

A Framework for Research and Curriculum Development in Undergraduate Mathematics Education

Mark Asiala
Purdue University
W. Lafayette, Indiana

Anne Brown
Indiana University South Bend
South Bend, Indiana

David J. DeVries
Georgia College
Milledgeville, Georgia

Ed Dubinsky
Purdue University
W. Lafayette, Indiana

David Mathews
Central Michigan University
Mt. Pleasant, Michigan

Karen Thomas
University of Wisconsin-Platteville
Platteville, Wisconsin

©November 4, 1997

Abstract

Over the past several years, a community of researchers has been using and refining a particular framework for research and curriculum development in undergraduate mathematics education. The purpose of this paper is to share the results of this work with the mathematics education community at large by describing the current version of the framework and giving some examples of its application.

Our framework utilizes qualitative methods for research and is based on a very specific theoretical perspective that is being developed through attempts to understand the ideas of Piaget concerning reflective abstraction and reconstruct them in the context of college level mathematics. Our approach has three components. It begins with an initial theoretical analysis of what it means to understand a concept and how that understanding can be constructed by the learner. This leads to the design of an instructional treatment that focuses directly on trying to get students to make the constructions called for by the analysis. Implementation of instruction leads to the gathering of data, which is then analyzed in the context of the theoretical perspective. The researchers cycle through the three components and refine both the theory and the instructional treatments as needed.

In this report the authors present detailed descriptions of each of these components. In our discussion of theoretical analyses, we describe certain mental constructions for learning mathematics, including actions, processes, objects, and schemas, and the relationships among these constructions. Under instructional treatment, we describe the components of the ACE teaching cycle (activities, class discussion, and exercises), cooperative learning and the use of a mathematical programming language. Finally, we describe the methodology used in data collection and analysis. The paper concludes with a discussion of issues raised in the use of this framework, followed by an extensive bibliography.

1 Introduction

The purpose of this paper is to set down a very specific methodology for research in the learning of mathematics, and curriculum development based on that research. We are concerned with theoretical analyses which model mathematical understanding, instruction based on the results of these analyses, and empirical data, both quantitative and qualitative, that can be used to refine the theoretical perspective and assess the effects of the instruction.

This report caps several years of research and development during which a framework for conducting this work has been developed and applied to various topics in collegiate mathematics. Now that the main outlines of our approach have stabilized, and the group of researchers using this framework has begun to grow, we wish to share our work with others in the mathematics education community, including potential collaborators.¹ We offer here the details of our approach, both as a report on what has taken place and as a possible guide to future efforts.

After a few general remarks in the introduction about paradigms, and about previous discussions of the specific framework we have been developing, we proceed to an overview of our approach as it now stands, and the goals we associate with it. Then, in the main portion of the paper, we describe in detail our framework and its components along with several examples. Next, we consider some larger issues that arise in connection with our framework. Finally, we summarize what has been said in this paper and point to what can be expected in future reports.

1.1 What is a paradigm and why is one needed?

The seminal work of Thomas Kuhn [19] teaches us that scientific research proceeds according to what he calls *paradigms*.² *A paradigm is a collection of understandings (explicit or implicit) on the part of an individual or group of individuals about the kinds of things one does when conducting research in a particular field, the types of questions that are to be asked, the sorts of answers that are to be expected, and the methods that are to be employed in searching for these answers.*

We also learn from Kuhn that “paradigm shifts” do not come quickly or easily, but they do tend to be sharp. They are caused by what Kuhn calls a “crisis state” which can be the result of one of two different situations. One is that an event or discovery is so far reaching that it is impossible to assimilate it into the current paradigm, hence the need for a new paradigm. The second situation is a developing dissatisfaction with the current paradigm that reaches a level where answers to certain questions (often basic to the field) can not be easily or satisfactorily obtained. This brings about a sort of declared rejection of the current paradigm by individuals who begin to search for a new paradigm. Whatever the cause, the crisis state continues until an alternative can be agreed upon by the community and then the change *must* be a simultaneous rejection of the old paradigm and an acceptance of the new paradigm by the research community as a whole. Thus, dissatisfaction with a particular paradigm builds up gradually over a long period of time until there occurs a moment, or “scientific revolution” in which the understandings change rather drastically and fairly quickly.

It seems that the second situation which causes Kuhn’s scientific revolution has occurred for research in mathematics education. For a long time, research in this field consisted almost exclusively of statistical comparisons of control and experimental populations according to designs proposed by Sir R. A. Fisher some 60 years ago for the purpose of making decisions about agricultural activities [12]. In the last decade or so, there has been a growing concern with the impossibility of really meeting the conditions required to make application of statistical tests to mathematics education valid, a dissatisfaction with the small differences and unrealistic contexts to which these designs seem to lead, and a developing understanding that the fundamental mechanisms of learning mathematics are not as simplifiable and controllable as agricultural factors. Traditional statistical measures may apply, for example to paired-associate learning, but if one wishes to build on the work of Jean Piaget, and/or use the ideas of theoretical cognitive structures, then new methods of research, mainly qualitative, must be developed to relate those structures to observable behavior. (For a discussion of the implications of cognitive science on research methodology in education

¹We have organized an informal community of researchers who use this framework, known as *Research in Undergraduate Mathematics Education Community* or *RUMEC*, with the intention of producing a series of research reports by various subsets of the membership. The present paper is the first in the *RUMEC* series.

²For a critical look at Kuhn’s and others’ philosophical views on the structure of scientific revolutions and theories see Suppe [30] or Gutting [16].

see Davis [4].) Workers in the field have stopped insisting on a statistical paradigm and have begun to think about alternatives.

This represents only the conditions for a scientific revolution and is not yet a paradigm shift because no single alternative point of view has been adopted to replace the accepted paradigm. The ideas of Jean Piaget have influenced many researchers to turn from quantitative to qualitative methods, but there are many forms of qualitative research that are being used at present. In considering the variety of approaches being used, there are two aspects which must be addressed. The first is the theoretical perspective taken by the researchers using a particular approach, and the second is the set of actual methods by which data is collected and analyzed.

Patton [20] lists some of the theoretical perspectives used by qualitative researchers. These include the ethnographic perspective, in which the central goal is to describe the culture of a group of people, and related perspectives including phenomenology and heuristics, in which the goals are to describe the essential features of a particular experience for a particular person or group of people. The tradition of ecological psychology seeks to understand the effect of the setting on the ways in which people behave, and the perspective of systems theory seeks to describe how a particular system (for instance, a teacher and a group of students in a given classroom) functions. In addition, there are orientational approaches (feminist, Marxist, Freudian, etc.) within each of these perspectives.

Methods of data collection for qualitative studies vary widely, and in many of the theoretical perspectives it is considered important to use a wide variety of data collection methods, studying the phenomena of interest from all available angles in order to be able to triangulate data from many sources in reaching a conclusion. Romberg [24] discusses the use of interviews, which may range from informal discussions between researchers and participants to very structured conversations in which a predetermined list of questions is asked of each participant. He also discusses observational methods ranging from videotaping to the use of trained observers to participant observations. Patton [20] discusses other sources of data, including documents and files which may be available to the researcher, photographs and diagrams of the setting in which the research takes place, and the researchers' field notes.³

Thus we see that there is a wide variety of frameworks in which it is possible to work. In this paper we describe our choice which has been made consciously with concern for the theoretical and empirical aspects as well as applicability to real classrooms in the form of instructional treatments.

According to Kuhn, researchers shift to a new framework because it satisfies the needs of the times more than the existing paradigm. There are two reasons why we feel that researchers should make conscious choices about the framework under which they work.

One reason is necessity. The variety of qualitative research methodologies indicated above does not appear to be leading to any kind of convergence to a single approach (or even a small number of approaches) that has general acceptance. We are finding more and more that research done according to one framework is evaluated according to another and this is leading to some measure of confusion. Therefore, we feel that researchers should make explicit the framework they are using and the basis on which their work is to be judged. Consumers of the results of research need to have a clear idea of what they can and cannot expect to get out of a piece of research.

Second, we feel that a conscious attention to the specifics of one's framework is more in keeping with the scientific method as expressed by David Grieser who interpreted science as "a department of practical work which depends on the knowledge and conscious application of principle" [15]. Grieser decided 20 years ago that it was time to move computer programming to an endeavor in which it was possible to teach the principles so that they can be consciously applied. We believe that it is important today that those who study the learning of post-secondary mathematics attempt to make available to others the methodologies under which they work.

1.2 Previous discussions of this framework

Components of our framework have been discussed in several papers over the last several years: [2], [5], [7], [8]. The overall framework with its three components have been discussed at length, especially the theoretical component, but only very fragmented discussions of the other two components, instructional treatments and

³For information on qualitative methods used in the social sciences see Jacob [17], or Patton [20]. For information on qualitative methods used in mathematics education research, see Romberg [24] or Schoenfeld [26]

gathering/analyzing data have been given. Moreover, the framework and its components are continuing to evolve as we reflect on our practice. Finally, the authors of this article are part of a larger community (see p. 2) which is in the process of producing a number of studies of topics in calculus and abstract algebra using this framework.

Therefore, it seems reasonable at this time to present a complete, self-contained and up-to-date discussion of the entire framework.

2 A framework for research and curriculum development

2.1 Overview of the framework

The framework used in this research consists of three components. Figure 1 illustrates each of these components and the relationships among them. A study of the cognitive growth of an individual trying to learn a particular mathematical concept takes place by successive refinements as the investigator repeatedly cycles through the component activities of Figure 1.

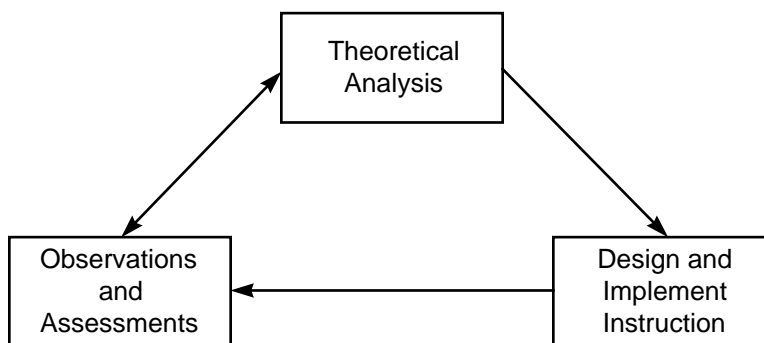


Figure 1: The Framework

Research begins with a theoretical analysis modeling the epistemology of the concept in question: what it means to understand the concept and how that understanding can be constructed by a learner. This initial analysis, marking the researchers' entry into the cycle of components of the framework, is based primarily on the researchers' understanding of the concept in question and on their experiences as learners and teachers of the concept. The analysis informs the design of instruction. Implementing the instruction provides an opportunity for gathering data and for reconsidering the initial theoretical analysis with respect to this data. The result may well be a revision of the theoretical analysis which then lays the foundation for the next iteration of the study. This next iteration begins with the revised theoretical analysis and ends with a further revision or deeper understanding of the epistemology of the concept in question which may become the foundation for yet another repetition of the cycle. These repetitions are continued for as long as appears to be necessary to achieve stability in the researchers' understanding of the epistemology of the concept.

2.2 Goals and issues associated with the framework

Research using this framework is inevitably a synthesis of "pure" and "applied" research. Each time the researchers cycle through the components of the framework, every component is reconsidered and, possibly, revised. In this sense the research builds on and is dependent upon previous implementations of the framework. We observe students trying to understand mathematics and offer explanations of successes and failures in terms of mental constructs and the ways in which they transform. Our specific goals are: to increase our understanding of how learning mathematics can take place, to develop a theory-based pedagogy for use in undergraduate mathematics instruction, and to develop a base of information and assessment techniques which shed light on the epistemology and pedagogy associated with particular concepts. The goals are thus associated with the three components of the framework.

There are many issues raised as a result of the use of this framework. In the component of theoretical analysis there are the following issues: 1. How does one go about developing the theoretical perspective? 2. How do we see the relationship between this theory and what actually happens; that is, to what extent can a theoretical analysis provide an accurate or even approximate picture of what is going on in the minds of the learners? In the component dealing with pedagogy there is the issue of explaining the relationship between the instructional treatments and our theoretical analysis. With respect to data analysis there are the following issues: 1. To what extent do our theoretical ideas work? 2. How much mathematics is being learned by the students? 3. What would it take to falsify specific conjectures or our theory in general? 4. Since data can come from this study but also from assessment of student learning which may not be part of this study, what is the appropriate use of these in drawing conclusions?

We will return to these issues later in this paper in Section 4, p. 20, but in order to do so, it is important to develop more fully the description of the three components of the framework and their interconnections (as illustrated by the arrows in Figure 1.)

3 The components of the framework

3.1 Theoretical analysis

The purpose of the theoretical analysis of a concept is to propose a model of cognition: that is, a description of specific mental constructions that a learner might make in order to develop her or his understanding of the concept. We will refer to the result of this analysis as a *genetic decomposition* of the concept. That is, a genetic decomposition of a concept is a structured set of mental constructs which might describe how the concept can develop in the mind of an individual.

The analysis is initially made by applying a general theory of learning and is greatly influenced by the researchers' own understanding of the concept and previous experience in learning and teaching it. In subsequent iterations through the framework, the analysis of data increasingly contributes to the evolving genetic decomposition.

In working with this framework we make use of a very specific theoretical perspective on learning which has developed through our attempt to understand the ideas of Piaget concerning reflective abstraction and to reconstruct these ideas in the context of college-level mathematics. The initial development of this theory and its relationship to Piaget is described in some detail in [6]. The perspective is continuing to develop and we describe it here in its present form. We should note that, although our theoretical perspective is closely related to the theories of Piaget, this is not so much the case for the other components of our framework. Indeed, considerations of pedagogical strategies are almost absent from the totality of Piaget's work and our methodology for gathering and analyzing data is influenced in only some, but not all, of its aspects by the methodology which Piaget used.

3.1.1 Mathematical knowledge and its construction

Our theoretical perspective begins with a statement of our overall perspective on what it means to learn and know something in mathematics. The following paragraph is not a definition, but rather an attempt to collect the essential ingredients of our perspective in one place.

An individual's mathematical knowledge is her or his tendency to respond to perceived mathematical problem situations by reflecting on problems and their solutions in a social context and by constructing or reconstructing mathematical actions, processes and objects and organizing these in schemas to use in dealing with the situations.

There are, in this statement, references to a number of aspects of learning and knowing. For one thing, the statement acknowledges that what a person knows and is capable of doing is not necessarily available to her or him at a given moment and in a given situation. All of us who have taught (or studied) are familiar with the phenomenon of a student missing a question completely on an exam and then really knowing the answer right after, without looking it up. A related phenomenon is to be unable to deal with a mathematical situation but, after the slightest suggestion from a colleague or teacher, it all comes running back to your

consciousness. Thus, in the problem of knowing, there are two issues: learning a concept and accessing it when needed.

Reflection, in the sense of paying conscious attention to operations that are performed, is an important part of both learning and knowing. Mathematics in particular is full of techniques and algorithms to use in dealing with situations. Many people can learn these quite well and use them to do things in mathematics. But understanding mathematics goes beyond the ability to perform calculations, no matter how sophisticated. It is necessary to be aware of how procedures work, to get a feel for the result without actually performing all of the calculations, to be able to work with variations of a single algorithm, to see relationships and to be able to organize experiences (both mathematical and non-mathematical).

From this perspective we take the position that reflection is significantly enhanced in a social context. There is evidence in the literature (see [32], for example) for the value to students of social interaction and there is also the cultural reality that virtually all research mathematicians feel very strongly the need for interactions with colleagues before, during, and after creative work in mathematics.

The statement describing our theoretical perspective asserts that “possessing” knowledge consists in a tendency to make mental constructions that are used in dealing with a problem situation. Often the construction amounts to reconstructing (or remembering) something previously built so as to repeat a previous method. But progress in the development of mathematical knowledge comes from making a reconstruction in a situation similar to, but different in important ways from, a problem previously dealt with. Then the reconstruction is not exactly the same as what existed previously, and may in fact contain one or more advances to a more sophisticated level. This whole notion is related to the well known Piagetian dichotomy of assimilation and accommodation [21]. The theoretical perspective which we are describing is itself the result of reconstruction of our understanding of Piaget’s theory leading to extension in its applicability to post-secondary mathematics.

Finally, the question arises of what is it that is constructed by the learner, or, in other words, what is the nature of the constructions and the ways in which they are made? As we turn to this issue, it should become apparent that our theoretical perspective, which may appear applicable to any subject whatsoever, becomes specific to mathematics.

3.1.2 Mental constructions for learning mathematics

As illustrated in Figure 2, we consider that understanding a mathematical concept begins with manipulating previously constructed mental or physical objects to form actions; actions are then interiorized to form processes which are then encapsulated to form objects. Objects can be de-encapsulated back to the processes from which they were formed. Finally, actions, processes and objects can be organized in schemas. A more detailed description of each of these mental constructions is given below.

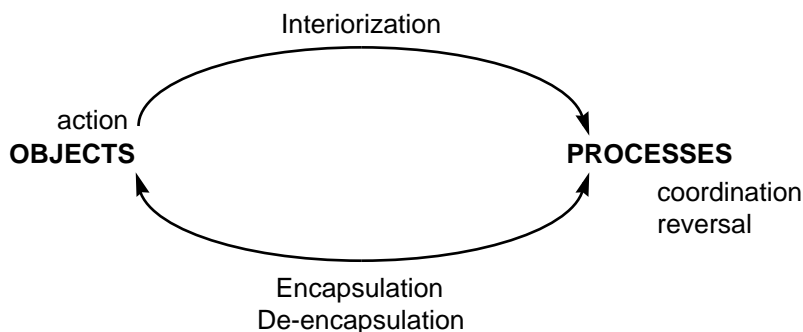


Figure 2: Constructions for mathematical knowledge

In reading our discussion of these specific constructions, we would like the reader to keep in mind that each of the four is based on a specific construction of Piaget, although not always with exactly the same name. What we are calling actions are close to Piaget’s *action schemes*, our processes are related to his *operations*, and *object* is one of the terms Piaget uses for that to which actions and processes can be applied. The term *schema* is more difficult, partly because Piaget uses several different terms in different places, but

with very similar meanings, partly because of the difficulty of translating between two languages which both have many terms related to schema with subtle and not always corresponding distinction, and partly because our understanding of this specific construction is not as far as advanced as it is for the others. Our use of the term in this paper is close to what Piaget calls *schemata* in [22] where his meaning appears to be in some ways similar to the *concept* image of Tall & Vinner [31], and where he talks about thematizing schemas which, essentially refers to making them objects. For a more detailed discussion of the relation between the theories of Piaget and (an early version of) our theoretical framework, the reader is referred to [6].

Action. An action is a transformation of objects which is perceived by the individual as being at least somewhat external. That is, an individual whose understanding of a transformation is limited to an action conception can carry out the transformation only by reacting to external cues that give precise details on what steps to take.

For example, a student who is unable to interpret a situation as a function unless he or she has a (single) formula for computing values is restricted to an action concept of function. In such a case, the student is unable to do very much with this function except to evaluate it at specific points and to manipulate the formula. Functions with split domains, inverses of functions, composition of functions, sets of functions, the notion that the derivative of a function is a function, and the idea that a solution of a differential equation is a function are all sources of great difficulty for students. According to our theoretical perspective, a major reason for the difficulty is that the learner is not able to go beyond an action conception of function and all of these notions require process and/or object conceptions. (See [1] for an elaboration of these issues.)

Another example of an action conception comes from the notion of a (left or right) coset of a group in abstract algebra. Consider, for example, the modular group $[Z_{20}, +_{20}]$ — that is, the integers $\{0, 1, 2, \dots, 19\}$ with the operation of addition mod 20 — and the subgroup $H = \{0, 4, 8, 12, 16\}$ of multiples of 4. As is seen in [10] it is not very difficult for learners to work with a coset such as $2 + H = \{2, 6, 10, 14, 18\}$ because it is formed either by an explicit listing of the elements obtained by adding 2 to each element of H or applying some rule (e.g., “begin with 2 and add 4”) or an explicit condition such as, “the remainder on division by 4 is 2”. Understanding a coset as a set of calculations that are actually performed to obtain a definite set is an action conception. Something more is required to work with cosets in a group such as \mathcal{S}_n , the group of all permutations on n objects where simple formulas are not available. Even in the more elementary situation of \mathcal{Z}_n , students who have no more than an action conception will have difficulty in reasoning about cosets (such as counting them, comparing them, etc.) In the context of our theoretical perspective, these difficulties are related to a student’s inability to interiorize these actions to processes, or encapsulate the processes to objects.

Although an action conception is very limited, the following paragraphs describe the way in which actions form the crucial beginning of understanding a concept. Therefore, our learning-theory-based pedagogical approach begins with activities designed to help students construct actions.

Process. When an action is repeated, and the individual reflects upon it, it may be interiorized into a process. That is, an internal construction is made that performs the same action, but now, not necessarily directed by external stimuli. An individual who has a process conception of a transformation can reflect on, describe, or even reverse the steps of the transformation without actually performing those steps. In contrast to an action, a process is perceived by the individual as being internal, and under one’s control, rather than as something one does in response to external cues.

In the case of functions, a process conception allows the subject to think of a function as receiving one or more inputs, or values of independent variables, performing one or more operations on the inputs and returning the results as outputs, or values of dependent variables. For instance, to understand a function such as $\sin(x)$, one needs a process conception of function since no explicit instructions for obtaining an output from an input are given; in order to implement the function, one must imagine the process of associating a real number with its sine.

With a process conception of function, an individual can link two or more processes to construct a composition, or reverse the process to obtain inverse functions [1].

In abstract algebra, a process understanding of cosets includes thinking about the formation of a set by operating a fixed element with every element in a particular subgroup. Again, it is not necessary to perform the operations, but only to think about them being performed. Thus, with a process conception, cosets can be formed in situations where formulas are not available. (See, for example, [10].)

Object. When an individual reflects on operations applied to a particular process, becomes aware of the process as a totality, realizes that transformations (whether they be actions or processes) can act on it, and is able to actually construct such transformations, then he or she is thinking of this process as an object. In this case, we say that the process has been *encapsulated* to an object.

In the course of performing an action or process on an object, it is often necessary to de-encapsulate the object back to the process from which it came in order to use its properties in manipulating it.

It is easy to see how encapsulation of processes to objects and de-encapsulating the objects back to processes arises when one is thinking about manipulations of functions such as adding, multiplying, or just forming sets of functions. In an abstract algebra context, given an element x and a subgroup H of a group G , if an individual thinks generally of the (left) coset of x modulo H as a process of operating with x on each element of H , then this process can be encapsulated to an object xH . Then, cosets are named, operations can be performed on them ([10]), and various actions on cosets of H , such as counting their number, comparing their cardinality, and checking their intersections can make sense to the individual. Thinking about the problem of investigating such properties involves the interpretation of cosets as objects whereas the actual finding out requires that these objects be de-encapsulated in the individual's mind so as to make use of the properties of the processes from which these objects came (certain kinds of set formation in this case).

In general, encapsulating processes to become objects is considered to be extremely difficult ([22], [28], [29]) and not very many pedagogical strategies have been effective in helping students do this in situations such as functions or cosets. A part of the reason for this ineffectiveness is that there is very little (if anything) in our experience that corresponds to performing actions on what are interpreted as processes.

Schema. Once constructed, objects and processes can be interconnected in various ways: for example, two or more processes may be coordinated by linking them (through composition or in other ways); processes and objects are related by virtue of the fact that the former act on the latter. A collection of processes and objects can be organized in a structured manner to form a schema. Schemas themselves can be treated as objects and included in the organization of "higher level" schemas. When this happens, we say that the schema has been *thematized* to an object. The schema can then be included in higher level schemas of mathematical structures. For example, functions can be formed into sets, operations on these sets can be introduced, and properties of the operations can be checked. All of this can be organized to construct a schema for function space which can be applied to concepts such as dual spaces, spaces of linear mappings, and function algebras.

As we indicated above, our work with, and understanding of, the idea of a schema has not progressed as much as some of the other aspects of our general theory. We are convinced, however, that this notion is an important part of the total picture and we hope to understand it better as our work proceeds. For now, we can only make some tentative observations concerning, for example, the distinction between a genetic decomposition and a schema.

A related question, on which our work is also very preliminary, has to do with relations between the mental constructs together with the interconnections that an individual uses to understand a concept, and the way in which an individual uses (or fails to use) them in problem situations. This is what we were referring to in our use of the term "tendency" in our initial statement. Now, a genetic decomposition for a mathematical concept is a model used by researchers to describe the concept. Our tentative understanding suggests that an individual's schema for a concept includes her or his version of the concept that is described by the genetic decomposition, as well as other concepts that are perceived to be linked to the concept in the context of problem situations.

Put another way, we might suggest that the distinction between schema and other mental constructions is like the distinction in biology between an organ and a cell. They are both objects, but the organ (schema) provides the organization necessary for the functioning of the cells to the benefit of the organism. An individual's schema is the totality of knowledge which for her or him is connected (consciously or subconsciously) to a particular mathematical topic. An individual will have a function schema, a derivative schema, a group schema, etc. Schemas are important to the individual for mathematical empowerment, but in general, we are very far from knowing all of the specifics, nor have we studied much about how this organization determines mathematical performance. All we can do now is to link together mental constructions for a concept in a generic road-map of development and understanding (genetic decomposition) and see how this is actualized in a given individual (schema). Clearly, an individual's schema may include actions or knee-jerk responses such as, "whenever I see this symbol I do that".

3.2 Instructional treatments

The second component of our framework has to do with designing and implementing instruction based on the theoretical analyses. The theoretical perspective on learning we have just described influences instruction in two ways. First, as we have indicated earlier, the theoretical analysis postulates certain specific mental constructions which the instruction should foster. Later in the paper (Section 3.2.2, p. 11) we will consider that effect in relation to specific mathematical content. Before doing that however, we shall describe a second, more global way in which the general theory influences instruction.

3.2.1 Global influences of the theoretical perspective

Returning to our formulation of the nature of mathematical knowledge and its acquisition (p. 5) we consider four components: the tendency to use one or another mathematical construct, reflection, social context, and constructions or reconstructions.

When confronted with a mathematical problem situation, it has frequently been observed that an individual is not always able to bring to bear specific ideas in her or his mathematical repertoire. (See, for instance, [25] for a very sharp example of students clearly possessing certain knowledge, but not being able to use it.) An individual's mathematical knowledge consists in a tendency to use certain constructions, but not all relevant constructions are recalled in every situation. This is one of the reasons that we cannot expect students to learn mathematics in the logical order in which it can be laid out. In fact, according to our theoretical perspective, the growth of understanding is highly non-linear with starts and stops; the student develops partial understandings, repeatedly returns to the same piece of knowledge, and periodically summarizes and ties related ideas together [27]. Our general instructional approach acknowledges this growth pattern by using what we call an *holistic spray*, which is a variation on the standard spiral method ([10]).

In this variation, students are thrust into an intentionally disequilibrating environment which contains as much as possible about the material being studied. The idea is that everything is sprayed at them in an holistic manner, as opposed to being sequentially organized. Each individual (or cooperative learning group) tries to make sense out of the situation — that is, they try to solve problems, answer questions or understand ideas. Different students may learn different pieces of the whole at different times. In this way the students enhance their understanding of one or another portion of the material bit by bit. The course keeps presenting versions of the whole set of material and the students are always trying to make more sense, always learning a little more.

The social context to which our theoretical perspective refers is implemented in our instruction through the use of cooperative learning groups. Students are organized at the beginning of the semester in small groups of three to five to do all of the course work (computer lab, class discussion, homework and some exams) cooperatively. For details see [23]. One consequence of having students work in cooperative groups is that they are more likely to reflect on the procedures that they perform [32].

A critical part of our approach is to implement the results of our theoretical analysis in regular classrooms and to gather data on what happened in those classrooms. Consequently, our curricular designs are tied to the current curricular structures of the semester or quarter system in which we operate. Today, many people are thinking seriously about alternative ways to organize the entire educational enterprise — from

kindergarten through graduate school. As new structures emerge, the generality of our approach will be indicated by the extent to which it can be adapted to these new forms.

Our instructional strategies also try to get students to reflect on their work through the overall structure of a course. A particular pedagogical approach we use, which we refer to as the *ACE Teaching Cycle*, and which we now describe, is not a necessary consequence of the theoretical perspective but is one possible overall design supporting our theoretical analyses. In this design, the course is broken up into sections, each of which runs for one week. During the week, the class meets on some days in the computer lab and on other days in a regular classroom in which there are no computers. Homework is completed outside of class. As indicated above, the students are in cooperative groups for all of this work.

Thus there are three components of the ACE cycle: activities, class discussion, and exercises.

Activities:

Class meets in a computer lab where students work in teams on computer programming tasks designed to foster specific mental constructions suggested by the theoretical analysis. The lab assignments are generally too long to finish during the scheduled lab and students are expected to come to the lab when it is open, or work on their personal computers, or use other labs to complete the assignment. It is important to note that there are major differences between these computer activities and the kinds of activities used in “discovery learning.” While some computer activities may involve an element of discovery, their primary goal is to provide students with an experience base rather than to lead them to correct answers. Through these activities students gain experience with the mathematical issues which are later developed in the classroom phase. We will discuss the computer work in more detail below.

Class Discussion:

Class meets in a classroom where students again work in teams to perform paper and pencil tasks based on the computer activities in the lab. The instructor leads inter-group discussions designed to give students an opportunity to reflect on the work they did in the lab and calculations they have been making in class. On occasion, the instructor will provide definitions, explanations and overviews to tie together what the students have been thinking about.

Exercises:

Relatively traditional exercises are assigned for students to work on in teams. These are expected to be completed outside of class and lab and they represent homework that is in addition to the lab assignments. The purpose of the exercises is for students to reinforce the ideas they have constructed, to use the mathematics they have learned and, on occasion, to begin thinking about situations that will be studied later.

One must consider the question of how our pedagogical strategies are supported by the textbook for the course. We have found that traditional textbooks are not very helpful. For example, template problems (usually a key feature of a textbook) circumvent the disequilibrium and formation of rich mental constructions which we consider necessary for meaningful understanding. Also, as we have indicated, mathematics is not learned in the logical order in which it is presented in most textbooks. It has therefore been necessary to produce appropriate texts for various courses. This has been done, using essentially the same style, for courses in Discrete Mathematics, Precalculus, Calculus, and Abstract Algebra. Other works are in progress. The textbooks are designed to support the constructivist approach to teaching that is described in this paper.

Our textbooks are arranged according to the ACE Cycle. Each book is divided into sections, each of which begins with a set of computer activities, followed by a discussion of the mathematics involved in these activities, and ending with an exercise set. It is expected that a section will be covered in one week with the students doing the computer activities in the lab, and possibly during “open hours”, working through the discussion material in class and doing the exercises as homework.

The textbooks have certain features that, although not always popular with students *or* teachers, we feel are necessary to relate to how learning actually takes place. There are almost no cases of “worked problems” followed by drills in the exercises and no answers to exercises in the back of the book. Wherever

possible, the student is given an opportunity to figure out for her or himself how to solve a problem using the ideas that are being learned. In the computer activities, the students are often asked to solve problems requiring mathematics they have not yet studied. They are encouraged to discuss these issues with their group members or other colleagues, read ahead in the text (where explanations can be found, although usually not directly), or even consult other texts. In other words, the students are forced to *investigate* mathematical ideas in order to solve problems. The explanations that are given in the discussion section are interspersed with many questions whose answers are important for understanding the concepts. These questions are not always answered in the texts (at least not right away) but they are generally repeated in the exercise set.

Although we feel that our textbooks make a major contribution to student learning, the books we produce are not as good for later reference. To overcome this, we try to include an extremely detailed and complete index, but also expect that the student might find some other text which is not so helpful for learning, but may be a better reference book.

3.2.2 Mental constructions

Our main strategy for getting students to make mental constructions proposed by our theoretical analysis is to assign tasks that will require them to write and/or revise computer code using a mathematical programming language. In the sequel, we will refer to this as an **MPL**. The idea is that when you do something on a computer, it affects your mind. This effect occurs in three ways. First, we attempt to orchestrate the effect of these tasks so that we foster the specific mental constructions proposed by our theoretical analyses. Second, the computer tasks which we assign can provide a concrete experience base paving the way for later abstraction. Finally, there is an indirect effect from working with computers which has been reported and is less clearly understood.

Fostering mental constructions directly. This general discussion of our strategy is organized around actions, processes, objects, and schemas. The initial activities included in this section show how students familiarize themselves with the syntax of the programming language. At the same time, since the activities involve direct calculations of specific values, students gain experience constructing actions corresponding to selected mathematical concepts. This experience is built upon in subsequent activities where students are asked to reconstruct familiar actions as general processes. Later activities presented exemplify those that are intended to help students encapsulate processes to objects; these activities typically involve writing programs in which the processes to be encapsulated are inputs and/or outputs to the program. Finally, we describe a more complicated activity in which students need to organize a variety of previously constructed objects into a schema that can then be applied to particular problem situations.

Although our choice for a language in which to work is **ISETL**, there are other possibilities such as **Mathematica** and **Maple**. Because the syntax of **ISETL** is so close to that of standard mathematical notation, the programming aspects of the language are particularly easy to learn. This does not mean, however, that learning to write correct mathematics in **ISETL** is easy. On the contrary, students encounter a great many difficulties in using the language — difficulties that are most often directly associated with mathematical difficulties. For this reason, **ISETL** provides an ideal environment for mathematical experimentation, reflection, and discussion.

For those who know mathematics and have some experience with programming, little explanation is necessary in describing examples of the use of **ISETL**. Hence we will minimize our explanations of syntax in the sequel. (See [8] for a discussion of the use of this language and [3] for details on its syntax.)

Actions We begin with activities in which students try to repeat on their terminal screens what is written in the text, or to predict what will be the result of running code that is given to them, or to modify code they have been given.

Following is a set of **ISETL** instructions as they would appear on the screen followed by the computer's response. It is taken from the first pages of the textbook used in the C^4L calculus project [11].

The `>` symbol is the **ISETL** prompt and lines which begin with this symbol or `>>`, which indicates incomplete input, are entered by the user. Lines without these prompts are what the computer prints on the screen.

```

> 7+18;
25;

> 13*(233.8);
3039.400000;

> 5 = 2.0 + 3;
true;

> 4 >= 2 + 3;
false;

> 17
>> + 23.7 - 46
>> *2
>> ;
-51.300000;

> x := -23/27;
> x;
-0.851852;

> 27/36;
0.750000;

> p := [3,-2]; q := [1,4.5]; r := [0.5,-2,-3];
> p; q; r;
[3, -2];
[1, 4.500000];
[.500000, -2, -3];
> p(1); p(2); q(2); r(3);
3;
-2;
4.500000;
-3;
> p(1)*q(1) + p(2)*q(2);
-6.000000;

> length := 0;
> for i in [1..3] do
>>   length := length + r(i)**2;
>>   end;
> length := sqrt(length);
> length;
3.640055;

```

Students are asked to do these exercises for the purpose of becoming familiar with their **MPL**, but at the same time, there are several mathematical concepts which they have an opportunity to construct at the action level. For example, there are simple propositions, the formation of pairs and triples of numbers and the action of picking out an indexed term of a given sequence. Also the concept of dot product appears as an action. Finally, the algorithm for computing the length of a vector in three dimensions appears as an action because the calculation is explicit and is applied to a single vector.

Processes. In the previous paragraph we indicated how students might be led to construct actions by writing code that makes a computation once with specific numbers. Now we try to get them to build on these actions by performing activities intended to help them reconstruct their actions as processes. For example, students are led to interiorize actions to processes when they replace code written to perform a specific calculation by a computer function which will perform the action for any values. Thus, for the calculation of the length of a three-dimensional vector (which is our last example above), we might ask students to write the following computer function. (The last two lines assume that \mathbf{r} has been defined previously as above, or as any three-dimensional vector.)

```
> length := func(v);
>>   l := 0;
>>   for i in [1..3] do
>>     l := l + v(i)**2;
>>   end;
>>   return sqrt(l);
>> end;
> length(r);
3.64005;
```

There are a variety of ways that programming activities can be used to help students to understand functions as processes. Research suggests that students who write code to implement the point-wise sum, product, and composition of functions tend to make progress in developing a process conception of function [1]. Programming activities have also been used to help students get past the well-known difficulty many have in seeing that a piece-wise defined function is still a function, and that properties can be studied at the seam points as well as other places. A process conception of function emerges as students write programs in which the definition of a piece-wise function is implemented through the simple use of a conditional:

```
f := func(x);
    if x <= 1 then return 2-x**2;
    else return x/2 + 1.5;
    end;
end;
```

Following is how the code would look in **Maple**.

```
f:=proc(x);
if x<=1 then 2-x**2;
else x/2 + 1.5;
fi;
end;
```

A completely different approach considered by Kaput has students (in grades 3-13) *begin* with piecewise-defined functions and deal with them graphically using what is referred to as SimCalc simulations. (See [18] for details.)

We can also mention the example of boolean-valued functions of the positive integers. Our research suggests that one of the difficulties students have with proof by induction is at the very beginning. A student is faced with a problem: show that a certain statement involving an arbitrary integer is true for all (sufficiently large) values of the integer. This kind of problem is very new and difficult for most students. It really is a (mental) function which accepts a positive integer and plugs it into the statement to obtain a proposition which may be true or false — and the answer could be different for different values of n . Once again, expressing this problem as a function in an **MPL** is a big help for students in figuring out how to begin.

Suppose, for example, that the problem is to determine if a gambling casino with only \$300 and \$500 chips can represent (within the nearest \$100) any amount of money beyond a certain minimum. We encourage

students to begin their investigation by writing a computer program that accepts a positive integer and returns a boolean value. The following is one solution they generally come up with in our elementary discrete mathematics course.

```
P := func(n);
    if is_integer(n)
        and n > 0
            and exists x,y in [0..n div 3] | 3*x + 5*y = n
        then return true;
        else return false;
    end;
end;
```

Objects. Objects are obtained by encapsulation of processes, and an individual is likely to do this when he or she reflects on a situation in which it is necessary to apply an action to a dynamic process. This presents a difficulty because the action cannot be applied to the process until after the process has been encapsulated to an object. But, as we have said before, mental constructions do not seem to occur in simple logical sequences. In fact, the following three things can all be happening at the same time, initially in some amorphous combination:

1. the need to create an object (in order to apply an action to a process),
2. the encapsulation of the process to form the object, and
3. the application of an action to that object.

Gradually, as the learner reflects, he or she is able to differentiate, reorganize, and integrate the components of this experience so that a clear application of the action to the object is apparent.

Consider, for example, a student who has just learned what a permutation of $\{1, \dots, n\}$ is, and is now faced with composing permutations. On the one hand, he or she could focus on permutations as processes, and simply perform the linking of the processes to get the composition. That would require only a process, but not an object conception of permutation. At a higher level, though, if the student is trying to think of composition as a binary operation, he or she would begin to see permutations as inputs to the binary operation process, and thus as objects.

Getting students to do all of this is another matter and there are very few effective pedagogical methods known. As we have indicated, one such method is to put students in situations where a problem must be solved or a task must be performed by writing programs in which the processes to be encapsulated are inputs to, and/or outputs of, the programs. Thus, in addition to having syntax similar to standard mathematical notation, we require that an **MPL** treat functions as first-class data types, that is, as entities which can be passed as input or output parameters to and from other student-defined procedures.

Continuing with our treatment of mathematical induction, we can report that students learn to treat propositions about natural numbers as objects. At the same time they develop an understanding of the “implication from n to $n + 1$ ”, that is, $P(n) \longrightarrow P(n + 1)$ understood as an object whose truth value as n varies is to be considered. Our approach is to have them write and apply the following program which accepts a function whose domain is the positive integers and whose range is the two element set $\{\mathbf{true}, \mathbf{false}\}$. This program returns the corresponding implication valued function. (The symbol **\$** refers to a comment and anything after this symbol on the line in which it appears is ignored by **ISETL**.)

```
implfn := func(P); $ P is a boolean-valued function.
    return func(n);
        return P(n) impl P(n+1);
    end;
end;
```

In calculus, two extremely important examples of construction of objects occur in connection with derivatives and integrals. Although it is very simple for mathematicians, our experience suggests that the idea

that the derivative of a function is a function is not immediate for students. Writing a program such as the following, which accepts a function and returns an approximation to its derivative appears to help.

```
df := func(f);
    return func(x);
        return (f(x + 0.00001) - f(x))/0.00001;
    end;
end;
```

In this program, **f** is the variable and so it does not need to be defined before the program is run. Once it has been run, a function can be defined and given any name, for example **g** and then **df(g)** will be a function that can be evaluated, assigned to be the value of a variable, graphed or treated in any other way that functions are treated.

It is important to note that the reason we are using **ISETL** here is because there are essential understandings we are trying to get students to construct as specified by our genetic decomposition, and we cannot do this with many other systems. For example, writing a program that constructs a function for performing a specific action in various contexts tends to get students to interiorize that action to a process. Perhaps even more critical is our method of getting students to make a function an object in their minds by using function programs as input to another program which students write. This latter program performs a process and returns a function as output. This can also be done using systems such as **Maple** or **Mathematica** as alternatives to **ISETL**. For example, if **f** is a simple **proc** in **Maple**, then the following **Maple** version of **df** could be used in a similar manner.

```
with(student);
df:=proc(f) local dq;
dq:=(f(x+.00001)-f(x))/0.00001;
makeproc(dq,x);
end;
```

Integration is more difficult. The idea of defining a function by using the definite integral with one limit of integration fixed and the other allowed to vary is a major stumbling block for calculus students. In our treatment of integration, students have written a program called **Riem** which accepts a function and a pair of numbers, and computes an approximation to the integral of the function over the interval determined by the points. The students are then asked to write the following program.

```
Int := func(f,a,b);
    return func(x);
        if a <= x and x <= b then return Riem(f,a,x);
        end;
    end;
```

Using this program, students are able to construct and study approximations to the logarithm function and inverse trigonometric functions.

Schemas. Our use of programming activities to help students form schemas to organize collections of individual constructs and other schemas is, at the time of this writing, somewhat ad hoc. Some progress is being made, and has been reported in [2] for a limit schema. Roughly speaking, we ask students to write a set of computer programs that implements a mathematical concept and then to apply their code to specific situations.

For example, developing a schema for the Fundamental Theorem of Calculus requires very little code but it is extremely complicated. Having written the two computer functions **df** to approximate the derivative and **Int** to approximate the integral (see above), students are asked to write code that will first do one and then the other, in both orders. This problem gives the students considerable difficulty and they struggle with it for a long time. We feel that this is a useful struggle because it has to do with their ability to interpret

functions as objects, to develop processes corresponding to differentiation and integration, and to put it all together in what is essentially a statement of the Fundamental Theorem.

The actual code to solve this problem is very short:

```
> df(Int(f(a,b));  
  
> Int(df(f),a,b);
```

We ask students to apply their code to a specific function and to construct a table with four columns: values of the independent variable, corresponding values of f , and corresponding values of the above two lines of code. When the example is a function that does not vanish at a , then the second and third columns are identical, but the fourth is different. The students see the point right away — all three columns are supposed to be the same, but they feel they have made an error in connection with the last column. After some investigation, many students tend to discover on their own the idea of the “constant of integration.”

We should note here that it is not our intention to suggest that the approach we are describing is the *only* way to help students understand the ideas surrounding the Fundamental Theorem of Calculus, such as velocity and accumulation and the relationship between them. There are other approaches, such as that being pursued by Jim Kaput [18] in which children control simulated motions on a computer screen. In this case, the functions are defined graphically rather than analytically. It would be interesting to see whether the theoretical framework we are using would apply in the same way to describe student understanding of functions.

Sometimes we don't ask students to write code but rather to investigate code which we provide. We do this in situations where the particular code involves more in the way of programming issues than mathematical issues. This is the case for the following example which simulates the operation of induction. This code makes use of the computer function, `implfn`, which they have written (see above) and is applied to a boolean-valued function P . The first few lines find a starting point and the rest of the code runs through the induction steps. If the proposition does hold from the selected starting point on, then the code will run forever.

```
start := 1;  
while P(start) = false do  
  start := start + 1;  
end;  
  
L := [];  
n := start; L(n) := true;  
while L(n) = true and implfn(P)(n) = true do  
  L(n+1) := true;  
  n := n+1;  
  print "The proposition P is true for n = ", n;  
end; print "P is not proven for n = ", n+1;
```

Making the abstract concrete A second way in which working with an appropriate computer language can help students construct mathematical concepts is that the computer can provide an environment where students are able to make certain abstract notions concrete. Consider, for example, the statement that a function f maps its domain D onto a set S :

For each $y \in S$ there exists an x in D such that $f(x) = y$.

Students may consider such a precise definition to be a difficult abstraction. They can be helped by working with a **MPL** such as **ISETL** in which such a statement can be made, run, tested, and reasoned about. Following is **ISETL** code that expresses the same mathematical statement.

```
forall y in S | (exist x in D | f(x)=y);
```

If S , D , and f are defined, then this code can be run to return the value `true` or `false`.

Our way of using this feature in instruction is to begin with a somewhat vague discussion of the essential idea, perhaps in the context of students working on a problem that requires this idea. Then we ask students to write a program such as the above and use this program in solving problems. It appears that writing and working with a program that he or she has written helps the student make concrete the ideas embodied in the program.

Indirect effects of working with the computer One example of an indirect effect on students is reported in [1]. Students were asked at the beginning of a discrete mathematics course to give examples of functions. A very high percentage of their responses were functions defined by simple expressions such as $x^2 + 1$ and many students displayed no more than an action conception of function, or even no useful conception at all. After several weeks of work with a mathematical programming language involving procedures, sets, and finite sequences, but before any explicit study of functions in the course, students were asked again to provide examples. This time the examples given were much richer and more varied, with a number of students' responses indicating they were moving from action conception or no conception to a process conception of function.

3.3 Data

The third component of the framework has to do with the collection and analysis of data.

There are several kinds of data which must be gathered in studies under this framework. There must be information about the students and the course(s) taken. In some cases, we gather data about students who have previously studied the mathematics we are concerned with under traditional instruction. In other cases, we will study students who have experienced the kind of instruction we describe in this paper. Where appropriate, we will summarize rough comparisons of performances of the two kinds of students. Sometimes, students' attitudes about mathematics and about the particular subject matter is of concern in the study. Most important, of course, is data that allows us to analyze the students' relationship to the particular material; it is this last category that we will consider now in some detail.

3.3.1 Forms of data

We see two reasons to gather a number of different kinds of data about the students' relationship to the material. First, the methods used do not provide precise information leading to inescapable conclusions. The best we can hope for is data that is illustrative and suggestive. Our confidence in any tentative conclusions that might be indicated is increased as we widen the sources of information about student knowledge and how it develops. The second reason is that we are interested in two different kinds of issues: what mental constructions students might be making and how much of what mathematics do they seem to be learning and using? That is, we are comparing the mental constructions the students appear to be making with those called for in the theoretical analyses, and at the same time we are searching for the limits of student knowledge. Often, the same information will shed light on both questions, but sometimes it is necessary to use different kinds of data to investigate these different issues.

In our studies we gather data using three kinds of instruments: written questions and answers in the form of examinations in the course or specially designed question sets; in-depth interviews of students about the mathematical questions of concern; and a combination of written instruments and interviews. For purposes of data analysis, all of our data is aggregated across the set of students who participated in the study.

The written instruments contain fairly standard questions about the mathematical content and they are analyzed in relatively traditional ways. We grade the responses on appropriate scales from incorrect to correct with partial credit in between, and then count the scores. Where appropriate we list the specific points (both correct and incorrect) in the responses of all the students and collate those points. This information tells us about what the students may or may not be learning and also about possible mental constructions.

The interviews of students form the most important and the most difficult part of our observation and assessment activity. The audio-taped transcripts of the interviews complement the record of written work which the student completes during the interview. An even more complete picture might be obtained by videotaping the interviews, but we have not yet added this component to our interviews. One reason why the interviews are far more valuable than written assessment instruments used alone is that for one student

the written work may appear essentially correct while the transcript reveals little understanding, while for another student the reverse may be true. There is no set recipe for designing interview questions. The research team proposes, discusses, and pilots questions intended to test the hypotheses set forth by the current version of the genetic decomposition for the topic under study. An instrument is then put together which is administered to a number of students by a number of researchers. Communication among the interviewers before and between interviews is important to increase consistency.

A second combination of written instruments and interviews is used in the following way. The written instrument is administered to a total population and the responses are used in designing interview questions. For example, the student might be asked in the interview to explain what was written, or if he or she wished to revise the response. If the written instrument were administered to a group (as in a group examination such as described in [23]), we would ask whether the interviewee was fully in accord with and understood the group response. Individuals are selected for interview based on their responses. The idea is that it may not be necessary to interview all of the students who gave a certain written response. In selecting students to interview, we try to access the full range of understanding by including students who gave correct, partially correct, and incorrect answers on the written instruments. We also routinely select students who appear to be in the process of learning some particular idea rather than those who have clearly mastered it or those who had obviously missed the point. In this way, it is possible to interview only a small percentage of the total population, but still investigate every written response that appeared. Usually it is feasible, for each specific response, to interview more than one student who made that response. A combination of the two kinds of instruments is discussed in [10].

3.3.2 Analyses of interview data

One of the most serious practical difficulties in doing qualitative research is the very large amount of data that is generated and that must be analyzed. We believe that our framework offers an alternative to the approach used by some researchers (see, for example, [27]) who attempt to make a full analysis of the total set of data. In the following paragraphs we begin by discussing the goals of our analyses of interviews and then describe the specific steps by which we attempt to achieve these goals.

Goals of the data analyses Consider, in Figure 1, p. 4 the bi-directional arrow connecting the theoretical analyses with the observations and assessments. What the data tells us can support, or lead to revisions of, the particular analyses that have been made of the concept being studied, and even the general theoretical perspective. This is the meaning of the arrowhead pointing up towards the theoretical analysis in the figure.

On the other hand, the arrowhead in the opposite direction, pointing down, indicates our method of using the analysis of the concept to focus our investigation of the data. In other words, the theoretical analysis tells us what questions to ask of the data. More specifically, our study of the data is narrowed by focusing on the question of whether the specific constructions proposed by the theoretical analysis (which are the main determinant of the design of instruction) are in fact being made by the students who succeed in learning the concept. Put another way, we ask if making, or failing to make, the proposed constructions is a reasonable explanation of why some students seemed to learn the concept and others did not.

Of course, actual student learning is seldom characterizable in binary, yes/no terms. Students range on a spectrum from those who seem to understand nothing (about the particular piece of mathematics) to those who indicate a mature understanding compatible with the understanding of mathematicians. The goal of our analysis of the data is to establish a parallel spectrum of mental constructions, going from those who appeared to construct very little, through those who constructed bits and pieces, to those who seemed to have made all of the constructions proposed by the theoretical analysis.

It is easy to see how such an approach requires iterating through the steps in Figure 1 as the parallel spectra of mathematical understanding and mental constructions are unlikely to be completely similar initially and the researcher must endeavor, in the repetitions, to try to bring them in line with each other.

Steps of the data analyses Our interviews are audio-taped and transcribed. These transcriptions, together with any writing performed by the interviewee and any notes taken by the interviewer, make up the data which are to be analyzed. We do this in five steps.

1. **Script the transcript.** The transcript is put in a two-column format. The first column contains the original transcript and the second column contains an occasional brief statement indicating what is happening from that point until the next brief statement. It is convenient to number the paragraphs at this point.

2. **Make the table of contents.** A table of contents is constructed. The statements in the scripting should be a refinement of the items in the table of contents.

These first two steps are designed to make the transcript more convenient to work with and to give the researcher an opportunity to become familiar with its contents.

3. **List the issues.** By an issue we mean some very specific mathematical point, an idea, a procedure, or a fact, for which the interviewee may or may not construct an understanding. For example, in the context of group theory one issue might be whether the student understands that a group is more than just a set, that is, it is a set together with a binary operation.

The researcher begins to generate the list of issues by reading carefully through the transcript for each interviewee writing down each issue that seems to be discussed and noting the page numbers (or paragraph numbers) where it appears. These lists of issues for individuals are then transposed to form a single list of issues and, for each, a list of the specific transcripts (and location) in which it occurred. We believe that the best results are obtained at this step if these lists are generated independently by several researchers with subsequent negotiations to reconcile differences. Since the list of issues varies widely from one set of interviews to another, and since the issues are often not characterizable in terms of number or type of occurrence, we have not attempted to produce information such as inter-rater reliabilities. However, we have found that in most cases, the various researchers independently produce lists of issues which are very similar, and that differences in these lists are often a matter of a single issue being referred to by several different names. Negotiation amongst the researchers serves both to reconcile these differences and to clarify the issues and the terminology being used to describe them.

At this point a selection is made. If an issue occurs for only a very small number of interviewees, and at only a few isolated places, then it is unlikely that this data will shed much light on that issue. There are several reasons why this might occur. Perhaps no student came close to understanding the concept; perhaps all students were well beyond their struggles to understand that concept; or, it is possible that the interview questions were not successful in getting many students to confront these issues very often. In the latter case, it is natural to relegate the issue to future study.

The research team chooses for further study those issues which occur the most often for the most interviewees, and for which there is the largest range of successful, unsuccessful, and in-between performance to be explained.

4. **Relate to the theoretical perspective.**

Each issue is considered in detail. The researchers try to explain the differences between the performances of individual students on the issue in terms of whether they constructed (or failed to construct) the actions, processes, objects, and schemas proposed by the theoretical analysis. If necessary to bring the theoretical analysis more in line with the data, the researchers may drop some constructions from the proposed genetic decomposition, or look for new ones to add to the theoretical analysis.

Focusing on the successes which have occurred, the researchers attempt to reconcile the way in which successful students appear to be making use, in their thinking, of the constructions predicted by the theoretical analysis. Again, adjustments in that analysis are made as necessary.

If the data appear to be too much at variance with the general theoretical perspective, consideration is given to revisions of the perspective. This can take the form of adding new kinds of constructions, or revising the explanations of constructions already a part of the theory.

If drastic changes are required very often, or if each new iteration of the framework continues to require major changes and the process is more like an oscillation than a convergence, then consideration must be given to rejecting the general theoretical perspective.

In each of these steps, each member of the research team makes an independent determination and all differences are reconciled through negotiation. Nothing appears in the final report that is not agreed

to by all authors. This is the closest we come to objectivity in these considerations, and it is one reason why our papers tend to have long lists of authors.

5. **Summarize performance.** Finally, the mathematical performance of the students as indicated in the transcripts is summarized and incorporated in the consideration of performance resulting from the other kinds of data that were gathered.

4 Discussion of accuracy and assesment

We return now to the issues raised in Section 2.2, p. 5.

We have already discussed the development of our theoretical perspective beginning with its origins in the ideas of Piaget and as a result of our work within the framework (Section 2.1, p, 4). We also explained in some detail how the theoretical analysis influences our instructional treatment (Section 3.2, p. 9.) and the relationship between the theoretical analysis and the analysis of data (Section 3.3.2, p. 18.) Regarding the use of data from other studies, our approach is to use all data that is available to the researchers at the time that they make their analyses.

There are two main issues that remain to be discussed: the relation between our theoretical perspective and accuracy about what is going on in the mind of the student, and the question of assessment, including a consideration of the circumstances under which our theory could be falsified.

4.1 Our theory and reality

It is important to emphasize that, although our theoretical analysis of a mathematical concept results in models of the mental constructions that an individual might make in order to understand the concept, we are in no way suggesting that this analysis is an accurate description of the constructions that are actually made. We believe that it is impossible for one individual to really know what is going on in the mind of another individual. In this respect our theoretical framework is like its underlying radical constructivist perspective which Glasersfeld notes, “is intended, not as a metaphysical conjecture, but as a conceptual tool whose value can be gauged only by using it [14].” All we can do is try to make sense out of the individual’s reactions to various phenomena.

One approach would be to try to make inferences from these reactions about the actual thinking processes of the respondent. We reject this because there is no way that we could check our inferences. Rather, we take something of the view of Glasersfeld [13] and consider only whether our description of the mental constructions is compatible with the responses that we observe. That is, we ask only whether our theoretical analysis is a reasonable explanation of the comments and written work of the student. With respect to the instructional treatment, we confine ourselves to asking whether those strategies that are derived from our explanations appear to lead to the student learning the mathematical concept in question.

4.2 Assessment

We make both an internal and external assessment of our results as the work proceeds.

Internally, we ask whether the theoretical analysis and resulting instruction “work” in the sense that students do (or do not) appear to be making the mental constructions proposed by the theory. That is, are the students’ responses reasonably consistent with the assertion that those mental constructions are being made?

Externally, we ask if the students appear to be learning the mathematical concept(s) in question. We ask and answer this question in more or less traditional terms through the results of examinations and performance on the mathematical questions in our interviews.

Finally, at the extreme end of assessment is the question of falsification. Any scientific theory must contain within it the possibility that an analysis, or even the entire theory should be rejected. In our framework, revisions including major changes in, or even rejection of, a particular genetic decomposition can result from the process of repeating the theoretical analyses based on continually renewed sets of data. As we indicated above (Section 4, p. 19), the need for continual and extensive revisions could lead to complete rejection of the general theory and, presumably, the entire framework with which we are working.

In a more positive vein, the continual revisions of our theoretical analyses and the applications to instruction in ongoing classes with, presumably, successful results over a period of time tends to ensure that the longer the work continues, the less likely that our framework and the general theoretical perspective it includes will turn out to have been totally useless. This is not to say, of course, that in the future, some more effective and more convenient framework and/or theory might not emerge and replace what we are using.

This paper has focused on the research methodology in a general approach to research and development in undergraduate mathematics education. It is closely related and complementary to two other papers about this same approach: [8] and [9]. The former concentrates on a somewhat historical description of the programming language **ISETL** and the latter focusses on the pedagogical aspect.

5 Conclusion

In conclusion, we would like to make one final point about the use of the framework described in this paper. Throughout our discussion we have given examples from previous studies in which developing forms of this framework were applied. These investigations had to do with the concept of function, mathematical induction, and predicate calculus. In subsequent publications, we expect to report on studies of students learning concepts in calculus and abstract algebra. Taken as a totality, these papers express a particular approach to investigating how mathematics can be learned and applying the results of those investigations to help real students in real classes. It is our intention to provide enough information in these reports to allow the reader to decide on the effectiveness of our method for understanding the learning process and for helping students learn college level mathematics.

6 Bibliography

References

- [1] Breidenbach, D., Dubinsky, E., Hawks, J., & Nichols, D. (1992). Development of the process conception of function. *Educational Studies in Mathematics*, *23*, 247–285.
- [2] Cottrill, J., Dubinsky, E., Nichols, D., Schwingendorf, K., Thomas, K., & Vidakovic, D. (in press). Understanding the limit concept: Beginning with a coordinated process schema. *Journal of Mathematical Behavior*.
- [3] Dautermann, J. (1992). *Using ISETL 3.0: A Language for Learning Mathematics*. St Paul: West.
- [4] Davis, R. (1984). *Learning Mathematics: The Cognitive Science Approach to Mathematics Education*. Norwood, NJ: Alex Publishing Corporation.
- [5] Dubinsky, E. (in press). On learning quantification. *Journal of Computers in Mathematics and Science Teaching*.
- [6] Dubinsky, E. (1991). Reflective abstraction in advanced mathematical thinking. In D. Tall (Ed.), *Advanced Mathematical Thinking* (pp. 231–250). Dordrecht, The Netherlands: Kluwer.
- [7] Dubinsky, E. (1994). A theory and practice of learning college mathematics. In A. Schoenfeld (Ed.), *Mathematical Thinking and Problem Solving* (pp. 221–243). Hillsdale, NJ: Erlbaum.
- [8] Dubinsky, E. (1995). ISETL: A Programming Language for Learning Mathematics, *Communications in Pure and Applied Mathematics*, *48*, 1–25.
- [9] Dubinsky, E. (in preparation). Programming to Learn Advanced Mathematical Topics: One of Many Computational Environments.
- [10] Dubinsky, E., Dautermann, J., Leron, U., & Zazkis, R. (1994). On learning fundamental concepts of group theory. *Educational Studies in Mathematics*, *27*, 267–305.

- [11] Dubinsky, E., Schwingendorf, K. E., & Mathews, D. M. (1995). *Calculus, Concepts and Computers* (2nd ed.). New York: McGraw-Hill.
- [12] Fisher, Sir R. A. (1932). *Statistical Methods for Research Workers*. Edinburgh: Oliver & Boyd.
- [13] Glasersfeld, E. von (1987). Learning as a constructive activity. In C. Janvier (Ed.), *Problems of Representation in the Teaching and Learning of Mathematics* (pp. 41–69). Hillsdale, NJ: Erlbaum.
- [14] Glasersfeld, E. von (1995). *Radical Constructivism: A Way of Knowing and Learning*. New York: Falmer Press.
- [15] Griese, D. (1981). *The Science of Programming*. New York: Springer-Verlag.
- [16] Gutting, G. (Ed.) (1980). *Paradigms and Revolutions: Appraisals and Applications of Thomas Kuhn's Philosophy of Science*. Notre Dame, IN: University of Notre Dame Press.
- [17] Jacob, E. (1988). Clarifying qualitative research: A focus on traditions. *Educational Researcher*, 17(1), 16–24.
- [18] Kaput, J. & Roschelle J. (unpublished). *Year 2 Annual Report to NSF: SimCalc Annual Report to NSF: Materials, Tests, and Results, 1995*. (Available from first author at Department of Mathematics, UMass Dartmouth, No. Dartmouth, MA 02747.)
- [19] Kuhn, T. (1970). *The Structure of Scientific Revolutions* (2nd ed.). Chicago: University of Chicago Press.
- [20] Patton, M. Q. (1990). *Qualitative Evaluation and Research Methods*. Newbury Park: Sage Publications.
- [21] Piaget, J. (1972). *The Principles of Genetic Epistemology* (W. Mays, Trans.). London: Routledge & Kegan Paul. (Original work published 1970)
- [22] Piaget, J. & Garcia, R. (1989). *Psychogenesis and the History of Science* (H. Feider, Trans.). New York: Columbia University Press. (Original work published 1983)
- [23] Reynolds, B., Hagelgans, N., Schwingendorf, K., Vidakovic, D., Dubinsky, E., Shahin, M., & Wimbish, G. (1995). *A Practical Guide to Cooperative Learning in Collegiate Mathematics* (MAA Notes Number 37). Washington, DC: The Mathematical Association of America.
- [24] Romberg, T. (1992). Perspectives on scholarship and research methods. In D. A. Grouws (Ed.), *Handbook of Research on Mathematics Teaching and Learning* (pp. 49–64). New York: Macmillan.
- [25] Schoenfeld, A. H. (1986). On having and using geometric knowledge. In H. Hiebert, (Ed.), *Conceptual and procedural knowledge: The case of mathematics* (pp. 225–264). Hillsdale, NJ: Erlbaum.
- [26] Schoenfeld, A. H. (1994). Some notes on the enterprise (Research in collegiate mathematics education, that is). *Research in Collegiate Mathematics Education*, 1, 1–19.
- [27] Schoenfeld, A. H., Smith, J. P., III, & Arcavi, A. (1993). Learning — the microgenetic analysis of one student's understanding of a complex subject matter domain. In R. Glaser (Ed.), *Advances in Instructional Psychology*, 4. Hillsdale, NJ: Erlbaum.
- [28] Sfard, A. (1987). Two conceptions of mathematical notions, operational and structural. In A. Borbàs (Ed.), *Proceedings of the 11th Annual Conference of the International Group for the Psychology of Mathematics Education* (pp. 162–169). Montreal: University of Montreal.
- [29] Sfard, A. (1991). On the dual nature of mathematical conceptions. *Educational Studies in Mathematics*, 22, 1–36.
- [30] Suppe, F. (Ed.) (1977). *The Structure of Scientific Theories* (2nd Ed.). Urbana, IL: University of Illinois Press.

- [31] Tall, D. & Vinner, S. (1981). Concept image and concept definition in mathematics with particular reference to limits and continuity. *Educational Studies in Mathematics*, 12, 151–169.
- [32] Vidakovic, D. (1993). *Differences between group and individual processes of construction of the concept of inverse function*. Unpublished doctoral dissertation, Purdue University, West Lafayette, Indiana.

Acknowledgements

The authors would like to express their most sincere gratitude to two groups of people. First, to all of the researchers whose works we have cited in this monograph for contributing immeasurably to our collective understanding. Second, to the members of our Research in Undergraduate Mathematics Education Community (RUMEC) in general with special thanks to Bernadette Baker, Julie Clark, Jim Cottrill, and Georgia Tolia for their thoughtful comments and criticisms of earlier drafts of this manuscript. Finally, we are grateful to James Kaput and Alan Schoenfeld for their numerous and penetrating suggestions.

This work was partially supported by grants from the National Science Foundation, Division of Undergraduate Education (DUE) and the Exxon Education Foundation.