make certain that the teams know how to work effectively. Similarly, the student team members need to reflect on the success of their teamwork and prepare to improve their work on future exercises[4].

Many models exist for using CL in the classroom. In the jigsaw[3], the instructional material for a class session is divided up into a number of parts. Groups of students receive one of the parts and work together to prepare to teach this part to other groups of students. In the final step, teams are formed with someone representing each part of the material and the students on the teams take turns teaching each other their parts. The group concludes by summarizing all

Another way that new material is learned in CS 456 is by having groups of 2-3 students discuss how the material applies to a particular situation or problem. Below is an example that was used to help the students to learn the stages of group development.

> Interview an individual who is **NOT** a member of your software engineering team to obtain answers to the following questions. Write the answers in the spaces provided. After you have completed the interview, switch roles and let that person interview you.
>
> 1. What is the current stage of development of your software engineering team? Why do say that? (See pages 246-249 for descriptions of the stages.)
>
> 2. Name the stages of development that you have observed in your team. What are the specific behaviors that you observed in each of the stages?

This activity worked very well. It was not necessary to lecture on the stages of group development. The students were interested in knowing the stages of development traversed by their groups and thus eagerly learned the material from the textbook. The act of discussing the stages with another provided a cooperative, supportive setting for learning new material; the

Writing-to-learn activities and strategies differ substantially in means and ends from traditional, formal writing. The following comparison highlights a few of these differences.

TRADITIONAL ASSIGNMENTS:	**WRITING-TO-LEARN ACTIVITIES**:
Assigned as homework (often a relatively lengthy paper or report)	Assigned impromptu, often completed in class, may also be homework, often short (less than a page)
Process	

throughout the course or at selective key moments, read and responded to at length or not at all by the teacher, graduate assistants, or other class members.

Examples of writing-to-learn activities used in the software engineering course are provided in the remainder of this section. It is important to note that these are different from short-answer questions or quizzes. WTL questions are intended to be exploratory and to help students to examine the possibilities. WTL is informal, so the point is not to offer a tidy, singular answer, but rather to explore what, why, and how something might be answered.

Writing-To-Learn Activity 1: Guided Writing

New concepts are sometimes introduced to students through in-class writing-to-learn activities. Instead of lecturing on a topic, students are asked to read short segments of an article or of the textbook and to answer questions regarding the reading. The writing activities are typically followed by discussion. This method of teaching is superior to the lecture method, because *every* student is involved in the learning process and the level of engagement is deeper than when facts are presented by a teacher. This section presents 2 examples of this form of writing-to-learn. The first example involves learning software unit testing methods and the second example involves learning how to prepare a speech goal for a software engineering presentation.

EXAMPLE 1: **Unit Testing Activity**

```
The following questions refer to the article “Bridging the gap
between black box and white box testing,” by Brain Bryson.

Read page 1 of the article (and read the first two sentences at
the top of page 2) and then follow the instructions below.

According to the paragraph at the bottom of page 1, what are
“black box testing” and “white box testing?”

The last paragraph on page 1 illustrates the concepts of black
box (BB) and white box (WB) testing for a soda machine. Write
your thoughts about what BB and WB testing would involve for
software.

Read the first complete paragraph at the top of page 2 and then
perform the following steps.

You will now perform a series of steps that will lead you through
the development of BB tests for one of the components for your
team project.

1. Select a software component that you have already implemented.
What is the name of the component?
```

2. Briefly describe the specification (set of requirements) for the component.

3. Define a series of inputs that can be used to perform BB testing of the component. For each input, indicate the expected output.

4. Is the test suite sufficient? Would you assume liability for the software after running your tests? Why or why not?

Read the second complete paragraph on page 2 of the article and then perform the steps below.

You will now perform a series of steps that will lead you through the development of WB tests for the components for which you developed BB tests.

1. How many execution paths are there through the code of the component? (Note: each if-statement results in two or more paths.)

2. What percentage of the paths would be covered by performing your BB tests?

3. Produce a series of inputs that will exercise every line of code in the component. For each input, indicate the expected output.

4. Would it be safe to say that the component is bug-free after you have performed your BB and WB tests? Explain.

EXAMPLE 2: SPEECH GOAL ANALYSIS

Each team uses powerpoint to make presentations of the software products that it produces. Each student must make part of each presentation. To teach them how to prepare a speech/presentation, students are given a set of questions, which guide them through the process of preparing a speech; they learn the concepts by *doing* them in class. This is an effective method because the questions posed motivate them to learn the material. The specific questions used to teach students how to write a speech goal are:

1. According to Communicate![2], an important step in preparing a formal presentation is determining the goal. The general goal is the intent of the speech, which can be to entertain, to inform or to persuade. What is the general goal for the presentation that you will make in the next class?

2. The specific goal is a single statem

Support for the warrant is referred to as *backing*.

Rebuttals are counter arguments, counter examples, conflicting information, flaws, and other reasons for not accepting the claim, grounds, warrant and/or backing.

Read the accompanying article ("A usage-model-based approach to test therac-25," by P. Hsia and others, 1995) and identify the claim(s), grounds for each claim, warrant and backing. Produce your own rebuttals. In addition, if the author has presented rebuttals, identify those.

Writing-To-Learn Activity 3: Minute Paper

A minute paper is sometimes used to assess student learning. At the end of a class session, students are asked to summarize what they learned. To limit the lengths of their responses and to cause them to write focused responses, they are usually asked to write their responses on 3x5 index cards. This has resulted in valuable feedback about the areas where student understanding is inadequate. Below is a minute paper ass

really have to learn the material because they have to defend their answers and explain why they think one answer is better than another.

But are students learning more than in the traditional lecture classroom? The answer is not without complexity. When asked this question several students were not sure. There were things they liked about the lecture style of teaching. And yet, they all spoke about Professor Welch's alternative approach as a class where they could not get away with not reading, with not taking responsibility for learning the material. They all recognized that they were now better prepared to face conflict and to explain their ideas and concepts to others. Some felt their formal writing had not gotten significantly better, and yet, most of them recognized that the writing-to-learn activities helped them understand the course material more fully. Two Asian students commented that the interactive group work and problem solving as well as the writing helped them hone their English communication skills—both oral and written. They were both aware that they would be working in a global community in which English was the primary language of idea exchange. One young woman, an advanced undergraduate in her fifth year, argued eloquently that this course has taught her more in a few weeks than all of her lecture courses had in four years. She explained that she understood concepts fully, engaged in the material deeply, now had an understanding of how to work in a team context, and finally, that she could face problems and contexts with a positive end result.

In summary, this paper discussed cooperative learning and writing-to-learn in general and showed how these methods can be used in software engineering education. The results reported in this paper are regarding the use of the techniques in a software engineering course over the past two years. During this time its has been observed that

- student learning improves,
- engagement (of both students and teacher) increases,
- depth of learning increases,
- students enjoy in-class activities more than listening to lectures, and
- students have to work harder and are responsible for their own learning.

However, there also can be negative aspects of these approaches. Students may resent having to take responsibility for their learning. Furthermore, students who are in early stages of intellectual and ethical development[11, 12] may be scared by the new context, wherein the professor is not a guru who imparts knowledge and wisdom for the entire class session, and students must sort through information and draw their own conclusions.

We have found that negative effects can be lessened in several ways. Adopt new teaching methods gradually. Early use of cooperative learning and writing-to-learn methods in the software engineering class were not well received by the students, in part because too many changes were made too quickly. Do not get discouraged if you see many negative responses

when you first employ these techniques; this is commonly occurs for approximately the first three times that the techniques are used. Try to gain perspective; when negative responses are observed, focus on the deep learning and increased student engagement and remember that students may be resisting your attempts to cause them to work hard.

There are several important factors to consider regarding groups. Keep group sizes small; three persons is ideal. If you need to employ larger sized groups, begin with an initial part of the exercise that includes three people and then combine the two small groups into a group of six for the rest of the exercise. Vary the composition of groups. This helps to increase learning by exposing individuals to many different perspectives and by making the class activities more interesting and exciting. Carefully set up each activity. For example, in an activity that considers conflict management styles, groups could be asked to summarize each style and to tell how it applies to a specific scenario that is given by the instructor. Allow adequate time for groups to perform activities. This requires observing each group, monitoring progress, and asking groups how much time they need to finish; completing activities before all groups are finished is frustrating to students who are deeply engaged with course material. Provide adequate summary time at the end of an activity. This can be accomplished by having students write a minute paper about the group feedback they received or about what they learned in the activity; this can be followed by the instructor enumerating the points that the students should have learned and asking the students if they would like more information about any of the points. An alternate summary method is to ask one or two groups to report to the class about what they learned from each other and about how they answered the questions, addressed the issues or covered the points.

6. Acknowledgements

This work was supported in part by a grant from the Undergraduate Learning Pool of the Ohio University 1804 Endowment.

Many of the techniques described in this paper were evolved from ideas learned from workshops conducted by Ohio University's Center for Teaching Excellence and Center for Writing Excellence and from *Engaging Ideas*: *The Professor's Guide to Integrating Writing, Critical Thinking, and Active Learning in the Classroom* [10].

7. References

4. D. W. JOHNSON et al., *Active Learning: Cooperative in the College Classroom*, Interaction Book Company, Edina, MN, 1991.

5. B. J. MILLIS and P. G. COTTELL, Jr., *Cooperative Learning for Higher Education Faculty*, Oryx Press, Phoenix.

6. A. W. CHICKERING and Z. F. GAMSON, "Seven Principles for Good Practice in Undergraduate Education," *AAHE Bulletin*, 3-7, 1987.

7. C. BAZERMAN and D. R. RUSSELL, eds. *Landmark Essays on Writing Across the Curriculum.* Hermagoras Press, Davis, CA, 1994.

8. J. EMIG, "Writing as a Mode of Learning," in *Landmark Essays on Writing Across the Curriculum,* Charles Bazerman and David R. Russell, eds., Hermagoras Press, Davis, CA,1994.

9. C. THAISS, *The Harcourt Brace Guide to Writing Across the Curriculum,* Harcourt Brace, New York, NY, 1998.

10. J. C. BEAN, *Engaging Ideas*: *The Professor's Guide to Integrating Writing, Critical Thinking, and Active Learning in the Classroom,* Jossey-Bass, San Francisco, CA, 1996.

11. W. PERRY, *Forms of Intellectual and Ethical Development in the College Years: A Scheme*, Holt, Rinehart and Winston, New York, NY, 1970.

12. W. PERRY, "Cognitive and Ethical Growth: The Making of Meaning," in *The Modern American College*, Arthur W. Chickering et al., eds., Jossey-Bass, San Francisco, CA, 1981.

13. L. R. WELCH, *CS 456: Software Design and Development*, http://zen.ece.ohiou.edu/cs456.html.