

# **An Intelligent Tutoring System for Robust Division Skill**

Stefan King  
s1234668  
22-8-2008

Supervisors:

V. Aleven, Carnegie Mellon University  
D.H. van Rijn, University of Groningen

Artificial Intelligence  
University of Groningen

## **Preface**

This document is the result of my work at Carnegie Mellon University in Pittsburgh. During six months of research I completed two related projects: first I designed an Intelligent Tutoring System to support children in learning division. Then I evaluated the system, with a study at an elementary school, for my Master thesis.

The first text in this document is my Master thesis. There I introduce the theory of teaching division and the use of Intelligent Tutoring Systems for this end. Then I present the results of the evaluation. For the technical details of the system and the method of design, see the second text.

I am grateful for the things I have learned, and for the help and encouragement I received from my supervisors, Vincent Aleven and Hedderik van Rijn.

Stefan King

## **Contents**

<b>Evaluation of an Intelligent Tutoring System for Robust Division Skill ...</b>	<b>3</b>
<b>1.Introduction .....</b>	<b>3</b>
<b>2.Theory.....</b>	<b>7</b>
<b>3.Research Question .....</b>	<b>22</b>
<b>4.Method .....</b>	<b>25</b>
<b>5.Results .....</b>	<b>35</b>
<b>6.Discussion and Conclusion .....</b>	<b>40</b>
<b>7.References .....</b>	<b>42</b>
<b>Designing and Implementing an Intelligent Tutoring System for Robust</b>	
<b>Division Skill .....</b>	<b>46</b>
<b>1.Introduction .....</b>	<b>47</b>
<b>2.Theory .....</b>	<b>48</b>
<b>3.Research Question and System Design .....</b>	<b>49</b>
<b>4.Discussion and Conclusion .....</b>	<b>58</b>
<b>5.References .....</b>	<b>59</b>
<b>Appendix A .....</b>	<b>60</b>

# Evaluation of an Intelligent Tutoring System for Robust Division Skill

Master Thesis

Stefan King, s1234668, [Stefan@ai.rug.nl](mailto:Stefan@ai.rug.nl)

**Abstract** This master thesis describes the evaluation study of an Intelligent Tutoring System (ITS) for Robust Division Skill. The study focuses on the results of 4<sup>th</sup> grade students after working with the system in two conditions where different aspects of basic division learning are promoted. In one condition fluency was supported by having the student work on as many division problems as possible, and in the other condition “sense making” of division was supported, by encouraging reflection on the material. There is a trade-off between the number of problems solved, and the depth of understanding the problems solved. The students were tested on learning gains, fluency at division, and the transfer of division ability to new situations. The results show no significant learning gains. The most plausible explanations for this is that the students who participated in the experiment were already too proficient at division to benefit from further practice. Other possibilities are covered in the discussion.

## 1. Introduction

Cognitive scientists and educators have long been interested in the question: How can we get students to learn with understanding (National Council of Teachers of Mathematics, 2003)? Just exactly what it means to learn with understanding is somewhat difficult to pinpoint. Transfer to new and unfamiliar types of problems is often seen as a key criterion of understanding (Simon, 1987). An ability to explain or justify problem-solving steps is also an important

indicator (Aleven, Koedinger & Cross, 1999). At the Pittsburgh Science of Learning Center, the term “robust knowledge” is used to refer to the three related and mutually supporting concepts of understanding, fluent computation, and transfer to unfamiliar problems. Research on what it takes to teach robust knowledge has yielded interesting instructional programs that elaborate, make visible, support, and help students reflect upon metacognitive processes that are conducive to the construction of knowledge. (Aleven, et al., 1999). Educators hope that new knowledge learned by students in school is learned deeply (NCTM, 2003). “Deep learning” is the term used by the NCTM to refer to the same ideas that are summarized in the term “robust knowledge” that is used by the Pittsburgh Science of Learning Center: problems are solved fluently and with conceptual understanding. How these ideas are related, and what this implies for teaching mathematics, is explained later sections.

This Master thesis is an evaluation of an Intelligent Tutoring System for robust division skill. Intelligent Tutoring Systems (ITS) have been developed to aid instruction on various domains, including mathematics (Corbett, Koedinger & Anderson, 1997). ITS provide support for guided learning by doing. They assign problems to students on an individual basis, monitor students’ solution steps and provide context-sensitive feedback and hints (Anderson, Corbett, Koedinger & Pelletier, 1995). The system provides direct customized instruction or feedback to students. It can employ different methods for feedback and has flexibility in the way learning material can be presented. The intent is to engage students in sustained reasoning, to interact based on understanding of student behavior, and realizes aspects of human tutors.

The first work in intelligent tutoring was undertaken to explore how a scientific theory of learning could be converted into an engineering theory for optimizing learning (Anderson et al., 1995). Some tutors were built as part of the development of the ACT\* architecture of cognition and learning, and later of the ACT-R architecture. A theory of the acquisition of cognitive skill had implications for instruction, and it was considered an important test of the theory if it could be used to optimize learning.

ACT-R distinguishes between declarative and procedural knowledge (Anderson, Bothell, Byrne, Douglas, Lebiere & Qin, 2004). The units of declarative knowledge are called “chunks” and represent things remembered or perceived. It may represent the fact that  $25 / 5 = 5$  or that Amsterdam is the capital of the Netherlands. Units of procedural knowledge are called “productions”. They are rules that fire under specific conditions and execute specified actions. Procedural knowledge is goal-oriented performance knowledge that can be executed efficiently. It is acquired through practice. Procedural knowledge is *implicit knowledge* that is not available to awareness whereas declarative knowledge is *explicit knowledge* that we are aware of in visual or verbal form (Dienes & Perner, 1999).

In 1993 ITS were evaluated in a High School setting (Koedinger, Anderson, Hadley & Mark, 1997). That project showed that laboratory tutoring systems could be scaled up and made to work in real settings like urban high schools.

The system that this thesis document is about is the first ITS designed for grade school students. It teaches single-digit division skills (like  $36 / 6 = 6$ ), which is typically learned in 4<sup>th</sup> grade. It covers all the problems of dividing a

two-digit number by a single digit; all the facts that are the inverse of the 10-by-10 multiplication table. When children learn mathematics in elementary school, multiplication and division are a good part of the mathematics curriculum in grade levels 3 to 5. A good understanding of multiplication and division is crucial at this level, since it is the basis of many mathematics skills that follow (Mathematics Learning Study Committee, National Research Council National Academies, 2001).

The design challenges for the ITS were to decide on the aspects of division that should be practiced. With “think aloud” sessions with 4<sup>th</sup> grade students we got a first hand feel of the strategies that grade school students use to solve division problems. The results of these sessions were combined with the theory of mathematics learning in grade school. This allowed us to construct a set of problems that would support the learning of the strategies we observed. In addition, we applied knowledge of how to make the students self-explain the strategies they used. Self-explanation has been shown to be beneficial to the learning process in other experiments (Aleven & Koedinger, 2002). The resulting thesis describes how to best leverage computer technology and how to apply tutor learning theory on the task of doing basic division (King, 2008). The contents of the design that are relevant here will be covered in later sections.

The aim of the study described in this Master thesis is to assess the effectiveness of this ITS. The ITS was designed to teach robust division skill by integration of declarative and procedural knowledge, and this document investigates to what extent the design has successfully accomplished this. After the student has worked with the tutor, he or she should be able to perform division operations fluently and with a deep understanding of division, so he or

she can transfer the division ability to new situations. It is also important to know whether the attitude towards division and mathematics have improved, and which aspect of the tutoring process has caused this.

## **2. Theory**

### **2.1 The ITS for Robust Division Skill**

ITS for division skill has an empirically informed design. It is founded in current practice of teaching fluent division in elementary school as prescribed by the Mathematics Learning Study Committee and the National Research Council National Academies. It was also based on the theory of mathematical proficiency. It was designed based on ‘think aloud’ sessions with fourth graders and educational theory on grade school mathematics.

The implementation is a flash program with four interfaces, two for fluent division and application of divisibility rules, and two for sense making of division and divisibility. We distinguished between ‘fluent division’ and ‘sense making of division’ for reasons which will be explained in coming sections. To make learning robust, the system lets the student practice fluency and sense making separately by presenting two types of division problems and divisibility questions.

The tutor interface, which is extensively described in King (2008), is structured around the multiplication table and the selection of divisibility rules. This is because in grade school, the learning of division facts is based on multiplication facts. The division learning process relies on known multiplication facts, and the table serves three functions: (1) it primes the multiplication facts in which the division facts are based, (2) it shows the



student that multiplication and division are inverse operations, and (3) it acts as an interface the student uses to make the solution strategy visible. In another interface the student answers a divisibility question and indicates which divisibility rule was used to arrive at the answer

## **2.2 Teaching arithmetic**

Children learn the basic operations of arithmetic in grade school. From prekindergarten through grade 12, they learn how to understand numbers, ways of representing numbers, and number systems (NCTM, 2003). They also learn the meanings of operations and how they relate to each other, learn to compute fluently, and make reasonable estimates. Computational fluency will be defined below.

People have an innate “number sense”, the ability to quickly understand, approximate and manipulate numerical quantities (Deheane, 1997). Deheane postulates that:

“Higher level cultural developments in arithmetic emerges through the development of linkages between this core analogical representation (the “number line”) and other verbal and visual representations. The neural and cognitive organization of those representations can explain why some mathematical concepts are intuitive, while others are so difficult to grasp.”

The distinction between mathematics concepts that are difficult- versus easy to grasp, has been described by other researchers too (Lebiere, 1999; Rittle-

Johnson & Siegler, 1998). The earlier mathematical development of children is often described as sophisticated. They have conceptual understanding in the domains of counting and simple addition. In contrast, school-age children are described as having impoverished conceptual understanding. It may be that some domains are “privileged” by evolution. Other researchers believe that the environment offers different frequencies of exposure observation and imitation (Rittle-Johnson & Siegler, 1998). Counting is often cited as an example. In any case, learning, and therefore teaching, arithmetic is a challenge. Lebiere notes that other arithmetic is formally a simple task, but hard for humans to perform. This in contrast to vision, for example, which is effortless but extremely complex. Lebiere:

"This suggests that human cognition at the most basic level embodies some assumptions about its environment that are at odds with the structure of arithmetic as it is taught."

There are a number of features of classrooms that contribute to computational fluency (NCTM, 2003). The less effective traditional methods involve two phases: The teacher's presentation of a topic, with students passively observing, followed by independent practice, with or without help. The more effective methods had three phases: First the teachers introduced the topic through explanation, questions, and discussion, and students were active learners whose initial knowledge was elicited. Second, over a long period the students were helped to move from a teacher-regulated to self-regulated solution processes. A significant period of help was gradually phased out. They

received sustained feedback on their performance, and had visual supports. The third phase of effective instruction was brief assessment of students' ability to apply knowledge to new problems.

The implications for computational fluency are that students had sustained, supported time to learn deeply and accurately. This is necessary for domains requiring multistep solutions. Effective practice is supported by monitoring and help that are focused on doing and understanding. In contrast, rote practice has little connotations of monitoring and lacks visual, conceptual and motivational support of the practice phase.

At present not enough is known about effective ways to orchestrate the helping period to deliver feedback and help students as they need it (NCTM, 2003). Designing and testing effective helping methods and effective ways to give feedback on answers is vital to further research. In the case of teaching computational fluency and conceptual understanding of division facts and strategies, the ITS presently under discussion is a suitable candidate medium for providing feedback and supporting the learning process (King, 2008).

The design of the ITS also supports many of the aspects of teaching and learning materials that help diverse learners. Diverse learners, as defined by Kameenui and Carnine (1998), are the students who may experience learning difficulties because of low-income backgrounds, language deficiencies, or other reasons. They identified six crucial aspects of teaching and of learning materials: (a) Focus on essentials and big ideas. (b) Make linkages obvious and explicit by organizing instruction with of graphic organizers, for example semantic webs and concept maps. (c) Prime background knowledge. (d) Provide

temporary support for learning. (e) Use conspicuous steps and strategies. (f) Review for fluency and generalization.

In the design document (King, 2008) is covered how these aspects and feedback are realized in the design of the ITS.

## **2.2 Teaching division**

Less research has been conducted on single-digit multiplication and division (concerning the dividends up to 100) than on addition and subtraction. What is known is that U.S. students go through an experiential progression of multiplication methods that is somewhat similar to addition (NCTM, 2003). Students learn count-on lists. For example, “Counting by 4s” is 4, 8, 12, 16 etc. They count up and down with their fingers representing the products. They can also start at a number they know and count from there, so they invent strategies to derive related products.

Division combinations are the inverse of related products. For example,  $30 / 6 = ?$  can be thought of as  $6 \times ? = 30$ . It is not clear whether children can be introduced to division-multiplication relationship very early, and learning practicing both at the same time, or whether they should learn the products first. Because in current practice they learn multiplication first, the tutor design follows this, and the division learning process relies actively on primed, if not known, multiplication facts. This happens by means of the multiplication table in the interface.

## **2.3 Robust Learning**

Robust learning means that a skill is learned deeply, which means being able to use it fluently and use it in new, unfamiliar situation. This is explained in the theoretical framework of the Pittsburgh Science of Learning Center (2006), for the remainder of this document referred to as PSLC.

In order to evaluate whether we achieved the goal of teaching the students robust division skill, we first need to understand when division skill is considered robust. Learning is robust if the acquired knowledge or skill meets at least one of the following three criteria (PSLC, 2006):

**Retention:** It is retained for long periods of time, at least for days and preferably for years.

**Transfer:** It can be used in situations that differ significantly from the situations present during instruction.

**Future learning:** it accelerates future learning. That is, when related instruction is presented in the future, this knowledge allows the student to learn more quickly and effectively.

We can also define deep, robust learning as the opposite of ‘shallow’ learning. We can interpret shallow learning with ACT-R theory (Aleven, Koedinger, Sinclair & Snyder, 1998). In ACT-R theory, learning a procedural skill means acquiring a set of production rules. Production rules are induced by analogy to prior examples. Superficial knowledge results when students pay attention to the wrong features in those experiences and examples. These features may be readily interpreted but do not connect to deeper reasons. Not much is known about what types of instruction are more likely to foster shallow learning (Aleven, Koedinger, Sinclair & Snyder, 1998).

Robust learning happens when the student “makes sense” of the material and builds fluency with application. In the context of mathematics teaching, “sense making” is here interpreted as the activity that produces “conceptual understanding”, as defined by the Mathematics Learning Study Committee and National Research Council National Academies (2001). Likewise, “fluency” is taken here to be identical with “procedural fluency”. Both these concept pairs recognize the distinction between explicit declarative knowledge, and implicit procedural knowledge. Therefore, it seems justified to let these concept pairs have the same meaning. Conceptual understanding and procedural fluency will be explained in the next section.

## **2.4 Strands of proficiency**

The Mathematics Learning Study Committee (2001) distinguishes five strands of mathematical proficiency that are strongly intertwined and develop together: *conceptual understanding*, *procedural fluency*, *strategic competence*, *adaptive reasoning* and *productive disposition*.

*Conceptual understanding* refers to an integrated and functional grasp of mathematical ideas. Students with conceptual understanding know why a mathematical idea is important and the kinds of contexts in which it is useful. Note that this strand is most relevant to robust learning: knowledge that has been learned with understanding provides the basis for transfer to new and unfamiliar problems. *Procedural fluency* is defined as knowledge of procedures, knowledge of when and how to use them appropriately and skill in performing them flexibly, accurately and efficiently. *Adaptive reasoning* refers to the capacity to think logically about the relationships among concepts and

situations. It includes knowledge of how to justify the conclusions. *Strategic competence* refers to the ability to formulate, represent and solve mathematical problems. *Productive disposition* refers to the tendency to see sense in mathematics, to perceive it as both useful and worthwhile, to belief that steady effort in learning mathematics pays off, and to see oneself as an effective learner and doer of mathematics.

We need to consider which of the strands are relevant to our case of a division ITS. Two categorizations of the strands seem relevant here: First, productive disposition and strategic competence are not much touched by the short time span that is covered by the experiment. These two strands are important, but can not be evaluated or taught directly. Second, a case can be made that all strands reduce to the first two: conceptual understanding and procedural fluency. These two strands are the most clearly defined in the literature through the much researched distinctions between declarative, explicit knowledge and procedural, implicit knowledge (Rittle-Johnson, B. & Alibali, M.W., 1999; Dienes, Z. & Perner, J., 1999).

Dienes and Perner apply the implicit-explicit distinction to knowledge representations. The clearest case of explicit knowledge of a fact are representations of one's own attitude of knowing that fact. Note that this is true for self-explanation of strategies and for retrieved facts, like division facts. They discuss the similar distinctions of conscious-unconscious, verbalizable-nonverbalizable, direct-indirect tests and automatic-voluntary control, and how these relate to models of learning, including the ACT-R theory. In ACT-R, all knowledge is modelled as either declarative or procedural. If we assume that all the kinds of "knowledge" that are referred to in the definitions of the strands can

be reduced to either declarative or procedural knowledge as modelled by ACT-R, then the five strands of mathematical proficiency are reduced to the first two: conceptual understanding is explicit knowledge of all the facts relevant to the task, and procedural fluency is implicit knowledge realised by the production rules for the perceived and handled facts. Strategic competence, adaptive reasoning and productive disposition are assumed to improve as a consequence of the intertwining with conceptual understanding and procedural fluency.

The other three strands are thus taken in the present context to be just less clear mixtures of declarative and procedural knowledge. They were assumed to be indirect properties of the learning process as it would be modelled in ACT-R. These reasons are only here covered, but they were also behind the intuitive judgement to not have the ITS support them individually. Therefore this argument is not mentioned in the design document (King, 2008).

## **2.5 Conceptual Understanding and Procedural fluency**

Conceptual understanding frequently results in students having less to learn because they can see the deeper similarities between superficially unrelated situations. Students with conceptual understanding have organized their knowledge into a coherent whole, which enables them to learn new ideas by connecting those ideas to what they already know. It is supported by helping the student make sense of the division process by having him or her explain the steps towards the solution, while they get immediate feedback on whether their explanation is correct.

As stated in the former section, procedural fluency refers to knowledge of procedures, knowledge of when and how to use them appropriately, and skill



in performing them flexibly, accurately, and efficiently. They need to be efficient and accurate in performing basic operations with whole numbers (6 + 5, 8 x 4, etc.) Fluency is a property of the learning process as a whole. It is intertwined with conceptual understanding (Mathematics Learning Study Committee, National Research Council National Academies, 2001). The solving process becomes more fluent when errors are first reduced and then eliminated, and the application of correct steps speeds up. Through proceduralization, steps can blend together which also speeds up the solving process. Fluency also applies to conceptual understanding, because the recalling of understood concepts is also proceduralized.

In the case of this ITS for division skill, robust learning means that students can fluently solve a problem of dividing a number equal to or below 100 by a number equal to or below 10, and can use the division facts when solving novel mathematics problems. In this context “procedural fluency” simply means reaching the correct answer fast, without making mistakes in the process, like clicking the wrong button or asking for a hint. The tutor was designed to support the two intertwined strands of mathematical proficiency of fluency and conceptual understanding. If the ITS has an effect, the students’ division proficiency improves by practice. Then he or she ends up with a set of production rules that manipulate declarative facts to produce a correct answer quickly, without making an error. The question is whether the learning gains produced by working with the ITS learning are robust, and which features of the tutoring process are responsible for the learning gains.

Conceptual and procedural knowledge can have four possible relations (Rittle-Johnson & Siegler, 1998):

1. Procedural knowledge develops before conceptual knowledge.
2. Procedural knowledge develops after conceptual knowledge.
3. Procedural knowledge and conceptual knowledge develop concurrently.
4. Procedural knowledge and conceptual knowledge develop iteratively, with small increases in one leading to small increases in the other, which trigger new increases in the first.

Current theory has only addressed the first two, because repeated assessments from beginning to mastery. Rittle-Johnson and Siegler focus on five mathematics domains that have been studied to draw conclusions regarding which concepts precede which procedures and vice versa, but unfortunately for our present case, division was not one of these domains.

## **2.6 Self-explanation**

Self-explanation has been shown to be a beneficial strategy in the learning process (Aleven & Koedinger, 2002). It fosters learning with understanding. In many forms of instruction, it is difficult to ensure that students get a deep understanding of the material and avoid the learning of shallow heuristics. It has been shown that self-explanation, supported by means of intelligent instructional software, can help towards this goal and can enhance learning in actual classrooms (Aleven & Koedinger, 2002).

The greater understanding that resulted from self-explanation seems due to qualitatively different knowledge. It can be explained in terms of more integrated visual and verbal declarative knowledge, used more reflectively, and less shallow procedural knowledge. In terms of learning processes, the act of explaining problem-solving steps appears to help

students to integrate two modes of learning: implicit visual induction and explicit verbal

knowledge acquisition (Aleven & Koedinger, 2002).

Although these results were achieved with high school students using the PACT geometry tutor, we can expect the results to generalize to grade school students when they use a division tutor. This is because conceptual understanding of division is constituted by declarative knowledge of division and multiplication facts and divisibility rules, just like geometry. Earlier we saw how conceptual understanding and procedural fluency was reduced to facts and productions in ACT-R. ACT-R describes how memory for facts is activation based, and how it is connected to related facts through productions. Thus, declarative knowledge of division is will to be strengthened by reflection on the material, just like knowledge of division. The reflection is encouraged by the self-explanation of rules and solution steps.

The PACT geometry tutor was enhanced to combat shallow learning (Aleven, Koedinger, Sinclair & Snyder, 1998). Some of those features were used to enhance the design division tutor (King, 2008). Students who worked with the geometry tutor were better able to provide numerical answers than to articulate the reasons that are involved in finding these answers. This suggests that students may provide answers using superficial and unreliable visual associations. Therefore, in a newer version of the tutor students were required to state the reason for their answers. The corresponding features in the division tutor are the explicit indication of the divisibility strategy used (for answering a divisibility question), and the explicit indication of the divisor, dividend and

quotient, after completing a division problem. For a complete description of the interface, see King (2008).

Grade school students will have less developed self-explanation ability, but the learning material is also proportionally simpler. In the case of division, explicit verbal knowledge is the list of divisibility rules and the concepts of dividend, divisor and quotient. The multiplication and division facts will integrate with the perceptual pattern knowledge students get when they click on the multiplication table as part of the solving. The solution strategy is made explicit, and is corrected if it is wrong. The rules and facts connect with the table, the divisor, dividend and quotient. This will yield a deeper conceptual understanding of division and divisibility.

The activities of self-explanation will be referred to as “sense making” for the remainder of this document. This is because the activity of self-explanation supports conceptual understanding, but these words don’t emphasize the process of acquiring declarative knowledge. “Sense making” means working towards conceptual understanding by means of self-explanation.

## **2.7 Speed and repetition**

Given the above definitions of fluent division, we can see it as a path to robust learning of division. The declarative rule facts of division are consolidated by the repeated practice of self-explanation. The student “makes sense” of division. The division facts (like  $25 / 5 = 5$ ), are consolidated by repeated references to their position on the multiplication table. The student must act to solve the problem, and thus all the steps are proceduralized by repetition.

In terms of ACT-R, new production rules are learned on the basis of existing rules and declarative knowledge (Taatgen, 2004), by a mechanism called “production compilation”. The process by which children learn division was explained above. It is easy to see this process in terms of production compilation. The student starts out with known or primed multiplication facts, and initial rules of division from first instruction. Such rules may concern regularities in the multiplication table (like the rule that all number ending with a 0 are divisible by 10 and 5, and those ending with a 5 by 5). In the case of the division ITS, these rules are explicitly trained and supported. By production compilation the rules and multiplication facts (in the table or from memory) fuse until all division facts are retrieved as independent facts.

There is also an ACT-R model to account for arithmetic in general (Lebiere, 1997), that by gradually raising the activation of the necessary facts with practice, provides a general account of the transition from general problem-solving strategies toward more efficient ones.

When all facts are proceduralized, the learning process can still continue. Since declarative facts have an activation function that makes retrieval faster as a fact is referenced more often, the student of division can theoretically train until all facts are recalled instantly. This is the main argument for assuming possible learning gains for all students, even the those that are already proficient. However, it is likely that learning will suffer because of lesser motivation after a level of proficiency is reached and returns on further practice start diminish.

When practicing with the tutor, the student progresses to unsupported retrieval. Initially errors are corrected by the tutor and are thus reduced in the

next problems to be solved. All the facts and processes of division are consolidated by repeated practice. The student will get robust knowledge of single-digit division over time. With the division ITS, the student is encouraged to develop fluency, therefore speed can be promoted by feedback and timing. The problems are served in ‘bundles’ of twelve. The student sees the number of problems solved without error, and on the time spent on the current ‘bundle’.

## **2.8 Summary**

This thesis describes the evaluation study of an ITS for division skill, developed at the PSLC. We need to know if the ITS is effective, and if so why. The design of the ITS is based on the distinction between declarative and procedural knowledge. Declarative knowledge is explicit and is also referred to as conceptual understanding. In the context of division, it means knowing the rules that are used to solve a division problem, and knowing division and multiplication facts. The learning of declarative knowledge is supported by the ITS by making the students explain the rules and facts to themselves. The ‘making sense’ of division strategies. This is done by making the solution strategies visible.

Procedural knowledge is implicit and is referred to as procedural fluency. The ITS teaches fluency by repetition. This consolidates the production rules of the solving process. Speed is promoted by feedback and score and time.

The ITS for robust division skill was designed to tutor the students on the following aspects of the division process.

1. To understand that multiplication and division are inverse operations.
2. And so be able to use multiplication facts to solve division problems fluently.
3. To understand divisibility rules.
4. To use divisibility rules fluently.

When the student has conceptual understanding of division and applies it fluently, with long-term retention, then the learning has been robust. Robust division skill should accelerate future learning and should enable the student to transfer division skill to new situations. This thesis describes the experiment that shows whether this has happened, besides the measurement of the learning gains in single-digit division skill.

### **3. Research question**

The two strands of proficiency are both important to robust learning. But in the case of tutoring division with an ITS, their relative importance is unclear. Conceptual understanding and fluency are obviously both important, but we don't know their roles and relative importance to the learning gains.

The research question the study attempted to answer is:

Q: Which educational technology is most effective in teaching 4th graders robust division skills: An Intelligent Tutoring System that focuses on fluency by repetition only, or an Intelligent Tutoring System

that combines fluent repetition with making sense of the material, by making strategies visible and training the student in self-explanation.

Obviously, there is a trade-off between the number of problems a student can solve in a set amount of time, and the depth with which these problems are understood. When the task only consist of retrieving the answer from memory, or retrieving a few steps towards the answer, then the problem will be solved quickly. In contrast, when every step towards the solution must be made explicit, then the problem takes longer to be solved. The question is whether this extra time is well spent. If making the solution strategies explicit by self-explaining makes the student reflect on the material and understand the relationships among facts, then conceptual understanding will develop faster. But when more problems are solved in the same amount of time, the process of proceduralisation to direct retrieval will likely develop faster.

Which aspect must be promoted more? Should the student just practice as many problems as possible, or should he or she also spend time making sense of the material to develop conceptual understanding? All students who become proficient at division reach the same end state: proceduralization of multiplication and division facts. The question is whether the most efficient route to complete proceduralization is maximal repetition of solution steps, or deeper conceptual understanding by self-explanation, mixed with less repetition. The goal of this evaluation study was to answer that question, or at least gain insight in the trade-off between depth and quantity of practice. Since practice time is the same for each condition, any difference can be attributed to the



specifics of that condition: many references to division facts and production rules in the case of the Fluency condition, and a more efficiently connected network of facts and rules in the case of conceptual understanding.

### **3.1 Hypotheses**

We expect a combination of reflection and speed to work best. If conceptual understanding and proceduralization are both supported, then both pathways to robust knowledge are travelled. The hypothesis is that this produces better results than repetition only, even when this means that not as many division problems will be solved in the same amount of time. In short: Alternating repetition with making sense of problems by self-explaining them, is expected to be more beneficial than repetition alone.

The research question translates into the following set of hypothesis:

1. There are larger learning gains when Sense Making alternates with Fluency only, vs. Fluency only, for a fixed amount of time, with the same types of problems, but potentially a different number of problems per type.
2. The Sense Making + Fluency condition will have more robust learning gains.

In order to test these hypotheses, we conducted a number of corresponding tests.

### **3.2 Operationalizations**

4th graders using the Sense making + Fluency ITS for a total of 120 minutes score significantly better than students using the Fluency ITS for 120 minutes on a:

1. Test for division skill.
2. Transfer test for division skill.
3. Test for motivation

And conversely:

4th graders using the Fluency only ITS for a total of 120 minutes score significantly better than students using the Sense Making + Fluency ITS for 120 minutes on a:

4. Test for fluency.

## **4. Method**

### **4.1 Participants**

The sample included 16 male and 16 female fourth-grade students from a suburban elementary school in Pittsburgh. The experiment took place near the end of the school year, and division was already covered in this class. However, the interviews and assessment related to the tutor design process suggested there was room for improvement. The retrieval times of division facts was generally longer than that of higher grade students, even with the two students that where

indicated by the teacher to by most proficient. The students were deliberately selected from the whole range of proficiency, as judged by their teacher. The levels of the students were revealed to the researcher after the sessions.

Thus we assume that the individual differences are big enough to expect learning gains on average after additional practice with the ITS. Also, all the six students who participated in the “think aloud” sessions, even the more proficient, did not execute all the steps perfectly, which indicated that even they could become more fluent with additional practice. Although we obtained no data on the division proficiency of the whole class, we could roughly estimate the average level of proficiency from the results of the two students of middle-level proficiency, who would clearly benefit from both a deeper understanding of the division process and more practice of procedural fluency. This was later confirmed by the test results, which indicate a fairly normal distribution of proficiency. This assumes that the teacher based his judgment of their level of proficiency sufficiently enough on their grades for arithmetic or division. This was the tacit assumption when the teacher was asked to categorize the students.

## **4.2 Conditions**

The students were randomly assigned to one of two conditions: (a) Fluency only, and (b) alternating Fluency and Sense Making. There were 13 students in the Fluency condition, and 16 in the Sense Making + Fluency condition. The experimental data of three students had to be rejected because of a technical problem, so the final sample consisted of 29 children. The pre- and posttest and experimental sessions were done in the school computer lab during mathematics class.

The ITS is made of 4 interfaces in total, which are presented to the student after each other, each serving a bundle of 12 problems. The interfaces are numbered 1, 2, 4 and 5. Interface 3 was similar to Interface 1 and was omitted in this experiment. The ITS was split up according to condition. The four interfaces each support a different combination of division proficiencies (Fluency and / or Sense Making) and a learning component (divisibility or division).

*Fluency only condition:* The interfaces 4 and 5 focus on fluency only. They facilitate and emphasize speed, because that is identical to fluency in the case of simple division problems. Fluency is further supported by actively encouraging the student to work faster. The problems are served in ‘bundles’ of 12. When these 12 problems are solved, interfaces are switched and a new bundle of problems is presented. To keep the solving urgent and challenging, the score of the number of correct problems is shown on the screen, and a timer shows how long the student has been working on the current bundle. This feedback enables the student to monitor and improve their performance, even when the answer is known quickly and being correct may no longer be challenging in itself. The features of scoring and timing are promoting fluency in this condition, as opposed to sense-making. See Figure 2 in section 4.3 for screenshots of the fluency interfaces.

*Sense Making + Fluency condition:* The division learning is based on prior knowledge of divisibility rules and multiplication facts, and the strategies that draw on this knowledge are supported by Interface 1 and 2. Multiplication facts

and divisibility strategies each have their respective support feature. Multiplication facts are made available for reference by the multiplication table. When a student solves a division problem, he or she can rely on the inverse multiplication fact when necessary, and a list of divisibility rules are available in Interface 2 that tutors divisibility. See Figure 1 in Section 4.3 for screenshots of these interfaces. In this condition, all interfaces are used. After a bundle of twelve problems the next interface is presented with a new bundle of problems. In this condition, the interfaces 4 and 5 have no scoring and timing. With Interfaces 1 and 2 the student must make sense of the invoked divisibility rule by self-explanation. In Interface 1 the student must self-explain by indicating the solution steps towards solving a division problem. This is done by clicking the divisor, dividend and quotient in the multiplication table, making the solution strategy visible.

Take note that interfaces 4 and 5 (the Fluency interfaces) present the same type of problems as Interfaces 1 and 2, but have no multiplication table and no list of divisibility rules. Students reach the same types of answers in each interface, but the difference is that Interfaces 1 and 2 make the student reflect longer on fewer problems, and interfaces 4 and 5 maximize the number of problems solved.

## **4.3 Procedure**

### **4.3.1 Experiment**

#### **Session 1: pre-test**

The pre- and post-test for measuring the learning gains was a new implementation of a computer-based test developed to evaluate a game that also teaches division (Habgood, 2007). The game is described below. The test was

specifically designed to avoid ceiling effects, in order to be able to measure the learning gains of students who were already somewhat familiar with division. In his own first evaluation of the game, Habgood encountered ceiling effects because the level of proficiency of the students was underestimated. Therefore, we expected this ‘tried and proven’ test to be a suitable tool for measuring learning gains with the ITS. It would also enable us to compare the gains with the learning gains of the students who used the game in

The original pre-post test had a theme that resembled the look and feel of the game that was used in Habgood’s experiments. The game will be described below. In our implementation of this test the game theme (pictures, colors) was removed, leaving just a basic interface and the response buttons.

The test consists of 63 multiple choice questions with four options in each case: 1 correct answer + 3 distracters. The first three questions are non-assessed practice questions designed to ease students into the test, for example:

Select the number of legs that a dog has:

4, 5, 6 or 7

The remaining 60 are division questions equally comprised of two recurring formats, either asking the child to select the divisor that divides a given dividend (dividend-based) ; for example:

Select one number that 45 can be divided by:

4, 6, 9 or none of these

Or to select the dividend that can be divided by a given divisor (divisor-based) ;  
for example:

Select one number that can be divided by 5:

35, 13, 29 or 41

There is an equal balance of divisor-based and dividend-based questions among the non-conceptual question in the test. These are comprised of five questions on each divisor from 2 to 10, excluding the divisor 7 as it was not included as part of the games' learning content, which was embedded in action. The game avatar is a Greek hero who destroys zombies by hitting them with a weapon. Each weapon corresponds to a divisor, and the zombies wear a number that is the dividend. If the correct weapon is used, the zombie crumbles. For some reason there was no weapon corresponding to 7, perhaps because this divisor is never used in combination with others. For example, a zombie wearing the number 24 can be successfully attacked with either 2, 3, 6 or 8 after which it splits in the remaining divisors. Anyhow, because 7 was not part of the content, it was not part of the test.

Another five questions are added where the answer was "none of these". Fifteen conceptual questions are also included in the test. Three of these are included to test for knowledge of the heuristic patterns associated with numbers that divide by 2, 5 and 10. The remaining twelve questions test for an understanding of relationships between divisors, or for applying rules outside of

normal limits (i.e. dividends greater than 100). The order of the questions is randomized, but remains consistent between subjects and between tests (Habgood, 2007). The time is not displayed on the screen and no feedback is provided on the choices made.

The first session was used for pre-testing. Prior to testing, the students were told that they were going to help the experimenter with finding out how to make the tutor better at teaching division. They were also told that they were first going to be tested on how good they were at division. They were instructed to work as fast as they could and try to get as many answers right as possible, and not to stay long with a question if they did not know the answer.

### **Sessions 2 and 3: experimental sessions**

The practice sessions with the ITS were divided over two days and three sessions of 40 minutes, summing up to a total of 120 minutes practice time. The practice took place in the school's computer room. First the researcher explained how the tutor interfaces worked, showing a few example problems on a projection screen. When there were no more questions, the students were instructed to log in using their familiar student number. At this point the server of the ITS assigned each student to a condition. All students started practicing when the timer was started.

The students in the Sense Making + Fluency condition worked alternately on the interfaces, which are shown in Figure 1. When 12 problems were completed, the other interface would load with a new series of problems.



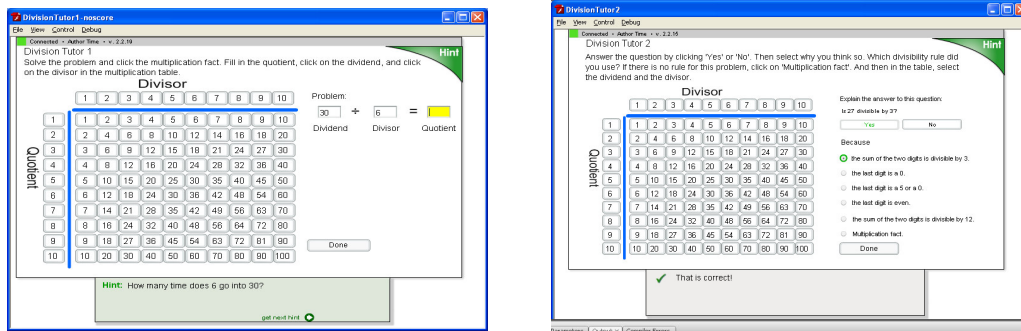


Figure 1. The panel at the left shows the interface focusing on ‘making sense’ of division. The student clicks the dividend, divisor and quotient on the table, and enters the quotient in the answer field. The panel at the right shows the interface focusing on ‘making sense’ of divisibility rules. The student must select the rule that was used to answer the question, or when no rule applies, he or she must click the dividend, divisor and quotient in the table. The ITS provides feedback through a panel that slides down when an error is made, or when a hint is requested, or when the answer is correct.

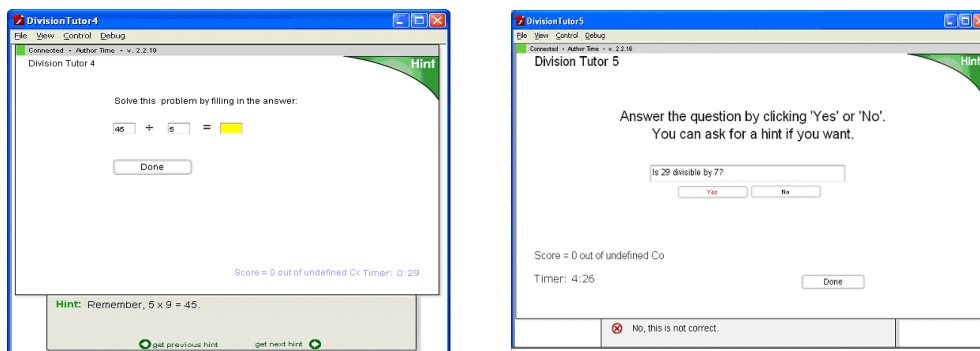


Figure 2. At the left the interface focusing on fluent division. There is no multiplication table to help find the solution, or to self-explain the steps of the solution. The right panel shows the interface focusing on fluent application of divisibility rules, without selecting the used rule. Scoring and timing are displayed at the lower half of the interfaces. Note that in these screenshots there is no number because the interface is not running on a server. Also, the text is in gray, while it is in black in during the actual practice sessions.

#### **Session 4: post-testing**

The post-test was identical to the pre-test, but was preceded directly by a ‘Near’ Transfer test, which is explained below.

##### **4.3.2 Transfer tests**

There were two tests of transfer of division skill to new situations.

1. “Near” Transfer test: This test took 5 minutes, and consisted of 12 problems from the practice session of the Sense Making + Fluency condition. Both conditions did the same problems. With this test we measured differences between the two groups when they self-explained solution steps. Both groups got the same instruction on how to use the interfaces, but the students of the ‘Fluency only’ condition were not presented with the Sense Making interfaces during the practice sessions. Thus we could test how well they generalized the fluency practice to the ‘new’ situation of self-explaining and reflection.

2. Transfer test: When students practice division facts with different strategies and reflect on the material, as when using the ‘sense making’ interfaces, they are expected to be able to transfer their skills to new situations. If deep learning has occurred when using the ‘sense making’ interfaces, then the students can use their learned division facts to solve new, more complex problems that contain the learned division facts.

To test this, each student did a transfer test with items that test whether the knowledge of division can be used in a new context. If the student can apply the knowledge flexibly they can answer questions like:

“Suppose you have a number that is divisible by 3. And suppose you add 12 to that number. Do you know if this new number can be divided by 3? Why or why not?”

“Is there a number that 53 can be divided by? If yes, what number?”

“What is the smallest number that can be divided by 3 and by 8?”

#### **4.3.3 Fluency test**

The test on fluency was a sheet of paper with 100 division problems on it of the type  $25 \div 5 = 5$ . The students were familiar with this notation. Their score was the number of problems they could complete in 2 minutes. They were instructed to work as fast as possible.

#### **4.3.4 Motivation questionnaire.**

The motivation test was a questionnaire with 5 point Likert-scale items. There were two categories of questions: self-reported division ability, and liking of the tutor interfaces they worked with. Examples are:

“I am good at math.”

“I am not good at division.”

“I don’t like math.”

“I like the division tutors I have worked with.”

“I like the tutors more than my favourite board game.”

## **5. Results**

### **5.1 Learning gains**

The strategy for examining the learning gains was taking the mean and standard error of the test results, and evaluated the mean differences with a  $t$  significance test. See Figure 1. In the case of a two-sample  $t$  procedure, the distribution is approximated with an approximation for the degrees of freedom  $k$ . (Moore & McGabe, 2004). Values are computed with software that approximated the degrees of freedom from the data. The numbers indicated however is  $k$  equal to the smaller sample size  $- 1$ , which is also a valid approximation. We use that value here because the software does not return the actual value. The difference could only skew our conclusions if any of the results were marginal, which is not the case.

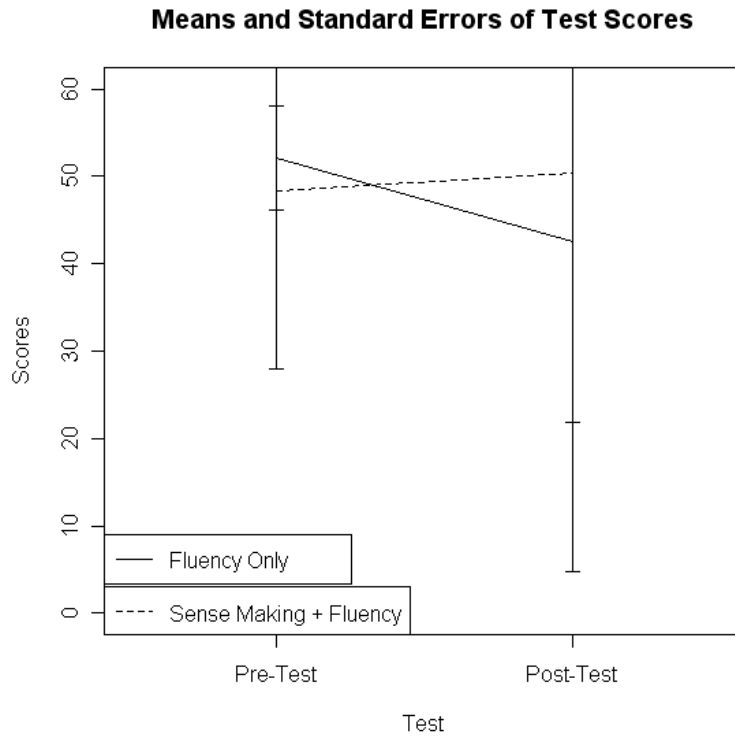


Figure 1: Means and Standard Errors

In the Sense Making + Fluency condition the standard errors for both the pre- and post test are large, and the standard error of the post-test in the fluency condition is even larger. We can see that there are no learning gains; in the Fluency condition the ‘gains’ are negative and significant:  $t(12) = 3.22$ ,  $p = 0.0074$ . Students in the Fluency condition are clearly in worse shape than when they started. The pre-test differences between the conditions are not significant,  $t(12) = 1.89$ ,  $p = 0.0837$ . But the post-test differences between the conditions are significant with  $t(12) = 0.96$ ,  $p = 0.0357$ . The Sense Making + Fluency gain are far from significant with  $t(15) = 1.13$ , because the two high SE’s.

Note the maximum score is 60; the average scores are quite high. The students turned out to be more proficient at division than we initially expected.

This is surprising, since the test was explicitly designed to avoid ceiling effects that were encountered during its development (Habgood, 2007). Nonetheless, there are enough lower scoring students, which also shows in the high standard error. They still had room for improvement. We can also look at the lower scores for clues to the cause of lack of learning gains. The ceiling effects in the data arise from the inability of the test to measure proficiency accurately. The closer a score is to the ceiling, the larger the probability that the ceiling has influenced the score. Therefore we can expect the lower half of the scores to be a closer to the true level of proficiency of these students. Thus, the learning gains of this subset of students can only be attributed to the ITS. The question remains whether this smaller sample will allow us to see the effect at all.

If we divide both conditions at the median and only look at the lower half of the pre-test scores, we can see the results for the students who still had some division skill to gain before the experiment. Because the smallest of the two sample is only 6 after the split, we will have only 5 degrees of freedom to find any effects. See figure 2. The format is the same as Figure 1.

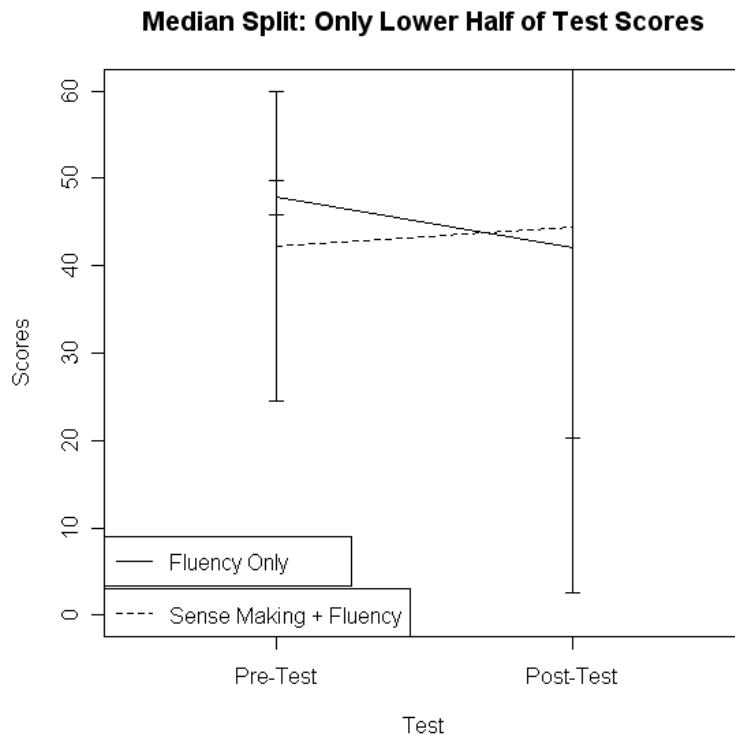


Figure 2: Median Split

The direction of the learning gains are similar to Figure 1, just like the and the pattern of the standard errors (SE). In the Fluency condition the post-test SE is lower, and the SE of the Sense Making + Fluency is higher.

Because only half the students are included, there are problems of sample size; the difference between the pre-test scores is significant with  $t(5) = 3.16$ ,  $p = 0.025$ . The lower scoring half of the Sense Making + Fluency condition started out already less proficient than the other students. The extreme SE of the post-test makes it impossible to draw any conclusion from these data. The negative gain in the Fluency condition are not significant  $t(5) = 2.49$ ,  $p = 0.055$ , but it is close to it. Because of the low number of degrees of freedom we can speculate that the result would have been significant with a larger sample. In that case the lower scoring students' performance has is diminished with

practice. They are more likely to be affected by boredom and fatigue than the lower scoring students. If this is really the case, then this could have implications for the ITS practice. Not-so-good students may have to be tutored differently, for example with a longer series of shorter sessions.

## 5.2 Fluency test and Transfer test results

We hypothesized that the Sense Making + Fluency condition would score highest on each test except the fluency test. However, none of the differences between means are significant. See Table 1.

Table 1

Means and *p*-values of the Fluency Test and Transfer Tests

Condition	Fluency Test	Transfer Test	Near Transfer Test
Fluency only	39.6	9.2	10.9
Sense Making + Fluency	33.5	8.4	10.9
<i>p</i> -value	0.156	0.123	-

A median split of these test scores yielded no significant result either.

## 5.3 Motivation results

As explained in section 4.3.4, one of the item categories measured the student's liking of the tutor interfaces he or she used. All the answers were aggregated and compared by conditions. The mean of the Fluency condition is 3.2. The mean of the Sense Making + Fluency condition is 3.5. The difference



between them is not significant, indicating that the students don't prefer one condition over the other.

## **6. Discussion and Conclusion**

Taken together, the findings provide no support for the hypothesis that self-explanation combined with fluency practice on an ITS is beneficial to division proficiency. The results either show no effect, or an effect in the wrong direction. Therefore, we must reject the hypothesis that self-explanation produces superior results with the conditions of this experiment.

There is insufficient data to determine the cause of the lack of evidence. We don't know whether the ITS simply does not work, or it does, and we can not detect the learning gains. There are several possible explanations for these findings:

1. High starting proficiency of the subjects.
  2. Practice with the ITS is not beneficial under any condition.
  3. Practice with the ITS is beneficial, but not at this intensity and duration of practice.
  4. This ITS design in general are not useful in a grade school setting.
- 
1. Possibly the ITS does improve division proficiency, but only up to a point. Then the lack of learning gains means that this group of students had already reached that point on average. If the starting level of proficiency had been lower, the students could have reached their individual maximum point of proceduralization by using the ITS. This

can be tested by repeating the experiment with a group that is less proficient, for example testing earlier in the school year, or in a class of an earlier grade.

2. Possibly the design of the ITS is flawed. But it seems unlikely that actions that resemble normal practice so much are of absolutely no use. Both the fluency interfaces and Sense Making interfaces contain only elements the student is familiar with from earlier division instruction.
3. Possibly the ITS works in principle, and students can learn from it, but not under these conditions. Maybe the pressure from scoring and timing is distracting from the practice, and maybe self-explanation is not identical with making strategies visible, in a division context.
4. Possibly grade school students are not suited for lengthy training sessions on an ITS. During the experiment we noticed fatigue and boredom, and after the last practice trail the teacher commented that they were tired. This could be fixed by using shorter practice trails.

A future experiment will have to decide which of the 4 explanations is correct. At present, no. 1 or no. 4 seem the most plausible. Drawing any further conclusion from these results is speculative. To learn more about the actual differences between Fluency and Fluency + Sense Making, another experiment must be conducted. The large standard errors suggest that a larger sample size is not the remedy. First we must fix the conditions to make conclusions possible.

Ideally, you would have students tested after their first division instruction, have them practice with the ITS for two sessions in stead of three, and then conduct the post-test.

Then, after classroom instruction on the basic division curriculum is completed, repeat the experiment. Then we can see what the initial benefits of the ITS are, and possibly draw conclusions on whether there is a ceiling of proficiency that is reach with ITS practice, while not being the ceiling of proficiency for a specific student in an absolute sense. Until this experiment is conducted we can not draw interesting empirical conclusions about this first ITS for grade school students.

## **7. References**

Aleven, V., Koedinger, K. R. (2002). An effective metacognitive strategy: Learning by doing and explaining with a computer-based Cognitive Tutor. *Cognitive Science*, 26, 147-170.

Aleven, V., Koedinger, K.R., Cross, K., (1999). Tutoring Answer Explanation Fosters Learning with Understanding. *Proceedings of the 9th International Conference on Artificial Intelligence in Education*, AIED '99, 199-206.

Aleven, V., Koedinger, K.R., Sinclair, H.C. & Snyder, J., 1998. Combatting shallow learning in a tutor for geometry problem solving.

Retrieved

from:

<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.34.6307>

Anderson, J.R., Bothell, D., Byrne, M.D., Douglas, S., Lebiere, C. & Qin, Y. (2004). An Integrated Theory of the Mind. *Psychological Review*, Vol. 111, No. 4, 1036–1060

Anderson, J.R., Corbett, A.T., Koedinger, K.R. & Pelletier, R. (1995). Cognitive Tutors: Lessons Learned. *The Journal of the Learning Sciences*, (4)2, 167-207

Chi, M. T. H., Bassok, M., Lewis, M.W., Reimann, P., & Glaser, R. (1989). Self-Explanations: How Students Study and Use Examples in Learning to Solve Problems. *Cognitive Science*, 13, 145-182.

Corbett, A. T., Koedinger, K. R., & Anderson, J. R. (1997). Intelligent tutoring systems. In Helander, M. G., Landauer, T. K., & Prabhu, P. V. (Ed.s) *Handbook of Human-Computer Interaction*, (pp. 849-874). Amsterdam, The Netherlands: Elsevier Science B. V.

Deheane, S. (1997). The Number Sense: How the Mind Creates Mathematics. *Oxford University Press US*.

Dienes, Z. & Perner, J. (1999). A theory of implicit and explicit knowledge. *Behavioral and Brain Sciences*, 22 (5).

Hiebert, J., & Carpenter, T. (1992). Learning and teaching with understanding. In D. Grouws (Ed.), *Handbook of research on mathematics research and teaching*. (pp. 65-100). New York: MacMillan.

Kameenui, E.W. & Carnine, D.W. (1997) *Effective Teaching Strategies that Accommodate Diverse Learners*, Prentice Hall.

King, S.T. (2008). *Designing and Implementing an Intelligent Tutoring System for Robust Division Skill*. Bachelor thesis, University of Groningen.

Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education*, 8, 30–43.

Mathematics Learning Study Committee, National Research Council National Academies, (2001). *Adding It Up: Helping Children Learn Mathematics*, National Academies Press.

National Council of Teachers of Mathematics, (2003). A research companion to *Principles and Standards for School Mathematics*.

Moore, D.S. & McGabe, G.P. (2004). *Introduction to the practice of statistics*. W.H. Freeman and Company.

Pittsburgh Science of Learning Center. (2006). PSLC Theoretical Framework.

Retrieved

from:

[http://www.learnlab.org/clusters/PSLC\\_Theory\\_Frame\\_June\\_15\\_2006.pdf](http://www.learnlab.org/clusters/PSLC_Theory_Frame_June_15_2006.pdf)

Rittle-Johnson, B. & Alibali, M.W. (1999). Conceptual and procedural knowledge of mathematics: Does one lead to the other? *Journal of Educational Psychology*, 91, 1-16.

Simon, H. A. (1987). The information-processing explanation of Gestalt phenomena. In *Models of Thought, Volume II* (pp. 481–493). New Haven, CT: Yale University Press. (Reprinted from *Computers in Human Behavior*, 2, 1–15, 1987.)

Taatgen, N.A. (2004). Production compilation: a versatile cognitive mechanism. *AISB Quarterly* p. 7.

# DESIGNING AND IMPLEMENTING AN INTELLIGENT TUTORING SYSTEM FOR ROBUST DIVISION SKILL

Stefan King, s1234668, [stefan@ai.rug.nl](mailto:stefan@ai.rug.nl)

**Abstract:** This bachelor thesis describes the research behind the design and implementation of an Intelligent Tutoring System (ITS) for robust division skill. Literature on math instruction and tutor design informed the general approach for designing this ITS. The details of the tutor design were based on a qualitative empirical study on division operations. A group of 4<sup>th</sup> grade students was assessed on the use of division and divisibility strategies to see how the ITS could support the learning of these strategies. The resulting ITS is a flash program with four interfaces, two for fluent division and divisibility, and two for sense making of division and divisibility. To make learning robust, the system lets the student practice fluency and sense making separately by presenting two types of division problems and divisibility question.

## **1.Introduction**

Computer technology can be used in schools to aid instruction. Intelligent Tutoring Systems (ITS) have been developed to aid instruction on various domains, including math (Anderson, Corbett, Koeding & Pelletier, 1995). An ITS is a computer system that provides direct customized instruction or feedback to students. It can employ different methods for feedback and there is flexibility in the way learning material can be presented. Another way to use computers for instruction is to let the student play an educational game. Educational computer games are designed to teach about a certain subject and assist the student in learning a skill as they play. These games have a different potential as an educational technology because they have more possibilities for “fun”, and the aim in designing them is that enjoyment supports and promotes the learning process by increased motivation (Habgood, Ainsworth, & Bensford, 2005). On the other hand, an ITS can present problems more efficiently because practice is not diluted by game elements. The question is which educational technology is the most effective: Educational computer games or Intelligent Tutoring Systems.

In order to compare the effectiveness of the two technologies within the domain of math, we need to measure the learning gains of two separate groups of students, each group using one of the technologies, and then compare their learning gains. The comparison is only valid when the subject matter learned by the students is the same for each educational technology. Studies have already been done to measure the learning gains of an educational game that supports learning of fluent division skill (Habgood, 2007). We can compare the effectiveness of each technology if we also have an evaluation study on the learning gains achieved with an ITS for division skill. A study that evaluates both technologies as separate conditions is also possible then. This thesis describes the research and design choices for such an ITS, and the results of the evaluation study are discussed elsewhere (King, 2008).

The design challenges for the ITS were to deciding on the aspects of division that should be practiced, and how the learning process of division problem solving should be supported. The question was how to best leverage computer technology and apply tutor learning theory on the task of two-digit division. There are different components in the learning process of math skill, and educators are divided on what should be emphasized when and in what order.



## 2. Theory

While there is not much research done yet on how this kind of division is learned (National Council of Teachers of Mathematics, 2003), there is a good theoretical understanding of mathematical proficiency in general, and educators have clear learning objectives for elementary students.

The math curriculum of 3-5 graders includes multiplicative reasoning, equivalence and fluency. They must understand the situations in which multiplication or division is an appropriate operation, and having efficient, accurate and generalizable methods for computing that are based on properties and number relationships. Other components include understanding the uses and methods of computation. Students who understand the structure of numbers and relationships among numbers can work with them flexibly. The learning materials should focus on the meanings of, and relationship between, multiplication and division. Computational fluency, the ability to solve a problem without hesitation or error, is one of the strands of developing mathematical proficiency (NCTM, 2003).

### 2.1 Intertwined stands of proficiency

Computational fluency (or *procedural fluency*) and conceptual understanding (or *sense making*) are two of the five intertwined strands of mathematical proficiency. They are interwoven and interdependent in the development of proficiency in mathematics (Mathematics Learning Study Committee & National Research Council, 2001). The challenge for educational technology is to accommodate these math education requirements in the learning process to optimize learning results, by balancing trade-offs that may exist between different math skills.

*Conceptual understanding* refers to an integrated and functional grasp of mathematical ideas (MLSC & NRCN, 2001). Students with conceptual understanding have organized their knowledge into a coherent whole, which enables them to learn new ideas by connecting those ideas to what they already know. *Procedural fluency* refers to knowledge of procedures, knowledge of when and how to use them appropriately, and skill in performing them flexibly, accurately, and efficiently. In the domain of number, procedural fluency is especially needed to support conceptual understanding of place value and the meaning of rational numbers.

Procedural fluency and conceptual understanding are often seen as competing for attention in school mathematics. But pitting skill against understanding creates a false dichotomy. As we noted earlier, the two are interwoven (Rittle-Johnson & Alibali, (1999). Research indicates that these two strands of proficiency continuously interact (Hiebert & Carpenter, 1992). Naturally the design must also comply with the current standards for mathematics teaching, which are formulated in the literature, and these two strands of mathematical proficiency are promoted by the design of the division tutor. The National Council of Teachers of Mathematics formulates 17 specific expectations of students in 3-5 grade, related to the aforementioned understanding of numbers, operations and fluency. These will inform the tutor design and serve as a yardstick for evaluation. (See appendix A)

### **3. Research Question and System Design**

Building a tutor that successfully teaches the skill of division requires insight in the steps, processes and strategies used by 3-5 graders when solving a division problem. These steps were identified and they informed the design of the type and set of the division problems that the ITS made the student practice. The design of the ITS crystallized from of the answers to the following research question, consisting of three consecutive steps, from strategies to design:

- 1. Which strategies do 3-5 graders use when solving a problem of dividing a number  $< 100$  by a number  $< 10$ ?**
- 2. How can knowledge of these strategies be used to construct a problem set?**
- 3. How do we apply self-explanation to the problem set?**

The next paragraphs each discuss how part of the research question is approached and solved, setting up the next part of the research question. The final answer is the resulting tutor design.

#### **3.1 Division Strategies**

How are division problems solved? The textbooks on division for 4<sup>th</sup> graders teach explicit strategies for solving division problems, but we wanted augment and verify this instruction material

with a first hand experience of what division means for a 4<sup>th</sup> grader, in order to know for what is important and what is not. If we are informed by the textbook only, this could lead to a misguided design that might emphasize the wrong elements and level of detail of the division learning process. Getting this first hand feel of the division learning process is the reason of the “think aloud” sessions, to establish a firm ground for the design.

1. Which strategies do 3-5 graders use when solving a problem of dividing a number  $< 100$  by a number  $< 10$ ?

Students need to learn about the divisibility rules because then they can decide quickly whether a certain number is divisible in the first place, without relying on calculations. When learning division in school, the divisibility rules are combined with the earlier learned multiplication facts as stepping stones for solving a division problem. For example, when the problem is dividing 40 by 4, the rule that numbers ending with 0 can be divided by 10 is remembered, and the known multiplication fact that  $10 \times 4 = 40$  can be used to confirm that the answer to the division problem is 4. As more instances are created by repetition, the combinations of divisibility rules and multiplication facts become known division facts. The process of acquiring these facts varies per student, but all develop proficiency with division on a basis of known multiplication facts.

See table 1 for all the divisibility rules for numbers below 100:

- |   |  |
|---|--|
| 2 | If the last digit is even, the number is divisible by 2.                 |
| 3 | If the sum of the digits is divisible by 3, the number is also.          |
| 4 | If the last two digits form a number divisible by 4, the number is also. |
| 5 | If the last digit is a 5 or a 0, the number is divisible by 5.           |
| 6 | If the number is divisible by both 3 and 2, it is also divisible by 6.   |

7        Take the last digit, double it, and subtract it from the rest of the number;  
if the answer is divisible by 7 (including 0), then the number is also.

9        If the sum of the digits is divisible by 9, the number is also.

10       If the number ends in 0, it is divisible by 10.

Table 1: Divisibility rules below 100

Some examples of division steps and strategies are:

-Knowing that all even numbers are divisible by 2.

-Knowing how to check if a number is odd or even. Is the last digit even? Is it 0,2,4,6,8?

-Knowing that numbers above 10 that end with a 5 or 0 are all divisible by 5 and 2.

### 3.1 Think aloud

A series of "think aloud" sessions were held at a suburban elementary school to get a qualitative empirical understanding and a first hand feel of how 4th graders do division. Before the sessions were held, one of the teachers was interviewed on the teaching method used for division at this school. Learning division involved the learning of division facts based on known multiplication facts. These 4<sup>th</sup> graders were expected to be fluent at multiplication at the time of the "think aloud" sessions, and they had already learned how to use these multiplication facts to solve division problems.

During the "think aloud" sessions, six 4th graders were interviewed and observed separately while solving various division problems. Parental consent was obtained, and in addition, at the beginning of each session it was made clear to the subject that he or she did not have to do something they did not want to, and that it was OK to quit anytime. The subjects were selected by their teacher and were from three categories of math grade levels: "low", "middle", and "high", as

categorized by the teacher. The interviewer (the author) was informed by the teacher on that category after the interviews.

The procedure for the interviews was based on a procedure for observing users (Gomoll, 1992):

1. Set up the observation
2. Describe the purpose of the observation
3. Tell the user that it is OK to quit anytime
4. Talk about and demonstrate the equipment in the room
5. Explain how to "Think aloud"
6. Explain that you will not provide help
7. Describe the tasks and introduce the product
8. Ask if there are any questions before you start; then begin the observation
9. Conclude the observation
10. Use the results

The subjects did different kinds of division problems of two classes: regular division problems and divisibility problems, which only have a yes/no answer. They were asked verbal questions and they did paper tests. See appendix. These "think aloud" sessions showed that students vary in the methods they use to solve the same division problems. Typical dialogues include:

Interviewer: "Is 15 divisible by 2?"

Responses:

- "No, because there is nothing you can do times 2 to get to 15."

- "If you count by two's you never get to 15."

- "6 times 2 is 12, which is too low, 7 times 2 is 14, which is too low, 8 times 2 is 16, which is too high, so 15 is NOT divisible by 2."

- "No, because it ends with a 5 and 5 is an odd number."

Or:

Interviewer: "Is 60 divisible by 10?"

Response: "Yes, because it ends with a zero and numbers that end with a zero are divisible by 10."

On paper test items typical verbal responses were:

-“30 divided by 6 is 5 because I know that 6 times 5 is 30.”

-“75 divided by 5 is... I know it is more than 12 because 5 times 12 is 60... And I have to go up 15 more, which is 3 times 5.. so it is 15.”

The subjects were able to answer most questions correctly, but the two subjects that were categorized as having lower math grades made more mistakes, often being unaware they made them. Most subjects were able to use the rules to answer divisibility questions, but they used multiplication facts more often than the divisibility rules, i.e. to check whether 14 is divisible by 2 they “counted by two’s”, from 2, 4, 6 up to 14. This method is less efficient than using the divisibility rule that even numbers are divisible by 2, and then using the fact that 4, the last digit, is an even number. This suggested they could benefit from more training on using the divisibility rules when appropriate.

All but one subject used multiplication facts fluently when needed, but there were noticeable differences between students in division speed and accuracy, and certain kinds of problems were clearly harder to solve than others. It seems that the problems higher in the multiplication table ( $63 / 7$ ,  $48 / 6$ ) are more difficult, as is to be expected when multiplication facts are used to solve them. If they have to “count up” the multiplication table to solve division problems, the lower multiplication facts are both referenced more often and reached earlier when counting. Both these effects probably accumulate over the learning process of division, and so facts lower in the table are likely to be used more fluently.

The conclusion drawn from these “think aloud” sessions was that division problems are solved in one of the following ways:

1. The division fact is recalled and the solution is given immediately.
2. The multiplication facts are used to estimate the solution of a division problem and progress towards it or the multiplication fact that is the inverse of the division problem is used correctly on the first try.

3. A divisibility rule is applied to a divisibility question. Divisibility questions can also be solved with multiplication facts when the applicable rule is not recalled (this happened often during the “think aloud” sessions).

We can see now that a division problem has an identifiable anatomy and that the possible processes of solving it are only variations of these three ways. Therefore, acquiring and improving division skill with the ITS for division was decided to have to happen based on these division solving methods. These design choices were made based on these methods for solving division problems:

The interface will include a multiplication table, so all multiplication facts will be available for solving a division problem, and there are references to these multiplication facts as the foundation of division facts.

Divisibility problem will be tutored from on a known multiplication facts or divisibility rules. The different divisibility rules vary in their usefulness. The divisibility rules for 4, 6 and 7 are redundant or too bothersome to be useful for 2-digit numbers. These strategies were never used by the subjects, as opposed to those for 2, 3, 5, 9 and 10.

The learning goal for the student is to be able to recall division facts faster than he or she can calculate them with multiplication facts or divisibility rules, and then to become fluent at this recall. The design choices are explained in more detail in the next section on the kind of problems the tutor presents to the students, the “problem set”.

### **3.2 Problem set**

The answer to the second part of the research question shows how the observations of the “think aloud” sessions, and the theory of mathematical proficiency, translate into the kind of division problems the tutor should teach, and how they should be tutored. The question was:

2. How can knowledge of these strategies be used to construct a problem set?

The design features of the ITS constitute four interfaces, and each interface supports a combination of a division proficiency (fluency or sense making), and a problem type (divisibility or division). It includes a multiplication table to support the link between multiplication- and division facts.

The supporting features for the strand of procedural fluency are the easiest to implement. The tutor must emphasize speed because that is identical with fluency in the case of simple division problems. Fluency is supported by actively encouraging the student to work faster. To keep the solving urgent and challenging, the score of the number of correct problems is shown on the screen. Also, a timer shows how long the student has been working on the current “bundle” of 12 division problems. This feedback enables the student to monitor and improve performance, even when the answer is known quickly and being correct may no longer be challenging in itself. See figure 1 and 2.

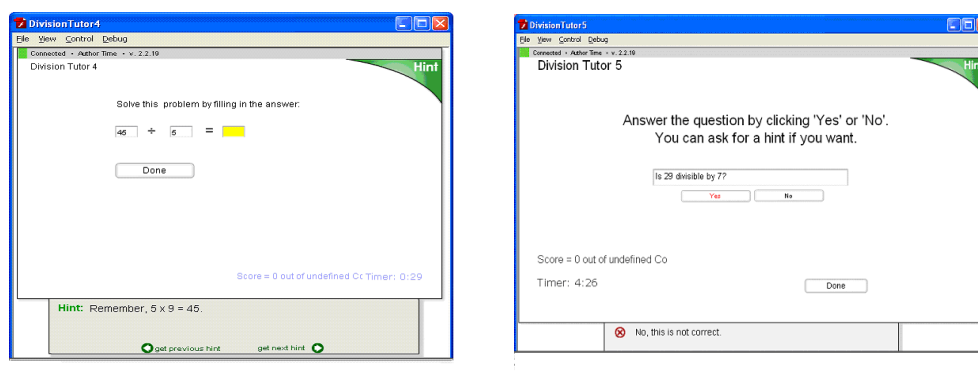


Figure 1 (left): Interface focusing on fluent division. Figure 2 (right): Interface focusing on fluent application of divisibility rules.

As explained above, the learning process of division is based on prior knowledge of divisibility rules and multiplication facts, therefore the strategies that draw on this knowledge are supported by the tutor interface. Multiplication facts and division strategies each have their supporting feature in the interface: a multiplication table and a list of the divisibility rules, and hint and error messages when the student interacts with interface to solve a problem.

Multiplication facts are made available for reference in the multiplication table in the tutor interface, so a student solving a division problem can rely on the inverse multiplication fact when needed. A list of divisibility rules are made available in the two interfaces that tutor on divisibility. Not all strategies are supported by this interface. As suggested by the “think aloud” sessions, the divisibility rules for 4, 6 and 7 are redundant or too bothersome to be useful for 2-digit numbers.



These strategies were never used, as opposed to those for 2, 3, 5, 9 and 10, and the interface design reflects this. See figure 3 and 4.

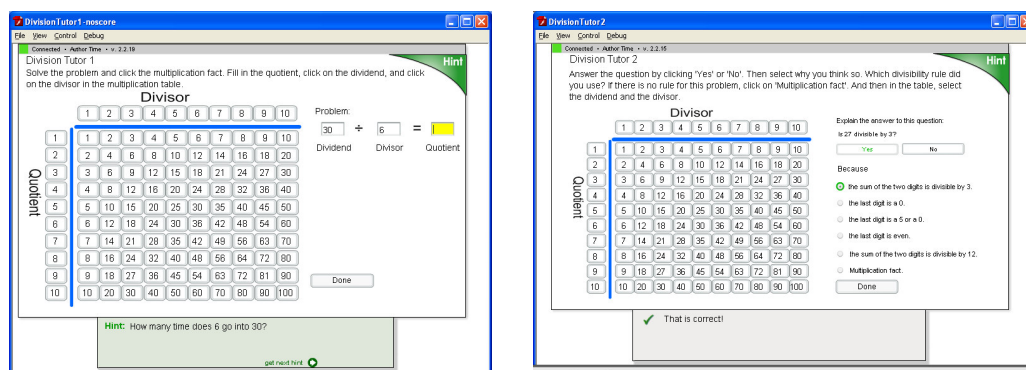


Figure 3 (left): Interface focusing on reflection on division. The student clicks the dividend, divisor and quotient on the table, and enters the quotient in the answer field. Figure 4 (right): Interface focusing on reflection on divisibility rules. The student must select the rule that was used to answer the question, or when no rule applies, he or she must click the dividend, divisor and quotient in the table.

Another design decision that was informed by the sessions was the level of detail at which the divisibility rules are taught. At one extreme we can split up the rules (like showing which numbers are even for the divisibility rule for 2), or make the 5 and the 10 rule visible with lines in the multiplication table, or show the proof behind the 3 and 9 rule. Then the student would learn the rules from the inside out, but in the “think aloud” sessions the students seemed to be able to recall and use the rules very well as a single fact. And for 2-digit division problems the divisibility rules are mostly used as a stepping stone in the learning process, so deep comprehension of them is less important for 2-digit division problems. The students must be able to answer divisibility questions to check whether a solution exists for a division problem. Once they are fluent at division, they no longer need the rules. Therefore the level of detail of these tutor problems was set as the use of the rules only, to solve divisibility questions, without tutoring on the details of the rules.

### 3.3 Self-Explanation

3.How do we apply self-explanation to the problem set?

A study shows that self-explanation is beneficial to the learning process, because explaining the answers to yourself when interacting with the tutor leads to greater understanding (Aleven, Koedinger & Cross, 1999). Subjects in that study also did better at a type of transfer problem in which they were asked to judge if there was sufficient information to find unknown quantities. Also, students who explained solution steps were significantly better at steps where quantities sought were difficult to guess (in other words, require deeper knowledge) while answer-only students did better on easy-to-guess items indicating superficial understanding. Finally, students who explained answers did fewer problems during training, which suggests the training on reason-giving transfers to answer-giving. While these results were achieved with a geometry tutor, they can be expected to apply to division skills as well.

The self-explanation of the multiplication fact that is the inverse of a division problem should help the conceptual understanding (sense making) of division, and self-explanation of divisibility rules should help conceptual understanding of divisibility. The tutor interfaces support this in two ways: In the interface for division problems with a multiplication table, the student must click the complete multiplication fact that is the inverse of the division problem (the dividend, divisor and quotient), and in the multiplication table after the solution (the quotient) is entered.

In the interface for division problems, the student enters the quotient in the appropriate field, and then self-explains the division problem by entering the components in the multiplication table in an arbitrary order. This re-enforces the links between the dividend, divisor and quotient, and the inverse multiplication fact. When all components of the problem are entered, the student may click the “done” button available to the student to get the next problem.

As the student is exposed to the components of the problem, the repeated linking between the components of a division problem and the corresponding inverse multiplication fact is expected to strengthen declarative and procedural knowledge. As explained above, both specific division problems and division problem solving in general have a clear anatomy, and being fluent at them is identical with having procedural ability. A fluent problem solver behaves like an expert by recognising the patterns in the problem, which then trigger cached responses to solve it.

With the interface for divisibility strategies the student first has to answer a divisibility question, like “Is 15 divisible by 2?”, by clicking ‘Yes’ or ‘No’, without seeing the divisibility

rules. These only become visible when the answer is 'Yes', since it not very useful to explain why something is not divisible. So the student answers the question by him or her self, either by using a divisibility rule, or by referencing the multiplication facts. When the correct response is 'No', the next problem is presented. When the correct response is 'Yes', and the student clicks it, radio buttons with the divisibility rules plus a button for 'Multiplication fact' appear. Together they are the divisibility strategies the student is likely to have used to give the answer. The question and the strategies are bridged by the word "Because," to change the student from 'answer mode' into 'reason-giving' mode.

The student selects the radio button of the strategy he or she used. If a rule was used, the student clicks the rule, and then clicks 'done' to get the next problem. When the inverse multiplication fact was used, the student selects "Multiplication fact", and then clicks the dividend, divisor and quotient in the multiplication table, analogue to the division interface.

#### **4. Discussion and Conclusion**

The tutor was build and implemented with a flash toolbox called Cognitive Tutor Authoring Tools (CTAT), developed at the PSLC. These tools can be downloaded at the Learnlab website, presently at <http://ctat.pact.cs.cmu.edu/>

Fluency is taught by drill, with feedback on scoring and timing, encouraging the student to work as fast an accurately as possible. Sense making is facilitated by making the student self-explain actions by having him or her click the multiplication fact related to the division problem, or by selecting the correct divisibility rule. Sense making is limited to the two interfaces with a multiplication table, while fluency features are applied to all interfaces. When students become faster at self-explanation, fluency also applies to that. In the current implementation, scoring and timing are not applied to the sense making interfaces, since we wanted to contrast conceptual understanding and procedural fluency in the evaluation experiment (King, 2008). But when comparison and evaluation is not an issue and the tutor is used to teach division in a real class situation, it is probably best to apply scoring and timing to all interfaces.

The tutor is made of 4 interfaces in total, numbered 1, 2, 4 and 5, which are presented to the student after each other, each serving a bundle of 12 problems. Interface 3 was much alike interface 1 and was omitted later. The interfaces are designed around the strands of procedural fluency and conceptual understanding, based on the structure of division problems observed in the think aloud.

While we can only be sure of the learning gains achieved with the tutor after analyzing the data of the evaluation study, at this point it can be concluded that answering the research question has translated into an empirically informed design, founded in current practice of teaching fluent division in elementary school, and mathematical proficiency theory.

In addition to the features discussed above, hints and error messages direct the student's actions to completion of a division or divisibility problem. CTAT has a useful standard for showing hint and error messages, and all the components needed for building and implementing the interfaces of the division tutor are included in the newest release. Together with the hint and error messages listed in appendix 2, one should be able to recreate the interfaces.

If you want to use this ITS in a real class setting, then note that serving the interfaces and the problems to a website relies on systems currently only present at Carnegie Mellon University. Considerable knowledge and expertise of server networks is needed to set up a situation where the tutor is on line. However, with some HTML knowledge it is possible to make it work on a single machine.

## **5. References**

Aleven, V., Koedinger, K.R., Cross, K., (1999). Tutoring Answer Explanation Fosters Learning with Understanding. *Proceedings of the 9th International Conference on Artificial Intelligence in Education*, AIED '99, 199-206.

Anderson, J.R., Corbett, A.T., Koedinger, A.R., & Pelletier, R. (1995). Cognitive Tutors: Lessons Learned. *The Journal of the Learning Sciences*, 4(2), 167-207

Gomoll, K. (1992). Some Techniques for Observing Users. Retrieved from: <http://download.srv.cs.cmu.edu/~pausch/Randy/Randy/Some%20Techniques%20for%20Observing%20Users.pdf> Advanced Technology Group Apple Computer, Inc.

Habgood, M.P.J., Ainsworth, S.E., Benford, S. (2005). Endogeneous Fantasy and Learning in Digital Games. *Simulation and Gaming*, 36(4) 483-498

Habgood, M.P.J. (2007). *The Effective Integration of Digital Games and Learning Content*. PhD thesis, University of Nottingham.

Hiebert, J., & Carpenter, T. (1992). Learning and teaching with understanding. In D. Grouws (Ed.), *Handbook of research on mathematics research and teaching*. (pp. 65-100). New York: MacMillan.

King, S.T. (2008). *Evaluation of an Intelligent Tutoring System for Robust Division Skill*. Master thesis, University of Groningen.

Mathematics Learning Study Committee, National Research Council National Academies, (2001). *Adding It Up: Helping Children Learn Mathematics*, National Academies Press

National Council of Teachers of Mathematics, (2003). *Principles and Standards for School Mathematics*.

National Council of Teachers of Mathematics, (2003). A Research Companion to *Principles and Standards for School Mathematics*.

Rittle-Johnson, B. & Alibali, M.W. (1999). Conceptual and procedural knowledge of mathematics: Does one lead to the other? *Journal of Educational Psychology*, 91, 1-16.

## **Appendix 1: Number and Operations Standard for Grades 3-5**

<b>Instructional programs from prekindergarten through grade 12 should enable all students to-</b>	<b>Expectations</b> <b>In grades 3-5 all students should-</b>
Understand numbers, ways of representing numbers, and number systems	Understand the place-value structure of the base-ten number system and be able to represent and compare whole numbers and decimals;
	Recognize equivalent representations for the same number and generate them by decomposing and composing numbers;
	Develop an understanding of fractions as parts of unit wholes, as parts of a collection, as locations on number lines, and as divisions of whole numbers;
	Use models, benchmarks, and equivalent forms to judge the size of fractions;
	Recognize and generate equivalent forms of commonly used fractions, decimals and percents;
	Explore numbers less than zero by exploring the number line and through familiar applications;
	Describe classes of numbers according to characteristics such as the nature of their fraction.
<b>Understand the meanings of operations and how they relate to each other</b>	<b>Understand various meanings of multiplication and division;</b>
	<b>Understanding the effects of multiplying and dividing whole numbers;</b>
	<b>Identify and use relationships between operations, such as division as the inverse of multiplication, to solve problems;</b>
	Understand and use properties of operations, such as the distributivity of multiplication over addition;
<b>Compute fluently and make reasonable estimates</b>	<b>Develop fluency with basic number combinations of multiplication and division and use these combinations to mentally compute related problems, such as <math>30 \times 50</math>;</b>
	<b>Develop fluency in adding, subtracting, multiplying and dividing whole numbers;</b>
	Develop and use strategies to estimate the results of whole-numbers computations and to judge the reasonableness of such results;
	Develop and use strategies to estimate computations involving fractions and decimals in situation relevant to students' experience;
	Use visual models, benchmarks, and equivalent forms to add and subtract commonly used fractions and decimals;
	Select appropriate methods and tools for computing with whole numbers from among mental computation, estimation, calculators, and paper and pencil according to the context and nature of computation and use the selected method or tool.

