On Second Generation Hypermedia Systems

K.Andrews, F.Kappe, H.Maurer, K.Schmaranz
(IICM, Graz University of Technology, Austria
hmaurer@iicm.tu-graz.ac.at)

Abstract: In this paper we claim that the navigational and structural tools currently available on the Internet are not sufficient to fully exploit the tremendous power of the largest information and communication ressource mankind has ever had. We contend that current hypermedia systems and its most prominent specimen WWW do not have enough functionality to provide the power that is needed. We explain important features that are absent, claim that "second generation" hypermedia systems incorporating such features are essential and mention a first such second generation hypermedia system called Hyper-G, which is just becoming available and is starting to be used for a wide variety of applications.

1 Introduction

The steady growth of the Internet [Fenn et al 1994] has made resource discovery and structuring tools increasingly important. Historically first was the introduction of various dictionary servers, Archie [Deutsch 1992] being probably the first and most prominent. As an alternative to having servers that need constant updating, WAIS [Kahle et al 1992] introduced a powerful search engine permitting full-text searches of large data-bases and returning a ranked list, the ranking based on a heuristic approach. Although directory servers like Archie (to locate a database) and search engines like WAIS (to locate a desired item in that database) alleviated the problem of finding information in the Internet somewhat, it soon became apparent that other techniques would also be necessary.

The most important such technique is to emphasise the a-priori organisation of information, rather than try to search for information in a universe of completely different databases. Two efforts in this direction have proved particulalry successful: Gopher, originally developed at the University of Minnesota [Alberti et al 1992] and the World-Wide Web (WWW, W3, or "The Web" for short) originally developed at CERN in Geneva [Berners-Lee et al 1994].

In both cases information is stored in a simple structured fashion on servers, and can be accessed via clients, with clients available for most major hardware platforms. Over 3000 Gopher and WWW servers are currently reachable in the Interent, albeit most of them with little more than token presentation of the institution running the server. There are some very notable exceptions, however. Examples of substantial Gopher and WWW databases include the EARN ("European Academic & Research Network") Gopher server (gopher.earn.net), the ACM Siggraph Gopher server (siggraph.org), he CERN WWW Server - the birthplace of WWW (http://www.cern.ch), MUSE - a Hypermedia Journal from the Johns Hopkins University (http://muse.mse.jhu.edu), Nando ("News and Observer") News net (http://www.nando.net), and The Canadian Internet Handbook WWW Server (http://www.csi.nb.ca/handbook/handbook.html).

Information in Gopher is structured in a hierarchical fashion using menus, an access technique which, though simple to use, has many well-known weaknesses. Information in WWW is structured in documents; documents are linked together according to the hypertext-paradigm (see [Conklin 1987], [Tomek et al 1991] and [Koegel-Buford 1994] for a general and thorough discussion of hypertext and hyperemdia). "Anchors" within documents are associated with "links" leading to other documents. Although many stand-alone systems using the hypertext-metaphor have emerged since the introduction of HyperCard on the Mac in 1987, WWW can claim to be the first wide-spread hypertext system whose component servers are accessible via the Internet. Indeed, WWW is not just a hypertext system but a hypermedia system, i.e. documents can comprise text, image, and audio and film clips.

WWW servers are easy to install and clients are available on all major platforms. All software is free and sources are available. The node-link technique for navigating and finding information is quite appealing at least for small to medium amounts of data, and the mix of media makes the use of WWW aesthetically pleasing. All this has contributed to the proliferation of WWW, recently overtaking Gopher in terms of number of installed servers. Indeed there is no doubt that WWW is not only the first widespread hypermedia system available through the Internet, but that WWW has actually replaced some earlier more traditional information systems. The success of WWW, the number of WWW proponents and freaks, and its publicity even in non-scientific publications like Time magazine may create the impression that WWW is the solution for most information needs and will remain the dominating system for the forseeable future.

The reality is different, however. Whilst WWW is undoubtedly a big step forward compared to pre-WWW times, experience shows that much functionality required for sizeable applications is missing from WWW. In this sense, WWW should be considered a first generation networked hypermedia system. More advanced "second generation" hypermedia systems are required to cope with the problems currently being encountered on the Web. Just to give one example, while pure node-link navigation is satisfactory in small systems it tends to lead to confusion and disorientation, if not chaos, when applied to large amounts of data [Conklin 1987]. For substantial applications, some additional structuring and searching facilities are clearly required. That links may actually be more harmful than useful has been already pointed out in [Van Dam 1988] and elaborated in [Maurer et al 1994]. Similarly, the necessity to keep links separate from rather than embedded in documents as is the case in WWW has already been demonstrated in the pioneering work on Intermedia at Brown University [Haan et al 1992].

In Section 2, we concentrate on features we find desirable in second generation hypermedia information systems. We compare the features found in the first generation hypermedia system WWW with those found in what might be the first second generation hypermedia system, Hyper-G. This is not to belittle WWW or to glorify Hyper-G, but rather to clarify why certain facilities are needed. In Section 3, we briefly look at communicational and cooperational features that will have to be integrated in hypermedia systems if they are to be successful: such features are currently scarcely supported by any hypermedia system, they are often dealt with in the context of computer supported cooperative work, rather than hypermedia. In Section 4 we look in a little more detail at Hyper-G: it provides a smooth transition from WWW and hence from the first generation to the second, its increased functionality supporting a range of new applications.

2 Hypermedia Information Systems: Why First Generation Features are not Enough

In this section we explain some of the properties of first generation hypermedia systems, using WWW as the most prominent example. We contrast them with those of Hyper-G, the first second generation model. We confine attention to networked hypermedia systems with a client/server architecture.

Information in a hypermedia system is usually stored in "chunks". Chunks consist of individual documents which may themselves consist of various types of "media". Typically, a document may be a piece of text containing a picture. Each document may contain links leading to (parts of) other documents in the same or in different chunks. Typical hypertext navigation through the information space is based on these links: the user follows a sequence of links until all relevant information has hopefully been encountered.

In WWW, a chunk consists of a single document. Documents consist of textual information and may include pictures and the (source) anchors of links. Pictures and links are an integral part of the document. Pictures are thus placed in fixed locations within the text ("inline images"). Anchors can be attached to textual information and inline images, but not to parts of images. Links may lead to audio or video clips which can be activated. The textual component of a document is stored in so-called HTML format, a derivative of SGML.

In Hyper-G the setting is considerably more general: chunks, called "clusters" in Hyper-G terminology consist of a number of documents. A typical cluster may, for example, consist of five documents: a piece of text (potentially with inline images), a second piece of text (for example in another language, or a different version of the same text, or an entirley different text), a third piece of text (the same text in a third language perhaps), an image and a film clip. Anchors can be attached to textual information, to parts of images, and even to regions in a film clip. Links are not part of the document but are stored in a separate database. They are both typed and bidirectional: they can be followed forward (as in WWW) but also backwards. The textual component of a document is usually stored in so-called HTF format, also a derivative of SGML, but can also be stored as a PostScript file.

The support for multiple pieces of text within a cluster allows Hyper-G to handle multiple languages in a very natural way. It also elegantly handles the case where a document comes in two versions: a more technical (or advanced) one and one more suitable for the novice reader. As indicated, pictures can be treated as inline images or as separate documents. Often, inline images are convenient, since the "author" can define where the user will find a picture in relation to the text. On the other hand, with screen resolution varying tremenduously, the rescaling of inline images may pose a problem: if a picture is treated as separate document, however, it appears in a separate window, can be manipulated (shifted, put in the background, kept on-screen while continuing with other information, etc.) independent of the textual portion (which in itself can be manipulated by for example narrowing or widening its window). Thus ,the potential to deal with textual and pictorial information separately provides more flexibility when required. As has been mentioned,text can be stored in Hyper-G not only in HTF, but also in PostScript format. Since most printers are geared towards printing PostScript files, almost all word processing packages are capable of producing PostScript files as output: thus, all word processing packages can be used to prepare data for Hyper-G using this approach. Also, PostScript files allow the incorporation of pictures and formulae; they offer the user the possibility to view documents exactly as if they were printed (given high enough resolution screens), and such Hyper-G documents can be printed with the usual professional PostScript quality. For one of the major applications of Hyper-G, the Journal of Universal Computer Science (see Section 4), a full-text search engine has been implemented for PostScript files as well as full hyperlinking facilities. The use of standard compression techniques allows the PostScript files to be compacted to about the same size as equivalent HTF and HTML files. Thus, the use of PostScript with high-quality PostScript viewers built into the native Hyper-G clients Amadeus and Harmony (for MS-Windows and X Windows respectively), gives Hyper-G the necessary professionalism for high quality electronic publishing of journals, books, and manuals.

In addition to the "usual" types of documents found in any modern hypermedia system, Hyper-G also supports 3D objects and scenes. The native X Windows client for Hyper-G (Harmony) provides four different ways to navigate within such 3D models. Finally, Hyper-G allows the use of documents of a "generic" type. This permits future extensions and the encapsulation of documents otherwise incompatible with Hyper-G.

Let us now turn to the discussion of the philosophy of links in WWW versus Hyper-G. The ability to attach links to parts of a picture is clearly desirable, when additional information is to be associated with certain sub-areas of an image. That links are bidirectional and not embedded in the document has a number of very important consequences: first, links relating to a document can be modified without necessarily having access rights to the document itself. Thus, private links and a certain amount of customisation are possible; second, when viewing a document it is possible to find all documents refering to the current one. This is not only a desirable feature as such, but is of crucial importance for being able to maintain the database. After all, when a document is deleted or modified, all documents refering to it may have to be modified to avoid the "dangling link syndrome", or to avoid being directed to completely irrelevant documents. Hyper-G offers the possibilty of automatically notifying the owner of a document that some of the documents that are being refered to have been changed or deleted, an important step to "automatic link maintainance". Thirdly, the bidirectionality of the links allows the graphic display of a "local map" showing the current document and all documents pointing to it and being pointed at, an arbitrary number of levels deep. Harmony makes full use of this fact and provides local maps as an invaluable navigational aid that cannot be made available for WWW databases ([Andrews et al 1994], [Fenn et al 1994]). Finally, the fact that links can have types can be used to show to the user that a link just leads to a footnote, or to a picure, or to a film clip, or is a counter- or supporting argument of some claim at issue: typed links enhance the perception of how things are related and can be used as tool for discussions and collaborative work.

Navigation in WWW is performed solely using the hypertext paradigm of anchors and links. It has become a well accepted fact that structuring large amounts of data using only hyperlinks such that users don't get "lost in hyperspace" is difficult to say the least. WWW databases are large, flat networks of chunks of data and resemble more an impenetrable maze than well-structured information. Indeed every WWW database acknowledges this fact tacitly, by preparing pages that look like menus in a hierarchically structured database: items are listed in an orderly fashion, each with an anchor leading to a subchapter (subdirectory). If links in WWW had types, such links could be distiguished from others. But as it is, all links look the same: whether they are "continue" links, "hierarchical" links, "referential" links, "footnote links", or whatever else.

In Hyper-G not only can have links a type, links are by no means the only way to access information. Clusters of documents can be grouped into collections, and collections again into collections in a pseudo-hierachical fashion. We use the term "pseudo-hierarchical" since ,technically speaking, the collection structure is not a tree, but a DAG. I.e., one collection can have more than one parent: an impressionist picture X may belong to the collection "Impressionist Art", as well as to the collection "Pictures by Manet", as well as to the collection "Museum of Modern Art". The collection "hierarchy" is a powerful way of introducing structure into the database. Indeed many links can be avoided this way [Maurer et al 1994], making the system much more transparent for the user and allowing a more modular approach to systems creation and maintainance. Collections, clusters and documents have titles and attributes. These may be used in Boolean queries to find documents of current interest. Finally, Hyper-G provides sophistacted full-text search facilities. Most importantly, the scope of any of such searches can be defined to be the union of arbitrary collections, even if the collections reside on different servers. (We will return to this important aspect of Hyper-G as a distributed database below).

Note that some WWW applications also permit full-text searches. However, no full-text search engine is built into WWW. Thus, the functionality of full text search is bolted "on top" of WWW: adding functionality on top of WWW leads to the "Balkanisation", the fragmentation of WWW, since different sites will implement missing functionality in different ways. Thus, to stick to the example of the full text search engine, the fuzzy search employed by organisation X may yield entirley different results from the fuzzy search employed by organisation Y, much to the bewilderment of users. Actually, the situation concerning searches in WWW is even more serious: since documents in WWW don't have attributes, no search is possible on such attributes; even if such a search or a full text search is artificially implemented, it is not possible to allow users to define the scope for the search, due to the lack of structure in the WWW database. Hence full-text searches in WWW always work in a fixed, designated part of the WWW database residing on one particular server.

As was mentioned before over 3000 WWW servers are currently installed, and are accessible via the Internet. However, there is no coherence bewtween the servers: if a user wants to search for an item in a number of WWW servers the user has to initiate a new search for each server. This problem is compounded by the fact that WWW knows only two types of access rights: everything allowed (webmaster) or read-only access. Neither are there shades in between, nor is it possible to allow certain users to edit some parts of a WWW server, other users to edit other parts.

In contrast Hyper-G provides various types of access rights and the definition of arbitrarily overlapping user groups. Hyper-G is also a genuine distributed database: servers (independent of geographical location) can be grouped into collections, with the hyperroot at the very "top". Thus, users can define the scope of searches by defining arbitrary sets of collections on arbitary servers. Different groups can work with the same server without fear of interfering with someone else's data. To be more concrete, suppose 10 departments within a university intend to operate a WWW database. If they operate one server together and if all want to input their own data, the data of department X is not protected from any kind of access or change by department Y! Hence the tendency would be to operate 10 different servers. (Indeed, there are many more WWW servers than there are server sites, clearly demonstrating this phenomenon.) However, if the 10 departments operate 10 different servers and a user from outside wants to look up a person without knowing the department, the user is forced to query all of the servers, one after the other.

Hyper-G, being a distributed database with well-defined access rights of fine granularity, offers a much more satisfactory solution: the 10 departments operate a single server, different users have different access rights: not only can department X not influence the information of department Y, certain parts of the database may have even their read access restricted to certain groups or even to single individuals ("private collections"). Hyper-G may be used anonymously, but if users identify themselves they will automatically be shown their "home collection", where they have collected the information most important for them, and from where they can enter all those parts of the database to which access is permitted for them. Continuing the earlier example, suppose an outside user looks for a certain person. Accessing the single Hyper-G server operated by the 10 departments with a full-text search will find the information, assuming it is present. However, suppose the departments insist on each operating their own Hyper-G server: by simply defining a collection "servers of this university", a single Hyper-G search will still examine all of the 10 databases (assuming they are in a LAN or on the Internet). Even now, the servers offer possibilities not available without proper access control: members of the departments may keep some information just for themselves, or for a group they collaborate with, etc.

If one has access to a local Hyper-G server, all accesses to other Hyper-G servers, but also WWW, Gopher and WAIS sites are routed through the local Hyper-G server. Documents retrieved once are automatically cached (for all users of that server), so they will no longer be retrieved from the remote database next time around. Using the separate link database, it can be assured that new versions of a cached document are automatically retrieved when a request for accessing the document is issued. Although recent WWW servers also support caching, the consistency of cached documents cannot be guaranteed. As we will discuss in Section 4, caching in Hyper-G applies equally to documents from non-Hyper-G servers. Hence, using a local Hyper-G server may be quite valuable, even if that server is used for nothing much beyond caching!

The acceptance of a hypermedia system is certainly not only dependent on deep technical features, but above all on the information content and the ease of use. Due to the fact that large hypermedia systems tend to lead to disorientation, second generation hypermedia systems have to try very hard, both at the server and at the client end, to help users with navigational tools. Some navigational tools, like the structuring and search facilities have already been described; others, such as maps, history lists, specific and personal collections can also be of great help and are available in Hyper-G; a particular speciality of the Harmony client (assuming an OpenGL environment) is a 3D browser: the "information landscape" depicts collections and documents (according to their size) as blocks of varying size spread out across a three-dimensional landscape, over which the user is able to fly.

3 Hypermedia for Communication and Collaboration

First generation hypermedia systems like WWW have traditionally been seen mainly as (simple) information systems. Most applications currently visible support this view: very often WWW servers offer some pleasantly designed general information on the server-institution, but only rarely does the information go much deeper. If it does, usually a "hybrid" system is used, WWW with some add-ons using the scripting interface of WWW.

It is our belief that hypermedia systems acting as simple information systems, where someone inputs information to be read by other users, do not offer much potential: they will disappear into obscurity sooner rather than later. To ensure the success of a hypermedia system, it must allow users also to act as authors, allow them to change the database, create new entries for themselves or other users, create a personal view of the database as they need it, and, above all, allow the system to be used also for communication and cooperation.

First generation hypermedia systems like WWW almost entirly lack support for such features. Emerging second generation hypermedia systems are bound to incorporate more and more features of the kind mentioned; Hyper-G provides a start.

The native Hyper-G clients Amadeus and Harmony are designed to allow the easy import of data into the server. (Note: At the time of writing not all functions desirable are available yet. They will be by the end of 1994). They are also designed to allow point-and-click link generation: select the source anchor location with a mouse-click, select the destination anchor with a mouse-click and confirm that a link should be created.

Hyper-G supports annotations (with user-definable access rights): in contrast to some WWW clients which also allow annotations which are then kept locally, Hyper-G annotations become part of the database, i.e. are also available when working with other clients, or from another user account or machine. Annotations can themselves be annotated; the network of annotations can be graphically displayed using the local map function of Harmony. Thus, the annotation mechanism can be used as the basis of (asynchronuous) computer-conferencing, and has been sucessfully employed in this fashion. The client-server protocol in WWW is "static" in the sense that the server can only react to requests by the client, but cannot become active itself. In Hyper-G the client-server protocol is "active" in the sense that the server can contact the client: this can be used for example to send update notification to a client, and provides the first (rudimentary) possibillities for client-client communication for synchronuous communication, conferencing and cooperation.

We believe that many of the features discussed in the area of computer supported cooperative work [Devan 1993] will eventually be incorporated into second generation hypermedia systems. This approach is also planned for Hyper-G, but will not be fully supported for some time yet.

Some of the most widely used functions of the Internet are file transfer (FTP) and electronic mail. Hence, second generation hypermedia systems have to support both FTP and particularly email. Without leaving their hypermedia environment, users must be able to edit, send, and receive email. Email should automatically be presorted by criteria such as subject, author, date, etc. Related pieces of email can be linked together, the local map feature presenting a good graphical overview of the flow of the email discussion pertaining to a certain subject. The hypermedia system should also have the possibility to send mail with delays or on certain dates to act as reminder and as an active personal scheduler. A number of relevant ideas are collected in [Kappe et al 1993b] and are currently under implementation for Hyper-G.

4 Hyper-G: A Smooth Transition from First to Second Generation Hypermedia Systems

As has become clear from the above discussion, first generation hypermedia systems such as WWW do not have enough functionality to serve as a solid and unified basis for substantial multi-user information systems with a strong communicational component.

Hyper-G is a first attempt to offer much more basic functionality, yet to continue the path started by WWW and remain fully interoperable with WWW: every WWW client can be used to access every Hyper-G server, albeit occasionally with some loss in functionality; a Hyper-G client may, through a Hyper-G server, access WWW, Gopher, and WAIS servers without any loss of functionality, indeed providing "free" caching as a by-product.

The compatibility of Hyper-G with WWW and Gopher actually goes much further: tools to import complete WWW and Gopher databases into Hyper-G without manual intervention are in preparation. Thus, users of WWW can migrate up to an environment allowing all kinds of searches, access control, etc., without being forced to abandon their current database or their favourite WWW client.

Hyper-G was released in July 1994, and already has substantial following. A slate of further tools to make working with Hyper-G even easier will be completed by the end of 1994.

Hyper-G is the most powerful networked multimedia system currently available, and is free of charge for all educational institutions. The source code of the major clients (with the exception of some proprietory segments) is or will be available for developers.

Due to its functionality, Hyper-G is used for a wide variety of applications:

- As a basis for university information systems (with substantial information content at Graz University of Technology and The Univesity of Auckland, and in experimental use at a number of other universities)
- As an organisation-wide information and communication system (the European Space Agency, ESA, being the largest user sofar)
- As a multimedia infrastructure for museums and exhibitions (MONZ, the new Museum Of New Zealand, the Interactive Information Center in Styria, and the AEIOU project as three major examples)

- As an infrastructure for teleteaching experiments (the University of Auckland in cooperation with NZ Telecom as a first test-site)
- As an infrastructure for electronic publishing (with a German publishing consortium including Meyer/Brockhaus/BI/Langenscheidt for reference books, and Springer in connection with the new Journal of Universal Computer Science with a backbone of 65 Hyper-G servers as examples)
- As an infrastructure for the cooperation amongst mathematicians in Germany (as recommended by the "Deutsche Mathematiker Vereinigung")

Whether Hyper-G will ever be as wide-spread as Gopher still is and as WWW has started to be, nobody can know. What is clear, is that time is working against first generation systems and in favour of the increased functionality and universality, of second generation systems. Hyper-G will certainly contribute to and speed up developments in this direction by demonstrating the feasability of new concepts and of a new generation of networked multimedia systems.

More information and software concerning Hyper-G is available by anonymous FTP from "iicm.tu-graz.ac.at" directory "pub/Hyper-G". For more information on J.UCS, send an email with the subject "[info]" or the word "info" as first and only line of your message to "jucs@iicm.tu-graz.ac.at" or look on the above FTP server in directory "pub/JUCS".

References

[Alberti et al 1992] Alberti,B., Anklesaria,F., Lindnder,P., McCahill,M., Torrey,D.; Internet Gopher Protocol: A Distributed Document Search and Retrieval Protocol; FTP from boombox.micro.umn.edu, in directory pub/gopher/gopher_protocol.

[Andrews et al 1994] Andrews,K., Kappe,F.; Soaring Through Hyperspace: A Snapshot of Hyper-G and its Harmony Client; Proc. of Eurographics Symposium on Multimedia/Hypermedia in Open Distributed Environments, Graz (1994).

[Berners-Lee et al 1994] Berners-Lee,T., Cailliau,R., Luotonen,A., Nielsen,H., Secret,A.; The World-Wide Web; Communications of the ACM 37,8 (1994), 76-82.

[Conklin 1987] Conklin,E.J.: Hypertext:an Introduction and Survey; IEEE Computer 20 (1987), 17-41.

[Devan 1993] Devan,P.: A Survey of Applications of CSCW Including Some in Educational Settings; Proc. ED-MEDIA'93, Orlando (1993), 147-152.

[Deutsch 1992] Deutsch,P.: Resource Discovery in an Internet Environment-the Archie Approach; Electronic Networking: Research, Applications and Policy 1,2 (1992), 45-51.

[Fenn et al 1994] Fenn,B., Maurer,H.: Harmony on an Expanding Net; ACM Interactions 1,3 (1994), to appear.

[Haan et al 1992] Haan,B.J., Kahn,P., Riley,V.A., Coombs,J.H., Meyrowitz,N.K.: IRIS Hypermedia Services; Communications of the ACM 35,1 (1992), 36-51.

[Kahle et al 1992] Kahle,B., Morris,H., Davis,F., Tiene,K., Hart,C., Palmer,R.: Wide Area Information Servers: An Executive Inforamtion System for Unstructured Files; Electronic Networking: Research, Applications and Policy 1,2 (1992), 59-68.

[Kappe et al 1993a] Kappe,F., Maurer,H., Scherbakov,N.; Hyper-G – A Universal Hypermedia System; Journal of Educational Multimedia and Hypermedia 2,1 (1993), 39-66.

[Kappe et al 1993b] Kappe,F., Maurer,H.: From Hypertext to Active Communication/Information Systems; IIG Report 363 Graz, Austria (1993); To appear in the Journal of Microcomputer Applications.

[Koegel-Buford 1994] Koegel-Buford,J.: Multimedia Systems; ACM Press, SIGGRAPH Series (1994).

[Lennon et al 1994a] Lennon,J., Maurer,H.: You Believe What Multimedia Is? And What Internet Will Do For You? Well...Think Again!; To appear in Datamation.

[Lennon et al 1994b] Lennon,J., Maurer,H.: Lecturing Technology: A Future With Hypermedia; Educational Technology 34,4 (1994), 5-14.

[Maurer 1994] Maurer,H.: Advancing the ideas of World-Wide Web; Proc. Conf. on Open Hypermedia Systems, Honolulu (1994), 201-203.

[Maurer et al 1994] Maurer,H., Philpott,A., Scherbakov,N.: Hypermedia Systems Without Links; Journal of Microcomputer Applications 17,4 (1994).

[Tomek eta l 1991] Tomek,I., Khan,S., Muldner,T., Nassar,M., Navak,G., Proszynski,P.: Hypermedia – Introduction and Survey, Journal of Microcomputer Applications 14,2 (1991), 63-100.

[van Dam 1988] Van Dam, A.: Hypertext'87 Keynote Adress; Communications of the ACM 31,7 (1988), 887-895.