katzen] and database theory [Date], seeks to integrate representations and media in an interactive environment, linking sounds, images, text and numbers with a computer-controlled interface. The principal idea is 'linkage' between items of information. The nature of these links is the subject of active speculation and research, ranging from simple 'buttons' providing static cross-reference to operating systems that support 'hot-links' between applications. The increasing availability of large-volume storage devices (CD-ROM, optical disks, video disks, e.g.), revolutions in signal compression, and available hardware for sound and other output, as well as multiple forms of input, have made both hardware and software developments for hyper-media attractive and possible. The technology has been developed and is in place for publishing and entertainment markets; similar environments for designers are not very far away.

Like the technological approach to computer aided design, it is possible to approach multi-media as a problem of management and hardware. The proliferation and incompatibility of standards, capabilities, new directions and technological improvements make this side of the question a full time job. The concern for researchers in design theory, however, is an alternative question: How do media and forms of representation enter into and effect design reasoning; how are they incorporated into or reflective of, design knowledge? Advances along all technological dimensions are bound to continue to occur exponentially. As designers and design researchers we are confronted with the much more difficult problem: what to do with all the technology?

Choosing representation media is not just a cosmetic choice. Representations are value-laden; every drawing, indeed every CAD system, contains a set of assumptions, biases and normative positions. Knowing as we do that even 'photo-realism' is an abstraction, we must choose representations for rhetorical purposes, to support design arguments, to display design knowledge. The push in computer graphics to achieve ever higher numbers of pixels, finer resolution, truer color, more realistic radiosity and illumination models, is a technically challenging, but epistemologically empty pursuit. In design practice, it is often the case that the simpler the drawing, the richer the design reasoning. Simple line drawings tend to be more full of 'propositional' content, which can be argued about; 'pictorial' representations leave little room for argument, other than 'I (don't) like what I see'. Diagrams and back-of-the-envelope sketches are the chosen media during early design stages, full of content and complexity; perspective renderings come at the more sterile presentation stage. Diagrams and sketches are made for real communication, with ourselves and others; renderings are for final adjustments, and sales pitches.

Representation -- graphic and otherwise -- in the service of reasoning has been explored by art historians, psychologists, and AI researchers [Arnheim, Bertin, Larkin & Simon, Newell & Simon, Tuftie]. One conclusion is that switching representations may often be a key to problem solving. In the case of diagrams, for example, just the act of converting from a verbal, mathematical or logical problem into one of two-dimensional space often serves to clarify relationships and to direct design development or problem solving. Knowledge representation in general is recognized as a critical component of problem-solving and AI approaches [Brachman], and is the foundation of a body of work in architecture and environmental design [Coyne et al]. Graphical representations, both as containers of and media for exploration of knowledge, are particularly important in design, and haven't received as much scrutiny as other symbolic forms.

In design practice, (at least) four essentially different kinds of graphic representation can be found at different times and places, and used for different purposes. *Diagrams*, *maps*, *graphs* and *pictures*, all instances of two-dimensional graphical representation, are here taken as distinct media, both for exploration and communication. The underlying data, in form and content, and the graphic representation (ditto) are different and somewhat incompatible from one medium to the next. (I'll argue later on that they are <u>not</u> just different projections from a single database.) All of these media are essential to the reasoning processes – including visual inference -- in design, and all are interconnected in common use. The nature of those interconnections is a pressing research question for the next generation of CAD tools.

Diagrams

Diagrams are those schematic, abstract, and usually simple drawings used to express and explore structural relationships among parts. They are distinguished from maps by being essentially not to scale, and not true to shape [Bertin]; and from pictures by being concerned with propositional content (ideas) more than appearance. Like maps, and pictures, they are used for visual inference, so their shapes, scales and appearance are important. In the following discussion I use the vocabulary of [Ervin 1989].

'A-Diagrams' -- the most abstract ones -- are used to express high-level organizational intent. A city may be described as 'radial or 'grid-based, for example; a path as 'linear or 'branching. The use of these diagrams is often in communication, as a visual reminder of some essential property of a proposed plan. 'B-Diagrams' -- less abstract -- are perhaps more common, as they begin to express form as well as concept. Branching may be alternate or opposite, regular or irregular, hierarchical or network-like. These diagrams give rise to particular shapes and forms when plans are drawn, and their use in visual inference is to explore generic and geometric properties of the class of form, without yet reference to the particular details of the problem at hand. Such diagrams have been called 'form-diagrams' [Milne].

All diagrams have the essential characteristic of conveying, and containing, expressions of constraint which are to be maintained as design development progresses. Design alternatives may be produced within a single diagrammatic context; shifting to another diagram is a qualitatively different change, at another level of abstraction [Ervin 1987, Ligget et al.] Used in this way, diagrams are constraints on further refinement, and so on other represen-tations

The inferential purpose of diagrams is primarily for testing high-level abstract concepts (types and relationships of objects, e.g.) and beginning to explore spatial arrangement, but without commitment to size, shape, or other refinements. The cognitive processes associated with reading diagrams are more like analogy and association than the more literal readings given to other graphics, such as the three described below.

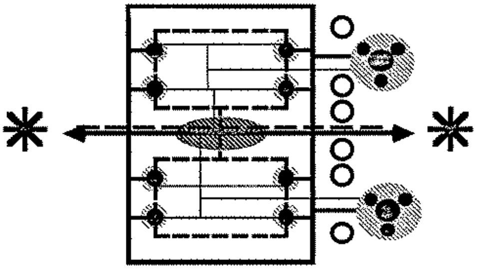


Figure 1a. A-Diagram

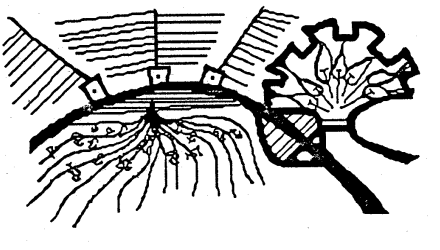


Figure 1b. B-Diagram

The production of diagrams is still largely unsupported by computer software, except as simple line-graphics in vector drawing systems. The knowledge-based production and exploration of diagrams remains a largely unexplored area deserving more research.

Maps

Maps include any <u>scaled</u> representation of some features across some region of a coordinate space, at any size from regional to backyard. I include traditional plans and sections in this category; they are just maps at different levels of resolution with different cutting planes, or points of view. Their inferential use depends largely on our ability to relate elements of one map to another through some consistent scale and coordinate system conversion, and our ability to relate from the map to the territory. Maps are of course abstractions, like other representations, but their element of scaled-reference makes them distinct.

The inferential purpose of maps (and plans etc) is primarily for testing dimensions and spatial relationships including coincidence. Their use is primarily by 'identity', that is, a one-to-one mapping is made from the representation to (some aspect of) the world, a more literal reading than the reading given to diagrams. GIS and CAD software provide ready-made tools for the production and limited analysis of maps, plans and other drawings, but little opportunity to embed design knowledge into the representations.

Graphs

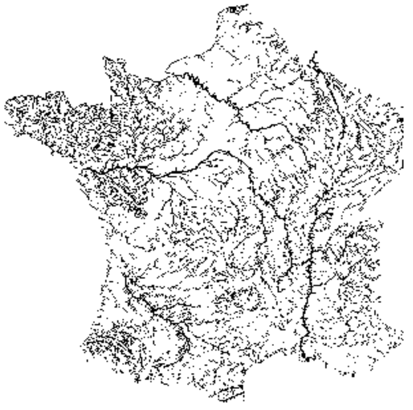
Graphs include all sorts of quantitative displays including tabular and graphic (pie charts, scatterplots, histograms, bar graphs, etc.) They are associated with data that is often represented as rows and columns (or n-dimensional arrays) of related information, and hence most commonly with spreadsheet and data-base management software. Their use in planning and design includes both quantitative comparisons between alternatives and depictions of trends. Spreadsheet software and other mathematical, statistical and quantitative modelling tools provide powerful assistance in numerical exploration, but offer little integration with other software or forms of representation except through limited specially tailored interfaces or cumbersome data interchanges.

Many models and algorithms may be included in this category since many approaches to describing relationships or interactions depend on a matrix or equivalent network representation. Economic input-output models, for example, systems of difference equations including exponential growth and decay models, and other systems describing constraints and relations among parts are conveniently represented in tabular and array form. The inputs and outputs of such models, of course, may be from maps and text, for example, and need not be limited to tables and/or graphs.

Charts and graphs are the most obvious way to present much numeric data visually; the human eye remains the most powerful pattern recognition and trend-spotting apparatus designers and planners can call upon. The inferential purpose of graphs is primarily for testing qualitative relationships among numbers -- pattern recognition by eye. The mechanism is more primitive and less 'cerebral' than the reading of maps or diagrams: recognition of clusters, slopes, edges, outliers, etc. are likely very low-level visual abilities.

Pictures

Pictures or images is the term reserved for those graphic representations whose primary characteristic is 'impression', 'expression', or 'realism'. Photographs, sketches, etc. are still abstractions, but in a sense the least abstract of the set of representations considered here. Montages and other results of image processing and 'painting' are included; these include purely digital as well as combined digital/analog representations. CAD and Image Processing software provide the typical tools for computational manipulation of pictures, but the integration of image processing into a comprehensive design workstation is an effort with little theoretical underpinning to guide it.



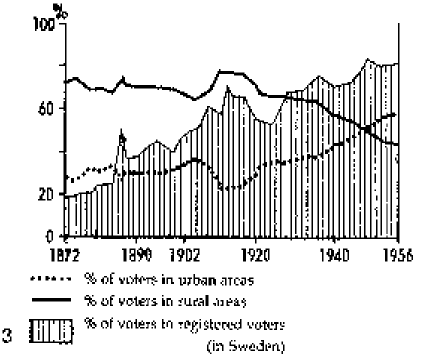


Figure 2 a) A Map, **b)** A Graph,

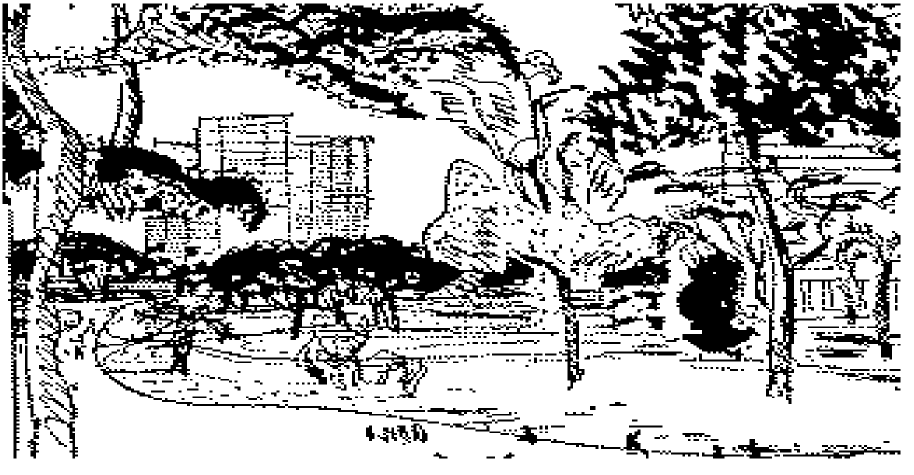


Figure 2 c) A Picture

The inferential purpose of pictures is primarily for qualitative and often inarticulate judgements -- such things as visual preference, overall form, proportion, balance, etc. These characteristics may be computable, but no designer or planner can well-enough describe her criteria to abdicate responsibility for looking. These readings may be intensely analogical in nature, or extremely literal -- visual images and their knowledge-based interpretation remain a mystery and an intriguing if formidable research area. Pictures have the added unique characteristic that they are used, and judged, both for form and substance. We are most likely to be concerned with the 'surface' characteristics as the 'inner' content of pictures when designing, often in ways that are inexplicable but real.

Constraint Management

Constraint management in design is an established research area [Fleisher et al.,Gross et al 1987, 1988] turning into an established technology. Expressing, propagating and satisfying numeric and logical constraints is beginning to be well understood, and appearing in production software, typically in mathematical, electrical and mechanical engineering fields [Robinson]. The basic formalism is to declare a variable value as falling within some constrained range, usually as some function of other variables' values. Confronted with a network of such constraints, the problem is to find some solution such that all the variables have values and a minimum number of constraints are violated (ideally, of course, none.) Optimization, numerical approximation, simulated annealing, constraint relaxation and other techniques have all been developed for dealing with the problem. Part of the appeal is the ability to explore alternatives --"What results or conflicts appear from variations in base parameters?". Constraint satisfaction systems represent next-generation spreadsheets, with features such as symbolic solvers and interval values, that serve to extend their functionality. Techniques for constraint expression, satisfaction and propagation are also starting to appear in 'smart CAD' systems [Borning, Freeman-Benson, Gross, Nelson 1985].

Designers, of course, have been engaged in 'what-if' analyses and generation of alternatives for a long time. For budgets and pro-forma analyses it is fairly clear how to identify all the elements that contribute to the bottom line; for design problems much less so. Even supposing we can identify a large number of the elements we are concerned with in a design, and identify the relationships between the elements, we are for the most part lacking computer software that can help with simulations of interactions and other aspects of graphic knowledge representation and reasoning.

Constraint satisfaction systems have been shown to be valuable in well defined engineering disciplines; they have yet to be shown effective in complex environmental design situations, although speculation and optimism can be found in the literature [Gross et al.] I present here some more speculation in the hopes of furthering the research and development of the technology in the direction of multi-media.

In general, in multi-media or multi-view systems, constraint satisfaction and propagation mechanisms should provide for automatic linkages, so that changes in one medium or view are reflected in other media or views. What the details of that 'reflection' operation might be, and how such reflections are used by designers, is the research agenda that this paper advocates. Since each medium, or representation, may serve as input or output, and since each depends on its own database(s), it is tempting to think of media as isolated and somewhat incompatible, requiring different approaches to information exploration and management. It is easy to imagine both superficial linkages and more substantive ones that multiply the power and the potential of individual representations.

Superficial linkages take the form of simple cross-reference pointers (for example, associating a chosen illustration with a particular paragraph in a report.) These 'hyper-links' provide a skeleton over which to travel between pieces of information in the system; this skeleton may be based on *a-priori* knowledge and constraints, but is bound to be modified during the processes of exploration and annotation.

More substantive links --constraints -- take the form of dynamic relations between pieces of information, controlled by explicit models developed by designers. For example, a planner may recognize a relationship between public expenditures on infrastructure and resultant private development of residential and commercial real estate. In such a model, increasing a value in a spreadsheet as part of a gross budget should result in a change in a map -- more (or less) residential development in specified parts of the region. Conversely, changes in a map of residential density should trigger revisions to a spreadsheet model predicting energy demand, or sewage production, or transportation loads. Through these dynamic links, complex models that can be viewed as maps, as graphs, or even as pictures, may be developed. Some possibilities for -- and problems with -- constrained linkages (six in all) between each of the four representations -- diagrams, maps, graphs and pictures-- are discussed below.

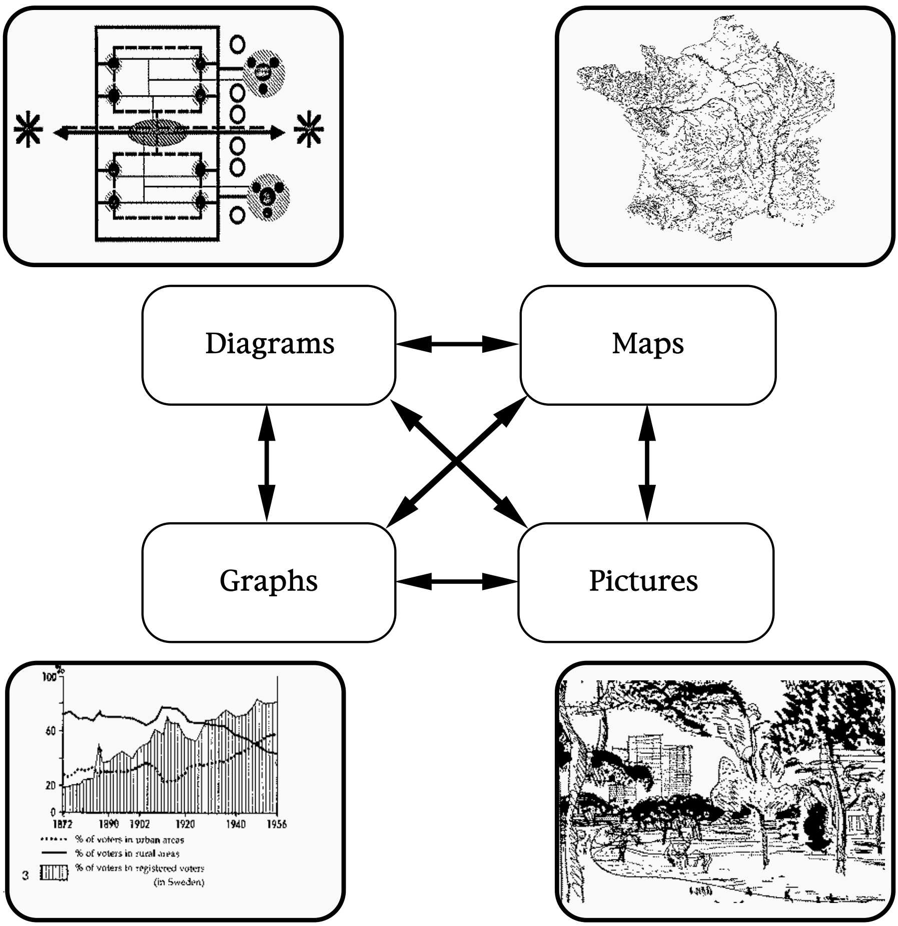


Figure 3. Inter-Media Linkages

Diagram <--> Map Linkages

The possible linkages between diagrams and maps are many. In urban design, we often start with a simple concept diagram, and seek to apply the concept to a particular site , with all its details of topography, hydrology, vegetation, etc, to produce a plan. Or, starting with a map, we may produce a diagram revealing the essential structure of some phenomena. The process of abstracting here separates form-concepts from particular topography and details of geography. Many urban design studies start with just such abstractions, simplifying the essence of a study area into a ring, or star, or grid, or 'spaghetti' structure, etc. As maps are already abstractions, it is largely a matter of degree when crossing the threshold. Many useful graphics are ambiguous in this regard -- subways line maps are typically more diagrammatic than topographic maps, but more map-like than the most abstract diagrams.

Automating the extraction of a diagram from a map, or the elaboration of a diagram into a map, is a challenge to pattern recognition techniques and knowledge based approaches. Exploring the relationship between maps and diagrams in the broadest sense -- as, for example, between a land-use diagram and an illustrative site plan -- is in many ways the essence of the early stages of design. Moving easily between these media, reconciling inconsistencies and letting insights from one medium constrain developments in the other, is a quintessential design process. It is likely that a computing environment which supports both representations from the outset, and therefore contains diagrammatic constraints on maps, will be a useful one for comprehensive design exploration.

Diagram <--> Graph Linkages

The most obvious connection between diagrammatic representations and graphs and charts is to identify some aspects of essential form in a chart -- for example, rising, falling, erratic, stable -- and to represent these diagrammatically, as an abstraction of the graph analogous to the abstraction of a map discussed above. The diagram may, in a sense, be considered a first derivative of the chart.

The mechanics of using diagrams as input, and getting graphs as dependent output are less obvious. One possibility is to understand that certain diagrammatic decisions have quantitative implications - increased circulation conflict or reduced access to open space, for example - and that graphs of these implications can be produced as output from diagrams.

Diagram <--> Picture Linkages

The automated extraction of a diagram from a picture is another challenge to pattern recognition, though there may be some obvious examples in landscapes with strong horizontal components identifying a vanishing point, or clear skylines, for example.

To produce a picture from a diagram is a many-from-one proposition, that crosses the extreme from most propositional to most pictorial, and whose purpose or utility is therefore unclear. The query "What might this diagrammatic decision look like?" - i.e., what are the implications in another medium - is often asked, but usually answered by passing through an intermediate less abstract form; typically a plan or a map. This suggests that one path from diagram to picture is through an intermediate map.

Diagrams might serve as a powerful indexing and retrieval mechanism for stored images, using diagrammatic criteria to classify and characterize complex visual information. A retrieval system that produces images previously indexed by diagram is an easily conceived, but doubtless difficult to implement, image archiving device.

Map <--> Graph Linkages

The most obvious connection between map representations and graphs and charts is to use the map(s) as input, and derive and illustrate characteristics of the map(s); or to illustrate comparisons between two or more maps.

For example, in the visual analysis of a road, we might ask the question "What is the sequence or rhythm of wooded versus open spaces along the length of the roadway?" The answer to this query, which depends on map information (i.e. land cover, vegetation), is best represented as a linear graph, in which the x-axis is station (horizontal distance), and the y-axis some representation (bars, or connected lines, e.g.) of variation in land cover. Multiple such attributes (land cover, presence of water, residential density, etc.) can be conveniently graphed as multiple bars or lines in a graph. In another typical planning application, given a map of land uses, a simple procedure could extract Floor Area Ratios (F.A.R.'s), and produce a graph of these ratios.

The mechanics of using graphs as input, and getting maps as dependent output is essentially an exercise in knowledge based generation [Coyne, et al.]. The questions "Where along a road is the residential density 1.0 units/acre?", or "What site plan arrangement would produce a specified F.A.R.?" are examples of one-to-many relationship - there are likely several places along the highway with the requested density, and there are indefinitely many possible maps of land use that could result in any specified F.A.R. Procedures to automate the production or generation of one specimen from a set of possible answers are not obvious, nor are they obviously useful Seen as a simple search, a list of answers in the first case is perfectly acceptable; take them all, and subject them to further scrutiny. In the second example, the generation of a plan subject to constraints could be managed, with sufficient additional constraints and assumptions about the production of the plan - such an approach is one way of testing the validity of those constraints and assumptions.

Map <--> Picture Linkages

The linkage between map representations and pictures is an obvious one. Some examples [e.g.- Ervin, Gimblett & Itami] have demonstrated the feasibility of linking stored video images to geographic coordinates so that map data may be 'illustrated' with images. More abstract pictorial renditions are possible as projected views from the cartographic data base; digital elevation models with roads and vegetation overlaid, for example. Given a method for projecting a view in the first place, changing a road alignment, or a configuration of land cover, could result in a corresponding change in the projected view.

Going the other way, from picture to map, is again complicated by the one-to-many problem. Inferring three dimensional coordinates in a two-dimensional projection is subject to ambiguity that can only be resolved by a set of external conventions, so automation is difficult. Reconstructing plan views from several perspective views of known orientation is possible, and has been used in archaeological and architectural investigations. Making a change in a picture, and having the change reflected in a map, is a prime example of a constrained interaction between media, requiring considerable research and development.

Graph <--> Picture Linkages

The above discussion of graphs<->maps linkages applies equally to linkages between graphs and pictures. With an appropriate algorithm for choosing or generating examples, an illustration of any quantitative criteria may either be retrieved from a stored data base; the example of residential densities is easy to imagine, for example. Graphical analyses of visual images are known -- a histogram of frequencies in a satellite image analysis is just such a tool. Are there other analyses besides bandwidth that usefully summarize or capture information contained in a drawing or picture? If so, these might serve as the basis of other quantitative analyses of visual data.

Linkages - constraints - between media are critical in design reasoning. When a designer makes a diagram, she may or may not have an image of what the result should look like. Sometimes, when she does have such an image, it is incompatible with the diagram; a fact that remains to be discovered and reconciled in the process of changing the image, changing the diagram, until they are in agreement. Similarly, the traffic engineer may have a target graph of traffic frequency; the design problem is to find the map, or road alignment and intersection plan, that produces the desired graph. Or perhaps the target is a desired road alignment; the problem is to find a set of spot elevations that make it work. And at the same time, to have a particular view of the lake in the distance. And to be a grid, rather than a cul-de-sac system... Map, chart, picture, diagram; all must come together in a synthetic whole, but each has its own time, place and method for use. These media are distinct, but related.

Towards a Designers' Workstation

The previous sections have touched on some of the advantages of and possibilities for constrained linkages between representations (inter-media constraints). The remaining section of this paper is devoted to exploring some aspects of implementing this functionality in a "designers' workstation", for comprehensive multi-media computer aided design.

At designers' traditional drawing boards, media are mixed, explored and invented all the time. The choice of a representational medium for any particular design task is in part guided by convention, and in part by consideration of the inferential needs of the design task at hand. The level of abstraction, of detail, the appropriateness of such attributes as color, texture, etc. all enter into the decision. To support these needs computationally, we need in the first place a rich repertoire of representational devices and techniques, including multiple simultaneous graphic displays, animated and static, color and monochrome, large and small, etc. In addition, the requirements for exploring constrained interactions between media and representations include new forms of constraint management, as yet only anticipated.

A necessary first step is to begin to provide simple multi-media and 'hyper-media' technology at an integrated design workstation, and then to begin to provide constraints between representations. It is useful to imagine the kinds of uses to which such constrained linkages might be put in the service of environmental planning and design.

Each of the sections above identified some possibilities and problems with two-way linkages between media. In order to focus on the knowledge representation and manipulation, rather than the details of the technology (which are bound to change), the following scenario presumes an infinitely capable hardware manager, with all sorts of protocol interchange capabilities and arbitrarily large memory, large display(s), deep color, small resolution, etc. The following hypothetical scenarios reveal some of the breadth and complexity of design exploration offered by such a system:

"Imagine that all the red maples along the highway were to succumb to a virus. What would sides of the road in the western part of the state look like? Where might more views of water be revealed? " This requires a geographical analysis in the first place, to identify tree species and locations of water. Then an image processing component is required to 'brush out' areas of images and simulate views behind the brushed out areas.

"Make a graph of the visual quality (according to whatever metric) along this stretch of the road. Now insert the following design changes as maps in the database. Graph the new visual quality, and make a graph of the change." This requires a metric of visual quality, a mechanism for inserting changes in images, and procedures for automating the analysis of quality according to some derived parameters. The derivation may come either from recording the proposed changes, as processes, or the results, as state. The distinctions between these two approaches (essentially transaction processing) may be both efficiency and comprehensiveness, and deserve to be tested.

"Change this view as indicated by hand (edit by removing buildings and signs, changing vegetation, inserting new landscape features, etc...) Now please produce a plan view showing the altered condition." This most resembles the plan/section/perspective technology now well-understood in CAD systems, but may require some object-oriented implementation rather than projections from a single data base.

"Use this graph of daily traffic by type as a control so that as numeric values are changed, animated views of traffic at selected toll-booths are illustrated on the screen. Use a modal selection algorithm so that as density builds up, more travelers choose to use alternate routes." Here, the linkage between models and graphics is envisioned to support interactive dynamic modelling, examples of which are available in a variety of disciplines already.

Research Agenda -- Computational Issues

A system of coordinated, constraint-based multi-media representations such as the one envisioned here, and the process or processes of constraint exploration, raise an immediate question about underlying database: is it one, or many? With one, data base consistency is possible; with many, almost impossible. With a single database, multiple representations are treated just as multiple projections, or views; with many databases, each may have several projections, and conversion between them becomes the operative question [McDonald, et al]. As appealing as a single database seems from a management perspective, the reality of the multi-media environment is that it is, by definition, a multiple data base [Parsaye]. Different media, or representations, are not just different projections from a single database; that's why they are used! Indeed, translations between them are not always possible - this is one of the key issues in the development of such a system.

In addition to the problems of projection and translation, there are the related questions of database consistency, indexing and retrieval. Consistency is much to be aspired for in many traditional database applications -- 'truth maintenance' is a lofty and admirable goal -- but in designing, inconsistency, ambiguity and creative conflict resolution are essential components. The database manager in such a system must be willing to 'suspend disbelief' at times, to carry inconsistencies and conflicts, and to put up with ambiguous and imprecise definitions. Designers, of course, work towards final states in which consistency is to be found, and ambiguity and conflict resolved, but the process maybe halting, circuitous, and almost certainly non-monotonic [Ginsberg]. Not only are the algorithmic aspects of this question complex, but the associated interface issues - how and when to signal conflicts to the user, whether the manager is conceived of as an 'error detecting daemon' or a specially invoked procedure, for example-- are perplexing as well.

The essential characteristics of a designers' workstation as envisioned here is the organized management of complexity and multiplicity of representation. The computational techniques of *object-oriented design*, *message-passing* and *concurrent processing* have evolved in response to such complexity in other disciplines, and are likely candidates for the implementation of any designers' workstation or network of such stations, as they address most directly the problems of abstraction, information sharing and asynchronous operation.
Object-oriented design seems inevitable, because of the advantages of encapsulation, inheritance, generic functionality, and exception handling offered by the paradigm. Objects may be though of as databases, in the multi-database approach, and representations may be part of the objects' interface. Modelling real-world systems as objects is a well-established technique for achieving modularity in software design, as well as forcing detailed exploration of the system at hand. Maintaining systematic databases of objects also raises questions of 'persistence' over time [Smithers], and offers possibilities for exploring design variants as variants of saved state. Object-oriented programming and data base development are surely tools that the programming designer will want to avail herself of in the process of articulating exploring design knowledge.

Message passing is a technology related to (but not necessarily a part of) object oriented design. The notion that objects send and receive messages simply highlights the characteristic of encapsulation that the inner workings of objects are hidden from public view; the possibility that different objects respond differently to the same message captures an important part of the semantics of real world behavior. The possibility of building up a language of communication unique to a particular design problem or set of problems, by customizing objects and messages, seems to be an important part of communicating between multiple agents. The role of inter-agent communication, and variants such as 'blackboards', etc, has been the focus of a branch of AI research; the equivalent study of inter-agent communication in design processes is relatively new.

Finally, concurrent processes seem desirable, and inevitable, both on grounds of technological change and availability, and because, again, this approach models the real world so directly. A design project is as likely a collaboration between an architect, landscape designer, painter and civil engineer, whether these are in fact different people or just different points of view of a single designer. Each point of view has preferred media and representations, and may or not be willing to have his or her view translated to another's. This view of concurrent, communicating design systems closely resembles the computer science view of 'open systems' [Hewitt & De Jong].

The integrated designer's workstation may also embody the principles of object-orientedness, message-passing and concurrent systems at a 'meta-' scale, that is, the design studio of the future [Mitchell] may well be modelled as an interacting network of just such workstations. A major problem when multiple agents interact -- designers, consultants, and so on - is the misunderstandings that arise when each use their own medium, embodying their own viewpoint. The solution, I suggest, is not to force all concurrently operating agents into a common representation, but rather to understand the problems related to constraints and conversion between representations.

Conclusions

Intra-medium constraint management systems for design -- those that manage algebraic and logical relationships in relatively well-defined problem areas, are coming to be valuable partners in all kinds of design fields. The questions that remain, primarily involving languages for expressing and combining constraints, mechanisms for handling the inevitable conflicts between constraint systems - including questions of interface and human engineering factors - are non-trivial. This paper suggests that, in addition to pursuing that line of research, we must begin to understand the complexities of *inter-media* constraints, in which the underlying semantics, and not just the syntax or implementation, are still at issue. In addition to the computational approaches outlined above, more research on the cognitive role(s) of various forms of representation and on the forms of inference that accompany them, is required. We still know very little about the use of representations in design reasoning, the relationship(s) between maps, diagrams, graphs and pictures, or the essential nature of abstraction levels in environmental design.

The development of design tools and eventually CAD systems that integrate multi-media and constraint management is bound to both depend upon, and engender, experiments in representation. These experiments may very well result in new conventions, or new ways of seeing, explaining and understanding, that are alien to our present ways of graphic communications. Multi-media presentations should not become the equivalent of glossy renderings - reserved for final presentations - but should become useful working and exploratory devices for designers. Simulations of proposed environments need not be 'realistic' (the apparent goal of much computer graphics work and of the 'virtual reality' movement), but rather should be 'informative' - full of information both intentional and accidental. It is the intentional pursuit of accidental insights that characterizes synthetic design reasoning, and encouraging and enabling this kind of search is a major performance goal of the design of the designers' workstation.

Summary

Expression, propagation and satisfaction of constraints among diverse representations and multiple media are both necessary for comprehensive computational design support, and considerably more perplexing problems than the (relatively) well-understood numeric or logical constraint problems. A designer's workstation providing computational support for design will have to support at least four distinct representations: *diagrams*, simple line drawings representing propositional design knowledge; *maps*, whether raster or vector, representing attributes of the earth's surface; *graph* , simple graphics displaying relations among tables of numeric data; and *picturer*, arbitrarily complex visual depictions. Translating from intentions expressed in one medium to implications in another requires considerable design knowledge, and more than a single database. Designers also require constraint systems that can support inconsistency, ambiguity and multiple meanings on the one hand, and that can help them explore implications in a variety of media on the other. It is likely that such a designer's workstation, to be effective, will have to implement such functionality and support dynamic interactions and simulations, using computational techniques including object-oriented database design, message passing and concurrent processing. Experiments, hypotheses, and solutions in this direction, however partial or tentative, will be central to any effort that seeks to explain, emulate or assist the integrative, synthetic reasoning that characterizes environmental design and planning.

References

Figures:
1.a) after Chermayeff and Tzonis - diagram of Columbia New Town
1.b) after Alexander - diagram of Indian Village
2a) after Bertin p.153 - Map of France
2b) after Bertin p. 223 - Graph of registered voters
2c) after Fishman - Sketch of Green City by LeCorbusier

Alexander, C. 1966 *Notes on the Synthesis of Form.* Cambridge: Harvard University Press.

Appleyard, D., K. Lynch, and J. Meyer, 1961. *The View from the Road.* Cambridge, Mass: MIT Press.

Arnheim, R. 1969 *Visual Thinking.* Berkeley: University of California Press

Bertin, J. 1983. *Semiology of Graphics: Diagrams Networks Maps.* Madison, WI: University of Wisconsin Press

Borning, Alan, Robert Duisberg, Bjorn Freeman-Benson, Axel Kramer, & Michael Woolf. "Constraint Hierarchies", in *OOPSLA Proceedings 1987.* ACM SigPlan October 1987. pp 48-60.

Brachman, R. and H.Levesque (Eds.) 1985 *Readings in Knowledge Representation.* Los Altos, California: Morgan Kaufman Publishers

Chermayeff, S. and A. Tzonis, 1967 *Advanced Studies in Urban Environments: Toward an Urban Model.* Yale University, Institute for Applied Technology

Coyne, R.D, M.A. Rosenman, A.D. Radford, M. Balachandra & J.S. Gero. 1990 *Knowledge -Based Design Systems..* Reading, MA: Addison-Wesley

Date, C.J. 1981 *An Introduction to DataBase Systems.* New York, NY: Addison- Wesley.

Delisle, Norman, and Mayer Swartz, 1986. "Neptune: A Hypertext system for CAD Applications". *Proceedings of ACM, SIGMOD, '86.* Washington, D.C.

Ervin, S. 1987, "Levels of Abstraction in Environmental Design: A Computational Approach", *Proceedings, 1987 Conference on Planning and Design in Urban and Regional Planning* 91-96.

Ervin, Stephen, 1989. "Designing with Diagrams: A role for Computing in Design Education and Exploration", in McCullough, Mitchell & Purcell (Eds.) *The Electronic Design Studio*. Cambridge, MA: MIT Press. pp 107-121.

Fishman, Robert. 1989 Urban Utopias in the Twentieth Century. Cambridge, MA: MIT Press

Fleisher, A. , S. Ervin & M. Gross 1989 "CM2: A constraint-based environment for design exploration", Final Report to NSF #DMC86-11375, MIT DTM Group

Freeman-Benson, Bjorn N.. 1990. "Kaleidoscope: Mixing Objects, Constraints and Imperative Programming" in *ECOOP/OOPSLA Proceedings* 1990. ACM Press/ Addison-Wesley. pp 77-88

Gimblett, R.H. and R. Itami, "Linking GIS with Video Technology to Simulate Environmental Change", Proceedings, GIS/LIS 1989. Falls Church, VA: ASPRS/ACSM

Ginsberg, M.L. (Ed.) 1987, *Readings in Nonmonotonic Reasoning*. Los Altos: Morgan Kaufman Publishers

Gross, Mark. 1990 "Relational Modeling: A Basis for Computer Assisted Design", in McCullough, Mitchell & Purcell (Eds.) *The Electronic Design Studio*. Cambridge, MA: MIT Press. pp 123-136

Gross, M., S. Ervin, J. Anderson, and A. Fleisher, 1988, "Constraints: Knowledge Representation in Design", *Design Studies*, July 1988

Hewitt, Carl & Peter De Jong, 1981. "Open Systems", A.I. Memo #691, Massachusetts Institute of Technology, Cambridge, Ma.

Jonassen, David H., 1989. *Hypertext/hypermedia*. Englewood Cliffs, N.J. : Educational Technology Publications

Katzen, May (Ed.), 1981. *Multi-media communications*. Westport, Conn. : Greenwood Press.

Larkin, Jill and Herbert Simon, 1987 "Why a diagram is (sometimes) worth 10,000 words", *Cognitive Science* Vol 11, pp 65-99

Liggett,Robin S., William Mitchell and Milton Tan, 1990. "Multi-Level Analysis and Optimization of Designs"

McAleese, Ray (Ed.) 1989. *Hypertext : theory into practice* . Norwood, N.J. : Ablex Pub. Co

McDonald, John Alan, Werner Stuetzle & Andreas Buja, 1990. "Painting Multiple Views of Complex Objects" in *ECOOP/OOPSLA Proceedings 1990.* ACM Press/ Addison-Wesley. pp 245-255

Milne, M. 1966 *Computer-Aided Design: An experiment with a method...,* University of Oregon Department of Architecture, Eugene: University of Oregon Press

Mitchell, William J. 1990. "The Design Studio of the Future", in McCullough, Mitchell & Purcell (Eds.) *The Electronic Design Studio*. Cambridge, MA: MIT Press. pp 479-494

Nelson, Theodor H. 1984. "Literary machines ..." (6th Ed.) Swarthmore, Pa. [Ted Nelson]

Nelson, G. 1985 "Juno - A Constraint Based Graphics System", *Computer Graphics (19) 3: 235-243 Siggraph Proceedings*

Newell, A. and H. Simon 1972 *Human Problem Solving*. New Jersey: Prentice Hall

Nielsen, Jakob, 1990. *Hypertext and hypermedia*. Boston : Academic Press.

Parsaye, Kamran et al. 1989. *Intelligent databases : object-oriented, deductive hypermedia technologies*. New York : Wiley.

Purcell, Patrick & Dan Applebaum, 1990. "Light Table: An Interface to Visual Information Systems", in McCullough, Mitchell & Purcell (Eds.) *The Electronic Design Studio*. Cambridge, MA: MIT Press. pp 230 - 238.

Robinson, P. 1989 "The Design Debate", *Computer Graphics World* (12) 4. pp 101-106

Smithers, Tim, 1987. "AI-Based Design and Persistent Object Systems", DAI Research Paper #329. University of Edinburgh, Edinburgh, Scotland.

Steinitz, Carl 1990. "A Model of Visual Preference: Arcadia National Park", *Landscape Journal*, Fall 1990.

Tuftie, Edward. 1990. *Envisioning Information*. Cheshire, Conn: Graphics Press

van Bakergem, Dave. 1990 "Image Collections in the Design Studio", in McCullough, Mitchell & Purcell (Eds.) *The Electronic Design Studio*. Cambridge, MA: MIT Press. pp 261 - 271.

Vanier, Dana J. "HyperText: A Computer Tool to Assist Building Design", in McCullough, Mitchell & Purcell (Eds.) *The Electronic Design Studio*. Cambridge, MA: MIT Press. pp 283-300.