Triangles: Design of A Physical/Digital Construction Kit

Matthew G. Gorbet
Tangible Media Group
MIT Media Laboratory
20 Ames Street
Cambridge, MA 02139 USA
+1 617 253 0128
mgorbet@media.mit.edu

Maggie Orth
Opera of the Future Group
MIT Media Laboratory
20 Ames Street
Cambridge, MA 02139 USA
+1 617 253 0804
morth@media.mit.edu

ABSTRACT

This paper presents the design of a new form of computer interface that uses physical objects to embody digital information, providing a means for interacting with data. *Triangles* is a physical interface in the form of a construction kit of identical, flat, plastic triangles. The triangles connect together *both physically and digitally* with magnetic, conducting connectors. When the pieces contact one another, specific connections can trigger specific digital events, allowing a simple but powerful tool for physically interacting with stored digital information. We will briefly describe the system itself, and design steps taken in its development. We highlight the importance of collaborative, iterative, and multi-disciplinarian design for a project of this nature.

Keywords

Interface design, physical interface, digital connector design, magnetic connector.

INTRODUCTION: The Triangles System

The *Triangles* project is an effort to create a 'tangible interface'[1] – a system of physical objects for embodying and manipulating digital information. Programming languages and computer systems often make use of 'objects' and 'kits' as metaphors for managing information. This can make concepts easier to understand, but manipulating these systems requires using an abstract syntax to assemble basic elements into more complex structures. If we could apply the benefits of physical construction kits like puzzles or tinker-toys to such systems, we could achieve a significant user interface advantage. Complex physical structures can be easily assembled using two hands and basic elements can be identified and sorted quickly. The ability to pick up and feel complex structures can sometimes offer the user more information about complexity than a typical computer visualisation.

CHI '97, Atlanta GA USA

Copyright 1997 ACM 0-89791-802-9/97/03 ..$3.50

Triangles pieces are identical, flat, plastic equilateral triangles, each containing a PIC microprocessor. They can be tiled on a flat surface or used to construct three dimensional forms. The triangles connect together *both physically and digitally* with magnetic, electrically conducting connectors. The data connection is as immediate, easy to make, and easy to undo as are connections between cardboard puzzle pieces. When the pieces connect, information about their identities is exchanged, and messages are relayed to a computer (Fig. 1). In this way, an application can determine relationships between all of the connected pieces at any time, and specific connections can trigger specific digital events.

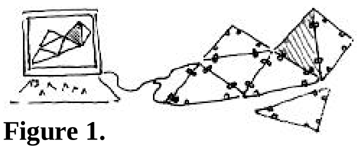

Figure 1.

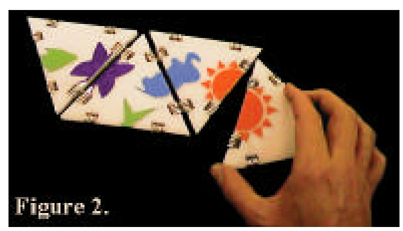

Figure 2.

By placing icons or pictures on a triangle, an author can give each piece and each of its sides a unique identity, suggesting meaningful couplings of the pieces (Fig. 2). The user can then connect two or more 'appropriate' triangles, exploring the particulars of the author's application. The user can also create three dimensional 'data structures' – polyhedrons such as pyramids, constructed of individual triangles and their symbolic properties (Fig. 2b). The triangles can be reused for different applications by changing their images and corresponding software events. In one instance, the triangles could be used as a storytelling interface; in another, the same set of triangles could become a method for configuring an audio or video system.

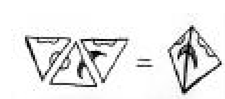

Figure 2b.

MOTIVATION AND CONCEPTUAL DESIGN

Creating everyday objects endowed with communication and sensor technology that will play an essential role in our daily interactions with computers is a difficult design task. Until now, much of the effort towards the design of such objects has led to a practice of simply 'gutting' existing objects and inserting chips or other components into them. While tagging and tracking objects such as plants and coffee cups may seem appealing, the specific functioning of such an assortment of objects is unclear. What will they say to each other? How will they say it? With these issues in mind, we set out to design a new type of object – an object whose physical and digital design grew together, each reflecting the needs of the other.

For objects to communicate digitally, they must either be physically connected with one another or be able to transmit information. The shape of most everyday objects is highly specific, reflecting their function and cultural meaning. Although infra-red signals, radio-frequency transmissions, magnetic fields, and copper or fiber cables are methods of digital connection, there is still no standard method to connect objects of various shapes and materials. To create a system where many objects could easily communicate involved overcoming this diversity of form.

Our approach was to create a set of physically identical objects, with 'digital signatures'. This accomplished two goals: the objects could each be given a unique symbolic identity, while easily connecting with one another and exchanging data. We were influenced by existing standardized systems of symbolic objects whose relationships to one another imply meaning, including checkers and dominos. Similarly, puzzle pieces each provide some unique piece of information to create an overall image.

ITERATION AND BALANCE IN DESIGN

The design of *Triangles* was an iterative process of mockup, prototyping and experimentation. Issues arose on three levels: the physical design of the objects, the electronics involved in passing messages between the triangles, and the software that would handle message-passing, both on-board each triangle's microprocessor and on the host computer. It was critical that these three areas of design be considered simultaneously, for they often deeply affected one another.

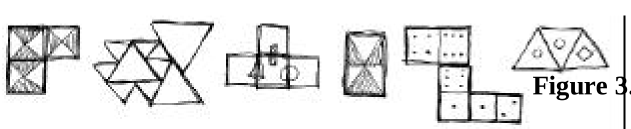

Figure 3.

Geometry

Many shapes were considered for the individual pieces (Fig. 3). Ideally, they would tile on a flat surface, and they would each be identical, so that any edge could connect with any other edge. These criteria were well satisfied by equilateral triangles. Triangles also have the minimum number of sides required to physically reflect the complex possibilities of branching structures and relations in computer information. Tiling the triangles into hexagons can create loops and recursive situations. In contrast, objects with only two points of connection, like stacking blocks, are structurally linear, with only one input and one output. The triangular pieces also allow more complex shapes and three dimensional forms to be built, suggesting applications that demonstrate high-order logic, such as non-linear narrative, programming, or modeling of complex systems.

Sensing

The central feature of *Triangles* is that they communicate their position and orientation information instantly when they are connected. We considered environment-based sensing technology such as computer vision and instrumented desk surfaces, but these require a specific sensing environment, so the system would not be portable. Nor would three-dimensional structures be possible without fear of obstructing the sensing of any objects. It was clear that on-board sensing would be required. We then investigated and discarded other commonly-used sensing technologies such as infrared or inductive coupling in favor of a connection between the tiles that would be both physical (keeping the pieces in contact with one another) and digital (communicating identities and relationships). IR and inductive coupling technologies can operate at a distance, and consequently cannot guarantee simultaneous connections and digital events.

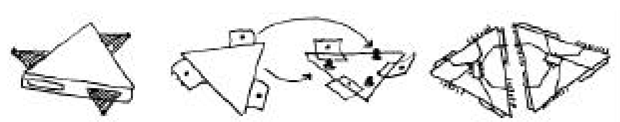

Figure 4.

Physical Connection

Several designs for the physical connectors were examined, including slotted edges, snaps, zippers and even conductive Velcro fasteners (Fig. 4). Many prototypes were built, and eventually it was decided that conductive magnets 'feel' best, when making connections. They make the triangles easy to attach and remove, and because they require no lateral motion, as do zippers or slots, connecting complex forms such as hexagons is easy.

Communication Structure

We needed to design a message-passing system through which each triangle would be able to relay information about the identity of each of its neighbors, and to which edge of each it was connected, back to the host. It was also important to minimize the size and complexity of the program running on each triangle, since extensive debugging and revising of many chip-level programs can be difficult and tedious.

We initially discussed a 'common bus' hardware architecture, whereby triangles would communicate with the host via a single, common line. This is similar to the way computers in a local area network communicate over 'ethernet'. We rejected this architecture because although each triangle would easily be able to identify itself using such a system, identifying neighbors and relationships between triangles would require additional local communication among the edges of the triangles. This would require extra circuitry, programming, and more pins on the connectors. Instead, we decided that our system should use only local message-passing between each triangle to route messages back to the host connector.

Message-Passing Software

Several communication schemes were explored, including random 'broadcasting' and regeneration of messages to every triangle with the expectation that the host would eventually receive the message, 'maze-walk' algorithms to attempt to find the host connection, 'topography maps' stored within each microprocessor, and 'perimeter-hugging' to trace a path back to the host. Each of these has significant flaws that were not always immediately apparent, but that prevent such algorithms from working well with low-memory microprocessors arranged in 2D and 3D triangular tilings.

The message-passing algorithm which we finally implemented is based on a 'gradient-descent' algorithm (Fig. 5), whereby messages are passed along a gradient, established between the host computer and each of the triangles. A good analogy is that of altitude: if we think of the connection to the host computer as being at the lowest altitude, then our gradient is established by slightly augmenting the 'height' of each new tile as it is added. This ability of the *Triangles* network to 'self-organize'[2] ensures a sloping, direct path from every tile back to the host connection. When any triangle generates or receives a message, it simply seeks its *next lowest* neighbor, and passes the message along in the 'downward-sloping' direction. This system guarantees that each message will eventually reach the host computer, and requires each tile to store very little topographical information. It also avoids redundant and unnecessary passing of messages.

Figure 5. Simulation using the 'gradient-descent' algorithm.

This method for acquiring and relaying topographical information minimizes demands on each microprocessor in terms of memory and functionality. Each PIC chip only has to store five items of information: its unique ID, the most recent IDs of its three neighbors, and its 'height' value. Functionally, each chip needs to be able to perform the following functions: set its height value, poll its neighbors (detecting changes in IDs and heights), and generate and pass messages. The circuitry required for this system is also fairly simple in that each triangle needs only a direct local connection to each of its neighbors.

Physical Design

Since equilateral triangles are radially symmetrical, connections such as magnets, snaps or Velcro, which have 'male' and 'female' components (or polarity, in the case of magnets) needed to be arranged so that 'male' would always meet 'female' and vice-versa (Fig. 6). Depending on the design of the electronic circuit, issues of symmetry arose in ensuring that each transmit pin would meet a receive pin, and that shared pins, such as power and ground, would always find the correct mate when connected.

Figure 6.

Single-pin connectors

The first prototypes we built with magnetic connectors worked very well. They had batteries on-board, and thus required only two pins. However, we decided that it was important to avoid batteries for a number of reasons. First, lithium batteries are fairly large, and would take up a significant amount of the triangular circuit-board. Batteries also require regular replacement and on/off switches. Requiring the user to explicitly turn each triangle on or to change the power cells weakens the hybrid (digital/physical) nature of our objects, slanting them more towards physical objects with digital components. For this reason, we decided to provide power through one pin on the data cable.

Multi-pin connectors vs. rectified power

In order to provide power and electrical ground as well as data transmit and receive lines for each edge of each triangle, we needed more than two pins on each edge. When we built a prototype multi-pin magnetic connector, a new issue arose. By their nature, magnets have fairly low precision tolerances, and they needed to be mounted on the triangles so that their edges would always meet, to guarantee electrical conductivity. This made a fixed mount for multiple magnets practically impossible.

Multiplexed power and data over a single pin

We investigated circuit designs that would provide 'multiplexed' data and power over a single line, allowing us to go back to the two-pin connector model. Such a circuit was designed, but in order to avoid a common data bus, it would have involved elaborate filtering and rectification circuitry for each pin. This would have been difficult to implement in the small physical space of our triangles, and although it would have solved the power problem, the triangles would always be limited to just two pins. Simplifying the connectors made the circuitry more complex, and vice-versa. Rather than implementing such a complex and limiting electronic circuit, we returned to the physical connector design.

Flexible mounts for multi-pin connectors

After further experimentation with multi-pin connectors, we were able to design a mount for the magnets that would allow them to move towards or away from each other slightly as they made contact. The magnetic attraction would automatically attract each pin's mate, ensuring a good electrical connection. Designing a non-standard connector with moving components created problems in prototyping and fabrication, but these problems were not insurmountable, and the advantages of having multiple pins were clear. Multi-pin connectors simplify electronic circuitry and create the potential for future expansion of the system to handle new types of data messages like sensor readings from specially outfitted triangles.

Application Programming Interface

By concentrating on the very basic design of the objects themselves, we have produced a system that could be used in a broad range of possible fields, from non-linear storytelling to genetic programming. As our prototypes mature, we hope to offer the *Triangles* system as a platform for further research in these fields. This goal raises further design issues, at the level of the application programming interface (API).

We hope to provide developers with a powerful, yet simple-to-use API, with which they can build a variety of applications. To that end, we specified function calls for the API very early on in the design of the system, and have been writing applications using this API with a simulation of the *Triangles* system even as we develop the physical components. This design practice has enabled us to rapidly refine and build upon our initial software libraries, finding flaws and weaknesses in our design before the circuits have even been designed.

In order to provide the greatest flexibility to authors using this system, we are also designing a suite of GUI software tools for configuring the *Triangles*, in addition to the C++ libraries of the API. These tools will allow non-programmers to easily associate digital events with the physical action of connecting two triangles, to the timing of specific interactions, or to the resulting configurations of triangle groups. Children could use this tool to write and explore their own narratives, or researchers might use it for rapidly prototyping extremely complex relations of objects.

One Application Scenario

Early in the development of *Triangles*, we devised the following storytelling scenario using four triangles connected to a multimedia PC. The story consists of audio, visual, and text information about an owl, a frog, and a dragonfly (Fig. 7).

When a user connects the triangles to complete the bell symbol and the frog, the frog's voice is heard: "Ribbit… ribbit… ribbit." The user then disconnects the bell and completes the 'picture frame' symbol. The computer displays a photograph of the frog. Now the user completes the dragonfly with the triangles, and suddenly the frog on the screen eats a dragonfly. If the user disconnects the dragonfly and instead completes the owl, an owl swoops down and eats the frog. Had the user connected the book symbol at any point, text concerning the current characters would have appeared, providing more detail.

By recording the history of a user's interactions in a given session, the software system allows authors to use this information to trigger time-sensitive events. For example, had the owl eaten the frog before the dragonfly appeared, the dragonfly might have survived.

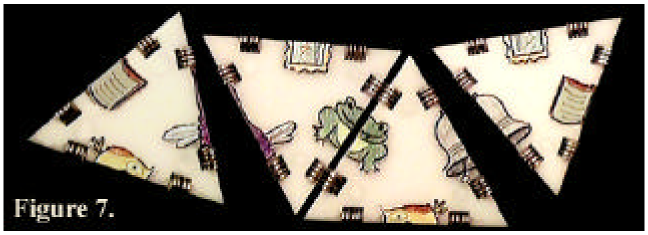

Figure 7.

PROJECT ORGANIZATION

Both of the researchers involved in this project had extensive prior experience which helped in the design of the *Triangles* system. Each had been involved in large-scale interactive projects which combined art and technology, and each had their own design perspective. One was well-versed in the physical and mechanical design of objects, and the other had extensive experience in software systems and traditional user interface design. Each of the two designers benefited from their ability to determine what aspects of the design were unrealistic in terms of implementation, and to clearly understand and discuss aspects of the others' field. Thus, a tight design loop was established, with software, hardware, concept and electronics all maturing simultaneously. New hardware and software prototypes were made weekly, and new concepts and decisions discussed daily. The rest of the implementation team was also kept small, consisting of undergraduate students with very specific skills, performing directed tasks.

CONCLUSION

Integrating the needs and constraints of materials, electronics, software and mechanics, the *Triangles* project has been a good example of a multi-disciplinarian approach to physical interface design. The development of *Triangles* has demonstrated the importance of exploring objects whose entire design, materials, and properties support the transfer of digital information. Objects whose initial goals involve digital communication will lead to new methods of physical interface with information, rather than mirroring models of the past through metaphor. Such objects are not aimed at making existing computers more comfortable, invisible, tasteful or simply smaller. They are aimed at radically re-defining the way we interact with and understand digital information. The design of such objects is an undertaking that requires new thinking and very close collaboration between designers from fields that have until now been quite distinct.

REFERENCES

1. Ishii, H. and Ullmer, B. Tangible Bits: Towards Seamless Interfaces… to appear in Proceedings of CHI '97 (Atlanta GA, March 1997), ACM Press.
2. Smith, Joshua R. Self Organizing Device Networks Submitted to Physica D. 1997.