Experiences about Compound Dictionary on Computer Networks

Kyoji Umemura (NTT Basic Research Laboratories)
Akihiro Umemura (NTT Basic Research Laboratories)
Etsuko Suzuki (Tsuda College)

Suite 5-310A, 3-9-11, Midori-cho, Musashino, Tokyo 180, Japan
Email: umemura@nuesun.ntt.jp

ABSTRACT

This paper reports on the implementation, user interface, and experiences with the on-line dictionary system we developed. We call it *Avenue*. Avenue consists of several different kinds of dictionaries. Since it simultaneously searches all of its dictionaries for each query, the user does not need to specify which dictionary to search for the desired word. Japanese people usually consult several different dictionaries to look up a word. It is troublesome to find appropriate dictionary all the time. Avenue does not have this problem. Even if only one dictionary contains the word, the information will appear without fail.

1. INTRODUCTION

When Japanese people have trouble with an English word, they usually consult an English-Japanese dictionary, but often this is not enough. An English-English dictionary is needed to understand the nuance. When the concept expressed by an English word does not exist in proper Japanese, Japanese people have been creating a new Japanese word. English-Japanese dictionaries show these new Japanese words that are rarely used. They sound as if they might be still another foreign word. Thus, a Japanese-Japanese dictionary is also needed. Therefore, we sometimes need three dictionaries to understand English.

Many words have been imported into Japanese from Chinese, English, German, French, and so on. When a loanword is not in the Japanese-Japanese dictionary, we have to consult a foreign language dictionary. This happens often with technical words.

Japanese researchers always have several dictionaries on their desks. When they encounter an unknown word, they have trouble selecting the appropriate dictionary to check. Though there are many on-line dictionary systems, they also have the same problem. It is rather difficult or troublesome to use multiple dictionaries simultaneously. The problem is not limited to Japanese; English also has many kinds of dictionaries, such as field-specific dictionaries, thesauri, lexicons.

To make life simpler, we adopted a simple policy: "Combine all the dictionaries and always refer to all of them." We have developed an experimental on-line dictionary system based on this policy. We combined a Japanese-Japanese dictionary, an English-Japanese dictionary, an acronym dictionary, an information science dictionary, and our office telephone directory.

This paper is constructed as follows. Section 2 describes the implementation. Section 3 shows output examples. Section 4 describes the user interface. Section 5 describes how user behavior is recorded. Section 6 discusses the problems found from the analysys of access record. Section 7 discusses the importance of the record-keeping and cooperation with the publishers. Section 8 reports other problems derived from dictionary combination. Section 9 compares our system with other network information aggregations. Section 10 presents the conclusion.

2. IMPLEMENTATION OVERVIEW

The experimental on-line dictionary system implemented at NTT Research Laboratories is called Avenue. It is used daily by researchers. Most of them are Japanese. Avenue has one central server machine and many client machines. About 3000 researchers are able to access this system. More than 500 people of them have used the system, and it currently averages more than 100 requests per hour during the day. More than two hundred machines have been connected to the server. Ten to twenty machines usually connect to the server during the day. All of the information is stored in the central server.

Avenue consists of a Japanese-Japanese dictionary, an English-Japanese dictionary, an information science dictionary, a computer jargon dictionary, an acronym dictionary, and an office telephone directory. We are currently working on adding more dictionaries, such as Japanese-English dictionary, and an English-English dictionary, and a thesaurus.

All of the source information is converted into a uniform format. The server reads all of the dictionaries and then builds a combined index. Since this index is in memory, only a few disk operations are required to handle a request. This means that the server responds quickly.

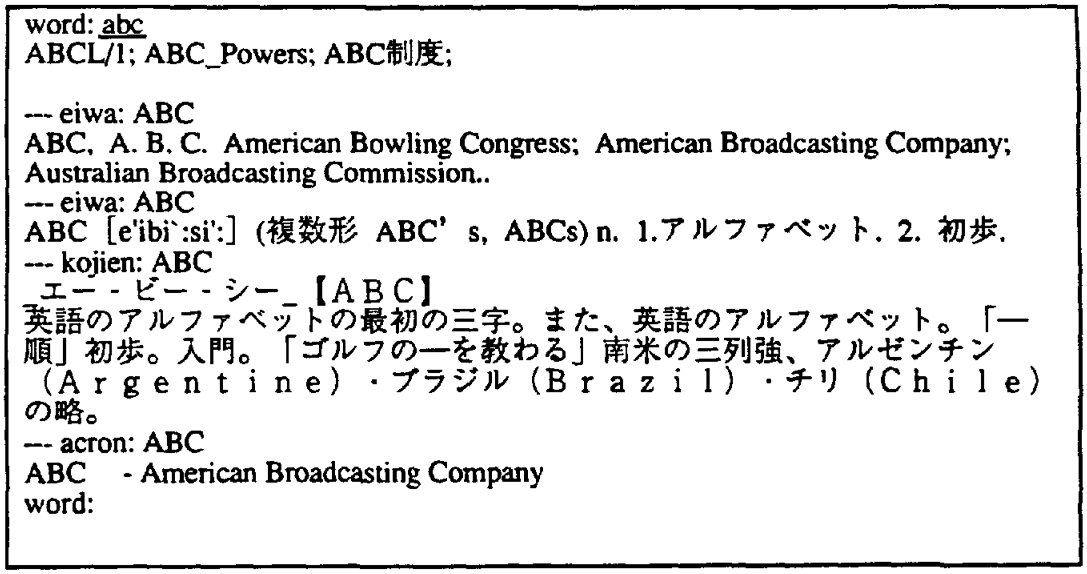
There are three ways to access the server: through remote shell command, through Emacs, and through HyperCard. UNIX users usually use the Emacs interface. Macintosh users usually use the HyperCard interface. These interfaces are built as extensions of the existing software.

The remote shell provides a simple command line interface. Though it does not provide sophisticated functionality, it does have one important characteristic. It does not require any installation. This feature is very important for gaining new users.

The Emacs interface is more sophisticated. By pressing one key, the word at the cursor is selected, a dedicated window appears, and the meanings appear. Mouse action or retyping is not required. It produces outlined text if the word exists in multiple dictionaries. Users can therefore quickly see which dictionaries contain the word. They can then explore the meanings in detail by using familiar Emacs commands. The HyperCard interface provides similar functions.

3. EXAMPLES

Avenue simultaneously presents information from various dictionaries. The example in Fig.1 shows the information provided for "abc." The underlined part is user input. The first list contains words that begin with ABC. "ABCL/1" is the name of a programming language and is an entry in the information science dictionary. "ABC_Powers" is an entry in the English-Japanese dictionary. "ABC制度" is an entry in the Japanese-Japanese dictionary. Japanese sometimes uses English letters for imported words.

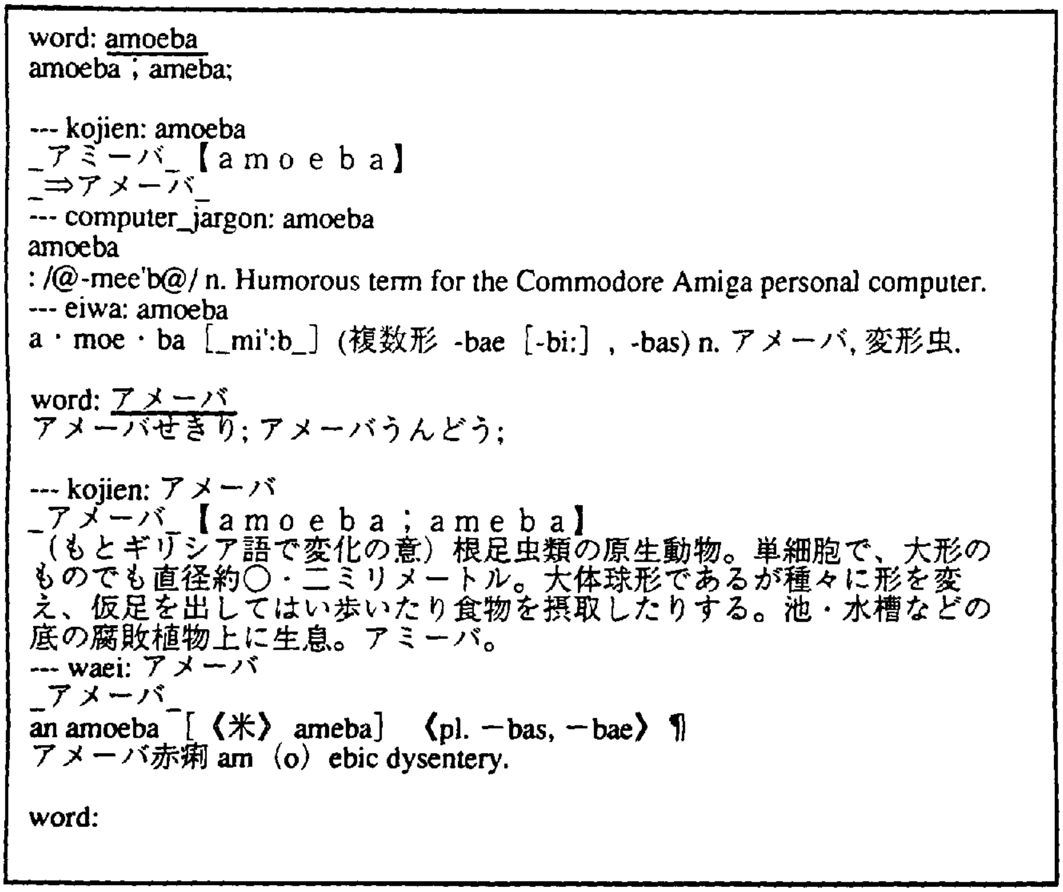


```
word: abc
ABCL/1; ABC_Powers; ABC制度;

--- eiwa: ABC
ABC, A. B. C. American Bowling Congress; American Broadcasting Company;
Australian Broadcasting Commission..
--- eiwa: ABC
ABC [e'ibi`:si':] (複数形 ABC' s, ABCs) n. 1.アルファベット. 2. 初歩.
--- kojien: ABC
 エー・ビー・シー_【ABC】
英語のアルファベットの最初の三字。また、英語のアルファベット。「―
順」初歩。入門。「ゴルフの―を教わる」南米の三列強、アルゼンチン
（Ａｒｇｅｎｔｉｎｅ）・ブラジル（Ｂｒａｚｉｌ）・チリ（Ｃｈｉｌｅ）
の略。
--- acron: ABC
ABC    - American Broadcasting Company
word:
```

Fig.1. Example for "abc".

In this example, three dictionaries have an entry for the word "ABC." They are English-Japanese (eiwa), Japanese-Japanese (kojien), and Acronym (acron) dictionaries. This entry is in the Japanese-Japanese dictionary because, as in English, Japanese people use it to denote "the first step". They also sometimes use it to denote Argentine, Brazil, and Chile. The latter description does not appear in English dictionaries.

Another example is shown in Fig. 2. This example shows an effect derived from a field specific dictionary. When we enter the word "amoeba," Avenue shows that it is also a computer jargon. This jargon dictionary has 1532 entrys of words. Though this number is relatively small, combination with other dictionaries is useful. It is useful to know that some words have specialized meanings. Since the English-Japanese dictionary and the special dictionary have the same interface, users can obtain various kinds of information in a uniform manner.

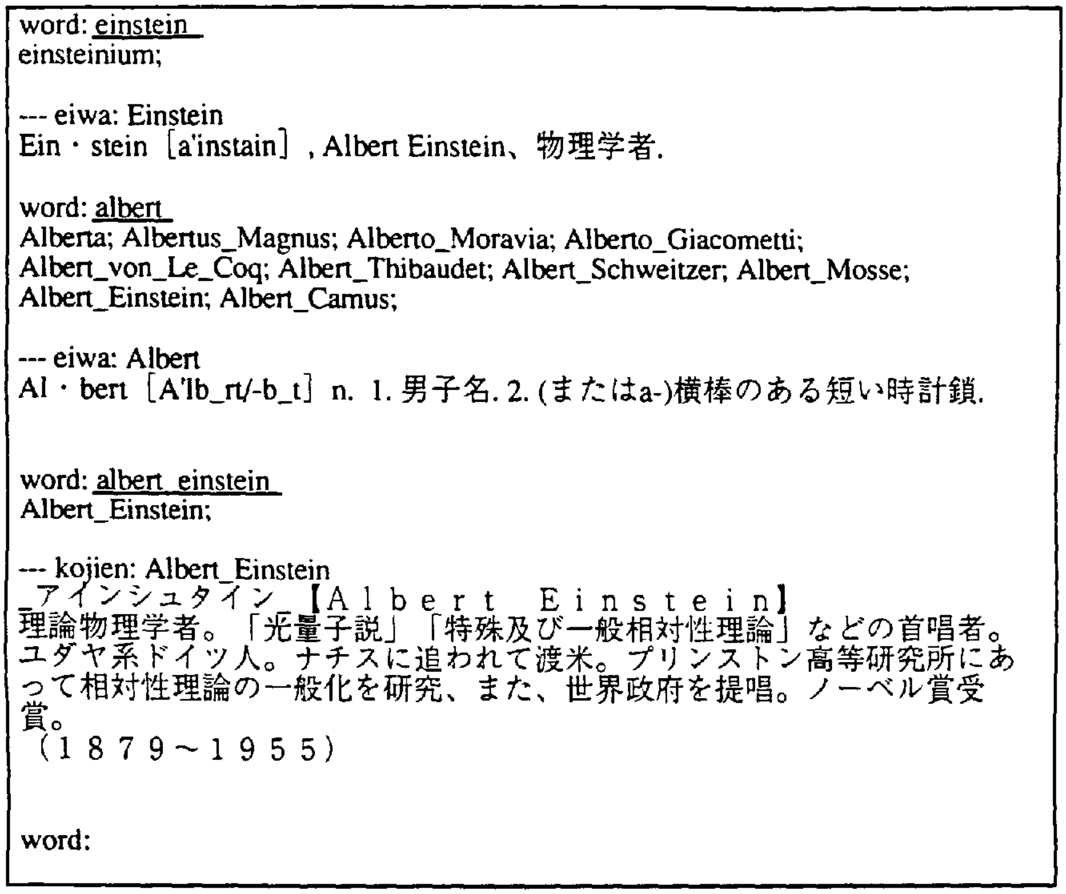


```
word: amoeba
amoeba ; ameba;

--- kojien: amoeba
_アミーバ_【a m o e b a】
_⇒アメーバ_
--- computer_jargon: amoeba
amoeba
: /@-mee'b@/ n. Humorous term for the Commodore Amiga personal computer.
--- eiwa: amoeba
a · moe · ba [_mi':b_] (複数形 -bae [-bi:] , -bas) n. アメーバ, 変形虫.

word: アメーバ
アメーバせきり; アメーバうんどう;

--- kojien: アメーバ
_アメーバ_【a m o e b a ; a m e b a】
（もとギリシア語で変化の意）根足虫類の原生動物。単細胞で、大形の
ものでも直径約〇・二ミリメートル。大体球形であるが種々に形を変
え、仮足を出してはい歩いたり食物を摂取したりする。池・水槽などの
底の腐敗植物上に生息。アミーバ。
--- waei: アメーバ
_アメーバ_
an amoeba [《米》 ameba] 《pl. －bas, －bae》 ¶
アメーバ赤痢 am（o）ebic dysentery.

word:
```

Fig. 2. Example for "amoeba"

The second word entry in Fig.2 is "アメーバ", which is how amoeba is written in Japanese. It appears in the English-Japanese dictionary (eiwa). It is a loanword and is read "ame:ba." Japanese understand that this word comes from foreign language since it is written in the character set used for loanwords. However, this word conveys no information about the small creature but the pronunciation. Since there are many loanwords in Japanese, we have to consult a Japanese-Japanese dictionary (kojien) to get the detailed information. The Japanese-Japanese dictionary shows that it has only one cell, and is less than 0.2 mm in size.

In the example shown in Fig.3, Avenue responds to the words "Albert," "Einstein," and "Albert Einstein" and presents information about Albert Einstein. The user gets information for both the given name and the family name. It is a result of the combination since one dictionary has only the word, "Albert" and another dictionary has only the word, "Einstein."



```
word: einstein
einsteinium;

--- eiwa: Einstein
Ein・stein [a'instain] , Albert Einstein、物理学者.

word: albert
Alberta; Albertus_Magnus; Alberto_Moravia; Alberto_Giacometti;
Albert_von_Le_Coq; Albert_Thibaudet; Albert_Schweitzer; Albert_Mosse;
Albert_Einstein; Albert_Camus;

--- eiwa: Albert
Al・bert [A'lb_rt/-b_t] n. 1. 男子名. 2. (またはa-)横棒のある短い時計鎖.


word: albert einstein
Albert_Einstein;

--- kojien: Albert_Einstein
_アインシュタイン 【Ａｌｂｅｒｔ　Ｅｉｎｓｔｅｉｎ】
理論物理学者。「光量子説」「特殊及び一般相対性理論」などの首唱者。
ユダヤ系ドイツ人。ナチスに追われて渡米。プリンストン高等研究所にあ
って相対性理論の一般化を研究、また、世界政府を提唱。ノーベル賞受
賞。
（１８７９〜１９５５）


word:
```

Fig.3. Example for Einstein, Albert and Albert Einstein.

Avenue is more likely to find a word than a single dictionary. Users find this to be very important; they seem to feel as if it contains complete information. They use Avenue even if they are not sure whether it contains the word or not.

4. USER INTERFACES

The user interface is a key element to gain users. It is difficult to determine what kind of interface is good for users. Some clear policies are necessary to design user interface. We attempted not to change the way users use the computer. We therefore use several existing systems for the user interface: the Rsh command, Emacs, and HyperCard.

The Rsh command is an existing UNIX command, as we have already explained. Since it is a standard network command, no installation is required and documentation is always available. A user only has to know the server's name to start using our system.

Furthermore, users can combine the Rsh command with other commands in the standard manner. Since information goes through standard input and standard output, users can easily write additional programs in order to format the output. If the service were to ask the user to logon to another computer, this additional programming would become more troublesome.

After using the "Rsh" command for a while, most users find retyping the word cumbersome and become annoyed with excessive output. When the output does not fit onto one screen, the user has to suspend output in order to read all of it. Using Emacs program and a HyperCard stack overcome these problems, that is, retyping and excessive output.

The Emacs program picks up the word at the cursor. At any time, one key stroke will initiate dictionary access for that word. The HyperCard stack picks up a selected region. In both cases there is no need for typing.

The Emacs program displays the information in outline mode. If the output is long, the detailed information is hidden by the interface program. Users can explore the hidden parts after they scan all of the headings. The HyperCard stack has a dictionary preference list and cursor movement buttons. Users can arrange the order of the dictionaries and even ignore some of the dictionaries. They can also go backward and forward, dictionary by dictionary by using a dedicated button on the stack.

5. RECORDING USER BEHAVIOR

It is important for an information system to record each user's behavior; who uses what. Dedicated interface programs usually solve this problem. However, they introduce another problem: installation. Our observations show that users tend to continue using printed dictionaries if software installation is required at the user's site. A special trick is needed to record user names and their requests when users will not install related software.

UNIX has a standard command called "Rsh" or "Remsh." It executes commands at a remote machine. This command sends the user's name when it requests a job to be run on another machine. Rsh's protocol is designed so that the regular user cannot disguise himself as another user, even if he builds his own network programs. The problem with "Rsh" is that it requires strict registration in order to ensure system security. If a new user should register himself before using Avenue, he would refrain from using Avenue.

The problem of installation and registration was solved by creating a modified server. After our modification, it responds to everyone, but limits the commands that can be run. Since "Rsh" provides user identification, it is easy for the modified server to record who uses what. There is no installation or modification at the user site. Only the central server has a special program.

From the user's point of view, new machines and new users can access the dictionaries without registering by using this method. The only thing a user has to know is the name of the server machine. From the operator's point of view, he will have a record of user behavior without installation or registration.

6. PROBLEMS FOUND FROM ACCESS RECORD

We analyzed Avenue's access record in order to find various problems. We assume that a user has encountered a problem when he makes several requests within a short time period, that is, several minutes. We therefore picked up those places where a user repeatedly accessed Avenue. We then entered the same words so that we could see what the user actually got. We thus identified five common problems.

Problem (1): The user needs a variation of the given word.
If the word is a headword, it will be in the candidate list from Avenue. If it is not a headword, he must guess the spelling. Sometimes he enters Japanese word to get some hints.

Problem (2): The user needs an idiom.
Idioms usually appear among the definitions, not as headwords. It is difficult to find the correct headword for a given idiom. Furthermore, the dictionary may be inconsistent. For example, " ~to" may be used in one place, while "~ to," which has a space, is used in another place.

Problem (3): The user needs an example.
After finding an English word in the Japanese-English dictionary, a user frequently consults the English-English dictionary to get an example. When the entry does not contain an example sentence, he sometimes starts entering relatively simple words, hoping to find some examples. When this word is a relatively rare word, this search for examples happens more frequently.

Problem (4): The user cannot enter the character
Japanese characters are hard to read and harder to enter since Japanese uses thousands of

Chinese characters (Kanji) and many other characters. It often happens that a user can understand the meaning of a character, but cannot pronounce it. Unless the character can be pronounced, it is very hard to input the character into the computer. Users sometimes enter words that have some relation in meaning in order to obtain the character. Once it is obtained, he enters the desired word using cut and paste.

Problem (5): The user is not sure of the correct spelling of the word
When the spelling is uncertain, the user will often enter words that have similar spelling. If the correct one is not found, Japanese words are often entered.

Problems (1), (2), and (3) indicated the need for additional dictionaries: a thesaurus, an idiom dictionary, and a corpus of English. It is important that this fact comes from the actual record of usage.

Problem (4) reflects a problem in handling Japanese. Currently, Japanese characters are converted from pronunciation to characters when they are input to a computer. If the character cannot be pronounced, it is very hard to enter the character. Though this is an apparent problem, we had failed to recognize it. This is because we have taken it for granted unconsciously. Though this problem is not specific to Avenue, it is important to know that our user actually have this problem.

Problem (5) means that users sometimes fail to specify what they want to know. This is a common problem in information retrieval. Though we do not have a good idea for overcoming it, we can recognize it based on actual usage.

Several problems have been identified by focusing on repetitive access from one user. It is important for us to be aware of the problems so that we can improve the system. Some problems are due to the lack of certain dictionaries. We have thus identified a specific improvement that needs to be made.

7. COOPERATION WITH PUBLISHERS

Cooperation with publishers is essential in operating network dictionary systems. Since these systems and printed dictionaries are in a competitive relation, cooperation is a rather subtle issue. Luckily, publishers are searching for new ways of publishing. For example, they are initiating CD-ROM publication. Network systems are another future publication form.

The recording mechanism is a key to making cooperation possible. With it, the users can be identified, along with their number of uses. It also provides valuable information to publishers to revise their dictionary. For example, the record shows which words may be candidates for addition. Publishers have agreed to provide their information to us in return for a fee and a complete record of user activity.

8. OTHER PROBLEMS AND FUTURE WORK

One problem for future work is that, though many entries may appear for one word, each may have different format. This sometimes makes the information hard to read. Since the information originally comes from printed dictionaries, there is some variance in format. It is rather difficult to reformat all of them. Although the Avenue interface has a mechanism to add a formatting program for each dictionary, it is troublesome to write such a program for all dictionaries. Furthermore, it is hard to write program that will produce a clean and neat format.

Another problem is that too much information may sometimes be given for one word. If it does not fit one screen, it is difficult to find the needed information. Although the interface has outline control and cursor movement control, which reduce the trouble, it will become a more severe problem as the number of information sources increases.

Dictionary preference is another technical issue. Users will prefer different sets of dictionaries, depending on their speciality. Computer engineers and linguists consult different dictionaries.

Furthermore, A person's preference may change over time as their interests change; He may be computer engineer one time and linguist another time. Avenue currently provides only one-dimensional list of dictionaries. If there are many information sources, a one-dimensional list may be too limited for many users. A more flexible and powerful mechanism is needed to specify the relations among dictionaries.

9. RELATED WORK

Many information systems have become available. WAIS, Gopher, and WWW are notable [1] among them. They also combine various information sources and they handle not only dictionary, but also may other kinds of information. While Avenue calls up all sources simultaneously, they call up one at a time. Although they have a single dedicated keyword search server, users still have to select one among sources to get the information. Generally speaking, selecting information sources is difficult and time consuming.

There are also many on-line dictionaries for personal computers. The variety of information and uniform interface are the main advantages of Avenue.

10. CONCLUSION

We have implemented a combined dictionary system that combines a Japanese-Japanese dictionary, English-Japanese dictionary, information science dictionary, an acronym dictionary, and a telephone directory. They are all consulted simultaneously when responding to a query. We found that this feature is very important for users. The recording of user behavior shows the need for additional dictionaries. This record is also valuable to the dictionary publishers.

ACKNOWLEDGMENTS

We thank Chisato Furuya at the Nagaoka University of Technology for her valuable suggestions and discussion about the value of this work. We also thank Yasuki Saito at the NTT Basic Research Laboratories for his guidance and support.

[1] Ed Krol, "The Whole Internet User's Guide & Catalog", ISBN 1-56592-025-2, O'Reilly.