PAMELA FRORER, ORIT HAZZAN and MICHELLE MANES

REVEALING THE FACES OF ABSTRACTION

We adapted this paper from a presentation at the Research Conference in Collegiate Mathematics Education (RCCME), held at Central Michigan University in September, 1996. The presentation reflected our project's discussions over several months. We began with the goal of beginning to define abstraction. The result was not a definition, but a dialogue (condensed here from a cast of ten to two characters) about the complexity and the various faces of abstraction. Participants in the discussions included: Al Cuoco, Ed Dubinsky, Pamela Frorer, E. Paul Goldenberg, Wayne Harvey, Orit Hazzan, Uri Leron, Michelle Manes, Tammy Jo Ruter, and Despina Stylianou.

We chose a new technique for presentation: a dialogue. We experimented with it because we wanted to present our own thinking mixed in with others' thoughts and written statements. Retaining the conversational format from which the ideas first emerged preserved some of the differing perspectives.

Our intention here is not to survey all literature and opinions expressed about abstraction, nor is the conversation only about abstraction in mathematics education. A good source for a variety of discussions about abstraction is Noss and Hoyles (1996). There are, however, a few places where we felt a specific reference was needed and thus the reference is made explicit.

THE SETTING

A conversation between two mathematics instructors at a conference. Both instructors graduated from the same university. Allie is working with colleagues on a paper about abstraction to be published in the *Monthly*. Bert is now working in computer science.

Bert: Well I'm glad we have some time to talk before the next conference session. I've been wanting to tell you that I read the initial draft of your paper on abstraction and enjoyed it.

Allie: You did? Great. Speaking of abstraction, what do you think of the talk scheduled for the next session this morning? I don't remember

International Journal of Computers for Mathematical Learning **2:** 217–228, 1997.

who the presenters are . . . but, I wonder what they mean by the title: "Revealing the Faces of Abstraction"?

Bert: I heard they plan to present different perspectives on abstraction as it applies to mathematics, and to learning mathematics. What I'm still wondering is exactly what is meant by the word "abstraction."

Allie: I think a lot of us are wondering how to reach a shared understanding of the term, or at least a definition we can use in common – there seem to be a number of different interpretations out there. My paper is an attempt to provide some common ground. So tell me: what comes to mind when you think of abstraction?

Bert: Well let's see, I can at least locate abstractions in mathematics. And I've solved a lot of problems I would consider very abstract. I've been relatively successful with abstraction; yet sometimes I do question whether I've always known what was going on, or what I thought I was doing in solving a particularly abstract problem. But for a lot of people, and certainly for our students, abstraction in a mathematical sense may very well be just that, an abstraction!

Allie: That's true . . . And because I've been involved in writing that paper on abstraction, I've had quite a few discussions with my students about just what is abstract, or what is an abstraction in mathematics. In the process, my students have come up with this list of phrases and words they associated with the term as it applies to mathematics:

> hidden, complex, requiring deep thought, not concrete, apart from actual substance or experience, not easily understood, a mental construction, a theoretical consideration . . .

But I agree with you Bert: Most people, or most of my students anyway, view all of mathematics beyond arithmetic and basic algebra as an abstraction, a mental construction.

Bert: Your list is helpful, and thought provoking . . . I wonder if, with regard to mathematics, "abstraction" means a way of thinking, or perhaps a tool to classify mental constructs or ways of thinking.

Allie: Absolutely. I think an abstraction can be a way of thinking, or a process; I'd even argue that an abstraction can be an object as well. Actually – have you read any of the chapters in *Advanced Mathematical Thinking*, edited by Tall (1991)? – I'd go further to say that abstractions can be actions, processes, objects, and schemas; although you might call some of the objects the byproducts of the process of abstraction! But nonetheless they become things, objects that we talk about and work with, such as: **R**n, a universal coordinatizing domain; or polynomials, which provide a universal calculation domain for all kinds of extensions of **Q**.

Bert: Yes, I have looked at that book – several chapters were referenced in a critique by Confrey and Costa (1996) published in the *International Journal of Computers for Mathematical Learning*. Confrey and Costa's piece made some interesting claims about the ideas of a few of the authors in Tall's book and other theorists, all of whom they have grouped together as "reification theorists." One of the claims was that these theorists see "mathematics as strictly hierarchically ordered," and further that they believe "the history of mathematics is a metaphor for straightforward, purifying progress." I haven't read all of the literature on the subject, but that certainly doesn't describe my impression of the perspectives in Tall's book.

Allie: Right, I saw that article, and Tall's response (1997) in the next issue of the journal where he tries to clarify some of his intentions and theory. Confrey and Costa claim a clear contrast between the approach they advocate, and the approach they say the reification theorists take. But Tall points out that only two of the chapters produced by the group referred to are considered "reificationist;" and he clarifies some of the differences between at least how he and Dubinsky (1988, 1991) see various forms of mathematical objects. Given the range of views among the theorists involved, it seems to me that Tall is on to something when he says he has more in common with Confrey and Costa than not.

Besides, selecting the language of tools, as Confrey and Costa suggest, doesn't necessarily obviate the language of objects, at least at the elementary level, which is what Confrey and Costa seem most concerned with in the article. For example, I don't believe reification theorists would agree with their statement that, according to reification theory, fractions must be introduced earlier than ratio and proportion, which "introduce harder symbolic and conceptual demands." (p. 162) To the contrary, children of third grade may more readily understand that one-half of a pizza is the same as three-sixths or six-twelfths of a pizza, than that one-half plus one-third is five-sixths. What becomes reified here is not the addition or multiplication involved in calculating an answer, but the fractions themselves. In order to add fractions of unlike denominators such as one-half plus one-third, children must think of fractions as objects (as the numbers on a number line), and move beyond their thinking of fractions as "some *part of* something," or as the action or process of taking one-half or one-third *of* something.

In any case I hope the paper pushes the conversation along, for more discussion needs to take place among the community of researchers, theorists, and practitioners. But you and I were talking about abstraction . . .

1. IGNORING THE DETAILS

Bert: Yes, well it's all related. You were saying that abstraction can be a way of thinking, an action, a process, or an object. But if we want to define it further, why don't we start with the components of abstraction and build from the bottom up? If I recall what I read in your paper, one interpretation of abstraction is the process of identifying (I mean, making identical) two different things by choosing to ignore *some* of their properties while emphasizing some others. In mathematics for example: similarity or congruence of triangles, congruence of numbers mod n, and equivalence relations in general. We do this outside of mathematics when we call a lot of fairly different creatures "dogs."

Allie: Right: the process of identifying, or making equivalent through classification . . . You know, I can still remember when for the first time I had to understand an object by finding a function that maps it to another object that I knew before, (essentially, do a homeomorphism), and then accept that as a satisfactory description of the new object. At the beginning that was a difficult notion for me, and I had to struggle with it for a while. Yet if someone had likened that process to talking about some new friend of hers that "looks just like our mutual friend Julie," I'd have had no problem understanding it.

Bert: Sure. Or in my field I'd say that when I'm presented with an idea or a problem situation I don't understand very well, I can learn about its essential properties by studying a related idea or situation . . . reducing the problem to one I'm more familiar with.

Allie: That's an extremely useful habit of mind in and out of mathematics, but one that hardly ever gets discussed in curricula. It connects with what you were saying about reducing a problem to one you're more familiar with. In mathematics, for example, when you have an unknown structure of a certain type, one way to study it is to set up a structure-preserving map between it and a better understood object of the same type, and then estimate how far off from a 1-1 correspondence the map is.

Bert: Actually, this is the way of thinking expected of young children when they're asked to count things by matching, right? The structure preserving maps are just maps between sets. It's also one of the central techniques of algebra, where there are universal objects that can be used to study all kinds of algebraic structures via structure preserving maps.

Allie: In my paper, I call this kind of abstraction "Ignoring the Details," and I distinguish between two different uses. The first one is: when you don't know enough to distinguish among the individual elements of a class. For example, walking into a new crowd of people. At first, they are just undifferentiated faces. Later, when you've had time to observe, they *are* differentiated (cf. Papert, 1980, p. 137).

The second one is what we were just talking about: when you *do* know a lot and you want to suppress the details to get a better idea of what's going on. This might happen when you know the individuals in a group very well, and you start classifying people by important (or unimportant!) attributes.

Bert: That's a good distinction, but I'm wondering about a third possibility. It might be a mix of the two you just described, let's see. When I write a paper, I might begin with the outline – the abstraction – and then fill in the details. If it's a paper about a subject that I understand well, then the function of the outline-first approach is to suppress the details for the purpose of organization. This is a writing analog of successive refinement (or top-down programming) in Computer Science – when you first roughly outline the main program, ignoring the details and giving procedure names without worrying yet about how they will do their jobs.

But the third possibility that I'm suggesting as a kind of "Ignoring the Details" meaning for abstraction is like what's called bottom-up programming. It's the Artificial Intelligence notion of tool-building, where the elements that are being built are, in one sense, very much the details but they're also, in another sense, the most abstract elements. For example, I may be writing a paper about a subject I'm hoping to come to understand *during* the writing process. In one sense, all I can write are the details – the organization may come much later.

Allie: I like that. It's an interplay between the outline and the details of a paper . . . or between the structure of a program and its pieces, or tools. But I think that even this interpretation of the "ignoring the details" abstraction is captured by the definition of abstraction that I quote in my paper. If I remember, abstraction is defined by *Webster's Third* as:

> The act or process of leaving out of consideration one or more qualities of a complex object so as to attend to others (p. 8).
> [from *Webster's Third New International Dictionary* (1966)]

Bert: You know, one of the big things now in computer science is "object-oriented programming." It's like the "top down" approach in some ways, but uses what are called abstract objects even more explicitly. Here, a well-designed object should be a member of a class with well-defined attributes and behavior. By combining both state and behavior in a single unit, an object becomes more than either alone. If the objects are carefully encapsulated, they can be successfully used and reused without the programmer knowing about the internals.

Allie: This is what enables them to ignore the details . . .

Bert: Yes, because in programming, if you continually have to tend to the business of keeping the right data matched with the right procedure, you're forced at all times to be aware of the entire program at a low level of implementation. By providing another, higher level of abstraction, object-oriented programming languages give you a larger vocabulary and a richer model to program in.

Allie: I was noticing that while the use of the concept of abstraction is different, some of the language you used in talking about object-oriented programming is very similar to the language of the action-process-object-schema theorists. They talk about "encapsulating objects" too. Did one of the fields have influence on the other?

Bert: I hadn't thought about that connection. I know that Dubinsky was involved with computer science in the 1980's. I wonder if he's among those responsible for some cross-use of terms like that . . .

But I was thinking about something: It's interesting that in computer science, abstraction as the act of hiding the implementation details is an explicit part of the curriculum. It's used for the organization and control of complexity in programs, and is taught even in the elementary computer science courses. Abelson and Sussman (1985) is a standard text, and its first two chapters have "Building Abstractions" in their titles.

Allie: And yet while abstraction in mathematics has some additional qualities or meaning, we rarely find it explicitly discussed let alone defined. You can pick up a book entitled *Abstract Algebra* and not find a real discussion of abstraction as a process, or of abstractions as objects, or . . .

2. RELATIONSHIPS BETWEEN THE PERSON AND THE OBJECT ABOUT WHICH S/HE THINKS

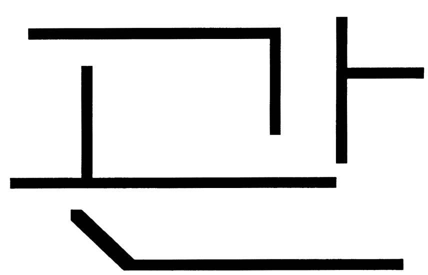
Bert: . . . That's true; but notice that some of those same books, covering the same material, are titled *Elementary Algebra*. The difference between the titles is reminding me of another way of thinking about it. Have you read the paper written by Wilensky (1991): "Abstract meditations on the concrete . . .

Allie: . . . and concrete implications for mathematics education;" ? Of course. Why?

Bert: Well, while you examine the abstract vs. the detailed, Wilensky looks at the abstract vs. the concrete. Wilensky's point is that whether something is abstract or concrete (or anything on the continuum between those two poles), is not an inherent property of the thing, but rather a *relationship between the thing and a person* (or a community). Thus "number" is concrete for me, but not for a 3-year old child in a Piagetian experiment.

Allie: That's a nice example, and a good point: what is abstract for one person can be quite concrete for another, and furthermore that what's abstract for someone at one *time* may be quite concrete at a later time. The notion of "number" is still a mental construction, but it takes on such a familiarity and ease of use that it seems concrete to you and me.

Bert: Here's an example outside of mathematics: Would you say that this picture of a Korean word is abstract?



Allie: Yes!

Bert: More so than a Spanish word you don't know the meaning of, like, "depurar"?

Allie: Yes, definitely, but then I'm familiar with the symbols strung together there – they're all letters from the Latin alphabet.

Bert: That's part of my point; most linguists would say that all languages are equally abstract. The phonemes k-a-t have no relation to the animal we call by that name. The cat could have been named anything and it would be the same animal. Even sign languages, which seem more iconic or pictographic, are abstract until the learner has made a connection between the sign and the action, object, or idea being represented. It's only as we learn and study languages that they begin to *seem* less abstract to us.

Allie: In short, what seems abstract to you may not seem to be for me. And we could extend this to make the claim that for a particular community, and for a given problem, there's likewise an optimal level of abstraction to work with. One of the clearest examples of this lies within the student/teacher communities. Good educators continually seek the boundary between working too abstractly or too concretely for their students' best learning.

3. ABSTRACTION AND PROPERTIES

Bert: I agree; but we're still missing something important with regard to abstraction. As you said before, it's about leaving out of consideration certain details. Because abstraction is also about structure, *which* details to leave out is an important consideration in itself. To me this means it's about choosing to look at recognizable parts and then analyzing those parts and their properties.

Allie: OK, this seems related to what we were saying about bottom-up programming, but now you're adding that abstraction means thinking of structures in terms of their properties rather than the actual components (the objects and operations) that make them up?

Bert: Well, let's look at properties of mathematical concepts. I have an example: consider the set of binary operations. The elements of this set can have various properties: commutative, associative . . .

Allie: But why do you have the need to describe properties or to assign properties to an element?

Bert: Well, "positive" isn't an important property of numbers unless there are negative numbers around. "Rational" wasn't an interesting property until it was discovered that some numbers did not have that property. (I'm glossing over some history here, but you know what I'm getting at.) Properties are invented when there's more than one element in a set; they bind together elements defining the set or subset.

Allie: I don't agree. Sometimes properties come about just because they're useful to move you ahead in a proof. You're not necessarily comparing an object to others, you just need it to behave in a certain way to move forward. So sometimes you try to find out if it does behave that way – if it does have some property or other. Other times you just *assume* it does in order to move forward, and then you come back later to try to verify that assumption, or remove the condition, or do something to fix up the proof.

Bert: I don't understand. What do you mean you assume it behaves some way?

Allie: Think about early proofs of the Fermat conjecture. They assumed that certain number fields had unique factorization. After the proof was done, they found out that the fields *didn't* have the property, so the proof didn't work. But the "property" of unique factorization didn't come about because we suddenly found fields that didn't have it. It came about because if the fields *did* have it, we could prove Fermat.

Bert: But when you say: "If these number fields have unique factorization, then I can move forward," there you are, conceiving of the possibility of number fields which don't have unique factorization. You have divided the world into two kinds of things: things that have this property and things that don't. It's possible (and maybe even hoped for) that the second set is empty, but . . .

Allie: No, I don't think so. I don't picture things that way. I just say to myself: "It needs to work like this." Then I make the assumption I need to make. Maybe, when I go back later to iron things out, maybe then I'll ask: "So, what kinds of things do and don't have this property?" But earlier, when I'm involved in doing the proof or investigation, I'm focused on the problem at hand.

Bert: Perhaps. But my point still holds that after the properties are defined, for *whatever* reason, they are interesting only when there is more than one element in the set. At that point you go back and say, “what kinds of things have this property and what don’t?” If the answer is “everything does and nothing doesn’t,” then it’s part of the thing and not a property. It’s only in the binding of things together, and separate from other things, that we can talk about . . .

Allie: Wait a minute, you’re saying that we need to talk about properties of objects (hence use an abstract approach) only if there are more than one in the family, but this isn’t quite the case in mathematics. We can find lots of examples of an abstract definition of a *single* object. In fact, mathematicians find great joy in defining something abstractly (by axioms or postulates) and then proving that there’s a unique object satisfying the definition. For example, we have the rational numbers (the smallest field containing the integers), or the real numbers (a complete ordered field containing the rationals), or the complex numbers (a minimal field containing the reals and the square root of -1), and there’s lots more . . .

Bert: But still, in the process of such “definition-and-proof-of-uniqueness,” we are in fact conducting a thought experiment in which we assume the existence of more than one object and then show that they must all be equal. For example, to prove uniqueness of the identity element of a group – which is defined abstractly by its properties – you assume there are two such elements, and show they must be equal. By the way, in all such cases (except when the new object is defined as a member of a previously constructed set), the uniqueness only goes up to isomorphism – *another* issue of abstraction.

EPILOGUE

But look, the presentation started 10 minutes ago, why don’t we get going?

Allie: OK, but you know this is very interesting. We’ve had this discussion about abstraction, and three themes have emerged: a (helpful) ignoring of details; the relationship between the person and the object of thought; and abstraction and properties. I think this is what abstraction is about. And in a way, our discussion itself was abstract. I wonder if we can describe it in terms of these themes? I mean, along the way *we ignored a lot of pesky details in our examples* . . .

Bert: And *the discussion has relied on our relationships to the topics being discussed*, to the concept of abstraction as well as to the specific examples from mathematics, from linguistics, and from computer science . . .

Allie: The reaction from anyone overhearing our conversation would certainly depend on their relationship to the things we've talked about. But what about the last one: abstraction and properties?

Bert: Well, we've just done that, haven't we? *We listed and examined at least three properties of the concept of abstraction relating to our discussion*, and . . .

Allie: Of course! It all comes together.

ACKNOWLEDGEMENT

This work was supported by the National Science Foundation, grant DUE 9450731. It is part of the *Gateways to Advanced Mathematical Thinking* (GAMT) project at Education Development Center, Inc. directed by Al Cuoco and Wayne Harvey. The authors of this paper are listed alphabetically. We gratefully acknowledge the contributions made by other members of the GAMT project: Al Cuoco, Ed Dubinsky, Wayne Harvey, and Paul Goldenberg. Special thanks are due to staff members Tammy Jo Ruter and Despina Stylianou who contributed greatly to early drafts and organization of the paper. The opinions expressed here are those of GAMT project members and are not necessarily shared by the NSF.

REFERENCES

Abelson, Sussman and Sussman (1985). *Structure and Interpretation of Computer Programs*. Cambridge, Mass: MIT Press.

Confrey, J. and Costa, S. (1996). A critique of the selection of "mathematical objects" as central metaphor for advanced mathematical thinking, *The International Journal of Computers for Mathematical Learning* 1(2): 139–168. Kluwer Academic Publishers.

Dubinsky, Elterman and Gong (1988). The student's construction of quantification, *For the Learning of Mathematics* 8(2): 44–51. Montreal, Quebec: FLM Publishing Association.

Dubinsky, E. (1991). Reflective abstraction in advanced mathematical thinking, *Advanced Mathematical Thinking*. Kluwer Academic Publishers.

Lakatos, I. (1976). *Proofs and Refutations*. London: Cambridge University Press.

Leron, U. (1987). Abstraction barriers in mathematics and computer science. In Hillel, J. (Ed.), *Proceedings of the Third International Conference for Logo and Mathematics Education*.

Noss, R. and Hoyles, C. (1996). *Windows on Mathematical Meanings – Learning Cultures and Computers*, Kluwer Academic Publishers.

Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.

Tall, D. (Ed.) (1991). *Advanced Mathematical Thinking*. Kluwer Academic Publishers.

Tall, D. (1997) (in press), Critique of a critique, *The International Journal of Computers for Mathematical Learning* 2(1). Kluwer Academic Publishers.

Webster's Third New International Dictionary (1966).

Wilensky, U. (1991). Abstract meditations on the concrete and concrete implications for mathematical education. In I. Harel and S. Papert (Eds.), *Constructionism* (pp. 193–203). Norwood, NJ: Ablex Publishing Corporation.

Education Development Center
Newton, MA, US
E-mail: pamelaf@edc.org, michelle@edc.org

Technion – Israel Institute of Technology
Department of Education in Science and Technology, Israel
E-mail: oritha@techunix.technion.ac.il