3. HyperGami

3.1 Description

HyperGami (Eisenberg and Nishioka, 1994, 1997) is a programmable design environment for paper sculpture implemented in Scheme to run on the Macintosh platform. The interface of HyperGami consists of a Scheme interpreter; a window of polyhedral starting shapes; a ThreeD window showing a three-dimensional representation of a polyhedron; a TwoD window showing the corresponding folding net for the polyhedron; a palette of shape rotation options; and palettes of drawing and painting tools. Appendix A is a gallery of mathematical sculpture designed with the software.

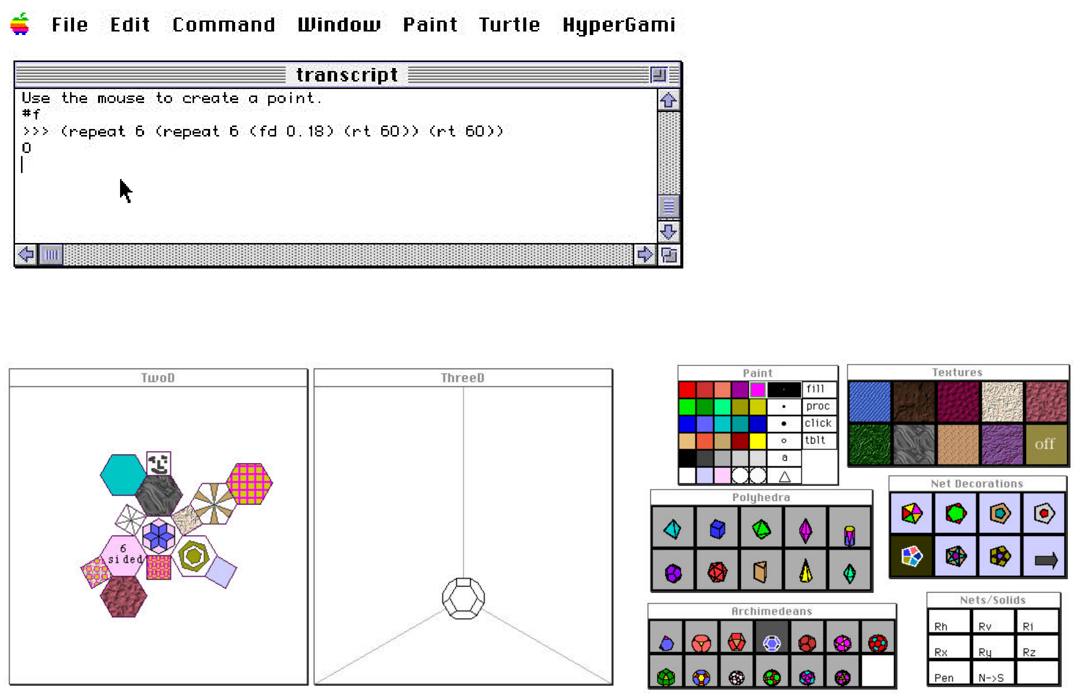


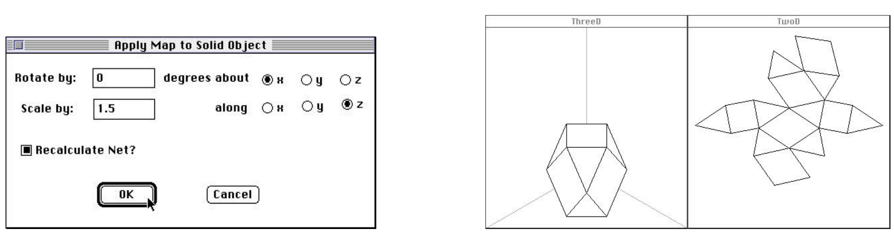
Figure 3-1. A view of the HyperGami screen in the course of a typical scenario. The TwoD and ThreeD windows toward the bottom left show the folding net and three-dimensional rendering of a truncated octahedron, respectively. The folding net has been decorated with textures, patterns, solid colors, a hand-drawn figure, a turtle-drawn design, text, and geometric designs. The transcript window at top allows the user to type expressions into the MacScheme interpreter. The windows toward the bottom right include tools for choosing, decorating, and viewing polyhedra (several other optional windows are not shown).

• A Sample Interaction with HyperGami

A student begins work with HyperGami by clicking on one of a number of starting polyhedra -- these are shapes which have been pre-built into the system. A rendering of the polyhedron appears in the ThreeD window and the corresponding folding net appears in the TwoD window. In early work with the system, a student will often investigate the folding nets of the various shapes already built into the system and then select a folding net and decorate it with colors, textures, or patterns. He may then print the net to the color printer and assemble it into a paper model.

Once he is more familiar with the software, the student may employ direct manipulation (pointing and clicking with the mouse), or he may write a Scheme expression to apply a function to the starting shape. Among many other functions, he may add pyramidal caps to faces, stretch or shrink the shape along the x, y, or z-axes, truncate the shape at a vertex, or slice a shape into two parts.

The figure below shows an example of one particular (and widely used) function that can be accessed via direct manipulation in HyperGami. Here, the beginning shape is a cuboctahedron; a menu choice brings up a dialog box through which the user can specify a linear scaling map along any of the three coordinate axes. In this case, a linear map is applied to the shape in which all of the z-coordinates in the solid are multiplied by 1.5. The system displays the new solid and after a brief calculation, draws the new folding net:



(a) (b) (c)

Figure 3-2. Applying a linear map to a polyhedral object by direct manipulation. The student (a) selects a menu choice and the system (b) displays the new solid and (c) the corresponding folding net.

Although some shape operations -- like the scaling operation shown above -- can be performed solely through direct manipulation, most shape operations involve a combination of using the mouse and typing expressions into the Scheme interpreter. For example, to add a pyramidal cap to a dodecahedron, the student types an expression such as:

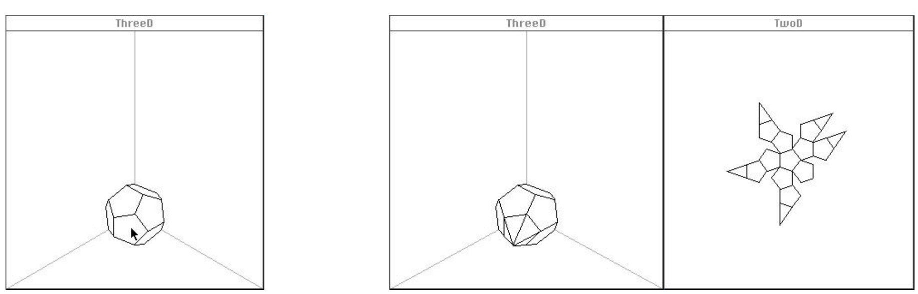
```
>>> (define *penguin-head*
     (add-cap-vertex *current-solid-object*
                     (read-a-current-solid-face)
                      1.0)))
```

Here he has defined a new solid object named *penguin-head* by applying the function add-cap-vertex to the *current-solid-object* (the current shape). The cap will be placed on the face that he clicks on (read-a-current-solid-face), and will have a height of 1.0. He then clicks on the face to cap, and the system will display the new solid object when he enters:

```
>>> (3d-show-solid-object *penguin-head*)
```

The software will generate the corresponding folding net when he enters:

```
>>> (calculate-net-and-make-current-solid *penguin-head*)
```

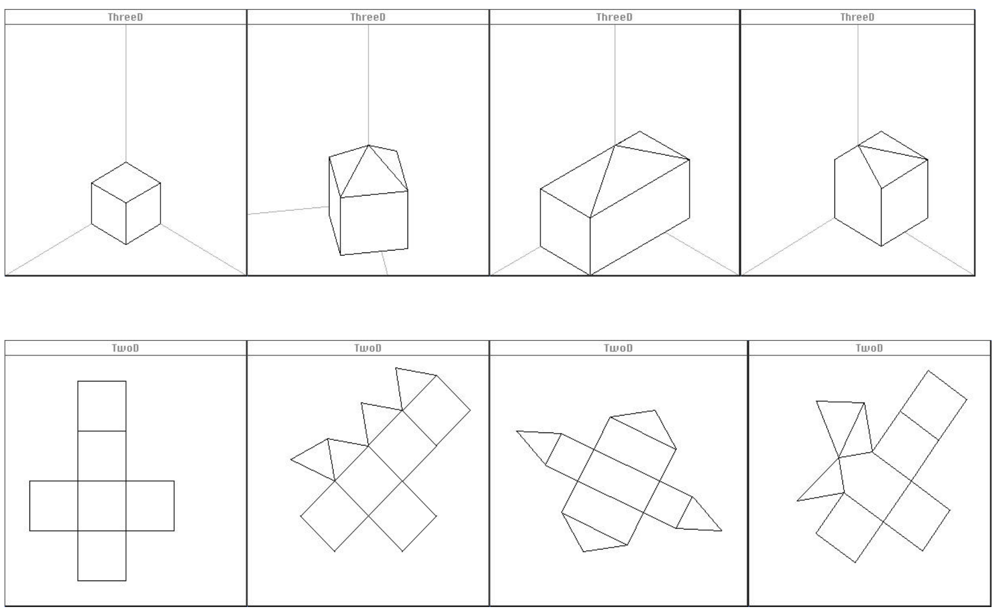


(a) (b) (c)

Figure 3-3. Creating a capped dodecahedron object from a dodecahedron. The student first (a) enters a Scheme expression in the transcript window and clicks on the face to cap. The new solid object is shown in (b) and the corresponding folding net is shown in (c).

- Shape Algebra

Central to the design philosophy of the software is a *shape algebra* -- that is, the idea that operations on shapes will create new shapes which may in turn be operated upon. Figure 3-4 below is a sample of the operations that can be performed on a cube to turn it into a new solid which is capped, stretched, and sliced. The software may be employed to generate a folding net for the solid each time a solid operation is performed. HyperGami has to date never failed to generate a folding net for any convex solid. It has also successfully generated folding nets for a large number of non-convex polyhedra.



(a) (b) (c) (d)

Figure 3-4. The cube (a) is capped (b); stretched along the x-axis (c); and sliced (d). Corresponding folding nets generated by the software are shown under the solid objects.

3.2 Work with elementary, middle, and high-school students

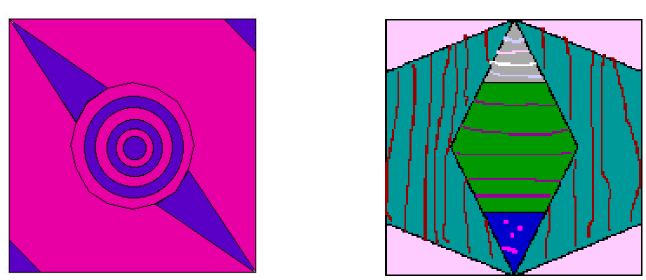
Work with children on the HyperGami system provided feedback which was to have an impact on the design of its JavaGami successor to be discussed in Chapter 4.

A brief summary of the testing of the software with children as well as examples of their work follows in the next two sections; the design lessons taken from HyperGami pilot testing follows. A discussion of children's spatial learning as a result of using the software follows in Chapter 5.

3.2.1 HyperGami Experiences of Children: Origami and Polyhedra

Pilot studies with elementary- and middle-school children interacting with HyperGami took place from Fall 1994 through Spring 1997. This form of pilot testing with children involved their attendance once a week for one-on-one or small group interaction with the software. A total of 21 students -- 8 boys, 13 girls -- took part in sessions. Their levels of participation averaged 11 weeks, although four students chose to work for the system for a year or more.

The earliest system building in HyperGami (cf. Eisenberg and Nishioka, 1994) included efforts toward creating an interface for the design of origami objects and mathematical toys such as hexaflexagons (cf. Gardner, 1957); thus, children in our first pilot group (Fall 1994) created decorated origami forms in addition to simple polyhedra. After Fall 1994, virtually all our efforts in software implementation and educational activities focused on polyhedral models.



(a) (b)

Figure 3-5. Folding nets for (a) an origami snail, and (b) an origami turtle decorated by a pair of girls, ages 8 and 13. The children often worked by coloring "outline nets" for which we had previously computed the boundaries for the main fold lines. The folded sculptures are both pictured toward the bottom of Figure 3-6 below.

Figure 3-6. Student work from Fall 1994 - origami snails, turtles, frogs, and a giraffe; a rotating mathematical toy 'hexaflexagon' in the center; a modular origami piece on the left; a dodecahedron; and a great stellated dodecahedron. All figures were decorated by students using the early HyperGami system.

The change in implementation emphasis from origami to polyhedral sculpture was in part driven by the difficulty that our students had with the manual folding tasks involved in origami. A typical origami creation involves multiple -- and often intricate -- types of folding. Although children tended to enjoy sessions in general, the manual folding tasks proved to be the primary source of difficulty and frustration for them:

> [LEI, age 13]: Folding. I can't fold very well. I get confused ... it's kind of hard.
>
> [EMM, age 8]: I didn't like making some of them because they got hard sometimes -- like making the frog or the mask ... the folding part ... I'd fold it and either it wouldn't stay or I wouldn't have folded it right.

Another major reason for moving away from origami implementation was that it is difficult for children (and for us as well!) to invent original origami forms -- an important part of making a paper sculpting experience completely one's own. All of the origami figures that the children decorated and folded were based on the work of professional origami artists (cf. Fuse, 1990; Rojas, 1993; Aytüre-Scheele, 1990). We wanted children to have the freedom to go beyond what we had pre-built into our software and to develop their own sense of sculptural style.

3.2.2 HyperGami Experiences of Children: Polyhedral Sculpture

After about a year of experimenting with origami forms, we decided to concentrate on polyhedral paper sculpture for both implementation and design reasons:

1. Creating a polyhedral modeling environment with real-time unfolding into two-dimensional nets was more feasible for us to implement than modeling real-time origami paperfolding.

2. Hollow polyhedra involve joining together at edges rather than the type of folding upon itself found in origami. Our students found this type of construction much more accessible than origami folding.

3. Polyhedra are rich with opportunities for mathematical experiences. This is not to say that origami does not offer mathematical experiences -- many activities in combinatorics, trigonometry, and plane geometry can be developed in the context of origami folding (cf. Fuse, 1990), but the mathematical aesthetics of polyhedral objects themselves can be appreciated on a wide range of levels. The nature of the shapes can be enjoyed by our youngest students, yet the shapes provide mathematical puzzles rich enough to challenge even college professors.

4. Polyhedra can be used as building blocks for larger, multipart paper sculpture.* This idea was central to the evolution of HyperGami and JavaGami, and crystallized in

---

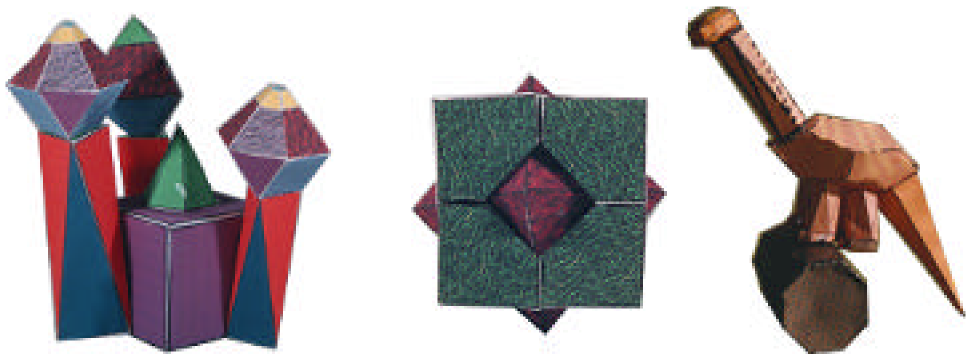
* It should be noted that HyperGami and JavaGami do not support windows for displaying multiple polyhedra on-screen. Instead, the student develops the polyhedra in a multipart sculpture individually and then glues the shapes together when assembling the model.

the development of the set of polyhedral penguins shown in Figure 3-7 below. It is much more feasible for students to create their own building blocks and then their own paper sculptures out of polyhedra, as opposed to trying to design original origami figures.

Figure 3-7. *Penguinhedra* developed by the author illustrate the idea of polyhedra as building blocks for paper sculpture. The heads of the penguins are made by capping dodecahedra, their bodies are stretched cuboctahedra, and their feet are custom prisms. The bowtie on the large penguin is made from two tetrahedra.

After our initial group of students in Fall 1994, students working with us concentrated on the design of polyhedra and polyhedral sculptures. Sessions were for the most part student-directed: students selected sculpture topics which included dinosaurs, circus scenes, various holidays, pets, buildings, and monsters among many other categories. After some guidance and brainstorming about the types of polyhedra they might use and the operations that they could perform to customize starting polyhedral shapes, students spent one or more sessions both employing the software to design their shapes and using real-world construction materials such as glue, tape, and scissors to assemble the folding nets of their actual paper models.

These pilot tests were performed for a range of purposes: to determine whether children could use the software and the types of difficulties they experienced; to see what children would build; to determine appropriate age levels for different activities with the software; and to study children's spatial thinking as a result of interacting with the software (discussed in detail in Chapter 5).



(a) (b) (c)

Figure 3-8. (a) A polyhedral castle designed by a 8th grade girl; (b) a polyhedral sculpture by a 6th grade boy; and (c) a dinosaur standing on a log by a 7th grade boy.

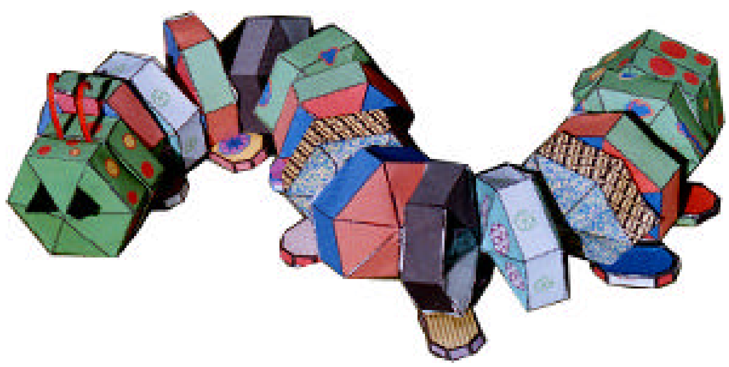


Figure 3-9. A *caterpillarhedron* which was a collaborative effort between six children and their parents. Each child decorated two body segments and two feet and then attended a polyhedra-building party where children and parents assembled the pieces of the sculpture.

Students have built a wide range of sculptures over the years that we have pilot tested HyperGami. They have generally worked on projects individually or in pairs, but have also participated in larger-scale group activities. The "caterpillarhedron" in Figure 3-9 above is an example of a group effort. A few samples of individual sculptures are shown in Figure 3-8 above, and additional samples of elementary- and middle-school student projects are found in Appendix B.

Using polyhedra as the building blocks for paper sculpture provided a way of making mathematical activities accessible to children at different ages, varying levels of software sophistication, and different levels of manual dexterity. The youngest children who worked with us were able to complete simple one-session projects of which they were immensely proud (for example, a third grader created a "raft" by fashioning a few prisms together); our middle school children created more complex sculptures like the ones shown in Figure 3-8, and as we will discuss in the next section, high school students employed HyperGami to create satisfying sculptures as well.

3.2.3 Work in Schools: High School Students

Older students in grades 9-12 either employed HyperGami to create projects for part of a design-based mathematics class, or enrolled in a technology-based crafts workshop taught by the author and M. Eisenberg at a local area high school. The mathematics-class based sessions involved 6 groups of students, with one group of 3-8 students per quarter for a total of 17 boys and 16 girls over a period of six school-year quarters from Feb. 1997 through May 1998. The craft technology class enrollment was 13 students (8 boys, 5 girls) and took place from Sept. to Oct. 1998. The total number of high school students who worked with the system is 25 boys and 21 girls, for a total of 46 students.

The students were self-selected volunteers and the number of students who wanted to participate often exceeded our resources, so a lottery of interested students was drawn to select the ones to participate for the quarter. Students in groups 1-6 worked with us for an average of 3 hours a week for 4-5 weeks. Sessions took place in a computer lab adjacent to their mathematics classroom. Students in group 7 (the craft technology class) met two hours per week for 8 weeks.

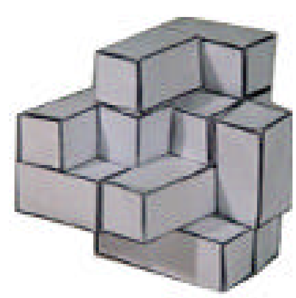
The students came from a wide range of academic backgrounds. Most were average students; a few were the top students in their mathematics classes; some had been previously diagnosed with learning or attention-deficit problems; some were enrolled at the school because of high interest or aptitude in the visual arts or music.

Work sessions with the high school students were more directed than the pilot studies with the younger children both because of the size of the class and as a result of the students' need for more concrete direction. The first few sessions with a high school group consisted of instruction in various aspects of the software: decoration, applying linear maps, capping, slicing, truncation, and so on. Once the students learned the basic techniques, they worked on individual projects.

Student projects included mathematical sculpture and puzzle-like objects (cf. Holden, 1971), a lantern made by inserting a flashlight into a polyhedron, animals including a bear, frog, flamingo, bird, rhinoceros, and dragon; rockets and jets; and a basketball player. A few samples of student work are shown below; others are pictured in Appendix C.

(a)

(b)



(c)

Figure 3-10. (a) A capped cuboctahedron sculpture by two 9th- and 10th- grade girls (printed in black-and-white and colored with markers) (b) a paper rocket by a 9th grade boy (c) a polyhedral sculpture by a 9th grade boy based on a design by mathematician Alan Holden.

3.3 Design Lessons Learned

Although students were able to make use of HyperGami to create sculptures such as those pictured above, a variety of design problems in the system became evident through these sessions. The students' difficulties are described briefly here; these same issues will be addressed more fully in the next chapter in the discussion of design decisions chosen in the implementation of JavaGami.

3.3.1 Students were confused by the language interface and found the amount of typing required to enter Scheme expressions too difficult.

> [TMO, 8th grade boy]: I think the most difficult thing I had to do was typing in some of the programs ... because they were hard to memorize and they were also hard to type ... Some things need to go into the menu bar instead of into the transcript window -- the random color effect, the entire list of colors, and the start-up.
>
> [IAM, high school boy (who became very adept at writing Scheme expressions)]: ...put buttons in and stuff instead of having to type codes ...
>
> [KEP, high school boy]: take out the use of the programming language
>
> [KRW, high school girl]: have it explain itself better -- not having to put in a bunch of words ...
>
> [ALD, high school girl]: not so many codes
>
> [KAS, high school girl]: not have to type so much in the transcript

This is the design issue which by far came up most often with students of all ages. Elementary and middle-school students in general did not have enough skill in typing; older students -- even if they were reasonably adept at typing -- were daunted by the Scheme interface.

For the majority of the students, this was their first exposure to any kind of formal programming language (although a few had spent a small amount of time with Logo). Absorbing the concept of the programming language along with learning to use the shape-modeling environment seemed to be frustrating for the students to do in the limited time we had during class sessions.

3.3.2 Students found the coordinate interface too cumbersome.

The folding net in HyperGami is displayed in a window with a Cartesian coordinate system. If the folding net is cut off by the window at the edges, the student selects a menu item and resets the window coordinates as shown in Figure 3-11 below.

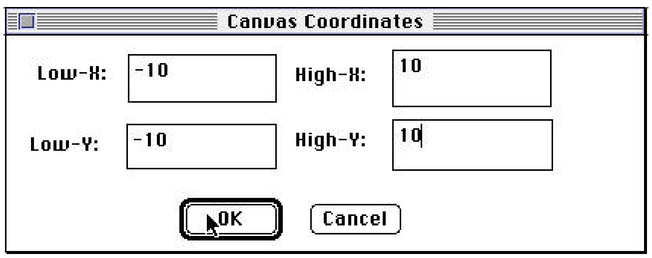


Figure 3-11. The interface for changing the coordinate scale in HyperGami.

This was especially difficult for the third through sixth graders to understand, since some had not had much experience with 2D coordinates in school:

> [ULR, 5th grade boy (an advanced math student for his age)]: The x and y coordinates -- sometimes it was confusing to get the x and y coordinates because ... I don't know how, but make an easier way to do the coordinates.

3.3.3 Students were confused by the multiple-window interface.

> [AVG, high school girl]: Too many windows! They freeze and redraw too much!

Students of all ages sometimes lost track of where various tools or functions were located. They also sometimes forgot to first click on a window to activate it. Monitors in the high school classrooms had 12" screens, so it was difficult for the students to work with all of the necessary windows at once.

3.3.4 Students grew impatient/sometimes needed to be reoriented to the activity when the net generation times were very long.

Depending on the computer used and the solid to be unfolded, net generation times could take up to a few minutes. Our youngest students had the most difficult time waiting while folding nets were generated, and even the high school students grew distracted when they had to wait more than a few minutes for the folding nets. This led to some general classroom disruption among the high school students, and as a result we were faced with the occasional task of reorienting a number of students to the activity.

3.3.5 There were too many steps needed to decorate folding nets.

> [ARD, 8th grade girl]: Some of the functions are different from what I'm used to, like, you have to click a lot of things before you can fill in the things, and I wasn't really used to that ...
>
> [TAJ, high school boy]: have easier drawing tools ...

Students needed to remember a number of steps when they decorated their folding nets. For example, to fill a folding net with a solid color, the student would need to click on the "fill" box, then select a color, then double-click the folding net. Then, to add a pattern, he would first select colors for background and foreground and click on the corresponding spaces in the Patterns window, then he would need to click on the pattern, click on the "On" square on the Patterns window and finally he needed to be sure that the "fill" button was active in the Paint palette. Our students used the paint tools primarily by trial and error, but it would eliminate frustration to have a simpler interface for the decoration tools.

3.3.6 Students sometimes were confused about *what* to color.

Because they could draw and paint on the ThreeD window as well as the folding net, students sometimes became so involved in coloring the three-dimensional picture of their sculpture that they forgot that they would be printing out the two-dimensional folding net to assemble -- there was no way to "map" the designs on the three-dimensional shape back to the folding net.

3.3.7 Students were not likely to use the software without adult guidance.

This is most likely due to a combination of the following factors: (1) The software requires an initial set-up step when it is booted. (2) When a student makes an error, the software drops into the debugger, which requires that the child realize that the software is in this state -- he must then enter a keyboard command to leave the Scheme debugger and enter another command to begin working again. (3) The software runs best with virtual memory off for the Macintosh, a setting that is ordinarily easily toggled through the Macintosh control panel interface. However, the high school computer lab had the computers networked and password-protected so that students could not get into the control panels to reset the memory configuration.

3.3.8 Summing up

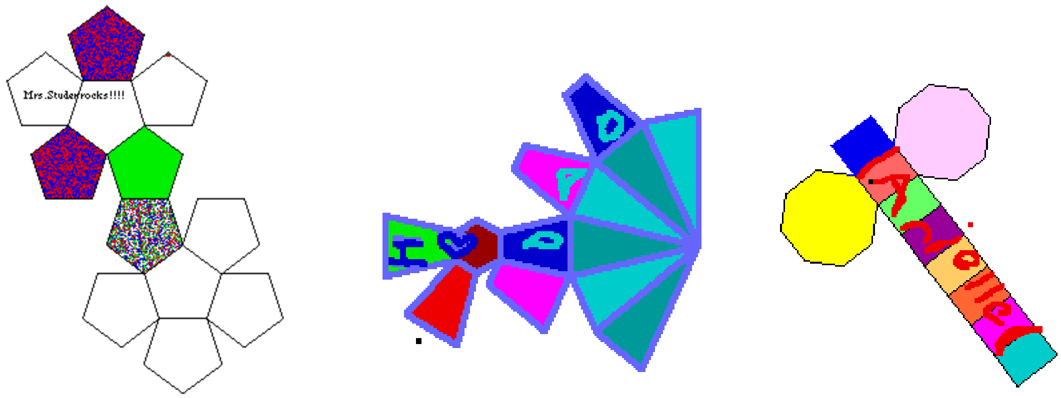
Some of these design issues -- such as net generation times, multiple window confusion, and number of steps to generate folding nets -- are relatively uncontroversial, while others such as the language interface and the coordinate interface represent legitimate tradeoffs. Section 4.3 in the next chapter provides a description of the steps taken in the implementation of JavaGami to discuss these issues and also discusses tradeoffs in design when such decisions were made.

3.4 Themes Discovered

It was not the case that we started out with HyperGami knowing what questions were important to ask about the software. It was rather in the course of our interaction with students that we discovered a few recurring themes--themes that are not often sounded in educational research, particularly that style of educational research that focuses on cognitive issues to the exclusion of emotional or social ones.

For example, at the outset of our creation of HyperGami and our work with children, we did not realize the importance of combining real-world materials with software. After working with children for a few months, however, we became more fully aware of the value that students placed on having real objects that they could take away when they were finished with the day's software session. German educator Friedrich Froebl characterized the materials used in kindergarten -- objects such as balls, blocks, paper, and clay -- as "gifts" (Brosterman, 1997, p. 35), and we have seen our own HyperGami objects likewise take on the role of a kind of *social currency*.

Students gave polyhedra to favorite teachers as gifts (a folding net for one such gift is shown in Figure 3-12a); they created them for siblings (Figure 3-12c); many children presented their creations to parents to celebrate birthdays and other occasions (a gift for a father is shown in Figure 3-12b); they used them to adorn the family Christmas tree; and a few children proudly reported that their pieces were displayed in places of honor at home.



(a) (b) (c)

Figure 3-12. (a) "Mrs. Studer rocks": A folding net for a dodecahedron for an eighth-grade boy's favorite teacher; (b) "I love dad": a folding net for a gift for a fifth-grade girl's father; (c) "Arielle": a folding net for a gift for the same fifth-grader's sister.

In our own work with the system, we have created polyhedral sculptures as wedding gifts and thank you "notes", and we have tailored them for friends to celebrate happy personal occasions. We experienced this kind of social currency exchange on an even larger scale at a meeting of origami artists in Japan -- paper folders created objects and gave them to others as gifts like the ones shown in Figure 3-13 below -- but until now we had not known this phenomenon to take place within the context of work with educational software.

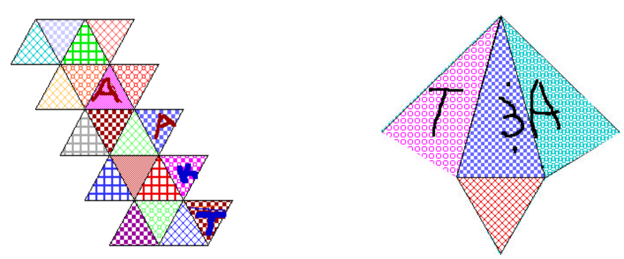
Figure 3-13. Paper gifts received at a conference of origami artists.

In Csikszentmihalyi and Rochberg-Halton's (1981) study on why people value certain objects in their lives, they found that "...objects ... are valued because of the social meanings they embody, such as ties to kin, or effort ..." (p. 80), and in the case of artwork:

> The most surprising feature of people's comments on the significance of paintings and sculpture is that the intrinsic qualities of such objects were rarely mentioned. Instead, art was valued primarily because it recalled memories of events, family, and friendship. (p. 178).

Possessing a tangible real-world object to take away from a software environment leaves the student with a positive reminder of his experience. And unlike printouts of lessons, the object is aesthetically pleasing and reflects the student's own artistic taste while also remaining linked to a rich history of geometry and art.

Children's personalization of their objects -- and their personalization of their experience with the software in general -- is another theme that we found in the course of working with HyperGami. One of the younger children decided to give her dodecahedron the name "Fred", and proudly took "him" to school to introduce to her friends. A different kind of personalization is shown in Figure 3-14a below, where a folding net for a "friendship polyhedron" created by a pair of fifth grade girls contains both sets of their initials. Figure 3-14b below shows folding nets for another such creation by the same pair of girls.



(a) (b)

Figure 3-14. (a) A "friendship icosahedron" net with the initials of two girls. (b) A net for one of the pyramids of a twenty-point great stellated dodecahedron, also with the girls' initials.

In more conceptual terms, we found the older children constructing their own personal analogies of HyperGami's shape algebra. An eighth grade girl compared the process of constructing shapes to painting:

> I think everything builds off these [the starting set of] shapes. They are sort of like the primary colors and you get different shapes with the [basic] shapes ... You can chop off edges, and give it more flat surfaces, and you can change the proportions to make the sides a little longer and the ends a little thinner ...

An eighth grade boy formed a different analogy by comparing the set of "starting shapes" in HyperGami to DNA:

> ...there are many, totally many kinds of polyhedra out there, but they all start from a simple shape, which is basically their 'genes'.

Such emotional concerns--gift-giving, personal attitudes toward paper constructions, and the issue of social currency -- are not the focus of the assessment procedures described later in this thesis. Those procedures focus instead on the more purely intellectual concerns of spatial cognition. But it is worth taking note of them here nonetheless, since it is precisely these emotional factors that lend motivation and interest to children's mathematical work. As Seymour Papert writes in his book *Mindstorms*, "[T]he 'laws of learning' must be about how intellectual structures grow out of one another and how, in the process, they acquire both logical and emotional form." (p. vii).