5

Language of the Internet

NAOMI S. BARON

NOTE: A revised version of this chapter appears in Ali Farghali, ed. *The Stanford Handbook for Language Engineers*. Stanford: CSLI Publications, pp. 59-127.

1 Introduction

Over the past twenty years, the Internet has radically transformed the way people communicate, both locally and globally.[1] This chapter examines the linguistic tools we use to make this communication possible.

1.1 Chapter Goals

Our discussion of the language of the Internet is divided into seven sections. The goals of each section are as follows:

Section 1 Introduction
- lay out the essential terms of discussion relating to language and the Internet (1.2)
- introduce the need for using natural language processing to facilitate working on the Internet and the World Wide Web (1.3)

Section 2 Linguistic Issues
- consider the distinctions between speech and writing (2.1)
- understand the historical impact of technology on written and spoken language (2.2)
- introduce relevant cross-linguistic and cross-cultural issues (2.3)

Section 3 Natural Language Usage on the Internet
- define computer mediated communication (CMC) and introduce its various types (3.1)
- analyze the workings of CMC (3.2)

Section 4 Coding Systems for Communicating via the Web

[1] As with so many aspects of the computing world, the nature of the language of the Internet is rapidly evolving. This chapter represents a linguist's perspective from late 2002. I am grateful to Ali Farghaly and Vincent Ribière for their assistance at various points in the development of this chapter.

- summarize the types of Web coding systems (4.1)
- present an overview of Web markup languages (4.2)
- introduce Web programming languages (4.3)
- introduce Web quasi-programming languages and applications programs (4.4)

Section 5 Web Search Systems
- introduce traditional search engines and strategies for being "found" on the Web (5.1)
- introduce intelligent agents and the Semantic Web (5.2)

Section 6 Cross-Linguistic Challenges on the Internet
- outline the challenges of working on the Internet in a multilingual world (6.1)
- outline the problems of using machine translation on the Internet (6.2)
- outline the problems of dealing with multiple written scripts on the Internet (6.3)

Section 7 Further Reading
- suggest resources (print and online) for learning more about the language of the Internet

1.2 Language, the Internet, and the World Wide Web

The title of this chapter, "Language of the Internet", can be interpreted to refer to four different senses of the word "language":

(1) natural language usage carried via the Internet (e.g., in email, listservs, Web pages)
(2) special coding systems for constructing communication carried via the Internet (e.g., markup languages, programming languages, applications programs)
(3) natural language, coding system, or translation interfaces used for gathering information from the World Wide Web (e.g., search engines, intelligent agents, machine translation programs)
(4) special terminology used in talking about the Internet (e.g., networking, servers, browsers, HTTP)

This chapter is devoted to the first three of these senses of the term "language".

When discussing how natural language and special coding systems are used in contemporary networked computing, it is important to be aware of the distinction between the Internet and the World Wide Web (WWW). The Internet, whose historical antecedents predate the WWW by roughly twenty years, is a number of computers that are concatenated into a communication network. Many Internet functions (e.g., email, FTPs [file transfer protocols]) are independent of the WWW. In contrast, the WWW, created by Tim Berners-Lee in the early 1990s, is a collection of software and protocols that are designed to make it easier for computers to communicate across the Internet. With the coming of the Web, a number of communication functions (such as email) that

predate the Web (and even the modern Internet) are now predominantly Web-based. Figure 1 presents an historical timeline summarizing the major developments that made possible computer networking as we know it today.

-- Insert Figure 1 Here --
Networking Timeline

Generally speaking, when we talk about "language and the Internet", we are referring to language issues that arise in constructing natural language to be carried across the Internet (e.g., email, chat, the content portion of Web pages). When people talk about "Web languages", they usually mean coding systems (e.g., HTML, JavaScript) for constructing Web pages. At the intersection of these domains is the problem of figuring out how to design search procedures and coding systems that enable end users to employ natural language input (typically a word or phrase) to locate information posted somewhere in cyberspace on one or more of the billions of Web pages that exist.

1.3 Challenges for Natural Language Processing (NLP)

Information is of little value if you cannot find what you are looking for. A mislaid book or a document buried in a heap of papers is as good as lost if you need its contents now.

The information retrieval problem is an old one, dating back at least to the great libraries of Alexandria and Pergamum in the third and second centuries BC (Casson 2001). Early classification schemes were often based on the order in which a library acquired a volume – a cumbersome system that was still maintained, in part, by the Library of Congress and the British Library well into the second half of the twentieth century.[2]

Categorization schemes based on subject matter (e.g., the Dewey Decimal call number system, the Library of Congress call number system) would seem to offer better information retrieval. However, Aristotle notwithstanding, there is no "natural" set of categories into which matter or knowledge is divisible. Therefore, library classification schemes (or, these days, choices of key words for finding materials in online databases) are essentially best guesses as to how users are likely to categorize the books or information they are seeking. In a world of small libraries (or bookstores), if the categories made no sense, one could always ask a librarian or proprietor, or simply walk through the stacks. Today, however, we are increasingly dependent upon navigating databases ourselves to find hardcopy books and articles housed in traditional libraries.

Such library challenges are multiplied many hundred-fold as we try to navigate the World Wide Web. Not only is the amount of material being posted to the Web expanding exponentially, but there is no commonly agreed upon set of categories into which that material is sorted. Hence, the retrieval problem becomes truly daunting. Realistically, the only solution for coping with the embarrassment of Web riches is to rely on automatic processing systems that can extract relevant information from natural

[2] For example, until online cataloguing was introduced, patrons not physically at the Library of Congress in Washington who wished to determine if the Library owned a particular volume needed to look through multiple sets of the National Union Catalogue, organized in chronological year blocks.

language text and then, in some cases, categorize, summarize, or even translate such information from one language to another. The question then becomes, are existing natural language processing (NLP) programs adequate for the task?

Most of our successful models for doing natural language processing (or machine translation) presuppose that the. subject matter, grammar, and vocabulary being processed are constrained. Written texts are assumed to follow basic grammatical conventions, as well as allowing for a defined set of special conventions (e.g., abbreviations, truncation of full sentences such as "Have you the time?" to "Got the time?"). Moreover, parsers are designed to expect written materials to employ conventional spelling and punctuation.

Natural language usage on the Internet is anything but conventional or constrained. No grammar teacher or subject classifier vets the billions of emails, instant messages, chat contributions, or Web pages that shuttle across the Internet daily. Instead, we find language that is fragmentary, laden with typographical errors, often bereft of punctuation, and sometimes downright incoherent. We also find text dealing with every subject imaginable. If search engines are ever to effectively apply NLP techniques to the breadth of natural language material that appears on the Internet, we will need to understand what sorts of linguistic patterns we are likely to find on the Internet and how they relate to the more conventional written and spoken language that NLP programs now handle.

2 Linguistic Issues

Since much of the language we find on the Internet is intended as natural language usage (e.g, a posting to a listserv, the home page of corporation), it is important to be aware of the linguistic considerations that come into play when natural language is transmitted via the Internet. The linguistic dimensions we will consider are

(1) speech versus writing
(2) the effects of technology on language
(3) cross-linguistic and cross-cultural issues

Most of the issues surrounding natural language usage on the Internet are not unique to the medium. Rather, the variables shaping such usage are familiar from studies in linguistics concerning transmission modality (here, speech versus writing), discourse participants, style, transmission medium (e.g., spoken language delivered face-to-face vs. over the telephone) and culture. An understanding of these issues is important for anyone hoping to tackle the challenges of creating Web search tools that work successfully with the natural language appearing on the Web.

2.1 Speech versus Writing

The overwhelming majority of natural language appearing on the Internet is written (i.e., as opposed to auditory). However, Internet users have often commented that written language on the Internet, especially in email or now instant messaging (IM), is more like

speech than like writing (see Baron 1998, Crystal 2001). Before analyzing such comments (much less assessing their veracity), we need to understand the relationship between speech and writing as modalities through which to formulate and convey human language.

2.1.1 Writing as an Object of Linguistic Analysis

For most of its history, modern linguistics has taken spoken language as its essential object of study. Writing, said Leonard Bloomfield, is only a speech surrogate, "merely a way of recording language by means of visible marks" (Bloomfield 1933:21). Both European and American structuralists resorted to the written word essentially as a necessary evil, using historical documents to reconstruct earlier linguistic stages or transcribing (i.e., committing to written form) contemporary speech for use in synchronic analysis. For Noam Chomsky's transformational school and its derivatives, who restrict their inquiry to the linguistic competence of the ideal speaker-hearer, written texts are essentially irrelevant.

Over the past few decades, a growing number of linguists have argued that written language is a distinct modality worthy of both independent analysis and formal comparison with speech (see, for example, Baron 1981, Chafe and Tannen 1987, Biber 1988, Olson 1994, Taylor and Olson 1995, and Harris 2000, along with such journals as *Visible Language* and *Written Communication*). Historical comparisons of the evolution of spoken and written language reveal that within a given sociohistorical context, the relationship between speech and writing may change over time. In the case of English, for example, writing has gone from largely functioning as a re-presentation of formal speech (c. 800 AD to c. 1600), to being a significantly independent linguistic modality (c. 1600 to c. 1950), to now commonly re-presenting informal speech (see Baron 2000).

2.1.2 Distinguishing between Speech and Writing

Despite historical evolution in the functions of written language, we can still define basic parameters in terms of which written and spoken language generally differ from one another – see Figure 2.[3] These distinctions can be divided into two major categories: form (i.e., how is the linguistic act constructed or perceived) and content (i.e., what is the substance of the linguistic message). Keep in mind that these are paradigmatic distinctions, not intended to describe all empirical spoken or written formulations. (Notes passed during a class are likely to be more informal than a Senator's maiden speech in Congress.) Nonetheless, by identifying distinctions between the two linguistic modalities, we have a baseline against which to compare specific contemporary uses of written language, such as on the Internet.

-- Insert Figure 2 Here --
Distinctions between Written and Spoken Language

[3] Figure 2 incorporates elements from Baron 2000:21 and Crystal 2001:26-28.

Each of the variables identified in Figure 2 is relevant to characterizing the written natural language appearing on the Internet. However, for our purposes we focus on issues relating to discourse participants and style.

2.1.3 Discourse Participants

While writing typically takes the form of a monologue, speaking is more often dialogic. Writing is monologic because in principle, the writer cannot know in advance who his or her potential audience might be. Admittedly, most writing done by the average literate person (e.g., letters, reports) is directed to specific individuals or groups of individuals. Yet by virtue of writing's durable nature, even targeted documents may find their way to wider audiences. Moreover, professional writing (e.g., journalism, novels, academic papers) is intentionally crafted with the aim of reaching an unidentified audience – generally the larger the better. Many of the common attributes of traditional written text (e.g., decontextualization, formal tone, tendency to be edited before being distributed) follow from the fact that writers typically strive to make a good impression on both known and unknown readers.

The discourse assumptions underlying face-to-face speech diverge from those associated with traditional writing. Face-to-face speech is commonly dialogic, although in practice there may be multiple participants. In either event, the speaker knows who the participants are, and face-to-face discourse allows for immediate feedback. The nature of participation in traditional writing versus face-to-face spoken discourse will prove important for understanding the kind of natural language used on the Internet.

2.1.4 Style

Three of the content variables identified in Figure 2 – formality, internal structuring, and editing – help define the broader linguistic notion of style. It is commonly said that natural language usage on the Internet is more casual than traditional written language. But what does the word "casual" mean here, and is Internet writing actually more "casual" than contemporary off-line composition? It is important to be aware that the trend towards casual use of written American English is at least half a century old, long predating networked computing. The same tendencies towards being conversational in tone, mechanically sloppy, and sometimes lacking in logical coherence are found in both online and off-line composition (Baron 2000).

2.2 Effects of Technology on Language

One of the important forces that can help shape both written and spoken language (and the relationship between them) is the emergence of new technological means for producing, recording, or transmitting language. To understand whether the Internet is affecting the way we write (and, derivatively, the kinds of language parsers we need to process such writing), it helps to reflect upon how technology has affected language in the past.

2.2.1 Effects of Writing Technologies

Technological developments have long affected the nature and use of written language. The establishment of print technology eventually fostered not only standardization of spelling but the growth of literacy in Western Europe (Baron 2000: 99, 83-91). Successful marketing of typewriters at the end of the nineteenth century fueled invention of the business memorandum (Yates 1989), contributed to the decline of American handwriting skills, and encouraged the production of more lengthy typewritten prose (compared with handwritten text – see Haefner 1932), a trend also evident in the early years of word processing (Stoddard 1985).

Some effects of technology on writing are more subjective. The transition from the manuscript of medieval Europe to a print culture in early modern Europe entailed a shift from the notion of text production as a serial enterprise with contributions from multiple authors to a view of a text as the fixed work of a single author (Bruns 1980:113). Today, some scholars working on composition theory argue that thanks to the technology of the Internet, textual composition can (and should) return to the medieval model of collective composition (where the contributions of individual authors are not always formally acknowledged), and texts should be seen more as works in progress than as finished products (e.g., Lunsford and Ede 1994, Howard 1999, Bolter 2001).

2.2.2 Effects of Speech Technologies

Alexander Graham Bell's introduction of the telephone in 1876 made possible two shifts in the way people communicate orally. The first shift affected the kinds of spoken messages speakers were likely to formulate. Since telephone conversations do not entail face-to-face encounters, many speakers are less hesitant expressing themselves on the phone than when in physical proximity with their interlocutor. More than a century of familiarity with the telephone appears to have contributed to the increased candor commonly reported in computer mediated encounters such as email or online conferencing, compared with face-to-face exchange (Sproull and Kiesler 1991).

The second shift concerns access to interlocutors. Before the invention of the telephone, the average person had no opportunity to insinuate his or her way into conversation with, say, members of the upper class, bank presidents, or politicians, all of whom had their visitors screened. However, in the early days of the telephone, gentlemen and business owners complained that "any person off the street may for a trifling payment … ring up any [telephone] subscriber and insist on holding a conversation with him" (Marvin 1988:103). Open access has been a key feature of the Internet. Consequently, unless users make concerted efforts to conceal their email addresses, email users have almost limitless access to one another, radically redefining previous rules for constraining social discourse.

2.3 Cross-Linguistic and Cross-Cultural Issues

Human linguistic exchange is shaped not only by modality and technology but by linguistic and cultural diversity. These issues are important in working with language on the Internet because failure to be aware of subtle cultural considerations can wreak havoc

on what may, at first glance, seem to be straightforward tasks of writing Internet content, parsing texts, or constructing translation programs.

2.3.1 Language Choice and Translation Issues

There are approximately 5000 different languages in the world today. Historically, there have been four basic approaches to the challenge of communicating across these linguistic boundaries:

(1) remaining monolingual (i.e., not communicating across linguistic boundaries)
(2) multilingualism (i.e., learning the language of another language community, such as German speakers learning French)
(3) creation of a contact language (e.g., a pidgin or creole, such as Tok Pisin in Papua New Guinea or Krio in Sierra Leone)
(4) adoption of a lingua franca (i.e., a common language that may be the native language of few or none of the specific discourse participants, such as historically happened with Latin and French, and is now happening with English). While some non-native speakers may become fluent in the full-fledged version of the lingua franca, others may function with a restricted subset of the language (e.g., Airspeak for international aviation, the somewhat simplified version of English used by Voice of America, and what some describe as an emerging simplified version of English used by millions of non-native speakers in communicating across linguistic boundaries) (see Crystal 1997b)

Since the Internet is a global system, issues relating to linguistic diversity are critical in determining how to deal with natural language carried via the Net.

Although the Internet developed as an English-based network, the use of other languages has rapidly been expanding. There is the well-known projection (appearing in a 2001 airport billboard advertising Accenture Consulting) that by 2007, the dominant language on the Internet will be Chinese.[4] This projection is hardly surprising in light of the estimate that by the year 2050, there will be nearly 1.4 billion native speakers of Chinese but only slightly over 500 million native speakers of English (Graddol 1997:27). To the extent that languages other than English proliferate on the Internet, the task of using NLP systems to parse messages greatly increases because such systems will require not only NLP parsers but linguistically satisfactory translation programs as well.

As of now, the Internet seems to be heading simultaneously in two directions regarding language diversity. The first is to produce increasing amounts of content in languages other than English. This trend poses serious translation challenges for language engineers with regard to searching text created in other languages or creating adequate translations for sites that wish to be multilingual (see Section 6 below).

The second trend is towards adoption of English as the lingua franca of the Internet. However, two issues arise with this solution. On the one hand, since the majority of English speakers in the world already are non-native users of the language (Crystal 1997b), content writers for the Internet cannot assume that even the majority of readers

[4] www.powerinvestdrips.com/archive/01feb07.htm

will understand complex grammatical constructions, idioms, or less common vocabulary. On the other hand, even among native speakers (or speakers of English-based pidgins or creoles), there remains the question of which dialect of English to select.

2.3.2 Dialect Choice

The famous Peter Steiner cartoon that appeared in the *New Yorker* in 1993 had one canine confiding to another, "On the Internet nobody knows you're a dog". As of now, the overwhelming majority of natural language text carried across the Internet is written, thereby neutralizing differences in the spoken-language accent structure of Internet users from, say, Dallas, Dublin, and Delhi. However, other tell-tale differences between English dialects reveal themselves in writing: through spelling, vocabulary, and occasionally grammar. Is the word spelled *color* (American English) or *colour* (British English)? Does a mail-order company *send* a product or *dispatch* it? When you are sick, do you go "to the hospital" or "to hospital"?

Search engines and parsers are unlikely to have much difficulty handling differences between the major dialects of English. However, members of distinct cultural communities may be highly sensitive to linguistic nuances. A single spelling of the word *colour* immediately signals "foreign" to an American reading the word on the Internet.

2.3.3 Cross-Cultural Issues

Written distinctions encoding dialects of the same language are just one aspect of the broader issue of how different cultures vary in their presuppositions regarding composition of language sent across the Internet. In many societies (e.g., China, France), there have traditionally been strict distinctions between the style of language acceptable in face-to-face spoken communication and in formal letter-writing. Even in more laissez-faire societies (such as the US), there are cultural conventions governing spoken versus written usage.

The Internet challenges these traditional assumptions in several ways. First, because users are themselves confused whether Internet communication follows the conventions of spoken or written language, there is considerable uncertainty (on the part of both senders and recipients) over what linguistic register is appropriate. Second, among native English speakers from different cultural backgrounds, we find considerable diversity regarding appropriate Internet usage conventions. And third, among the millions of Internet users who are non-native speakers of English, limited knowledge of usage conventions in written English can make for messages that seem rude or disrespectful to native English-speaking recipients.

Among the culturally-based issues of writing style that arise with natural language usage on the Internet are these:

- **audience**
  Are there people to whom it is inappropriate to send particular messages (e.g., a freshman biology student emailing a Nobel prize winner for help with her homework)?

- **content**
  Are there some messages that should not be sent over the Internet (e.g., condolences) but only in a formal off-line letter? Are there some messages that should only be delivered face-to-face (e.g., dissolving a personal relationship)?

- **form**
  How formal or informal should the style be (e.g., should you use a salutation in email, when should you use first name versus title plus last name)? Is the message edited before it is sent? (In some cultural contexts, unedited messages are perceived as rude.)

3 Natural Language Usage on the Internet

Section 2 laid out parameters we need to keep in mind when dealing with natural language usage that happens to be conveyed via the Internet. We now turn to Internet language itself by

(1) defining computer mediated communication (CMC)
(2) analyzing CMC

3.1 What is Computer Mediated Communication (CMC)?

The notion that computers can be used as language transmission devices presupposes that computers are networked with one another. As we saw in Figure 1, networking for the purpose of transmitting completed documents, data, and computer programs dates back to the late 1960s. By the early 1970s, the earliest experiments were underway for transmitting messages in natural language intended for specific individuals or groups of recipients.

3.1.1 Definition and Timeline

Computer mediated communication (CMC) is loosely defined as any natural language messaging that is transmitted and/or received via a computer connection. Generally speaking, the term CMC refers to a written natural language message sent via the Internet. However, the term can also be applied to other written venues that employ computer-based technology to send messages across a distance, including both email and computer conferencing done through in-house intranet systems and contemporary short text messaging (SMS), which is normally transmitted through mobile phone connections.

Figure 3 presents an historical timeline indicating the chronological appearance of different forms of computer mediated communication.

-- Insert Figure 3 Here –
Computer Mediated Communication Timeline

We now look, in turn, at each of the major CMC venues, grouping them conceptually rather than chronologically. Section 3.1.2 presents the notion of a CMC spectrum, while Section 3.1.3 introduces each of the major types of CMC.

3.1.2 The CMC Spectrum

We can think of CMC as a cover term encompassing a range of writing options. At one end is writing that resembles traditionally composed (off-line) texts, the difference being only the means of transmission. At the other end is dialogue between two people that highly resembles speech, again, save for the medium of message exchange. If we think of traditional writing as a "product" (in the sense of being a finished work) and face-to-face speech as a "process" (in that a conversation is typically a work in progress, with the outcome being determined by interaction between participants), we can lay out a spectrum of CMC – see Figure 4.

-- Insert Figure 4 Here --
Computer Mediated Communication Spectrum

At the far left ("product") end of the CMC spectrum are completed works such as academic papers or business reports that are available through a personal or organizational Web site or through attachments sent via email. Since the language style and editing assumptions of such completed works are largely indistinguishable from those pertaining to formal off-line writing, we will not deal further with this category in this chapter.

As we move to the right of the spectrum, all of the other categories of CMC show linguistic adaptation of the written medium as a result of being formulated for Internet transmission. The first kind of adaptation involves fluidity: the farther to the right of the spectrum we move, the greater the fluidity of the CMC message. That is, we expect, for example, a Web page to be more of a finished product and instant messaging to be more of an ongoing discourse process. However, as the Internet matures as a transmission medium, both user expectations and industry software tools are changing. As a result, Web sites that used to be essentially static products are becoming increasingly interactive, making use of some of the newer markup languages and other Web design tools (see Section 4 below).

The second adaptation of the written medium involves anonymity: in the last three categories on the right of the spectrum, the issue of participant identity becomes blurred. Historically, chat, MUDs, and MOOs invited anonymous participation (involving not just pseudonyms but invented identities). Listservs were reserved for "vetted" (i.e., registered and perhaps even screened) participants, and email was exchanged between interlocutors who already knew one another or who accurately introduced themselves when initiating communication with a stranger. However, these boundaries are increasingly shifting. On the one hand, many chat, MUD, or MOO forums now vet their users. On the other hand, large numbers of individuals create multiple email accounts to mask their identity from certain mail recipients. What is more, messages (whether on listservs, email, or IM) are easily forwarded to individuals not on the original recipient list, thereby distributing

information to unintended participants. Such forwarding may result not only in breaches of confidentiality but in distribution of hastily composed messages not intended for public scrutiny.

3.1.3 Types of CMC[5]

Here are the major types of CMC, organized with respect to the extent to which they represent dialogue or monologue. A second important distinction to keep in mind is whether the communication is asynchronous (i.e., participants do not have the potential to interact together in real time) or synchronous (i.e., real-time communication).

One-to-One Dialogue

- **email**

Email ("electronic mail") is an asynchronous form of CMC, prototypically between a single sender and single recipient. However, contemporary email systems permit multiple recipients, along with forwarding of a message one has received to third parties.

The email function arose out of experimentation by Ray Tomlinson, a computer engineer working at Bolt Beranek and Newman, the firm hired by the US Department of Defense to build ARPANET. Tomlinson's first test message, an arbitrary string of letters, was actually sent between two PDP-10 computers in the same room that were connected via ARPANET. To clarify the recipient and machine location to which a message was addressed, Tomlinson selected the @ symbol, which separated a user's login name from the name of his or her computer, e.g., jcaesar@rubicon.gov.[6] Very quickly, email became the most prevalent use of ARPANET, and eventually emerged as the "killer ap" of networked computing.

Figure 5 presents a sample email exchange between two American university professors attempting to coordinate a luncheon date.

-- Insert Figure 5 Here--
Example of Email

There is enormous variation in the language style used in email, determined by such variables as age and computer experience of user, function (e.g., replacement for a formal office memo, casual invitation to lunch next week, teenage online flirting). Email is, in principle, not intended for public view. Therefore, the kind of language used in email (sometimes ungrammatical, lacking in standard punctuation or spelling) should not be an issue for natural language processing or search procedures, since such mail is usually only stored on an individual user's computer (and perhaps, in the case of a company, in the organization's back-up files), and not the subject of mechanical analysis. However, given the enormous popularity of email, many of the informal (even careless) writing

[5] My thanks to Elizabeth Constable, Jessica Marks, and Anne Godlasky for their efforts in gathering examples of computer mediated communication.
[6] See Campbell 1998 for a history of the first email.

conventions that have emerged in writing email are finding their way into other CMC venues that are intended for wider audiences.

- **instant messaging**

Instant messaging (IM) is a synchronous form of CMC that, like email, is prototypically utilized between a single sender and a single recipient. Given the synchronous nature of the communication, IM messages tend to be quite short and even more casual than email.

One-to-one synchronous communication systems have been in use for some time, dating back at least to the 1980s and early 1990s with the use of such UNIX applications as "talk", "ytalk", and "ntalk", and the Zephyr IM system through Project Athena at the Massachusetts Institute of Technology.[7] However, IM did not become a widespread phenomenon until the late 1990s, thanks in large part to the technology and marketing efforts of America Online (especially AIM – AOL Instant Messenger) and Mirabilis Ltd's IQC ("I Seek You"). It has been estimated that as of the end of 2002, there were 1.38 billion instant messages sent daily using AOL's network.[8] While AIM is heavily used in the United States, IQC – which has 120 million registered users – is predominantly used outside of the US. Note that ICQ, which first appeared in 1996, was purchased by AOL in 1998.[9] Other contemporary players in the IM market include Yahoo (Yahoo! Messenger) and Microsoft (MSN Messenger).

Over time, IM systems have added many bells and whistles, intended either for convenience or to attract young users. For example, recent versions of AOL's AIM enable users to create individual profiles (in essence, brief online biographies) and away messages (to indicate that you are temporarily away from your computer and what you might be doing), along with buddies lists (revealing which of your IM "buddies" are presently online). As with many youth-oriented products, these features, along with their actual functions, can be expected to shift rapidly with time.

Figure 6 presents an example of IM messaging between American college students using AIM:

-- Insert Figure 6 Here –
Example of Instant Messaging

While instant messaging captivated teenage and young adult audiences in the years immediately surrounding the turn of the century (e.g., Lenhart *et al*., 2001), the newest markets for instant messaging are in business applications (Hellweg 2002). It was predicted by the International Data Corporation that "the corporate IM market will grow … from 5.5 million users worldwide in 2000 to 180 million in 2004. By that time, the number of messages sent will approach 2 trillion annually."[10] The Gartner research firm

---

[7] See http://web.mit.edu/olh/Zephyr/Revision.html. I am grateful to Alan Sondheim, Sheizaf Rafaeli, and Karim Lakhani for background on early IM systems.
[8] Anyone can send an IM using AIM, whether or not a paying member of the AOL network.
[9] The source of these data on AOL and ICQ is http://corp.aol.com/whoweare/who-datapoints.html (November 28, 2002).
[10] Cited in *InfoWorld*, October 25, 2000. Available at http://www.infoworld.com/articles/hn/xml/00/10/25/001025hnidcim.xml

has projected that by 2005, use of instant messaging (presumably in business) will have surpassed use of email.[11]

As in the case of email, as long as instant messages remain in the private domain of sender and recipient, the language used in such exchanges does not become the concern of language processing and/or search programs. However, given the nature of corporate activity (including not only intranet messaging but, for example, major expansion of already existing IM systems on commercial Web sites), it becomes increasingly probable that business enterprises will look to parsing and search programs that will be able to make sense of these rapidly composed (and often cryptic or garbled) forms of electronic messaging.

- **SMS**

The abbreviation "SMS" formally stands for "short messaging system", though it is generally interpreted as meaning "short text messaging". SMS is used on mobile telephones throughout much of the world, though market penetration in the United States still remains small by comparison.[12] Messages are generally created by tapping the numbers of the phone keypad one or more times, corresponding to the letter of the Roman alphabet that is intended.[13] Thus, for example, "U" (a common SMS abbreviation for the word "you") would be generated on the phone's display screen by tapping the number "8" twice rapidly in succession, since the "8" key historically also bears the letters "T", "U", and "V". Letters were originally used on phone sets to represent the telephone exchange (i.e., the word appearing at the beginning of a telephone number). For example, my phone number as a child was "GR 4-2525", with the "GR" standing for "Greenbelt", the name of the town having that exchange. Today, the same number would be "474-2525". Americans who are not familiar with SMS nonetheless use the same system of multiple taps on the numerical keypad to program their mobile phones.

SMS was developed in Europe, first appearing in late 1992. The protocol was developed as part of a multinational European effort known as GSM (Group Spécial Mobile) that was constituted to establish a uniform mobile telephone system for Europe. Over time, "GSM" has come to mean "Global System for Mobile Telecommunications", as the historical origins of the system receded in users' minds and as the GSM protocols for mobile telephony spread worldwide. In mid 2002 it was estimated that more than three billion SMSs were sent each month in Europe alone.[14]

The US has been slow to adopt SMS for a variety of reasons. First, Americans already have entrenched one-to-one messaging systems based on networked computers, i.e., email and IM. Second, while Americas are now catching up with other parts of the world in market penetration of mobile phones, they still lag behind. And third, unlike Europe (and other GSM adopters), the US has no single mobile phone protocol. As a

---

[11] http://www.newsfactory.com/perl/story/18008.html (May 31, 2002)

[12] Surveys in late 2002 suggest that SMS use is growing rapidly in the US (see, e.g., Romero 2002). However, usage seems to be higher among teens and young adults than in other age cohorts.

[13] In the case of languages such as Japanese, which employ ideographic characters in their written system, the process is more complex: capitalizing upon the fact that Japanese (as Chinese) has many homonyms, the user types in a Romanized representation of the word and then chooses among several characters (each graphically – and semantically – different) that correspond to the phonological form of the word.

[14] www.m-indya.com/mwap/sms/sms.htm

result, the American population of mobile phone users has not had access to a shared mobile phone texting system. (Think of the historical software compatibility issues of Macs and PCs.)

Figure 7 presents some examples of short text messages on mobile phones between a Londoner who works in publishing and several of her friends.

--Insert Figure 7 Here—
Examples of Short Text Messaging

In many ways, the language of SMS is reminiscent of that seen in instant messaging: short, full of abbreviations, and casual.

Technically, SMS is not computer mediated communication, since it was designed to be sent and accessed through mobile telephones (via satellite technology), not through computer networks. However, in recent years, many digital technologies have become interchangeable platforms for transmitting and receiving linguistic messages: email messages can be accessed on mobile phones; SMSs can be sent – and received – on computers. In the coming years, it is likely that as platforms become increasingly interchangeable (and as Americans become heavier users of mobile communication devices), the kind of language appearing in email, IM, and SMS will tend to become more homogeneous: short, informal, and full of space-saving devices such as abbreviations and truncated syntax.

One-to-Many Dialogue

- **listservs**

Listservs (also sometimes known as mailing lists or distribution lists) are asynchronous, text-based communication sent by a single user to multiple email addresses. In its simplest form, a listserv provides a forum for a single individual to send a message (e.g., announcement of a meeting) to two or more recipients. Frequently, however, postings are made by multiple members of the mailing list, thereby providing an electronic forum for discussion. Today, listservs are commonly used by professional organizations, academic classrooms, or groups sharing common interests, enabling individual members to voice opinions or raise questions. Lists may be unmoderated (postings are automatically distributed without review by anyone) or moderated (someone collects messages received over a short period of time and edits them in some way before posting – e.g., summarizing the topics, summarizing the contents of the posts, or censoring objectionable material).

Mechanisms for distributing text to multiple recipients are almost as old as the early ARPANET. Early discussion lists appeared by the mid 1970s (e.g., MsgGroup, used by ARPA researchers), and listservs became an important component of BITNET, the networking systems created by Ira Fuchs and Greydon Freeman at the City University of New York in 1981. As the popularity of mailing lists spread in the 1980s, software written by Eric Thomas in 1986 (and named LISTSERV) helped automate the process of actually maintaining such lists (e.g., adding or deleting members, posting and distributing messages).

Figure 8 provides an example of a lightly moderated listserv from the Association of Internet Researchers (www.aoir.org). The figure illustrates how postings are organized, along with the first message in the day's offerings.

-- Insert Figure 8 Here --
Example of a Listserv

Because listservs constitute written, archived, and often quasi-public text, the potential for needing to search them with NLP tools is significant. However, the task may be less daunting than for, say, email or IM, since the language in many listservs (especially in academic or business circles) is more formal and grammatical than in one-to-one CMC.

- **newsgroups**

Newsgroups are public forums for asynchronous one-to-many dialogue that originally were designed to be accessed through USENET (a non-governmental network developed in 1979 at the University of North Carolina). Unlike listservs, which send messages directly to all users on a distribution list, newsgroups constitute postings to a common public site, which can be accessed whenever users choose to log on.

The network of different newsgroups is vast. Tens of thousands of available newsgroups represent seemingly every topic imaginable, from sex to antique cars to medicine. Because newsgroups are written, publicly posted, and archived, they invite textual analysis. However, unlike listservs, newsgroups are neither moderated nor restricted in membership. As a result, the language appearing in posts can vary enormously, both in style and propriety.

Since the days when newsgroups were all accessed through USENET, newsgroups have been hierarchically organized into major categories, each of which is then subdivided. Currently, major divisions include "comp" (computer science subjects), "humanities" (humanities subjects), "misc" (miscellaneous topics), "news" (news topics), "rec" (recreational topics), "sci" (science topics), "soc" (sociological subjects), and "talk" (controversial topics).[15] Each category is then further subdivided. For example, rec.arts.tv.soaps (also known by its acronym r.a.t.s.) is a newsgroup for discussing television soap operas. Figure 9 presents an example of a posting to r.a.t.s.

-- Insert Figure 9 Here --
Example of a Newsgroup Session

- **MUDs and MOOs**

MUDs (originally meaning Multi-User Dungeons; now commonly interpreted to mean Multi-User Dimensions) are synchronous environments in which multiple players interact with one another in a textually-created imaginary setting. The first such adventure game was created in the late 1970s by Roy Trubshaw and Richard Bartle at the University of Essex. The early versions of such games drew heavily upon "Dungeons and Dragons", a

[15] www.livinginternet.com

popular board game from the early 1970s. At the time the early MUDs were created, computers had very limited graphics capability. Players were necessarily restricted to verbal descriptions of both scenes and actions.

Unlike newsgroups (which talk about the world that is, using asynchronous posts), MUDs allow a comparatively restricted set of users to synchronously act on situations of their own construction. Players assume pseudonyms and interact according to pre-established navigation rules for moving through a defined terrain. Figure 10 offers an example of a piece of a MUD session, drawn from a paper by Pavel Curtis and David Nichols.

-- Insert Figure 10 Here --
Example of a MUD Session

For their first decade, MUDs were heavily dominated by male players engaged in fantastical adventure games. By the late 1980s and early 1990s, the use of MUDs began expanding to include wider ranges of participants and more social functions. During this same period, object-oriented programming was introduced into MUDs, yielding the concept of MOOs (MUDs, Object Oriented), so named by their creator, Stephen White at the University of Waterloo.[16] In 1990, Pavel Curtis at Xerox PARC added several features to White's program, creating the well-known LambdaMOO, a name Curtis choose because he had used the name Lambda in some of his earlier MUD experiences. Unlike MUDs built on adventure themes, MOOs commonly define the virtual space of a real-world location (e.g., a university campus, a house), inviting participants to speak and act within particular zones (e.g., a room, a walkway).

Contemporary MOOs are being employed in social and educational contexts (see Bruckman and Resnick 1995). Use of non-textual material (e.g., graphics, sound) is also now appearing in MUDs and MOOs.

- **chat**

Chat is a synchronous CMC venue for holding conversations with multiple participants. While an early version of chat was possible through the UNIX "talk" program (allowing multiple users to engage in instant messaging – see above), chat as we now know it was not born until 1988. In that year, Jarkko Oikarinen, a student at the University of Oulu (in Finland), wrote a program that came to be known as Internet Relay Chat (ICR), which was intended as an improvement on UNIX "talk". By the early 1990s, ICR became known to the wider public, serving as a template for more generic chat programs available through Internet providers such as America Online and through the Web.

As in the case of newsgroups, participants in chat enter into a "channel" (for IRC) or "room" (for AOL), ostensibly dedicated to a particular topic. However, with chat, not only is the medium synchronous but it invites both playful and manipulative behavior. Users log on through nicknames (akin to participation in MUDs), free to camouflage their real-world personal characteristics (age, gender, background, etc.). While conversation takes place in real time, users can (as in the case of newsgroups) scroll back through the archive to respond to earlier conversations.

[16] http://www.techtv.com/screensavers/print/0,23102,3388608,00.html

Figure 11 offers an example of chat, taken from a public IRC channel, and analyzed by Sean Rintel, Joan Mulholland, and Jeffery Pittam.

-- Insert Figure 11 Here –
Example of Chat

Like listservs, newsgroups, and MUDs or MOOs, chat generates a quasi-public linguistic record that can subsequently be analyzed. However, given the nature of the conversation in chat, it is primarily linguists and Internet researchers who are interested in analyzing such text, not organizations or commercial ventures.

Web Sites

Unlike the forms of CMC we have discussed thus far, Web sites have historically been a monologic form of communication. That is, they have posted material on the World Wide Web that others might view rather than respond to. In recent years, there is increasing momentum to create Web sites that invite interaction (e.g., currency converters, translation programs, and personal feedback, not to mention the enormous category of online commerce). Note that a Web site is composed of one or more Web pages (see below).

▪ Web pages

Web pages (individual, institutional, or commercial) form the backbone of the World Wide Web. Such pages became possible in the early 1990s when Tim Berners-Lee introduced the notion of what came to be known as a URL (Uniform Record Locator), whereby every Web page could be located by a unique address.

Today, there are billions of Web pages, with the number continuing to grow seemingly limitlessly. Figure 12 offers an example of a Web page, in this case the home page of an academic institution.

-- Insert Figure 12 Here --
Example of a Web Page

In Section 4 below, we will review the coding systems used for creating Web pages, including systems for making Web pages interactive. In Section 5, we will address some of the challenges of locating information on these billions of pages.

▪ Web logs

Web logs are actually Web pages that serve a restricted, though loosely defined set of functions. Also known as blogs, Web logs were created in 1997 by Jorn Barger. Initially, Web logs were designed as lists of Web sites that the blogger found to be of interest and wished to share with others (i.e., via the blogger's own Web site). Sometimes Web logs of this genre simply provide a set of headlines (complete with Web links) that the compiler has put together (with frequent updates) – for example Jorn Barger's robot

wisdom weblog at http://www.robotwisdom.com. Other topic-oriented Web logs offer brief news summaries (e.g., Lawrence Lee's Tomalak's Realm at http://tr.pair.com) or discuss contemporary topics of interest to the blogger and/or readers who have responded by email to earlier "issues" of the Web log (e.g., Virginia Postrel's Dynamist at http://www.dynamist.com).

Use of Web logs has expanded from the link-and-commentary mode to include more personal journals or diaries. Such Web logs may be devoted to posting one's creative writing (sometimes with requests for commentary from readers) or even quite personal revelations about one's daily life and thoughts, perhaps complete with live video from a Web camera.[17] Given the popularization of Web logs, it is hardly surprising that a number of software programs have appeared that enable novice users to create and maintain their own Web logs.

Figure 13 presents an example of a Web log oriented to library issues.

-- Insert Figure 13 Here --
Example of a Web Log

3.2 Analyzing CMC

Stepping back from individual types of CMC, we can now look at such communication more analytically. What linguistic variables define this form of exchange? How has CMC changed over time? And is CMC perhaps a new type of language?

3.2.1 Variables Shaping CMC

The range of CMC communication can be analyzed not simply with regard to the product/process spectrum but also with respect to a number of other linguistic variables. Four such variables are:

(1) function of the message
(2) device constraints
(3) special linguistic features
(4) profile of participants

- **function of the message**

Why is a particular Web page posted or email sent? Just as in traditional off-line writing, there is an enormous range of purposes in sending natural language messages over the Internet. Diversity of purpose is commonly reflected in diversity of writing style. The language used in emailing a buddy to commiserate about your team's soccer loss is vastly different from that in a job application letter composed as an email.

- **device constraints**

[17] See www.rebeccablood.net for a good brief history of Web logs.

As we saw in Section 2.2 above, the technology via which language is conveyed may affect the form and content of the message itself. Such constraints are particularly evident in the formulation of CMC.

Begin with the distinction between asynchronous versus synchronous CMC. Asynchronous communication (especially classic email exchange and most listservs) is more like a traditional postal system. Since there is no assumption that sender and recipient will both be logged onto their respective computer networks, senders can, in principle, linger over formulating a message, composing any length text and (if they choose) revising the text until they are ready to send it. Synchronous communication (especially chat and IM) is more like a face-to-face conversation. Message transmission is immediate. Synchronous messages tend to be short and unedited, because, as with spoken conversation, interlocutors are poised to interrupt or respond.

Other device constraints involve the physical hardware used for formulating messages. Small mobile computing devices such as Personal Digital Assistants (PDAs) or dedicated email devices generally have tiny keypads or use pointing devices rather than keyboards, making it cumbersome to formulate email messages. By the same token, the small screens on such devices discourage reading large amounts of text. In the case of text messages sent over a mobile phone, there are not only the challenges of inputting and reading text, but the messaging system itself restricts the number of characters that can appear in a single text message. All of these device constraints lead users to formulate shorter, less edited texts than they might in classic, full screen, asynchronous email.

- **special linguistic features: emoticons and flaming, abbreviations and acronyms**

Since CMC assumes the tangible form of written communication, recipients of such messages can only rely upon the text itself to decipher the sender's intent. The same limitation has always applied to traditional off-line writing, be it a personal letter, a memorandum, or a short story. However, unlike traditional written communication, CMC that entails dialogue has a greater sense of immediacy. As a consequence, many CMC users have argued that the written CMC medium is inadequate for expressing nuances of meaning (e.g., sarcasm, bemusement, tentativeness, irritation) that facial expressions and/or vocal features typically convey in face-to-face spoken conversation.

Two linguistic features of CMC have emerged from these assumptions about the conversational nature of CMC and the inadequacy of writing to express conversational intent. The first feature is emoticons (also sometimes known as smileys). The second is the phenomenon known as flaming.

Emoticons first appeared in 1982, the creation of Scott Fahlman who was at Carnegie Mellon University. Fahlman wrote at the time: "I propose … the following character sequence for joke markers: :-). Read it sideways. Actually, it is probably more economical to mark things that are NOT jokes, given current trends. For this use :-(."[18]

Emoticons are constructed by combining punctuation marks (sometimes along with characters or numerals) on the computer keyboard to represent emotions or semantic nuances such as happiness, sadness, or qualification. Among the most commonly used emoticons are

[18] Discovery Channel, September 19, 2002, available at http://dsc.discovery.com/news/briefs/20020916/emoticon_print.html

:-)	happiness, humor	:-o	shocked, amazed
:-(	sadness, displeasure	:.(	crying
;-)	winking	:-]	sarcastic

Although a number of emoticon lexicons have appeared (e.g., Sanderson 1993, plus an ever-changing number of emoticon pages on the Web), new emoticons can arise at any time, especially among restricted groups of users (e.g., students attending a particular high school). Moreover, different cultural groups may have radically distinct emoticon systems (see Section 6.1 below). Variations across users in the construction and interpretation of emoticons present special challenges to those attempting to design search procedures for locating semantic information on the Web.

The second linguistic features deriving from the conversational nature of CMC is flaming – that is, use of rude or profane language. For decades, email was notorious for its general rudeness and the apparent ease with which senders resorted to profanity. In fact, ARNANET (so the story goes) routinely had to purge the message stream of its most egregious cases. Historically, flaming has been described as an intrinsic quality of email resulting from the lack of auxiliary auditory and visual cues in CMC (Lea *et al*. 1992, Dery 1994).

Besides emoticons and flaming, a quintessentially written feature that has also been prominent in CMC is use of abbreviations or acronyms. Unlike emoticons, abbreviations (and acronyms) have long been part of the written language tradition, appearing both in handwritten manuscripts and print. Their most common function in both off-line writing and CMC is to conserve energy and/or space. In the case of medieval manuscripts, for example, use of abbreviations allowed additional words to be inscribed on a single page, reducing the number of animal skins needed to produce a book. In the case of CMC, saving time and energy is often a motivation when writing chat, IM, or SMS messages. Space considerations become especially important when using portable devices that restrict the number of characters that can be transmitted in a single message.

There is, however, a second, socially-based motivation for incorporating abbreviations into CMC messages: to indicate one's membership among network cognoscenti. Linguist David Crystal, in describing the function of in-group language, observes that "The chief use of slang is to show that you're part of the gang!" (Crystal 1997a:53). Like the use of slang in face-to-face speech or written language, abbreviations in CMC often are a way of indicating group membership.

Most CMC users are familiar with some of the basic CMC abbreviations,[19] e.g.,

brb	be right back	imho	in my humble opinion
btw	by the way	irl	in real life
cul8r	see you later	lol	laughing out loud
gr8	great	rotfl	rolling on the floor laughing

These same abbreviations (or versions of them) crop up in multiple forms of CMC. Less well-known abbreviations appear in the messages sent by restricted groups of users. For

[19] See Crystal 2001:85-86 for a sample contemporary list.

example, "pos" (parent over shoulder) is used by some American teenagers who are in the process of sending an IM to a friend when a parent walks into the room, and the sender wants to alert the recipient that open discourse is no longer possible.

As in the case of emoticons, there is no master lexicon of abbreviations, in part because they evolve (and sometimes disappear) quickly. Like the proliferation of in-group emoticons, CMC abbreviations that have restricted usage pose a challenge for search programs trying to parse word-like units that are not in the parser's lexicon.

- **profile of participants**

In discussing the CMC spectrum, we focused on issues relating to participants in the exchange. First, we noted that some CMC is essentially monologic and other dialogic. Second, we distinguished between open sets of participants (where new interlocutors come and go, often without any vetting process) and closed sets (where all participants either are known to one another or share some tangible affiliation, e.g., in a professional organization). Third, in many forms of CMC, participants have the option of camouflaging their true identity (e.g., in a chat room) or are even required to create a new persona for themselves (e.g., in classic MUDs).

We can also distinguish among participants on the basis of personal demographics: age, gender, education level, language skills (e.g., among non-native speakers using CMC). In many forms of CMC, participants commonly misrepresent their age or gender (whether in innocent fun or for purposeful deceit). However, using the tools of sociolinguistic analysis, it is sometimes possible to discern the actual gender of the sender.[20]

3.2.2 Myths, Reality, and Evolution of CMC Style

When linguists set about to write the "grammar" of a language, they presuppose the existence of a stable set of linguistic elements, combination rules, and usage conventions that define the linguistic competence (in Noam Chomsky's sense) and communicative competence (as articulated by Dell Hymes) that characterize the knowledge that users of the linguistic system have. In practice, linguists from Ferdinand de Saussure onwards have recognized that the system defined by such elements and rules is a convenient fiction, which may not fully or accurately characterize the linguistic knowledge of any specific individual. This fiction functions reasonably well in defining natural spoken languages that have significant numbers of users and that persist over time.

The problem with attempting to write grammars characterizing CMC is that CMC tends to be at once a diffuse and a moving target. Although its roots are barely three decades old, the scope of CMC usages is quite broad. Moreover, given the explosion in CMC messaging in the second half of the 1990s, the range of users has expanded dramatically. Not surprisingly, the email messages of seventy-year-olds who are new to the medium have little in common with the IMs of teenagers; the home page of a government watchdog organization is predictably distinct from that of an amusement park.

[20] On gender issues in CMS, see Herring 2003.

Ongoing developments in CMC technology may render obsolete many of our linguistic assumptions about the character of natural language usage conveyed by the Internet. As video capabilities improve, use of still photographs, along with streaming recorded video and live Web cams could redefine the balance of information conveyed in written versus graphic form. And as voice recognition technology improves, we can envision a future in which CMC messages are spoken into a device that converts the spoken signal into written text. Such a transformation of input modality might yield radically different sorts of CMC messages from those users presently type out themselves. Emoticons and abbreviations would presumably disappear; spelling, punctuation, and even grammar might improve (handled by the speech recognition program); and messages would probably become much longer, since when we dictate, we generally use more words than when we write text out ourselves (Baron 2000).

Some people have suggested that use of written CMC will be replaced altogether by audio transmission of messages (perhaps complete with Web cams). However, written CMC has a number of clear advantages over receiving audio signals: production of a durable record; potential to conceal age, gender, physical condition, or ethnic or linguistic background; rapidity with which a written message can be read or scanned (in comparison with listening to an oral message); relative privacy in public places; and the ability to multitask (e.g., compose and receive IM while speaking on the phone). Consequently, those who study the Internet generally doubt that written CMC will disappear, regardless of technologically available alternatives.

3.2.3 The Linguistic Character of CMC

Computer mediated communication can be thought of as a kind of linguistic centaur, incorporating features from both traditional writing and face-to-face discourse but ending up being more than a simple amalgam of the two. As the number of linguists studying CMC increases, there is growing interest in characterizing what kind of linguistic modality the language of CMC is, as well as in studying the influence of CMC on traditional spoken and written language.

Early attempts at linguistic characterization tended to be based on one-to-many forms of CMC (especially university listservs), because such texts were easily available for analysis. Most studies in the 1990s queried whether CMC was more like speech or more like writing (see Baron 1998 for a summary analysis). In a comprehensive overview of CMC, David Crystal (2001) argues that while discourse on the Internet (of the sort found in email, chat, and MUDs) actually has more in common with writing than with speech, it is appropriate to recognize what he calls Netspeak as a new language variety, complete with its own lexicon, graphology, grammar, and usage conditions. Such a notion is provocative. However, it remains to be seen if there is sufficient cohesion in Netspeak to warrant labeling it a distinct variety of language.

What is clear, however, is that a number of distinctive linguistic conventions characterizing many people's use of language on the Internet are beginning to seep into traditional spoken and written language. Historically, it is common for concepts or terminology deriving from developments in technology to find their way into everyday language. From the development of railroads, we have come to talk about plans being "derailed". From early computer usage, the terms "input" and "output" gained currency.

CMC is beginning to have similar effects on contemporary speech and writing. With respect to spoken language, some American adolescents, in face-to-face conversation, uses such locutions as "We were all ROTFLing when the teacher walked in". Many college students "google" potential professors before a new semester of classes begins, using the results of a Google Web search (see Section 5 below) to decide if they want to take a course with a particular professor.

Since CMC is physically a written medium, it is not surprising that some of the linguistic conventions common in more informal types of CMC (especially synchronous CMC) are finding their way into traditional off-line writing. Abbreviations and acronyms, devil-may-care spellings, and haphazard grammar – all of which are generally accepted in the world of CMC – are increasingly showing up in student written composition (see, e.g., Lee 2002). Whatever one's feelings about prescriptive written grammar usage, those doing natural language processing of written text may need to contend with language on Web sites looking increasingly like that of conversational CMC.

A more subtle effect of CMC on traditional written language may be on text length. Historically, the emergence of telegraphy at the end of the nineteenth century helped shape an American writing style that was more direct and used shorter sentences than its predecessor, given the physical constraints and related cost issues of the new technology (Hochfelder 1999, Baron 2002). While it may be too soon to tell what effects the brevity of IM and SMS messages may have on writing in general, there are already signs that the Internet is fostering a very different type of prose from the well-developed, logically connected paragraphs that formal education has traditionally modeled for students.

A troubling model of what writing in the future might look like appears in Steve Krug's *Don't Make Me Think*! (2000), a book on how to write good Web sites. Krug writes: "Web users tend to act like sharks: They have to keep moving, or they'll die. We just don't have time to read any more than necessary"(p. 22). Or: "most Web users don't have time for small talk; they want to get right to the beef.…The main thing you need to know about instructions is that no one is going to read them – at least not until after repeated attempts at 'muddling through' have failed"(p. 46.) While this is an accurate description of how most users "read" on the Web (and therefore good advice for Web page design), the larger question will become, what effects are the design of "readable" Web pages likely to have on written language not intended just for rapid browsing.

4 Coding Systems for Communicating via the Web

We move now from natural language usage conveyed over the Internet to the range of formal coding systems employed to create the programming infrastructure that makes communication over the Internet possible. While some of these coding systems constitute formal programming languages, others are more properly termed markup languages, quasi-programming languages, or simply applications programs.

Our purpose in this section is to map out the genres of coding systems presently available and to introduce common exemplars as of the year 2002. The discussion centers on communicating via the Web, not across the Internet more generally. That is, we are not dealing with coding issues specifically relevant to such Internet (but non-Web-specific) functions as email, original newsgroups, or FTPs. Also note that programming

tools for the Web continue to evolve, and new languages and products (especially in the domain of quasi-programming languages and applications programs) will continue to appear.

Our discussion of coding systems is organized into four parts :

(1) summary of Web coding systems
(2) overview of Web markup languages
(3) introduction to Web programming languages
(4) introduction to Web quasi-programming languages and applications programs

4.1 Summary of Web Coding Systems

Conceptually, we can group coding systems for accomplishing Web communication into three broad groups, as shown in Figure 14.

-- Insert Figure 14 Here --
Web "Languages" and Programs

The first group, known as markup languages, is made up of coding systems that technically are not computer languages in the traditional sense of languages such as FORTRAN, Pascal, or C^{++} (see Baron 1986 as well as Section 4.3 below) but rather tagging systems that add meaning to documents posted on the Web. The vast majority of Internet users who become involved with creating and posting Web pages work with these markup languages.

The second group, programming languages used in working with the Web, includes formal computer programming languages that existed independently of Berners-Lee's creation of the Web, but that have, in some cases, been adapted to meet the needs of Web applications. However, these are stand-alone general purpose programming languages that can be used for non-Web functions as well.

The third group, quasi-programming languages and applications programs, is the most fluid. It contains an evolving collection of coding systems that are not full-fledged computer languages but rather programming tools that enable Web designers to make Web pages interactive and to perform computations (which were not options under the original markup standard, HTML). Coding systems in this category run the gamut from tools whose usage requires a fair amount of training to more user-friendly applications packages.

4.2 Overview of Web Markup Languages

A markup language is a system for adding tags to text. The notion of "marking up" a text comes from the publishing world, where editors and printers have a long history of hand writing special notations onto manuscripts to indicate how components of the manuscript should be typeset (e.g., noting type face or point size, spacing between lines or sections, etc.). With the coming of computers, a number of text formatting languages were devised to aid in the digital "typesetting" process. Among the better known are PostScript (a page description language, created by Adobe Systems, that communicates printing instructions

to desktop printers) and TeX (created by Donald Knuth at Stanford to format texts containing mathematical and scientific notation). However, these text formatting systems are not directly related to contemporary markup languages used for constructing Web pages.

Markup languages used on the Web trace back to the late 1960s, when the chairman of the Graphic Communications Association (William Tunnicliffe) and a New York book designer (Stanley Rice) independently noted the desirability of creating a set of description tags for aiding in the process of printing texts electronically.[21] In 1969, Charles Goldfarb, a lawyer working for IBM, embarked on creating a markup language that filled three functions: text editing, text formatting, and information retrieval (i.e., permitting stored documents to be queried and retrieved). The product that emerged from the work of Goldfarb and two of his colleagues became known as Generalized Markup Language (GML). GML, which is still used as a publishing system by a number of industries, is the source from which all standardized text markup languages derive.

Figure 15 presents a more detailed schematic of the historical and structural relationship of markup languages designed for creating Web pages.

-- Insert Figure 15 Here --
Web Markup Languages

An important resource on markup languages is the Web site of the World Wide Web Consortium (W3C), located at http://www.w3.org. Directed by Tim Berners-Lee, the consortium establishes standards for and does research on Web coding systems. When the W3C issues a "recommendation" for a particular language specification, that recommendation indicates that members of the World Wide Web Consortium have agreed that the specification is appropriate for widespread distribution. The W3C site contains important archival information as well as reports on current projects.

- **SGML (Standard Generalized Markup Language)**

SGML is a text description markup language standard based on GML. After completing GML in 1969,[22] Goldfarb worked with colleagues to create a derivative form of the language that would receive approval from the International Standards Organization (ISO). SGML won ISO approval in 1986. Today, SGML is still used is working with large technical documents that go through multiple revisions and need to appear in multiple formats. Given the size and complexity of the system, SGML is not used on personal computers. However, SGML remains relevant in contemporary Web coding, given the growing importance of an abbreviated offshoot of SGML, namely XML (see below).

- **HTML (Hypertext Markup Language)**

[21] Information on the history of the early markup languages GML and SGML is drawn from the Web site of Charles F. Goldfarb (http://www.sgmlsource.com/).
[22] The language did not appear under the name GML until 1973.

Understanding the origins of computer markup languages for handling text enables us to contextualize the development of HTML. Rather than springing to life from nowhere, HTML represents the creative synthesis of the idea of a text markup language (that allows users to edit and format text as well as to query and retrieve text from a database) with the emergence of protocols enabling efficient communication via networked computers. That linkage – which resulted in the creation of the World Wide Web – was made by Tim Berners-Lee.

A theoretical physicist, Berners-Lee worked at the particle physics research center in Geneva known as CERN, first in 1980 and then later again, beginning in 1984. Berners-Lee was interested in creating a decentralized computer networking system that enabled scientists to share documents and data, regardless of where they were or what computer equipment they were using. For his networking backbone, Berners-Lee looked to using the emerging Internet, which already had a strong presence in the United States but was "nearly invisible" in Europe at the time (Berners-Lee 2000:17).

To make his communication system function, Berners-Lee needed three protocols that would allow information to be transferred seamlessly from one computer to another. The first was a label for each address where information was located. These addresses are today known as URLs (Universal Resource Locators). The second protocol defined the discourse rules for communication between computers. These discourse protocols (e.g., between Web servers and Web browsers) are known as HTTPs (Hypertext Transfer Protocols). The third component was a system of protocols for coding the text that was being transmitted or searched. The system of coding protocols that Berners-Lee created is known as HTML (Hypertext Markup Language).

HTML combines elements of SGML with the notion of hypertext, which derives from the work of Vannevar Bush and Ted Nelson. In 1945, Bush envisioned a microfilm-based storage system that enabled users to link related ideas with one another. Twenty years later, Nelson coined the term "hypertext" to describe " 'non-sequential' text, in which a reader was not constrained to read in any particular order, but could follow links and delve into the original document from a short quotation" (Berners-Lee 2000:5). Today the notion of hypertext has been incorporated into a lot of computer software (including Apple Computer's early use of hyperstacks) and, of course, into HTML.

HTML enables users to create Web pages that can be posted on the Web. These pages can be located by other Web users in one of three ways: by knowing the page's URL, by clicking on a link (i.e., to the desired page) that appears on some other Web site, or by using a search engine that looks for HTML tags or meta tags that are embedded in the original HTML document (see Section 5.1 below).

The most essential function of HTML tags is to format content material on a Web page with regard to such basic publishing considerations as title placement, text color, or insertion of a blank line between two lines of text. HTML tags, which occur between angle brackets, appear at the beginning and the end of each block of content to be acted upon. HTML code can also be used to insert hyperlinks into a text. When users click on the link that appears on the screen, they will be led either to another point in the same document or to another URL entirely.

The example below provides a simple illustration of how HTML coding works. The HTML code

```
<html>
```

```
<head>
<title> A Nursery Rhyme Sampler </title>
</head>
<body>
<i> Hickory </i> , dickory, dock. <p> The <i> mouse <I> ran up the
clock. <p>
</body>
</html>
```

appears on the screen as

Hickory, dickory, dock.
The *mouse* ran up the clock.

The <p> indicates a paragraph break and <i> indicates italics. The title (A Nursery Rhyme Sampler) does not appear in the text itself but at the top of the Web browser.

The full set of HTML codes used by adept Web page designers is quite extensive. As growing numbers of Internet users have wanted to create their own Web sites, the software industry has responded by writing applications programs that function as WYSIWYG (What You See Is What You Get) HTML editors. That is, they enable users to see the document that HTML describes as they are creating the HTML code.

HTML has a number of shortcomings. First, the standard HTML tags only create static Web pages. Second, all of the tags are defined in advance, leaving developers no opportunity to create their own tags that might be well suited for their specific purposes. Third, HTML was designed to work with full-fledged computers, not with small mobile devices such as PDAs or mobile phones. And fourth, HTML is not able to perform computations (e.g., calculating real-time weather or currency conversion). The successors of HTML 4.0 (sometimes coupled with Web programming languages and/or quasi-programming languages or applications programs) address these issues.

- **DHTML (Dynamic HTML)**

DHTML targets the problem of HTML-generated pages being static. By "static" we mean that once HTML tags are created, the images that appear on the page are visually fixed. Graphics cannot move on the page, colors cannot change, and type fonts and sizes always remain the same unless the HTML code itself is changed. DHTML allows these elements to change as the document is being displayed on a screen by a browser. The changes are brought about by using an embedded script (that is, code written in a quasi-programming language) that works in tandem with the HTML tags. Most commonly, this embedded code is written in JavaScript (see Section 4.4 below).

DHTML is not itself a new markup language. Rather, it is a version of HTML that incorporates dynamic options. In the W3C scheme, the version of HTML that incorporates the dynamic additions of DHTML is known as HTML 4.01.

- **XML (Extensible Markup Language)**

XML, which is a simplified version of SGML, was created in 1998 by the World Wide Web Consortium to improve Web coding by providing much greater flexibility than available under HTML and its direct offshoots. In fact, unlike HTML (which, as we have

seen, defines a static set of tags), XML allows individual users to define tags that best suit their needs.

Here is an example of how XML might work.[23] Imagine you are a primary care physician who wants to email a patient's medical record to a specialist. The specialist would like to directly input that information into her hospital's database. The problem is, her hospital's computer has no way to make sense of the incoming information. Trying to encode the data in HTML would be hopeless, since HTML has a pre-defined set of tags that do not include categories such as <patient> or <drug-allergy>. XML allows the medical profession to identify profession-specific tags – in essence, creating a customized markup language to use for medical records. Such tags might include <patient>, <allergies>, <drug reaction>, <insurance carrier>, etc. Now both physicians can communicate with one another (and with their respective data bases) in language that makes sense to them and to their computers.

Work on XML is still continuing. One of the goals of XML is to incorporate a range of functionalities not available in HTML. Besides extensible tags, XML also makes use of a coding system known as Unicode (see Section 6.3 below), which makes it possible for XML to function in most of the world's scripts, even within the same document. With HTML, users can only handle a single script at a time, thereby restricting the potential for global communication.

- **XHTML (Extensible HTML)**

XML is not a replacement for HTML but rather complements its functions. In fact, the World Wide Web Consortium is looking to standardize a new coding system, XHTML, which introduces into dynamic HTML the tag-extensibility of XML while also extending the system to work not only on small mobile devices such as mobile phones and PDAs, but also on televisions, cars, and public kiosks.

XHTML is still evolving. As of May 2001, version 1.1 had been reached the stage of a Recommendation.

4.3 Web Programming Languages

The second type of Web coding systems is formal computer programming languages. These include both traditional computer languages and languages that have been adapted to meet the needs of Web applications.

A programming language is set of rules (represented in a computer program) that allows its users to provide instructions to a computer for the solution of any problem amenable to numerical or linguistic manipulation. Programming languages differ from Web markup languages in that markup languages specify how text will appear on a page but are not capable of general-purpose problem solving.

Computer programming languages have evolved from unstructured programming (e.g., COBOL, FORTRAN, BASIC) to structured programming (e.g., Pascal and Ada) and then to object-oriented languages (e.g., Simula, Smalltalk, C^{++} and Visual Basic).

[23] This example is based on Bosak and Bray (1999).

Object-oriented languages such as C++ and Visual Basic have proven extremely popular in constructing software programming for the Internet.

In addition to general-purpose programming languages that happen to be used for the Web, some existing programming languages have been tailored to be particularly effective for Web programming. For example, the language Java (based on C++, but smaller and simpler) is an object-oriented language developed by Sun Microsystems in the early 1990s to run consumer electronics. It soon became clear to Sun that with some adaptation, Java would be very useful for creating dynamic and interactive Web content. In time, Java has become one of the favored Web-adapted programming languages. Another example is the programming language PERL (= Practical Extraction and Report Languages). Though originally developed in 1987 for monitoring large software projects and creating reports, PERL has been adapted for Web use (see discussion of Common Gateway Interfaces below).

4.4 Web Quasi-Programming Languages and Applications Programs

Besides markup languages and general-purpose programming languages, Web coding is increasingly done through an expanding set of quasi-programming languages and applications programs.[24] These coding devices constitute tools that make Web pages dynamic and interactive. Some of these tools are readily accessible to lay users while others are generally left in the hands of experienced programmers. Among the programming tools in this general category are "languages" such as JavaScript, ASP (Active Server Pages), PHP (Personal Home Page Tools), and JSP (Java Server Pages).

Because this area of Web programming is undergoing rapid change, interested readers should locate up-to-date texts that survey the field.[25] Here, we restrict ourselves to noting three programming concepts relevant to quasi-programming languages and applications programs: server-side versus client side coding, scripting languages, and Common Gateway Interfaces.

- **server-side versus client-side coding**

Networked programming entails using two types of programming systems. Server-side programs are used to provide documents to client-side systems, which display such documents. We can think of the Web as a large collection of clients connected to a smaller collection of servers, through the Internet. By dividing up the work between servers and clients, Web servers can be freed up from performing "local" tasks and left available to handle a larger number of clients. "Clients" can be thought of as both the Web browser a person is using and the specific computer on which that browser is running.

The earliest Web servers were written at CERN in Geneva and at the National Center for Supercomputer Applications at the University of Illinois. Today, most Web servers work on a UNIX-based platform. The predominant server used is Apache. As of the late 1990s, there were several hundred thousand servers in use on the Web (Sebesta 2002:8).

---

[24] In actual practice, there is a tendency to combine all three types of coding systems, sometimes embedding one within another.

[25] As of 2002, three useful texts are Andrews 2002; Deitel, Deitel, and Nieto 2002; and Sebesta 2002.

Client-side programs are resident on computers accessing the Web. Their function is to add interactivity to Web sites. Client-side coding systems are closer to general-purpose programming languages than to markup languages.

- **scripting languages**

Scripting languages are coding systems that create programs (called scripts) that have restricted functions (in comparison with general-purpose programming languages). These scripts, which can be embedded within HTML, make Web pages more dynamic and interactive. Historically, scripting languages are sometimes smaller subsets of general-purpose languages (e.g., VB Script is a subset of Visual Basic).

Scripting languages can work either on the server side or the client side. Among the server-side scripting languages are PERL, ASP (Active Server Pages), PHP (Personal Home Page Tools), ColdFusion, and JSP (Java Server Pages). Server-side scripts perform such functions as allowing dynamic access to databases and responding to user input.

Client-side scripting languages include JavaScript and VB Script.[26] Client-side scripting languages are used for such functions as creating pop-up windows and performing calculations for shopping cart purchases on commercial sites.

- **Common Gateway Interfaces (CGIs)**

Common Gateway Interfaces (CGIs) are programs that enables a browser and a server to communicate in that the server runs a program and returns the output of that program to the browser (i.e., client side), where the user can view it. CGIs are used on the server-side to create data-driven Web sites. Unlike HTML programs, which are static, CGIs are executed in real time, enabling the outputting of dynamic information. Such output might be a weather report or the result of a query made of a database.

CGIs can, in principle be written in any coding system that is supported by the server being used. However, in the 1990s, the most common language used for writing CGIs was PERL.

In addition to the quasi-programming languages described above, a number of client-side applications tools are appearing that enable users to access special types of materials on the Web. Among these tools are RealOne Player and Flash (to add multimedia capabilities to Web pages) and Adobe Acrobat Reader (for viewing PDF – Portable Document Format – files). Many of these tools can be downloaded for free from the Internet by end users.

5 Web Search Systems

We turn now from the question of how coding is done on the Web to the issue of how end users extract information from Web pages. The Web poses an enormous problem for information retrieval. The cornucopia of text (and now graphics and audio files) available

[26] Occasionally, JavaScript is also used on the server side.

through the Internet is useless unless there is an effective search procedure for finding what you are looking for. Our discussion here focuses on

(1) how traditional search engines locate information on the Web, using key words and phrases
(2) new approaches to Web searching, including intelligent agents and the Semantic Web.

5.1 Traditional Search Engines

Contemporary computer users are familiar with the frustrations of seeking information through a commercial search engine. Admittedly, we can often track down references on the Web that might have taken hours to find through a trip to the library. Yet we have all experienced the exasperation of finding no "hits" for a topic we are sure must be out there somewhere. Similarly, we encounter totally irrelevant sites, need to sort through duplicate sites, and have to deal with sites that contain the right key words but clearly are not what we are looking for. (A recent search of mine for a specific thoracic surgeon repeatedly turned up sites for a pornography star.) To understand why these kinds of problems arise, we need to understand how search engines work.

5.1.1 How Search Engines Work[27]

A search engine is a software program that assists users in locating Web pages that contain information they are looking for (be it the text of Shakespeare's *King Lear*, the final score of last year's Army-Navy football game, or the times a movie will be playing this Saturday at a local theater). In principle, search engines come in two varieties: crawler-based search engines (that use software programs called robots, spiders, or crawlers to locate links from one document to another on the Web) and human-powered directories (that is, Web directories that result from human editors sorting through sites). In practice, many contemporary search engines are hybrids of the two systems.

Crawler-based systems are made up of three components. First, there is the crawler, which works its way through millions of Web pages, returning at specific intervals (e.g., monthly for sites that do not change often; daily for high-traffic, dynamic sites) to look for changes in Web pages that need to be noted. Second, there is the index (or catalogue), which contains a copy of every page that the crawler finds. The index is updated to incorporate Web page changes noted by the crawler. However, because these updates are not automatic, it may take several weeks or more before changes a user makes to his or her Web site are registered in a particular search engine's index. And third, there is the search engine software, which sifts through pages contained in the index to identify matches with the search terms a user has input. An additional function of the search engine software is to rank the results, presumably in order of importance to the user. Such ranking becomes extremely important, given the vast number of hits (sometimes into the millions) that many Web searches now generate.

As of 2002, there were hundreds of different search engines available to end users. A few of the best known are:

Google: crawler-based

[27] Discussion in Sections 5.1.1 and 5.1.2 draws upon information from http://searchenginewatch.com.

originally called BackRub; name changed in 1998
google = the term in mathematics for the figure 1 followed by 100 zeros

AltaVista: oldest crawler-based search engine (1995)

Yahoo: oldest Web directory (1994)
originally used human editors; now uses Google crawler-based listings, though retains human review

Other popular search engines include Lycos, HotBot, MSN Search, Ask Jeeves, and AOL Search.

5.1.2 The Art of Being Found on the Web

A major challenge for Web designers is figuring out how to be sure that search engines will find their site in the first place, and then how to get the site high in a search engine's rankings. Few users are willing to tunnel through thousands of results from a Web search.

Since every search engine uses its own algorithm for working through the vast array of pages available on the Web (and different engines search different pages), there is no sure-fire method to guarantee a high ranking (especially across search engines). However, there are a number of general principles that can facilitate being found and being placed higher up in the pecking order of results. Many of these strategies involve strategic placement and frequencies of keywords in the HTML code for the Web page. However, sometimes the answer is essentially monetary.

Five basic ways to improve your chances of being found on the Web by a traditional search engine are: use of meta tags, understanding what search engines "read" on a Web page, use of links, direct submission of Web pages, and advertising.

- **meta tags**

Meta tags are additional lines inserted into the <head> area of HTML code that clearly spell out what the Web page is about, potentially making it easier for search engines to find the page. There are two basic components to meta tags: descriptions (which explain, in natural language, the content of the page) and key words (which identify the key concepts appearing in the page <body>. Building on the short example of HTML code we presented in Section 4.2 above, here is an example of how meta tags are incorporated into HTML code:

```
<html>
<head>
<title> A Nursery Rhyme Sampler </title>
<meta name= "description" content= "Collection of nursery rhymes from
Mother Goose and other sources.">
<meta name= "keywords" content= nursery rhymes, Mother Goose,
children's rhymes">
</head>
<body>
…
```

The choice of key words is especially important. Not only should they appear in meta tags but they should appear prominently in the text as well – in the HTML <title> tag and in the <body> of the text, especially high up. While not all search engines specifically look at meta tags, inserting them does no harm and may help your page be found.

- **understanding what search engines do (and do not) read**

Search engines can only find what they are designed to look for. For example, search engines ignore many words when they go about doing a search. Such selectivity is illustrated by the classic example of looking for the opening phrase of Hamlet's "To be or not to be" soliloquy. Run the search on Google, and you come up with millions of hits on the word "not", but not on the phrase "to be or not to be". This undesired result comes about because Google does not search on "to", "be", or "or". A Google search on the key words "to be or not to be Shakespeare" or "to be or not to be Hamlet" are not much more promising. (Though the latter search does yield some hits, success results from the presence of the words "Hamlet" and "not", not the Shakespearean phrase.) Obviously, an ideal search engine would find a way around this problem.

Search engines have other peculiarities that hinder being found on the Web. For example, use of certain programming tools (such as large amounts of JavaScript or tables) can push key words farther down into the Web code, making it harder for search engines to find them. Similarly, search engines may not read image maps or pages generated by Common Gateway Interface code.

- **links**

Search engines rely heavily on the use of links to locate pages. Therefore, strategic use of links can increase "findability" on the Web. Not only is it important to use links in constructing pages, but it is smart to link to sites that are publicly respected and that themselves receive significant numbers of hits. It is also useful to request other sites (especially those that are highly ranked in Web searches) to link to your own.

- **direct submission of Web pages to search engines**

If you are an actor hoping to break into Broadway, it is best to audition for plays rather than sit by and hope to be discovered. Just so, rather than hope search engines will notice your Web site, you can submit pages directly to search engines. There are also commercial services that will submit pages for you.

- **advertising**

Have you ever wondered why some books happen to be placed next to the cash register in bookstores, increasing the probability that shoppers will scoop them up as they go to check out? Often publishers pay bookstores for such placements. The same is increasingly true of Web sites and search engines. A substantial fee can rank your site at the top of a list of hits. Another alternative for receiving a high placement is to pay a small fee each time a user clicks on your link ("pay per click").

5.2 Emerging Search Techniques

Search engines are obviously a useful step to bringing order to the chaos of billions of pages on the World Wide Web. However, soon after the invention of the Web, it became clear that we need smarter methods of finding precisely what we are looking for, avoiding the kinds of search problems we have been describing. One solution – intelligent agents – has emerged from the artificial intelligence community. The second solution (which incorporates intelligent agents) is the Semantic Web – a notion coming from the Web's inventor.

5.2.1 Intelligent Agents

In the world of artificial intelligence, an agent is a computation system that has longevity, has goals, and can make autonomous decisions about how to act in the present circumstance so as to make progress towards such goals. Agents come in a number of varieties, including free-standing robots, expert assistants, and software agents (also known as "knowbots", "softbots", or "intelligent agents"). Much of the initial work in developing intelligent agents has come out of the MIT Media Lab, especially through the efforts of Patti Maes.[28]

Intelligent agents are computer programs residing on individual computers or computer networks that gather information, make inferences, draw conclusions, and act independently from human users. They are designed to help computer users perform everyday tasks that are computer-driven (e.g., finding restaurants, doing efficient Web searches, or getting updates on targeted news items). Users "train" their intelligent agents how best to work on their behalf. Unlike static software programs, intelligent agents can "learn" from previous experience (and input from their users), thereby becoming increasingly "intelligent" (and valuable) over time.

A good example of an intelligent agent is Pattie Maes' email-sorting program, Maxims. The task of working through email messages can be tedious and time-consuming. Maxims goes to work on your email inbox, sorting the mail into priority order. While sorting itself is not a network function, Maxims is also able to seek "advice" from other Maxims agents on a network on how to handle particular messages.

Intelligent agents are also used by online companies or services to track customer behavior and then advise these customers on future purchases. For example, many online booksellers track user clicks and sales. The intelligent agent compares these choices with selections made by other people who have, for example, bought the same book, and then the agent recommends additional books a person might be interested in, based upon purchases made by the comparative group.

The need for such intelligent agents has become abundantly clear. An intelligent agent doing a Web search for my thoracic surgeon would never have troubled me with the porn queen sites. And an intelligent search of the Web for "to be or not to be" would not have yielded approximately 416 million hits for me to sort through, all selected because they contained the word "not". As the amount of information available on the Web continues to mushroom, the task of handling such information increases

[28] This section draws upon Maes 1997, Moschovitis *et al*. 1999:157-259, and Feldman and Yu 1999.

proportionately. Moreover, as the number of Internet users with no training in networked computing grows, there is great need for user-friendly retrieval interfaces. Let loose on the Web, software agents have the potential to perform faster and smarter searches than most humans.

5.2.2 The Semantic Web

How might Web usability be improved if you combined the power of intelligent agents with major programming enhancements of the sort made possible by XML code? That is the question upon which Berners-Lee is now working, under a project known as the Semantic Web. The Semantic Web is conceived of as a sort of global data base that users can access in far more sophisticated ways than now possible with either traditional search engines or isolated intelligent agents.

How does the Semantic Web work? Consider a scenario envisioned by Berners-Lee and his colleagues.[29] The players are Pete and Lucy (brother and sister) and their mother, whose medical care they are attempting to coordinate with the help of the Web:

> Lucy…was on the [phone to Pete] from the doctor's office: "Mom needs to see a specialist and then has to have a series of physical therapy sessions. Biweekly or something. I'm going to have my agent set up the appointments."… Lucy instructed her Semantic Web agent through her handheld Web browser. The agent promptly retrieved information about Mom's prescribed treatment from the doctor's agent, looked up several list of providers, and checked for the ones in-plan for Mom's insurance within a 20-mile radius of her home and with a rating of excellent or very good on trusted rating services. It then began trying to find a match between available appointment times (supplied by the agents of individual providers through their Web sites) and Pete's and Lucy's busy schedules….

The emphasized words in the passage above represent the keywords whose meaning (that is, whose semantics) has earlier been defined, via the Semantic Web. Using these definitions, a network of agents goes to work to determine what the best providers and appointment times might be. If the results are satisfactory to the end users (here, Lucy and Pete), the agents then book these appointments via the Semantic Web.

As of now, such a scenario is still in the development stage. However, the vision of the Semantic Web both recognizes the need to improve Web functionality and offers a concrete proposal for how to begin.[30]

6 Cross-Linguistic Challenges on the Internet

We move now from the task of searching the Web in a single language to challenges engendered by the multilingual character of a software network that strives to serve a global audience. Our discussion looks at

---

[29] The scenario is from Berners-Lee, Hendler, and Lassila 2001. More technical discussion of the Semantic Web can be found in Geroimenko and Chen 2002, and in Fensel, Wahlster, and Lieberman 2002.

[30] For updates on progress on the Semantic Web, see http://www.w3.org.

(1) the challenges of communicating in a multilingual world
(2) issues of cross-linguistic translation
(3) issues relating to the use of multiple scripts to represent the world's languages

6.1 The Challenges of a Multilingual World

In 1999, a Montreal photographer named Michael Calomiris was fined by the Quebec government because of the content of the Web site he used to display his works. The issue was not the photographs but the language of the site. By Quebec law (Section 52 of the Charter of the French Language), it is illegal in Quebec not to do business in French. Calomiris's site was in English because, as he explained, "my aim was global, not local….We have a lot of customers in the United States, and …they don't care if [the site is in] French". But, as *Wired News* reported, the provincial government of Quebec does.[31]

As we saw in Section 2.3 above, linguistic diversity adds new dimensions to communicating via the Internet. Such problems become increasingly acute as the numbers of non-native speakers of English (or speakers with no command of English) grows. According to Global Reach, the non-English-speaking population of online users is expected to reach over 800 million by 2005.[32] In 2002, Global Reach (quoting from eMarketer) estimated that of the 313 billion Web pages currently on the Internet, over 68% were in English. How will that percentage change as an increasing number of non-English speakers (or non-native English speakers who prefer to use their native language) become users of the Internet?

In the commercial domain, there is the question of how a business can make its presence accessible on the Web (and, in the case of multinational corporations, how to maintain corporate identity across countries) while still "localizing" the site for use by national audiences. Not surprisingly, there is a rapidly developing market for Web site localization, which often involves expertise in both linguistic translation and cross-cultural sensitivity.

Cross-linguistic issues also impact Internet users at the individual level. Consider the case of emoticons. While we discussed the use of emoticons in American English CMC, a very different emoticon tradition known as *kaomoji* (literally "face marks") has emerged in Japan (Katsuno and Yano 2002). Unlike smileys (which focus on the image of the mouth), *kaomoji* focus on the eyes. Moreover, while smileys are meant to be read sideways, e.g., :-), *kaomoji* are read right-side up, e.g., ^_^. Even more importantly, the social role that *kaomorji* play in online discourse is quite distinct from that of smileys in the US.

Beyond the lexical issue of emoticons, there is the question of how bilinguals choose to construct CMC messages. Do they consistently use one language or another, or do they employ both languages, even in the same message (code-switching)? Assuming there is an option, which script do they employ if one of their languages (e.g., Greek, Russian, Arabic, Hebrew, Japanese) is typically written in a non-Roman script?[33] And if

[31] http://www.wired.com/news/politics/0,1283,20082,00.html (June 8, 1999)
[32] http://global-reach.biz/globstats/evol.html
[33] See Waschauer *et al*. 2002 for discussion of script choice and code-switching in Egyptian email.

there is only a Romanized keyboard available, how do message senders make creative use of the Roman script to represent sounds in the other language?[34]

To better understand the cross-linguistic challenges of language on the Internet, we turn in particular to issues involving translation and scripts.

6.2 The Translation Issue

There is a well-known story, though apparently apocryphal (Hutchins 1995), about early problems in using computers to translate between languages. A generic version of the story goes like this: Soon after the onset of the Cold War, scientists were interested in testing whether computers could successfully translate between English and Russian, given the importance of accessing Russian documents. Choosing a Biblical text (Matthew 26:41), the computer operator fed in the English words "The spirit is willing but the flesh is weak" to produce a Russian translation. The Russian result was then fed back into a Russian-English translator program, yielding "The vodka is strong but the meat is rotten" – hardly a result that might bolster national security.

For more than forty years, computer scientists have been working to make machine translation (that is, using parser and translation programs to translate between natural languages) a successful enterprise. In some limited domains (e.g., meteorology, technical manuals), machine translation (MT) has met with reasonable success. However, it may be many years before we can expect MT to successfully handle unrestricted text, complete with idioms, complex syntax, and nuanced meaning.

Successful MT is a critical task for Internet designers, given the importance of sharing information (as well as doing business) across linguistic boundaries. Human translation, while far more accurate, is extremely costly. In commercial contexts (including "localization" of Web sites into the language of the local community), the solution is often to use MT for a "first pass" at translation and then to bring in skilled human translators to refine the initial results.

How do individuals using the Internet cope with the translation problem? Since few users are competent in more than two or three languages at best, and most of us do not work with human translators at our sides, the "poor man's solution" has been the appearance of free, online translating tools.

The best known of the online translating tools is BabelFish, named after a small animal in Douglas Adam's *Hitchhiker's Guide to the Galaxy*. (You inserted the Babelfish into your ear, and it translated whatever language it heard.) Owned by AltaVista and powered by Systran software,[35] BabelFish goes to work on many non-local sites (e.g., in French, German, Italian) to translate them into English. While the results may be better than nothing, they clearly prove that MT has a long way to go before it becomes a reliable technology for coping with the translation problem on the Internet.

To illustrate the current state of MT on the Internet, consider the text from the Book of Matthew, "The spirit is willing but the flesh is weak". An experiment I performed in December 2002, translating first from English into the target language, and then from the target language back into English, yielded uneven results. Translations

[34] See Tseliga 2002 for discussion of script adaptation in Greek email.
[35] http://babelfish.altavista.com

involving French, Italian, Portuguese, and German were done using BableFish. The Russian translations were performed on Prompt Company's Online Translator[36]:

French:	The spirit is laid out but the flesh is weak.[37]
Italian:	The spirit? arranged but the meat? weak person.
Portuguese:	The esp?rito? made use but the meat? weak.
German:	The spirit is ready, but the flesh is weak.
Russian:	The spirit wishes, but the flesh is weak.

The challenge for doing MT on the Internet is that natural language texts appearing on the Net (be they Web pages, listserv postings, or instant messages used in the business world) are generally not restricted to the specialized domains for which current MT has proven successful. While tools such as BabelFish (and their successors) may offer a first pass at translation (e.g., enabling users to get the basic gist of a translated text), it may be a long time before we can rely upon these MT devices for any purpose where precise textual understanding is critical.

6.3 The Script Issue

Beyond the challenge of dealing with thousands of different world languages, there is the problem of handling the dozens of scripts used to encode those languages. Linguists commonly divide scripts into three categories. The first, logographic, uses symbols to stand for whole concepts or words. Chinese is the best known example. The other two basic script types represent sounds. Syllabic scripts (e.g., Japanese *hiragana* and *katakana*) use a single symbol to stand for an entire syllable (e.g., the sound sequence *ka* in the word *katakana*). Alphabetic scripts (e.g., Greek, Arabic, and Roman) pair individual sounds with individual letters (e.g., representing the sound sequence in the word "cat" with three distinct graphemes, {c}, {a}, and {t}).

However, the situation with scripts quickly becomes far more complex. Within a single script type (e.g., alphabetic), there is considerable variation. Greek, Russian, Arabic, and English are all written with alphabets, but with different alphabets, due to the different historical circumstances under which their writing systems arose.

A second complication is that even within a given script subset (e.g., the Roman alphabet), there is variation across languages using that written script. For example, written French uses several diacritical marks (e.g., *accent grave*, as in *è*; *cedilla*, i.e., *ç*) not found in English. Written Polish has nine alphabetic symbols (written by adding diacritical marks to existing Roman graphemes – e.g., *ł* and *ź*) that are not found in written English (Daniels and Bright 1996:666). These slight alphabetic variants can prove disastrous for contemporary search engines, since no common conventions have been established for how search engines handle variants of Roman script (see Callahan 2002). For example, as of late 2002, the Google search engine only extracted exact lexical matches. If you entered a word with a French *accent grave* (e.g., *très*) it would not hit a

[36] http://www.translate.ru

[37] Interestingly, when a different online translator was used for French (translations.com, at http://mt.translation.com) the resulting back-translation into English had a modified subject and an active verb in the first half of the sentence: The easy spirit lays out but the flesh is weak.

site containing *tres* (i.e., without the accent) However, other search engines (such as AltaVista) provided matches for both spellings.[38]

Finally, there is the linguistic and political reality that the script used to encode a language may change over time, necessitating doing Web searches of the same language in multiple scripts. For example, traditional Chinese characters are used in Taiwan, while simplified Chinese characters are used in the People's Republic of China. Historically, Turkish was written in an Arabic-based script, though with the reforms of Kemal Ataturk in the early twentieth century, Turkish came to be written in a modified version of Roman script.

A growing number of countries that employ scripts other than the basic Roman character set used for English have developed keyboards and computer software enabling writers of, say, Arabic, Chinese, or Hebrew to do word processing, send email, and engage in native-language Internet activity. However, a problem arises when users of different scripts (or script variants) attempt to communicate with one another across the Internet .

One attempt to address this problem is Unicode, a project initiated in 1987. Using an extended model of the ASCII character set, Unicode is designed to assign a unique numerical code to every distinct character in every writing system of the world, either in current use or historically known. The goal, according to The Unicode Standard, Version 3.0, is to define "a consistent way of encoding multilingual text that enables the exchange of text data internationally and creates the foundation for global software".[39]

Unicode uses 16-bit encoding, which makes available unique code values for more than 65,000 written characters. (Version 3.0 of The Unicode Standard utilizes 49,194 of these codes to represent characters from the world's scripts.) Using an extension mechanism to the 16-bit encoding, Unicode is potentially capable of representing as many as one million additional script characters.

The task of encoding the world's scripts is still on-going. Universities and professional organizations (such as the Linguistic Society of America) are being enlisted to assist in the encoding project, especially for minority scripts (such as Balinese and Vai) and historical scripts (such as Aramaic and Egyptian hieroglyphs).

While the Unicode project holds promise for coping with multiple scripts on the Internet, we may still be a long way from having a functioning system. Given Unicode's current level of functionality, critics question whether the coding project will eventually live up to its promise.

7 Suggested Further Reading

"The language of the Internet" covers a wide range of issues, crossing several disciplines. Here are suggestions for further reading, grouped according to the major sections of the chapter. Full bibliographic information appears in the References section below.

---

[38] www.searchengineshowdown.com/newsarchive/000097/shtml. Also see Callahan 2002.

[39] See http://www.unicode.org. The quotation is drawn from the beginning of Chapter 1 of the online edition of The Unicode Standard Version 3.0.

- **Introduction**

Janet Abbate's *Inventing the Internet* (1999) and Tim Berners-Lee's *Weaving the Web* (2000) provide detailed introductions to the development of the Internet and the World Wide Web. For an interesting journalistic approach, see Katie Hafner and Matthew Lyon's *Where Wizards Stay Up Late* (1996).

- **Linguistic Issues**

Two useful discussions of the relationship between spoken and written language are Wallace Chafe and Deborah Tannen's 1987 article in the *Annual Review of Anthropology* and Naomi Baron's *Alphabet to Email: How Written English Evolved and Where It's Heading* (2000).

- **Natural Language Usage on the Internet**

Two thorough sources on natural language usage on the Internet are David Crystal's *Language and the Internet* (2001) and Susan Herring's chapter in the *Annual Review of Science and Technology* for 2002. In addition, a number of journals publish articles relating to CMC. A useful online source is the *Journal of Computer Mediated Communication* (http://www.asusc.org/jcmc). Relevant print journals include *The Information Society* and *New Media and Society*.

- **Coding Systems for Communicating via the Web**

The most up-to-date source for information on programming for the Web can be found at the Web site of the World Wide Web Consortium (http://www.w3.org). General books on programming for the Internet and the Web include Jean Andrews' *i-Net+ Guide to the Internet* (2002); Deitel, Deitel, and Nieto's *Internet and World Wide Web: How to Program* (2002); and Robert Sebesta's *Programming the World Wide Web* (2002). A particularly helpful article on XML is Jon Bosak and Tim Bray's "XML and the Second-Generation Web" (1999).

- **Web Navigation Systems**

Information on search engines can be found online (e.g., Search Engine Watch at http://searchenginewatch.com or The Spider's Apprentice at http://monash.com/spidap2.html). Rachel McAlpine's *Web Word Wizardry* (2001) is quite readable (see especially Chapters 13-17).

A good place to begin reading about intelligent agents is Pattie Maes' "Intelligent Software" (1995). For an introduction to the Semantic Web, see Berners-Lee, Hendler, and Lassila's "The Semantic Web" (2001).

- **Cross-Linguistic Challenges on the Internet**

David Graddoll's *The Future of English?* (1997) offers a perceptive analysis of the place of English in the global arena in the coming years, including its position on the Internet. Online statistics on current and projected language use on the Internet can be found at http:// www.glreach.com/globstats/.

References

Abbate, Janet. 1999. *Inventing the Internet*. Cambridge, MA: MIT Press.

Andrews, Jean. 2002. *i-Net+ Guide to the Internet*, second edition. Boston: Thompson Learning.

Baron, Naomi S. 1981. *Speech, Writing, and Sign*. Bloomington, IN: Indiana University Press.

Baron, Naomi S. 1986. *Computer Languages: A Guide for the Perplexed*. New York: Doubleday.

Baron, Naomi S. 1998. Letters by Phone or Speech by Other Means: The Linguistics of Email. *Language and Communication* 18:133-170.

Baron, Naomi S. 2000. *Alphabet to Email: How Written English Evolved and Where It's Heading*. Routledge: London.

Baron, Naomi S.2002. Who Sets Email Style? *The Information Society* 18 (5): 403-413.

Baym, Nancy K. 2000. *Tune In, Log On: Soaps, Fandom, and Online Community*. Thousand Oaks, CA: Sage Publications.

Berners-Lee, Tim. 2000. *Weaving the Web*: New York: HarperCollins.

Berners-Lee, Tim, James Hendler, and Ora Lassila. 2001. The Semantic Web. *Scientific American*, May, 35-43.

Biber, Douglas. 1988. *Variation across Speech and Writing*. Cambridge: Cambridge University Press.

Bloomfield, Leonard. 1933. *Language*. New York: Holt.

Bolter, Jay David. 2001. *Writing Space*, 2nd edition. Mahway, NJ: Lawrence Erlbaum.

Bosak, Jon and Tim Bray. 1999. XML and the Second-Generation Web. *Scientific American*, May, 89-93.

Bruckman, Amy and Mitchel Resnick. 1995. The MediaMoo Project: Constructionism and Professional Community. *Convergence* 1(1): 94-109.

Bruns, Gerald. 1980. The Originality of Texts in a Manuscript Culture. *Comparative Literature* 32:113-129.

Callahan, Ewa. 2002. Web Search Engines: Information Retrieval in Less Common Languages. Paper presented at the Third International Conference of the Association of Internet Researchers. Maastricht, The Netherlands, October 13-16.

Campbell, Todd. 1998. The First E-Mail Message. *PreText Magazine*. Available at http://pretext.com/mar98/features/story2.htm.

Casson, Lionel. 2001. *Libraries in the Ancient World*. New Haven: Yale University Press.

Chafe, Wallace and Deborah Tannen. 1987. The Relation between Written and Spoken Language. *Annual Review of Anthropology* 16:383-407.

Cherny, Lynn. 1999. *Conversation and Community: Chat in a Virtual World*. Stanford: CSLI Publications.

Crystal, David. 1997a. *The Cambridge Encyclopedia of Language*, second edition. Cambridge: Cambridge University Press.

Crystal, David. 1997b. *English as a Global Language*. Cambridge: Cambridge University Press.

Crystal, David. 2001. *Language and the Internet*. Cambridge: Cambridge University Press.

Curtis, Pavel and David A. Nichols.1993, MUDs Grow Up: Social Virtual Reality in the Real World. Available at http://www.zacha.net/articles/mudsgrowup.html.

Daniels, Peter T. and William Bright, eds. 1996. *The World's Writing Systems*. New York: Oxford University Press.

Deitel, H.M., P.J. Deitel, and T.R. Nieto. 2002. *Internet and World Wide Web: How to Program*, second edition. Upper Saddle River, NJ: Prentice-Hall.

Dery, M., ed. 1994. *Flame Wars: The Discourse of Cyberculture*. Durham, NC: Duke University Press.

Feldman, Susan and Edmund Yu. 1999. Intelligent Agents: A Primer. *Searcher* 7(9). Available at http://www.infotoday.com/searcher/oct99/feldman+yu.htm.

Fensel, Dieter, Wolfgang Wahlster, and Henry Lieberman, eds. 2002. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. Cambridge, MA: MIT Press.

Geroimenko, Vladimir and Chaomei Chen, eds. 2002. *Visualizing the Semantic Web*. Berlin: Springer Verlag.

Graddoll, David. 1997. *The Future of English?* London: British Council.

Haefner, Ralph. 1932. *The Typewriter in the Primary and Intermediate Grades*. New York: Macmillan.

Hafner, Katie and Matthew Lyon. 1996. *When Wizards Stay Up Late: The Origins of the Internet*. New York: Simon and Schuster.

Harris, Roy. 2000. *Rethinking Writing*. London: The Athlone Press.

Herring, Susan, ed. 1996. *Computer Mediated Communication: Linguistic, Social, and Cross-Cultural Perspectives*. Amsterdam: John Benjamins.

Herring, Susan. 2002. Computer-Mediated Communication on the Internet. *Annual Review of Science and Technology*, vol. 36, ed. Blaise Cronin, 109-168. Medford, NJ: Information Today, Inc.

Herring, Susan. 2003. Gender and Power in Online Communication. *Handbook of Language and Gender*, eds. Janet Holmes and Miriam Meyerhoff. Oxford: Blackwell.

Hellweg, Eric. 2002. Instant Messaging Goes Corporate. *CNN Money*, November 8. http://money.cnn.com/2002/11/08/technology/techinvestor/hellweg.

Hochfelder, David. 1999. *Taming the Lightning: American Telegraphy as a Revolutionary Technology, 1832-1860*. Unpublished dissertation, Department of History, Case Western Reserve University.

Howard, Rebecca M. 1999. *Standing in the Shadows of Giants: Plagiarists, Authors, Collaborators*. Stamford, CN: Ablex Publishing Company.

Hutchins, John. 1995. "The Whisky was Invisible", or Persistent Myths of MT. *MT News International* 11(June):17-18. Available at http://ourworld.compuserve.com/homepages/WJHutchins/Myths.htm.

Katsuno, Hirofumi and Christine R. Yano. 2002. *Asian Studies Review* 26(2): 205-231.

Krug, Steve. 2000. *Don't Make Me Think*. Indianapolis: Que.

Lea, Martin, Tim O'Shea, Pat Fung, and Russel Spears. 1992. 'Flaming' in Computer-Mediated Communication. *Contexts of Computer-Mediated Communication*, ed. Martin Lea, 89-112. New York: Harvester Wheatsheaf.

Lee, Jennifer. 2002. I Think, Therefore IM: Text Shortcuts Invade Schoolwork, and Teachers Are Not Amused. *New York Times*, September 19, E1.

Lenhart, Amanda, Lee Rainie, and Oliver Lewis. 2001. Teenage Life Online: The Rise of the Instant-Message Generation and the Internet's Impact on Friendships and Family Relations. Pew Internet and American Life Project. Available at http://www.pewinternet.org.

Lunsford, Andrea and Lisa Ede. 1994. Collaborative Authorship and the Teaching of Writing. *The Construction of Authorship*, eds. Martha Woodmansee and Peter Jaszi, 417-428. Durham, NC: Duke University Press.

Maes, Pattie. 1995. Intelligent Software. *Scientific American*, March, 84-86.

Maes, Pattie. 1997. CHI97 [Conference on Human Factors in Computing Systems 1997] Software Agents Tutorial. Available at http://pattie.www.media.mit.edu/people/pattie/CHI97/.

Marvin, Carolyn. 1988. *When Old Technologies Were New*. New York: Oxford University Press.

McAlpine, Rachel. 2001. *Web Word Wizardry*. Berkeley, CA: Ten Speed Press.

Moschovitis, Christos J.P., Hilary Poole, Tami Schuyler, and Theresa M. Senft. 1999. *History of the Internet: A Chronology, 1843 to the Present*. Santa Barbara, CA: ABC-Clio.

Olson, David. 1994. *The World on Paper*. Cambridge: Cambridge University Press.

Rheingold, Howard. 2000. *The Virtual Community: Homesteading on the Electronic Frontier*, revised edition. Cambridge (MA): MIT Press.

Rintel, E. Sean, Joan Mulholland, and Jeffery Pittam. 2001. First Things First: Internet Relay Chat Openings. *Journal of Computer Mediated Communication* 6(3). http://www.ascusc.org/jcmc/vol6/issue3/rintel.html.

Romero, Simon. 2002. Most Wanted: Drilling Down / Communication; Text Messaging Takes Off. *New York Times*, December 23, C8.

Sanderson, David. 1993. *Smileys*. Sebastopol, CA: O'Reilly and Associates.

Sproull, Lee and Sara Kiesler. 1991. *Connections: New Ways of Working in the Networked Organization*. Cambridge, MA: MIT Press.

Sebesta, Robert W. 2002. *Programming the World Wide Web*. Boston: Addison Wesley.

Stoddard, P. 1985. The Effects of the WANDAH Program on the Writing Productivity of High School Students. Paper presented at the UCLA Conference on Computers and Writing, May 4-5.

Taylor, Insup and David Olson, eds. 1995. *Scripts and Literacy*. Dordrecht: Kluwer Academic.

Tseliga, Theodora. 2002. Some Cultural and Linguistic Implications of Computer-Mediated Greeklish. Paper presented at the Third International Conference of the Association of Internet Researchers. Maastricht, The Netherlands, October 13-16.

Warschauer, Mark, Ghada R. El Said, and Ayman Zohrn. Language Choice Online: Globalization and Identity in Egypt. *Journal of Computer Mediated Communication* 7(4). http://www.ascusc.org/jcmc/vol7/issue4/warschauer.html.

Yates, JoAnne. 1989. *Control through Communication*. Baltimore: Johns Hopkins University Press.

NOTE: Internet addresses cited in the text and references were functional as of December 2002.

TERMINOLOGY

abbreviations (in CMC)
acronyms (in CMC)
Adobe Acrobat Reader
AIM (America Online Instant Messenger)
alphabetic scripts
AltaVista
AOL (America Online)
applications programs
ARPANET
ASP (Active Server pages)
asynchronous communication

BabelFish
BITNET
blogs (see: Web logs)
bulletin board systems (BBSs)

C++
chat
client-side programming
CMC spectrum
Common Gateway Interfaces (CGIs)
code-switching
ColdFusion
computer mediated communication (CMC)
crawlers (in search engines)
creole

DHTML
dialogue/dialogic (in CMC)
directories (in search engines)
distribution lists (see: listservs)

email
emoticons (=smileys)

file transfer protocol (FTP)
flaming
Flash

GML
Google
Gopher
Group Spécial Mobile/Global System for Mobile Telecommunications (GSM)

home page
HTML

ICQ (“I seek you”)
index (in search engines)
instant messaging (IM)
intelligent agents
Internet Relay Chat (IRC)
Internet (= the Net)

Java
JavaScript
JSP (Java Server Pages)

kaomoji

lingua franca
links
listservs
localization (of Web sites)
logographic scripts

machine translation (MT)
mailing lists (see: listservs)
markup languages
Maxims
meta tags
Microsoft MSN Messenger
monologue/monologic (in CMC)
MOOs (MUDs, Object Oriented)
Mosaic
MUDs (Multi-User Dungeons/Dimensions)

natural language processing (NLP)
Netscape
Netspeak
newsgroups

PDAs (Personal Digital Assistants)
PDF (Portable Document Format)
PERL (=Perl)
PHP (Personal Home Page Tools)
pidgin
process (in CMC)
product (in CMC)

programming languages

quasi-programming languages

RealOne Player

scripting languages
scripts (for natural languages)
search engines
Semantic Web
server-side programming
SGML
short text messaging (SMS)
smileys (see: emoticons)
spiders (see: crawlers)
syllabic scripts
synchronous communication

TELENET
Telnet
TeX
translation

Unicode
UNIX talk, ytalk, ntalk
URL (Uniform Record Locator)
USENET

VB Script
Visual Basic (VB)

Web logs (=blogs)
Web pages
Web sites
World Wide Web (WWW, = the Web)
World Wide Web Consortium (W3C)

XHTML
XML

Yahoo
Yahoo! Messenger

1968-1969	**ARPANET** (US Department of Defense Advanced Research Projects Agency Network) used FTPs (file transfer protocols) for sending documents, data, or computer programs
1969	**Telnet** allowed remote log-ins to ARPANET
1973	**TELENET** commercial packet-switching service; offshoot of ARPANET
late 1970s, early 1980s	**BBSs** (bulletin board systems using telephone dial-up) e.g., Fido, The Well (Whole Earth 'Lectronic Link)
1979-1980	**USENET** (UNIX Users Network) used different networking protocols than ARPANET "poor man's ARPANET" (Abbate 1999:201) distributed online forums called newsgroups (see Figure 3)
1981	**BITNET** (Because It's There/Time Network) cooperative network among IBM systems primarily used for electronic mail
1983	**Internet** ARPANET split into MILNET (for non-classified military information) and ARPANET (for computer research community) Old ARPANET became Internet NOTE: internet = any network using TCP/IP protocols Internet = federally funded network linking multiple networks running such protocols
1990	**WWW** (World Wide Web) created by Tim Berners-Lee
early 1990s	**Gopher** (name of University of Minnesota mascot) system for locating documents on the Internet (arranged hierarchically by topic)
1993	**Mosaic** Web browser created by Marc Andreessen at the University of Illinois system for locating information on the Internet (used word search)
1994	**Netscape** commercial version of Mosaic

Figure 1 Networking Timeline
(main sources: Hafner and Lyon 1996; Abbate 1999; Herring 2002; www.livinginternet.com)

	Variable	Traditional Writing	Face-to-Face Speech
FORM	participants	monologue (no immediate feedback)	dialogue (commonly incorporates feedback)
	time issues	time-independent, durable	real-time, ephemeral
	accessibility	scannable (linear or random access)	only linearly accessible
	structural accoutrements	document formatting (e.g., text layout, punctuation)	prosody (e.g., intonation, volume, pauses)
CONTENT	functions	heavily informational and documentary	heavily social (e.g., phatic, conveys attitudes)
	extralinguistic cues	minimal (e.g., handwriting, stationery choice)	kinesic cues (e.g., facial expression, posture)
	formality	more formal (e.g., no contractions, proper subject-verb agreement and antecedent-pronoun agreement are important)	less formal (e.g., "There's ten points here" and "A person should follow their own dream" are OK)
	internal structuring	more planning, more structured, more syntactically complex	more spontaneous, less structured, less syntactically complex
	editing	more likely to be edited (overall structure, grammar, spelling, punctuation)	little or no editing

Figure 2 Distinctions between Written and Spoken Language

1971	**email** first email sent by Ray Tomlinson
c. 1975	**mailing lists/listservs** ARPA (Advanced Research Projects Agency) shared information, via ARPANET, regarding email protocols LISTSERV (automatic mailing list server) written by Eric Thomas in 1986
1979	**MUDs** (originally: Multi-User Dungeons; today: Multi-User Dimensions) created by Roy Trubshaw and Richard Bartle, University of Essex
1980	**newsgroups** postings made to USENET bulletin board system organized by topic
1982	**emoticons** (also called smileys) created by Scott Fahlman
1980s, **early 1990s**	**early instant messaging** (e.g., UNIX talk, ytalk, ntalk; MIT's Project Athena Zephyr)
1988	**IRC** (**Internet Relay Chat** – earliest modern **chat** system) created by Jarkko Oikarinen to run on Unix computers
1990	**Web pages** identified by URLs (Universal Resource Locators) created by Tim Berners-Lee as part of design of the WWW (see Figure 1)
1990	**MOOs** (MUDs, Object Oriented) created by Pavel Curtis
1992	**SMS** (originally: short messaging system, but usually referred to as "short text messaging") part of European GSM wireless telephone standard
1996	**ICQ** ("I Seek You") instant messaging system created by Mirabilis Ltd (Israel) purchased by AOL in 1998
1997	**AIM** (America Online Instant Messenger, running on the Internet) largest IM system used in US major competitors: Microsoft MSN Messenger, Yahoo! Messenger
1997	**Web logs** (alternatively known as **blogs**) named by Jorn Barger

Figure 3 Computer Mediated Communication Timeline
(main sources: Hafner and Lyon 1996; Cherny 1999; Berners-Lee 2000; Rheingold 2000; Herring 2002; http://corp.aol.com; http://rebeccablood.com; www.livinginternet.com)

PRODUCT (monologue) **PROCESS** (dialogue)

←----------→

CATEGORY	completed works	Web sites	anonymous dialogue	one-to-many dialogue (identified interlocutors)	one-to-one dialogue (identified interlocutor)
EXAMPLES	academic papers, business reports	Web pages Web logs (blogs)	newsgroups MUDs, MOOs chat (including IRC)	listservs	email IM SMS
COMMENTS	available through self-archiving or attachments	increasing options for comments and interaction	some forums insist on vetting participants	some participants can enter under pseudonyms	email, IM may have multiple recipients

Figure 4 Computer Mediated Communication Spectrum

Note: `>' indicates text from earlier email exchange

Nadia,
Friday, April 12 at 11:45 at the University Club sounds perfect. But now I have to figure out another reading for your oblique message....
Naomi

NHARRIS <nharris@american.edu>

04/02/02 08:25 AM
Please respond to nharris

To: nbaron@american.edu
cc:
Subject: Re: Hi

```
Naomi,
The 12th sounds great.  I have a class at 12:45.  Would
11:45 suit you?  We'd have to go to the University Club and
save the Thai or other places for after the end of classes.
Looking forward to seeing you. Both of your guesses are
wrong!
Nadia

nbaron@american.edu wrote:

> Nadia,
>
> Now you have me very curious. Are you talking about crime
in Paris or a new decadent dessert?
>
> How does Friday, April 12 sound? Let me know what time
and where.
>
> Naomi
>
>   Nadia Harris
>  <nharris@ameri       To:     nbaron@american.edu
>  can.edu>             cc:
>                       Subject:     Hi
>   04/01/02 08:00 PM
>
> Hi Naomi, how about meeting for lunch some Friday soon so
I can tell you about my trip and you can tell me about your
break and all the rest. I need to warn you about a new
danger I discovered in Paris.  This Friday, I can't, but
any other will be fine.  Let me know.
>
> Nadia
```

Figure 5 Example of Email
(email exchange, with history, between Naomi Baron and Nadia Harris)

Two IM sessions are taking place simultaneously. The conversation on the left is between sirmixalot48 and Ari009 (user screen names). The conversation on the right is between sirmixalot48 and jyflwr6 (again, screen names).

Conversation 1	**Conversation 2**
sirmixalot48: hellooooooooo	iyflwr6: I feel like a loser
sirmixalot48: grr	sirmixalot48: you're mad
Ari009: sorry	jyflwr6: so?
Ari009: corey came in	sirmixalot48: definitely
sirmixalot48: k	
sirmixalot48: grr	sirmixalot48: who want to do that/
Ari009: grr?	sirmixalot48: no me
Ari009: why grr?	jyflwr6: poor thing…it could be worse…
sirmixalot48: dunno	you could be in class…

Figure 6 Examples of Instant Messaging
(IM exchanges courtesy of Sara Tench)

1. Christy: hi gem tx 4 ur card lovely 2 hear from u! cant do sat but wld luv 2 meet soon let me know when cx [tx = thanks; x = love or with a kiss, hence cx = love from Christy]

   Gemma: hellooo! not 2 worry! let us know some possible dates over crimbo / early jan & we can plan a soiree! hows new man? Gem xx [crimbo = Christmas vacation]

2. Chisty: hiya wont make it 2 gym this am as cldnt get out of bed sorry!cx

   Kate: no probs-couldn't face it after last night!will in thurs after work-are you?

3.Christy: hi f r we still meeting 2nite? [f r = addressee's initials]

  Francesca:yes cant w8! c u 8.30 EEE chiswick fx [EEE = a restaurant called Est Est Est [fx = love from Francesca]

4. Christy: hello dad hope ur meeting went ok 2day. tx 4 helping me move. c u @ 6 by chancery lane. lol cpk xxx [LOL =lots of love]

   [Dad: rings me back later, as he can read text messages but has never got the hang of sending them himself!]

5. Christy: hiya did i just send u random text re van hire? meant 2 send 2 jen sorry! tx for party last nite, whats marys surname as need to send stuff

   Heidi: stainforth. they really liked u. ys did get random txt but ignord. joind gym. induction tues.

6. Christy: hiya just realised cant do swimming tues as have bk launch, sorry! cx [bk = book]

   Alice: Hi chris, do u wanna sport another night instead/ wkend sport? what u doing fri? Have fun at bk launch and w LP! Lots of love al x x x [LP = friend's initials; al = Alice]

Figure 7 Examples of Short Text Messaging
(examples of SMS messages and responses courtesy of Christabel Kirkpatrick)

```
Send Air-l mailing list submissions to
             air-l@aoir.org

To subscribe or unsubscribe via the World Wide Web, visit
             http://www.aoir.org/mailman/listinfo/air-l
or, via email, send a message with subject or body 'help' to
             air-l-request@aoir.org

You can reach the person managing the list at
             air-l-admin@aoir.org

When replying, please edit your Subject line so it is more
specific than "Re: Contents of Air-l digest..."


Today's Topics:

1.    Technology & Tolerance (Philip N. Howard)
2.    job: .au: .qld: .cyberstudies (Adrian Miles)
3.    Re: Technology & Tolerance (Gina Neff)
4.    Internet use and social network diversity (Keith N Hampton)

 --__--__--

Message: 1
From "Philip N. Howard" <pnhoward@u.washington.edu>
To: <air-1@oir.org>
Date: Mon, 2 Dec 2002 10:46:53 - 0800
Organization: University of Washington
Subject: [Air-1] Technology & Tolerance
Reply-To: air-1@aoir.org

I'm editing a book with contributions from a number of Airers.
The collection will have one piece by John Robinson, Alan
Neustadlt, and Meyer Kestnbaum at Maryland called "Technology &
Tolerance: Public Opinion Differences Among Internet Users and
Non-Users" that takes a step closer to making a causal connection
between experience online and tolerant attitudes. Here's a
teaser:

    . . .
```

Figure 8 Example of a Listserv
(Portion of the listserv posting of December 3, 2002 of the
Association of Internet Researchers [AoIR], courtesy of AoIR)

Note: Use of `>' indicates lines quoted from an earlier post

>Well, it seems she got some QUALITY time in
>yesterday, firing Carter Jones!
>Go Brooke! Brainslap of the Week! She held up
>to him and didn't back off or squirm! I loved
>it … :-)

I thought the same thing, too, "Finally, Brooke is giving it to Carter with both guns." But then when she takes the incriminating picture, she ONLY TAKES THE PICTURE leaving Carter with the negative in the darkroom. You don't have to work for Kodak to know that he can still make more pictures which he does.

Figure 9 Example of a Newsgroup Session
rec.arts.tv.soaps, October 2, 1992
(from Baym 2000, pp. 125-126)

Note: The user's verbal input is preceded by the `>' prompt.

>look
Corridor
The corridor from the west continues to the east here, but the way is blocked by a purple-velvet rope stretched across the hall. There are doorways leading to the north and south.
You see a sign hanging from the middle of the rope here.
Gumby is here.
>:waves.
Munchkin waves.
>read sign
This point marks the end of the currently-occupied portion of the house.
Guests proceed beyond this point at their own risk.
-- The residents
Gumby says, "What're you up to?"
>"Just exploring this place. Bye!
You say, "Just exploring this place. Bye!"
Gumby waves bye-bye.
>go east
You step disdainfully over the velvet rope and enter the dusty darkness of the unused portion of the house.

Figure 10 Example from a MUD Session
(from Curtis and Nichols 1993, Figure 1. A typical MUD encounter)

1. [StinGer] vp: to whom?
2. [bourbon] wizz i here
3. [CestLV] mayhem : oh it is.....it is....but i wouldn't open it...it just might explode
4. [lion] hi everyone?
5. [Stinger] hey angie
6. [AJ] strawb: My computer is hanging wait
7. [Angie] Hey everyone!
8. [cookie] hi lion
9. [SERVER] AJ has quit IRC Leaving
10. [WiseOne] Angie is back!
11. [ACTION] mayhem kneels and says hi back to vp_man
12. [SERVER] vp_man has quit IRC Leaving
13. [walker] Hey Jfonda!!!
14. [strawb] hello lion
15. [ACTION] cookie thins the cafeteria food is getting worse
16. [Angie] Stinger hello
17. [Zool-MODE] Has changed Cinche's mode to +o
18. [lion] how goes it cookie?
19. [Lion] hello strawb
20. [SERVER] kwest!xx@xx.xx.xx.xx has joined this channel
21. [SERVER] ruby!xx@xx.xx.xx.xx has joined this channel
22. [SERVER] Fleet!xx@xx.xx.xx.xx has joined this channel

Figure 11 Example of Chat
(from Rintel, Mulholland, and Pittam 2001, adapted from Example 1)

[Please insert screenshot of home page noted below]

Figure 12 Example of a Web Page
(Home page of American University, Washington, DC, available at http://www.american.edu; courtesy of American University)

'brary blog

writing for me. (and you)

Monday, July 01, 2002 7:29 AM
Receiving Information Today's BUDDIE (Best Unknown Database) Awards, the Urban Legends Reference Pages. Covers more than a thousand urban legends, both of the pre-web variety and the mass-emailing kind.

Wednesday, June 26, 2002 12:13 PM
The 9th Circuit rules on the (un)constitutionality of the pledge of allegiance. Download the opinion here [pdf].

Monday, June 24, 2002 6:34 AM
An updated Enron bibliography at LLRX.

Wednesday, June 12, 2002 11:57 AM
A very rich presentation format for an online magazine, Exporatorium.

Tuesday, June 04, 2002 6:05 AM
Apparently May was historic preservation month (where did May go?), so consider this a better-late-than-never: The Georgetown University Law Center is working to publish the decisions of the DC Mayor's Agent for Historic Preservation. Not otherwise available, the Library is receiving them electronically and converting to .html and .pdf. Check it out!

Wednesday, May 29, 2002 6:07 AM
Did you know you could shortcut a Google search by typing your search query in the address bar, rather than waiting for Google's screen to load? If you have a simple search – looking for the American Airlines website, for example -- try entering a URL like this:

http://www.google.com/search?q=american+airlines

One click to the address bar, one round of typing, one hit of the "enter" button, and you have results! Cool, eh?
Advanced search features can be added in as well, for example:

http://www.google.com/search?q=northwest+airlines

previouslys
2002
1 | 2 | 3
2001
1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12
2000
1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12

fodder for the blog
daily
Arts & Letters Daily
Librarian.net
Chronicle
Research Buzz
Webnoize
weekly
Rogue Librarian
FreePint
Internet Scout Weblog
Library Juice
LISNews.com
Slashdot
Scout report
New Breed Librarian
AALLnet Career Hotline
LLRX Buzz
monthly or so
The CRIV page (formerly the Legal Publishers' List)
Direct Search
About.com
Jurist
who owns what
Legal Publishers' list
LibDex
tools
backflip
popup killer
IP lookup
other
me, etc.

Modified: *07/01/2002 17:57:55* ***Modified:*** *February, 2002*
Comments: *Stephanie Davidson*

Figure 13 Example of a Web Log
(Selection from Web log of Stephanie Davidson, available at http://chickeninthewoods.org/librariness; courtesy of Stephanie Davidson)

Markup Languages	**Programming Languages**	**Quasi-Programming Languages and Applications Programs**
tags adding meaning to Web documents e.g., HTML, XML	traditional computer languages, sometimes adapted for Web Use e.g., C^{++}, Java	programming tools designed for specific Web applications e.g., JavaScript, ASP

Figure 14 Web "Languages" and Programs

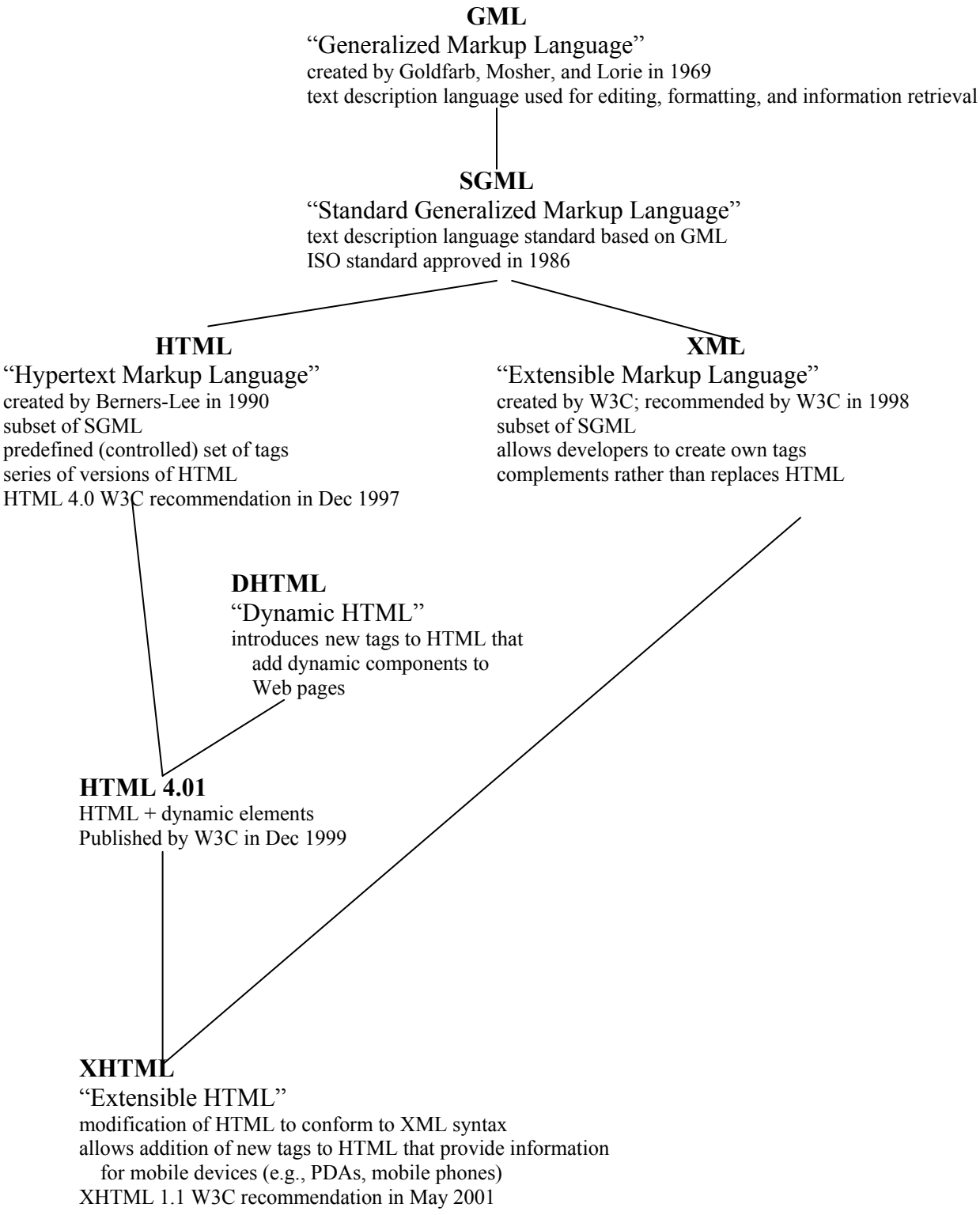

Figure 15 Web Markup Languages