

Differentiated Instruction: A Technology-Enabled Approach

Brian Wheeler

6.UAP

Spring 2010

Advisor: Dr. Kimberle Koile

1. Introduction: A Classroom Scenario

Imagine the following two scenarios.

Low-tech, Poor Differentiation

A classroom of 30 students is split up into 3 groups based on ability and given separate worksheets appropriate to their level. Some of the less advanced students have trouble with the worksheet and wave to get the teachers attention, distracting their neighbors in the process. Some of the more advanced students finish quickly, but their physical grouping prevents them from helping their less advanced classmates. Six of the thirty students make similar mistakes on the worksheet but the teacher is not aware until reviewing the students' solutions later that day (Collecting worksheets would be cumbersome and there is no way to easily review individual problems). Revisiting the problem the next day proves difficult because students are less interested the second time around.

High-tech, Good Differentiation

Students in a classroom are given tablet computers with educational software that contains a presentation of slides the teacher has prepared. The teacher has a similar tablet and is wirelessly linked to both the students as well as a projector. The teacher presents a lesson via the slides on the students' tablets and at chosen intervals prompts students to complete problems on the current slides. The students have been previously designated into groups based on their abilities and the problems they see differ depending on their group. Solutions to problems are wirelessly submitted to the teacher's machine. After quickly scanning some solutions the teacher notices a common mistake and illustrates how to fix it by writing on her tablet and displaying the results on the projector. Some advanced students finish early and help their neighbors. The teacher displays correct student solutions anonymously on the projector to show how different students came to the same solution in different ways.

2. Background: CLP

The previous scenarios describe, respectively, a common method for differentiating instruction and a more easily implemented, technology-aided method of differentiated instruction. The focus of this UAP has been the addition of a technology-aided differentiated instruction capability to the Classroom Learning Partner (CLP) software system, under development by Dr. Kimberle Koile's research group.

CLP was originally developed as a classroom interaction system for use by Dr. Koile in her undergraduate computer science classes. Recently, as part of the NSF funded project , *INK-12*:

Interactive Ink Inscriptions in K-12, CLP has been extended for use in K-12 classrooms. From the INK-12 project description:

In [the INK-12] project, we have been investigating the role that pen-based wireless technology could play in 4th and 8th grade science and math. We have thus far conducted tablet computing trials in five schools—in four 4th and three 8th grade classrooms. The technology used in the trials consists of a set of tablet computers running a software system called Classroom Learning Partner (CLP), developed by Kimberle Koile's research group. CLP is built on top of a wireless presentation system called Classroom Presenter (Anderson et al., 2004, 2007), and like Classroom Presenter, it embodies a “presentation slide” metaphor for classroom interaction: A teacher creates a lesson as a series of slides, e.g., a la Powerpoint. In the classroom, the teacher and students all have tablets on which the lesson is running; the lesson is also projected publicly on a screen at the front of the classroom. The teacher can annotate a displayed slide by using the tablet's stylus to create “digital ink”, which, by way of a wireless network, is then visible on the students' and public screens. Just as the teacher can send “digital ink” to students, students can send “digital ink” to the teacher. Classroom Presenter, and CLP, use this communication channel as a way for students to send the teacher handwritten “digital ink” answers to in-class exercises. The teacher then can select some, or all, of those answers to display anonymously on the public screen and discuss with the class. The teacher has the option to “link” student machines to her machine, meaning that the teacher controls which slide is visible on the student machines. The alternative is to let the students freely navigate through all the slides in the lesson

Kimberle Koile and Andee Rubin, January, 2008

2.1 Differentiated Instruction

A large and growing number of experts support the notion that students learn best when instruction is tailored to their specific needs (e.g. Bransford et. al. 1999). This differentiated instruction, as it is

termed, is implemented most basically at the grade-level differentiation and more specifically 1) by assigning same-grade-level students into different classes based on ability, 2) by assigning students in the same class into different groups based on ability, 3) by designating of students as special needs, and 4) using teaching methods target different abilities and different types of learning preferences.

2.2 CLP and Differentiated Instruction

Prior to this UAP, CLP implemented differentiated instruction via slides marked with group labels; students were prompted by the teacher to complete the problem that corresponded to their ability group. (See Figure 1). Students were assigned a letter for a particular problem, and they worked only the problems labeled with that letter. Discussions with teachers lead to the ideas that: 1) the extra slides for all the ability groups could be distracting and confusing to students, and 2) instruction without

Classroom Learning Partner 2.4.000 Instructor View admin@CLP16.ConferenceXP.Net Disconnected

File Connect Tools Role View Decks Student CLP Actions Help

slides

Graphing Data Example

Create a line graph that shows the growth of the plant over 7 days. Remember to label your axes with variable names and units, title your graph, and check your work.

Day	Height of Plant
1	1 cm
2	3 cm
3	5
4	7
5	10 cm
6	11 cm
7	12 cm

Data table



You do:

A

B

or

C

Draw a graph here.

Do you see any pattern in the height of the plant? _____

How tall do you think the plant might be on Day 8 if it kept growing the same way? _____

Put that point on the graph in a different color and explain why you placed it there.

Current: 2 / 11 Slides: 11

start uapy.bmp - Paint Presenter - Microsoft ... Classroom Learning P... 7:27 PM

knowledge of one's level might be helpful for eliminating any possible negative effectives of known differentiation.

Figure 1a: Instruction Slide

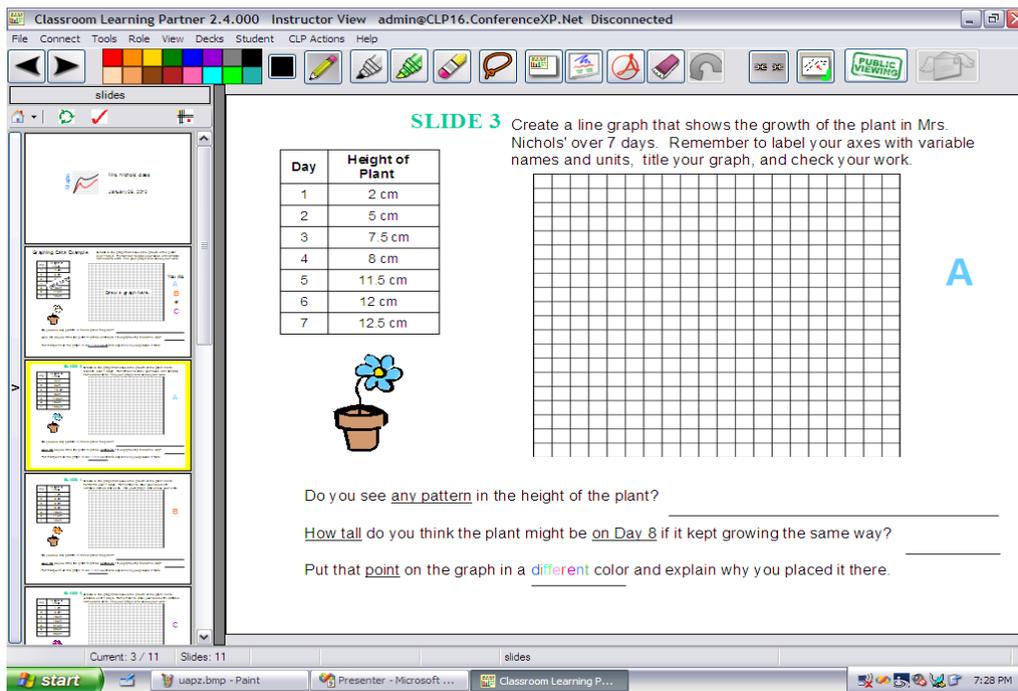


Figure 1b: Differentiated slide

3. Implementation

The implementation of a new differentiated instruction scheme involves two major components: 1) using information about students' groups to retrieve appropriate slides and 2) altering CLP to allow differentiated slides to be used.

3.1 Retrieving Slides

In the new ability level differentiated instruction scheme whereby a student only sees one problem (which has been tailored to his or her ability level), students need to receive their slides automatically and in a manner that keeps their ability level hidden. Implementing this new functionality was possible via modification to the existing startup scheme. Previously, students' CLP programs completed two

separate, non-interacting processes: 1) They automatically a lesson's slides from a classroom server, and 2) Students were prompted to choose their name to log in via a GUI populated with a class list stored on the server.

To implement the new slide retrieval functionality the code was changed to parse a class list file, stored in csv format, with an extra parameter—the students group number. The class list file, which previously contained information in the form 'last_name, first_name', can now contain information in the form 'last_name, first_name, ability_level'. This new format provides a student's ability level to CLP, which uses it to retrieve a set of slides associated with that level. This functionality was easily implemented by represented ability level by an integer and mapping that integer to appropriate slides. This was a simple solution that is extendable and can be used implement any number of other possible functionalities that require using student information to create a dynamic learning environment. (See Figure 2 for a diagram of current retrieval steps.)

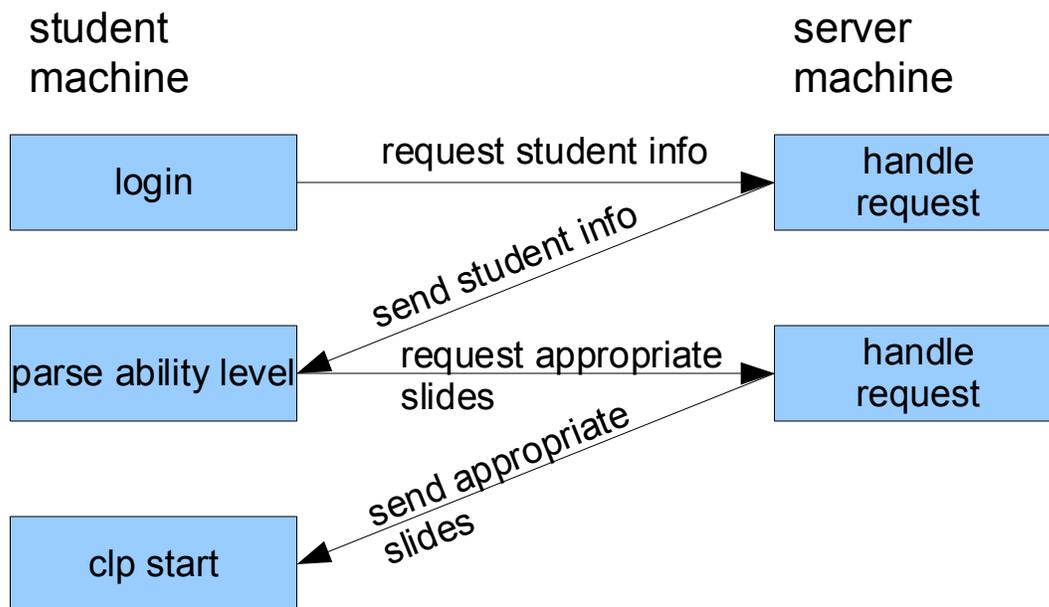


Figure 2: A student machine fetches appropriate slides

3.2 Altering CLP to Accommodate Differentiated Slides

The bulk of this UAP consisted of altering CLP to accommodate differentiated slides. For CLP to function correctly, the slide deck that students and teachers are using must be considered “the same”: CLP can accommodate machines having multiple slide decks open; the software matches a student's deck to the corresponding teacher deck when students and teachers transmit digital ink back and forth. The instances of identical slide decks, which CLP inherited from its underlying infrastructure (Classroom Presenter) proved problematic: the whole idea was to create different decks in order to differentiate instruction. To circumvent this problem, we needed a less restrictive definition for matching.

The CLP software is comprised of thousands of lines of code that rely heavily on comparing the structure and content of the students' slides and teacher's slides when making functional decisions; any alteration to slide structure requires extensive verification that the changes do not cause any errors. In the original version of CLP, slides were compared at two different levels: 1) slide content and 2) slide metadata. Fortunately, slides were not checked for content in any places that involve student-teacher interactions (this would have made differentiated slides more difficult to implement because, unlike slide metadata, slide content is seen by the user and cannot be modified). Instead, all of the functional decisions in student-teacher interactions rely on slide metadata: slide titles, slide deck ids, slide ids, slide indexes, and the number of slides. Slide ids, slide indexes, and the number of slides are not only used for matching teacher decks to student decks, but they are also used for the sending and receiving of slides—a key aspect of CLP. Because these attributes are so essential and embedded, the differentiated instruction project avoided altering them, which resulted in a key constraint—slide decks need to have the same number of slides.

Before continuing, it is important to illustrate how a student submission to a teacher's tablet is handled

in CLP. In the following figure, the top image is an example of an unaltered slide in a slide deck; the bottom image is a slide on which a student has written his or her answer on. When the student submits this answer, only the digital ink is sent to the teacher's machine in order to send as little data as possible and thereby increase the speed of ink display. CLP then overlays this ink on a copy of the teachers slide so that the teacher can see the answer: CLP assumes that the teacher and the student share the same slide so that the background image for the new ink need not be transmitted.

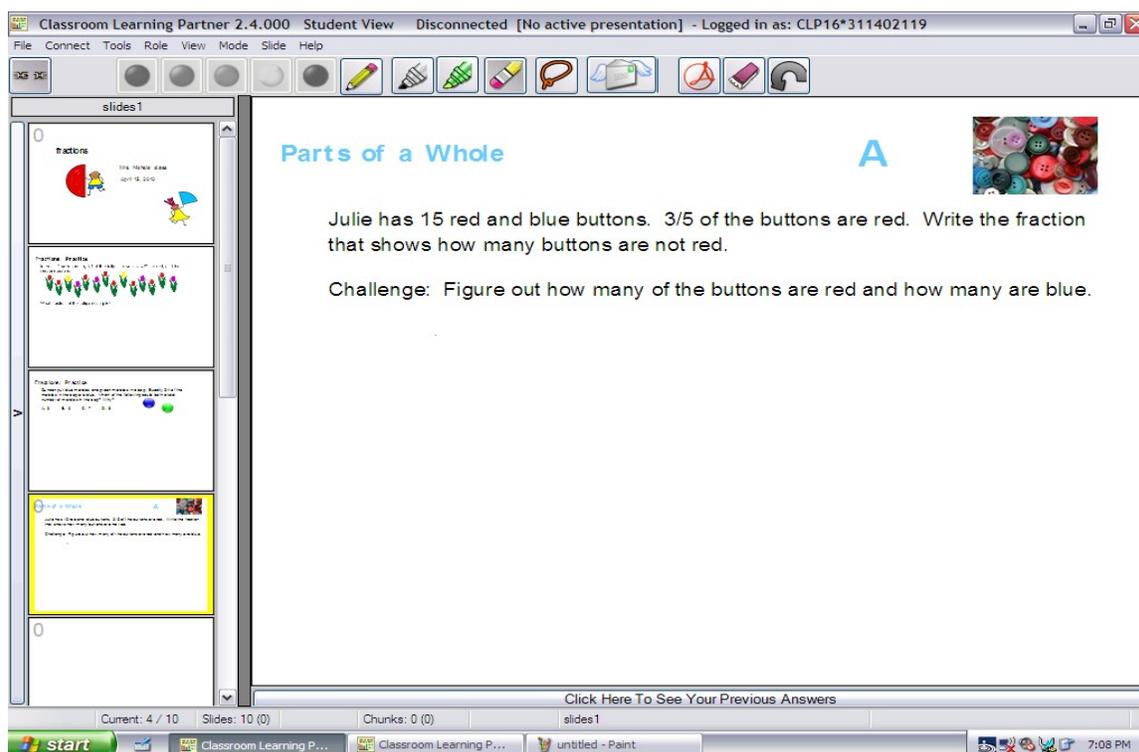


Figure 3a: Slide background (i.e. slide not written on).

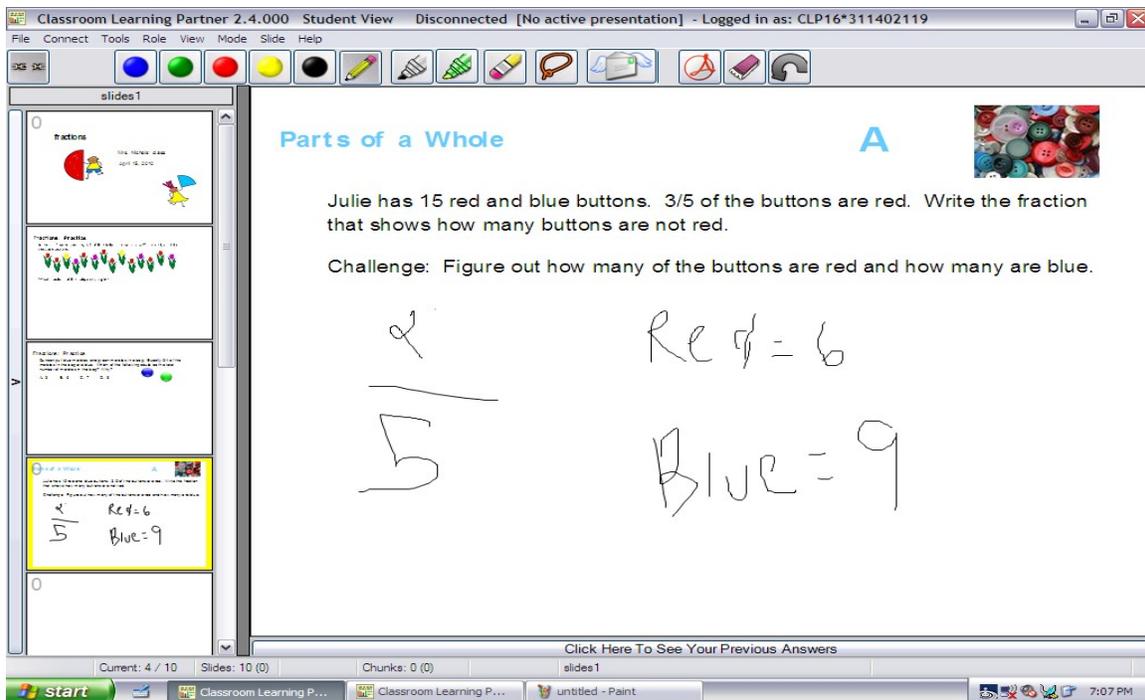


Figure 3b: Student answer represented as an ink overlay

Considering the constraints imposed by the inability to alter some metadata, such as the number of slides in a deck, the differentiated instruction scheme could have been designed two ways: 1) teachers and students could have one slide per differentiated problem, which introduces the issue of deciding which of the different slides the teacher should have and figuring out what to do about the fact that some student slides will have different backgrounds, necessitating transmittal of both background and ink overlay, or 2) make the students and the teachers each have three slides, one for each version of a problem, with the teacher having each version of the problem and the student having blank slides in place of problems not at their ability level. The second solution proved much easier to implement and less complicated for the teacher and it is the design that was chosen. This design required two changes: 1) designating slide deck ids differently, and 2) altering slide titles matching. It's important to note that the underlying structure's dependence on a slide presentation metaphor hampered the implementation of a more elegant solution that could have accommodated more dynamic selection of slides at runtime.

3.2.1 Altering Slide Deck ID's

The differentiated instruction scheme requires the creation of several slide decks—one for the teacher which contains problems for all the ability levels. And several student decks, one for each ability level. One student deck, for example, might contain a problem for level A, followed by a blank slide, which serves as a placeholder for a second ability level's problem. A second student deck might in turn contain a blank slide where the A level problem would be, followed by a B problem.

When the needed slide decks are created using CLP's Instructor Authoring Tool, they are saved in a format containing data stored in the following structure:

[clp version #] [20 bytes]

[data length parameters] [16 bytes]

[csd document] [data length parameter 1]

[xml data][data length parameter 2]

The csd document portion of the file, where the actual slide content is stored, is a CLP-defined structure that contains the slide deck id. Therefore, in order to make slide deck id's equal, a command line parameter (-diff) was created to be used in the following way:

```
[path_to_clp]\Presenter.exe -diff clpfile1 clpfile2 clpfile3
```

Here, the slide deck id is read from .clp file1 and used to overwrite the existing slide deck ids in the other two files. Three new files are written to verify correct functioning, and they all share the same slide deck id.

3.2.2 Slide Titles

The other pieces of metadata that are used to match slide decks were the slide titles. The existing verification system, however, used slide titles merely as extra, redundant data for matching slide decks. Therefore, the easiest solution to this problem was to simply remove the redundant check.

3.3 Implementation Overview

With the aforementioned changes in place, the differentiated instruction functionality works as follows:

1. A slide deck is created with multiple instances of problems for different ability levels
2. For each level, a new slide deck is created with black slides as placeholders for other levels.
3. The file names are passed as parameters to the -diff function and slide ids in the files are made equal.
4. The class list file is created with students' names and ability levels; the current implementation uses a csv file format.
5. Students log in and receive a slide deck that is unique for their ability level

4. In-Class Testing

We were extremely surprised by the difference in the teacher's and students' reaction to using the two differentiated instruction methods: The teacher thought it best if the students' ability levels were not emphasized, if the students thought they were just working a variety of problems. She liked the idea of tailoring the lesson for each student's ability level by giving each student a set of problems matched to the student's ability.

The students, on the other hand, overwhelmingly preferred to see all the problems, not just the one matched to their level. The reasons given:

I know that the problem is matched to my level.

This reason is particularly interesting because when the student giving this reason was told that the single problem given her was matched to her level, she seemed very surprised. By seeing the different levels of problems, she could verify that her problem was matched to her level. With a single problem, she couldn't explicitly see the matching.

I can see other problems and work one of them if it's better for me.

This reason also benefits the teacher: The teacher does not have to predetermine a student's ability level. In some cases, it may be difficult for the teacher to know how well a student knows the material, so letting the student choose which problem to work may result in a better matching to the student's ability level.

If I get finished early, there are other problems I can work.

Having students finish problems at different times is always a challenge. Providing the extra problems, even if not matched to a student's level, gave early finishers something to do.

When the teacher puts up answers for the other problems, it's easier to follow the discussion if I can see the problems on my on screen.

Students seem to lose interest when only seeing a problem for the first time when answers to it were projected.

I can see that other people are working on harder problems and that one of them could help me before

I send in my answer to the teacher.

This reason came from a student who was aware that she was not a strong math student and who liked

to learn from her classmates. She did not want to send the teacher wrong answers, so liked getting help before submitting an answer.

It is important to note that the skill of the teacher was the primary factor in the students being comfortable with knowing that they were divided into different ability levels: The teacher is an excellent teacher who succeeds in creating a classroom atmosphere in which students know that it is OK to make mistakes and know that they are all to be respected for whatever skill levels they have. In a classroom without this sort of atmosphere, the reaction to the two methods of differentiated instruction might have been different.

5. Future Work

The new differentiated instruction component for CLP offers a unique opportunity for studying differentiated instruction in the classroom. It's automation of differentiated instruction makes experiments more easily implemented and efficient; it's infrastructure can be built out to record any number of parameters. CLP, for example, can easily be extended to record student submission times, student submission grades, and other valuable parameters in studying an educational method's effectiveness. Our surprising result—that students preferred working problems matched to their abilities, but wanted to see all problems—leads to the need to test the idea of differentiated instruction further. As teachers and researchers advocate for tailoring instruction for students, that tailoring may not mean providing only ability-specific materials.

6 Contributions

This UAP focused on adding valuable functionality—differentiated instruction—into the Classroom Learning Partner software. The project helped develop a method of data transfer that enabled a more dynamic learning environment. Further, it explored the possibilities of implementing a differentiated

instruction scheme within the restrictions of the software. Upon examination of the software, key obstacles were identified and new functionality was created (e.g. the ability to have “matching” decks that are not exactly the same). Finally, the new implementation of the differentiated instruction functionality was tested in a classroom environment and helped uncover interesting results that suggest that more testing of what it means to differentiate instruction need to be carried out. A platform such as CLP, with the functionality added in this UAP, provides the framework needed for that testing.