

## 6. A Case Study in JavaGami

---

### 6.1 Overview

Much of the work with children using HyperGami and JavaGami took the form of case studies. This chapter profiles a middle-school student's work with JavaGami over the course of five weeks in order to convey an idea of the types of activities that children engaged in while working with software, the way sessions were run, and the kinds of interaction that took place between the children and the author. Jesse\* was a seventh-grader enrolled at a local public middle school. His case study was selected for this chapter because he was articulate, frank, and a generally sunny and interested participant -- and on the few occasions when he was frustrated with the software, he did not hesitate to voice his opinions.

Jesse's case study is presented below as a mixture of dialog transcripts, annotations, and screen shots of his work to provide a glimpse into his interactions with the software-in-progress and his thoughts as he designed and constructed polyhedra.

### 6.2 Session #1: Getting Started

Jesse first watched a brief demo of some of the software's functions: selecting shapes, coloring shapes, twirling the wireframe image, and stretching a shape along x, y, and z-axes. Jesse clicked on the different shape icons and then decorated a dodecahedron with solid colors. He asked: "How do I do the edges?" After learning about the idea of leaving "tabs" on the net, he began to assemble his shape.

---

\* A pseudonym breaking with the three-letter convention in this thesis for readability of the chapter.

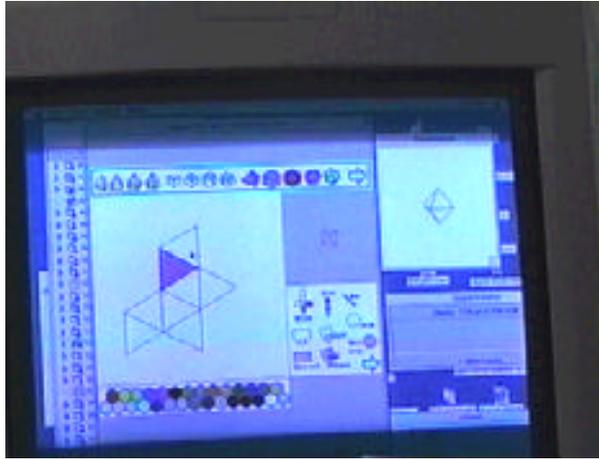


Figure 6-1. An overview of the version of the software Jesse worked with in his first session. The interface features the shapes in a bar across the top of the main screen; function buttons have not yet been built into the Polyhedron window to the right of the main screen.

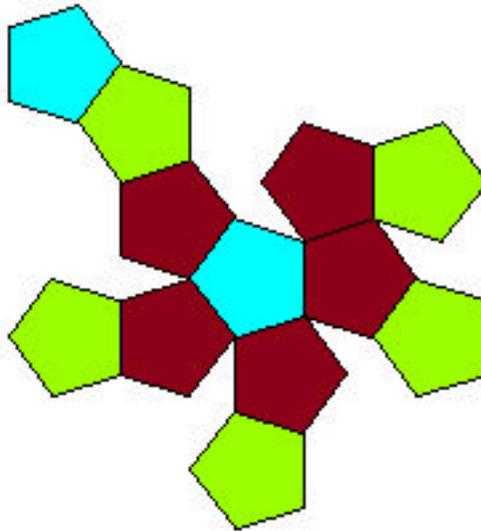


Figure 6-2. Jesse's first net decorated in JavaGami.

As he cut and pasted his shape, he remarked about his decoration method. The colors on the shape turned out in the pattern he wanted -- he evidently had a plan in mind when he was filling the polygons of the folding net:

J: I was trying to do the blue on the top here, and like ...the blue on the bottom, and red, green, and blue ...

When he was partially finished assembling the shape, he wondered out loud about how to glue the last polygon in place:

J: I guess that's a little lopsided, but oh well ... should I do those sides simultaneously, or what? ... I can't get the inside to press out.

We worked on completing the shape, and after a few more minutes, he held up the polyhedron for inspection. He felt that the gluing the last piece was the most difficult part of the process: "there was no way to get my fingers in there."

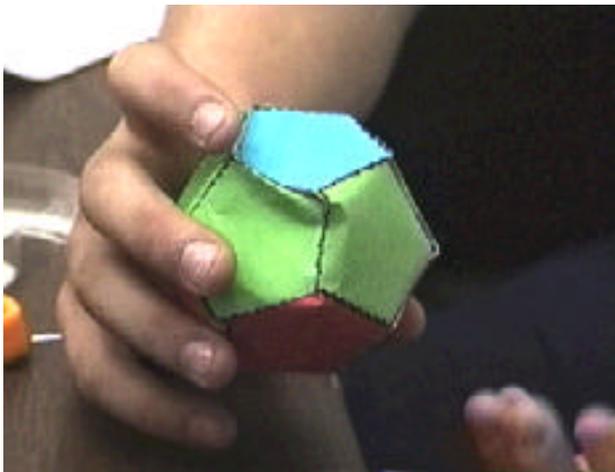


Figure 6-3. Jesse's finished dodecahedron.

Jesse then turned back to the computer and worked on the net of an octahedron. For this net he employed the line-drawing tool in addition to filling the polygons with solid colors. After a few minutes of decorating the net, he asked if he could erase a line, but there was no erase function built into the system.

A: So what are you making?

J: (Without turning away from the screen) Just a kind of neat design to go around it ... it may or may not look that way when it's finished ... this is only how I think it's going to look ...there will be, like, things going around it .... they're not going to match up exactly ... that's okay, right?

(He looked up). So you see what I was trying to achieve -- there'll be lines ... going all around it -- big and small.

(He looked at the design on the shape). Those look cool just like that, too ... like a bird

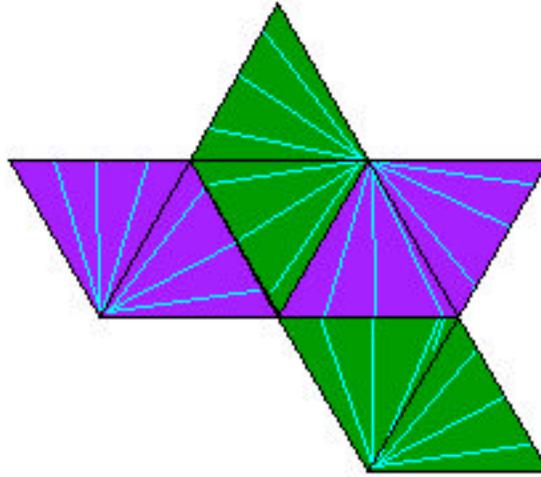


Figure 6-4. Jesse's folding net for the octahedron.

After the shape printed, he examined the result and explained a little more:

"What I was trying to do is so when I fold it, these two meet you know? ... it will get big, and small here, and get big again ... like a kite shape." In both cases -- the dodecahedron and the octahedron -- he had an idea beforehand of the way he wanted designs on the folding net to appear on the solid. He took the folding net home, planning to assemble it.

### 6.3 Session #2: Capping and Stretching

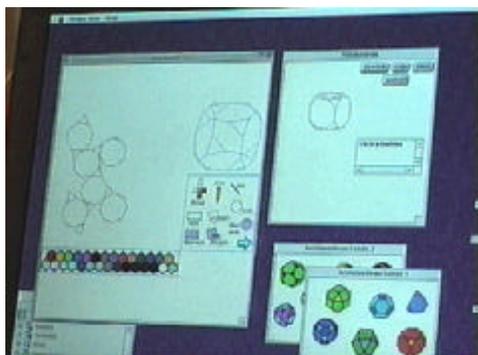


Figure 6-5. A new version of the software which included function buttons and separate windows for selecting shapes.

Upon his arrival, Jesse saw that we would use a PowerBook with a trackpad for the session and hooted with delight -- he touched the trackpad and exclaimed, "It freaks me out! ... It's like something you're not used to, you know, it's not normal." Since his session a week ago, the software had been modified considerably: the shapes had moved from the bar-oriented arrangement in Figure 6-1 into separate windows as shown in the lower right corner of Figure 6-5 above, and stretch, cap, slice, and unfold function buttons were now visible in the Polyhedron window to the upper right.

He looked at the newly built menu to toggle the shape set windows. He tried unsuccessfully to close one of the shape set windows, but the window-close feature had not been implemented. As he worked, he mentioned the folding net that he took home from the previous session: "Oh, by the way, the lines that I did on the other one [the octahedron] -- It worked out awesome."

He chose a truncated cube and asked how to do the "fill circle thing," referring to the polygonal-fill tool. He again mentioned that the trackpad was "weird" and compared it to a car-driving device in a James Bond movie. He asked "Can I draw stuff on it? How am I going to draw with this?" as he tried to figure out how to drag out a line using the trackpad and mouse button; then said, "Yay, I'm so happy!" as he

dragged the line across the screen. He used the line tool to connect opposite vertices on the octagons.

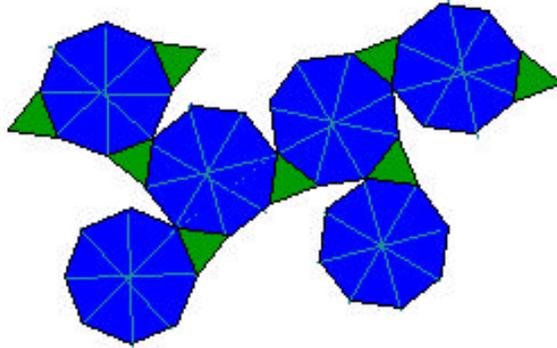
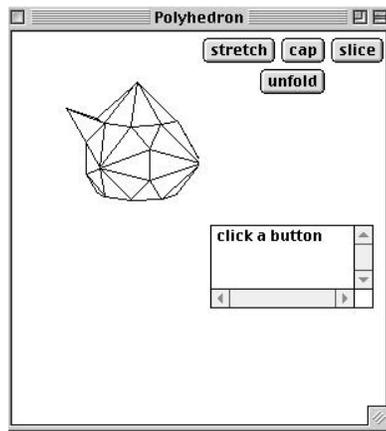
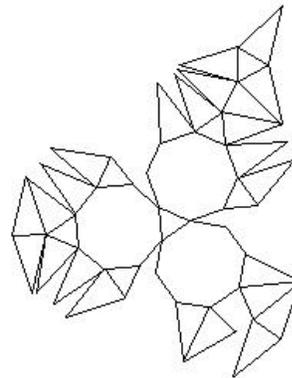


Figure 6-6. A folding net for the truncated cube.

We then went through a brief explanation of how to add a pyramidal cap to a polyhedron, using the truncated cube that he already had loaded. As soon as he saw the first pyramid appear, Jesse wanted to add a second pyramid to the shape. He added the second cap, and then successfully capped a third face. "Yay!" Then he decided to cap one of the caps, but clicked on the wrong polygon: "Whoa! It just capped that one ... can I undo that?" Although there was no undo function for solid modeling, Jesse was still pleased with the result, but decided that he did not actually want to fold the resulting net together.



(a)



(b)

Figure 6-7. (a) A picture of the triply-capped truncated cube, and (b) the corresponding folding net generated by the software.

He decided to work on a polyhedron with fewer faces. He clicked on the truncated tetrahedron and spent a minute or so turning the wireframe image. He tried to stretch the shape, but it apparently did not become stretched in the way he was planning. "Cool! That was the wrong side!" In order to "undo" this error, he needed to enter the inverse of his initial stretching value of 1.5. We talked about the inverse of  $3/2$ , which he first said was "3", but later understood was  $2/3$ . He knew the decimal approximation of  $2/3$ , and entered 0.6 to return the shape to its previous state.

Jesse then tried to cap the truncated tetrahedron, but on his first attempt forgot to select the face to cap. Once he had a successfully capped shape, he twirled the wireframe image and asked "if I rotate this shape [the wireframe] does the other shape rotate?" The solid in the Polyhedron window in fact did not rotate along with the wireframe, but this was a good idea. He decorated the folding net with solid colors, and we chatted about the software while the shape printed. The first suggestion from Jesse was to provide online help in the form of a character:

J: Something like ... a little character that shows you how to do it ... it could be this little guy like that dragon [a paper dragon sitting on one of the office shelves] ... made of JavaGami shapes -- saying, 'I'm here to take you through this program.' Like something that would be fun for kids. I'm sure that kids would want to do this, it's great for kids .. I mean ... At first, you look at it and you don't think, you know, oh wow, this is great for kids.

A: What kinds of tools would you want?

J: Color, definitely ... lines, and maybe, like, draw and you definitely need like a [built-in] print command.

A: Is there anything that could be included to help you understand 3D better?

J: Well, show it folded ... show a picture [of the shape].

He looked at his printout and noted: "It looks like a mercator projection." As Jesse worked, he asked how JavaGami was made, so we paused and took a mini behind-the-scenes look at a small portion of source code. "Why don't you put some of

this, like, in a museum?" he asked, referring to the finished paper sculptures on the shelves in the office.

Once Jesse finished assembling his shape, he took the video camera and did a voice-over as he scanned the shape:

This is [my] shape. Yay ... It started off with kind of the hexagonal shapes ... there were four of them ... and I stretched this hexagon into a cone shape, and made lots of little triangles there. And I taped it together because I forgot to cut flaps. And it's happy.

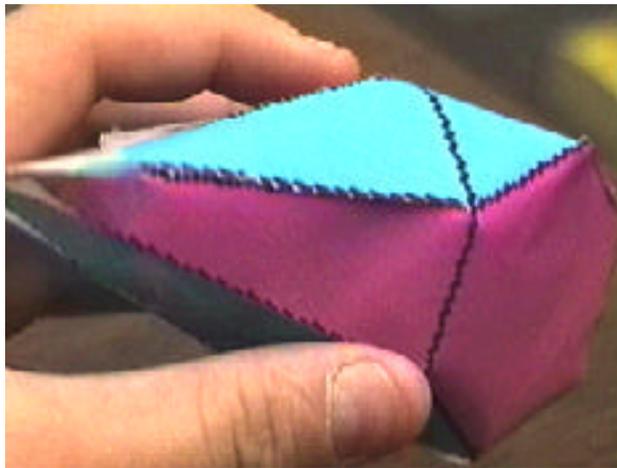


Figure 6-8. Jesse's truncated tetrahedron with a cap.

The author's dog then unfortunately pounced on Jesse's shape. Jesse good-naturedly constructed another and cheerfully remarked that he was glad that he had the chance to make a better-constructed shape, and he would have felt "guilty" throwing the first shape out. He showed it to the camera: "Here is [my] messed-up weird shape number 2 ... same formation ... this one had tabs, the other one didn't."

## 6.4 Session #3: A Goldfish Gift

J: How do I make a fish?

A: Well, you need to think of some fish-type shapes!

J: Make it, like, narrow, an oval ... I guess I can stretch this -- that's the shape I did last time.

Jesse began to think about shapes he would use for a goldfish sculpture as a gift for his mother. As the short dialog above indicates, he started to think about ways to customize existing shapes to fit his project. He stretched a truncated cube, and this time knew the inverse of 1.5 to undo the stretching of the shape:

J: That's a long fish. Can I make it shorter again?

A: You can load the shape back in, or you could do the inverse of  $3/2$ .

J:  $2/3$ ? That's 0.66.

He stretched the body a smaller amount, and then found a good "fishy color" for the net:

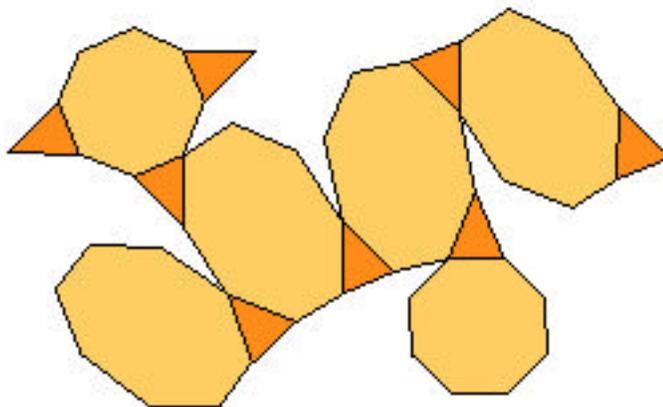


Figure 6-9. The folding net for Jesse's fish body.

In order to work most efficiently with the software, he decided to design all of the folding nets for the fish during the session, and to construct the nets at home on his own. He turned his attention to the head of the fish, and decided to add a cap to a truncated cube for a nose. It looked too pointed, and he wanted to slice off the tip. The slicing function had not been quite implemented at this point, so he could not do this, but he decided to keep the shape for the head.

"Which face is which?" he muttered to himself. He was trying to figure out how to decorate the net to make it look like the fish's head, and he needed to put the eyes onto appropriate places on the net. "I guess I could fold it ..."

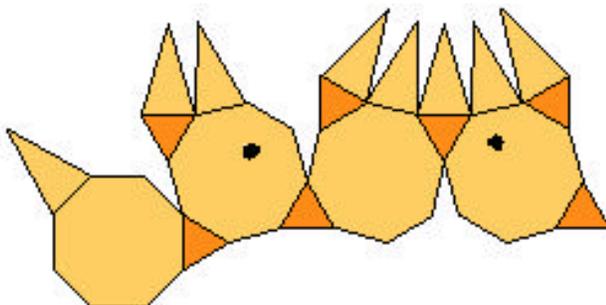


Figure 6-10. The folding net for Jesse's fish head. He took some time positioning the eyes in the correct place.

He then loaded the truncated tetrahedron to work on the tail-section of the fish. "I could cap it here, and cap it here," he thought out loud as he turned the wireframe image. Since it would be difficult to construct a shape with two caps, we decided that it would be easiest to print an uncapped truncated tetrahedron net and then attach the pyramids separately.

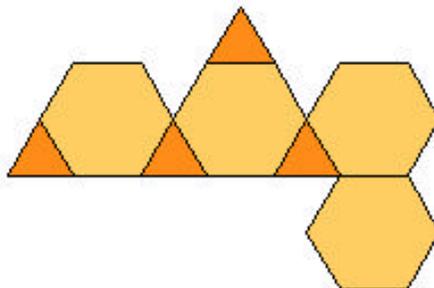


Figure 6-11. The folding net for the fish tail.

Once the tail finished printing, Jesse decided to add a cap to one of the triangular faces. He capped the shape, but felt that the cap was too tall and asked if there was a way to "uncap" it, again pointing out the need for an undo feature in the solid-modeling functions.

#### 6.5 Session #4: Slicing

Shape slicing had been implemented by this session, so after a brief demo of how to slice a shape, Jesse took over. He loaded a dodecahedron and asked: "Can I slice something exactly in half?" He decorated one of the slices and printed it out, as shown in Figure 6-12 below.

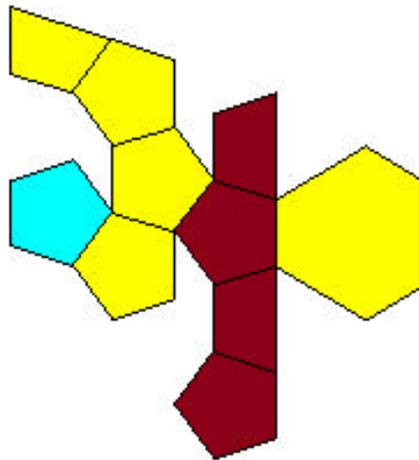


Figure 6-12. The sliced dodecahedron net.

As he cut out the nets, he automatically cut tabs with the rest of the folding net. We talked about his goldfish project from last week as he assembled the dodecahedron slices:

A: How long did you spend working on the goldfish at home?

J: Actually, this was sort of a longer one ... because of all the pieces ... it took me probably 32 minutes ... I mean, sometimes I'd have it, but then I'd realize that the way I folded it wouldn't work -- I did that on the

body once -- I'd have to take off the tape that I'd used. And actually, my head didn't have a base.

A: What did you use for the head? Could you describe what shape it was?

J: It was hexagons ... and then we stretched it ... to make a nose ... so there's the nose and the eyes, and it didn't have a neck ... maybe I chopped it off when I cut it out.\*

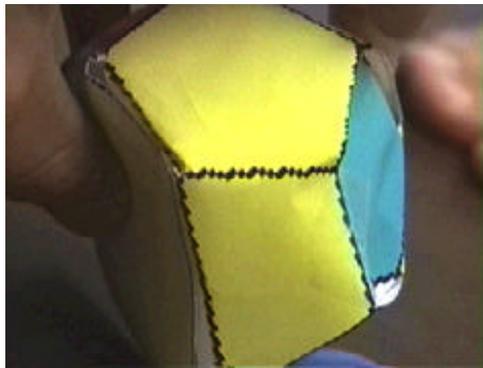
I was never able to finish my shapes by taping them on the inside.

A: So what'd you do?

J: I taped them on the outside.

Once Jesse finished the shape, he took the camera off the tripod and filmed the shape while describing it:

Well, here's my shape, and the colors did not work out the way I wanted them to, exactly... I thought that the yellow would go in a pattern around it, so I didn't expect these two to be next to this yellow. I thought there'd be the yellow here, then the red here, and then another thing of yellow, and then a blue ... I forgot that this would be slanted like this, I figured that it would be flat. I forgot that I'd cut it, so that it was not perfect. So the colors did not work out exactly the way I wanted them to. But the shape worked out the way I expected it.



(a)



(b)

Figure 6-13. (a) Jesse's sliced dodecahedron. (b) A different view of the same shape.

---

\* Jesse worked with a very early version of JavaGami -- the unfolding algorithm was still slightly buggy at this point, and the net for the fish head in Figure 6-10 was indeed missing a polygon.

## 6.6 Session #5: Mystery Shapes

This session was a mystery shapes exercise in which Jesse's job was to employ the software to design the five shapes shown below:

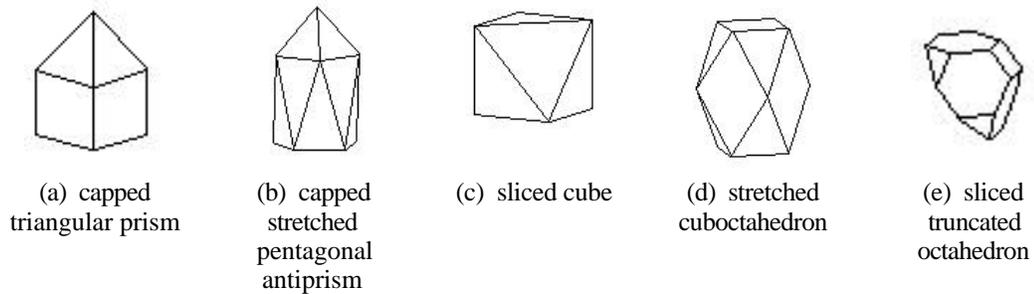


Figure 6-14. Mystery shapes: Jesse was given paper models of these shapes and asked to recreate the shapes in JavaGami.

An annotated transcript of this session is provided on the pages to follow. The time elapsed (in minutes and seconds) and the shape Jesse was attempting to design are shown in the first column, the session transcript is in second, and annotations are shown in the third. A discussion follows at the end of this section.

Table 6-1 (a). Annotated transcript of the mystery shapes session.

<p>1:15</p> 	<p>Jesse loaded a cube, stretched it, and tried to slice it, but was not able to click on one of the vertices all the way toward the back of the shape. He asked what to do.</p> <p>A: All those shapes were made with the same tools you have -- how about trying a different method?</p> <p>3:40 He reloaded the cube and stretched it out again, and this time added a cap. "Yay!" "What do I do if I can only cap two sides of the thing?"</p>	<p><i>Jesse wanted to cap a stretched cube on the four longer sides. The rotation buttons necessary to do this had not yet been built into the system; but this approach would not have yielded the correct shape anyway.</i></p>
<p>4:45</p>  <p>6:30</p>	<p>(a) He loaded a cube, sliced it, but diagonally through the middle. "Aaah ... I see what I did wrong."</p> <p>(b) He reloaded the cube, but this time selected three points at the top of the shape. "Hmmm. How do I slice it like this?" (indicating the points he needs) He reloaded the shape, sliced it, and got the correct shape.</p>	<p><i>He got this shape quickly, after only one incorrect attempt.</i></p>
<p>6:43</p>  <p>8:38</p>	<p>He asked if he could only use the set of basic shapes.</p> <p>A: You can use any of the palettes of shapes.</p> <p>He toggled through the different windows until he got to the prisms.</p> <p>"Aah! I knew it had to be in there somewhere!" He loaded the triangular prism, and after forgetting to click on the face to cap, successfully reloaded and capped it. The cap was a bit high, so he added a lower cap after reloading the shape.</p>	<p><i>This was another shape that was relatively easy for him.</i></p>
<p>8:58</p>  <p>11:11</p> <p>13:58</p>	<p>"This one is beyond easy." He loaded the truncated octahedron, clicked on vertices which would in fact slice off one of the hexagons -- the shape did not unfold. He tried it again, clicked on another set of three vertices, and failed to get slices.</p> <p>A: Why don't you load the shape back in, but be very careful when you slice it?</p> <p>After a couple more minutes, he sliced the shape, but got the wrong cross section. He thought it was the right shape, but could not figure out how to slice it. He set the shape aside and picked out another.</p>	<p><i>Jesse clearly knew that the truncated octahedron was the correct starting shape, but he selected incorrect points for the slicing plane.</i></p>

Table 6-1 (b). Annotated transcript, part 2.

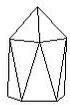
 14:29  16:10	<p>He loaded a hexagonal prism and stretched it in x, then reloaded the shape and stretched it in y, then reloaded the shape and stretched it in z. After comparing the shape the solid he had on the screen, he muttered, and tried to slice it. He exhaled loudly:</p> <p>"I stretched it on the z-axis, and then I sliced it." He looked at the paper model again. This time he loaded an antiprism and stretched it.</p> <p>"Oh, dude! I did it already!" He added a cap and got the correct shape.</p>	<p><i>The error here was that Jesse started from a prism (with rectangular sides) rather than an antiprism. He was trying to slice the sides of the prism to create the triangular sides of the antiprism.</i></p>
 17:40	<p>"Oh, I didn't know there were these shapes (prisms) when I tried to do this one." He loaded a square prism, and stretched it on x: "whoa! it's weird." "I wish you didn't have to click on the corners, I wish you could click on sides".</p> <p>He went back to the cube and stretched it in z, and then he capped it on one of the long rectangular sides. Then he capped it on the other rectangular side. He was now stuck because the rotation tools did not yet work. He went back to the prism. "Can I use two shapes?"</p> <p>A: Those were all done starting with one shape.</p> <p>He loaded the truncated octahedron again.</p>	<p><i>Jesse repeatedly returned to shapes like rectangular prisms and tried to cap them to get to the stretched cuboctahedron.</i></p>
 21:47  24:52	<p>"It won't do it!"</p> <p>"Is there an Archimedean Solids I on here?" He brings up the window. "You used all these same shapes?" [A: Yup]. He reloaded the truncated octahedron and moves the wireframe image a bit. "Do you like making people suffer?" He continued to twirl the wireframe. He twirled it faster, probably out of frustration.</p> <p>He reloaded the truncated octahedron. He started getting cranky: "I see where I'm supposed to do it, but it won't let me do it!" This time, he gets the slice. "I knew it, I knew it, I knew it!" "It wouldn't let me ... my dots weren't in a straight line, and I thought they would have to be."</p>	<p><i>Jesse now successfully sliced the truncated octahedron and got the right shape for the cross-section.</i></p> <p><i>This was a bug in his understanding of defining a plane to slice through a shape ...</i></p>

Table 6-1 (c). Annotated transcript, part 3.

	<p>26:40 He loaded the square prism back in, and again tried slicing it. He tried each of the function buttons in turn, and then decided to stretch the shape in z. He experimented with maps to shrink and stretch the shape in z, and then stretched it even further in z.</p> <p>29:46 "Oh! I pushed it up 1.9 instead of 0.9" He applied a map to shrink the shape back down: "Zjooop!" (He made this noise as the shortened shape appeared).</p> <p>"You sure you did this on this computer?" This time he tried slicing the prism, sliced a triangular piece, and then he added a cap to the remaining slice.</p> <p>32:46 "Can you tell me the group of shapes you started with?"</p> <p>A: Look at every group of shapes again.</p> <p>"There's two groups of shapes that will work. Prisms and Antiprism and Platonic Solids. Archimedean 1 and 2 won't work." Then he loads a truncated tetrahedron. He muttered under his breath about having to "call in reserves."</p> <p>35:59 He loaded different prisms, and then he worked on the Archimedean 2 set of shapes. "Can you just tell me which shape to start with?"</p> <p>A: Give it five more minutes.</p> <p>"I looked at every single one of the shapes." "Two options ... There are two options -- prisms and antiprisms ... a cube in prismatic solids."</p> <p>A: Look at Archimedean 1.</p> <p>"I thought about one of those -- it's not square. I can try." He loads in the cuboctahedron. "Oh!!"</p> <p>"Why didn't you say that!" He applied a stretch map in z and hid his face. "Yay! I tried that one earlier, but I couldn't get it to work!". [What made you think it wouldn't work?] "Just the way that it looks. When it's here (wireframe) you can see all of it - you can tell that it works, but when it's here (the palette) it doesn't". He was going to cap each side of the cube as an alternative.</p>	<p><i>Jesse returned to the strategy of trying to slice a rectangular shape to get the stretched cuboctahedron.</i></p> <p><i>He was increasingly frustrated at this point.</i></p> <p><i>He finally got the last shape, with much celebration. He explained that he had seen the cuboctahedron starting shape in the palette, but that it did not look like it would work, but once the shape was loaded, the wireframe object helped him to see that it would work.</i></p>
---	--	---

Jesse was able to reproduce the sliced cube, stretched-capped antiprism, and the capped triangular prism with little difficulty. He demonstrated understanding of the slicing, stretching, and capping concepts and was able to use the software to execute the functions. He was still unsure about the x, y, and z-directions and he stretched objects by trial and error along different axes until he arrived at the correct result, but he clearly remembered and understood the concept of applying an inverse map to undo a stretched shape.

He had slightly more trouble reproducing the sliced-truncated octahedron. He immediately knew which starting shape to use and loaded the truncated octahedron, but it took him several attempts to arrive at the correct cross-section in his slice.

The shape which gave Jesse the most difficulty in this activity was the stretched cuboctahedron, formed by applying a relatively simple map to one of the starting shapes. He had two different (incorrect) strategies -- his first plan was to stretch a cube and then cap it on the rectangular sides, to get a shape like the one shown in Figure 6-15 (a); his second approach was to make multiple slices on a prism to create triangular faces similar to the one shown in Figure 6-15 (b).

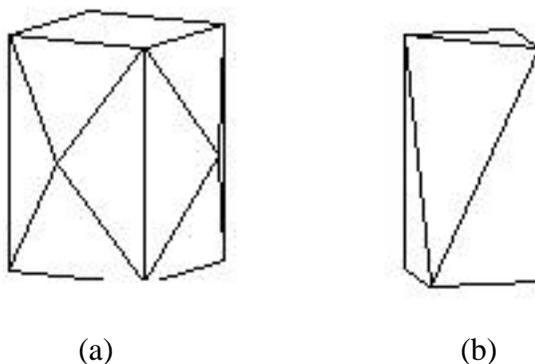


Figure 6-15. Jesse's strategies for creating the stretched cuboctahedron.

The key moment in solving this shape problem came for Jesse when he found the correct starting shape. He said that he had looked at the (correct) cuboctahedron

starting shape in the shape palette earlier in the session, but did not load it because it did not look like the right thing. However, once he did try it, he could tell by twirling the wireframe image that it would work. This is an interesting instance where the wireframe representation was instrumental in adding to Jesse's understanding of the shape.

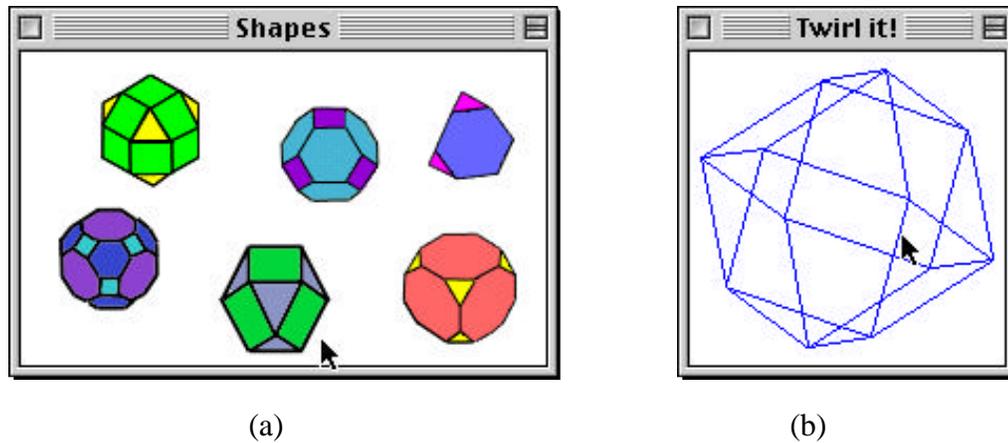


Figure 6-16. (a). The cuboctahedron as it appears in the Shapes palette; (b) the cuboctahedron in the wireframe window.

## 6.7 Discussion

This case study was presented to provide an idea of the type of interaction that took place between the children and the author, as both educational software designer and cognitive scientist.

In terms of software design, Jesse's work with the system pointed to a few much-needed features to be implemented in a future version: (1) an undo function for solid operations, (2) an undo function for net decorations, and (3) some sort of online help or guide. The undo features for net decoration and solid operation are a straightforward implementation task, and should be incorporated into the existing interface without much difficulty. It is less clear what form of online help should be integrated -- more investigation is needed to determine whether it should take the form

of an animated character as suggested by Jesse, or perhaps a more subtle text and graphics-based interface, or a combination of the two.

From the point of view of spatial cognition, Jesse readily grasped the ideas of the shape algebra -- building shapes from other shapes -- as well as transformations such as adding a cap, stretching, and shrinking shapes. He still occasionally confused x, y, and z-directions -- this can easily be remedied by building coordinate axes tools into the Polyhedron interface. He had more difficulty with slicing and determining the points to define the slicing plane. This again points to tools that might be built -- perhaps an outline of a slicing plane that connects the points selected -- in later versions. The experience in general had a marked effect on Jesse's thinking about three-dimensional shapes -- in the pseudonym scheme of the rest of the thesis, he is also known as student ADH, whose pre- and post- shape interview transcript is shown in Table 5-9.

Finally, Jesse's case study points to some of the less tangible themes discussed earlier: his activities alternated seamlessly between computer and real-world construction tasks; he personalized his mathematical experience by customizing a basic set of polyhedra to fit his sculpture needs; and he created a mathematical object to give to his mother as a gift.