

©Copyright 2016

Ivan Heitmann

Digital Media in Landscape Architecture Design Process

Ivan Heitmann

A thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Landscape Architecture

University of Washington

2016

Committee:

Kenneth P Yocom

Brian R Johnson

Program Authorized to Offer Degree:

Landscape Architecture

Contents

Figures	6
Abstract	7
Section 1: PRESUPPOSITIONS AND PURPOSE	9
1.1. Problem statement	9
1.2. Computation and the imperative of climate instability	9
1.3. Critical Stance	9
1.4. Performance in landscape urbanism theory	9
1.5. Performance in architecture	11
1.6. Advocacy, platform and relevance	11
1.7. Collaboration as a paradigm	14
1.8. Methods	14
Section 2: THEORY AND UNDERPINNINGS	15
2.1. Architecture theory	15
2.2. Topology in Architecture	15
2.3. Topology as a conceptual touchstone	16
Section 3: CASE STUDIES FROM CONTEMPORARY ACADEME	17
3.1. Case study: Gill thesis	17
3.2. Case study: Belesky thesis	18
3.3. Case study: EpiFlow	19
3.4. Summary and open questions	20
Section 4: FRAMEWORK FOR A PROPOSED LANDSCAPE INFORMATION MODEL	20
4.1. General characteristics and function	20
4.2. Encode	22
4.3. Form	24
4.4. Analyze	25
4.5. Perform	25
4.6. Iterate	26
4.7. Share	26
4.8. Document	27
Section 5: CASE STUDY: STUDIO DESIGN PROJECT	27
5.1. Process	27
Section 6: TOOLMAKING EXPERIMENTS	38
6.1. Overview	38
6.2. Terrain editing experiments	38
6.3. Field as an alternative metaphor	48
6.4. Proposal for further exploration	48
Section 7: REACTIONS AND SUGGESTIONS FROM PROFESSIONALS	53
7.1. Thoughts from a round table discussion	53
7.2. Chat over lunch	54
7.3. Office hours with an educator	55
Conclusions	56
Selected Bibliography	59

Figures

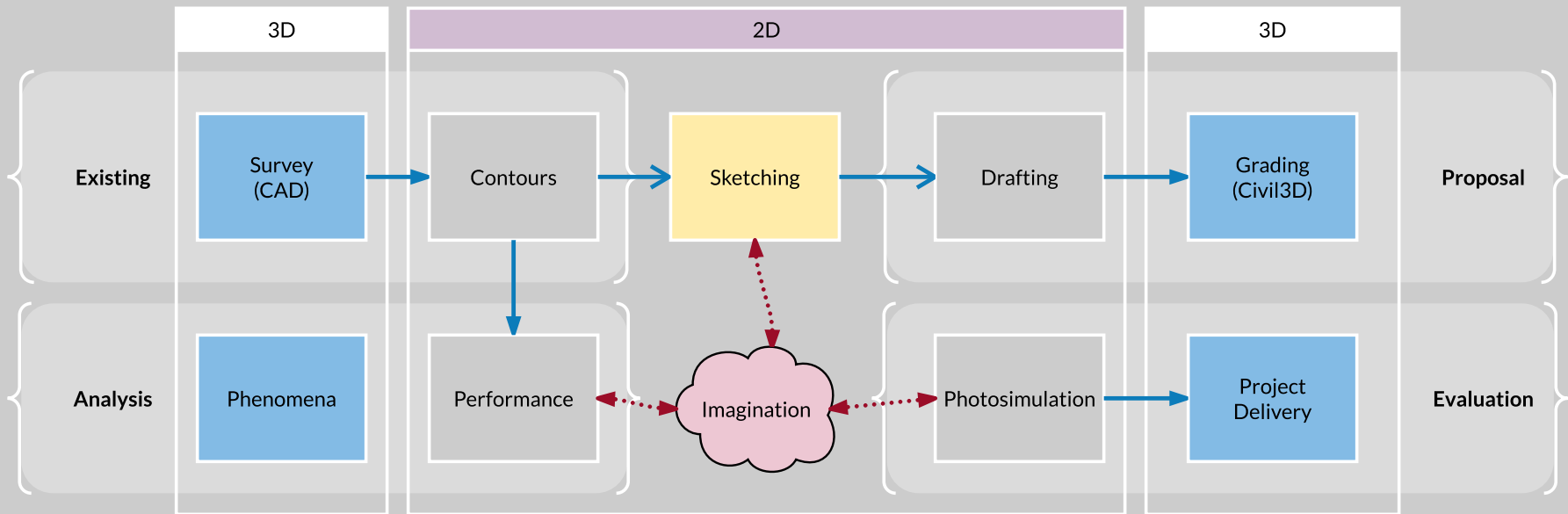
- A. Typical Design Practice
- B. Speaking Different Languages
- C. Not So Distant Future
- D. Challenges for Digital Process
- E. An Ideal Platform
- F. Case Study Design Plan View
- G. Prototype
- H. Information Flow
- J. Grading
- K. Terrain Analysis
- L. Raster Surface Analyses
- M. Not Cohesive Yet
- N. Topology

- O. Future Possibilities
- P. Surface from Raster Conceptual Flow Chart
- Q. DEM Translation Multiplication Screenshots
- R. DEM to Raster Script
- S. Grid Attractors Conceptual Flow Chart
- T. Grid Attractor Script
- U. Grid Attractor Screenshots
- V. Smarter Geometry with Python
- W. Python Terrain Editor Screenshots
- X. Python Terrain Editor Grasshopper Script
- Y. Python Terrain Editor Python Script
- Z. Vector Field as Medium Conceptual Flow Chart
- AA. Vector Field as Medium Grasshopper Script
- BB. Vector Field as Medium Screenshots
- CC. Visualization of Vector Field as Continuous

Abstract

In this work I argue for the urgency of active innovation in design process in the landscape architecture profession. I propose that reticence to innovate runs counter to current landscape architecture theory; that the relevance of the profession needs to be actively developed; that the influence of the profession among collaborating professions will diminish if newly emerging digital methods are ignored. I argue from architectural theory that there is an unrealized consonance between innovative design technologies and landscape architecture concerns. Contemporary theory in landscape architecture provides perspective for understanding and critiquing the state of the art. I summarize three efforts in academe to prototype innovative landscape design tools, noting what they achieve and what they do not. I provide a provisional framework for an adequate landscape modeling platform, informed by research in several modes: direct experimentation with toolmaking for landscape design process; applied consideration of how design process might change in this context; feedback from professional landscape architects. The latter effort is summarized in part, with key points related whether by confirmation or challenge to the fundamental tenets of this work. I conclude by emphasizing the urgency of research and development in this area.

TYPICAL DESIGN PRACTICE



A

Sites are typically surveyed in 3-dimensional data types and designs are often delivered in 3-dimensional models. Used as design proxies, 2-dimensional representations require specialized literacy (education, expertise, experience) and arguably enforce certain creative horizons. Given a schema like this one, how does imagination suffer from being located in the 2D box?

Section 1: PRESUPPOSITIONS AND PURPOSE

1.1. Problem statement

There is a worrying disconnect in the built environment design world. While a major shift is occurring in how design is practiced, that shift is not well represented by landscape architecture. The prevailing climate is moving toward integrated design, toward information modeling, toward better integrated spatial technologies (Abdirad, 2015). Cultural values are moving toward demonstrable performance gains, both before and after project delivery. Simulation tools and monitoring equipment are allowing cost and climate conscious stakeholders to make more educated decisions about where to spend the available funds. Algorithmic design tools are making it possible to test possibilities and optimize for multiple desired design outcomes. The rate of change is accelerating. Meanwhile, many landscape firms still use AutoCAD 2010 and Photoshop CS3 (Figure A). Innovative landscape firms are few and far between, and the profession is at risk of further marginalization. There is a need for significant investment in the foundations of landscape information modeling and the performance metrics that an integrated modeling environment can facilitate. Creativity, collaboration, and evaluation are disjointed and dispersed across platforms built for other purposes, none of which is adequate to represent landscape complexity. Put simply, landscape design process as it has been taught and practiced for the last 20 years is not up to the task. Delivering performance optimized landscape is not possible with Photoshop and AutoCAD—by no means the only objective, yet newly urgent. Further, many iterations are not possible when every project requires so much overhead in jumping between platforms. The promise of the landscape urbanists to reimagine the fabric of the built environment as fundamental green infrastructure is not possible without deep integration of the latest technology into the

design process. Landscape architecture is being left behind.

1.2. Computation and the imperative of climate instability

In the Anthropocene epoch of urbanization and climate instability, the built environment is vested with a new imperative. Design professionals are no longer at liberty to pursue capricious or prodigal projects in shared spaces (Ruff, 1982). The theoretical emphasis in recent years on process and performance are due to recognition that new modes of relationship between people and place are requisite to continued well-being of the human race (Swaffield, 2002). Hence the imperative of performance is not merely another passing stylistic choice on par with modernism or any other –ism. Rather it is an unavoidable necessity henceforth—though, again, not an exclusive purpose. The sooner this reality is integrated into everyday design, the better off societies will be in uncertain environments.

Computation provides powerful tools and new approaches to design thinking which directly address this necessity. While technology can never cure the ills of industrialism and should never be considered panacea, digital projections and simulations allow expert engagement with evaluating alternatives for conformance with sustainability goals—before construction. Thus it behooves the design community to invest in new methods, tools, and design strategies which make use of new technologies to reach for culturally stated goals. While economic, technical, and logistic challenges have slowed wide adoption, there has also been professional and cultural resistance to change (Picon, 2013). Sensitivity and proactivity about such pragmatics will help facilitate adoption.

1.3. Critical Stance

I hope that this research will stand as an instigation to cultural evolution by the demonstration of a value proposition as much as by any critique offered.

The basis of the argument is founded on the assumption that high performing landscapes are what we want, firstly. As a framework for thinking, I assume several things. First, representation is not value neutral, but in fact explicitly rhetorical. This means that how we represent our design has social and ethical consequences, and can be good or bad, not just effective or ineffective. Secondly, design itself I assume to be a practice with a burden of social responsibility. As such, designs can be disappointing to the public trust, or even willfully irrelevant. Finally, I will assume that theory has consequences, and can be both supportive of and incumbent upon design practice. Throughout I will return to pragmatic concerns, asking both how a firm can reasonably pursue what I argue for, as well as what measures might facilitate these new ideas in professional practice.

To successfully demonstrate the value of a proposed model of design practice, I experiment with implementations and propose practices which use computation in design to leverage the marginal advantages computers offer. Further, I consider these proposals in terms of for-profit business, and not strictly from an academic point of view. I am interested in understanding what it takes for a process to be adopted as advantageous in a for-profit environment, and not simply demonstrated before being relegated to the stacks of forgotten research. These elements constitute key practical considerations that should temper arguments for any radical evolution in landscape design process.

1.4. Performance in landscape urbanism theory

Present day theoretical discourse in landscape architecture practice and academe has engaged with the concept of ‘urbanism’, with one or another modifier. Landscape urbanism has been fleshed out well, but other modifiers such as ‘ecological’ have extended the perspective to suggest a set of approaches or values for design. Charles Waldheim has emphasized the concept of *Projective Ecologies*, the title of an

edited text (Reed, Lister, 2014) as well as a vital concept in his context. In the concluding remarks of his “General Theory” of Landscape as Urbanism (2016), Waldheim points to the indeterminate and relational character of the built environment as characterized by ecological urbanism. This new conceptual framework is famously exemplified by the James Corner Field Operations design for Fresh Kills Park. The competition winning design proposes a systems approach as much as any form or set of forms. By making the system the design object rather than a specific snapshot of an installed design, the design heralds a shift in how landscape is conceived.

Waldheim goes on to identify this approach with an emphasis on performance which he suggests has reenergized the profession. While traditional ecology and its experts play an essential role in implementing such a design approach, Waldheim mentions the role of computation in this context. Ecology itself has relied heavily on models for inquiry in recent decades, though not without controversy (Sklar, Costanza, 1990). Ecology as a way of thinking entails study and approximation of many interacting forces and feedback. While computational models have been criticized for simplifying or completely failing to predict complex ecological relationships, the intuition is easy enough to understand. Computation provides a practical approach to solving complex relationships in systems. So while landscape systems are at times different to or arguably less quantifiable than biological ones, the conceptual resonance is enticing.

This exciting potential has been referred to and described, even if rarely implemented. In her article arguing for the convergence of ecology with landscape design, Margaret Grose gestures at the possibilities of performative design testing pre-construction (2014). She calls the change toward parametrics and performance in design “profound” (p. 73). In service of landscape “action” rather than landscape “form” becoming an object of design, this shift is enabled by powerful computing. In a particularly bold formulation, she proposes that digital tools can give

greater sensitivity to physical, ecological or socio-cultural conditions. While her cited examples are experimental and within the milieu of education, few examples of built projects can claim to have attained this level of responsiveness to computationally mediated site parameters. Examples do exist, and are detailed in the latest book from Jillian Walliss and Heike Rahmann, the authors whom Grose cites. *Landscape Architecture and Digital Technologies* (2016) is an excellent overview of the state of the art. The assortment of professional projects do not, however, quite live up to the promise of Grose’s description.

I argue that while landscape urbanism has adopted ecology as a sibling, the potential of a more scientific or mathematically mediated design practice which takes the system as its object rather than the form goes largely unexplored. Fresh Kills and other watershed “performing” landscapes are generally sold to the public with diagram graphics and photosimulations of future panoramas. The buzz around “big data” exemplifies the urgency felt in the professional climate for outcomes oriented decision-making. Landscape urbanism assumes the task of designing resilient systems and allies with ecological practice, yet has failed so far effectively to integrate robust systems modeling, or even computationally mediated relational modeling. While a systems model, which predicts how animals and plants will thrive in a landscape, is a far more difficult and open problem, a simpler practice of modeling feedback relationships is accessible enough for a landscape architecture student with no scientific background (the author) to get high marks in an ecological modeling course. This exploration of quantifying forces and relationships has rarely occurred in practice, for many good reasons. The reliance of practice which Grose notes on “narrative, metaphor, or aesthetics” (p. 73) remains the dominant design evaluation mode. Still, the promise remains of actionable predictive models informing and evaluating a design concept.

Simple, first steps are certainly practical now. Hydrological analysis is often employed as an

evaluation feedback strategy, although typically outsourced to civil engineers and understood only cursorily by designers. This in spite of wide availability of basic raster based algorithms in GIS software, which might provide rough evaluation even if not rigorous enough for completing design. The behavior of water is relatively simple to model, and there are no real obstacles to integrating that evaluation more closely with the creative process. The limitation of fee structure and software subscriptions certainly constitute obstacles to growth in this integration, but example projects (notably by PEG and Keith VanDerSys, detailed in Walliss/Rahmann, 2016) show that rough hydrological evaluation of a proposed terrain is possible with a simple script in (the visual scripting platform), Grasshopper, which has a shallow learning curve and an inexpensive seat license. The essential obstacle, I infer, is that this new paradigm of computational design evaluation is more appealing as an idea than as a serious challenge to business as usual.

Challenge it is, though perhaps not as great a one as the theory might suggest. Sweeping proposals of designing algorithms instead of landscapes meet with understandable suspicion. The fear of being replaced by robots or being told what to design should remain in the realm of science fiction. The designer provides something which no computational routine can: judgment. Judgment is always strengthened by new information, especially if that feedback is easily had within the decision-making environment. Hence, avoiding integration of modeled evaluation forgoes strengthening the position of the designer. This is not only an absolute gain forgone, but a relative strengthening in the conversation with collaborators who have quantitative means at their disposal. Hence, continuing reliance on narrative to the exclusion of simulation erodes the credibility of designers in this technologically advancing climate, while missing opportunities for computationally mediated outcome projections. Data oriented design processes will never remove the value of narrative, but they certainly can improve it. Selling a plan is helped, not hindered,

by exploration and presentation of quantifiable outcomes.

While performance is appealing and well argued for its long term offerings—sustainability, resilience, etc.—it should not be forgotten that at bottom the idea is about the desire to better predict good function in singular sites. Leaving this line of inquiry in the hands of technicians refuses the opportunity to better inform the design decision-making process. I argue that it is therefore incumbent upon the landscape architecture profession to take seriously innovation in this space. The promise of projective ecologies cannot be fulfilled by beautiful graphic design.

1.5. Performance in architecture

Mainstream architectural theory has adopted this epistemological shift with gusto. Key digital theorist Branko Kolarevic describes how computational methods are directed toward new foci. To wit, computational methods are creative—“Instead of working on a parti, the designer constructs a generative system of formal production... and selects forms that emerge from its operation” (Kolarevic 2003, p. 103). The new methodology is inherently procedural, with its intention to “manifest formally the invisible dynamic processes that are shaping the physical context...” (Ibid.) Performance is defined very broadly, and iteratively achieved through “repetitive man/machine interaction (Maver, 1971, cited in Kolarevic, p. 106). Kolarevic argues that this objective entails a departure from heretofore bifurcation of generation from evaluation. This means that it is of key importance to provide a degree of “analytical and representational integration” which replaces “passive, ‘after-the-fact’” analytical computation with an environment in which “evaluation and modification take place dynamically within the design activity as determinants of, and in response to, the pattern of explorative search” (Maver, paraphrased by Kolarevic, p. 108). These prescient words are still of immanent relevance to design practice today, here particularly

in the context of their lacking development in landscape practice. Kolarevic adds that such tools would lead to new “synergies between architecture and engineering”—such is doubtless the case in the case of landscape architecture with civil engineering. One major obstacle to realizing this way of practice is the underdevelopment and lack of unity in software. As Hugh Whitehead points out (Whitehead, 2003), designers have the requisite modes of thinking in their habits, but have neither time nor inclination to learn programming skills, and hence have outgrown conventional CAD packages in their thinking, which cannot effectively describe structural relationships, physical behaviors, or the effects of time. It is of key importance that design groups with technical expertise begin to set terms for collaboration which facilitate robust information sharing and workflow structuring. Such activity serves not only as precedent but as foundational for disciplinary stored knowledge throughout design practice and ideally for agglomeration of modular computational functions available to the community with various licensing and sharing modes. An opportunity exists for leading landscape firms to begin the inevitable process of adopting computational methods.

1.6. Advocacy, platform and relevance

The premise of the legislation which protects the licensure of the design professions rests on the assumption of the public trust and stewardship of public health (Schatz, 2003). The professional organizations which provide structure for our profession take this for granted as a mandate to advocate on behalf of our discipline before the public. The credibility of the discipline, however, lies not only in the legal protection. It is essential also continually to develop and support the public platform of landscape design. This means that the onus on professional organizations goes beyond public relations, and should include territorial protection and, where appropriate, expansion. If the paradigm of contemporary landscape design is threatened or outmoded, then the public

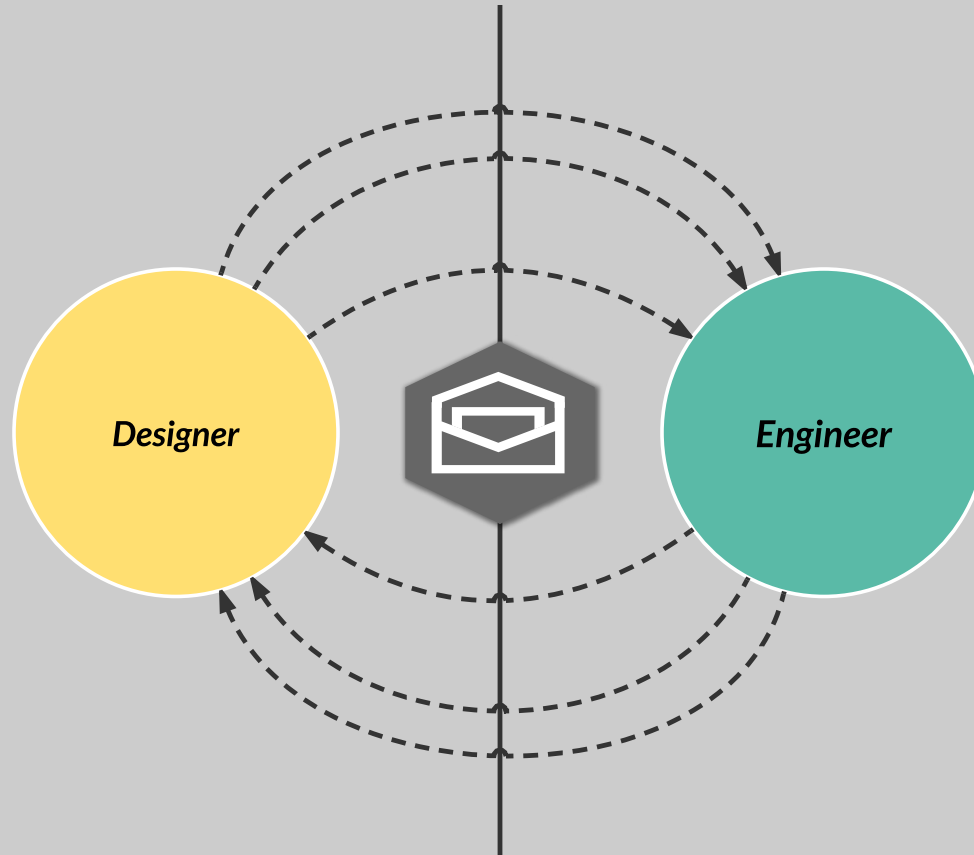
trust protected by landscape architects by law is also threatened.

One example is in the development of the Industry Foundation Classes (IFC) allowing a building information model to be shared across disciplines (Abdirad, 2015). The Classes refer to object classifications which allow their data to be stored in a standardized fashion. In a manner of speaking, it is a taxonomy of all of the types of things which there are in Architecture, Engineering, and Construction (AEC). The IFC publishers continually update the format to include changes in the disciplines, and an IFC compliant software package allows confidence that work at one office in one package will be accurately represented at a different office in a different package. Unfortunately, the landscape objects supported by the IFC are relatively impoverished in detail (Abdirad, Lin 2015). This means that IFC standards are formed with inadequate representation from landscape architecture interests. This kind of institutionalized minimization of the importance of landscape runs counter to the claims in prevailing landscape theory.

Development of the influence of landscape design in the AEC professions is also worth contending for. Already landscape architecture is often underfunded and first on the value engineering chopping block. It is essential, therefore, that the profession take steps to improve its consideration as the business of designing the built environment undergoes the tectonic shift presently underway. Without decisive and visionary action, designers will find they are fighting against the odds. Using tools intended for other markets forces landscape to fit paradigms intended for engineering or architecture.

Advocacy in software development falls not just to The American Society of Landscape Architects (ASLA), but to firms and individual designers as well. Progressive accommodation to other professions reduces the cause and the urgency for a purpose built landscape design environment. The more comfortable landscape designers become in Revit or Civil3D,

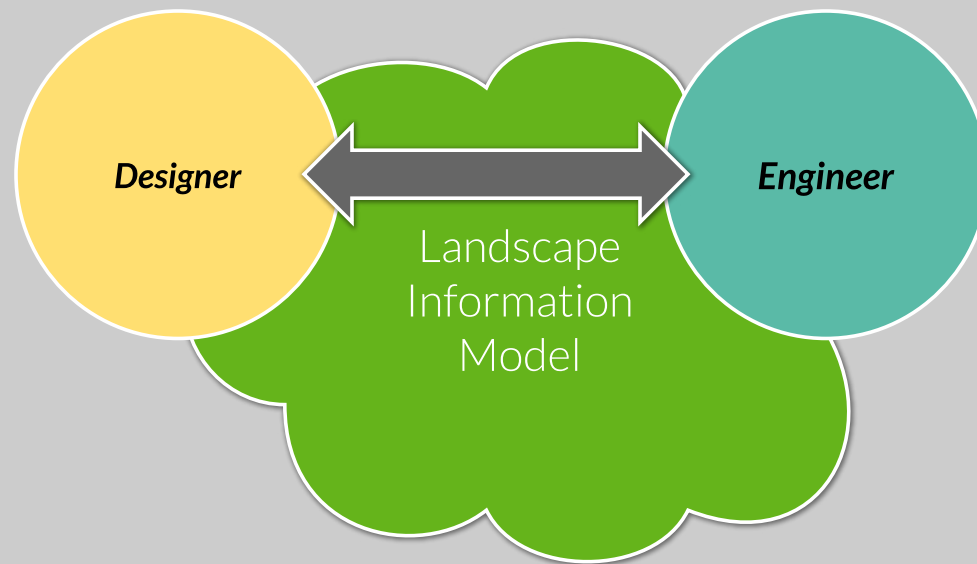
SPEAKING DIFFERENT LANGUAGES



B

Designers and Engineers rely on one another's work, although that work has to be translated or even recreated in order to fit native platforms. Emailing exported digital snapshots of a model invites miscommunication. Working in different media with different approaches means high collaboration overhead.

NOT SO DISTANT FUTURE



C

Structural architecture and engineering have already made strides toward this model, where different platforms can share data. Site design does not have a cohesive enough platform or standard for a workflow like this to be tenable.

the less hungry they are for Autodesk to provide something purpose-built. The opportunity exists for third party software development, at the moment most significantly represented by Vectorworks. Another possible route is exemplified by the open source paradigm of software development, and it is likely in this area that the best innovation is possible. The democratization of distribution allows for Grasshopper, for example, to be hugely influential within architectural design, and for one or two talented individuals to have a dramatic impact on what is possible in outcomes.

1.7. Collaboration as a paradigm

Collaboration between disciplines has become the typical way of doing business, and good collaboration enables strong design outcomes (Dossick, 2011). Collaboration comes with an overhead, as communication between disciplines is not always easy. Better tools can facilitate more efficient communication with fewer conflicts. To the degree that disciplines work on incompatible platforms, they are forced to translate one another's work back and forth (Figure B). This may be as simple as exporting between formats, or may require significant duplication of time and effort. In either case, it is an exposure to possible mistakes in translation. The value of the IFC standard, for example, is that it allows relatively frictionless collaboration (Figure C). The ideal of integrated design requires integration of design environments.

Collaboration is possible to the extent that sharing information is direct and intuitive. Any undue friction reduces the efficiency of collaboration.

The consequences of this inefficiency are potentially severe. Financial liability due to mistakes in translation is theoretically unlimited. Design will tend to be limited by what is more easily communicated, and ideas which move beyond the shared capabilities of software platforms will be less easily realized. The friction will also tend to exclude firms which are less equipped to pick up the slack, while frustrating innovative firms who are asked to make allowances for lagging technology at collaborating firms. Firms which are unable or unwilling to innovate will find it harder to win projects when the rest of the team is using the latest modeling technology. Finally, the quality of the design will suffer as landscape architecture is less able to collaborate.

1.8. Methods

As is true of the discipline of landscape architecture in general, the present work rests somewhere in the midst of several approaches. More of a conversation than a demonstration, several points of view converge in an interplay of goals and priorities. This research approach is at its heart an exploration, or perhaps an education in what is possible. Prompted by a particular value stance and the curiosity to investigate the possibilities and challenges in that stance,

this work should not be seen as normative. Rather the author seeks to be received as an investigative consultant, painting in broad strokes the present state and desirable next steps in the matter at hand.

In the following pages lie several sections. First, I review the theory of computation in design borrowed from practitioners and thinkers in object architecture. Understanding some of the development of the values and dreams proposed in that discussion sets the stage well for digging into the details of landscape computation. Second, some recent projects with direct relevance will be helpful to understand the scope and horizons of what is presently possible in the research world. With these foundations set, I turn next to a proposal for a unified landscape modeling suite. While no such software exists at the time of writing, the theory and case studies at hand may help outline what bare bones would be necessary for a useful integrated landscape platform. With such a skeleton in mind, I undertook a design studio project as an exploration of the tools, possibilities, and pitfalls. That studio design process is detailed here and some analysis provided. Finally, information and ideas provided by several professionals serve to ground-truth some of the ideas and processes explored. Through a series of informal interviews, I gathered the thoughts and experience of designers in landscape architecture who are at the forefront of innovation at their respective firms. Here I use their ideas as a sounding board for my own exploration.

Section 2: THEORY AND UNDERPINNINGS

2.1. Architecture theory

Beginning with intellectual roots in French post-structuralist philosophy, the theory of computation in architecture has evolved to incorporate ideas of performance, the generation of form, and ecologies, some of which ideas arguably would be more consonant with landscape design than with that of buildings. The community of architectural professionals and researchers has developed detailed and ambitious theoretical underpinnings for the changes in design culture over the last 20 years. In the case of building architecture, theory and literature have engaged with the introduction and adoption of digital means. In the case of landscape architecture, theory has focused on fleshing out of landscape urbanist ideas, and debate regarding their detail. This contrast reflects a corresponding difference in activity in professions. For reasons that begin to come clear through review of the literature, the revolution in technology has bewitched architecture while leaving landscape cold.

Through a combination of factors, practice, technology, and culture came together at the right moment to create a movement which has changed the architecture of objects forever (Carpo, 2004). The transition might be likened to historical developments of architectural technology such as the development of drawn perspective, or even to the invention of paper. If this sounds like an incommensurate parallel, consider that the full consequences have yet to be worked out, while our current implementations of computing technology typically persist in the paradigm and metaphors of the analog: ‘desktop’; (computer aided) ‘drafting’; ‘modeling’; collage; etc. Each of these and many more is an essentially analog process made marginally easier by digital aids, but none is a revolution. Practices that are seeds of more

radical changes are still novelties, but their growth is spreading.

Architects first challenged the currency of post-modern and deconstructivist ideology in building practice after introduction by essays by Peter Eisenmann to the philosophy of French thinkers Gilles Deleuze and Felix Guattari. Their wandering text, *A Thousand Plateaus: Capitalism and schizophrenia* (Deleuze and Guattari 1988), provided an alternative to Derrida’s shattering influence and laid the groundwork for a new architecture of continuity and of folding. The ideas introduced by Eisenmann were taken up by Greg Lynn in the seminal effort, *Folding in Architecture* (Lynn 1993 and 2004). It was here that architecture began to reclaim the primacy of material practice and the centrality of ‘tectonics’, but more importantly, struck upon and appropriated the concept of ‘topology’ for an architectural context. The concept was taken from advanced non-Euclidean geometry, in which the mathematical discipline examines the adjacencies and boundaries inherent in mathematically defined spaces, and their consistencies as they are transformed. To illustrate, a disciplinary joke runs that a topologist can’t tell the difference between a donut and a cup of coffee because they are essentially of the same form, simply stretched slightly. Writing in the early 1990’s, Lynn was more influenced by mathematics than by computer technology. In fact, computation is mentioned only tangentially in his volume. However, the availability of cheap computational power soon exponentially accelerated the popularity of this new theory of architecture.

2.2. Topology in Architecture

The conversation in building architecture has very closely held together theory and practice of architecture with developments in math and technology. In their anthology on *Theories of the Digital in Architecture* (Oxman 2014), Rivka and Robert Oxman write in their introductory materials of the arc of that conversation in the last two decades. This trajectory

as of late has become “a new form of digital design thinking”, in which “formation precedes form, and design becomes the thinking of architectural generation through the logic of the algorithm” (p. 3). A similar thought is expressed by Antoine Picon when he says that “formalism is to be understood here as synonymous with an inquiry regarding the mechanisms of formation” (Picon 2014; in Oxman 2014, p. 49). This is to say that to an increasing degree, Lynn’s emphasis on formal actualization of dynamic influence has given way to a focus on process. The first decade of the digital symbolized by Lynn is further explicated by Antoine Picon when he says of Blob architecture that “[a]s geometric analogues to events, they seemed to translate time into space while challenging some fundamental assumptions regarding the latter” (Ibid., p. 51). Instead, the latest trend shows that “complex ‘free-form’ geometry is de-emphasized as a theoretical precondition, and viewed as one possible formal result.” Material properties, tectonic practice, and fabrication technologies each quicken and influence the development of digital practice, but each in its way serves the new teleology of architecture: performance. The Oxman’s write that “digital morphogenesis will emerge as a prominent model of informed performative design in architecture” (Oxman 2014, p. 7). This they set in opposition to the representational modes of the past, characterizing the new way seeded by Lynn as “anti-representational,” and locating this observation at the “roots of architectural culture’s attempt to divest itself of the representational as the dominant logical and operative mode of formal generation in design” (p. 1).

Mario Carpo’s essay *Ten Years of Folding* (Carpo 2004), emphasizes that this result is due to the consonance of cultural factors, and not a direct result of the nature of computational tools. The required math had been long under development, beginning with Leibniz’s calculus. Deleuze’s fold had been published on paper for years. The impact of these influences came about through a “dialectical interaction—a feedback loop of sorts—between technology and society....” Digital technology is not prerequisite for or

predisposed to a particular formal style, but the theory and the tools have found a resonance which has been suitable for the prevailing culture in architectural design. Such a resonance has not occurred in landscape culture, although the broad theoretical shifts are not dissimilar between these two disciplines. In fact the narrative in the literature suggests that it is rather the mediating factors of material, tools, and making which have been missing from landscape adoption rather than a disconnect of teleology.

Certain passages lend themselves very well to direct parallels in the landscape disciplines. Greg Lynn's essay from the above cited edited volume, entitled *Animate Form* (Lynn and Kelly 1999), uses landscape directly as an extended metaphor. The authors go on for several paragraphs arguing for the product of design as a way of embodying or instantiating the processes of influence and of becoming which precede and are entailed by the realization of that design. The specific context of their argument is for architectural structure which is dynamic and responsive to its surroundings and encodes this in its form. A representative example is the concept of a frozen wave, or Hans Jenny's experiments with cymatics, which reveal the forces shaping them in their formation—the former notwithstanding its immobility. Such a concept is born out in Blob architecture with difficulty and to mixed effect, due to the inherent necessity of a building to remain static. Hence the parallel with landscape, which “can initiate movements across itself without literally moving...[.]” because it has “virtual force and motion stored in its slopes.” The point is in clear parallel with the emphasis in landscape urbanism on dynamic and procedural landscape conceptualization. Other similes in this text include discussion of ‘fields’ and ‘ecologies’ and the citation of holistic thinking that breaks down differentiation between single bodies and ecologies of organisms. In fact, the theoretical underpinnings of seminal projects such as Fresh Kills are part and parcel of the appropriated Deleuzian philosophical formulation hinted at above.

Despite this consonance and teleological overlap, little research or essay has been published translating the key concepts of this porous topological paradigm for the outdoor environment. In a 2011 *Architectural Digest*, Sheldon and Witt identified the topological properties of space as “connectivity, passage, edge and rupture” (Shelden and Witt 2011). These terms beg to be applied in the context of landscape. They juxtapose this taxonomy with reference to the emergent properties of circle and shape packings and to the circulation diagram. This association becomes somewhat clearer with Lars Spuybroek's consideration of Frei Otto's famous wool thread experiments (Spuybroek 2005). Deemed “analog computing” by Spuybroek, he details how the formfinding process creates structure more efficient than an orthogonal one. He represents the collections of fibers as singularities supervening on a system structured by holes. That goes to demonstrate that while the sizes of the holes and the thicknesses of the aggregated threads may vary due to the ‘computational’ process, the fixed points determine the final adjacencies, the topology of the form. Recall then that the central metaphor for this experimentation was urban connectivity by paths of travel. Each fixed point was cast as a dwelling, and the threads became roads. Hence the structuring spaces between the threads represented urban places of staying, libraries and parks and shops. That is to say, one of the most influential and well known experiments in algorithmic computation took the urban landscape as its primary framing. It is a short step to compare this to the idea of landscape as infrastructure, as organizing fabric of city life.

2.3. Topology as a conceptual touchstone

From its modest roots as an obtuse mathematical discipline, ‘topology’ has emerged as a carrier for architectural discourse (Shea, 2004). While mainstream theory in building architecture has made use of the term for its geometric connotations, Christophe Girot has proposed the term as a core idea for a unifying approach to landscape conversation. His essay is

entitled “The Elegance of Topology,” and appears in the third volume of the *Landscape* periodical (Girot et al, eds, 2013). What he has in mind seems to be a holistic inquiry into the logic of the terrain. Perhaps echoing the idea of the *genius loci*, his approach appropriates the term topology as a mediator of the essential structure of a landscape as visible in internal relations of form.

Girot begins by arguing that the primacy of the planimetric conception of landscape limits the range of possible interpretation of place. It goes without saying that a plan representation omits direct representation of terrain. He points out that the landscape overlay method confuses deductive analysis with the “poetics of place” (p. 82). He further argues that perspective itself detaches the viewer from the reality of a site for the sake of a “projected ideal” (p. 95). Over and against abstraction and reduction in process, he proposes reconnecting with the “aesthetic substrate of a site” (p. 91). He pans quantitative methodologies and functionalism while mourning the loss of planning and management projects to large engineering offices. He proposes that new survey and modeling techniques offer opportunity for landscape architects to claim benefit to their design approach from strategies generally used in engineering practice.

Girot is inspired by the landscape representation consequent to laser scanning technologies. The 3D digital point cloud offers a stunning fly-through interaction with a precisely detailed and colored proxy of site. With such at-a-glance feeling for the whole of the landscape, the technology facilitates a very different relationship with the site than a plan view can. It allows his call to a “renewed attention to terrain” and begins to suggest how “‘surgical’ interventions” into a terrain might allow for “art and technique to blend into a single elegant topology” (p. 92).

Though Girot relies on an etymological glossing of ‘topology’, his concept seems to carry forward some of the essential connotations of the mathematical term. He emphasizes the Greek meanings of ‘place’

and 'language' in the roots of the word, supporting his construction of a successor to the genius loci idea. Whatever else topology may be in his theory, it begins with a sensitivity to the relations of the parts to one another and to the whole. There is a resonance there from the idea of studying the invariance in form subjected to transformation. For example, his lectures often include exposition of some essential landscape typology, including the clearing and the walled garden. As landscape typology, these examples rely on a certain instinctive recognition of the relationships of parts to form enclosure, or opening. In that sense, both these types could be considered to have topological invariants which are recognizable as having cultural and functional attributes regardless of their particular expression. While Girot would doubtless explain it otherwise, the touchstone he proposes is a cogent successor to the mathematical concept regardless of how much weight he asks it to bear in his theory.

Interesting as a history of theory, the new idea of topology and the emphasis on terrain serve to underline the impact that a new technology can have on the essential approach to a design space. A useful parallel is the development of perspectival drawing, and the impact that had on architectural design. The laser scanned point cloud terrain model may prove to be an innovation of similar magnitude if it serves as the catalyst for such a shift in landscape thinking. Terrain, in his theory, becomes the substrate of the culture and symbolism of site, "because it relates to the genealogy of place" (p. 92). I think it fair to interpret this to mean that genealogy of place encodes culture and symbolism, and that landform embodies what is unique about a site, including its cultural and symbolic meaning. Understanding of landform is limited by orthographic views, and the new representation facilitates more direct conversation with this embodiment.

Laser scanning challenges the designer with an abundance of precision, demanding an account of terrain which is easily elided in perspectives. Hence

the picturesque aesthetic theory born of the landscape view is significantly undercut. Rather, grappling with the terrain roots the designer in the "gritty reality of today." One of Girot's summaries of this new approach describes that "topology is about the mastery of terrain and the creation of timeless landscapes inextricably linked to their respective site, culture, and environment" (Ibid., p. 113).

Though Girot dispenses with a quantitative approach, his proposal is nevertheless a rigorous one. It recognizes the impact that technology has on design process. It calls for a deliberate reckoning of ideation with precise modeling. It compellingly places at the center of this picture the terrain, the landform. Some may not agree with Girot's emphatic replacement of program with landform as the essential substrate of good design, but it is safe to agree that future innovation in landscape design process must account for and cannot avoid being influenced by this new representational mode.

Because good argument can be made for the foundational nature of terrain, and because terrain is a landscape element which lends itself well to computational means, there is great potential in exploring the interaction of technology with terrain. Many landscape phenomena and systems are influenced or defined by terrain, while mathematical models are well developed for inquiry into terrain, beginning with hydrology. Landform, then, is an excellent starting point for an exploration of the intersection of design with computation in landscape architecture. Its study is accessible to all, while its impact is without limit. Landscape modeling must give account for terrain, accurate data is readily available, and evaluation methods are well developed. Terrain, then, is the foundational element of the following inquiry; apt, because terrain is the least well accounted for landscape element in developing information modeling software platforms. I use this theory of good designer / terrain interaction as a standard to strive for in the following examinations of design tools and processes.

Section 3: CASE STUDIES FROM CONTEMPORARY ACADEME

3.1. Case study: Gill thesis

Lewis Richard Gill completed his PhD thesis in 2013 on A 3D Landscape Information Model. A computer science student by training, he completed his prototype under the supervision of computer science and landscape architecture faculty at the University of Sheffield. Given that the British government has mandated delivery of public projects in information model format, it is natural that a British student would be the first to attempt such a prototype. In his abstract, Gill describes his motivation as exploring the possibility of placing a virtual 3D model central to design process.

The contribution of the thesis begins with an inquiry into interaction design. He devises an experiment to compare fly-through of a 3D digital model with 2D representations. The research gathers responses from 30 individuals with expertise in the landscape field, and finds that there was some preference for the 3D model, though 2D plans were also highly rated. While his results in this regard find only weak correlations, the methodology is a sensible one. Setting out to create a modeling environment without research may be unwise. Gill does not, however, provide any experiments to test how designers want to interact with the landscape, only to what degree the representation is informative.

He turns next to defining a 3D Landscape Information Modeling (LIM). Taking the Steinitz model of landscape change (Steinitz, 1990) as a baseline set of functions, he argues that a LIM should also inform regarding future performance. The Steinitz model is sequential and cyclical, including mental models of representation, process (operation), evaluation, intervention, impact, and decision-making. Those familiar with the Steinitz approach will be aware of his emphasis on a regional scale land use planning

approach. The Steinitz model is well suited to that approach. In the fashion of Ian McHarg, layering determines result. Although this approach is conceptually applicable to a site scale, in practice sites tend to be less cooperative within that structure because they are more difficult to generalize. Many (including Girot) have criticized the layering approach as being essentially deterministic, and the Steinitz approach likely deserves the same criticism. I argue that Gill's prototype is conceived for representation at the village or city scale, and that his reliance on the Steinitz model reinforces this bias. While landscape design at that scale is certainly important, it could more accurately be called landscape planning. Design at the site scale would need a LIM with a conceptual framework very different to tools like CityEngine (one of Gill's precedents). Gill proposes that a representational model in a LIM should be able to link up to whichever analytical models are required by design process. While the sentiment underestimates the degree to which the tools determine the process, the idea of a modular interface which sends spatial data to various analytical tools is sound and exemplified by ESRI's ArcGIS suite. His structure anticipates performance predictions informing the user while also constraining the possible edits, in real time—admitting the risk of too much restriction when designers wish to break rules. Metrics would be displayed within the interface, whether overlaid on the model or floating as 2D charts or graphs.

Gill's implementation uses a procedurally generated representation of 2 dimensional data. GIS software has these capabilities, and Gill uses that metaphor and that data source as his foundation. He also determines to allow designers to sketch in 2 dimensions while automatically rendering their sketches digitally. The result is a 3 dimensionally rendered representation of a layered stack of coverages. Building heights, for example, encoded as metadata, determine the extrusion heights of simple building masses. Forest density is a further metadata entry which is translated into 3D. The landscape representation metaphor

here is very like GIS, and shares its strengths and drawbacks. Gill is building something like an interactive version of ArcScene. For example, building footprints are simply floated to the lowest intersecting terrain point. User edits to shapes and coverages are allowed in the 2D representation, with the model updating automatically to reflect changes.

The project includes provisions for rendering and export. Gill's approach focuses on color coding model elements based on their attributes. This allows for orthographic representations to be exported as images to represent the designer's intention.

This prototype has several limitations, some of which Gill describes. Firstly, there is no interactive function for editing terrain; the user must manually regenerate the model after a series of edits to the raw data. Secondly, the system assumes that most of the design has already occurred when the designer begins: editing polygons is not an interaction which lends itself to creativity. The workflow seems substantially identical to drawing a design in AutoCAD which is subsequently projected into 3D in ArcScene and converted to coverages which can be rendered in different styles or filled with randomly distributed objects. There is little room here for design to take place, and this schema is not likely to be integrated into landscape design practice.

Gill goes on to experiment with Bayesian network integration, microclimate modeling, and flood modeling. Each evaluation process interfaces with the base model and returns its result in a form which can be projected onto the display. He implements various robust analytical models and incorporates their results in visually accessible ways. Ultimately, his project has some impressive facilities for testing design alternatives. It could be very useful as a stand alone software package reading standardized spatial information and running the battery of performance tests against 3 or 4 alternatives. However, the real design work must happen outside the platform. The platform does not support a more refined level of

detail necessary for site construction, but focuses on larger scale planning measures. By comparison with functionality well used in Revit some of the gaps become more apparent. Relational modeling, in which geometry allows for dependencies (or 'hosting'), is not possible. Custom form for landscape features is kept an external procedure: all furniture must be modeled elsewhere, for example. Importantly, the export functions are not well suited to construction documentation, meaning that the digital model will never be the primary representation of the project, but only a parallel one used for its analytic abilities. Because this analytic information could be considered non-essential, it is unlikely to be incorporated as a regular part of a professional practice.

What can be learned from Gills approach includes the modularity of the analytical functions. The essential conceptual structure of designer -> intervention -> evaluation -> display -> information -> iteration seems well considered and applicable to design process. Reducing the friction in that feedback loop remains a good primary goal of a landscape information modeling platform. As an analytic suite of tools, his project excels. Considered as an extension of GIS technology, the work is very valuable. GIS technology has not, however, influenced design process nearly as strongly as it might have been anticipated (Picon, 2013). Hence there seems to be a disconnect between the design process as actually practiced and the digital editing workflow offered on a platform such as GIS. It should not be surprising that the creativity possible with pencil and paper is not easily translated to digital polygons.

3.2. Case study: Belesky thesis

Philip Belesky took on an exploration of the role of computation in landscape design process for his Masters thesis at the University of Victoria in New Zealand (2013). He begins with the premise that computation should be more prevalent in landscape architecture.

In the section entitled “The Root of the Problem,” Belesky argues that landscape urbanism fails in specificity. Because designers have no tools to adequately represent complex phenomena, they inevitably fall back on abstraction. The pitfall to such a process, according to Belesky, is that analysis in this mode “feeds into process-driven interventions which impose infrastructures and programmes that bear little relation to the pre-existing conditions of the landscape” (p. 30). The design narrative is supported by what he calls “self-referential aesthetic exotica which operate apart from a site” (Ibid.). This criticism recalls Girot’s complaint. Despite best intentions, design by narrative and diagram lacks specificity because it cannot encode the systems which define the spirit of the site, the poetics of place, whichever you will. Speaking of representations exemplified by the Fresh Kills contest graphics, Belesky says that “perverse, such landscape urbanist representations are productive because the imposed distance between design and site reduces the need to rigorously detail how the workings of the design will interface with the workings of the landscape.”

Belesky seeks a radically different design process, which allows intuition and rigor to act in tandem. He explores that possibility through a case study design project incorporated into his thesis. His own process is one subject of research, and he develops several algorithms to help determine the final design. Each algorithm is designed to be modular. Each is named: Seeds, Fields, Levels and Sections, Flows. Seeds uses site data to automatically generate planting plans from a defined plant list with attributes defined for each plant (size, moisture requirements, etc). Fields updates the coverage scheme of mapping systems with a dimension of intensity represented with dot fill of varying size. Levels and Sections automates the process of creating serial sections and observing tidal variations to submersion. Flows is a hydrological tool, which simulates flow direction over a surface.

Belesky admits that these tools were not used in generative capacity for the most part, though he did

experiment with the possibility. Each tool was instead a more sophisticated or better automated digital version of a process which is possible using manual or traditional means. In the case of automatically generating planting plans, the value of the automation is potentially significant. Parametric determination of plant placement due to site conditions is a clear opportunity for economization of effort. He also achieved the goal of reducing the size of the feedback loop between intuition and rigor, particularly with the Flows tool, which allows immediate feedback regarding hydrology when editing a surface.

With a background in programming, Belesky confirms the wisdom of using a modular approach. He also confirms the desire to collapse the distance between creativity and evaluation. Belesky’s work also shows the value of an open platform. Specifically, he is able to use the Grasshopper platform to implement modular solutions to well specified problems in the landscape domain. These scripts are in principle reusable for future projects. In fact, Philip is currently engaged with readying a suite of plugins for publication on the Grasshopper community publishing site. By tapping into an existing, well used platform, his work becomes more targeted, and his results become more relevant to others. This provides major advantages over the strategy of building a one-size-fits-all integrated platform. It allows landscape design process to gain from the open source effect, the vibrancy of collaboration. It also makes the implementation of computational tools more accessible, in that some baseline familiarity with the environment is already widespread.

Belesky concludes with the enjoiner that computation offers landscape architects a medium that is isomorphic with the object of design: complex, dynamic and open-ended. The rhyme of landscape system with algorithmic system is audible, but indeterminacy is something that computer savvy ecologists have struggled with. Stochasticity can be artificially introduced to break the determinacy of a model, but algorithms are by their nature deterministic: one set

of inputs will result in the same set of outputs every time (unless ‘randomness’ is introduced). The controversy around modeling in ecology serves as a reminder that natural systems are not as well represented by mathematical simplifications as we wish them to be.

Analytical modeling, then, is a valuable feature-set to hope for in a digital design process, but it will never be as magical as it sounds. Simplified versions of analytical tools are still of great help if they can be integrated well with creative process. Relational modeling or generative modeling are not well represented in Belesky’s work, though he does write of their importance.

3.3. Case study: EpiFlow

Dana Cupkova and colleagues (2015) describe a tool they created to incorporate hydrological tools into the 3D modeling environment. The tool was created by a team with the objective of enabling integration of natural systems into holistic design thinking. The abstract describes their intention to assimilate direct visual and quantitative feedback on water flow behavior within a single interface and to provide rapid design feedback. Ultimately, this facility serves to address public health and stormwater issues essential to environmentally conscious design in the 21st century.

EpiFlow, as it is titled, runs on the Grasshopper platform, with the functions implemented in custom code. The algorithm uses stormwater modeling math published in the New York State Stormwater Management Design Manual. The input data used is in typical GIS formats, both raster and vector. Site structures are generated from vector shape files, while terrain is represented by a mesh surface derived from a raster digital elevation model (DEM). All features are integrated into a single mesh geometry which provides the basis for hydrological analysis. As is typical with Grasshopper based tools, the algorithm is run every time a change is made to the

underlying geometry. This results in an environment which gives near real time feedback to every change made by the designer. The feedback is represented with coloration overlay in the modeling environment.

Their tool is excellent for its rigor in using industry standard math to provide accurate analysis of landscape performance. The suite provides information regarding water flow direction, water basin delineation, and computation of stormwater runoff during a storm event. They develop this functionality by integrating simulation of the performance of bioswales and green stormwater infrastructure into the overall hydrology of the site. This allows a comparison of the expected runoff with the capacity of the proposed facilities, for immediate feedback as to their adequacy.

They conclude that there is a need for further development of computational analysis within the design environment at early stages. Because the development of in-house algorithms is facilitated by the accessibility of the Grasshopper platform, it is possible to integrate analytical metrics into the design workflow. They call this “critical” because it brings to the fore the value of systems thinking in an integrated form-making environment.

While this tool is not publicly available, it could be of great use to firms engaged with stormwater design. However, its function is limited to an analytical set of tools which are not unique. As such, EpiFlow provides the efficiency of bringing a function into Rhino which was previously done in specialized software, thereby reducing the time required to iterate. It is not an innovation in kind, however. Though Rhino / Grasshopper has quickly gained ubiquity in the academic world, many firms do not use that platform for their design process. This means that engaging the provided functionality still requires migration of design proposals to a new platform for analysis.

3.4. Summary and open questions

The tools detailed above each amount to an improvement in analytical efficiency. While the Grasshopper platform is an accessible one, it is not familiar to the typical design practice at firms today. That means that each constitutes an additional tool to be added to a firm’s existing process. While this may be of use on certain projects, it is not likely to challenge the prevailing design practice.

Girot calls for a radical revision to the approach designers take. He advocates for a renegotiation of the relationship to the terrain which is holistic, precise, and accounts for the cultural significance of the shape of the space. These analytical tools are helpful, in that they make accessible to designers workflows typically reserved to engineers. Thus they reduce the necessity of enlisting engineering collaborators to work out the consequences of every design possibility. They do not, however, unseat the reliance on concept, narrative, and form which are argued here to be insufficient for the ambitions of ecological urbanism. A new relationship with terrain requires a new way of interacting with terrain.

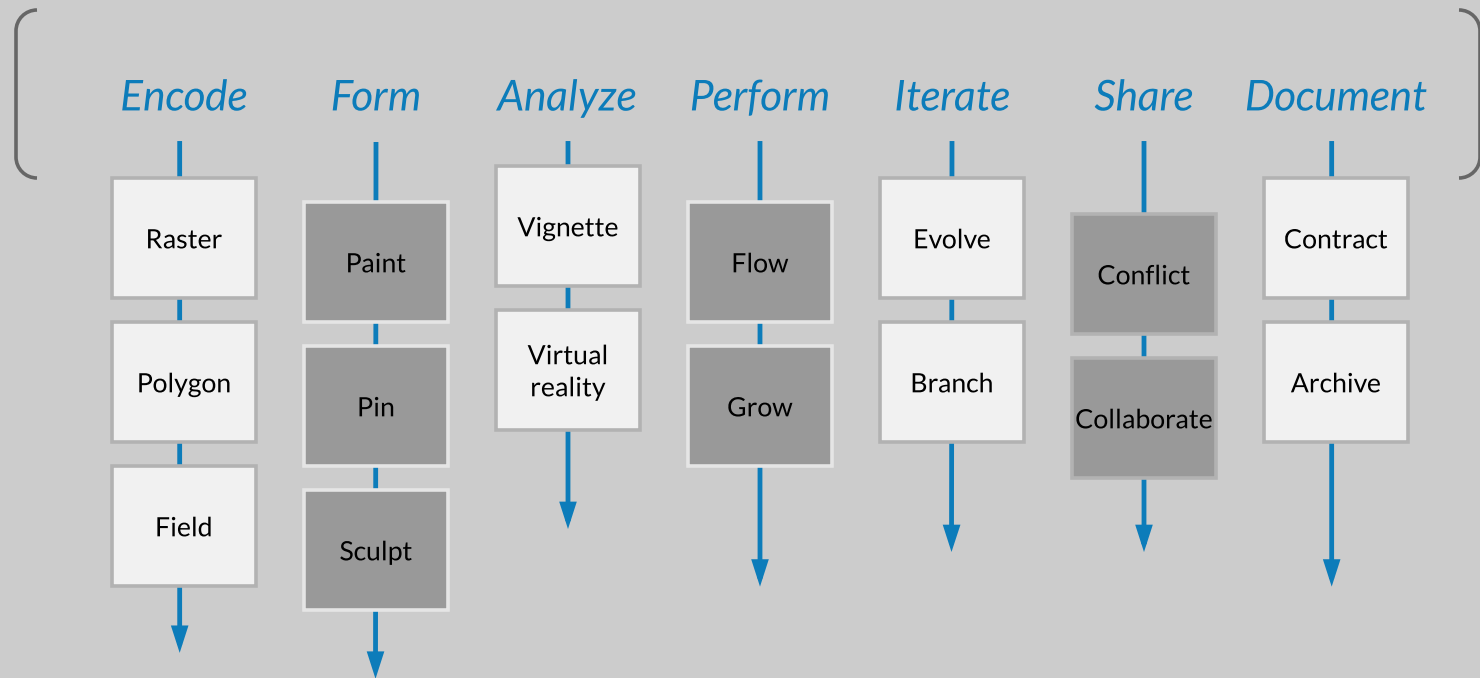
What would characterize such a medium of interaction? What affordances must a software platform provide in order to facilitate the direct relationship of design thinking with landform? What is the architecture of such an interaction, and how should an interface be implemented to make that interaction intuitive and effective? Figure D shows one possible taxonomy.

Section 4: FRAMEWORK FOR A PROPOSED LANDSCAPE INFORMATION MODEL

4.1. General characteristics and function

An ideal landscape software would incorporate many features presently available in various platforms. As a minimum requirement, such software would necessarily be capable of representing a landscape with enough specificity and in familiar enough metaphor to satisfy the expectations of trained landscape designers. Software which is inadequate in representing elements, processes, or spaces when compared with other traditional media would be ignored. To be useful, this package would facilitate several modes of interaction with the represented landscape (Figure E). It would be essential to allow for loose creativity as well as rationalization of creative gestures into precise features. The platform should provide feedback in the form of evaluation of the rationalized features. These evaluation algorithms would be immediate and provide visual feedback. The software would incorporate spatial analysis tools common to GIS platforms to reduce the overhead of translating the representation to another platform. These analyses would emphasize the characteristics derivative of form: hydrology, slope, etc. However, by adding another layer of metadata (the information in BIM or LIM), a further layer of performance evaluation would be possible. For example, well developed models from landscape ecology could begin to inform design decision. The cycle would be completed by that feedback informing the next iteration of creativity, perhaps a less gestural and more precise iteration. Thus the cycle of iteration might be facilitated by simplified versions of rigorous analytical modeling tools which are common to other disciplines (engineering, landscape ecology, etc).

CHALLENGES FOR DIGITAL PROCESS



D

This proposal for a taxonomy of essential design process functions stands as a provocation to considering the requirements for a possible landscape information modeling software package robust enough to gain wide acceptance.

4.2. Encode

4.2.1. Ethics of encoding

Representation entails elision. That is to say, models are useful when they leave things out, or even because they leave things out. The only complete model is the physical world, and every digital model will reduce the recorded detail of the realities it represents in order to contain the model. This, of course, is not a handicap but rather the essential utility of a model. In its simplicity a representation is more easily structured, explained, and examined. Because things are left out, the representation assists comprehension of the whole (Box 1979).

One consequence of this characteristic of models is that how a model is made will significantly shape what it can represent well. Hence the choices of what to include in a model impact results. This is an example of how representation presupposes or requires an ethical stance. The author may not be aware of his or her ethics while forming a representation, but his or her objectives and biases will nevertheless be influential in the product. The choice of media similarly have an impact on the resulting picture. A sketch and a photograph will represent different interactions with the environment and invite different reactions. So also digital representations of landscape will differ from other traditional means in what they emphasize and what they can encode well.

A parallel with linguistics might be of use. Languages are well understood to influence thinking, and different languages are prized for their facility with particular aspects of human experience. There are plenty of German words, for example, which have found their way into English for their excellence with a particular concept that is not otherwise expressed easily. Gestalt; zeitgeist; wanderlust. A careful consideration of a medium can yield plenty of wisdom about its tendencies, strengths, and blind-spots.

It makes sense, then, to examine different modes of representing landscape for their differentiators. This

practice helps choose the best tools and processes for the task at hand, as well as helping guard against blind spots. Insights here will also begin to show the more systemic consequences of process. In particular, it is valuable to critique efficiencies and habits in the professional atmosphere which limit the horizon of design thinking. This will include stylistic criticism of tendencies resulting from a particular software suite.

How representation is done matters. It encodes an ethic. It emphasizes and deemphasizes. Representational methods have their own inertia, and their own gravity. All other things equal, creativity in a given medium will tend to share characteristics from one work to another. Hence it is essential to choose media carefully. New media are no exception, no more neutral than old ones. Each data type has particular consequences, and each visualization mode has particular strengths.

4.2.2. Encoding Terrain as Surface

Terrain is an essential, perhaps the essential material of landscape. Any useful representation of landscape will encode terrain. In a design context, helpful encoding will facilitate interactions with terrain which are both informative and creative. Each data type, as a distinct medium of representation, has its predilections.

Landform may be encoded as a surface, with certain limitations. This representation of the terrain has two intrinsic dimensions, though a surface is extended in three dimensional space. That is to say, surfaces do not have thickness. Terrain, of course, has mass. That mass has resistance and stability. It is also heterogeneous, differing in density, composition, and qualities. Any designer who has drawn a landscape section knows that there is value to representing the depth, moisture, and earthiness of the terrain. Where the mass ends, where the earth is no more, that is where we touch it; hence a digital surface represents terrain by encoding its boundary. Such a surface can

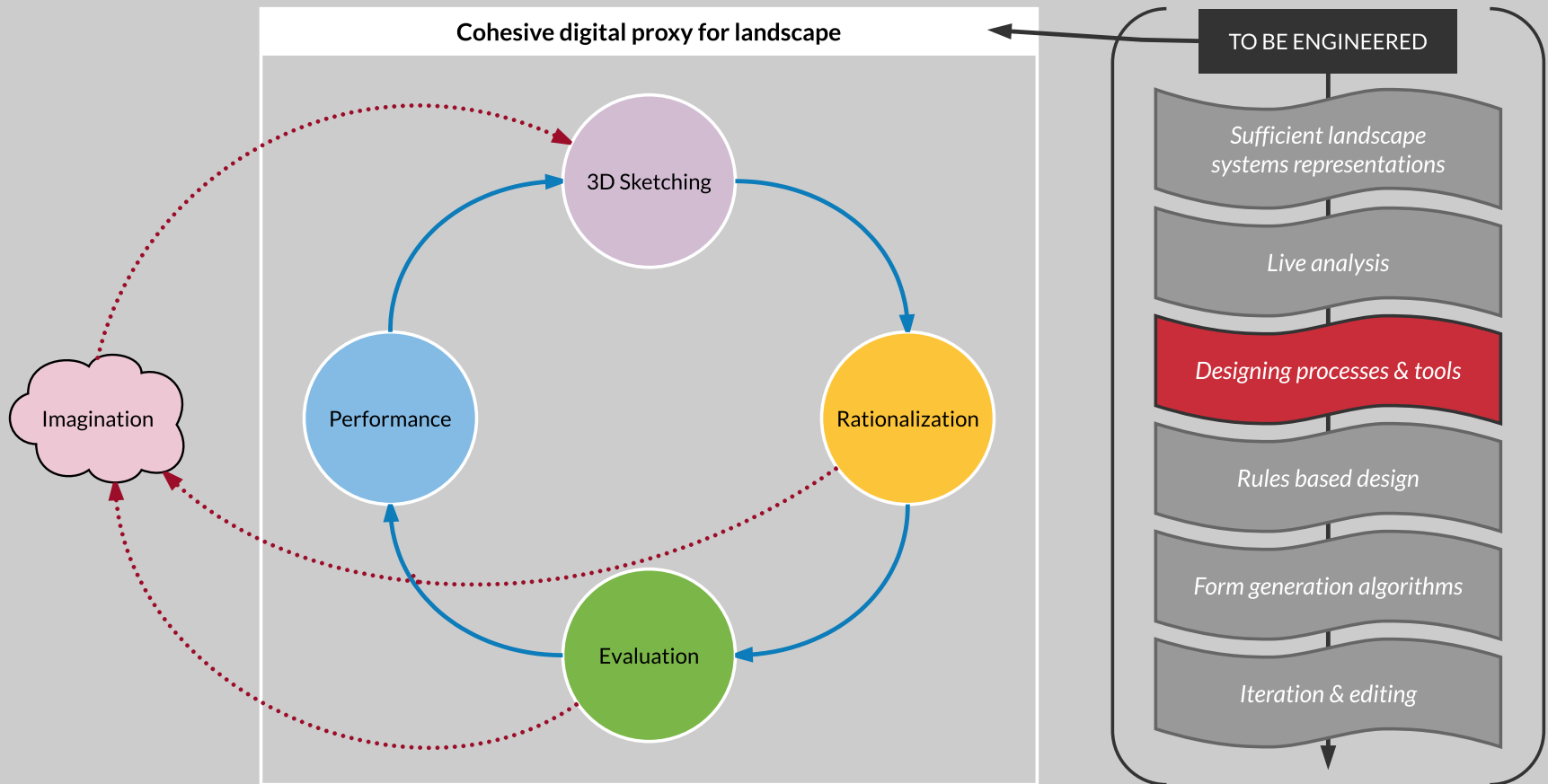
share certain, but not all characteristics of the mass which it represents.

It is simple enough to store and derive information about characteristics along the surface, but much less feasible to retain the characteristics of the terrain which result from its thickness and depth. Slope, for example, is mathematically derived by comparing positions adjacent to one another. Aspect is derived by comparing the direction perpendicular to the surface at a point (known as a 'normal') to the vertical axis, or to the cardinal directions in case the surface is using real spatial coordinates. By contrast, porosity is typically not encoded in a digital surface, and if ever, could only be as metadata (color, for example). Neither is it common to store information on fertility in a digital surface. The surface is simply not oriented toward storing information about chemical makeup or pH levels. A surface excels in representing form. Those characteristics of terrain which derive from the form of its surface are easily calculated given ignoring all others. Thus by relying on a digital surface to represent terrain, a model privileges particular attributes and frustrates exploring others.

Surfaces are in fact a category which comprise at least two common data types. Those types each have distinct characters.

One surface type is defined by NURBS geometry. Non-uniform rational basis splines (NURBS) are linear curves which are defined by control points which are weighted. The weights make the math rational, while the variable knot vector spacing makes the curve non-uniform. As an example, a symmetrical arc-like curve could be encoded as 3 points, one at each end and one in space outside the mid-point. Developing this paradigm into two dimensional geometry yields NURBS surfaces. A NURBS surface is defined by an indeterminate number of control points, each of which has a specified weight. Locations on the surface are defined by 2 dimensions along the surface, called U and V. Any given point on the surface has corresponding coordinates which are parameters

AN IDEAL PLATFORM



E

A computational environment can offer much more than digital versions of traditional tools. A cohesive platform could narrow the divide between creativity and analysis. These are some of the more ambitious functions that might be integrated into the iterative schema proposed at left.

that evaluate to a single (x,y,z) solution. That means that every point on the surface has a location determined procedurally by calculation on its (u,v) surface coordinates. These results will be determined by the control points, but they will not be coincident with them. The control points float in space away from the surface.

While NURBS surfaces of order 3 or higher are inherently smooth and relatively easy and intuitive to shape by moving control points, they are not well suited to encoding terrain because their determining elements are off the surface. This means that placing and fixing a particular point on the surface is difficult. While surveyors, engineers and designers will often think of terrain in terms of spot elevations, this way of working is not facilitated by NURBS math.

Another typical data type for encoding terrain is triangulated irregular network (TIN) meshes. A mesh consists in vertices which are numbered and connected to their neighbors with straight line segments. The resulting faces can be of arbitrary size. A quadrangular mesh can have non-planar faces, but a triangular mesh will by definition have only planar faces. The result is a surface composed of joined triangles, each of which is positioned by its three corners.

This data type is typically used by civil engineers, and has some significant advantages. First, unlike a NURBS surface, determining points are generally coincident with the surface. This means that the data points in a mesh record positions on the surface: they are excellent corollaries to spot elevations. This also means that a small number of points can accurately approximate a terrain surface. Mesh surfaces also have significant disadvantages for encoding terrain. They are typically very angular, and tend to have large areas which are planar. Interpolation between data points is linear. This results in the many examples of engineered landform (such as near transit infrastructure) being graded to form vast areas of continuous slope—not a formal language well appreciated by most designers. Beyond any criticism

of aesthetic paucity, encoding terrain as a mesh enforces planarity without cause. Areas between spot elevations are deemphasized, formless, yet restricted. While continuous slope is excellent for calculating and directing sheet flow of water, it lacks the porosity of variegated terrain which can trap and absorb water, organic matter, detritus, and seeds—the stuff of tiny volunteer ecologies which encourage biodiversity in urban environments. Mesh surfaces lend themselves to engineered solutions, and can only approximate sculpted or variegated landform through translation. Consequently, a data type, a logic for encoding spatial information, has had a profound and ubiquitous impact on the built environment.

4.3. Form

4.3.1. Gesture

A gestural interaction with a digital landscape is not a simple challenge. To begin with, the typical computer interface is at this time of writing still two dimensional, regardless of the three dimensional data underlying the interface. There is no reason to doubt that this will change in the following decades, with videogrammetric gestural interfaces (like Microsoft Kinect), virtual reality immersive representation, and more responsive pressure sensitive tablet interfaces becoming better developed and more economical. There will perhaps come a time very soon when a truly three dimensional interface will be possible and effective.

In the meantime, systems have been devised to cobble together something like. ETH, for example, has equipment which allows scanning of sandboxes which designers can form with their hands (Walliss & Rahmann, 2016). While this workflow is potentially very enabling of loose creativity, it does not provide instant feedback or provide for the system to set limits to sculpted form.

A more valuable interface would allow a sculpting metaphor to enable edits which are subject to

particular sets of rules. For example, a terrain representation might enforce the limit that all slopes must be less than 5%, or that every fill action must balance with a cut action. These landscape design specific guides would facilitate creativity which ended with more direct applicability to the next phase of design development, rather than resulting in a concept which has to be translated into another medium. The efficiency of reducing that gap would be a significant added value to a design process.

User interface is by no means a simple or easy problem to solve, and many iterations would be necessary to find a set of interaction tools that would be intuitive, accessible, and effective for producing tenable design. It is essential to begin detailing the possibilities and attributes of such a system so that future generations of landscape designers can enjoy the efficiency of familiarity and unity with tools that integrate into the new way of doing business in AEC.

4.3.2. Rationalization

Finding elegant and excellent form in design begins as a loose process which implies and invites imagination. For the vision to become reality, however, it must be rationalized. That is to say, it must be brought into a rigorous assemblage of contiguous forms. This is certainly not a process that could be simply automated, and will always require the judgment and wisdom of the experienced designer. However, certain rules can be helpful in enforcing inherent attributes of real world landscapes. For example, continuous and adjacent landforms must share at least one coincident edge. Landform cannot overlap itself in vertical displacement without concrete or other constructed support. These rules are simple enough and universal enough to be encoded as inherent attributes of a datatype.

The next echelon of rationalization is replacing gesture with accurate delineation. Landscape features, which are often represented by their boundaries, will ultimately be documented in terms of one

dimensional linework which specifies exact placement. The rigor of such placement is often premature, but inherent in a digital representation. Thus it will be necessary for heuristic algorithms to resolve gesture into accurate objects which relate to one another precisely. For example, a land form which terminates at a retaining wall should stop at the masonry, and not intersect it. Another form of rationalization necessary to design modeling is construction and materials specification. A common example of this in the buildings modeling workflow, for example in Revit, is the task of taking a doubly curved surface and rationalizing it into an array of panels which are manufacturable at reasonable cost. Digital fabrication technologies and collaboration in BIM have greatly facilitated this procedure in realizing complex and sculpted architectural forms. Examples exist of similar rationalization projects within landscape architecture, notably the MAX IV nuclear reactor in Lund, Sweden, designed by Snohetta (Walliss & Rahmann, 2016). Examples also exist of a more traditional rationalization procedure beginning with clay model landform and moving into careful translation and documentation of the physical model into two dimensional paper construction drawings. In typical design practice at the time of writing, a more likely everyday workflow is to rationalize design as a requisite side effect of the documentation process (that is, drafting in AutoCAD). The inefficiency inherent in this process becomes clear when considering that the translation of sketches and drawings to CAD linework relies entirely on the provided expertise of the designers experience with such matters: at no time is the reason behind the decisions encoded into the documents produced. That essential, or even primary, piece of the design process is inaccessible to the typical modes of landscape representation. I propose that this contributes to over-reliance on simplified diagram and glossy narrative: the nitty gritty of decision making is inaccessible to our representational and therefore communicative methods.

An ideal landscape platform will therefore encode the rules or heuristics used to make decisions about adjacency, placement, etc.

4.4. Analyze

A very powerful suite of functions in a unified landscape software tool would be focused on simple quantitative analysis of design exploration. Civil3D has several hydrological analysis tools, which are of great value for exploring engineering performance. However, their functions are limited to certain particulars, most notably surface flow. A more highly developed and broadly applicable suite of spatial analytical tools is available in the ArcGIS suite. Solar exposure analysis, and viewshed analysis are two prominent and highly useful analyses which can give valuable feedback as to the consequences of grading decisions.

One prerequisite to this sort of function would simply be a ‘snapshot’ function allowing comparison of different design configurations. Testing multiple design possibilities should be a core functionality of such a platform, allowing the designer and the client to investigate and record advantages and disadvantages of each possible configuration. Further function might provide metadata or database entries allowing evaluative data, comments, and cloud sharing to take advantage of the snapshots. That data could be leveraged for comparison in data analysis platforms from MS Excel to R to more sophisticated implementation of the latest machine learning libraries.

While LIM software might provide native analytical tools, it would be very useful to provide an interface to allow using existing libraries from other software packages. One promising avenue for this kind of facility is shown in Dynamo, a visual scripting platform that interfaces with Revit. Conceptually similar to Grasshopper, Dynamo provides the capability of extending Revit’s function into custom scripts represented as a flow of components. As of more recent versions, the software allows the direct import of functions programmed on Microsoft’s .NET

framework. That subset of publicly available software is significant, and includes implementations of key spatial libraries such as GDAL. Rather than redesigning such capabilities from the ground up, it makes more sense to harness the open source energy so vital to the Grasshopper community. The creativity and imagination of hundreds of designer / coders around the world can outpace any design team at a major vendor. Another very solid collection of analytical tools has been developed over time by the QGIS community. An open source alternative to digital mapping software, this platform provides robust spatial analysis tools available to all.

4.5. Perform

Landscape metrics should be distinguished from landscape performance. Calculation of flow accumulation might be called a landscape metric, but interpretation is required to incorporate any insights from the raw information. While implementing metrics into an interface with a landscape information model presents few conceptual obstacles, performance feedback may prove a more complex undertaking.

Not least, this is due to the fact that desired performance outcomes are likely to incorporate values statements. Hence a landscape information model cannot, by definition, record “good” or “bad” performance, only so many metrics. Establishment of a performance methodology will generally be site specific to some degree, as well as depending on the needs of stakeholders. Hence it makes little sense to try to automate any universally applicable methodology. Rather, a more modular approach would provide flexibility while allowing designers to control priorities.

One promising avenue for building on simple analytical metrics is now being explored in computational architecture: multivariate optimization. As mentioned elsewhere, these algorithms narrow in upon optimal balance among defined goals without testing every permutation. A platform which included goal searching or evolutionary algorithms in its analytical

mechanism would provide designers assistance in finding possible solutions to complex problems—more quantitatively focused problems such as grading or stormwater facility sizing would be more easily encoded algorithmically.

Performance measures which consider social or ecological systems would be more challenging to incorporate. While experiments in this space legitimize Margaret Grose’s interest in using landscape simulation in this way, implementation in a live project would be difficult. Much of the existing development in computational approaches to ecology, for example, has taken place in a more traditional or open source coding environment. Effectively incorporating ecological models into a landscape information model would not be a description of a plugin, but of a consulting firm.

Part of the difficulty lies in the conflation of the term “model” in the respective disciplines. The landscape information model consists fundamentally in representing what there is and tagging those things with attributes. A more interactive software would vary this by allowing representations of what might be. The model remains representational. Ecological models, instead, refer to a set of mathematical functions which share one or more variables and rates of interaction. The model is a good one to the degree that it is predictive or explanatory of observed phenomena. Hence connecting these two paradigms is by no means as simple as matching up the variables and hitting the Go button. Rather, model making in ecology is a rigorous, iterative, complex discipline developing problem specific implementations of controversial mathematical simplifications of incompletely understood real systems. The chances of integrating this workflow into a typical landscape practice are not promising.

Instead, an ideal landscape information platform would focus on simply informing the user and facilitating the user’s proposals. Performance would continue to be something argued for with the available

information, rather than being deterministically quantifiable.

4.6. Iterate

Design process is typically characterized by extensive iteration. The design process taught in landscape education requires iteration at the level of the individual student, as well as after feedback from peers or instructors. This training imbues the habit of trying ideas, examining the results, and trying further refined edits. Because great designs do not generally emerge fully formed, but are repeatedly massaged into final form, an efficient and comfortable habit of iteration is essential to the individual designer’s practice.

As any design student knows, iteration is limited chiefly by time. This means that tools and processes which enable quicker and more informative iteration will facilitate more iterations and possibly improve results.

The professional environment is no different, but adds many additional complexities. Collaboration requires iteration, as feedback is received by different members of a design team. This is true at both the scale within a firm as well as scale between firms. When firms in different professions collaborate on a project, they will necessarily bounce representations back and forth. Each successive communication may prompt another iteration.

An effective LIM should therefore incorporate a functionality allowing management of iterations. A states manager would provide this function. Another scheme would provide scenario comparisons. Whether in the form of quantitative comparisons or visual comparisons, some kind of feedback regarding the marginal result of an iteration would be highly useful.

Iteration would also be helped greatly by a structure of dependencies. While dependency would not be applicable in every context, there would be huge

time savings in the possibility of propagating a single change to dependent geometries. Hence an edit to a terrain might automatically update a planting plan or an area of hardcover.

The bottom line for integrating these functions into LIM software would parallel the value proposition in BIM vs. traditional 2D CAD. That is, less time spent representing, and more time spent designing. As the IFC format improves better collaboration between engineering and design, one payoff is more efficient iteration. This allows a greater share of the time and fee available to be used on design thinking.

4.7. Share

LIM software should provide for effective sharing of the proposed design. This would take several forms.

As mentioned above, LIM should integrate well with other software used by allied disciplines. This suggests that building on the IFC format is essential. A project such as Gill’s does not afford much possibility of professional use, because it is not oriented around sharing common formats with engineers or architects.

Additionally, sharing one model among multiple designers will be essential, perhaps through a system of ‘check-out’ similar to the coding world (eg. Github) or as implemented in Autodesk Revit. This requires a network deployment or a cloud deployment. Autodesk has already undertaken such a cloud solution, which fits well with their business model and constitutes a reasonable paradigm for the future of information modeling. LIM will have to integrate with this paradigm.

Sharing functionality should also provide client facing features. For example, simple and attractive 3D and 2D views will help communicate about design ideas without suggesting that the design is a finished product. More developed designs could be the basis of more advanced rendering, whether in a native featureset or through integration with a tool such

as Lumion. Regardless of the level of development of the design, a simplified interface designed for non-experts would be strong, so long as the simplification were done gracefully so as to retain sufficient detail and interactivity.

While these sharing features are no different conceptually than those already implemented in other BIM software solutions, they are essential in software implemented in a for-profit environment. As such, they should inform development of other features. They should also be robustly developed rather than deemphasized. Long term, there is no sustainable BIM solution that does not share well.

4.8. Document

Documentation function will be absolutely essential to an adequate landscape modeling platform. A model has some use in its visual representation. In a professional context, however, one of the core value propositions of information modeling is the time savings in documentation produced automatically from the digital model. This allows designers to spend a larger proportion of their billable hours engaged with conceptualization, design, and detailing. Hours which had previously been devoted to drafting are now devoted to further iteration of design ideas. Hence a useful landscape modeling platform must be able to adequately translate stored representations into a form which is easily understood by contractors tasked with building it.

Documentation may not require thick volumes of printed paper for very much longer. The most common form of contract documentation is of course printed paper technical drawings specifying dimensions and materials of each element. A major efficiency provided by Revit is the automation of this translation from 3D objects to printed linework. Such a transformation becomes unnecessary, however, if the paradigm of measured drawings inherited from architecture history becomes obsolete. If the contract drawing is replaced with a model, the contractor

is free to innovate in how laborers are given the specifications of individual elements. The result of such innovation may ultimately be to completely phase out the measured drawing on paper as the binding documentation of what is to be built. More ubiquitous virtual or augmented reality technologies could easily be imagined as workplace commonalities. In this scenario, a heads up display would allow contract labor to explore the contract model on site as the project is built.

In the meantime, some form of 2D representation is necessary, whether orthogonal printed drawings or more high tech solutions like tablet computers on the job site. The automation of that translation is a key function. While the translation entails some liabilities, these are well understood by practicing professionals and allowed for by present practice. The codification of practice in documentation standards followed across projects at a given firm provides consistency and systematization which eases clarity and accuracy. Automation of these standards adds value to an information modeling platform, but requires consistency in translation. That is, a standard will allow a 3D object to be represented in the same way throughout a document. This may be more difficult to standardize if the 3d objects are not purpose built for the elements they represent.

Section 5: CASE STUDY: STUDIO DESIGN PROJECT

5.1. Process

To explore the possibilities for a more direct union of creative with analytical aspects of design process, I engaged a 10 week studio design process. While the studio was not long enough to support full investigation of the schema discussed above, it did inform my understanding of the challenges of a terrain-first approach. By working in design, engineering, and GIS software, I was able to learn the points of friction while looking for opportunities for better optimization. The studio, conducted at the University of Washington, focused on an urban block in the city of Seattle in the heart of a growing central neighborhood (Figure F).

The object of my exploration was as much the design process as the final design. The project served to inform my understanding of the state of the art of various industry standard tools. My intention was to flesh out something of an iterative cycle facilitating steps from gestural, creative thinking, through rationalizing the ideas into concrete spatial proposals, evaluating those proposals in 3D for quality, analyzing the terrain for performative dimensions, and letting the results of the analysis inform the next iteration. As such, this project is an example of an economization through better cooperation between digital tools (Figure G). The design itself was the result of a more traditional attention to form and program, with few opportunities for relational or generative modeling.

After a good deal of tinkering, the process I used came together in the following way (Figure H). Using a mental model (my imagination) of the site, I drew lines with pencil and pen in plan view to outline a design. As shown in Figures H and N, the imagination does not fit tidily into any functional taxonomy of design process, but rather underpins and enables the process regardless of which tools are used.

PATIO

GRAND STAIR

GREENHOUSE

P-PATCH

ORCHARD

SWALE

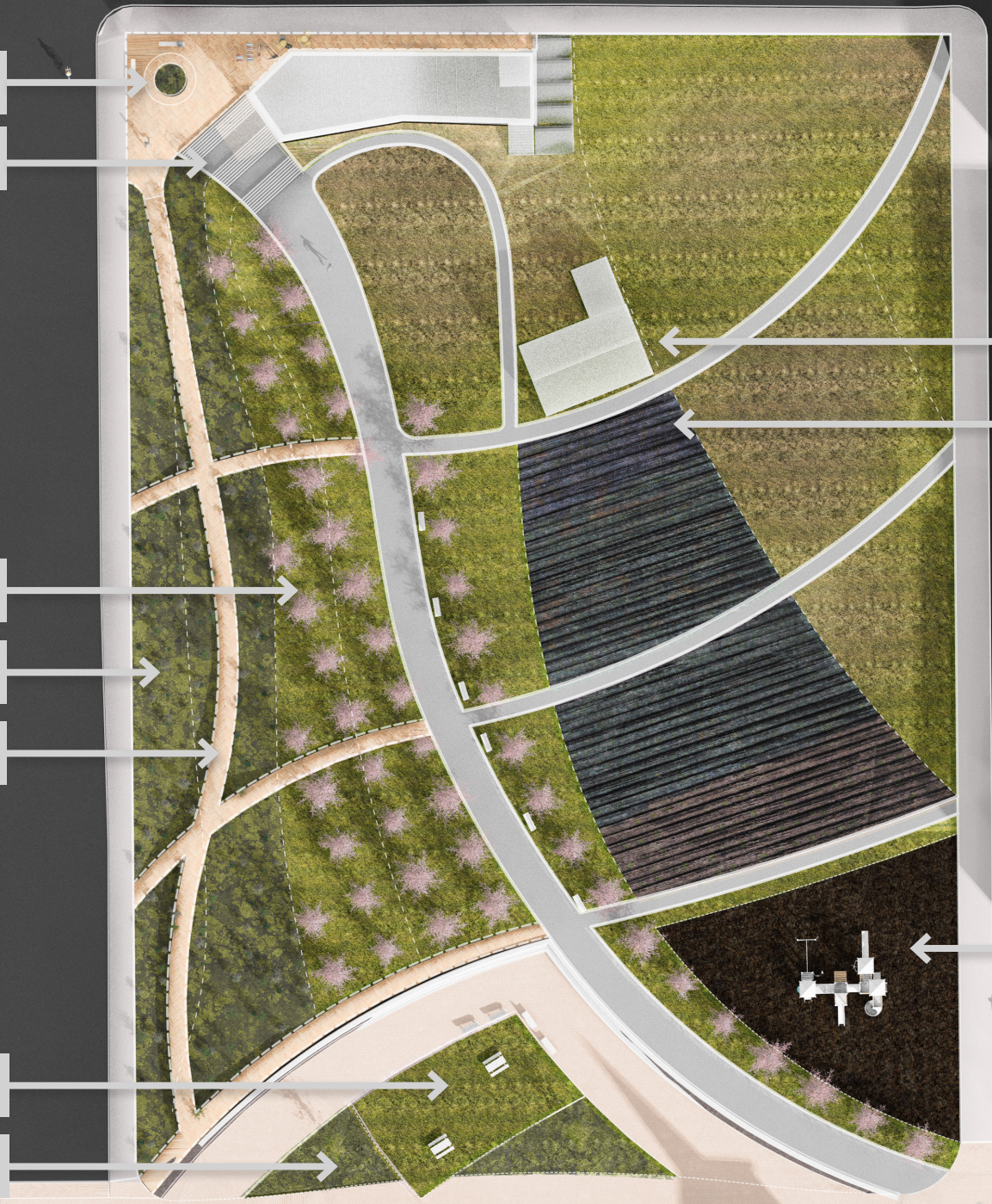
BOARDWALK

PLAYGROUND

LAWN

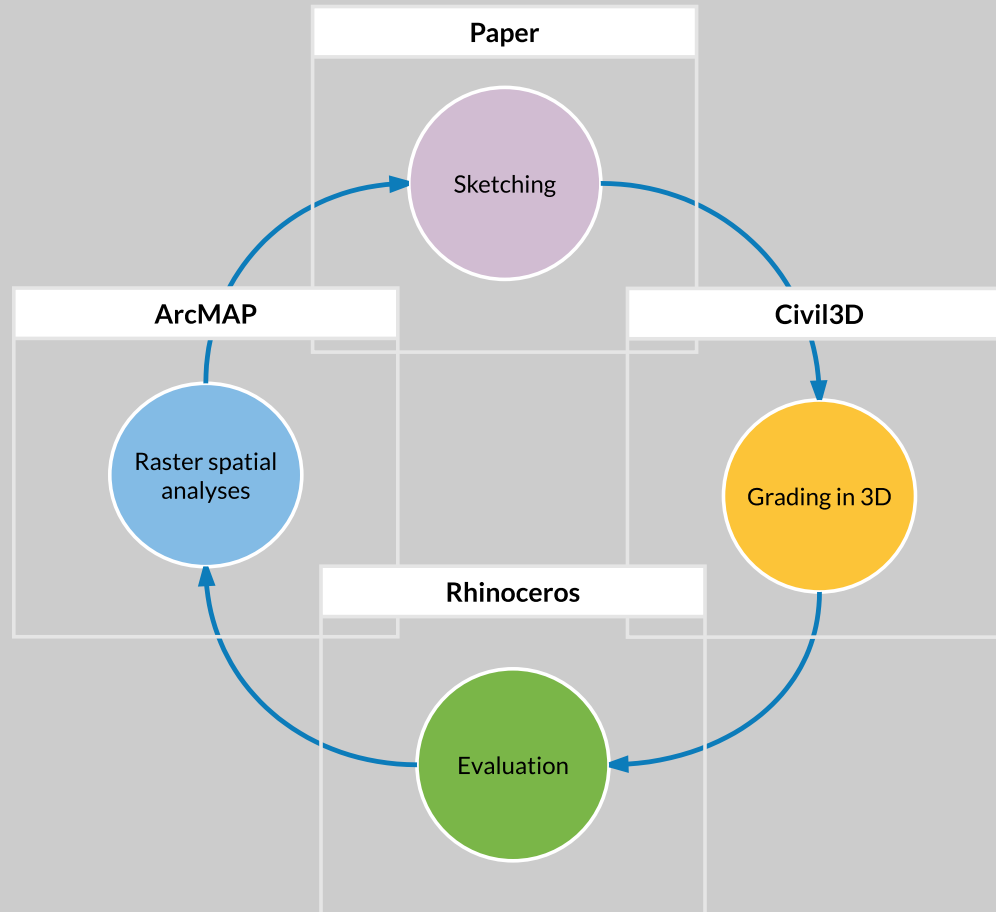
RAINGARDEN

WOONERF



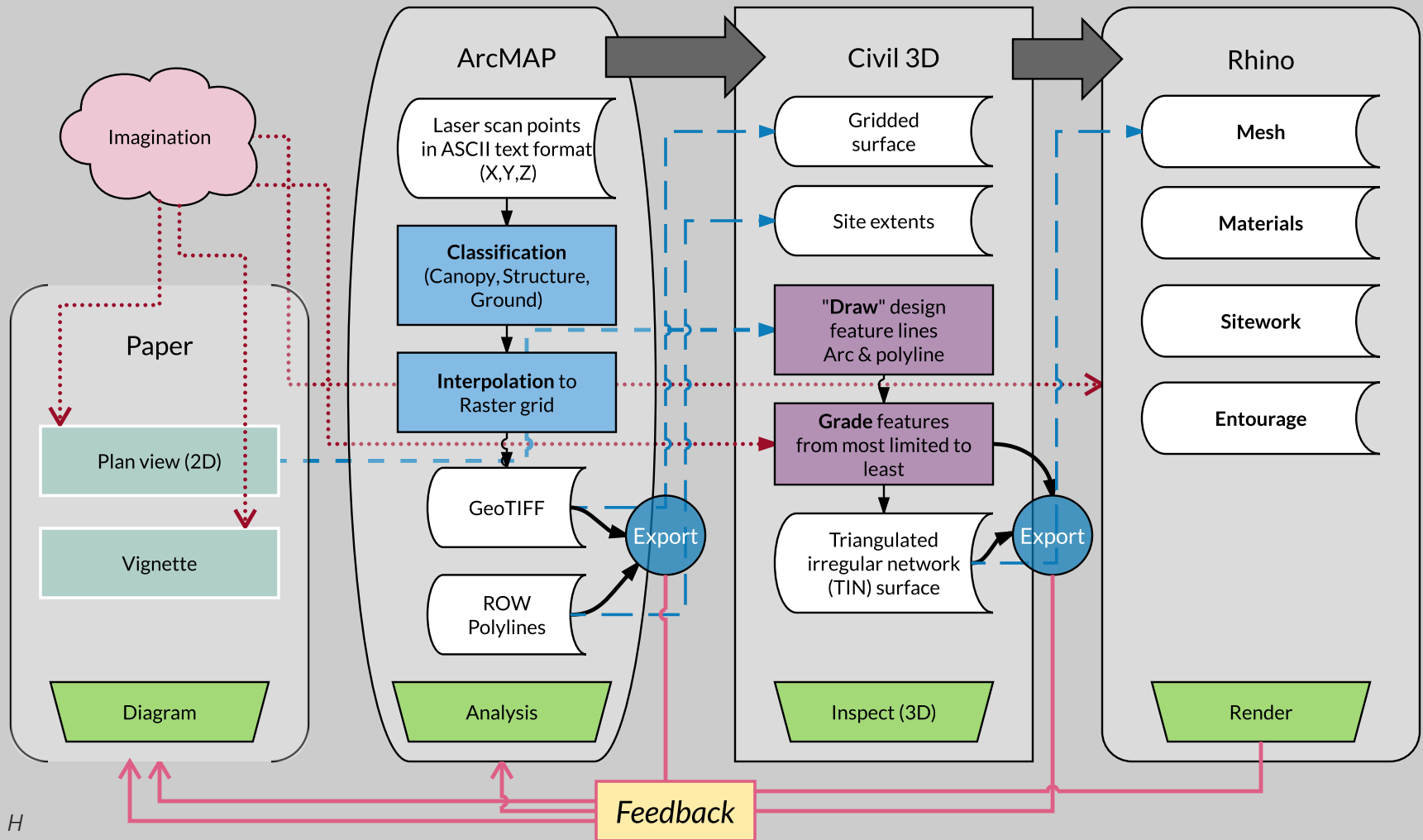
F

PROTOTYPE



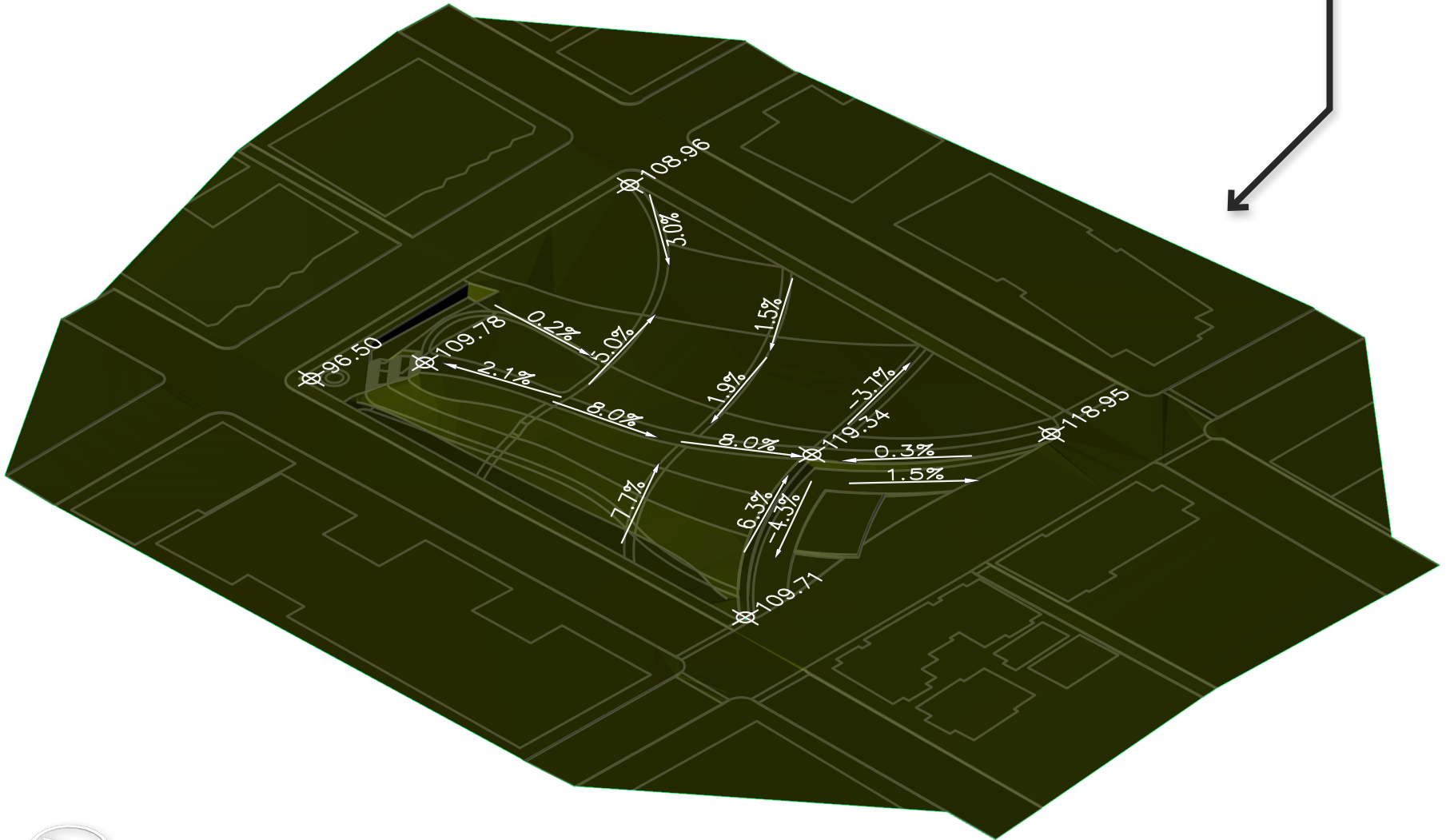
This flow chart represents the iterative process used in the studio project represented in Figure F. Unlike the 'ideal' paradigm shown in Figure E, this cobbled together process required heavy manual editing and translation to complete the circle.

INFORMATION FLOW



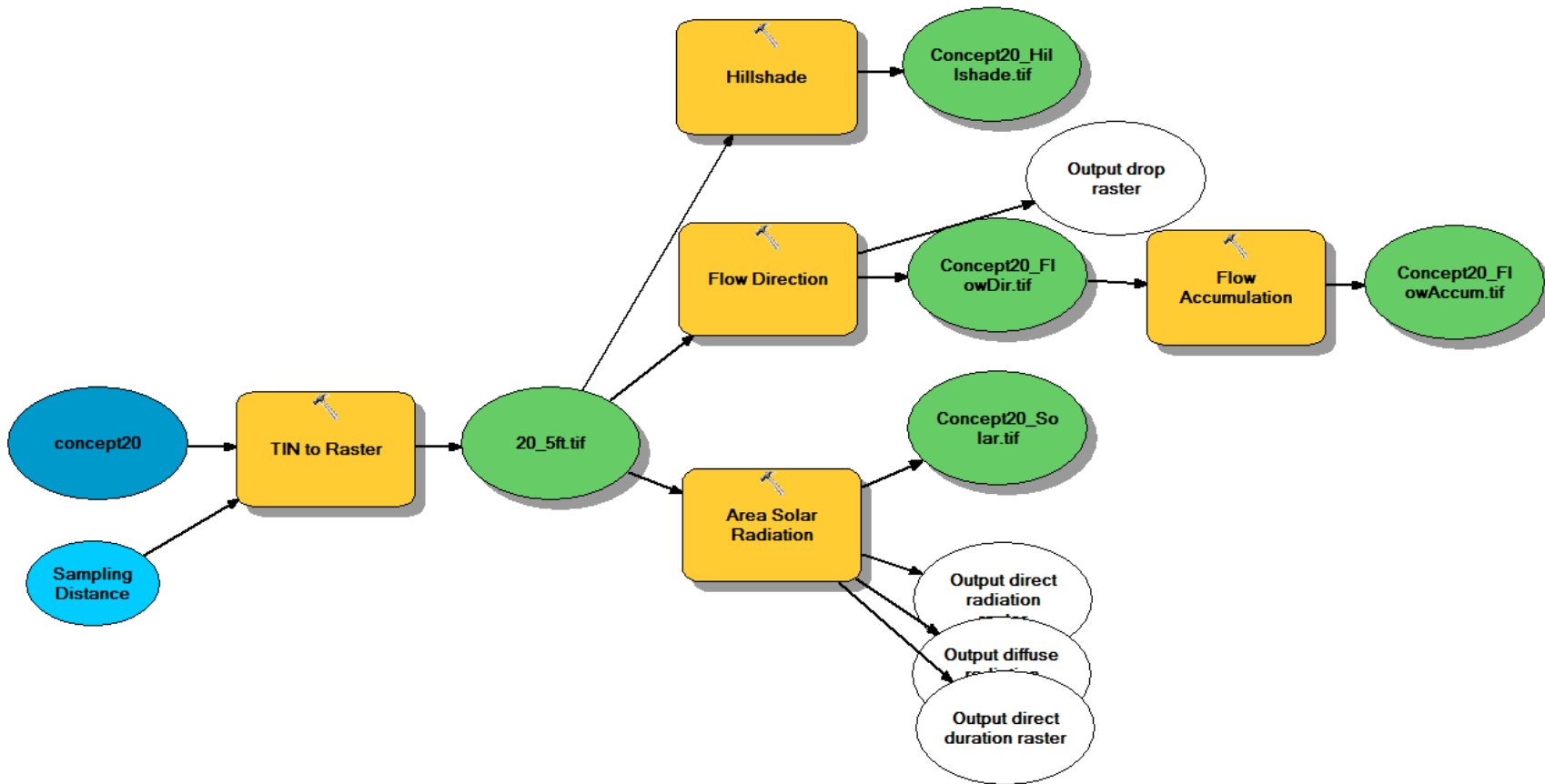
This chart represents in further detail the flow of information between platforms and data types. Each tool's strengths provide an opportunity for a feedback loop, although this was typically achieved manually.

GRADING



This grading diagram shows the major decisions in the Civil3D surface workflow. Each section of path was graded to cultivate particular spatial relationships and experiences, while maintaining limits within ADA guidelines.

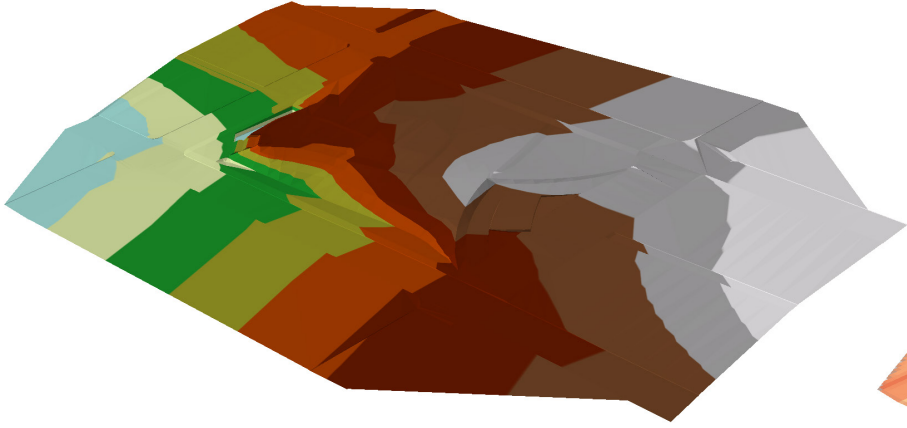
TERRAIN ANALYSIS



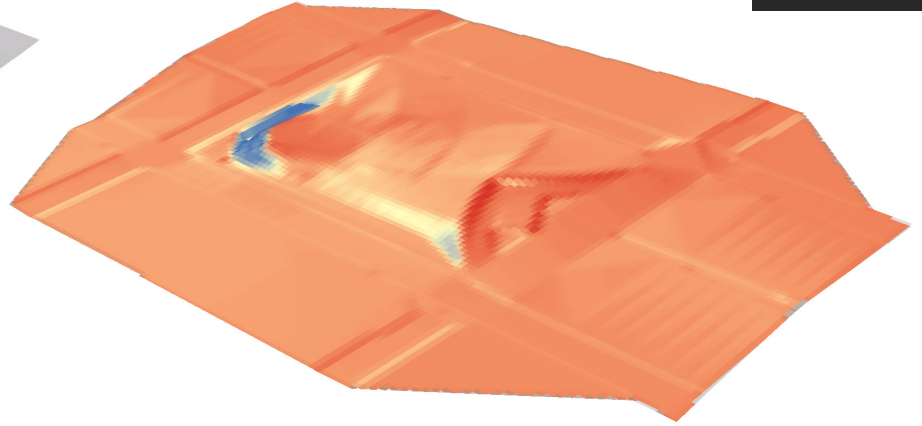
K

This ArcGIS model facilitated raster analysis of the landform, including solar gain and hydrology.

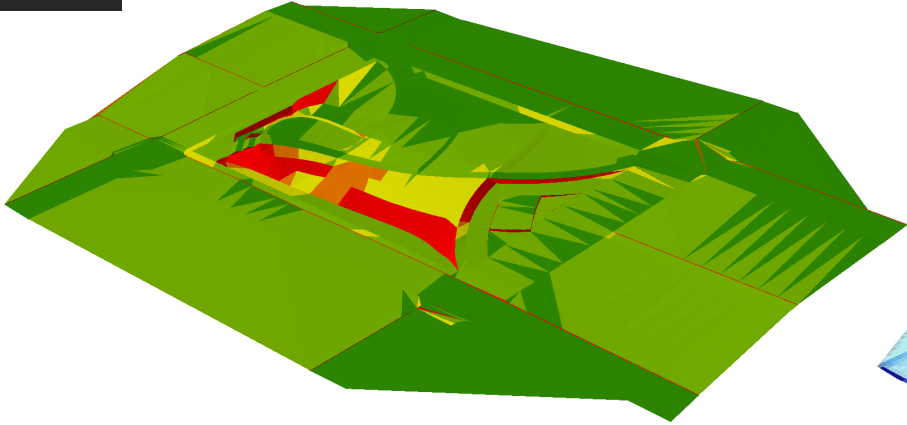
ELEVATION BANDING



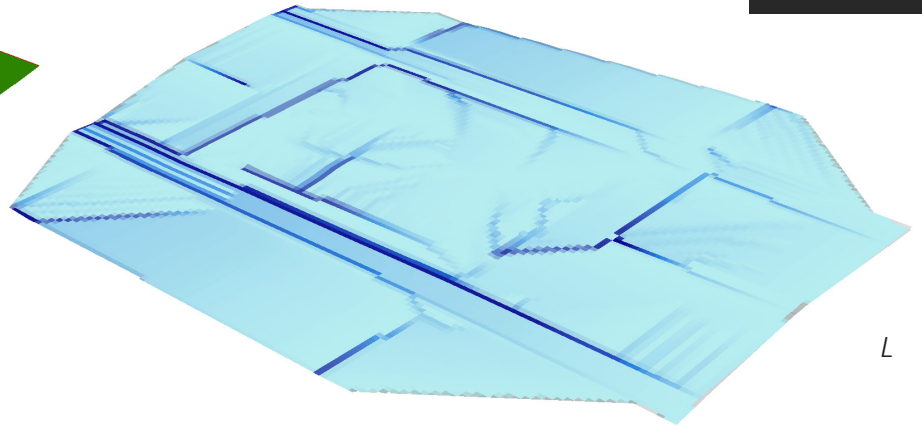
SOLAR GAIN



SLOPE

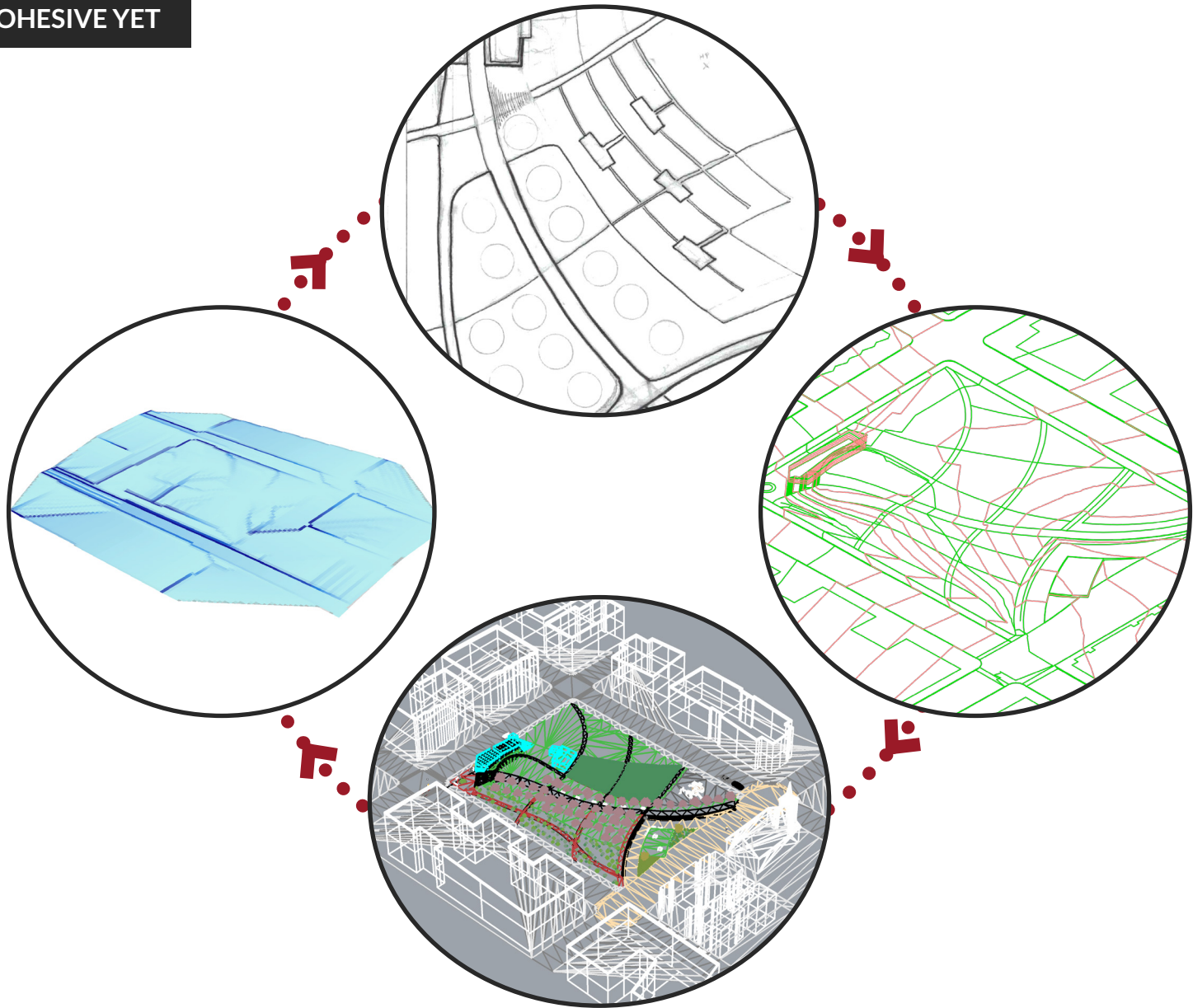


FLOW PATHS



L

NOT COHESIVE YET



A scanned version of that result formed the basis for drawing in CAD (first Rhino, then also Civil3D). The vector linework allowed me to use Civil3D to manually grade each segment, enforcing ADA regulations manually, moving from most limiting areas to less (Figure J). Having completed a draft Civil3D surface, I exported the surface as LandXML format. I was then able to use ArcMAP to import each proposed surface as a triangulated irregular network surface. Using grid subsampling, I converted each surface into a digital elevation raster. This raster was the basis for several stock GIS analytical algorithms. Using the Model Builder visual scripting environment in ArcMAP, I produced raster images for each analysis. Specifically, I used flow accumulation, hillshade and solar radiation to study the performance of the terrain surface (Figures K, L). I also devised a simple visibility methodology to test for blind spots in the proposed terrain to help understand the design from the point of view of Crime Prevention Through Design (CPTD). The array of raster images were exported for visual inspection. These results informed the next iteration of design development along with attention to aesthetic and programmatic concerns incorporated through sketching and desk critiques with the course instructor.

This process is parallel to those made possible by the tools detailed in the case studies above. This process limited the role of computation to an analytical one. The distinguishing characteristic of my process was reliance on existing tools and effort to streamline a quantitative approach using industry standard software platforms. Compared with EpiFlow, Belesky's Flows script, or Gill's modular analytical models, this process has certain advantages and disadvantages. First, because it used tools which are common to the professional environment, it could require less overhead or investment in a for-profit practice. That is to say, professionals who are already familiar with the use of Civil3D and GIS would be able to use this process with little training or software acquisition. Learning a new software platform such as Grasshopper would not be necessary. Another advantage is that using Civil3D makes this process

easy to integrate with the collaborative process with engineers. One major disadvantage is the amount of manual work required. Using several software platforms creates overhead in moving data from one to the next. The ideal of a unified platform for creativity and analysis is not met (Figure M). Another significant disadvantage is that manual grading in Civil3D is a time consuming and precise process, ill suited for broad design moves.

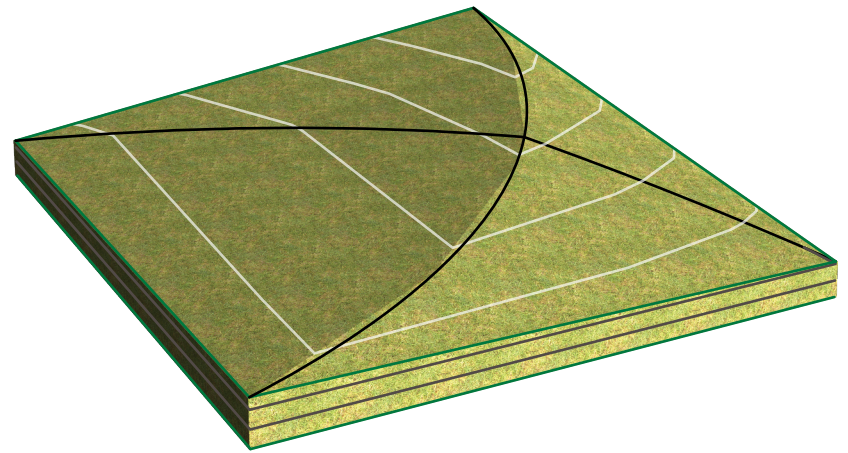
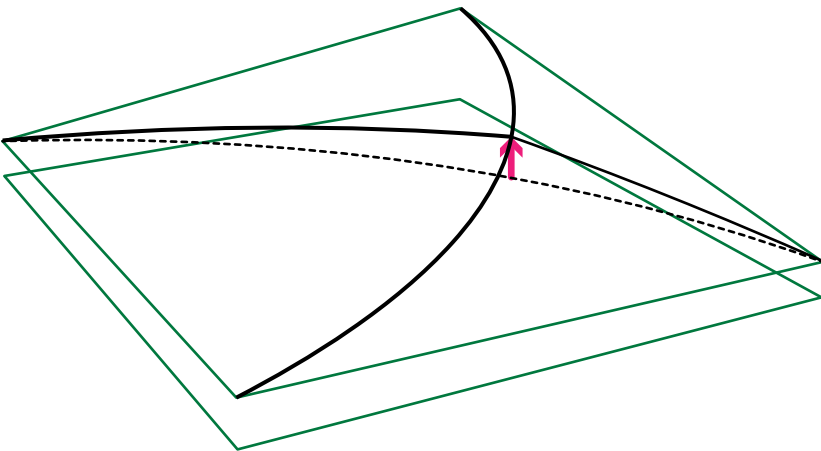
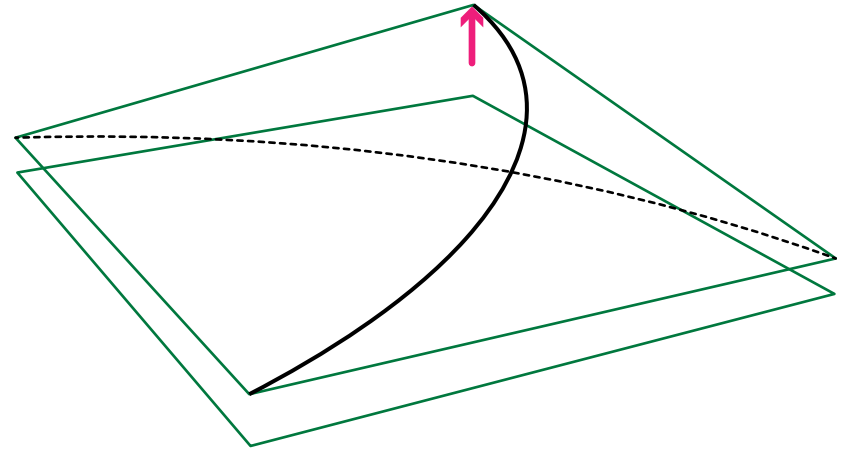
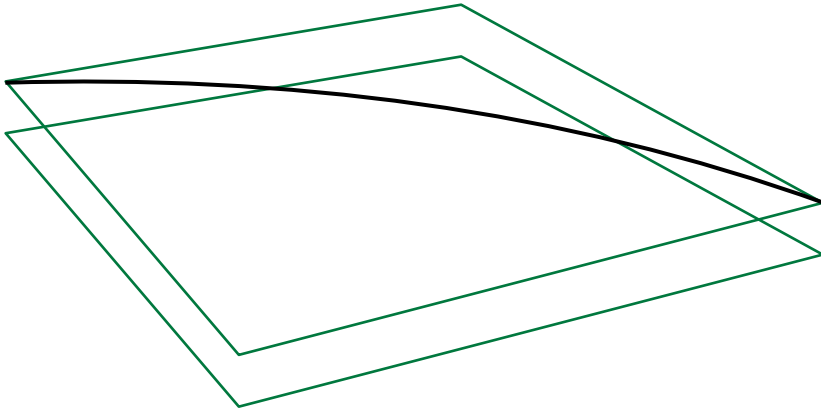
There is little opportunity to extend this workflow into a parametric schema. That is to say, the results of analysis are interpreted manually and worked into the next iteration by the effort of the designer. Because the form generation is done manually, there is no opportunity to optimize design moves parametrically based on analytical results. Likewise, only Civil3D provides a degree of relational intelligence. Updates to one segment in the network of curves propagate to contiguous areas of the surface, but more complex dependencies are not possible.

The exercise was an instructive one by showing the functionality of Civil3D which is uniquely topological. Here I use the term in its more traditional mathematical sense. Civil3D enforces certain rules which determine invariant qualities in the surface geometry. As such, Civil3D enforces a surface topology. Specifically, this requires that intersecting lines on the same surface will always share X, Y, and Z coordinates at the point of intersection (Figure N). This is an essential requirement to designing a valid terrain. This may seem an obvious point, but it is a unique function of the software which facilitates terrain design. This fact is suggestive of a possible way forward for experiment in a more direct relationship between designer and terrain. Civil3D also provides some dependency functionality. For example, feature lines can be dependent on an alignment (a technical term in engineering), such that changing the alignment updates all child geometry. The combination of these affordances allows the designer to make upstream changes to a design which are automatically propagated to downstream changes, while the resulting terrain is

updated according to a surface topology. It is easy to imagine a further development of this kind of workflow in which more sophisticated dependencies can be created, or in which attributes of the parent object can be algorithmically optimized to approximate one or more requirements in child objects. For example, the length of a switchback path could be determined computationally by the required elevation of the termination while the route of the path could be determined by minimizing cut and fill. An evolutionary algorithm would be well suited to such an endeavor. While a simple example like this would probably be easier for an experienced designer to simply sketch out on paper over contours, more complex dependencies and limits in a graded network could be optimized very quickly once the basic logic was implemented.

This line of thinking suggests a bias which I discovered in my process, which I believe to be somewhat inherent in Civil3D, and possibly in engineering practice in general. That is, the approach of optimizing a network of paths emphasizes the skeletal structure of the landscape as seen from the perspective of circulation. It is my experience that this bias is often true also of sketching on paper, where the linear nature of pen marks lends itself to drawing linear landscape features. While circulation is doubtless an essential element of a successful landscape, approaching every design as a composition of linear features does not guarantee good spaces. In fact, my project demonstrates that an overemphasis on the linear features can lead to spaces which are relatively arbitrary in their form and somewhat disconnected from one another. In my case, this bias produced a continuing tendency toward a sort of 'paint-by-number' approach to the site, in which different spaces were assigned a program and designed as somewhat independent spaces. This weakness is representative of a similar characteristic typical of the built environment we are all familiar with. In the United States, nearly all of our inhabited land is sliced up into rectangles, then bifurcated by linear circulation features. I argue that the very fabric of our built environment is the

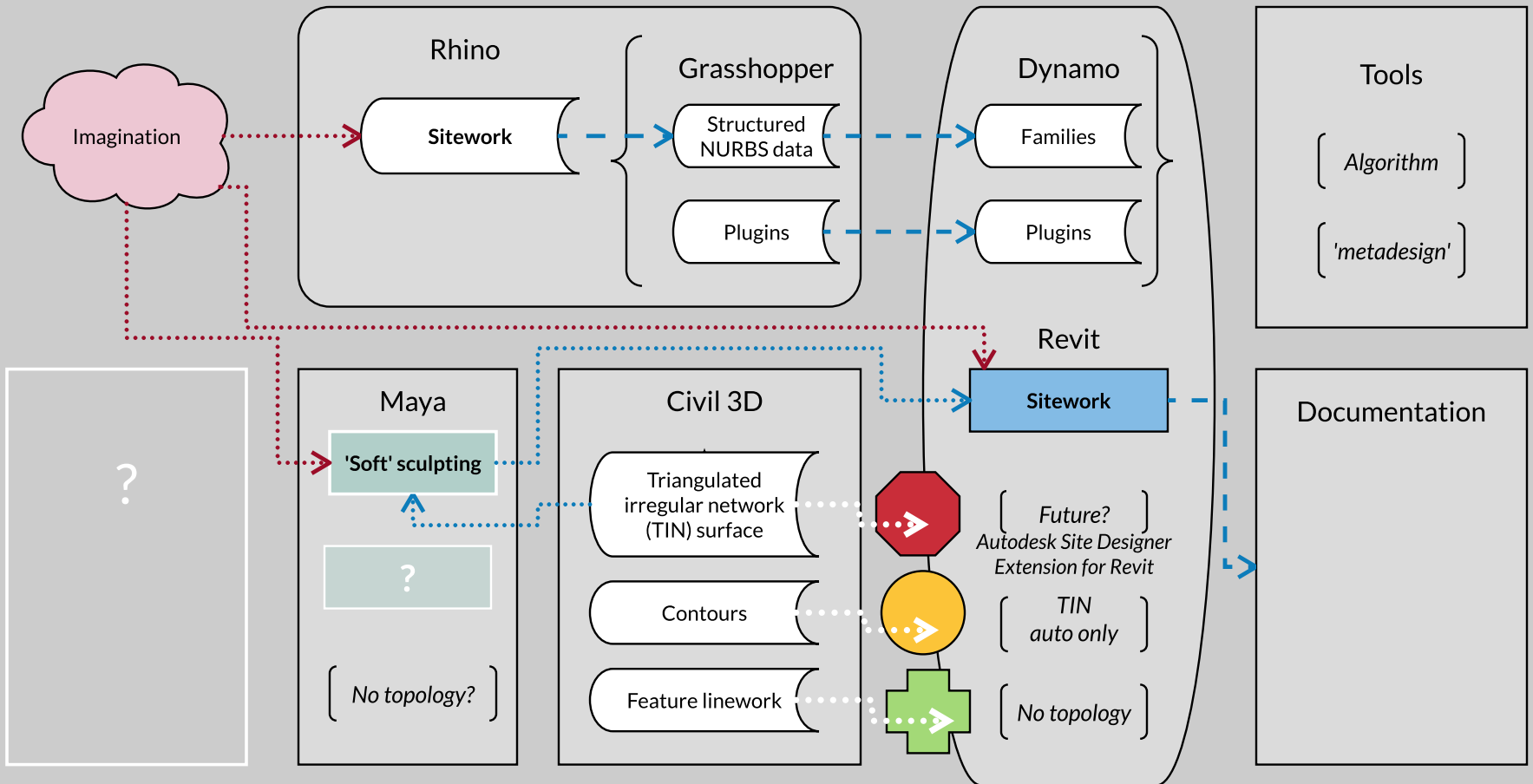
TOPOLOGY



N

Shown here is the value of a topological treatment of terrain geometry. Topology is the mathematics of adjacency. In the case of a surface topology, intersecting lines must share coordinates in all 3 dimension at the point of intersection. Civil3D enforces surface topology.

FUTURE POSSIBILITIES



This diagram represents directions worth exploring with further research. This may be in the form of individual toolmaking experiments, significant open source community based innovation, or major vendor software updates. Other tools not dealt with in depth here are worth consideration, particularly AutoDesk Maya.

result of the dominance of engineering as a design approach and as a way of thinking. The city of Seattle is a perfect example of the consequences of this approach. The city is split longitudinally by an Interstate which divides what were once cohesive neighborhoods. The application of a grid strategy for spatial organization in spite of the prevalence of shoreline in the city has caused the phenomenon of the 'street end', a peculiar urban typology which is now being creatively developed as the value of open space rises. While these peculiarities present interesting design opportunities, they should also underscore the degree to which the substance of landscape design work is an attempt to create good spaces and connections across or in spite of impassable linear features. This uncompromising and single minded focus by engineers on linear organization of space is only confirmed and further entrenched by the architecture of Civil3D. Hence, while the idea of an algorithmically optimized circulation network is an exciting one, it is not conducive to a design approach which begins with the desire to create great spaces first and foremost. It inherently prioritizes the boundaries of spaces, while forcing the spaces themselves to take up the slack. It is not necessarily bad, but it is a limitation to the conceptualization of site which novice designers are continually struggling to address.

By contrast, some lower tech methods of direct interaction with terrain do not share this bias. For example, working with clay is a very different metaphor for landform and for the space that rests upon it. It is additive, rather than topological. To elaborate, this means that the "poetics of the place" can be shaped from the middle outward, rather than by subdivision of a parcel. Clay is indeterminate with respect to boundaries. It can be gestural as well as precise. It tends toward continuous curvature in both dimensions, rather than assemblage of planar facets. Sculpting landform from clay has its own biases and drawbacks, but is useful to consider here as a foil for understanding the biases of design through topological linear networks. Still, because linear math is

simpler than volumetric, practical implementations of computational optimizations are likely easier to conceive and implement for a linear approach to landscape architecture.

While this design project provided significant opportunities to test ideas, the resulting process did not depart from typical heuristic ways of thinking. Using more rigorous analytical tools was valuable feedback for imaginative and instinctive decision making. Finding an efficient workflow capitalizing on the strengths of each platform also enables future use of the more integrated approach, while revealing opportunities for better integration. Innovation in design process is neither simple nor universal, and this project demonstrated that conceiving an ideal design process or a unified platform does not make either more likely. Rather, the expediencies of working under a deadline, and of the assigned deliverable (the design) taking priority over any procedural optimization, make getting it done the most important thing. This is borne out in conversations with professionals, who are intrigued by or excited about innovation, though time and fee structure limit what they can reach for. This fact recommends a distributed approach to innovation in tools. Modular thinking, as is typical in the open source development community, allows big problems to be broken up into smaller ones, with each solution made available as a building block. Experimentation with the building blocks for new tools and functions can begin to bridge the gaps between existing software packages.

Section 6: TOOLMAKING EXPERIMENTS

6.1. Overview

A primary value of the distributed paradigm for innovation in technology is the ability to redefine the terms of the problem and of the approach. When designers content themselves with using the platforms available from major vendors, the likelihood exists of certain limits to how practitioners can frame a set of problems. By taking on toolmaking, designers can gain the ability to define the problem in their own terms; design community can support deliberate engagement with the nuts and bolts of process when design professionals are empowered to extend and develop new media. Rather than specifying a unified platform which may be more or less ideally suited to an individual project, the distributed innovation schema exemplified by open source community has been very fruitful in the architectural design world, if only thus far for buildings. As an exploration of the toolmaking conceptual approach, I built several simple algorithms to facilitate direct interaction with terrain. Because there is a significant lack of terrain shaping tools in BIM, and because of the theoretical primacy of terrain in Girot's ideas, terrain representation and editing seemed a worthwhile function to focus on. Better function in dealing with terrain will be essential to better integration of design and engineering in any LIM environment. The experiments that follow inquire into the essential paradigms or metaphors we use when we represent. While none of these prototypes is developed enough to be used in production, each of them suggests possible directions and reminds of particular challenges.

6.2. Terrain editing experiments

Grid based elevation data is widely available and typically of reliable accuracy. While the derivative data is often in the form of a point cloud in the case that it was collected by laser scanning, raster data

SURFACE FROM RASTER

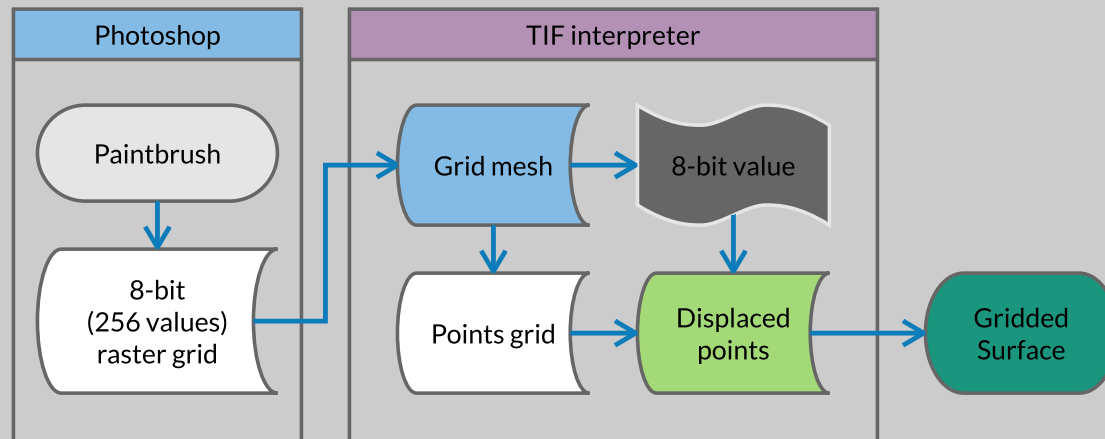
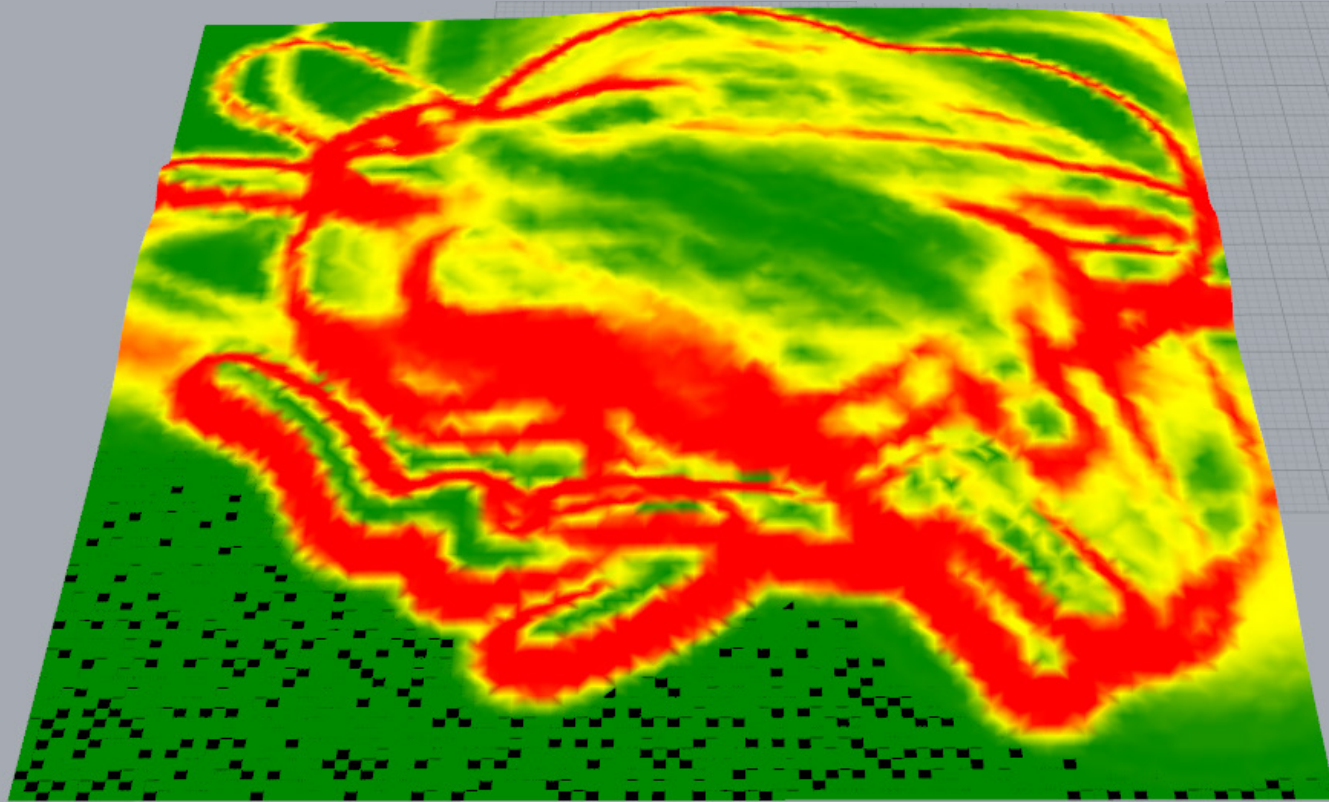
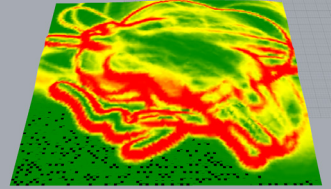
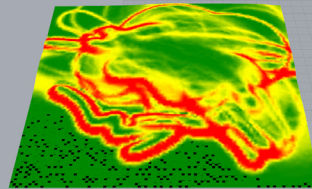
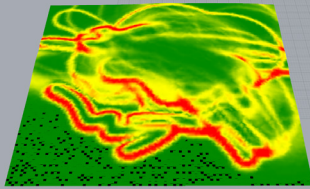
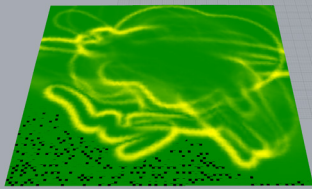
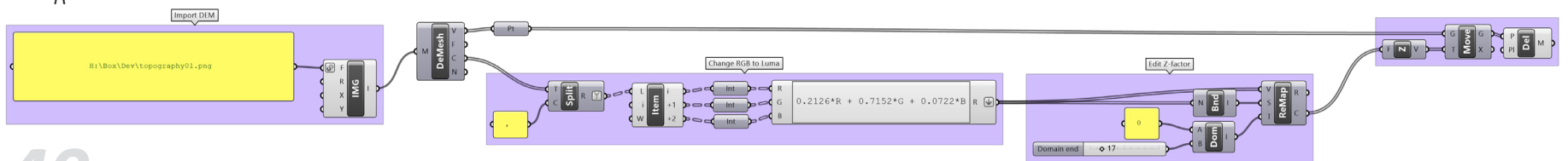


Figure P: a conceptual flow chart of the function of the script building a terrain model from a raster; Figure Q, frames of various translation multipliers; Figure R, the translation script in Grasshopper.



Q

R



GRID ATTRACTORS

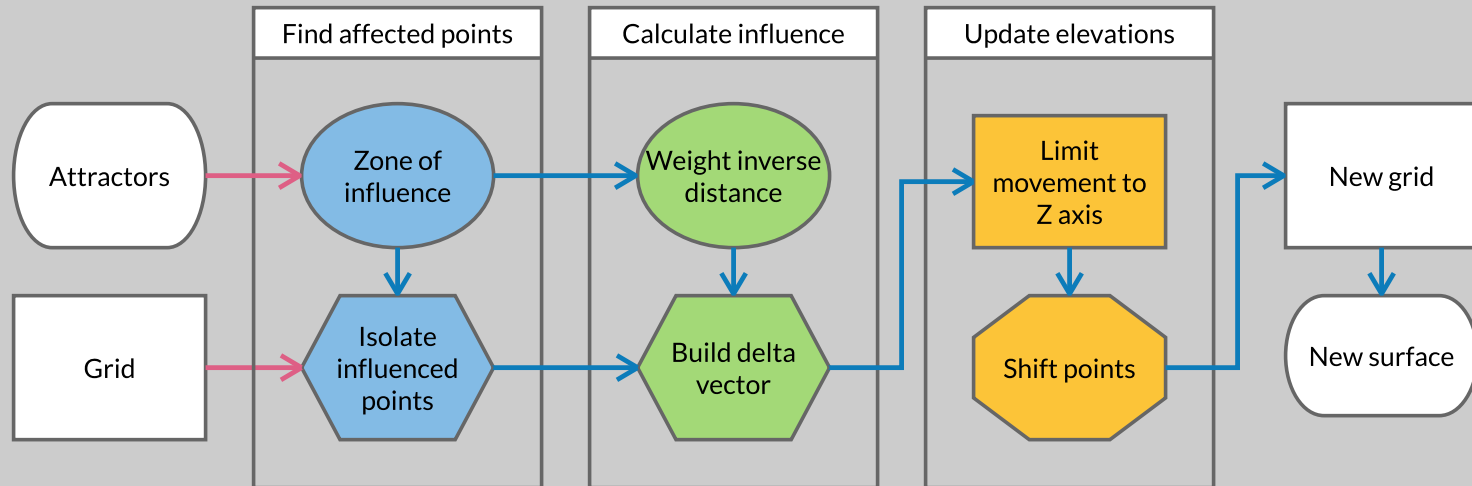
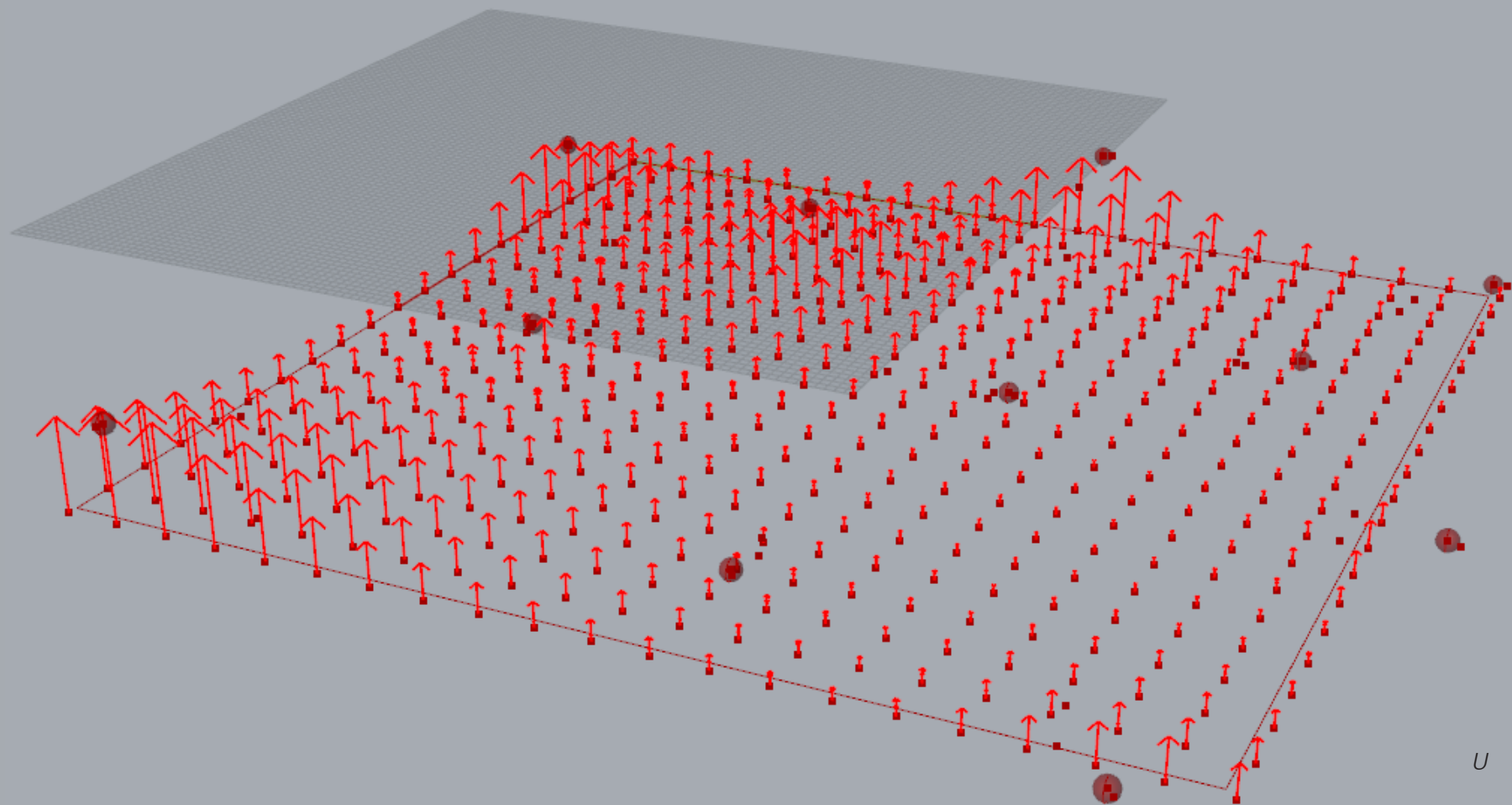
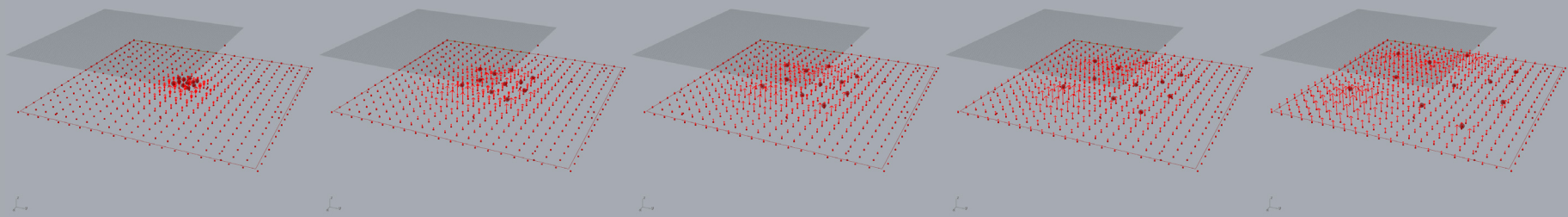


Figure S: a conceptual flow chart of the function of the script adjusting a points grid with vectors; Figure T, the Grid Attractor Script; Figure U, screenshots from the Grid Attractor in operation.



formats provide a very compact and universally readable encoding of terrain. While some software is designed to interpret and interact with DEM data, the best freeware libraries available for reading and writing GeoTIFF format are not well integrated with architectural design platforms. There are plugins available to begin bridging this gap, but a more plug-and-play capability between, for example, GDAL libraries and the Dynamo platform would open up new possibilities for designers to interact effectively with the data that is commonly available.

One experimental avenue engages with existing data and works to reduce the friction of using common data types or spatial representations in a design environment which is more accessible to design thinking. This approach underlies my sketch of a DEM interpretation algorithm (Figure P). In the taxonomy proposed above, this tool works to encode terrain and to form it, providing also a rudimentary analytical layer. Using a greyscale grid as a basis for representing terrain shares similarities with Lewis Gill's approach of procedurally computing the model based on 2 dimensional data plus attributes. The advantage of this approach is that the data is relatively compact: large areas of terrain can be represented accurately with a simple jpeg image. The down sides are that the computational energy to produce the 3 dimensional model can create lag, and that the 2D representation influences the interaction even when projected into 3D. The representation remains planimetric in fundamentals, even if displayed in 3D.

This experiment took a simple 8-bit jpg (possible values between 0-255) and interpreted each pixel as a point with an elevation (Figure P). The grid of points is then used as the basis for a quadrangular mesh which approximates a terrain surface. In this implementation, the transposition is parametrically controlled with a multiplier (Figure Q), as the bit depth of the jpg is not sufficient to record real elevation data. This reveals one liability of this approach, which is that raster images with greater bit depth are less universally readable, and may require

specialized libraries to be integrated into an open source approach.

The experiment served also to demonstrate some of the limitations in the raster data type for effectively encoding terrain. Because the raster is gridded, it does not distinguish between more or less variable areas of the terrain, and abrupt elevation changes lose detail when the change is not coincident with the grid. Hence the regularity of the data makes for areas of unnecessary resolution and areas of insufficient resolution. Hence while the raster data format is widely available and easily integrated, it is not ideal for the kind of holistic, surgical, intuitive interaction with terrain that Girot's theory requires. I argue that a 2 dimensional data type cannot ultimately be sufficient because it privileges an orthographic view of landscape rather than allowing for shifting perspective.

As Gill pointed out, interaction with a 2 dimensional representation is easy to mediate. Users are familiar with the planimetric metaphor. In the case of the raster interpreter experiment, it is easy to reshape the terrain by simply painting tone: brightening luma to lift the surface, and darkening to depress. This interaction lends itself well to smooth edits, but is limited by forcing manual editing. Parametric relationships are not easily supported by a gridded data type which treats every location as equivalent in value to every other. To challenge this limitation, I experimented with a modification of the interaction allowing for edits in three dimensions. As a challenge to the encoding of elevation as a grid of points, I explored shaping the procedurally defined surface through vector math. In this strategy, each three dimensional point is subject to influencing vectors determined by user defined attractor points (Figure S).

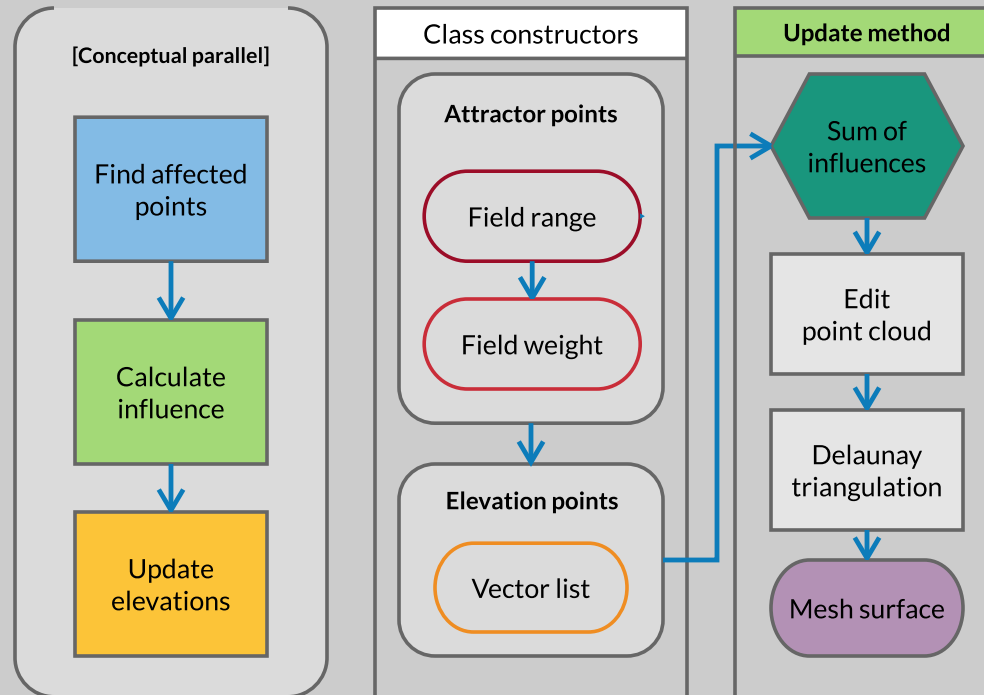
While the interaction is mathematically simple, there are significant inefficiencies in the medium. While vectors can be easily deconstructed, added, and applied to points of elevation, the grid data type enforces invariant X and Y coordinates for each

elevation point (Figure T). This means that vector influences, which are 3 dimensional data, unnecessarily record Cartesian coordinates which are only useful when converted. Edits are calculated based on distance and elevation delta, while every grid point must be recalculated on each iteration (Figure U). These are inefficiencies which are likely to reduce the scalability and utility of the interaction metaphor in the context of design.

Though raster data is easily distributed and facilitates using analytical tools common to GIS workflows, terrain data is increasingly collected in a point cloud format. As Girot makes clear, this data type makes possible a very different approach to terrain. Given that the format has the potential to be highly disruptive to prevailing 2D representations, a native environment for designers to interact with point clouds seems essential to take advantage of the innovation.

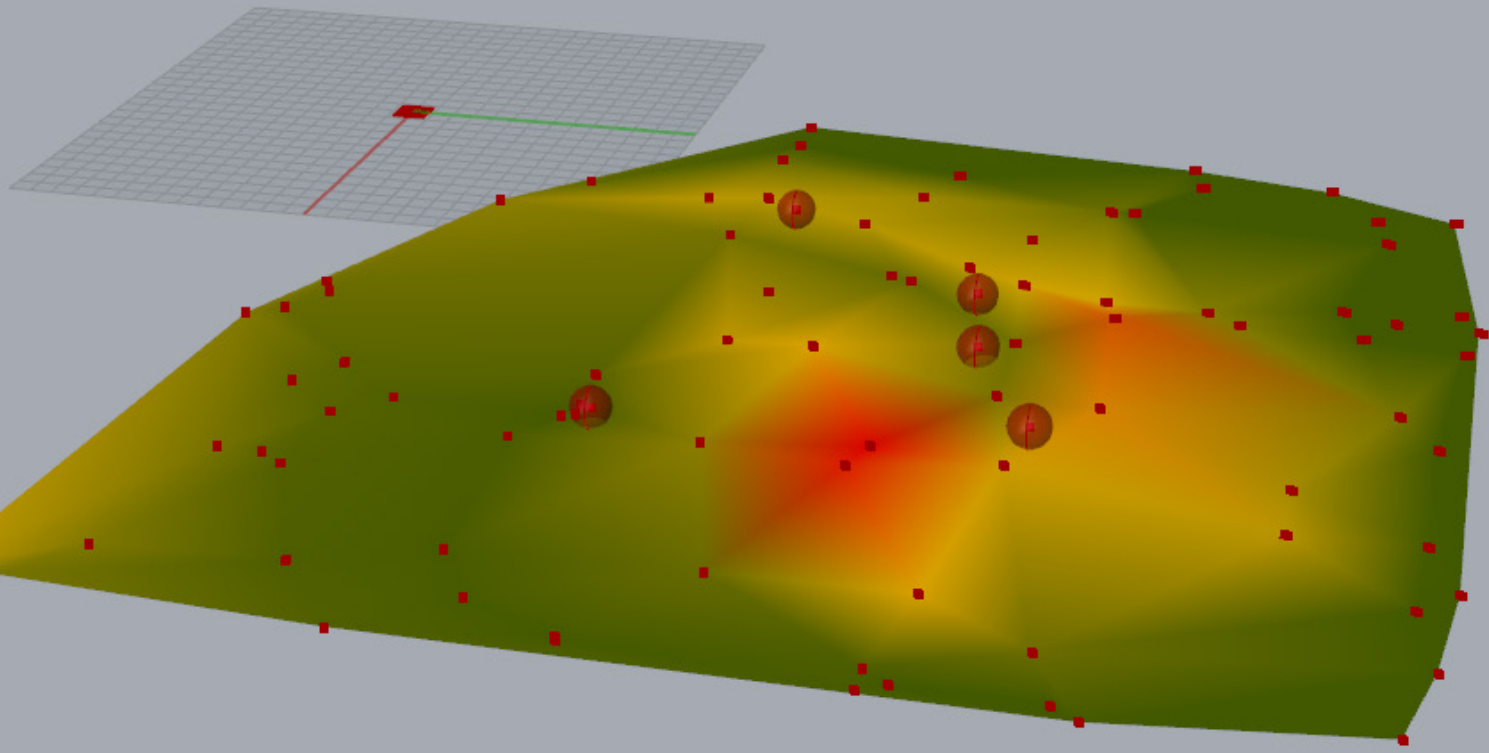
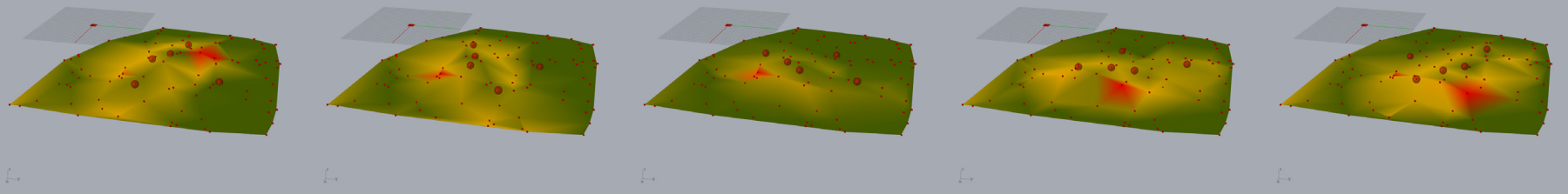
I explored an interaction which works on a representation of terrain as a heterogeneous cluster of spot elevations (Figure V). One strength of this approach is that surveyors typically record terrain by assembling individual spot elevations with spatial coordinates in a geographic reference. Hence a direct interaction with spot elevations would reduce the friction of translating specific and targeted elevation data into hierarchically flat interpolated data (Figure W). The point cluster attractor point experiment allows for multiple user placed points to influence spot elevations at arbitrary relative positions (Figure X). An inverse distance weight is applied to allow attractors to have graduated influence. While this interaction allows cumulative influence to be defined with multiple attractors, the results are not intuitively predictable. That is to say, intermediary points or points under multiple influence will be defined consequent to manual edits. This implementation did not allow the possibility to define dependencies or hierarchy of defining geometry in which a variable might be computationally optimized to approximate a resulting attribute. The script did allow for immediate analytical feedback showing the slope of each facet (Figure Y).

SMARTER GEOMETRY WITH PYTHON

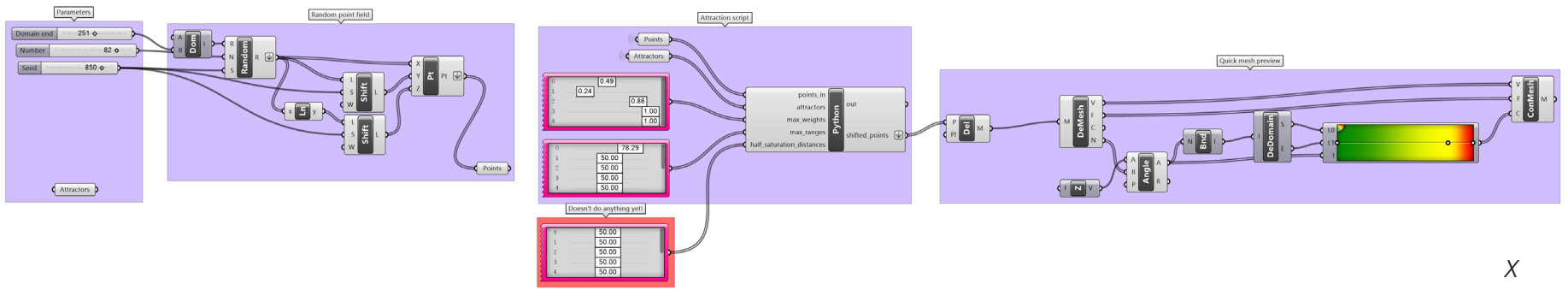


V

Figure U: a conceptual flow chart of the script storing points metadata in class instances; Figure V: screenshots from the operation of the script moving points with attractors; Figure W: the Grasshopper script; Figure X: python script housed in the GHPython component visible in Figure W.



W



X

```

1  import Rhino
2
3  living_points = []
4  shifted_points = []
5  active_attractors = []
6
7  class Attractor:
8      def __init__(self, p, max_weight, max_range, half_saturation):
9          self.point = p
10         self.weight = max_weight
11         self.range = max_range
12         self.falloff = half_saturation
13
14         def field(self, p):
15             distance = self.point.DistanceTo(p.point)
16             if distance < self.range:
17                 difference = p.point - self.point
18                 factor = (1-(distance/self.range))*self.weight
19                 z_delta = factor*difference[2]
20                 p.influence(Rhino.Geometry.Vector3d(0,0,-z_delta))
21
22     class LivePt:
23         def __init__(self, p):
24             self.point = p
25             self.vector_state = Rhino.Geometry.Vector3d.Zero
26             self.influences = [Rhino.Geometry.Vector3d.Zero]
27
28         def influence(self, vector):
29             self.influences.append(vector)
30
31         def move(self):
32             for v in self.influences:
33                 self.point += v
34
35             return self.point
36

```

```

37 #Add instances of attractor objects which influence the spot elevations
38 for i, p in enumerate(attractors):
39     global active_attractors
40     new_attractor = Attractor(p, max_weights[i], max_ranges[i],
41     half_saturation_distances[i])
42     active_attractors.append(new_attractor)
43
44 #Add instances of live editable 'point' objects to be pushed around
45 #by incoming vectors-----
46 for p in points_in:
47     global living_points
48     new_live_point = LivePt(p)
49     living_points.append(new_live_point)
50
51 #Calculate vector influences by distance, weight, range, falloff----
52 for a in active_attractors:
53     for p in living_points:
54         a.field(p)
55
56 #Calculate the new points after all the vectors have been applied---
57 for p in living_points:
58     global shifted_points
59     new_shifted_point = Rhino.Geometry.Point3d(p.move())
60     shifted_points.append(new_shifted_point)
61
62 for i, p in enumerate(shifted_points):
63     print(p - points_in[i])
64

```

Y

The analytical procedure also rested on a triangular irregular network built on the raw spot elevations. While this is a typical expedient in engineering, I argue that the representation limits landform ideation by enforcing planarity of undefined regions. Analysis which represents landform with this bias will influence the intuitive understanding of a designer who relies on it.

6.3. Field as an alternative metaphor

To explore challenging the limits of terrain conceptualization in terms of grid or TIN, I experimented with representing terrain as a field (Figure Z). While the visualization is mediated by translation of a field to a more traditional view, the fundamental data type might provide different utility, affordances, or efficiencies. This set of algorithms displays an edited terrain with a rectangularly parameterized NURBS surface (Figure AA). The underlying concept, however, suggests conceiving of the terrain surface as the result of a continuous and 3D data type. This concept is engaging for its resonance with the formulation of design as embodying systems outcomes—with form as result of a genealogy of place.

While the metaphor is attractive for a superficial isomorphism between the data type and a theory of landscape, interacting with terrain as continuous field is difficult (Figure BB). Using a species of the conceptual logic of the Fourier Transform might suggest a way of encoding a terrain as a complex wave interference, but those decomposed fields would bear no concrete relation to the confluence of actual landscape systems which shape a terrain (Figure CC). As such the metaphor is an artificial one, and unlikely to provide any intuitive paradigm shift in the fundamental relationship of designer with landform.

I conclude that the strength of Greg Lynn's and Mario Carpo's engagement with the metaphor of landscape as the crystallization of unseen interactions does not demonstrate any fundamental aptness for abstract representation in that paradigm. More

specifically, the fields / systems conceptual approach to landscape does not correlate as tidily with computational representations as seems superficially likely. Because the algorithmic process embodied by force interactions with vector fields does not necessarily correlate with any true account of the systems at work, the metaphor is weakened.

Typical diagramming practice in 2 dimensions also suggests promise within the metaphor of force interactions. For example, diagramming strategies often include arrows of particular priority, magnitude, and direction. These vectors can effectively represent system elements such as circulation, wind direction, water movement, etc. The forces are not precisely quantifiable, however, and any interaction between unlike forces will be approximated or worked out by design expertise, not by any quantitative mechanism. Hence the idea of landscape as the visible manifestation of continuous fields of varying type works better as a way of thinking than as a way of encoding or editing.

Representation which feels isomorphic to its object is compelling and suggests promise. However, the computer has its own logic which influences what is practical. An example of this fact comes from climate modeling, or computational fluid dynamics. The fluid is continuous, not discrete. It has flows and interactions at all scales. The fluid behaves as complex agglomeration of interrelated directional or energetic forces. Yet when modeling weather and the climate, the approach is discretization. The system is divided into a 3D grid, and each cell is evaluated for its influence on its neighbors. The bias of computation toward breaking down of complex things into many simplified things is considerable in how it draws innovation in a particular direction. Similarly, raster spatial analysis is the best developed because it is computationally accessible. Any proposal to replace discretization with on demand vector calculus integration of continuous force fields would necessitate a more intuitive relationship between the modeled systems and the real ones.

This set of experiments serves to elucidate the drawbacks of too much abstraction of terrain without specific cause. The result is a medium which is neither intuitive nor precise. Specifying precise results at particular spatial coordinates is disallowed by the complexity of the interactions and the distance between the underlying representation and the resulting terrain. By parallel, I have argued that a drawback of using a NURBS surface to interact with terrain consists in the defining elements (control points) not being coincident with the final result. This same issue holds when the terrain surface is solved for within nebulous 3D fields which may not have predictable relationship to the specific result.

6.4. Proposal for further exploration

The task of surgical intervention into the poetics of place requires an intuitive and holistic representation of terrain, but it also requires an intuitive interaction which is isomorphic in logic to the limitations, priorities, and considerations of design process. In light of this proposed isomorphism, a practical examination of typically available data and of ordinary landscape design iteration can suggest a way forward for bringing together the latest technology with a design approach that satisfies Girot's theoretical framework.

Each representational metaphor has its biases and limitations, and each provides certain advantages. Traditional contour editing allows for emphasis of continuity and naturalistic form, but privileges planimetric thinking and requires expertise for interpretation of an encoded stand-in. TINs allow precision and direct editing of individual points, but tend toward planarity and linear form. Fields are too abstract and difficult to predict.

Returning to the fundamentals of the data type which enables Girot's topology may suggest possibilities for a designerly interaction with a precise representation. Point clouds are potentially vast amounts of data, and that data is not all equally important. Point classification is common, and allows

VECTOR FIELD AS MEDIUM

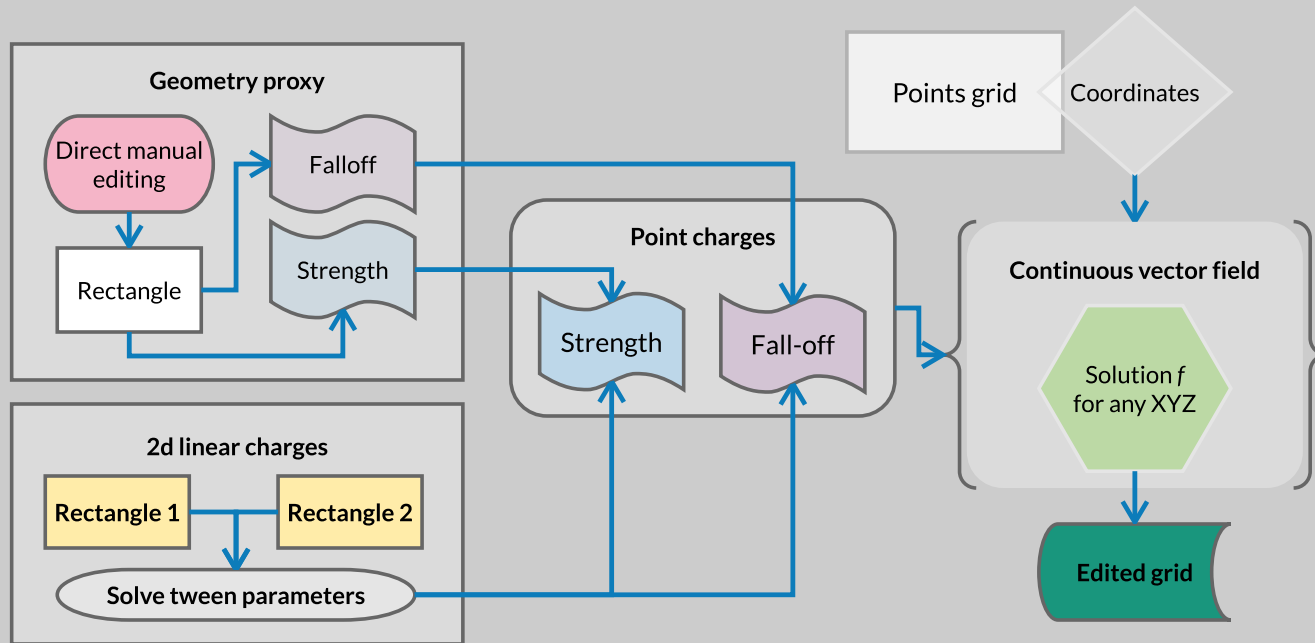
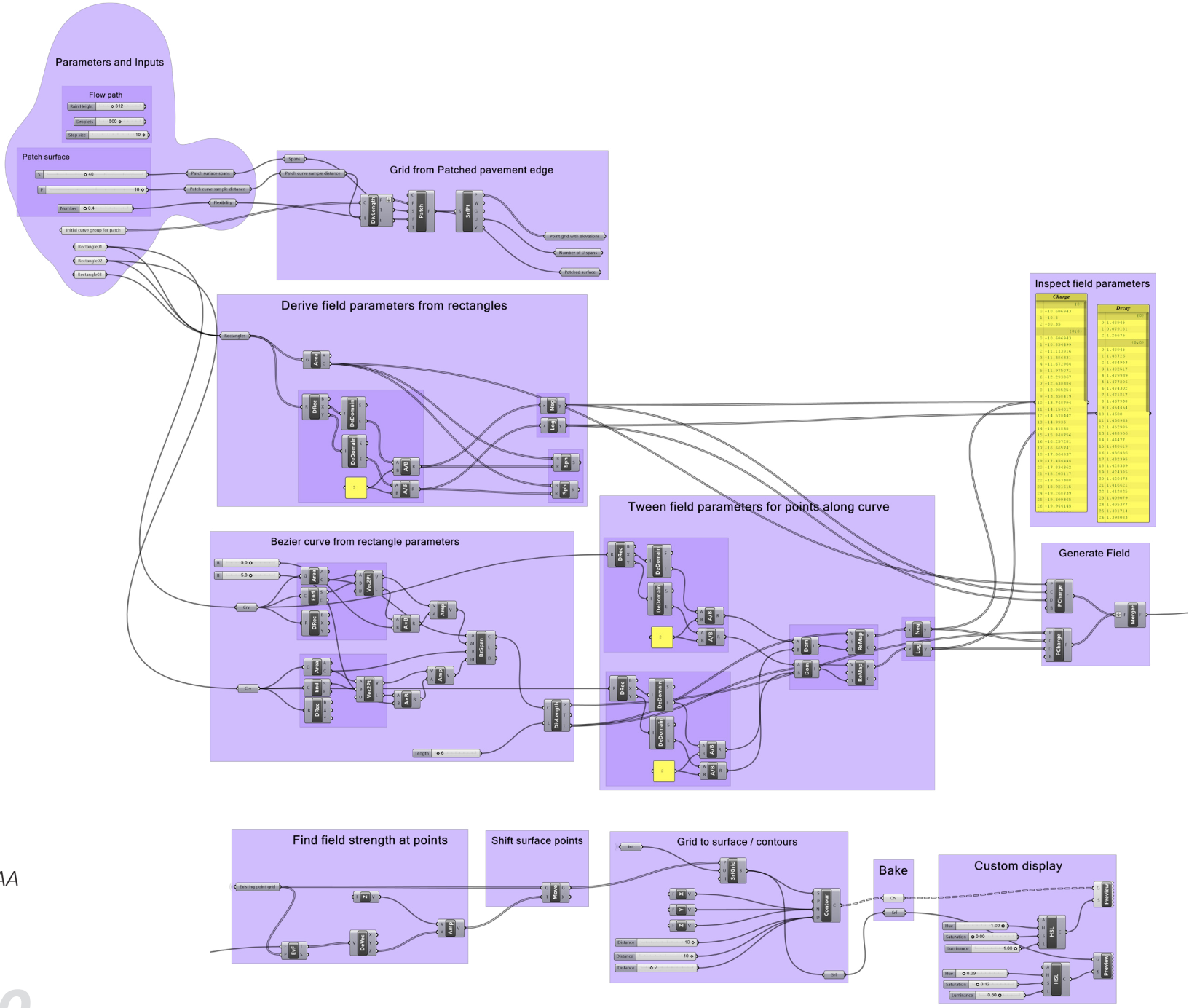


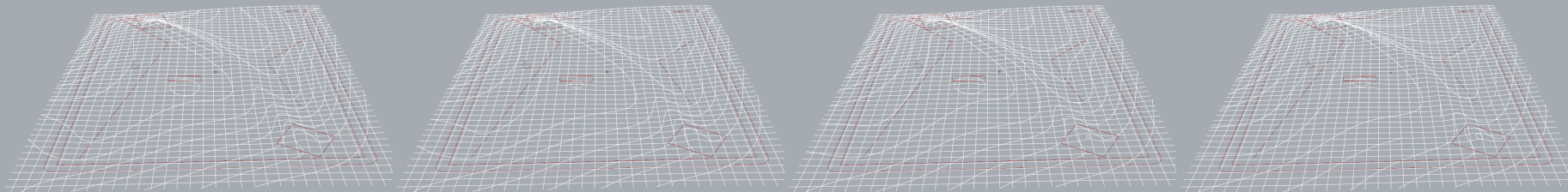
Figure Y: conceptual flow chart of using a continuous vector field as a proxy for terrain; Figure Z: The grasshopper script encoding terrain as a field and intervention parameters in directly editable geometry; Figure AA: demonstration of the continuity of a vector field; Figure BB: the script in action.



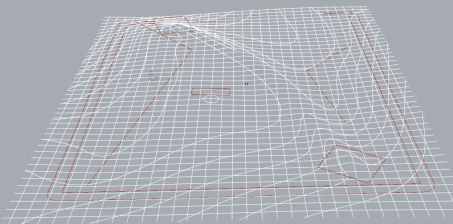
Inspect field parameters

Charge	Dvec
1 -13.604443	0.140193
2 -92.5	1.90279381
3 -20.26	2.176496
4 -13.604443	0.140193
5 -13.33316	1.140796
6 -13.306481	1.140493
7 -11.472044	1.1470939
8 -18.970702	1.1427506
9 -17.793667	1.1474302
10 -12.403388	1.1474302
11 -12.905258	1.1481271
12 -13.350419	1.1481271
13 -13.767374	1.1467733
14 -14.358117	1.1468444
15 -14.57447	1.14606
16 -18.9957	1.1456603
17 -19.63339	1.1458299
18 -15.640704	1.1440956
19 -14.33721	1.1444477
20 -16.60531	1.1410019
21 -17.066597	1.1404868
22 -12.608864	1.1414335
23 -17.336262	1.1410369
24 -19.202817	1.1414301
25 -19.583309	1.1420773
26 -19.521812	1.1414301
27 -19.760739	1.1420773
28 -19.609388	1.1400099
29 -19.944435	1.1410377
30 -19.944435	1.1400099

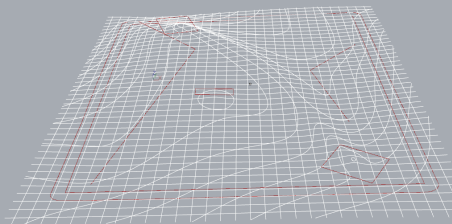
AA



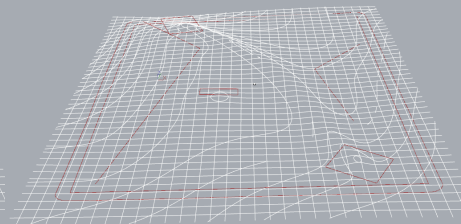
\hat{p}_x



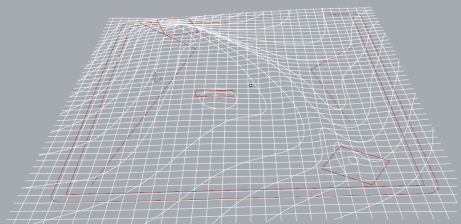
\hat{p}_x



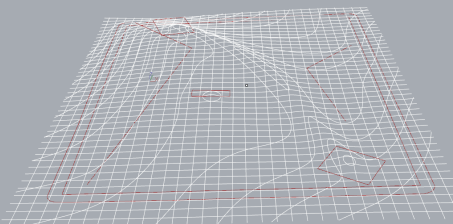
\hat{p}_x



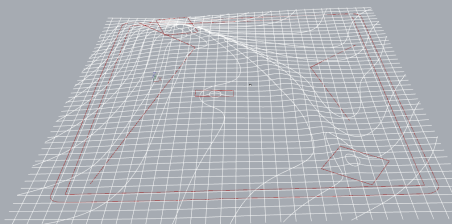
\hat{p}_x



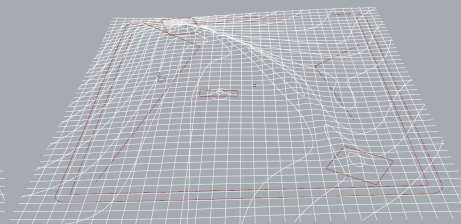
\hat{p}_x



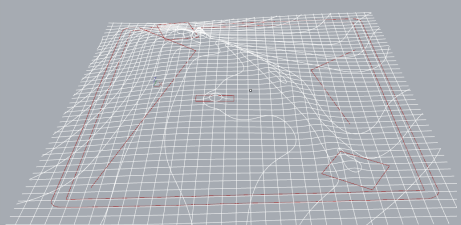
\hat{p}_x



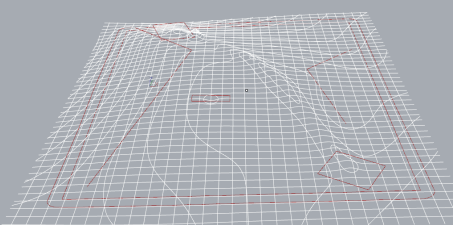
\hat{p}_x



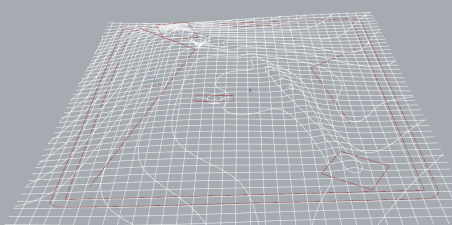
\hat{p}_x



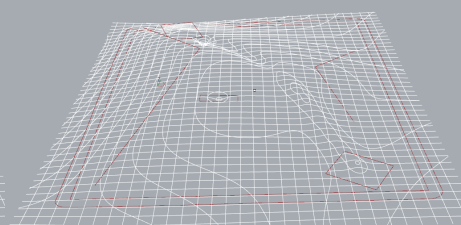
\hat{p}_x



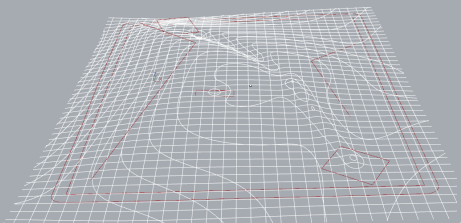
\hat{p}_x



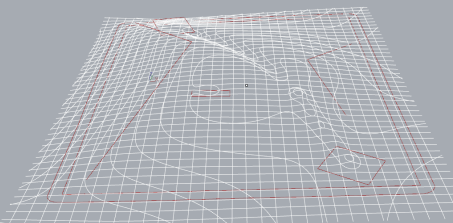
\hat{p}_x



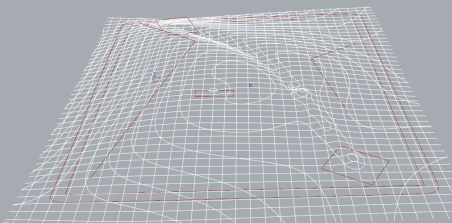
\hat{p}_x



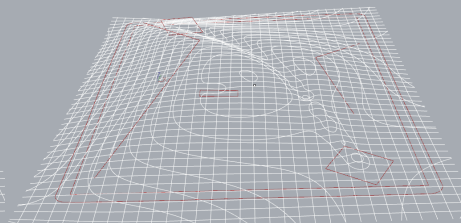
\hat{p}_x



\hat{p}_x



\hat{p}_x



\hat{p}_x

CC

a cloud to be organized by category of the objects represented. Algorithmic classification also allows trees and buildings to be automatically recognized. However, the points describing an object are still of flat hierarchy, meaning that, for example, roof lines are not recorded as such, but as simply the boundary between differently classified points. By contrast, because landscape design is concerned, in part, with optimization, a useful tool should allow for hierarchy and dependency.

The homogeneity of a point cloud makes the “surgical” quality of intervention more difficult when neighboring points have no relationship to one another. Hence a way of encoding dependencies is essential. The mathematical idea of topology, as those things which are invariant during transformation, is more at home in a TIN, where the surface has an inherent interrelatedness. Point clouds, instead, are full of noise and outlying points which are no distraction to a holistic view even if they are useless to the representation. This means that a big part of the challenge is to interpret the interrelatedness of represented features when no such relationships are actually encoded. If, as will commonly be the case, the relationships are the object of design, then discrete calculation will be required to bring each independent point into the desired form.

It is illustrative to recall that elevation is not the only attribute typically recorded at intermittent points. Another useful example is in geotechnical survey, where vertically continuous cores are commonly taken at multiple horizontal coordinates. This data is less likely to be modeled, but is necessarily subject to interpolation in order to apply the information to the decision-making process. The collection strategy is similar to survey, in which samples are recorded with georeferences. The attributes between sampled points are subject to interpolation. While laser scanning provides higher resolution information, the data types are essentially similar. Hence efforts to integrate the data type into a landscape platform could

be applicable to any information layer which is gathered by sampling points.

To effectively implement an interaction platform allowing point cloud editing would also require effective subsampling routines. Hierarchy would be essential, but moving across scales of precision and detail would be likewise prerequisite to computationally efficient utilization of a point cloud model.

A useful line of experiment would entail exploring how topological logic can be used within the point cloud data type. Good subsampling algorithms will be needed, as will a logic for ranking and integrating multiple specifications. While a 3D interaction with terrain will be highly useful in a more intuitive relationship with the poetics of place, the manual workflow implied does not account for parametric or optimizing strategies. Topological logic, however, seems helpful in allowing for rules based terrain forming. I conclude that toolmaking efforts investigating how point clouds might allow for parametric propagation of edits consequent to surgical interventions would continue the conversation in an appropriate and fruitful direction.

Section 7: REACTIONS AND SUGGESTIONS FROM PROFESSIONALS

Because this inquiry is oriented toward pragmatic understanding of what can be implemented in a for profit environment, it was valuable to converse directly with professionals who have in one way or another been engaged with technology in design. The Seattle professional community, as far as this research has found, has relatively few innovative firms in this realm. Certain firms which are pressing the possibilities of new workflows are motivated by the pressure put on them by a dominant architecture practice or by collaboration. While building information modeling has transformed the way business is done both by architects and the allied disciplines, the major software platforms have not focused on developing affordances for landscape design. This results in a design climate where landscape architects are forced to use a collage of tools while none is well suited to their needs. Firms which engage with BIM are therefore forced to find a workable way forward, grappling with the difficulties of using a mix of tools in production. A fair generalization might be that professionals in this situation are both excited for the possibilities and trepidatious about the drawbacks.

7.1. Thoughts from a round table discussion

Professionals from several Seattle firms spent time responding to my work and providing their own opinions and experiences. I enjoyed the generous attention of Brian Bishop, David Marshall, Andrew McConnico, Shawn Stankewich, and others. I spoke with practitioners in their offices, as well as at a round-table with individuals working for several different firms. Their opinions were shared freely, with the knowledge that I would integrate and paraphrase their thoughts, though those opinions only represent informal conversation. These discussions revealed

some of the common themes and concerns being considered in the the design world in Seattle.

One significant issue mentioned by Bishop is the influence that Revit has had on design outcomes in architecture. He points out that designs are easily recognized as coming from the platform. Moreover, Bishop states that architects who have been using Revit for a decade or more have become dependent on the platform. The point suggests that the software itself impacts the practice of individual architects, shaping what they imagine and how they implement a vision. According to Bishop, this also plays into the common reliance on parallel models in Sketchup or elsewhere. This makes clear the reality that a software package cannot enfold all possible design pathways, and creative designers will choose the tools which are appropriate to the task at hand. A fair inference is that designing a unified platform is in some ways a mistaken objective. Just as there is no universal pen or pencil, there can be no universal software package. A better approach might be to prioritize flexibility and pertinence of representational modes while guarding a cultivated agnosticism about how designers choose to use them.

Designers differ in their process and preferences. Within the community, trends are defined culturally and, in turn, reinforce how design is represented by establishing a common language. Hence, designing tools for the design professions is potentially endlessly complex. In addition, there is likely some discrepancy between the intention of a tool and its actual utility. Brian Johnson, advisor on this thesis, makes the excellent point that architectural designers may ignore well developed tools which do not reflect their mindset as they enter design process. Specifically, he points out that energy modeling tools which have been developed to fairly high maturity are often unused in the schematic design stage. He suggests that designers prefer to think in heuristics. If designers tend to rely on simple integrative ways of thinking, as Johnson suggests, more quantitatively focused tools may not be what the market will demand. In

this case, a unified analytical information modeling platform may be rejected by decision-makers. In the immediate future, it makes good sense for software developers to consult design professionals as they iterate—this is something which Autodesk and others certainly do, but landscape architecture has not been adequately represented.

Another major challenge to the premise of computationally guided landscape creativity comes from recognition that breaking rules is often very useful in process. Bishop points out that good ideas often come from accidents or intersections of conflicting moves. It is essential to remember that beautiful design is more than a complex optimization. Rather, inspiration, intuition, and accident all have roles to play. Bishop does, however, affirm that immediate availability of relevant information is of use. This point of view suggests that rules should be hard coded into software only with great caution and keeping open the possibility of customization. Parametric computation can be facilitated by more visible information, but should be scripted with care as a part of the design process. To the degree that design teams are able to define dependencies on a project specific basis, the network of dependencies can stay flexible. However, it is still easily conceivable that a rules based design approach would limit the ability to think outside the box: in this case the box itself is defined by the designer, but may still be limiting.

A vital pragmatic concern repeatedly brought up by the professionals I talked with is the way in which the technology facilitates contract relationships. While good collaboration should be valued for good design, shared media also serve to determine the character of professional relationships. One very simple example of this interaction is in how contract documents are produced and relied upon for legally binding specification of project details. A shift to BIM has significance for liability which is still being worked out in the professional climate. While these changes are explored through litigation or by insurance professionals, a similar shift may go unnoticed in how the

technology changes the influence exerted by each project role. For example, Ryan Storkman relates that general contractors are increasingly likely to build an information model themselves which becomes the ultimate source for what is constructed. Storkman feels that general contractors that own the model exercise more control in the team over the process. Stankewich builds on this point by describing the growing recognition among developers that a BIM oriented process can be streamlined. That is to say, commercial property developers may exert pressure toward particular outcomes which are facilitated by BIM. Designers who pursue out of the box thinking may find their creative thinking limited by the efficiencies afforded by a more cookie cutter approach to development, with creativity shaped to typical BIM forms. While this stricture may not apply for big ticket or high profile projects, it is much more likely to influence the ordinary architectural style trends over the scale of years or decades. Buildings which are designed and built now in quickly developing cities may not afford good opportunity for challenging the biases of Revit. Thus it is possible that developers will have a greater proportion of influence in the form of the city as a result of wide adoption of BIM. These arguments demonstrate that there is a political aspect to how technology impacts design process. Because landscape architecture is an essentially political discipline, it is essential to consider these consequences in adoption of new methods. New tools have the potential to improve working relationships, but also to reduce the influence of the profession. It is essential for landscape professionals to be proactive with advocating for the relative influence of our profession.

7.2. Chat over lunch

I chatted with three professionals over lunch about their experiences working at an integrated design firm which is architecture driven. In a professional context, this forces the landscape team to adapt and explore new workflows that enable them to operate

smoothly with other disciplines. The conversation with Jason King, Chuck McDowell, and Christian Runge served to confirm and challenge many of the ideas represented in this work.

They generously offered their time for an informal conversation about the dynamics of balancing resources for innovation vs. just getting the job done. While each of them expressed enthusiasm for the possibilities of technological evolution in their workflow, Jason emphasized that the challenge is often having the time available for experimentation. Further, there is the challenge of getting a workflow developed well enough to demonstrate to the firm's digital design lead that it makes sense to invest in—a Catch-22. Because the landscape team is small compared with the rest of the firm, they do not have the same bandwidth to innovate.

This team has been forced to develop workflows that allow them to collaborate with engineers and architects, although they struggle with different software packages not 'talking well' with one another. Jason spoke about the overhead of bringing Civil3D work into Revit, and felt that iteration is more difficult. He also joked about wishing that the grading were flat so that it could be more easily represented in the model. Chuck also points out that repetition of work happens when the terrain has to be imported in static chunks. The overhead in representation and translation means extra work.

Working in Revit has presented some significant hurdles. For example, Christian talks about the possible disconnect in what objectives are in mind when a model is started, and how that can continue to have significant impacts down the road. For instance, when the model is started to represent work for a competition, it is not built with the same attention to the details which will be important for documentation later. This means that sometimes a model has

to be remade from scratch. In fact, Chuck talks about reaching a moment where a decision has to be made about whether the model is for renderings or documentation. Although in theory one of the major efficiencies of information modeling is the elimination of repetitive documentation work, these practitioners talked about the present difficulties in moving from 3D creation to 2D documentation. Jason speculated that contractors might gain 3D literacy at some point, easing that difficulty. In the meantime, information modeling still faces the challenge of contract documentation.

While the complexity of projects with varying conditions has challenged this particular team with finding ways to use various tools in concert, their attitudes seem positive about the possible advantages of the latest software platforms. Both Chuck and Jason point out the immediacy of 3D modeling. Chuck suggests that modeling forces the designer to really grapple with whether the idea is buildable, while Jason recalls providing contractors with axonometric views of 3D modeled details. Drawing 2D details in plan and section does not always require the same kind of rigor. The landscape team has worked on very successful projects which were completely BIM based, and is continuing to evolve their practice. While their firm is still working out the possibility of funding R&D possibilities, the landscape team is staying proactive about the quick win opportunities in collaboration and creative work-flow.

7.3. Office hours with an educator

Keith VanDerSys is an instructor at Penn Design, as well as being a cofounder at his firm. He considers his work in innovation in workflow in terms of contribution to the field, and holds an open source ethic with regard to much of the tools and implementations he creates. His contribution is both a professional one,

working on a contract for major riverine hydrological problem-solving, as well as an academic one, structuring a pedagogical sequence to help students engage with the latest software. (For an introduction to some of the specific work Keith and his collaborators are doing, refer to Walliss & Rahmann, 2016).

Keith says he is motivated by empowering designers to be more confident in arguing for their vision. Rather than relying on a civil engineer for survey data and for hydrological analysis of proposed terrains, Keith is developing his own workflows using a variety of tools. He has used simple bespoke flow direction scripts to provide immediate feedback within the Grasshopper interface. He has also implemented agency standard numerical modelers such as ADH and SRH2D as part of his ongoing analysis of the Delaware River. The challenge at the moment is validating and calibrating the models he is using so that they can be considered trustworthy. Keith's goals include connecting the available platforms to allow for environmental modeling strategies to be incorporated into the design suite. In the context of rising sea levels, this is a particularly important practice for designers to take on.

Keith emphasizes that many things cannot be quantified, and it is actually imagination that moves capital. Rather, seamless transition of design process between analysis and design as complimentary domains remains a methodological objective. Keith is skeptical of overly deterministic schemas that suggest that the right logic structure will yield good design as its result. Automation is not, after all, an appealing design approach. Environmental performance is only one element to be considered along with other key design objectives such as aesthetics, program, etc. Still, Keith seems to consider innovation in landscape architecture into the engineering domain as pivotal to the design community responding well to contemporary sustainability concerns.

Conclusions

Environmental regulations and climate initiatives are increasing the pressure on new landscape projects to optimize their performance. The imperative to incorporate a more quantitative approach to evaluation of design ideation is achievable given the state of the art and widely available tools. The performative promises of resilience by landscape urbanist and ecological design theory are possible more rigorously to implement within current practice, and it seems likely that significant obstacles to this integration are fee structure and cultural inertia. This outcomes oriented pre-construction testing of design concepts is often completed through collaboration with third party consultants, especially when the project requires it (eg., a constructed wetland). In that regard, making analytical tools more accessible to designers would serve two limited purposes. First, it would reduce the overhead of communicating every proposal with an outside consultant and speed iteration. Second, it would better inform the designers' heuristics in the creative phases. These limited goals can be worked toward by firms who work with environmental concerns or who are interested in innovation.

Advocacy for the profession can likewise be advanced by better integration of quantitative methods usually left to engineers. Such an increase to the horizons of designers' faculties and area of comfort would serve to improve standing in conversation with collaborators. It would also extend the influence of landscape architecture into the nuts and bolts of implementation, thereby providing opportunities to ameliorate the engineering dominated urban form characterized by linear

subdivision and planar landform. The better designers understand the tools and process of engineering, the better equipped we are to meaningfully contribute to direct manipulation of the 'topology' of the city. These opportunities exist given the state of the technology available now.

Going forward, a need exists for new tools which are specifically oriented to the concerns and starting points of landscape architects. Landscape information modeling is likely an inevitable phenomenon, but its makeup need not be. While engineering or architecture metaphors are prevalent in the practice of landscape architects who are forced to work with Revit or Civil3D, innovators should not take this to be the only conceivable paradigm. Because the representations and workflow we choose will always have dramatic impact on the kinds of results they tend to produce, being proactive with shaping design process and the tools which enable it is crucial.

Christophe Girot proposes a valuable working conceptual framework for a new holistic relationship between designer and site. While his theory is doubtless influenced by a new technology for representing the landscape, that technology still requires development to become an effective mediator for useful manipulation of landform. As the need for collaboration through landscape information modeling grows and the pressure of climate instability mounts, it will become ever more urgent to take a holistic approach to landscape systems. I argue that it is inevitable that the present reliance on concept, narrative, and elegance of form as primary determining factors for landscape should become untenable. Diagram, photosimulation, and beautiful plan view graphic designs

cannot suffice for much longer; Girot's framework states this explicitly, and indeed requires a new design process.

Analytical tools in a digital environment are a big step toward a more performance oriented design process. It is important to recognize, however, that analytical feedback is only the tip of the iceberg of the possibilities presently being explored in mainstream computation in architecture. Beginning with geometric dependencies and relational modeling, research and development should delve into multi-variate optimization in landscape systems. Truly generative design may be farther off, or even prohibitively complex or impractical relative to architecture of objects. The more advanced modes of computation in design cannot be tackled before the simpler efficiencies are incorporated into the prevailing design culture in landscape architecture.

These marginal evolutions to typical design media may support better integration of landscape urbanist values, but will not be sufficient to achieve the reorientation proposed by Girot. The holistic understanding and surgical manipulation of terrain as foundational to the poetics of place will require further innovation in representational and interactive methods. While a singular and cohesive landscape design environment will come about only by significant development and ground testing, deliberate consideration of the requirements and ideal affordances of such a platform will make clear opportunities for critical and accessible changes to the abstractions and representational methods which designers rely upon as proxies for iterative and responsive design work. Such updates to design process should be regarded as urgent if the profession is to maintain and

build relevance and credibility in the AEC community. Design process should be continually pushed to better account for new technologies and more effectively optimize outcomes which mass urbanization requires.

Challenges to this evolution are many, as conversation with practitioners makes clear. An idealized environment is not likely to be engineered from scratch, nor if it were would it be likely to sweep aside entrenched design habits and heuristics. Any consideration of the ideal landscape information modeling environment should rather

be considered an inspiration to consider what interventions are most important and most practical for evolving typical design practice. Girot's theory privileges the genealogy of place and the cultural and social realities inherent in the shape of a landscape, a complexity for which he suggests a surgical response. This picture is a powerful metaphor for the 'landscape' of design practice: professional practice has a genealogy, a culture, and a symbolology. Effective intervention into this 'landscape' will require a holistic view of the working of its parts and a creative approach to targeted interventions.

As a profession concerned chiefly with exploration of the intersection and overlay of various dynamic systems, it should be natural for the design culture and processes themselves to be objects of design. There is a dissonance in allowing the method to remain static and entrenched when the method's object is dynamic and evolving quickly. The time to invest in innovation in landscape design process is now, and the alternative is to be marginalized in a world increasingly shaped by engineers, architects, and developers. The alternative is to shrub it up.

Selected Bibliography

Abdirad, Hamid, and Ken-Yu Lin. 2015. "Advancing in Object-Based Landscape Information Modeling: Challenges and Future Needs." In *Computing in Civil Engineering 2015*, 2015-Janua:548–55. Hoboken: American Society of Civil Engineers. doi:10.1061/9780784479247.068.

Alexander, Christopher. 1964. *Notes on the Synthesis of Form*. Vol. 57. Harvard University Press. doi:10.1086/601876.

Baker, W L. 1989. "A Review of Models of Landscape Change." *Landscape Ecology* 2 (2): 111–33. doi:DOI: 10.1007/BF00137155.

Ballantyne, A. 2007. *Deleuze & Guattari for Architects*. London: Routledge. <https://www.google.com/books?hl=en&lr=&id=9NjMCeiPvaYC&oi=fnd&pg=P-P1&ots=6RCoNr46od&sig=wDw0gJHI9>

Bansal, V. K. 2011. "Use of GIS and Topology in the Identification and Resolution of Space Conflicts." *Journal of Computing in Civil Engineering* 25 (2): 159–71. doi:10.1061/(ASCE)CP.1943-5487.0000075.

Belesky, P. 2013. "Processors and Processes: Exploring Computational Design and Landscape Architecture." Victoria University of Wellington. <http://researcharchive.vuw.ac.nz/handle/10063/2837>.

Bollinger, K, M Grohmann, and O Tessmann. 2010. "Structured Becoming: Evolutionary Processes in Design Engineering." *Architectural Design*. <http://onlinelibrary.wiley.com/doi/10.1002/ad.1103/full>.

Box, G. E. P. 1979. "Robustness in the Strategy of Scientific Model Building." In *Robustness in Statistics*, edited by R. L. Launer and G. N. Wilkinson, 201–36. Academic Press.

Burry, M. 2013. *Scripting Cultures: Architectural Design and Programming*. Hoboken: Wiley. https://www.google.com/books?hl=en&lr=&id=P5x2k_E_4CsC&oi=fnd&pg=PA1&dq=scripting+cultures

Calkins, Meg. 2005. "Strategy Use and Challenges of Ecological Design in Landscape Architecture." *Landscape and Urban Planning* 73 (1). Hoboken: 29–48. doi:10.1016/j.landurbplan.2004.06.003.

Cantrell, B, and N Yates. 2012. *Modeling the Environment: Techniques and Tools for the 3D Illustration of Dynamic Landscapes*. Hoboken: Wiley. https://www.google.com/books?hl=en&lr=&id=xb1_06BG

Cantrell, BE, and J Holzman. 2015. *Responsive Landscapes: Strategies for Responsive Technologies in Landscape Architecture*. London: Routledge. <https://books.google.com/books?hl=en&lr=&id=TZX4C>

Cantrell, Bradley, and Justine Holzman. 2014. "Synthetic Ecologies: Protocols, Simulation, and Manipulation for Indeterminate Landscapes." In *ACADIA 14: Design Agency* [Proceedings of the 34th Annual Confe

Carpo, M. 2004. "Ten Years of Folding." In *Folding in Architecture*. Academy Press. https://scholar.google.dk/scholar?hl=en&q=folding+in+architecture&btnG=&as_sdt=1,5&as_sdtp=#3.

Christenson, Mike. 2008. "Questioning the Primacy of Visual Simulation in an Epistemology of Digital Models." http://cumincad.scix.net/cgi-bin/works/Show?_id=ecaade2008_034&sort=DEFAULT&search=episte

Cupkova, Dana, Nicolas Azel, and Christine Mondor. 2015. "EPIFLOW: Adaptive Analytical Design Framework for Resilient Urban Water Systems." In *Modelling Behavior*, edited by MR Thomsen and Martin Tamke

Davis, D. 2013. "Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture." RMIT University. https://scholar.google.dk/scholar?q=davis+Flexible+Parametric+Models+in+the+Practice+of+Architecture&btnG=&hl=en&as_sdt=0,5#0.

Davis, D, J Burry, and M Burry. 2011. "Understanding Visual Scripts: Improving Collaboration through Modular Programming." *International Journal of Architectural Computing*. <http://multi-science.atypon.com/doi/abs/10.1260/1478-0771.9.4.361>.

de Monchaux, Nicholas, Shivang Patwa, Benjamin Golder, Sara Jensen, and David Lung. 2010. "Local Code: The Critical Use of Geographic Information Systems in Parametric Urban Design." In *ACADIA 10: LIFE In*

DeLanda, Manuel. 2002. "Deleuze and the Use of the Genetic Algorithm in Architecture." *Architectural Design* 72 (1): 9–12.

Deleuze, G, and F Guattari. 1988. *A Thousand Plateaus: Capitalism and Schizophrenia*. Bloomsbury Publishing. https://scholar.google.dk/scholar?hl=en&q=a+thousand+plateaus&btnG=&as_sdt=1,5&#

Dossick, Carrie Sturts, and Gina Neff. 2011. "Messy Talk and Clean Technology: Communication, Problem-Solving and Collaboration Using Building Information Modelling." *Engineering Project Organization Journal* 1 (2). Routledge : 83–93. doi:10.1080/21573727.2011.569929.

Ervin, Stephen M. 2001. "Digital Landscape Modeling and Visualization: A Research Agenda." *Landscape and Urban Planning* 54 (1-4): 49–62. doi:10.1016/S0169-2046(01)00125-6.

Frazer, J. 1995. *An Evolutionary Architecture*. London: Architectural Association. <http://cumincad.scix.net/cgi-bin/works/Show?2068>.

Shelden, DR, and AJ Witt. 2011. "Continuity and Rupture." *Architectural Design*. Hoboken. <http://onlinelibrary.wiley.com/doi/10.1002/ad.1266/full>.

Sklar, FH, and R Costanza. 1990. "The Development of Dynamic Spatial Models for Landscape Ecology: A Review and Prognosis." In *Quantitative Methods in Landscape Ecology*, edited by Monica Goigel Turner and RH Gardner. http://landscape.forest.wisc.edu/courses/Landscape565spr01/Sklar_Costanza1991.pdf.

Spirn, AW. 2013. "Ecological Urbanism: A Framework for the Design of Resilient Cities." In *Resilience in Ecology and Urban Design*, edited by STA Pickett. Springer Verlag. http://link.springer.com/chapter/10.5822/978-1-61091-491-8_50.

Spuybroek, L. 2005. "The Structure of Vagueness." *Textile* 3 (1). <http://www.tandfonline.com/doi/pdf/10.2752/147597505778052620>.

Steinitz, Carl. 1990. "A Framework for Theory Applicable to the Education of Landscape Architects (and Other Environmental Design Professionals)." *Landscape Journal* 9 (2): 136–43.

Terzidis, K. 2003. *Expressive Form: A Conceptual Approach to Computational Design*. London: Taylor & Francis. https://www.google.com/books?hl=en&lr=&id=qVsGvohzaJEC&oi=fnd&pg=PP11&dq=expressive+form+terzidis&ots=CZMgYMu2y&sig=fSVGaAj-5Ax_Q1KAya7G63sqWvY.

Treib, M. 2008. *Representing Landscape Architecture*. London: Taylor & Francis. <https://www.google.com/books?hl=en&lr=&id=0Jt5AgAAQBAJ&oi=fnd&pg=PP1&dq=marc+treib+representing&ots=HdbiSAYyQG&sig=BXbS-tFzGumU7uMWY-6qWDFOMPnQ>.

Treib, M. 2012. *Drawing/thinking: Confronting an Electronic Age*. London: Routledge. https://www.google.com/books?hl=en&lr=&id=CZnXsvq9BFsC&oi=fnd&pg=PP2&dq=marc+treib&ots=cSlayYK28E&sig=_avwxEEdrAKkD9bqGK_iSwhC1g8.

Waldheim, Charles. 2012. "Digital Landscape 3.0 (2013-2037)." In *Instigations: Engaging Architecture, Landscape, and the City*, edited by M. Mostafvi and P. Christensen. Germany: Lars Muller.

Walliss, J, and H Rahmann. 2016. *Landscape Architecture and Digital Technologies: Re-Conceptualising Design and Making*. London: Routledge. <https://books.google.com/books?hl=en&lr=&id=6YiPCwAAQBAJ&oi=fnd&pg=PP1&dq=landscape+architecture+and+digital+technologies&ots=BEfjcvmpDD&sig=zCnBgcyvmQwIhYx-Sh2wyTzhBLro>.

Walliss, Jillian, Zaneta Hong, Heike Rahmann, and Jorg Sieweke. 2014. "Pedagogical Foundations: Deploying Digital Techniques in Design /research Practice." *Journal of Landscape Architecture* 9 (3): 72–83. doi:10.1080/18626033.2015.968421.

Watson, David. 2010. "Are Landscape Programmes Meeting the Challenge of Educating the Second Generation of Digital Landscape Architects." In *Peer Reviewed Proceedings of Digital Landscape Architecture*. Wichmann. http://www.kolleg.loel.hs-anhalt.de/landschaftsinformatik/fileadmin/user_upload/_temp_/2009/2009_Proceedings/601_watson-2009-jun01-n.pdf.

Whitehead, H. 2003. *Laws of Form. Architecture in the Digital Age: Design and Manufacturing*. Taylor and Francis. https://scholar.google.dk/scholar?q=laws+of+form+whitehead&btnG=&hl=en&as_sdt=0%2C5#1.

Winn, Kelly, Jason Vollen, and Anna Dyson. 2012. "Re-Framing Architecture for Emerging Ecological and Computational Design Trends for the Built Ecology." In *ACADIA 12: Synthetic Digital Ecologies [Proceedings of the 32nd Annual Conference of the Association for Computer Aided Design in Architecture]*. http://cumincad.scix.net/cgi-bin/works/Show?_id=acadia12_251&sort=DEFAULT&search=%2bperformance%2becology&hits=13.

Winsberg, Eric B. 2010. "Sanctioning Models: Theories and Their Scope." In *Science in the Age of Computer Simulation*, 7–28. doi:10.1063/PT.3.1221.

Woodbury, Robert. 2010. *Elements of Parametric Design*. London: Taylor & Francis. <http://eprints.iat.sfu.ca/930/>.

Acknowledgements

To God for keeping me alive through all this, Thanks. To each of the professionals who took time out of their schedule to engage with me on this project, Thanks. To every known and unknown person who encouraged me in simple ways, Thanks. For my many generous academic colleagues and advisors, particularly my committee members Brian and Ken, who spent good Seattle fishin' days cooped up looking at another draft—Thanks.