The Art and Science of Computational Information Design

Aaron Siegel

Submitted to the CADRE Laboratory for New Media,
School of Art and Design,
in partial fulfillment of the requirement for the degree of
BFA in Digital Media Art at
San Jose State University
May 2006

http://cadre.sjsu.edu

Table of Contents

Preface

The past few years have been a confusing time for me. In a society where dichotomies are established to separate the arts and the sciences, I always tend to find myself somewhere in the middle, not quite leaning towards either direction. It is with this in mind that I decided to create a transvergent manifesto of sorts to establish my new disciplinary practice in my own mind and those of others.

On the advice of Joel Slayton, I decided to format this paper as though it was an interview of myself conducted by four influential thinkers: Suguru Ishizaki on the topic of multi-agent systems, Bas C. van Fraassen on the topic of scientific realism, Lev Manovich on the topic of media arts and society, and Wassily Kandinsky on the topic of visual aesthetics. This method allowed me to construct interesting questions from their estimated perspectives that I probably would not have even thought of before. The amalgamation of the individual short essay questions into a comprehensive body allowed for a heightened understanding of the subject, and a clearer formulation of a thesis.

It is the hope of this author that the creation of this document will help influence societies views on the separation of the arts and sciences, and allow new forms of experimentation and creativity to emerge from previously unexplored areas.

A copy of this paper is available online at www.datadreamer.com.

Introduction

Over the past decade there has been radical development in the area of information design. The use of print layouts has subsided to their digital counterparts, simply because of the dynamic and ever changing nature of modern information. It is out of necessity that the creation of means for the display of such information has become increasingly common place.

Most recently, the compilation of ideas into a single trans-disciplinary field, computational information design, has allowed for the increase in discourse and evolution of ideas regarding communication displays. Ben Fry of the Massachusetts Institute of Technology Media Lab originally coined the term and originated the fundamental principles behind computational information design in his doctoral thesis of the same name. The ideals of communication design behind the discipline are clearly established, but leave me wondering about the artistic and scientific contributions and products created from works of computational information design.

The fundamentals of computational information design can easily be applied to both artistic and scientific fields, since both influence the creation process as well as the intended result. It can even be said that the products of computational information design systems are both art and science, only viewed from separate perspectives. It is the intent of this paper to combine those perspectives.

Suguru Ishizaki

Suguru Ishizaki received his PhD from the Visible Language Workshop at the Massachusetts Institute of Technology Media Lab, where he did his doctoral thesis on Improvizational Design Systems. His research focuses on the development of tools for communication design. His work is most well known for the adaptation of multi-agent systems to the information design process, and treating visualization as the emergent behavior of active design agents.

He is currently a professor of communication design at Carnegie Mellon University and a Senior Staff Engineer at QUALCOMM Incorporated.

Why is decentralization of information elements important?

Information can most often be broken up into smaller packets consisting of individual samples along a common axis, all of which share common properties with differing values depending on the sample.

Each of these samples can be represented as an agent, allowing a self-regulating visual representation and computational methods independent of other data samples. By decentralizing the process of visual transposition and attribute assessment, it helps ensure against corruption. If all of the data was visualized and computed at once using a single method, and one sample of data was corrupted, it would in turn throw the rest of the results off.

It's easy to think about this in terms of power plants; the centralized nuclear fission reactor vs. the decentralized network of photovoltaic cells. Any problem within the centralized model could cause the entire power grid to go out, as opposed to a failure of a solar panel, which only decreases the collective output of energy generation by a slight amount, and not disturbing the operation of the rest of the network.[1]

Decentralization also assists in the processing of data entities. Rather than writing a program with the entire data collection in mind, a developer writes a program that handles each data sample independently. Since each data sample contains the same properties with different values, one algorithm can be written to perform the instruction set on each information agent separately.

Since information packets are often updated independently of each other, they can be individually re-calculated and re-visualized without the need for processing the entire information collection. This becomes incredibly useful when changing the properties or formal attributes of an agent, since a single entity can be told to move or adjust its color without affecting the rest of the agents on the stage.

A well developed multi-agent system applies the properties, calculative instructions, and visual transposition within each agent itself, making it possible for the agent to self-regulate and immediately adjust its behavior based upon a simple change in properties.

[1] Winner, Langdon. *The Whale and The Reactor*. 1986. p. 92.

How do multi-agent systems affect the scientific method?

Even without the use of multi-agent systems, computation is the scientific methods friend. Since computer programming is structured in a formal language, its explicit nature causes the linguistic expression of logical circuits. These circuits represent the flow of hypothesis, observation, and testing within the scientific method.

Most of the issues with the scientific method arise through inconsistency. Since programming languages allow a developer to write an instruction set in a manner that allows them to be run over and over without differentiation, it lends itself ideally to use in the scientific method.

Multi-agent systems, when applied to computation, are usually constructed utilizing object oriented programming. Possibly the strongest quality of object oriented programming is its capability for inheritance. Inheritance is applied through the use of classes, which are inherited by objects. By instantiating an object with the properties of an information element, it creates a copy of the class with the corresponding information properties. When creating a multi-agent system, an agent (or object) is instantiated for each information element in the database. This means that each information element is transposed in exactly the same manner, causing no internal inconsistency.

Outside of the internal attributes computing lends to the scientific method, it also changes the way the user interacts with and perceives the data, thus affecting the process again. The ability to visually refine the display of information assists in its analysis, which may affect the next step taken by the investigator. It is the responsibility of the developer to know how to transpose information in an appropriate visual manner for understanding.

How do multi-agent systems affect artistic control?

I like to make the analogy that each agent acts like a copy of the creator, and that it's merely following the same simple rules that the creator follows themselves if they were to create the same piece.

People become confused with the concept of automation, and expect that it's replacing the designer. This is an understandable vantage point, as automation has replaced many functions of humans in the past. The use of automation in computational information design is necessary because of not only the sheer amount of data that is usually dealt with, but also the rapidly changing content of that data.

It becomes apparent that a traditional artist attempting to represent a system can only do so in a static instance in time. A new media artist, utilizing computational means to maintain a constant incoming stream of information and create a visual transposition, is always contextually sensitive as it is viewing live data. This augments the artist's ability to reflect a situation by allowing a piece to constantly reveal the here and now.

When someone works in this medium of constantly updating information, they have no ability to determine what the exact output will be, since it is entirely dependent on the current state of the data. The developer creates rules for each agent, which essentially act as a paintbrush for the digital canvas. The data then acts as paint, loading the brush with the appropriate pigment and dilution to represent its state.

Multi-agent systems become process oriented artworks, similar to that of Sol LeWitts wall drawings. LeWitt would create explicit instructions written in English, such as:

“Lines not long, not strait, not touching, drawn at random using four colors, uniformly dispersed with maximum density, covering the entire surface of the wall.”[2]

The instructions are given to a draftsperson to interpret and apply to a wall. Since the instructions are not precise in how the image is to be drawn, the same instructions can be applied by different draftspersons over and over again with different results each time. Essentially, LeWitt acts as the algorithm (or paintbrush), and the draftsperson acts as the paint (or computer). LeWitts instruction based works are most appropriately associated with generative artworks, in which the instructions are the only input, as opposed to something transposed from constantly changing information.

[2]Reas, Casey. *Software Structures.* Whitney Museum of American Art. 2004.

http://artport.whitney.org/commissions/softwarestructures/text.html

How do design systems affect acquisition of information?

Since computational information design systems are constructed in a manner that allows the navigator of them to create the context of intention, the system can then re-arrange itself in a manner most suited for displaying the desired format. This often means not only re-arranging visual components, but also adding and subtracting information elements.

Many people see computational information design systems as a way to interface with data that has already been acquired, processed, and databased accordingly. I believe this is the most basic implementation of design systems, and that it is possible to create a design system which acts as an interface to the data acquisition process itself.

The process of data acquisition can be left up to the navigator of the piece. Perhaps a system of queries to launch a spidering system with live results could be implemented as a pre-parser to the visual arrangement and organization of the information retrieved.

Spidering follows the methodology of multi-agent systems very closely, in that it utilizes a program of simple instructions which clones its self to recursively investigate the branching network of associations for a given query.

This type of live spidering activity is usually frowned upon by network administrators and designers as it usually causes a great deal of network traffic, and can sometimes result in a lagged return of information. When well tested and constructed properly, a spider can be a useful and efficient tool for gathering data, but when constructed poorly and not tested thoroughly, it can become a network menace.

Computational information design is a transdisciplinary practice, colliding previously separate disciplines into something new, so developers within this practice must be aware of the entire process, and how each separate aspect impacts the others. With this in mind, a change to the design system could easily necessitate a change to the data acquisition process, or vice versa. Rather than having a separate person focus on each aspect with no real insight into the other aspects of the project, one person needs to be attentive to the entire process.

The back and forth exchange of changes between project aspects is necessary for the refinement of the final product. Being forced to complete one aspect before the other will result in a creative limitation of the aspects that come later.[3]

[3]Fry, Ben. *Computational Information Design*. 2004. p. 14.

How do multi-agent systems influence the use of improvisational design?

Improvisational design is a theory of design practice that bases the temporal presentation of design elements on the underlying information that is fed into the design system. It relies on simple rules to create a more emergent structure suitable for the current state of information.

Multi-agent systems are the most often used implementation for improvisational design applications, as it comes with the most flexibility. Improvisational design systems need to be able to add, modify, and remove temporal presentations of information from the stage, and adjust its self accordingly. It is for this reason that most improvisational design systems are written from within the object oriented programming paradigm. To do this through procedural programming practices would be inefficient and confusing.

By separating the properties and behavior of each information element, they can be responsible for making adjustments due to the context of the system around them. So opposed to a central processing program adjusting the behavior of a hundred different entities, each entity adjusts its behavior according to the behavior of those around them. This is a key aspect for simulation, which I will discuss more in depth further ahead.

Many people talk about object oriented programming as a paradigm, which I mentioned earlier. It is considered a paradigm for computation as it changes the way a computer programmer thinks about the composition and design of their applications.[4] By enabling a different structure with which developers can become acclimated to, it can augment their engineering and design capabilities. It is with this in mind that it becomes apparent how multi-agent systems have affected and will continue to affect the development of improvisational design systems.

[4] Wikipedia. *Object Oriented Programming.*

http://en.wikipedia.org/wiki/Object_Oriented_Programming

How does interaction extend comprehensibility?

Interaction with data is another key element unique to computational information design. With static graphic representations of data, the viewer is left with the singular perspective that has been provided by the developer. When interaction is incorporated, the viewer then becomes an active participant in the exploration and investigation of the information.[5] They can choose to change the perspective on the information, perhaps re-arranging elements and sorting them by a different dimension of data (alphabetically, chronologically, etc). Not only can the viewer manipulate the sorting, but also the scaling of the information, to reveal micro or macro trends.

It is easy to see how interaction can help in comprehension of a subject matter, but the developer must be careful to implement it wisely. Poor interaction design can lead to gadgetry without substance, and even the convolution of the data at hand.

Good interaction design usually results in extending the users capabilities to the system they are interfacing with. The mind creates a perceived correlation between action and feedback response to essentially augment the user's ability to understand the subject matter.[6]

Most importantly, interaction lends the user the ability to control the context based upon their intentions, which is often times the most difficult aspect to predict. A developer can create a design system that anticipates the intentions of the viewer, but by adding interaction, the developer allows the navigator to create their appropriate perspective, and perhaps find correlations from it to others.[7]

[5] Fry, Ben. Organic Information Design. 2000. p. 26.

[6] Clark, Andy. Natural Born Cyborgs. 2003. p. 62.

[7] Ishizaki, Suguru. Improvisational Design. 2003. p. 11.

What do multi-agent systems provide in terms of emergent behavior?

Multi-agent systems were originally created to address issues of simulation and emergent behavior. The initial introduction of objects as programmable entities was implemented in a language written expressly for simulation.[8]

Multi-agent systems provide the ability to create something once and instantiate it a seemingly endless number of times with a seemingly endless number of variations. The capability to provide a simple rule set to each agent and watch their behaviors "bounce" off one another allows for emergent activity.

The value behind emergence is the simulation of a system of behavior, where the entire stage is not defined, but merely the small elemental interactions. Studying the complexity behind systems often requires the use of these types of multi-agent simulation tools. The more components exist in a system, the more interactions occur, thus increasing the complexity of the situation. Of course, the introduction of more or less elements will change the emergent behavior of the overall system.[9]

Emergence is a unique attribute to computational information design, as it relies on the data being fed into the design system to create temporal transpositions. The emergence comes from the interactions between information agents depending on their respective properties. The resulting behavior of the system would give a clue to the relationships of the information element properties. The manner in which that relationship is visually translated is obviously the most essential aspect in understanding that information.

[8]Wikipedia. *Simula 67*. http://en.wikipedia.org/wiki/Simula

[9]Wikipedia, *Emergence*. http://en.wikipedia.org/wiki/Emergence

How can temporal abstractions be transposed from information attributes?

Before actually attempting to transpose information attributes into a formal presentation, one must determine the goal for analysis. This will help guide the construction of temporal abstractions in a manner that allows for the most intuitive understanding of the subject matter.

One must determine the key attributes for each information element that are to be compared. By establishing the types of information attributes and their according ranges, one can decide upon an appropriate visual form.

An appropriate visual form is one that provides the viewer a chance to compare formal qualities of separate information attributes, and perhaps even discover correlations between different attributes.

Multi-agent systems provide an easy means for creating temporal abstractions, since the instructions for creating the formal presentation of the information should be the same for every information element passing through it. By applying the transposition instructions to the multi-agent model, it can be written once and inherited by all information agents. In addition, by allowing each agent to handle its own visual form, it makes them capable of transforming appearance when their associated data properties change in value.

Although the act of transforming data values into visual representations is the computers task, it still takes a talented designer to determine how the data should be transposed, and with what methods to do so.

Bas C. van Fraassen

Bas C. van Fraassen is a member of the Princeton Philosophy Department, and previously taught at the University of Oklahoma, University of Southern California, University of Toronto, and Yale University. He is the editor of the Journal of Philisophical Logic and co-editor of the Journal of Symbolic Logic.

Van Fraassen is most noted for the coining of the term "constructive empiricism", which is an anti-realist philosophy of science standpoint that states scientifc theories are semantically literal and should aim for emprical adequacy.

Does computational information design strive for empirical adequacy?

First, we must deal with the concept of empirical adequacy as applied to computing itself. I have struggled with this idea over time, as empirical adequacy strives to describe observable parts of the world.[10] At the same time, our information can come from a variety of pre-filtered sources such as web sites, databases, network traffic, or user submissions. The fact of the matter is that computational information design can strive for empirical adequacy, but it is not limited to doing so.

For a product of computational information design to be considered empirically adequate, its source information must be acquired under strict regulation. The data should be restricted to observable phenomena, and for even more control over consistency, acquired by a single collector. As far as the visualization goes, it should not attempt to describe the structure of the world beyond the observable components collected.[11]

Although it is good for certain situations to strive for empirical adequacy, I feel that it is limiting to the potential of computational information design to draw inferences from patterns and trends spotted within the observable material. There are many statistical methods that can be utilized against collected data to gain meta-level associations among the elements, but this type of representation goes beyond what may be observable to human senses.

With other situations, computational information design can be applied with a sort of unificationist or epistemological anarchist method. This would involve the combination of a variety of data sources from a possible variety of ontological vantage points under a set of basic patterns and principles. Utilizing those patterns and principles, the information can be organized and mined for further relevant information.

[10] Godfrey-Smith, Peter. *Theory and Reality*. 1992. p. 184.

[11] Godfrey-Smith, Peter. *Theory and Reality*. 1992. p. 185.

Does computational information design strive to give accurate representations of aspects of reality that are unobservable?

Not all products of computational information design desire to reach this level of abstraction from information, but some intend to do so from the start. The evaluation of meta-level information or statistically gathered data can provide more dimensionality to the visualization of data sets.

What is unobservable to us may be detectable to computational apparatus, although scientific anti-realists have issues with this. Our belief in the detectable falls short of our belief in the observable, as we are reliant on previously created means to aid in detection. Our belief in those means defines the reliability of the detected outcome.[12]

[12]Godfrey-Smith, Peter. *Theory and Reality*. 1992. p. 186.

How does transposition of empirical data into temporal forms provide un-biased information insemination?

Controlling bias is a problematic issue in science, considering it can affect a project all the way back to data acquisition. When it comes to the presentation, there are two forms of bias that can occur: bias between the displays of different information elements, and bias between observers.

Computational information design aims to combat bias between information elements by designing a single algorithm for the translation of data into formal attributes, and propagates that algorithm into the separate agents responsible for their own temporal appearance.

The code still requires a structure that follows that of solid logical scientific method, and the transposition to abide by communication design guidelines. Although many developers create multi-agent systems that share the same code, the transposition process may be inherently flawed and give the viewer a misguided impression of the underlying information.

When bias occurs between observers, it is because there are elements in the design system that are not easily readable across a diverse audience. This can be because of the conventionality of certain visual devices, or the cultural context that they have been presented in previously.

There is also the dichotomy between "left-brained" and "right-brained" individuals, whom tend to absorb information in different manners from each other. While a table of numbers may be the most quickly understandable visual device by a right-brained person, a scatter plot may be more quickly understandable by a left-brained person.

Does transposition of information determine the empirical result of the image analysis as instrumentalism?

This is an understandable vantage point to take, as computational information design most often portrays a singular perspective of that of the creator, causing its form to take only one of the multiple possibilities of representation.

The transposition process is one with a certain amount of artistic liberty given to the creator. There are guidelines for the understandable presentation of information, but there is plenty of room for creativity and innovation. The fact that amidst the scientific methods their lies a singular moment of artistic influence causes many to view the product as anti-realist, and acts merely as an instrument for analyzing the theory, purporting certain qualities for its representation.[13]

It could be argued that due to the wide assortment of options for representation, that any form of visual transposition would be that of instrumentalism. Any filtering of the raw data into a visual organization acts as an instrument for analysis from a singular perspective. With the transposition of any form, one must justify the reasons for doing so, and explain the improvements in understandability of the subject matter.

On the other hand, communication designers have developed such a strong background in statistical representation that it is possible to logically justify the temporal structure of information elements, and would argue that the visual structure really is that of the underlying data. The visual transposition does not aim to describe the physical structure of the data, but rather the numeric or symbolic value of it.[14]

[13] Godfrey-Smith, Peter. *Theory and Reality*. 1992. p. 183.

[14] Fry, Ben. *Computational Information Design*. 2004. p. 12.

Can scientific realism be applied to metaphysical data?

Metaphysical data may give some scientific realists issues since it doesn't exist on the same level as most observable data. Meta-level information helps describe information elements, and therefore may not be observable in the same sense as the data itself.

For example, I have a project in which I continually collect my social bookmarks which have been tagged with descriptive keywords. The pages to which the bookmarks refer are quite observable in their reality, but then there are layers of descriptive information applied to each one, including the date and time it was bookmarked, an extended description, and keywords for classification.

The scientific realist might argue that the keywords, added after the initial observation and cataloging of the information, are merely instruments created by the observer to assist in the processing of such information. On the other hand, metaphysical qualities to the observations themselves, such as date and time, are just as real (although not as observable) at the instance of information acquisition.

The distinction necessary to make for meta-level information is the differentiation between manufactured metaphysical data and acquired metaphysical data. This is the central dichotomy between what can be considered real and what can be considered instrumentalist.

How does computational information design display how to derive a conclusion in a logical argument?

Often times the path to a conclusion is hidden behind technical accomplishments and a visual interface displaying only the end result. This may be sufficient as a product of art, but as empirical analysis it requires some sort of justifiable process that can be repeated with reproducible and even predictable results.

Sometimes the process to a conclusion can be displayed by animating the visual results of the procedure as it goes along. This is a useful tool for playing back elements in time, or iterating through a list of elements to combine their overall structure. The downside to using animation is that you only see a single instance in time at once, and can easily forget what the structure of the system looked like a few seconds ago.

There are various communication design techniques for showing a series of steps at once, such as the use of small multiples. Small multiples are thumbnail representations of a data system in which each one is slightly different along a common axis (whether it be time or density or step in the process, etc). By laying all of the iterations out side by side, it is easy for the viewer to visually compare the steps in the process and see how the conclusion was derived.

However, it's not always so easy to transform the display of the result into a display of the process as well. The process itself is structured in such a manner that it provides a logical language for the computer to follow. This means the process is captured in the source code, and easily displayed in this form (although only readable to the algorithmically oriented). One of my favorite aspects of new media practice is the ability to open source ones projects, and make them readable to everyone. In this same way, the iterations that the project goes through can be displayed to give the viewer a sense of the process and the developer's method.

Does the open presentation of diverse facts in a visual manner lead to a unificationist theory of empirical acquisition?

Unificationists believe that explanation in science is done by connecting a diverse set of facts by subsuming them under a set of basic patterns and principles.[15] This is also a pretty decent description of the goals of computational information design. It can also be said that computational information design is most powerful when utilized from the unificationist vantage point.

[15] Godfrey-Smith, Peter. *Theory and Reality*. 1992. p. 196.

Computational tools make it much simpler to parse large amounts of data to reveal underlying patterns or relationships. With this in mind, a unificationist design system developer could create a system for relating similar qualities between information entities. When applying design systems to this sort of behavior, a web of connections emerges between sources that may have been unexpected to the developer.

Computational information design is about constructing a perspective, and unificationism is about pulling together diverse information elements within a common perspective for analysis. It's only natural that the two find themselves working together.

It can even be argued that no form of computational information design can be non-unificationist. The fact that computational information design utilizes a decentralized structure of information organization lends itself to unificationism. Computational information design also stresses the ability to create a set of basic patterns and principles for organizing the unique information elements. Lastly, the emphasis on emergent formal structures clearly indicates the intention for certain design systems to develop unificationist theories on particular subjects.

Should computational information design purport its results to be true, or merely claim certain virtues for its display?

The underlying values to the content may actually be determined independently by the design system, often taking the responsibility over the content out of the developer's hands. The design system itself is all about containing virtues for that style of display. There are always multiple methods for displaying information, but one must justify the use of a particular design system methodology above the others.

It is the task of the designer to weigh the options for visual transposition and come up with the best method, as well as justify the purpose for creating that particular perspective. The two main categories for visual optimization of information are speed of understanding and capability to understand multi-dimensional datasets.

When optimizing a system for speed of understanding, a developer creates a design methodology that will utilize pre-attentive objects to represent information elements. This allows an understanding to be gained from the visuals without any active viewing.[16] As information elements decrease in relevance, their visual appearance decreases in prominence.

When a design system is created specifically for analysis of multi-dimensional data, the developer is intending to communicate details of the data set that would otherwise be unseen from other perspectives. This is, in a sense, augmenting the viewer's capability to understand the composition of the system.

By creating a design system, the developer must be purporting certain virtues for this perspective of display, or else they would not be creating it in the first place.

[16] Fry, Ben. *Computational Information Design*. 2004. p. 35.

Lev Manovich

Lev Manovich is a professor of Visual Arts at the University of California San Diego where he teaches new media art and theory. He received his MA in cognitive science from New York University, and his Ph.D in Visual and Cultural Studies from the University of Rochester, where he did his dissertation *The Engineering of Vision from Constructivism to Computers*.

Manovich has published numerous books and articles on new media theory, including his acclaimed *The Language of New Media*. Since 1999 he has given over 180 lectures on new media art and theory all over the world.

How does representational art provide the basis for information arts?

Representational art always maintained the goal of representing an underlying subject, emotion, or theme. Often times these aspects are part of the unobservable, intangible realm. It seems as though it lies somewhere within human nature to make sense of the things we do not yet understand, and help communicate the things we do understand to others.

Information arts tend to carry on the goal of representation, otherwise there would be no point to the use of the underlying information. Representation can come in many forms depending on the goal of the artist, from straightforward and analytical to abstracted and aesthetic.

It is merely a possibility that with technological breakthroughs it has enabled us to open our conception of representational art by utilizing a variety of computational means. Perhaps the limitations to the early forms of representational art only existed because of the tools and techniques they had at their disposal.[17]

[17] Manovich, Lev. *Social Data Browsing*. Feb. 12, 2006. http://www.tate.org.uk/netart/bvs/manovich.htm

How can statistical practices and the scientific method provide tools for new forms of representational art?

The development of technology obviously extends the capabilities of those that use it in a variety of different manners, so why should it not have an affect on art? By augmenting the artist in a means that allows them to express or convey ideas that would have previously not been accomplishable, it opens a window into a new genre of art.

Various new technological practices such as data mining, statistical language modeling, and spidering allow for a diverse assortment of new applications. When at one point it was only possible to attempt to create an accurate representation of an individual, it would now be possible to create a representation of a group, class, or institution.[18]

In order to give the results of such technological practice a sense of objectiveness and consistency, they are applied using the scientific method. The logical process allows the progression and development of the project in a more research oriented manner. Such projects tend to turn out as part speculative artwork, part science experiment.

A great deal of modern representational art creates a methodology for dealing with the real in order to develop an abstraction and represent it. Through creation of a methodology, the artist can present an instance of themselves in formal language in which to control and transpose the information. It's this type of formal instruction that ensures a consistent processing of the data. The artist has complete control over the creation of the instructions, but at the same time, the results are entirely dependent on the input stream.

[18]Manovich, Lev. *Social Data Browsing*. Feb. 12, 2006. http://www.tate.org.uk/netart/bvs/manovich.htm

How does computational information design diminish or increase social justice?

The interesting facet to computational information design is its ability to communicate concepts and perspectives that otherwise go unseen. By providing new perspectives, the viewer is augmented in a manner that allows them to approach a given situation with more preparedness or insight. It also provides the viewer with more knowledge about the situation, thus more power over its influence.

When applying computational information design to a social justice scenario, it is being done so in a way that augments a given party to give them more knowledge and more power over the situation. This power can of course be utilized to diminish or increase the social justice in a given situation.

Institutions that have already amassed great power (corporations, governments, etc) have already found the effectiveness of organizing the massive amounts of information they collect into easy to comprehend visual representations. They can use their augmented ability to organize and understand information to keep track of customers, citizens, targeted demographics, social groups, etc. This allows them a great deal of understanding to the inner workings of the individuals or groups, while at the same time convoluting and concealing their own internal processes. This overpowering of the individual can lead to a diminishment of social justice.

On the other hand, the access to computational information design tools is available to individuals with the right knowledge and initiative. By utilizing the same tactics that the large institutional entities use, but targeting them towards the massive amounts of convoluted yet publicly available data about and produced by those institutions, the table can then be turned towards the individuals favor. The individual can augment their own understanding and surveillance of that institution, thus increasing social justice for the common good.

How does computational information design reflect the postmodern condition?

"The postmodern would be that which, in the modern, puts forward the un-presentable in presentation itself; that which denies itself the solace of good forms, the consensus of a taste which would make it possible to share collectively the nostalgia for the unattainable; that which searches for new presentations, not in order to enjoy them but in order to impart a stronger sense of the un-presentable."[19]

What Lyotard conveys about postmodernism sounds an awful lot like the attempts made by computational information design to express the previously un-presentable in a new temporal manner. However, computational information design stresses creation with good form as it is an attribute of the communication process, which is in itself an essential piece of postmodernity.

Many people identify the postmodern condition by the commoditization of knowledge.[20] As knowledge becomes a commodity, those that control the commodity wish to make the largest profit margin. With this in mind, new methods of information acquisition, storage, and management have been created to decrease expenses and increase overall productivity. Only recently though has it become apparent that to further maximize understanding of knowledge collections new visual forms of representation should be created. It's interesting that in this situation, postmodernism and post-industrialism usually go hand in hand.

[19] Lyotard , Jean-Fransois, *The Postmodern Condition: A Report on Knowledge, trans. Geoff Bennington and Brian Massumi* (Minneapolis, 1984), p. 81.

[20] Wikipedia. *Postmodern Condition.* Apr. 17, 2006. http://en.wikipedia.org/wiki/Postmodern_condition

Is the necessity for computational information design a reflection of a post-industrial economy?

As I stated in the previous section, the commoditization of knowledge is a major attribute to the postmodern condition. It just so happens that post-industrialized society is defined by the use of information, knowledge, and creativity as new raw materials for the economy.[21]

The economy is naturally driven by certain factors; supply, demand, cost, gross. The first key to any business is keeping costs as low as possible while pushing the gross earning as high as possible, thus widening the profit margin. In order to execute this kind of control over business, one has to have as much forethought and knowledge about the situation as possible to determine which costs are important and which are extraneous. Maximizing efficiency is the key to lowering the costs involved with any business.

Computational information design is all about maximizing efficiency of understanding, which in turn may allow one to maximize the efficiency of the subject being understood. It seems only natural that this process is a byproduct of economic factors leading to the maximization of productivity.

Not all computational information design is driven by economic factors, though. Much can be attributed to the simply increasing information age that has presented itself, regardless of financial influence. The creation of more information necessitates new means for the consumption of that information.

[21]Wikipedia. *Postindustrial*. Apr. 18. 2006. http://en.wikipedia.org/wiki/Postindustrial

Should we consider computational information design as virtual navigation?

To virtually navigate something implies that you're not actually navigating the content itself, but rather some sort of simulation. It's important to stress the fact that the information is not to be filtered, only scaled in manners that allow for its intuitive readability. This type of transposition does not create a simulated instance of the information itself, but rather a new perspective on the real.

The type of navigation through information is unlike navigation through a physical space. Rather than seeking paths through the space, the viewer seeks trends within the data. The concept of navigation through information is more analogous to understanding rather than direction-finding.

The term navigation is still apropos to the actual action undertaken by the user. Understanding often comes through the navigation of an information narrative, which in a sense is quite similar to direction-finding, only through elements of information. Even if a narrative was not intended or structurally specified, it can emerge through the simple rules applied to information elements. The relationships that form between information elements create new narratives for the viewer to visually traverse.

Computational information design takes on a sense of the virtual when information is collected about a source that is unobservable by regular human senses. These virtual displays usually embody the ideals of presenting the numeric or symbolic values of the underlying information, rather than the physical re-manifestation of the collected data.

Does computational information design cross the divide between linguistic and visual representation techniques?

This is a somewhat misleading description of what computational information design is capable of. In a sense, it is a description of the capabilities that exist in most high level programming languages, especially those of a graphical nature. It is not unique to the act of computational information design, however it is utilized.

The construction of a design system usually begins with the analysis of a problem. In order to figure out a method of solving the given problem, the developer must create a linguistic instruction set that is capable of handling a situation (since a problem of representation in computational information design is usually dynamic in nature). This instruction set comprises the logical qualities of the scientific method.

The reason why this description is misleading is because representational techniques in the form of linguistics can take on either an instruction set for analysis of the situation (a narrative of how the situation is handled) or as a description of the situation at hand (the end result of the situation after processing).

Computational information design utilizes the first example, by creating a linguistic methodology for analysis, usually specially constructed for each design agent. This type of linguistic instruction representation is the same methodology the designer would implement in hand drawing if computational means were not available. The instructions are not so much a representation of the system as a whole, but a representation of the analysis.

Does computational information design revolutionize the individual's accessibility to information?

Since computational information design is intended to augment the viewer's capability to understand the underlying information, it does tend to revolutionize the users accessibility to that information.

Accessibility can be gauged on two tracks: availability of information, and simplicity of insemination. It is important that both of these factors be taken into consideration when creating a design system. Availability of the information regards the actual presence of the information, while the simplicity of insemination is all about how the information is actually arranged.

Availability of information has become a decreasing problem with the onset of new information management techniques, but the way it is absorbed and understood is an increasingly growing field of importance. Computational information design has taken a huge stride in the presentation of information for understanding.

While the arrangement and clarification of large data sets has been around for a while, it has rarely been implemented into situations of dynamic information content. New information distribution techniques necessitate the use of computational methods to handle large amounts of dynamic content. It simply could not be done using the old methods of designing by hand. Rather than collecting information and waiting for it to be transposed so that we may reflect on that information instance, we can prepare a design system to transpose information that has not yet been received, allowing us to absorb and analyze content live.

Wassily Kandinsky

Wassily Kandinsky is credited for being one of, if not the first abstract expressionist artist, founder of non-objective painting, and is acclaimed as one of the most important artists in the 20th century. His synaesthetic approach to painting allowed him to create and communicate some of the greatest work in the history of modern art.

Kandinsky taught at the University of Moscow where he created the Academy of Artistic Science, and went on to teach at the Bauhaus.

Kandinsky published two books: *Concerning the Spiritual in Art*, and *Point to Line and Plane*, both of which covered what Kandinsky considered the science of art regarding constructivist elements and composition.

How can the analysis of basic elements be utilized to ultimately arrive at an adequate graphic expression?

Computational information design follows a principle from communication design to reduce the amount of non-data ink that is used. This heightens the ratio of data representative elements to non-data elements. It becomes quite apparent that the less extraneous bits of visual distraction can be eliminated in favor of a clearer understanding of the underlying information.

This concept actually predates communication design and can be found in the fundamental practice of Russian avant-garde abstract expressionism. The construction of distinct forms was done with absolute care and precision, as each element is representative of a portion of tension.

"Aside from its scientific value, which depends upon an exact examination of the individual art elements, the analysis of the art elements forms a bridge to the inner pulsation of a work of art."[22]

The construction of a piece should be limited to only those elements necessary to portray its "inner pulsation". As such, the composition as a whole reveals more about that underlying information than analysis of the elements individually.

[22]Kandinsky, Wassily. *Point and Line to Plane.* 1926. p. 17.

How do pre-attentive visual elements compare to abstract constructivist elements?

Abstract expressionist works, especially those following along the lines of the Russian constructivist period, convey a certain compositional resonance due to the placement and organization of its elements. These elements can be analyzed individually, but communicate the most information about the inner content by the way they interact with each other. It's this same principle that allows computational information design to implement pre-attentive visual elements in a successful manner.

The purpose of pre-attentive visual elements is to communicate information to the viewer without them actually needing to think about what they are seeing. Pre-attentive elements are processed by the brain faster than 10 milliseconds, and are usually used as contrasting components to an otherwise similar system of elements.[23] This is used by computational information design applications to display trends, phenomena, and relationships to other elements.

While abstract expressionism seeks to convey an inner pulsation from the artists mind, it tends to differ from the representation of information systems. However the compositions created by both act in a similar manner, conveying a combined resonance of all the elements. This resonance is just as poignant a communicator of the state of the system as it is a communicator of expression.

The main difference between abstract expressionist elements and computational information design elements is the range of information transmission from macro to micro. While expressionist compositions rely on the combination of all elements to convey its content, it is impossible to glean information from the individual elements outside of the context of their relationship with others. With computational information design, elements are truly representative of a unique information entry, and its formal qualities are determined by such. This means that even without the relationships formed with other elements around it, one is still able to understand the state of that individual element.

[23]Fry, Ben. *Computational Information Design*. 2004. p. 35.

How does a composition of visual elements transposed from numerical values outweigh the numerical values themselves?

Visual elements contain a sort of literality that makes them ideal for the presentation of raw data, as opposed to just presenting numerical values by themselves. The layperson can understand the escalation of a curve as representing the ascent of numerical values, and can instantaneously understand the amount and rapidity of the increase. One would have to scan an entire chart of numbers a few times over to get a sense of trends and changes.

When one makes comparisons of numerical values, they can construct a visual analysis of the representation in their head. This is an unnecessary effort as computational information design can construct these images, allowing the viewer to make numerical sense from a technical graphic.

There is a certain amount of danger in transposing raw materials into graphic forms, since doing so without care can create illusively misrepresentative visual elements, which in turn communicate the wrong impression of the data to the viewer. A common occurrence of this is when people attempt to transpose numerical values to two-dimensional shapes (ie: a 1x1 rectangle has an area of 1, a 2x2 rectangle has an area of 4. The same applies to spherical shapes).[24]

Another valuable attribute to graphic transposition allows for the careful overlaying of different data sets on the same scale. This allows the viewer to spot phenomena or correlations between the two data sets. Attempting to visualize this in ones mind is incredibly difficult and most often inaccurate.

[24]Tufte, Edward. *The Visual Display of Quantitative Information.* 2001. p. 55.

Do products of computational information design contain an inner resonance, and if so, how is it useful in the analysis of the content?

Computational information design can produce results that contain a certain "inner resonance", which can basically be construed as a holism of the visual aspects. An information holism is the representation of a finite set of information in its entirety for the purpose of spotting trends within the macro scale.[25]

[25]Levin, Golan. *Information Visualization as Artistic Practice*. 2004. Apr. 30, 2006. http://andrew.cmu.edu/user/golan/datavis/

Information holisms have the tendency to reveal forces within a data set. They are useful at determining influence in interaction, relationships between elements, and analyzing efficiency. Holisms often have a more dramatic communicative impact on the viewer, since they present a perspective usually far larger than what the average human being can comprehend from the independent parts of a system. This type of impact is equivalent to the inner resonance produced by abstract expressionist visualizations.

Abstract expressionist requires the same sort of holism to convey its inner resonance, so it seems only logical that the information holisms created by computational information design convey some sort of analytical inner resonance. As I mentioned earlier, the compositions created by both abstract expressionism and computational information design create a resonance which is equally valuable at communicating information and expression.

How does quantitative or qualitative reinforcement of elements assist in readability of information?

The representation of quantities and qualities differs greatly in computational information design from abstract expressionism. Different uses of quantitative and qualitative reinforcement in abstract expressionism are meant to convey the inner resonance behind the pictorial and graphic aspects. When applied to computational information design, quantitative and qualitative reinforcement is usually scaled down to the lowest common denominator for the sake of readability. It is not always essential to display quantities literally, but rather the scaled relationships between elements that convey quantities.

Quantitative reinforcement is something that can be utilized in an emergent system. Through the creation of simple instructions, each information element determines its own temporal qualities. When many elements are presented, all functioning separately based upon their underlying information, they have the opportunity to emerge in a manner that creates quantitative reinforcement. If the location of the temporal forms is somehow used to signify a quality to the element, then the collection of many elements in one area clearly indicates a system wide phenomenon, which is communicated through quantitative reinforcement.

Qualitative characteristics often follow a different path, where they're used to convey attributes for certain elements. A design system is created with the intent of revealing trends in attributes, and by assigning a certain qualitative characteristic (for example, color) to a certain data attribute (for example, a numerical range from 0 to 128) it conveys information about that single element. When multiple elements are introduced to the stage, all conveying their unique attributes, a sense of reinforcement occurs in the system which communications trends in the data qualities.

It is often the most elegant and interesting solution to convey qualitative attributes for each unique element, and allow any possible quantitative reinforcement to occur based on the interaction of the system.

How does computational information design act to make visible the invisible?

The entire purpose of computational information design is to make visible the invisible by transposing invisible information elements into visible temporal forms. This transposition process is unique to the creator's vision, giving the developer the ability to present a perspective on the invisible. Since the information elements can be transposed in a variety of ways by different people, it is fair to say that multiple perspectives can be used to analyze a singular data value, perhaps giving a higher level of insight into the structure of the underlying system.

The perspective that a developer applies to an information system usually determines the intent under which to analyze the information. A perspective may be specialized to detect a certain occurrence or correlation between information elements, or it may be loosely defined to discover emerging attributes not expected before creation.

Although much of the underlying information was visible to begin with, it's only stored in a numerical or structured string format. Analysis of this raw data by the human brain is difficult, which leads us to transpose it into graphical forms for our understanding. However, there are elements to information systems which are in fact invisible, and those are the interesting facets that emerge from computational information design systems. The meta-level information derived from the interactions between information elements communicates to the viewer a much higher and more in depth understanding of the system as a whole, which simply can not be seen by viewing raw data.

In many cases it is not the data itself that is invisible, but the system in which the data resides, interacts, and behaves. The system as a whole is greater than the sum of its parts, and by assigning simple instructions to each part, a gestalt is revealed.

How does the basic plane affect the communication of information elements?

Like in abstract expressionism, the basic plane is an important aspect of the composition to take into consideration. While in non-objective painting it is utilized to emphasize an inner resonance, it often takes on a more precise and analytical role in computational information design.

When approaching a problem in computational information design, one must decide how the elements will be positioned on the basic plane. This often has some correlation with a specific data attribute, utilizing placement as a communicator of information. In order for that to work, one must analyze the aspect ratio and shape of the basic plane to fit the range of the particular data attribute being transposed for X and Y locations.

Often times the basic plane will fit some form of cultural conventions for the presentation of that type of data. This is an easy way of formatting information in a manner familiar to the general populace, giving them a clearer understanding of the underlying information. However, not all products of computational information design can fit with current cultural conventions, and new forms of presentation must be created to satisfy the communication requirements.

In computational information design, the basic plane has been transformed from the two dimensional pictorial canvas to a three dimensional spatial volume, and extended even further into four dimensions as a product may evolve over time. The extension of the basic plane should only be used when necessary. After all, simplicity is knowing when less is too little and more is too much.[26]

[26]Maeda, John. From an advertisement in WiReD Magazine.

Conclusion

Computational information design is the convergence of several disciplines; computer science, communication design, statistics, and graphic design, all of which must be combined and taken into consideration simultaneously by the developer. However, the combination of such disciplines requires a re-evaluation of other factors such as the philosophy of science, ethical and social implications, pedagogical effectiveness, visual aesthetics, and conceptual basis.

It is clear that the field of computational information design has evolved from previously distinct disciplines out of necessity. Increasing amounts of information combined with new information management techniques has changed the way information can and should be consumed. The optimization of information absorption is influenced economically and politically, and has a humongous impact on policy and decision making that should not be taken lightly.

It has also become apparent that with the development of new technologies, that new forms of artistic expression have become available for analysis of those technologies in critical, speculative, and social ways. Computational information design has become an apt tool for representational art, and will no doubt evolve to meet the needs of artists and developers as technology progresses.

Bibliography

Clark, Andy. Natural Born Cyborgs. New York: Oxford University Press. 2003.

Fry, Ben. Computational Information Design. Cambridge: Massachusetts Institute of Technology. 2004.

Godfrey-Smith, Peter. Theory and Reality. Chicago: University of Chicago Press. 1992.

Ishizaki, Suguru. Improvisational Design. Cambridge: Massachusetts Institute of Technology. 2003.

Kandinsky, Wassily. Point and Line to Plane. New York: Dover Publications. 1979.

Levin, Golan. Information Visualization as Artistic Practice. 2004. Apr. 30, 2006.
< http://andrew.cmu.edu/user/golan/datavis/ >

Lyotard , Jean-Fransois, The Postmodern Condition: A Report on Knowledge, trans. Geoff Bennington and Brian Massumi. Minneapolis. 1984

Manovich, Lev. Social Data Browsing. Feb. 12, 2006.
< http://www.tate.org.uk/netart/bvs/manovich.htm >

Reas, Casey. Software Structures. New York: Whitney Museum of American Art. 2004.
< http://artport.whitney.org/commissions/softwarestructures/text.html >

Tufte, Edward. The Visual Display of Quantitative Information. Cheshire: Graphics Press. 2001.

Wikipedia. Emergence. Apr 12, 2006.
< http://en.wikipedia.org/wiki/Emergence >

Wikipedia. Object Oriented Programming. Apr. 10. 2006.
< http://en.wikipedia.org/wiki/Object_Oriented_Programming >

Wikipedia. Postindustrial. Apr. 18. 2006.
< http://en.wikipedia.org/wiki/Postindustrial >

Wikipedia. Postmodern Condition. Apr. 17, 2006.
< http://en.wikipedia.org/wiki/Postmodern_condition >

Wikipedia. Simula 67. Apr. 12, 2006.
< http://en.wikipedia.org/wiki/Simula >

Winner, Langdon. The Whale and The Reactor. Chicago: University of Chicago Press. 1986.