Harvard University Graduate School of Design

A SYSTEM FOR GEODESIGN

Stephen M. Ervin

Abstract: GeoDesign enhances traditional Environmental planning and Design activities with the power of modern computing, communications, and collaboration technologies, providing on-demand simulations and impact analyses to provide more effective and more responsible integration of scientific knowledge and societal values into the design of alternative futures. For practitioners, as well as students and researchers, this requires integration of many kinds of information, and a number of software tools, together in a comprehensive system, as no single familiar software product or approach (CAD, GIS, BIM, etc.) will suffice. The elements of such a idealized system are outlined and described in this paper.

Keywords: System components, Environmental planning, Design

// KOMPONENTEN FÜR EIN GEODESIGN-SYSTEM

// Zusammenfassung: GeoDesign verbessert die traditionelle Umweltplanung und Entwurfsaktivitäten durch moderne Computer-, Kommunikations- und Kollaborations-Technologien und bietet On-Demand-Simulationen und Wirkungsanalysen zur effizienteren und verantwortungsbewussten Integration von wissenschaftlichen Erkenntnissen und gesellschaftlichen Werten in die Gestaltung alternativer Zukunftsszenarien. Für Praktiker sowie Studierende und Forscher erfordert dies die Integration vieler Arten von Informationen sowie einer Reihe von Software-Tools in einem umfassenden System, da kein einziges bekanntes Software-Produkt oder -Konzept (weder aus dem CAD-, GIS- noch BIM-Bereich) alle diese Facetten beinhaltet. Dieser Beitrag zeigt die Elemente eines solchen idealisierten Systems auf.

Schlüsselwörter: Systemkomponenten, Umweltplanung, Design

Author

Stephen M. Ervin, MLA, PhD, FASLA
Assistant Dean for Information Technology Lecturer in Landscape Architecture
Harvard University Graduate School of Design
48 Quincy Street
Cambridge MA 02138
USA
E: servin@gsd.harvard.edu

1 INTRODUCTION

'GeoDesign' is the current manifestation of an age-old practice – planning, designing, implementing and evaluating changes to our built & physical environment – transformed by modern tools including digital databases, representational and analysis software tools (CAD, GIS, BIM, et al.), and modern communications technologies, practices, interfaces and approaches, including embedded sensors, multi-media feedback, web-based interactions, group decision making, mobile devices, social networks, and others – and given new urgency by scientific advances in understanding and analysis of Earth's natural systems (geodesign.org).

A hallmark of geodesign activities is that they are usually multidisciplinary, across a range of domain areas, and that they feature a relatively tight coupling between ideation (design) and evaluation (including scientific, economic, social, et al.) or 'generate' and 'test' in Simon's immortal paradigm (Simon 1981) – that provide rich and complex interactions between design practices and scientific methods.

Geodesign projects leverage the powers of digital computing and communications technologies to foster information-based design and provide timely feedback (sometimes 'real-time') about implications of proposed designs, often including impacts and evaluations covering a larger area, greater complexity, or longer time-frame than the immediate design proposal (e. g. the impacts over time on watershed-scale hydrological processes of a proposed new dam, or the aggregate carbon footprint of many individual building component /system decisions.)

To date, the technical infrastructure for geodesign has been achieved by using the tools of existing GIS, CAD and BIM systems, coupled with spreadsheets, databases and emergent web techniques. This ad-hoc approach has not been altogether satisfactory, as there are still many 'interoperability' issues to resolve; and, perhaps more critical, there is still considerable debate about whether and how 'design' is supported and enabled (or thwarted) by these existing tools. Many designers report that the ideation phases of design – creating new representations in some medium to express, explore and refine concepts and relationships – are hampered by the relative clumsiness of using a mouse and software, especially compared to the fluidity of pencil on paper. Some report that 'tablet-based' interfaces such as using a 'Wacom tablet', improve the experience. And yet others, mostly (younger) 'digital natives', are immune to these frustrations and report that pencil on paper seems an impoverished medium to them. A new era of software design and delivery emphasizes multiple inter-operating 'apps' that perform specific tasks, with access to a rich array of databases and emergent interface methods, that may be 'mashed-up' to create new synergies. In that spirit, the following proposal is for a system of component parts, with more emphasis now on the parts, than on their systemic integration. That will have to come from practice.

2 SYSTEM COMPONENTS

A hybrid GeoDesign System (GDS) will require a combination of the best of various legacy tools (CAD, GIS, BIM, e. g.), and some new and best-practice techniques, such as object-oriented diagrams and key-indicator 'dashboards'. These will be obtained not by perfecting any one software product, but by leveraging interoperability and providing essential modularity and flexibility in component design.

The ideal system presented here has no current manifestation, and is not formulated within the constraints of any existing software products. The integration points required between the several components presented below are complex, but are glossed over here for the purpose of presenting the broad essential set of components required for a 'geodesign system'.

This proposed system will entail at least sixteen essential, interrelated, components, as identified below with simple descriptive names. In the next section, each is described in more detail.

GeoDesign System Components:

1. Environment/Context-Base
2. Configuration
3. Elements (Objects, Classes, Properties, Methods)
4. Constraints
5. Analyses
6. Simulations
7. Dashboards
8. Version Manager
9. Time/Dynamics Manager
10. Level Of Abstraction (LOA) Manager
11. Diagram Manager
12. Algorithmic Interface
13. Text/Media (Hyper-Annotations)
14. Library
15. Collaboration Tools
16. Design Methods Coach

3 SYSTEM COMPONENTS: DETAILS AND INTERRELATIONSHIPS

3.1 ENVIRONMENT/CONTEXT-BASE

Most geodesign projects happen within a specific geographic context, or area. While some digital information for that area may be available as 'layers' in a GIS or other database, sometimes all that is available is an air photograph, survey map, contour-/hypsometry plan, or other 'base' information, which sets the stage, and the frame, for the project. It's important that this base environment information extend a good amount beyond the working boundaries of the project, so that edge effects can be most effectively considered. Information in this "Environment/Context-Base" category (often, one or more separate display layer(s) includes that which is not expected to change or be changed for the purposes of the design project, and which serves as a 'reference' for visual orientation. Other elements of base information may well change over the course of a project, as conditions change, or other better or different information becomes available. Linkages from the active design layers and elements back to this reference become constraints that can be generalized in diagrams for describing the design ("the treatment plant should be located near the stream, on a flat slope".) This component is a multi-media, hyper-linked, multi-layer, geo-referenced repository – not just a 'background image'.

3.2 CONFIGURATION (THE 'PLAN')

The 'Configuration' at any instance is a record of all the elements specified by the designer, including their attributes, spatial layout, and other logical relationships with each other, as well as to base or other contextual information. Relationships such as 'house face parallel to sidewalk' or 30 maple trees at 12' on center, are aspects of a configuration, as are arbitrarily complex polyline shapes that cannot easily be described in words. The configuration may be a fully 3D detailed and dimensioned set of construction documents; or may be a simple 2D concept diagram. Keeping track of the configuration at different levels of abstraction, and its evolution and states over time, along with supporting evidence including inspirational notes and impact evaluations, is the true function of the GDS.

3.3 ELEMENTS (FEATURES, OBJECTS, CLASSES, PROPERTIES, METHODS)

The geodesigner will mostly work, in the 'Configuration' described above, with 'Elements' – the atomic and molecular particles of the design. These may be broad and aggregate, such as a forest or neighborhood, or specific and detailed, such as a doorknob or a single tree. Since these will also be created, refined, and manipulated at various levels of abstraction, and with kinship relations (e. g., a 'water park' is a special kind of 'park', etc.), the 'object oriented' paradigm of modern computer programming will be essential.

Objects & Properties

Object oriented programming (Budd 2002) and related approaches such as object oriented databases, have a number of advantages in the production and maintenance of software that have made them standard approaches for most big modern software systems. Two essential aspects of this approach stand out:

1.) 'Objects' are bundles of attributes such as height, color, cost, along with 'behaviors', embedded in 'methods', such as 'calculate cost of installation' or 'calculate new water level', etc. This makes these objects stand for, and often behave like, real-world objects, rather than just primitive computer graphics elements such as points, lines, polygons, tables, etc. (Even just the nomenclature helps designers, who benefit cognitively from feeling like they are 'placing a door', or planting a tree, rather than 'drawing a rectangle', or a circle.)

2.) Objects exist within a 'taxonomy', or 'hierarchy', such that some objects are 'classes, (such as 'trees'), while other objects are 'specialized instantiations' of the class, such as 'maple trees', which inherit all the attributes (including methods) of their class, and may also have some more detailed and possibly additional attributes beyond those they inherit; and there may be deeply nested class-subclass-object relationships (such as 'this particular maple tree', that has an age, location, height, maintenance record, etc.)

One efficiency that results is that attributes and methods that all individual instances share can be defined at the class level, and applied everywhere, while individual objects can also have additional local, or idiosyncratic, attributes and methods as needed. In software engineering this means that shared libraries of common methods and procedures and bundles of attributes can be developed and shared as 'class libraries', to be re-used and adapted as needed. Object-oriented is widely accepted as an efficiency in modern day programming. Geodesigners could benefit from the same efficiencies and expressiveness with access to libraries of object-orient geodesign elements (OOGDE's).

3.4 CONSTRAINTS (RELATIONSHIPS)

The relationships between elements in the design include simple position and adjacency and proximity – the most elemental geo-relationships – as well as other more complex or dynamic relationships, such as different wells tapping a single aquifer, or a sound absorbing wall whose height is a function of the width of an adjacent roadway. Some of these relationships are simple, linear and algebraic, like the last one; others are complex, dynamic, and possibly heuristic and incalculable (the behavior of neighborhood children as a function of the size and design of a neighborhood park and availability of water ...). The formalism of 'constraints', which have been shown to be very powerful in computational simulations of real world objects and systems, (Borning 1979), must be managed by the GDS, and are essential inputs into the 'Analysis and Simulation' functions below.

3.5 ANALYSES (MODELS)

Familiar to many GIS projects, which use geoprocessing models to extract A System for Geodesign information of create new, analysis models are essential to the GDS. The tight coupling between analytic models, simple or complex, and design moves, manual or algorithmic, is a hallmark of "GeoDesign". Regularly comparing design proposals to their predicted impacts, in addition to using automatic algorithmic approaches to design, are distinguishing characteristics of geodesign projects, that leverage the power of computing.

Analysis models may be purpose-built in some suitable interface (e. g. ArcInfo's "model builder") (Allen 2011), may be taken from a library of off-the-shelf analyses, or may be performed essentially outside of the GeoDesign System, by exchanging parameters extracted from the design configuration, running complex models synchronously or asynchronously, with results possibly reimported into the design, or reported on a 'Dashboard' (see below).

3.6 SIMULATIONS

'Simulations', like analytic models, are performed to learn more about the properties or behavior of the design, over time or under varying conditions. These usually step forward in time, in some controlled and granular way, and may produce animations that are observed for visual inference, or may generate other quantitative outputs, just like a simpler model.

Simulation toolkits are available for many particular design domains, e. g. transportation, hydrological and structural performance, an others. A particularly effective approach is 'agent-based' models, in which objects designed to simulate certain kinds of behaviors (those of animal species, or human shoppers, e. g.) are 'let lose' in a virtual environment, and their behavior and interactions recorded and analyzed. This is particularly effective in a distributed design infrastructure in which one team may be designing an environment, another designing suitable agents, and yet a third overseeing simulations.

3.7 DASHBOARDS

'Dashboards' are element of modern information systems that have gained popularity for managing complexity, and engaging human-computer interaction (Few 2006). Simply defined as 'a visual display of key performance indicators', there is a presumption that the dashboard display is concise, and taken in at a glance, using cognitive features such as color ('red' for a warning or danger situation, e. g.), as well as instantly updated, or at least current and timely, so that they can help guide the design processes. Dashboards can be used to monitor goals and thresholds by simple counting (such as 'need to provide 1000 parking spaces; 497 provided, 503 to go' or 'cut and fill not balanced', etc.) as well as warnings based on more complex simulations and analyses such as 'danger of downstream lake eutrophication' or 'increased public health risks over time'. Real-time distributed processing and feedback of results, in a suitably high-level/aggregate form to help inform evolving design decisions, is a must for highly effective geodesign. The specification of a suitable dashboard is a design problem in itself, inasmuch as it depends on situation-dependent determination of key performance indicators, some knowledge of both generic visual interface principles as well as user-specific customizations, and integration with analytic and other routines. Generic geodesign dashboard 'templates' are a likely place to start, based on real-world experience and community-approved defaults, that will doubtless be specialized over time in various applications and use-case scenarios.

3.8 VERSION MANAGER

Geodesign projects, like all design projects, may generate multiple variants and states over time. Managing these, with a consistent naming convention for retrieving different versions or demand, with sufficient modularity so that individual elements or arrangements can be copied and incorporated into yet more versions, is a demanding task. Most human designers or design teams depend on basic file naming and time-stamping to achieve this; some have access to more sophisticated version control systems such as the 'VCS' used by software programmers. When geodesign projects feature teams of collaborative designers, managing versions across time and teams becomes even more complicated – and essential. Stored versions need not just elements and relations, but 'metadata' describing the conditions, goals, special considerations, etc. so that they can be recovered in the future and interrogated for 'why is this?'.

3.9 TIME MANAGER

'Simulations', 'Analyses' and 'Versions' all need information relating to time, identifying special properties of certain moments in time (groundbreaking, construction calendar, future generations) as well as the properties and impacts of dynamic processes in the environment (flooding, growth, social change, e. g.) (Costanza & Ruth 1998). The time manager may be considered a cross-cutting set of properties attached to various elements, combinations, or arrangements; required since time based analyses, simulations, projections, and impacts are so important in many GeoDesign projects.

3.10 LOA MANAGER

'Levels of Abstraction (LOA)' are critically important to many design processes (Jaques 1978). Designs often start at a high level of abstraction (verbal goals, e. g., or abstract ideas like symmetry), proceed through a diagrammatic phase in which basic elements and relationships are identified, and finally move to more resolved, concrete specifications of materials and locations. Managing the relationships between high-level abstraction elements ("barrier"), and subsequent refinements ("18 thick concrete wall") is the job of the LOA manager, as well as keeping track of the current state, and enabling switching between LOAs (extracting diagrams from configurations, for example.)

Level-of-Detail (or LOD) is a closely related concept. In many digital imaging and rendering, and cartographic presentations, the idea is to present different levels of detail depending on distance away from the subject. A map at 1:100,000 shows different content (less detail) than one at 1:25,000; a tree in the foreground of a 3D rendering may have leaves with veins, whereas a tree in the distance has only a fuzzy green shape. In general, the greater the LOA, the lesser the LOD, but since one has to do with cognitive distance (LOA), and the other geometric distance (LOD) the two are not exactly inverse, or comparable. Determining appropriate relationships between them, both in principle and in practice, is an area for continued research.

3.11 DIAGRAM MANAGER

A special and central case of managing levels of abstraction is understanding diagrams. In diagrams, deceptively simple graphics are imbued with rich meaning (Ervin 2010). Circles, lines and arrows encode objects, relationships and constraints; and become instantiated with more concrete elements as the design is refined. The 'grammar' of diagrams – conventional elements and relationships, both graphic and conceptual – deserves study all on its own, and the development of an interface to software tools for diagram management are just as important as more freeform, expressive, 'sketching' tools.

3.12 ALGORITHMIC (SCRIPTING) INTERFACE

Some forms of geodesign will require not just graphical manipulation of objects and interfaces, but it is possible that some 'Scripting' opportunities will arise, as well. Both for automation of routine or repeated tasks, and for more complex algorithmic approaches, such as optimization or rule-based allocations, a programming language need to be provided. It should include at a minimum the three basic programming/scripting capabilities: named variables and procedures, repetition, and conditional branching (Terzidis 2006).

Whether this is simply a full-featured programming environment such as Java or .NET, along with an API provided for all other components; or a custom-provided scripting capability, is a detail for implementation. Many geodesigners will likely be able to get along with the later, while determined algorithmic designers will require the former. Agent-based modeling, mentioned above, is an example of the use of algorithmic tools for simulation and design.

3.13 TEXT/MEDIA (HYPER-ANNOTATIONS)

Both elements in a configuration, and steps in a process, can benefit from annotation, providing context, additional information, motivations, etc. The web has shown that 'hypertext annotations' – the ability to make any element of a document 'hot', and linked to more information, is a powerful technique for managing complexity, as well as enabling collaboration. So all geodesign documents should be like 'web pages', or 'hypertext documents', in that all elements are possibly linked to others (a movie of a particular location on a site, or a reference to an extended article supporting a particular analytic approach, or a note from the designer saying why a particular color was chosen …).

3.14 LIBRARY

As the geodesigner will need access to a collection of libraries of standard elements (objects, analysis routines, past designs and precedents, etc.), a library function will be required. More than just a typical file system, it will need a layer of finding aids on top of it, accessing meta-data, in addition to full text search and file-/folder-naming conventions. It will need to integrate both local repositories and collections, as well as shared, and global, ones.

3.15 COLLABORATION TOOLS

Geodesign projects are increasingly marked by requiring multidisciplinary collaboration, and often public (at least some amount of) participation (DesignCollaboration.org). The best of emergent collaboration tools will be needed, allowing for (automatically updated) shared documents, individually produced and (perhaps automatically) reconciled contributions, as well as shared decision making techniques such as Delphi methods, questionnaires, etc. In the management of any project, the results of collaboration may become hyperlinks, as well as stored in a library for future reference.

3.16 DESIGN METHODS COACH

There are many ways of approaching geodesign problems, varying from individual to individual, and from case to case, depending on a myriad of considerations. Nonetheless, some generalizations about design methods have been made. In particular, Steinitz has proposed a taxonomy of kinds of design problems and design methods (Steinitz 2010). His component of the GDS is closely related to the library, containing access to precedents and other geodesign projects; but can also be more, providing a virtual coach, with tools to help diagnose problem characteristics, suggest appropriate design methods, and help structure methods in the process. This module could record and classify design processes, and could 'learn' over time, over many different geodesigners & projects, and become more and more valuable, linked up to the library described above.

Much more research is needed on design methods, their details, and their utility and effectiveness in different circumstances, but this component of the GDS could be a valuable adjunct, for both novice and Expert designers alike.

4 DISCUSSION AND SUMMARY

In the above idealized description, no real reference has been made to many important 'implementation' and 'interface' questions: how many screens? How big? Mouse, keyboard, or other controller? Sound? Immersive displays? And many others … The evolution of computing devices, interfaces, and experiences is well along on its own exponential curve, and the options are seemingly endless. Some manifestations of the GeoDesign System may be all-cell-phone or tablet-computer-based; others will require a roomful of equipment and high-end multi-millions-of-pixels display devices; others will be entirely in 'the cloud'. None of the functional requirements described here are dependent upon those choices. In a System for Geodesign some combination of ergonomically convenient hand-held device for in-field work, coupled with multi-screen desktop environment, supported by asynchronous on-line access to some more massive computing power and databases (the web? the cloud?) will doubtless be common. Details of performance, as well as cognitive affordance, may well vary depending upon the form of the interface, inputs and outputs. That multiplicity of experience will be essential. There will certainly be discipline- and application-specific variants among the wide range of geodesign applications.

Whether 'sketching' in such a system is performed with graphite on paper, and scanned into digital form, or performed on a touch-screen device, also doesn't matter much. Right now, in 2011, sketching via computer software is generally a frustrating experience. Much work needs to be done on interfaces to support easy, expressive, gestural input, to match the sketchbooks and tracing paper of yore. When those developments are further along, the ability of the digital GDS to also record metadata, annotations, diagrammatic intent, contextual and environmental data, etc., will result in a powerful combination.

At its core, GeoDesign is not a new enterprise. Landscape architects, planners, architects, transportation engineers, and other allied professionals have long been engaged in this noble enterprise. What's new, and deserves a new term, is the rich deep and broad support given by modern computing and communications. Whether from sophisticated scientific models or inclusive participatory techniques, the scope and style of planning for better alternative futures (however defined) is radically affected by these technological supports and affordances. To take maximum advantage of these developments will require some shifts in working styles – away from 'single-tool' solutions, towards 'mash-ups', just as geodesigners have already evolved away from 'main-frame' or 'personal-computer' towards 'networked' approaches that may include aspects of both. It's in that spirit and direction that these outlines of essential components have been offered. Some years of experience and validation, adjustment and refinement are now required, as the practice, art, and science of geodesign co-mature.

References

Allen, D. W. (2011): Getting to know ArcGIS Modelbuilder. ESRI Press.

Borning, A. H. (1979): Thinglab – a constraint oriented simulation laboratory. Stanford University.

Budd, T. (2002): An Introduction to Object-Oriented Programming. Addison Wesley Longman.

Costanza, R.; Ruth. M. (1998): Using Dynamic Modeling to Scope Environmental Problems and Build Consensus. In: Environmental Management, 22 (2), pp. 183–195.

DesignCollaboration.org: http://www.design-collaboration.org/.

Ervin, S. (2006): On the Necessity of Diagrams. http://video.esri.com/watch/55/2010-geodesign-summit-stephen-ervin-on-the-necessity-of-diagrams.

Few, S. (2006): Information Dashboard Design. O'Reilly Media.

GeoDesign Summit (2010): http://www.geodesignsummit.com/.

Jacques, E. (1978): Levels of Abstraction in Logic and Human Interactions. Cason Hall & Co.

Simon, H. A. (1981): The Sciences of the Artificial. 2nd edition. MIT Press.

Steinitz, C. (2010): Ways of Designing. http://video.esri.com/watch/54/2010-geodesign-summit-carl-steinitz-ways-of-designing.

Terzidis, K. (2006): Algorithmic architecture. Elsevier.