

How to Enhance a Distance Learning Web Site for International Students

Bradley Kjell

Computer Science Department
Central Connecticut State University
New Britain, CT
kjell@ieee.org

Heinrich Gailer

Bildugszentrum der stadt Nurnberg
Nurnemberg
Bavaria, Germany
hgailer@t-online.de



Proposition: Good Web Design Enhances Access by International Students

Most of the guidelines for creating an effective Web site also work to increase usability by international students.

Rules stress small modules, simple writing, interactive style, use of color and graphics.

Experiment: Computer Aided Instruction Web Site

Computer Aided Instruction Web site ([chortle](#)) created as an informal experiment in using these design guidelines.

If the site is successful, then the design ideas must be working.



About the CAI Web Site

<http://chortle.ccsu.edu>

Tutorials in various computer science topics:

- Java Programming (80 chapters, 1200 pages)
- QuickBasic Programming (25 chapters, 300 pages)
- Assembly Language Programming (36 chapters, 600 pages)
- Matrix Math for Computer Graphics (16 chapters, 250 pages)

Freely Accessible, ad free, downloadable



More About ...

German translation done by Heinrich Gailer ([Bildugszentrum der Stadt Nurnberg](#))

Translation is a really good means of finding problems (both muddled explanations and internationalization problems).

Courses are used at several dozens (or more?) universities and high schools.



Web Statistics

- . Site has existed for 6 years
- . 15 Million pages downloaded
- . 15,000 pages per day, recently
- . 40% outside of US
- . 25% from nations where English is not primary
- . German site delivered 350,000 pages in its first year.

Since the site is working, the design ideas are validated.



Unsolicited Comments

...an amazing Java Tutorial (I. Chan, NJ)

I was going thorough your notes and found them really helpful. (N. Chopra, India)

I saw your site and started study on it. it is excellent (S. Akcekaya, Turkey)

You have the most amazing tutorial I have ever seen, and it's so easy to understand (L. Teitelbaum, FL)

I have taken 100's of on-line tutorials and yours was without a doubt the best I have ever seen (J. DeRider, MN)

Your notes are absolutely the best.. (C. Ayfer, Turkey)

Totally super-mega-excellent lecture notes... (I. Kokkarinen, Canada)

General Web Design Guidelines

- Modular
- Structured
- Interactive
- Graphical
- Reader-centered
- Examples rather than discussion



Web Writing Guidelines (selected)

- Short, direct sentences
- Conversational tone
- Short paragraphs (much shorter than in textbooks)
- Active voice
- Structure (bulleted lists, headings, divisions)
- Consistent Navigation
- Few modifiers
- Simple tenses
- Words used in their most common form



Re-read and Re-write

Guidelines are simple and obvious.

Following them is hard.

Knowing well what you wanted to say often blinds you to what you actually wrote. (Much like programming)

If you read a Web page months after you wrote it, you see what you actually wrote. (Yuck!)

Working with a translator uncovers problems.

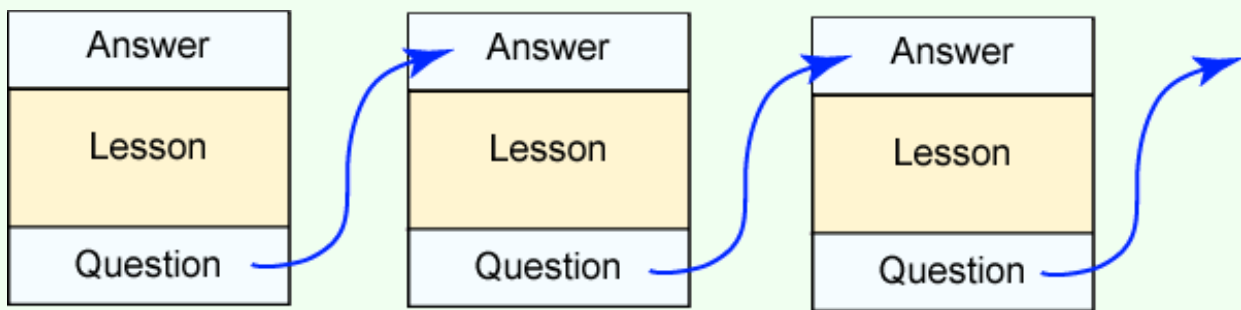
The Web audience is happy to provide long lists of problems.

Open-source courses? (future paper topic)

If my Web site is any good, it is because of much feedback, and many rewrites.

Programmed Learning

Sequence of short pages,
one concept per page,
linked together with questions and answers.



Implements Web Design Guidelines: Modular, Structured,
Interactive, ...



created: 12/23/02

CHAPTER 70 — Introduction to Recursion

This chapter introduces **recursion**. Previous chapters have discussed the fundamental control mechanisms of sequence, alternation (if-else), and iteration (while). Recursion is another control mechanism. Sometimes problems are stated in a recursive fashion, and a program that solves the problem is easily written using recursion.

Chapter Topics:

- Two Parts to Recursion:
 - Solve easy problems in one step.
 - Divide hard problems into smaller problems, and then solve the smaller problems.
- Examples of Recursion:
 - Walking a Distance
 - Smashing a Rock
 - Dividing a Line
 - Drawing a Snowflake

Forget programming for a while. Think about a real-world problem that does not involve computers:

QUESTION 1:

It is the day after Thanksgiving and the only parking spot you can find at the shopping mall is far from the entrance (horrors!) How do you get from your car to the mall?

Next

A good answer might be:

You walk.

How to cross a Parking Lot

The problem of crossing the parking lot is not insurmountable. You recall the instructions of the philosopher Lao Tse:

A journey of a 1000 yards begins with a single step.

(In the original, Lao Tse probably did not have mall parking lots in mind.)

So you bravely face towards the mall entrance and take a single step.

QUESTION 2:

What do you do next?

Next

A good answer might be:

Take another step.

Slightly Closer to the Goal

The instructions: *a journey of a 1000 yards begins with a single step* tell how to reduce the size of the problem. The problem is how to cross the parking lot. After taking a single step, the problem is slightly smaller.

Of course, now you reduce the size of the new problem by taking yet another step. Before long, the problem is so reduced in size that it can be solved in one step.

QUESTION 3:

How do you know when the "cross the parking lot" problem has been solved?

Next

A good answer might be:

The problem has been solved when you are at the mall entrance.

Two Parts to Recursion

Of course you know how to cross a parking lot. You would probably describe the process as "...keep walking until you reach the entrance."

Let us describe the solution in more detail:

1. If you are one step from the mall, take that step and you are done.
2. If you are further than one step from the mall, divide the distance into two parts:
 - o a single step, and
 - o the remaining distance.

Now take a step and then cross the remaining distance.

I've described the solution in a recursive style. There are two parts to recursion:

1. If the problem is easy, solve it immediately.
2. If the problem can't be solved immediately, divide it into smaller problems, then:
 - o Solve the smaller problems by applying this procedure to each of them.

The over-all strategy is to find some easily applied action that breaks the problem into smaller pieces. Some of the pieces can be dealt with immediately; others need to be further broken up.

QUESTION 4:

Say that you have a small rock that you wish to destroy. How can you do this?

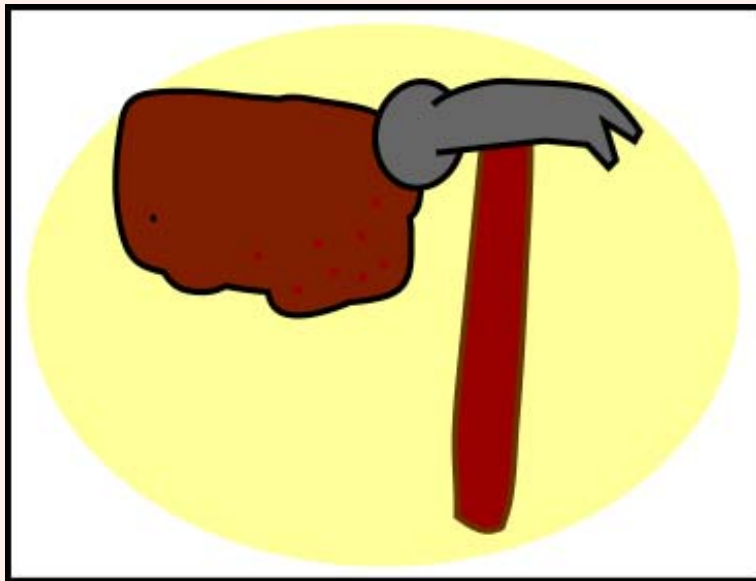
Next

A good answer might be:

You may destroy the rock by pounding it with a hammer.

Pounding a Rock to Dust

You have a rock. You have a hammer. The size of the problem "destroy the rock" may be made smaller by applying the hammer to the rock.



Click on the Image to Apply Recursion

After applying the hammer, you now have several smaller problems. Each of these smaller problems may be solved by applying the hammer.

QUESTION 5:

When do you stop applying the hammer?

Next

A good answer might be:

You stop applying the hammer when each sub-problem (rock) is small enough.

Recursion on the Rocks

Here (again) are the two parts to recursion:

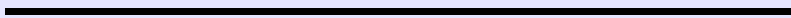
1. If the problem is easy, solve it immediately.
2. If the problem can't be solved immediately, divide it into smaller problems, then:
 - Solve the smaller problems by applying this procedure to each of them.

Here are how this applies to the rock-destruction problem:

1. When a piece is small, don't pound it any further.
2. To destroy a large rock, hit it with a hammer. The rock shatters, leaving small and large pieces.
 - Apply this procedure to each of the pieces.

QUESTION 6:

Say that you wish to divide a line into 16 equal pieces.



What is the first thing that you do?

Next

Internationalization Guidelines (selected)

- Avoid Americanisms, popular culture, holidays, politics
- Avoid American currency, history, geography, sports ...
- Avoid cliches
- Problems expressing numbers 1,000,000.99 and dates 6/8/04
- Problems expressing time, 1:34 PM EST

Unfortunately, following these guidelines drains the writing of its interest and humor. (e.g. Starbucks)

I like writing as if I am lecturing. Humor and allusions to pop culture help in the classroom.

The International audience for this site are probably interested in how an American professor teaches.

So I do not follow these guidelines. But I do think about them.

It is the rare citizen of the world that does not understand the American dollar.

Audio

Each page of the Assembly Language course (on [chortle](#)) is supplemented with audio.

Intended for a distance learning course for students in Vietnam offered by a Thai university, taught by an Indian professor from the University of Wisconsin.

But this project fell through (Americanism?)

800 MP3 files, approx. 2 minutes each.

Very popular. Mostly in US. Not clear why.



Audio Production

Little good advice on adding audio to Web pages.

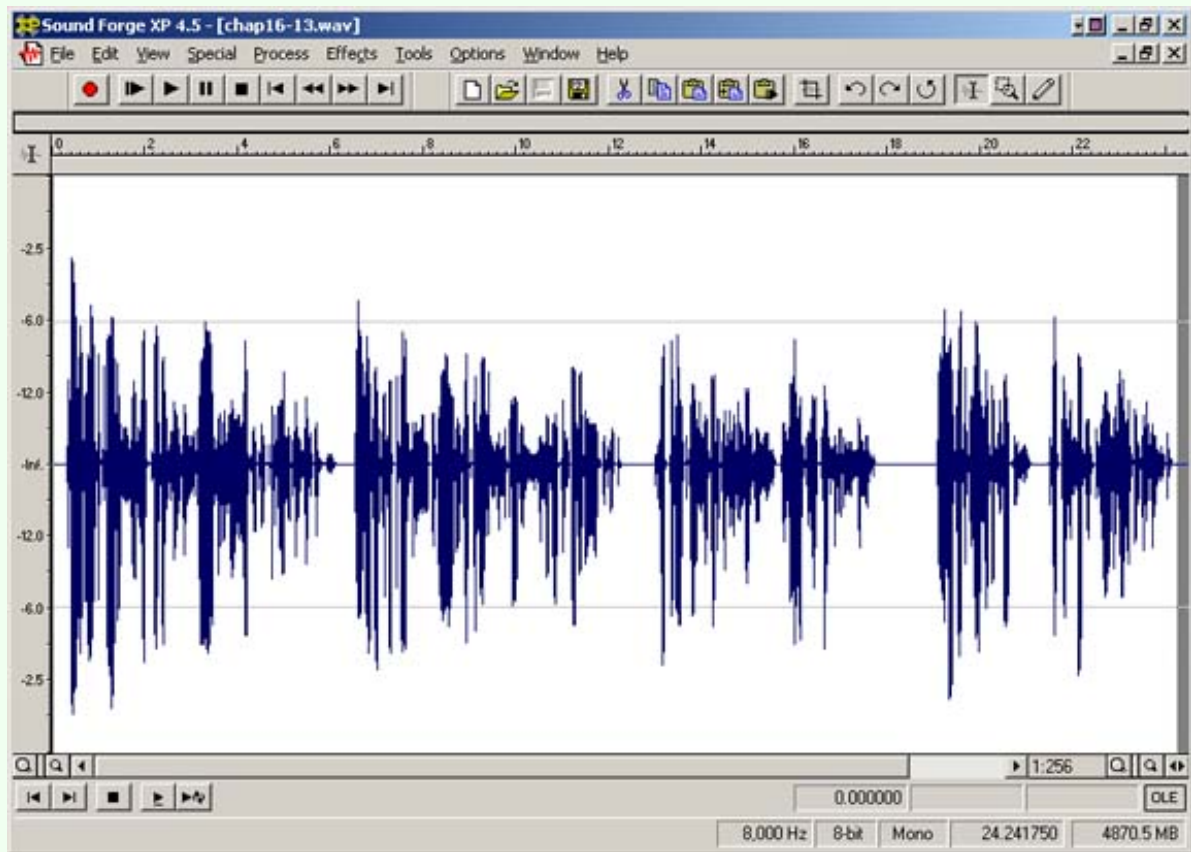
What I do:

- Sound Forge XP (but Adobe Audition might be better)
- record .wav files at CD quality (44,100Hz, 16 bits, mono)
- RMS power about -20db
- Cut and paste the audio
- Convert to 8,000Hz, 8 bit, mono .wav files
- Convert to MP3 files (30% the size of .wav files)

Good sound board is essential (build in audio not good enough)

Takes about 8 minutes for each Web page.





Conclusions

The usual guidelines for Web site design appear to work.

Programmed learning is an effective technique for the Web.

Internationalization guidelines are often too strict.

Audio instruction is very popular, possibly underutilized.

