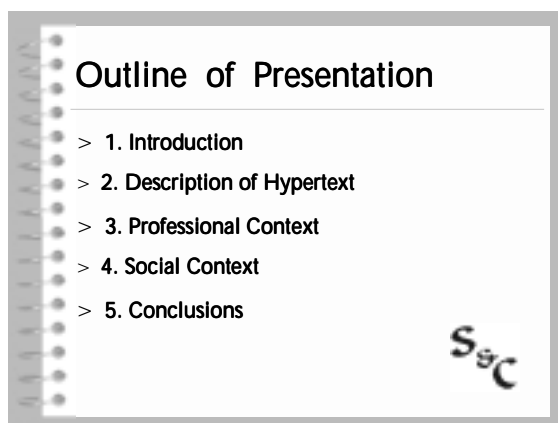


I. Introduction

Personal Info. Tech Writer / Communicator since 1984. Prior (going backwards): librarian, mediævalist, researcher, journalist. I didn't plan this broad perspective I backed into it.

Tie-in to Conference Theme. Tech Communicator much more rewarding and satisfying than Tech Writer: better pay; higher status; more impact on the project (therefore more satisfaction when it's done). (I experienced same improvement when I went from Librarian to Information Broker, and mastering relevant technology was the key.)

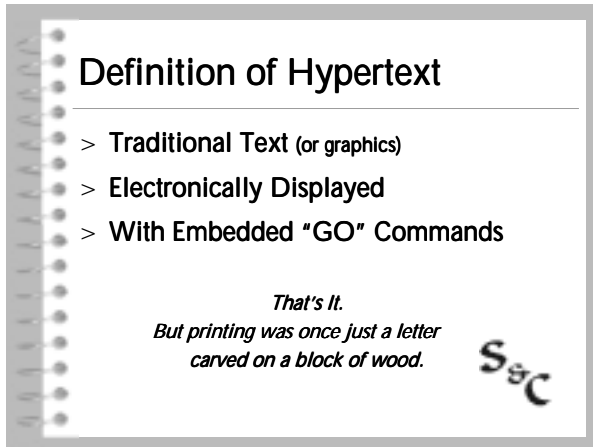
Plan of Action. We've got four more topics after this brief introduction:



Invitation to Dialog. I'm serious about this stuff, and I love dialog. Contact points are under the bio at the end of these notes, and our newsletter is listed in the "Resources" Supplement (in § 1-Per, under Scribble & Count).

II. Description of Hypertext

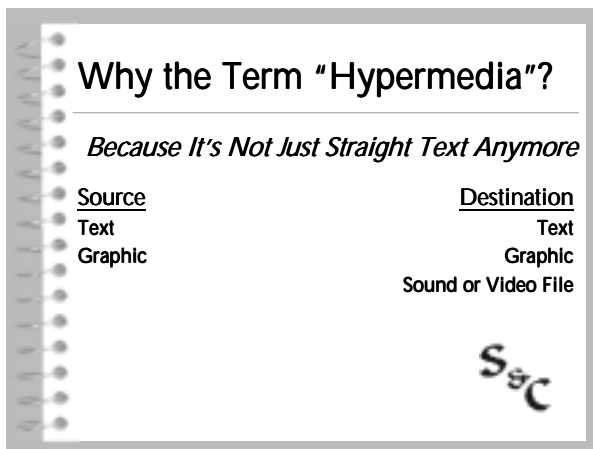
Definition. Hypertext has three simple elements.



This is important, and we'll come back to it:

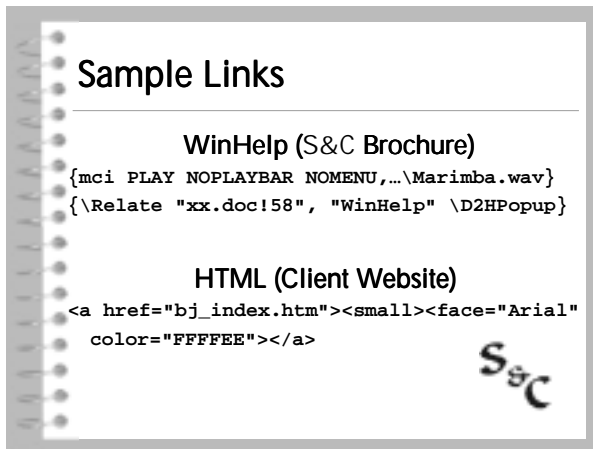
Printing revolutionized society with a very simple invention; if hypertext does 1/10 as much, it will keep you busy for a dozen careers.

Now, what's this term hypermedia?

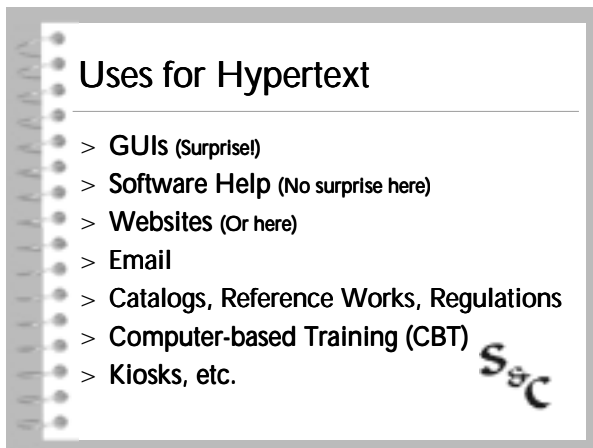


When "hypermedia" is used, normally an emphasis on the A/V component is intended; and when "hypertext" is used, you can generally tell whether it's used generically, or whether text only is meant.

Sample Links ("GO" Commands). This is just to give you a quick flavor if you've never seen it. It's programming, sort of, but maybe only 1/3 as complex as BASIC, or 1/5 the difficulty of C++. You can do this, particularly if you try it at home first.



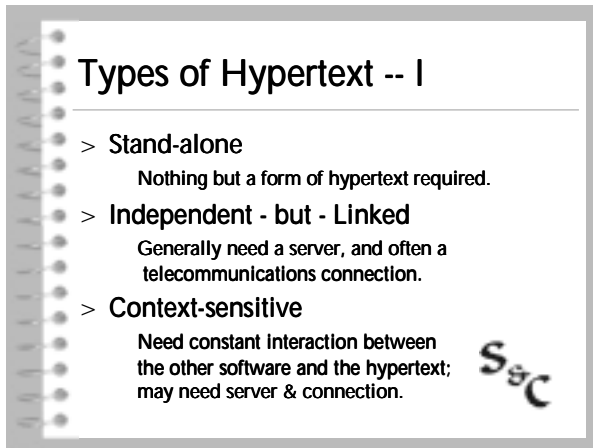
Uses for Hypertext / Hypermedia. I'll bet a couple of these will surprise you.



It's around the versions of hypertext developed for items #2 and #3 that the Hypertext Wars are raging. We'll touch that briefly later.

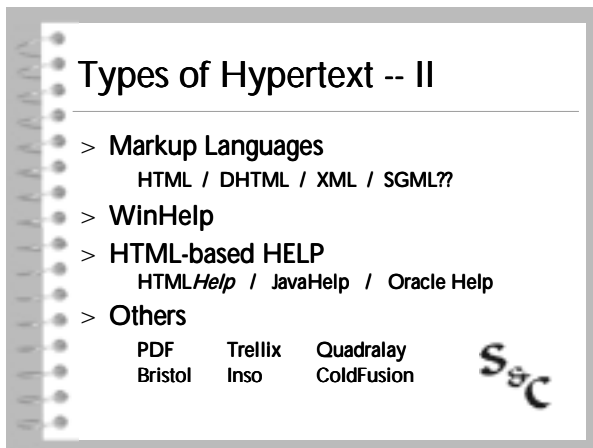
Make an experiment: Try to go through a day recognizing every instance of hypertext you see. Then, a little while later, spend a day identifying *places where hypertext isn't used, but should be*. You may look back on the time as some of the best you've ever spent in professional development.

Types of Hypertext (Method 1). It seems to me valuable to divide hypertext into three different types:



The subordinate, indented, line is the important one; don't attempt a project unless your skills and company/client infrastructure support it.

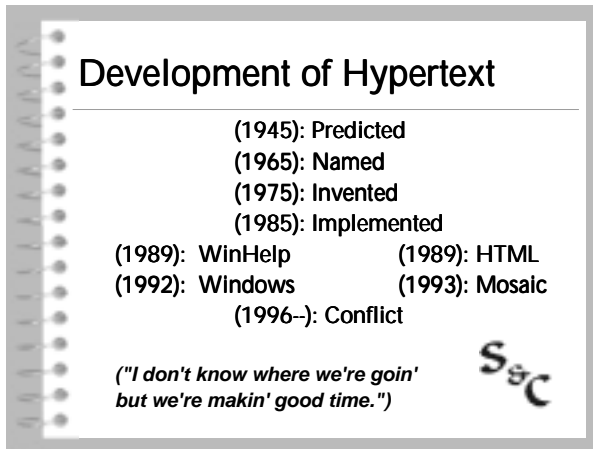
Types of Hypertext (Method 2). This division you've probably seen before:



The value here is to illustrate how confusing the stage has become (did Cecil B. deMille script this?). The next slide sketches, very briefly, how the situation developed; the following slide ("Essentials") describes what hypertext does well.

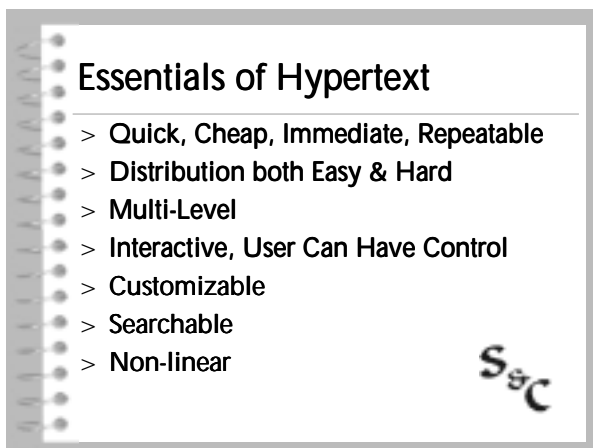
Then you've got some professional decisions to make; the rest of this presentation and the Supplement should help. (If not, don't let me come back and speak again.)

Development of Hypertext. The way hypertext developed explains a lot of the infighting.



What you need to get here are 1) how short this development has been, and 2) the simultaneous (and independent) development of the Big Two: HTML and WinHelp. (For the background on the quote, see **Cassady** in the "Quick-Reference" Suppl.)

Essences. This may be the most important slide of all; if you know the essential elements of something, you can do a lot with it. We'll spend the rest of our time building on this slide.



Quick, Cheap, Immediate, Repeatable. These are all economic issues (which argue that hypertext is no fad); the "quick" aspect has huge social implications as well.

Distribution both Easy & Hard. Hard (and initially expensive) because user has to be properly connected. After that, it doesn't get any easier.

Multi-level. You can write several documents at once, and by following or ignoring links, users create their own customized document out of millions of possibilities.

Interactive / User Control. If know the effects of passive TV, you know this is big.

Customizable. The days of Henry Ford's "one color fits all" mass production are over.

Searchable. I've indexed materials both ways; the best is still done manually by an expert, but you can do 1,000 times more 90% as well with automation. You decide.

Non-linear. Huge. As McLuhan said it would be. But not immediate.

III. Professional Context

Hypertext will change our professional lives in three ways:

- How We Write
- What Else We have To Know
- How We Manage Our Careers.

The bibliographical references refer to the "Resource List" in the Supplement; § 2 unless noted.

How We Write. Everything (well, *almost* everything) you learned as a tech writer you can keep; it will all be useful. But develop these styles and techniques more than you have so far:



I'll deal with more than are on this slide.

Audience. You often choose topic types according to who your audience(s) will be – and because you can write on several levels in one document, you'll often be dealing with more than one audience. [Deaton & Zubak. *Designing...*]

Planning. Organization and outlining commandments you learned 5th grade composition will, finally, come true. Only now it's called "storyboarding" and "housekeeping". If your project is of any significant size, you will *not* be able to avoid the disciplines this time. Plot your topics *and record what goes where if your software doesn't do it for you.*

Simplicity. Screen pixels are grainy; the user's position is not ideal; and the screen is narrow in scope. So you have to be simple and clear. "Chunking" means keep your topics small (unless it's a reference work, a max of one screen is the goal). Consistency means put the same stuff in the same place and order each time. The screen provides your user with no field of vision, so they have to be able to anticipate. [Boggan. *Developing...*]

Independence. The topics not only have to be simple, they've got to be as independent as possible, because you rarely have a clue how your reader will get to any sentence you write. (Writers who are control freaks have a real problem here.) The answers: good planning, and more repetition than you learned was acceptable. [Horton. *Designing...*]

Hypertext-Unique. Make your links clear, both in where they are, and what they do. Avoid print terminology ("described above" doesn't make much sense when a person entered the topic from somewhere else.) Most of the print terminology to avoid is related to the linear nature of print. [Horton. *Designing...*]

Indexing. Nearly all hypertext has a lot of built-in indexing. Write so that you use it.

What Else We Have To Know. As a print writer you had to know a little design, color was useful, and there were two or three software packages you needed. It's the same here, but significantly more demanding.



Hardware. This gets back to knowing your audience: what limitations will they have that you must consider? Screen size, resolution, and color are among the most important.

Telecommunications. Audience again, this time from speed and security perspectives. Like hardware above, Intranets are wonderful because you know the physical limits. [**Held.** *Understanding Data Communic...*]

Scripting. Learn to script (Active-X, JavaScript, CGI, Perl, Tcl), and learn markup. And learn to read code. [**Graham.** *HTML 4.0...*; **Turner.** *Readme.1st...*; **Connolly.** *XML...*]

Info Mapping. Sometimes related to storyboarding (below). In software documentation, another meaning is the linking of help topic numbers to parts of the program.

Flowcharting / Storyboarding. Outline, outline, outline. But flowchart or storyboard in between. Make the writing and the outlining/flowcharting an iterative process, at least until changes stop occurring (if ever). [**Horton.** *Designing & Writing...*]

Color / Graphics. Know what your users' equipment can handle, know what it takes to push something through their connection, and don't use either color or graphics without an explicit, defensible purpose. You'll have plenty, even after you prune ruthlessly. [**Siegel.** *Creating...*; **Horton.** *Web Design...*; **Horton.** *Icon...*; **Kent.** *Poor Richards...* – all in § 2.]

Multimedia. If you need it (and the warnings about overuse above apply in spades here), you'll have to spend a fair amount of time learning it; it doesn't bundle with other skills.

Analysis. Learning analytical skills may be the most important basic after English. It doesn't matter what you've learned to analyze as much as the generic technique. (If you've been a Research Analyst, as I have, it stays on your resume even after it's 30 years old.)

More Software Packages. You'll need most, if not all, of the following software: word processor; DTP; graphics; capture (if not in graphics); WinHelp and HTML editors; communications; both browsers;....

What Else? If you know anything about this field, you can add a dozen items I omitted. Like reading specs.

It's almost like becoming a Renaissance person.

How We Manage Our Careers: Predicting the Future. Did I mention metaphysics as one of those support fields a minute ago? Sometimes it seems that way.



Seeing Uses for Hypertext. We discussed this earlier: try to identify where hypertext is used -- and where it isn't but should be. When you see a kiosk in a motel, try to analyze how it works. When you read about sensors or jelly beans, think about applications for hypertext authors. It's good practice.

Anticipating the Future. These may or may not be logical subdivisions. They merge.

Try to judge (and act upon) the direction of **Technological Change**. We all know that a general trend is toward miniaturization, but do we act on it? Aren't we surprised at the dramatic increase in computing power? Software developers are never ready to take advantage of the new hardware. If we think in terms of anticipation rather than the status quo, we'll be a little less behind the curve. What should we know about telecommunications that would change the way we write? Don't pass up free or inexpensive classes, even if they seem only tangential to your perceived direction. This should take a lot of your time.

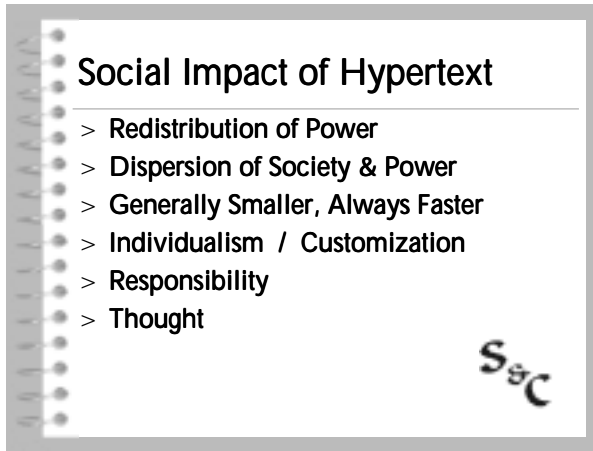
Stay up on the **Competing Hypertext Standards**. You may or may not need to know SGML, but know its strengths (powerful, robust) and its weaknesses (learning curve, expense, limited to big companies). Do the same thing for XML, DHTML, etc. The same thing is true of our software; constantly evaluate it: RoboHELP/Doc-To-Help/ForeHelp; HomeSite/ColdFusion/Net Objects/FrontPage. Can you make do with Paint Shop PRO, or do you need the much more expensive PhotoShop?

Finally, think about the **Hypertext Wars** being waged between Microsoft and what's been called the Gang of Four (Sun, Oracle, Netscape, and IBM). With the Justice Dept., make that 5. The Titans could decide tomorrow to split the trillion-dollar pie -- but until then, think about what the battle means for the future of information distribution and hypertext. A Java World is different from a Windows world.

Professional Involvement. I'm not addressing networking here; it's a valid concept, but I don't have anything to add. What I'm talking about is the sources of information to help you manage your careers. The STC and ACM are both good examples: their publications are put out by leaders who know the importance of career development. Read their product reviews, attend their seminars.

IV. Social Context

I'm sure time will be short when I get here, so this list is a mere sketch.



Power. Traditional Industrial Age groups (such as labor) won't automatically retain their position; previous "have-nots" (such as rural America and developing societies) can gain power if they take advantage of the opportunities.

Dispersion. The history of Western Civilization has been that of concentration: wealth, political power, production, demography, in that order. This is ending; big entities are breaking up.

Size and Speed. Smaller entities (whether governmental or economic) will have an advantage. When size and diversity are important, *ad hoc* alliances will be formed to accomplish a specific purpose. Some multinational companies (especially Telcos) may seem to get bigger, but the operating units will normally be smaller and more nimble.

Individualism. Hypertext is almost infinitely customizable; and since software drives most production, physical goods are following. Whether we'll get any more self-centeredness (whether it's *possible* to get any more) is uncertain.

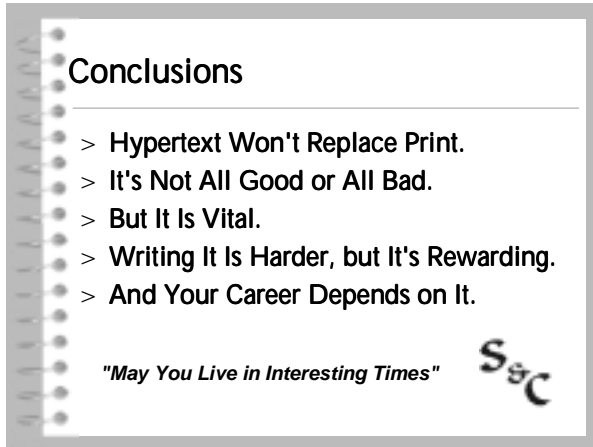
Responsibility. The judgement here is mixed. A linear (print) argument holds one's feet to the fire; that's why demagogues and salespeople hate to put it in writing. Hypertext, with its frenetic jumping around, doesn't do this well. But the speed of hypertext and telecommunications has frequently held governments accountable by disseminating information overnight. A good news / bad news story if there ever was one.

Thought. This is the least concrete aspect, but maybe the most important. Marshall McLuhan (§ 4) says that how we receive a message is as important as its contents. All previous communications has been in one direction – that chosen by the author or producer. Hypertext is immediate, tends toward scattered discourse, promotes active participation rather than passive reaction, and encourages individualism. This sounds like good news / bad news also.

Several people (Bill Gates and Charles Lynch among them) have said, with out qualification, that hypertext would affect society as much as printing has.

V. Conclusions

No one in 1500, not even the highly erudite English printer Bill Caxton or the Spanish lexicographer Covarubias, had a clue how far the revolution would lead. This was true for at least 250 more years. Comparing hypertext to printing, it's only 1520 or 1550; most of the revolution is still ahead.



Print Will Be Fine. There's no Faustian choice here; we need both hypertext and linearity. Story telling is linear. Cause and effect are linear. It will be a long time before a complex, fold-out schematic can be satisfactorily reproduced in pixels. Few people curl up with their laptops. "I love you" in hypertext loses something. This presentation was oral, the handouts were paper. Don't let the alarmists spook you.

Hypertext Not All Good or All Bad. Hypertext provides immediate data retrieval – but exacerbates the attention span problem. It provides immediate feedback – but that includes irresponsible flames. It foils government censorship – but makes it difficult to enforce child pornography laws. I firmly believe that hypertext is more good than bad; and that's a good thing, because it's here to stay.

Hypertext Is Vital. Kevin Kelly [§ 4] makes a good case that this isn't an *information* revolution (that has been going on for more than a century); it's a *communications* revolution – the transfer of information. Telecommunications and hypertext are going to remake your world as much as printing did.

It's Hard, but Rewarding. You saw all the adjunct fields that were involved with hypertext; mastering one or two and acquiring familiarity with most of the rest is no easy matter. But hypertext authors get paid more. They aren't treated (as often) as academic valets. And they generally have much more to do with the products they document than does a print author.

Your Career Depends on It. At least if you're under 40, it does. Have you done a survey recently of your want ads? How many of them *don't* require a knowledge of hypertext? And more than half of available jobs are announced online; what percentage of them do you think *don't* require hypertext?