First Principles of CS Instruction

A perspective on CS Education

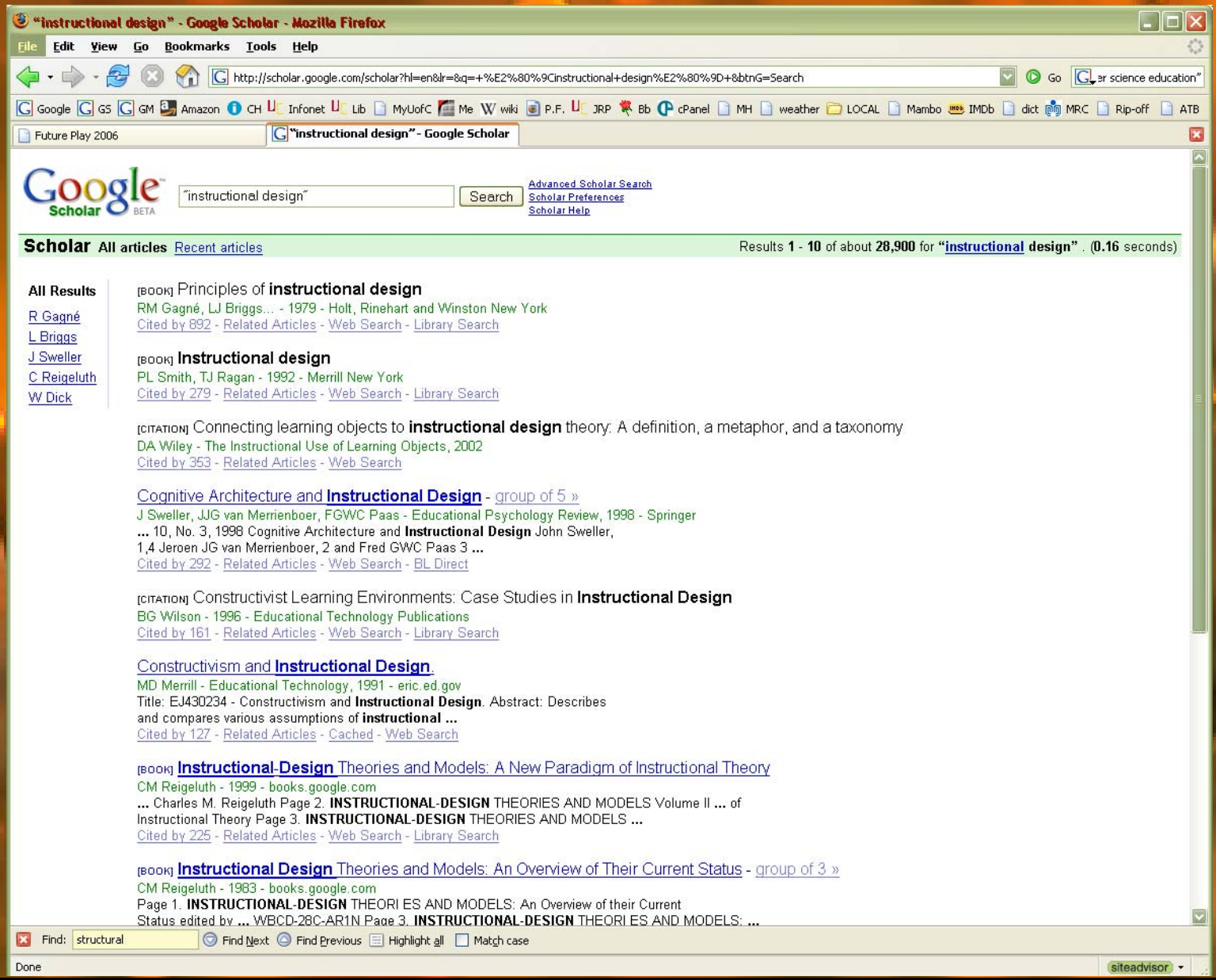
K. Becker

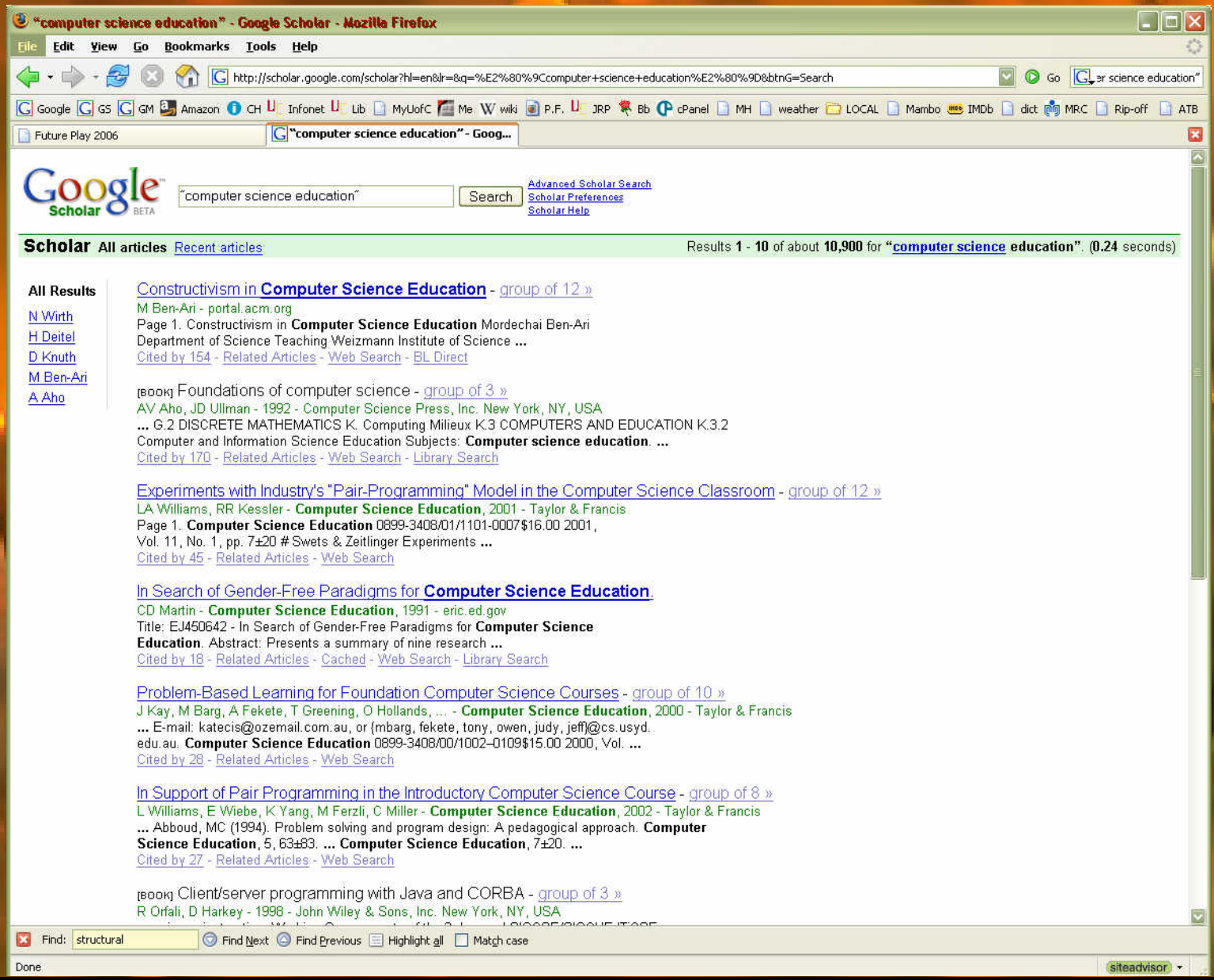
Overview

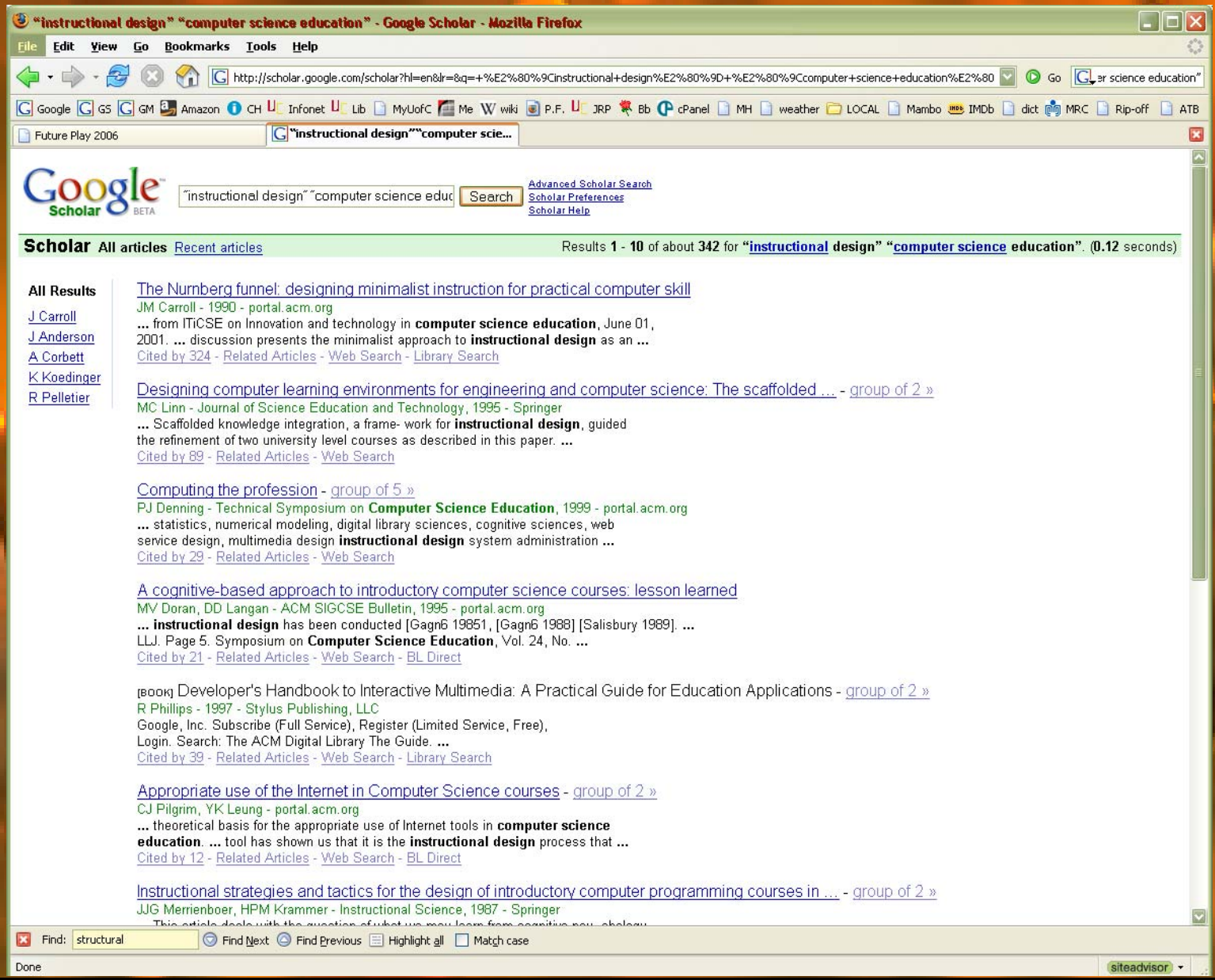
- Instructional Design & Computer Science Education
- M. David Merrill
- Merrill's First Principles of Instruction

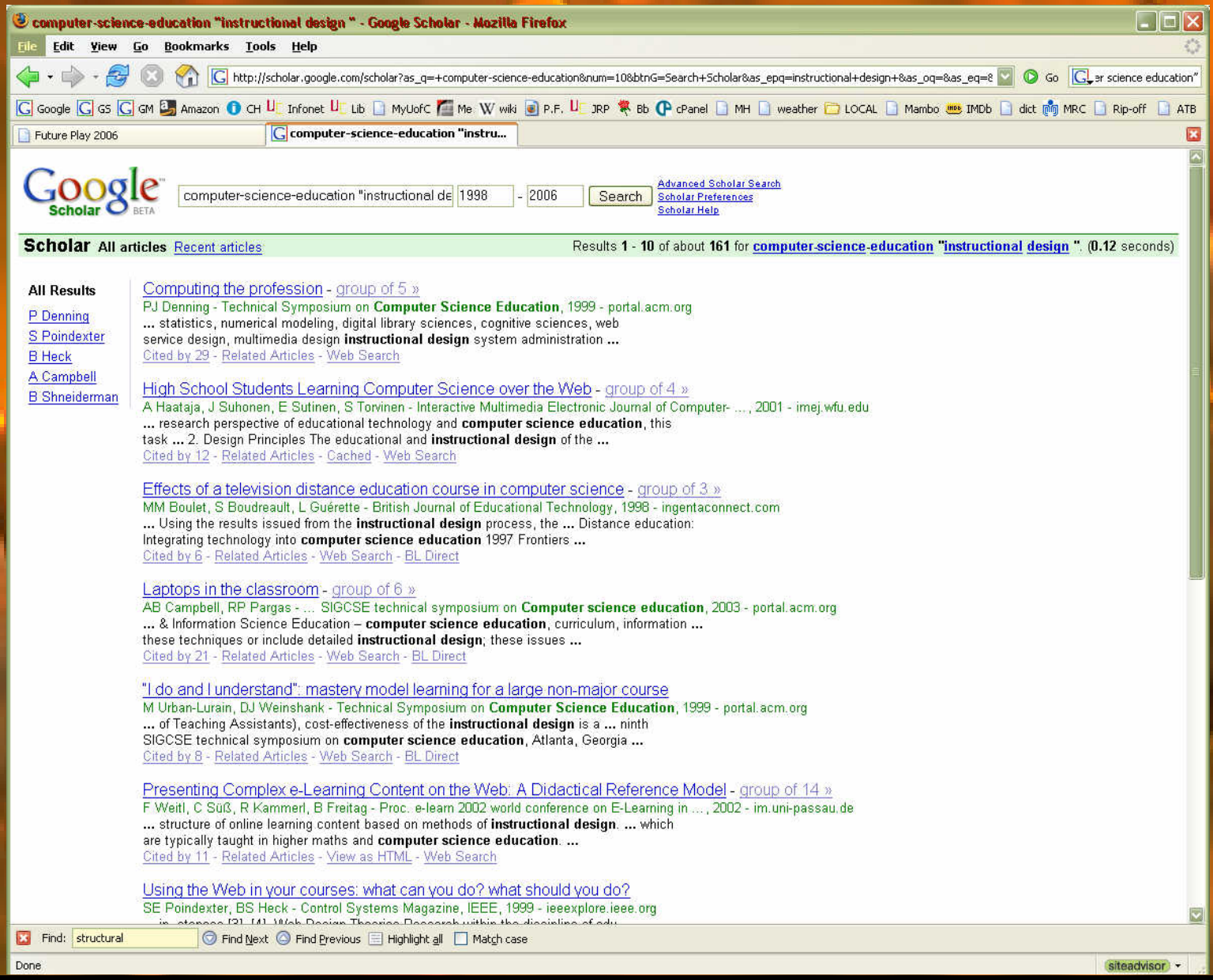
Instructional Design & Computer Science Education

- Learning and ID theories have had relatively little impact on how computer science is taught.
- A great many courses are still taught much the same way they were a generation ago.









Instructional Design & Computer Science Education

- Technology is used less as a system or tool for the support of learning and more as an apparatus on which to run programs and develop documentation.

M. David Merrill

- Component Display Theory
- Elaboration Theory (developed in collaboration with Charles M. Reigeluth)
- Instructional Transaction Theory
- the TICCIT CAI System
- First Principles of Instruction (2002)

PhD ’64 (Ed. Psych.) Brigham Young / Utah State

“If you don't provide adequate practice,
if you don't have an adequate knowledge structure,
if you don't provide adequate guidance,
people don't learn. “

Information is Not Instruction

- Many current instructional models suggest that the most effective learning products or environments are those that are problem centered and involve the student in a cycle of learning that involves four distinct phases:
  - activation of prior experience
  - demonstration of skills
  - application of skills
  - integration of these skills into real-world activities

Merrill's First Principles of Instruction

The success of a given instructional program will be directly proportional to how well and how deliberately the first principles are implemented.

Merrill's First Principles of Instruction

Problem

"Learning is facilitated when learners are engaged in solving real-world problems."

Activation

"Learning is facilitated when existing knowledge is activated as a foundation for new knowledge."

Demonstration

"Learning is facilitated when new knowledge is demonstrated to the learner."

Application

"Learning is facilitated when new knowledge is applied by the learner."

Integration

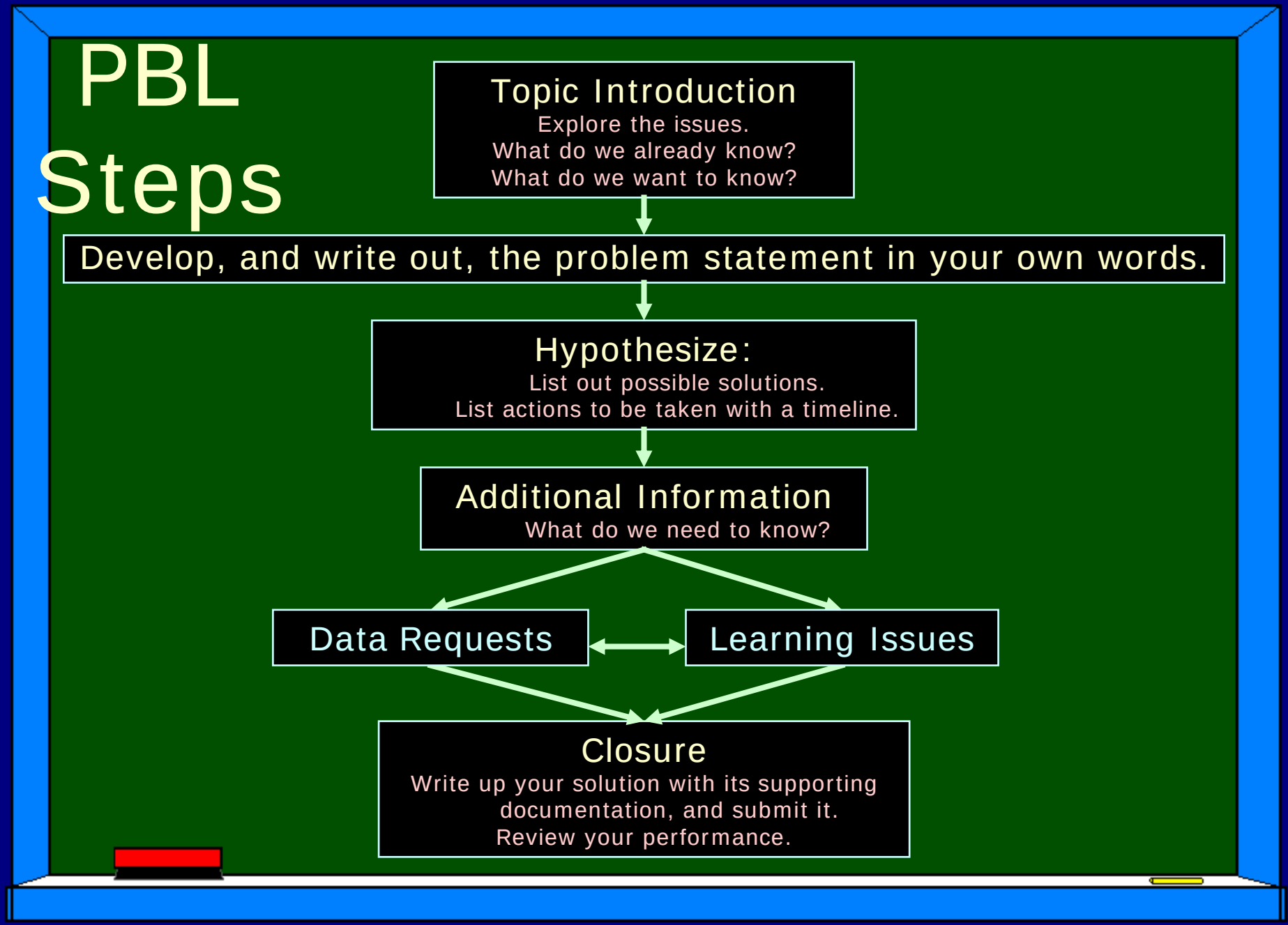
"Learning is facilitated when new knowledge is integrated into the learner's world."

Problem

engagement in solving real-world problems

- PBL is more deliberate than what we do.
  - structured, *guided* approach
    - delineation of the problem and its domain
    - research and discovery of relevant knowledge and data
    - presentation of a solution and a reflection on what was learned
    - deliberate and conscious connection with existing knowledge.
- **Accounting examples? Management? Employee Records?**
- look seriously at how students spend their time
  - Games, internet communication, music, …
- Offer choices (more for seniors / less for freshmen)
  - For example, the classic arcade game *Frogger*, a hospital emergency room simulation, and a dynamic restaurant menu system are all problems that lend themselves to the use of inheritance and polymorphism.
- If the pedagogical objectives are clearly defined, then the subject matter can become flexible, thereby creating relevance, and the potential for a personal connection with the problem.

PBL Steps


Topic Introduction
Explore the issues.
What do we already know?
What do we want to know?
Develop, and write out, the problem statement in your own words.
Hypothesize:
List out possible solutions.
List actions to be taken with a timeline.
Additional Information
What do we need to know?
Data Requests
Learning Issues
Closure
Write up your solution with its supporting documentation, and submit it.
Review your performance.

Problem

engagement in solving real-world problems

- Is the courseware presented in the context of real-world problems?
- Are learners shown the problem, engaged at the task as well as the operation level, and involved in a progression of problems?

Activation

activate existing knowledge as foundation for new knowledge

- To activate existing knowledge, we must know what that existing knowledge is.
  - cell phones, email, surf the web, find music and videos, and play games.
  - proficient, yet lack sophistication in searching and the critical assessment of resources.
    - acknowledge their skills, without either assuming too high a level of sophistication, nor too much nescience. One will discourage students, while the other will bore them. Either way, they will become disengaged and motivation will suffer.
- Example: Greedy algorithms by making change.
- Searching for a name in a phonebook as a means of introducing a binary search.
- *: relevance changes over time.*
- If we wish to start from where the students are, then we must be prepared to assess the knowledge they bring to the situation regularly.

Activation

activate existing knowledge as foundation for new knowledge

- Does the courseware attempt to activate relevant prior knowledge or experience?
- Are learners directed to recall relevant past experience or provided relevant experience?
- Are they encouraged to use some organizing structure?

Demonstration

- Learning objects can be useful here
- Live, in class demonstrations of programs are more effective than simply reading through and explaining code
- Modeling behavior
  - such as answering a question by performing a search on the internet, (including rapid assessment of potential resources)
- Develop a solution to a problem on the fly – including errors and blind alleys.
  - Maybe the 100th time for us, but the first time for them
- We learn a great deal from our mistakes and to some extent the mistakes of others.
- Allow our students to observe error recovery.

Demonstration

- Does the courseware demonstrate what is to be learned rather than merely telling information about what is to be learned?
- Are the demonstrations consistent with the instructional goals?
- Is learner guidance employed?
- Do media enhance learning?

Application
by the learner

- The CS cup runneth over with content.
  - The body of knowledge associated with the discipline has grown and evolved over the last 40 years and in our desire to provide students with as much information and knowledge as we can, we sometimes forget that they still need time to absorb the information - and this includes time for practice.
  - Chess masters, musicians, swimmers, and others have been shown to require on average ten years to achieve expert status!
  - What makes us think we can create expert programmers and computer scientists in just four?

Application
by the learner

- We can no longer cover the same ground in an undergraduate program as we did when we were students, yet we cling to a desire to do so, and to add all that we have learned since graduating as well.
  - Modern theories of education, including Merrill's imply that students will be better prepared through the acquisition of deep knowledge in fewer areas than through a shallow or cursory acquaintance with many.
  - We do our students a disservice by attempting to move on to the next topic too quickly.
  - One the other hand some topics can be taught using a spiral approach. The topic can be introduced but treated superficially in one course, and then addressed in greater depth in one or more subsequent courses.
    - Recursion, algorithm analysis, and program testing
  - Others, like ethics, professional practice, and communication skills can also be broken up and spread across multiple courses, but lend themselves more to sectioned, in-depth study than to layering from superficial overviews to deep learning.

Application
by the learner

- Do learners have an opportunity to apply their newly acquired knowledge or skill?
- Is the application consistent with the instructional goals, and does it involve a varied sequence of problems with feedback?
- Are learners provided with gradually diminished coaching?

Integration

new knowledge is integrated into the learner's world

- It is like coming full circle – we start from where the student is, and end by helping them convert new knowledge into a new starting point.
  - If the curriculum is well integrated, subsequent courses can quite literally pick up where the others left off.
  - If it is not, time will be required in each course to re-assess what knowledge the students bring.
- Typical : 5-10 distinct assignments.
  - There is no time in between to reflect, and in an effort to maintain interest, each problem bears little relationship to the last.
  - This kind of approach does not promote integration of learning.
  - While it is sometimes impractical to create assignments that logically follow one another, attempts should be made to do so.
  - When this is not possible, both practical and conceptual connections can be made explicit.
- At the course level, content can and should be tied to the students' lives, current events, research, and professional practice whenever possible.
  - For example, reading an article like Walpole's "Designing Games for the Wage Slave" connects well with student's lives, can be used to discuss software design on many levels, *and* can be used as the basis for assessment criteria on a programming assignment.

Integration

new knowledge is integrated into the learner's world

- Does the courseware provide techniques that encourage learners to integrate (transfer) the new knowledge or skill into their everyday life?
- Do learners have an opportunity to publicly demonstrate their new knowledge, reflect on their new knowledge, and create new ways to use their new knowledge?

Conclusions

- Courses can benefit from a more deliberate approach to the design of instruction.
- Freshmen have a broader base of experience than they used to, and integrating this experience into our courses forms connections that engages students.
- Contextualizing their experience draws them in, and maintaining relevance while guiding their practice will sustain them.

Learning *About*

vs.

Learning *To Be*

Merrill, M. D. (2001). First Principles of Instruction. *Journal of Structural Learning & Intelligent Systems, 14*(4), 459-466.

http://id2.usu.edu/Papers/5FirstPrinciples.PDF#search=%22First%20Principles%20of%20Instruction%22

Merrill, M. D. (2002). A Pebble-in-the-Pond Model for Instructional Design. *Performance Improvement, 41*(7), 39-44.

http://cito.byuh.edu/merrill/text/papers/pebbleinthepond.pdf