IBM Research Report

Globenet and RAISE:
Intelligent Agents
for Networked Newsgroups
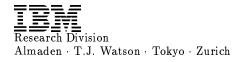### and Customer Service Support

Benjamin N. Grosof, Davis A. Foulger

IBM Research Division
T.J. Watson Research Center
P.O. Box 704, Yorktown Heights, NY 10598
(914) 784-7100 main
Internet: {grosof,folger}@watson.ibm.com
(alt.: grosof@cs.stanford.edu)
WWW: http://www.research.ibm.com

IBM

Research Division
Almaden · T.J. Watson · Tokyo · Zurich

Abstract and Summary

The **Globenet** system, developed at IBM T.J. Watson Research Center, performs knowledge-based retrieval and handling of newsgroup-like information from heterogeneous WAN sources. Users specify personal, rule-based, intelligent agents that control the retrieval and handling done on their behalves.

The work-in-progress version[1] of Globenet is enhanced by piloting the **RAISE** class library, also developed at IBM Watson. RAISE stands for Reusable Agent Intelligence Software Environment. RAISE enables powerful rules and flexible inferencing, as well as smoothly integrated and extensible control of additional actions and tests.

Reasoning is performed about both structured and unstructured attributes of newsgroup items/mail-messages. Structured attributes include those available from the message headers, e.g., author and subject. Unstructured attributes include those available from message bodies as well, e.g., free-text conditions of the kind used in classical information retrieval, such as boolean expressions in keywords and phrases.

Retrieval/filtering is distributed in space and time: a first, foraging phase at remote sites is followed by a second local phase. Handling functions are also interleaved in this second phase, including categorization, attentional prioritization, visual presentation, selective forwarding and dissemination, and personal information management.

Globenet is currently deployed in a customer service support application within IBM to aid human staff in their answering of customer questions and problems. Globenet includes features distinctive to customer service support: it automates identification of questions and supports humans' collaboration in the process of query-answering. Major productivity improvements of over 30% are reported in early field experience with the application.

In addition, the approach embodied by the Globenet+RAISE system is applicable to a wide range of other applications in which information has a similar combination of structured and unstructured attributes, e.g., retrieval and handling of e-mail or textual articles, including customized electronic news services.

---

[1]of which a prototype was first publicly demonstrated on June 22, 1995 at the IBM T.J. Watson Research Center's Computer Science Expo

Problems: Customer Service Support using Network Newsgroups

Organization of this section: we describe three concentric problems, each in turn being more narrowly focused.

Problem: Network Newsgroup Filtering

Electronic bulletin boards / asynchronous computer conferencing forums / network newsgroups (we will use the terms interchangeably below) pose several problems for navigation and retrieval. First, of course, is the sheer **volume** of discussion information: not only large but rapidly increasing. Second is the unintegrated **heterogeneity** of user interfaces, service platforms, and programmer interfaces. Third is the difficulty in mapping a user's search **topic** of interest to the rather simple and rigid topic indices, typically a fixed-definition hierarchical pathname scheme, and the rather impoverished search capabilities, that are provided by newsgroup systems. While some newsgroup systems support somewhat richer search capabilities (e.g., by author, date, or keyword), others lack them. There is a need for tools that provide a high degree of retrieval functionality that is common and integrating across the heterogeneity.

Problem: Customer Service Support using Network Newsgroups

Electronic Support is Cheaper: The latest major innovation in customer service support is found in electronic network newsgroups. Companies, including IBM, have found that the personnel costs associated with supporting a given number of customers electronically is a quarter the cost of supporting the same number of customers via the telephone.

Large Volume on Multiple, Heterogeneous Newsgroup Sources and Venues: The problems of network newsgroup filtering are particularly acute for the many companies like IBM that try to support customers in the on-line and network environments. Many companies, including IBM, and on-line services provide newsgroup forums that are specifically oriented towards customer service support, including customer questions. Nevertheless, customer questions get posted "all over the place" on newsgroups, not just where they are supposed to be. Customers ask questions about IBM products in a range of venues, including CompuServe, Prodigy, and America OnLine forums, Internet Newsgroups, the local bulletin boards of Fidonet, and company maintained systems like IBM's Talklink and PCCBBS (PC Company Bulletin

Board Service). Each of these venues produces a large number of customer questions about IBM products like OS/2, and the problems of finding and answering all these questions is daunting, particularly given the diverse range of user interfaces that must be negotiated to deal with them all.

Economic Importance: Customer service support is a major cost of business in the overall economy, especially in high-tech, and particularly for computer software and hardware. Today, it is highly intensive in skilled labor. Moreover, the quality, availability, cost in money and time, and convenience of customer service support is a major criterion for many customers in selecting vendors, especially for software.

Driving Problem for General Newsgroup / Information Retrieval: Customer service support is interesting as a driver application for the more general areas of navigation and retrieval and collaborative work. It is an intensive version of how people generally employ newsgroups. Attentional productivity is directly of economic value in the business setting: i.e., the staffers' time is money. More indirectly, the customers' time and convenience also are money. Customer service support personnel are especially goal-oriented and focussed; their job is to do query-answering. Better search capabilities and topic specifications are especially important for them. Customer service support is **collaborative** in two senses. First, staffers collaborate as an interdependent group to divide up the workflow of customer questions to answer and bulletin boards to monitor, according to the staffers' areas of expertise. Each question "case" needs to be managed: assigned, tracked, and closed out when answered. Second, the whole newsgroup participant population collaborates in a looser but nevertheless very powerful manner. Other customers often contribute answers to current or previous customer questions. Reading previously posted items often helps customers prevent having to ask questions, or at least helps them to better formulate their questions. Thus **dissemination** of answers and other notifications, to other staffers and to overall participants/customers directed but not necessarily authored by the customer service staffers, is important. A "prevent defense" is the most effective: the *un*asked question is the least expensive to respond to.

Newsgroup Problems Particular to Customer Service Support

While a great deal of attention has been given to the problem of making electronic bulletin board services productive for end users, little attention has been given to the special productivity requirements of electronic customer service support providers. Exceptions are some additional tools that allow system operators to delete and redirect customer postings.

Next, we list problems associated with previously available newsgroup tools designed for ordinary individual end users (for the rest of this section, we will call such end users "customer-users"), when those tools are applied to electronic support.

Desire to avoid huge browsing overhead from searching manually to identify questions: In measurements of electronic support costs in IBM, it was found that it took, on average, roughly 1 hour (average) to answer each question posted to electronic bulletin boards. Only half the time (30 minutes) was spent actually answering the question. Another 25 minutes was spent browsing the bulletin board manually looking for questions to answer! No attempt is made, in current on-line services, to identify questions as such in general newsgroups. The lack of such identification is the principal reason why support staffers had to browse every posting to relevant forums. It also makes it difficult to track the status of those questions and identify which questions have gone unanswered.

Desire for natural language content analysis as basis for search functions: Electronic support staffers desire more detailed content assessment based on keywords and phrases rather than just the subject line or author line information which customer-user tools typically focus on.

Desire to support the collaborative process of assignment: The collaborative process among staffers for managing the assignment of cases is left unaddressed by customer-user tools. Also, there is no provision to prevent duplicate effort: two support people could be sitting in adjoining cubicles browsing the same forum and never know that they both selected the same question to answer.

Desire to preserve accumulated knowledge; archiving and indexing: Existing on-line services generally age (i.e., throw away old) postings quickly (sometimes in as little as a few days). Also, often existing customer-user oriented user interfaces to bulletin boards make no attempt to preserve the postings an individual makes in a manner organized to index retrieval by that individual. The net result, in general, is that it is difficult for providers of electronic support to maintain an archive of their existing answers and cannot (easily) reuse existing answers to answer new questions.

Other problems less directly related to the opportunity for knowledge-based techniques include:

- **Desire for common user interface** across multiple on-line services: each service has a different interface.

- **Desire for measurements** of the effectiveness of electronic support, to aid tracking and management.

- **Desire to cut overhead from high frequency of accesses**: electronic support staffers access more frequently and continuously than ordinary customer-users; logon/logoff of BBS'es was found in the IBM measurements mentioned above to be a non-trivial overhead, occupying 5 minutes of the 1 hour time per question.

- **Desire to disseminate to multiple bulletin board services at once**: It is desirable to integrate across multiple bulletin board services for posting/appending/disseminating, as well as for reading/browsing/retrieving.

Solution: The Globenet System and the RAISE Class Library

Globenet is a system, developed by our group at IBM T.J. Watson Research Center, for retrieval and handling of newsgroup items. Its pilot application has been to assist customer service support personnel within IBM. It is currently deployed (since fall 1993) in the IBM OS/2 customer service support organization and has several dozen field users there. In addition, there is now a significant number of other users in a variety of other parts of IBM.

The new (work-in-progress) version of Globenet is enhanced by piloting the RAISE class library, also developed by our group. "RAISE" stands for Reusable Agent Intelligence Software Environment. RAISE enables powerful rules and flexible inferencing, as well as smoothly integrated and extensible control of additional actions and tests.

A running prototype of the new version (Globenet+RAISE) was first demonstrated publicly on June 22, 1995 at the IBM T.J. Watson Research Center's Computer Science Expo.

Below, we describe both the currently deployed version and the work-in-progress version. When our descriptions apply to both versions, we will simply speak of "Globenet". When specifically describing the work-in-progress version, we will speak of "Globenet+RAISE".

Overall Operation

For each user's instance of it, the overall Globenet system currently operates as follows.

Appends (newsgroup items/mail-messages) are collected from a set of heterogeneous remote WAN information sources. Currently, these sources are IBM company-maintained systems like IBM's Talklink and PCCBBS (PC Company Bulletin Board Service). In the work-in-progress version, these source venues for newsgroup information are being extended to also include Internet Newsgroups, CompuServe, Prodigy, and America OnLine forums, and the local bulletin boards of Fidonet.

A set of gateways interface the rest of the system to these sources and their associated servers. In the currently deployed version, the gateways run on servers. The server platforms are: OS/2 operating system, IBM PS/2 model 90 or 95 (486 family) processor. For the retrieval task, these gateways are used in the ingoing direction from sources to user client. They are also used in the outgoing direction to disseminate information, e.g., to post customer service support answers to the various remote newsgroup systems.

On each given source, only a subset of the appends available are selected for collection. This constitutes a first **foraging** phase of search / retrieval / filtering.

The collected appends are put into a common format and transmitted to a local client device.

On the local client, a second phase of search / retrieval / filtering is performed. In addition, other handling functions are performed.

The user interfaces with the system, e.g., accesses the retrieved information, via the local client. This interface includes ForBrowse, a browsing and presentation component, as well as ForAgent, an intelligent agent component.

In the currently deployed version, the platforms that ForAgent and ForBrowse run on are: OS/2 operating system, IBM-compatible PC, 386 or better (i.e., higher X86 series) processor.

Knowledge-based Aspects

Next, we give more details about the knowledge-based aspects of Globenet.

Each end user specifies a personal, rule-based, intelligent agent which controls the retrieval and handling done on his or her behalf. More precisely, what the user specifies is a knowledge base of rules and facts. An example rule is:

"If the Source is 'Compuserve', and the Newsgroup name is 'IBMPC/486/Software', and a contained Keyword is 'mobile' or 'PDA', then place in Folder 'me/software/home-machines/laptop'.

An inference engine reasons from this knowledge base to determine the details of which information to retrieve from where, and what to do with it once retrieved. A log (trace) of the agent's activity is kept to explain to the user for each append which rules led to its retrieval and handling.

The rules specify:

- which information **sources** to forage remotely
- which appends to **select** to keep for presentation to the user or to employ otherwise
- how to **categorize** these appends, e.g., for storage in the user's hierarchical folder / directory structure for **personal information management**
- how to **prioritize** the information for the user's **attention** when presented via **visual presentation** interface features such as color and spatial (e.g., sequential) positioning of the appends
- to whom selected appends will be **forwarded / disseminated**.

Structured and Unstructured Information and Attributes

In the area of information retrieval generally, and knowledge-based retrieval of networked information more particularly, an important distinction is often made in the research community[2] between **structured** information such as relational databases, which come equipped with a schema that describes the data's structure in a relatively deep way, versus **unstructured** information such as free text or video, for which schematization and deep indexing are difficult and

---

[2]for example, in the discussions and proceedings of the 1994 AAAI Spring Symposium on Software Agents and the 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments. E.g., (Kirk *et al.* 1995) emphasizes this distinction.

must often be constructed or learned. Many implemented systems for information retrieval or navigation, and their associated techniques, are oriented towards either only structured or only unstructured information.

Reasoning in Globenet is performed about *both* structured and unstructured attributes of newsgroup items/mail-messages. These attributes include information available by directly processing the incoming append, as well as derived and user-defined attributes inferred via the rules.

Structured attributes available by processing directly (usually by simple reformatting and extraction) the incoming append include, among others:

- the name of the newsgroup information source, e.g., "Compuserve"

- the name of the newsgroup within the source context, e.g., "IBMPC/486/software"

- properties from the items' headers, e.g.:

  - author ("from")
  - subject
  - date/time
  - reference line (i.e., if an append refers/replies to another previous append)

Unstructured attributes include, among others, properties of the items' textual bodies as well as those in their headers, e.g.:

- free-text NLP: conditions obtained by relatively simple natural language processing (NLP) techniques of the kind used in classical information retrieval. In the currently deployed version, such a condition simply specifies whether the append contains a particular boolean expression formed from **keywords** and phrases.

- **question identification**: whether the append contains a question (e.g., a non-rhetorical question that needs to be answered by a customer service support staffer). This is inferred heuristically by a special procedure.

Next, we describe the step by step operation of retrieval and handling.

To control the foraging phase, the total rule set is first analyzed and reformulated to generate a set of foraging queries. These search / retrieval queries are then issued to the remote newsgroup information sources.

The above pre-foraging analysis and reformulation is performed because of capability and resource constraints involved in remote querying. First, only limited kinds of query capabilities are provided in the remote newsgroup systems. Today, as lowest common denominator across such systems, one can query to retrieve all appends in a specifed (named) newsgroup that have been posted after a specified timestamp or append-identification-number. However, usually one can not make much more complex queries than this. Thus, only part of the information retrieval filtering specified by the users' rules can take place remotely.

Second, efficient use of resources is important during querying of the remote sources. For each source, it is typically preferable in terms of efficiency to query for a collection of newsgroups (on that source) all at once than to issue several different queries at different times. And it is preferable to avoid duplicate queries, e.g., for the same newsgroup with different overlapping timestamp periods.

Third, efficient use of resources is important in terms of WAN bandwidth since it is costly in terms of money and delay. This also militates towards one newsgroup-collection query. It also militates towards filtering tightly, insofar as this is possible, remotely. One would like to reduce the amount of foraged information that needs to be transported over the WAN back to the local client.

In the currently deployed version of Globenet, the pre-foraging analysis generates a list of selected (i.e., interesting) information sources, and for each source, a list of selected newsgroups and timestamp (or append-id) intervals. To forage, each source is then queried. Foraging is currently performed by polling at regular periods: each selected source is queried roughly every 20–60 minutes.

The result of foraging is a stream (often in batches) of incoming appends that arrives at the local client.

After foraging comes the second phase of processing, including further filtering and handling. This phase is controlled more directly by the user's knowledge base of rules and facts. Rule-based inferencing is performed, primarily in the forward direction (i.e., data-driven, as opposed to goal-directed), in a manner roughly similar to a production system. That is, incoming appends trigger the testing and firing of the rules. However, there are some interesting departures from the primarily forward direction of processing.

Goal-directed special condition analyses, e.g., natural language processing of the textual bodies, are invoked during the testing of rules. This NLP would be computationally too expensive to perform in a non-goal-directed manner. E.g., it is impractical to test every incoming append for every boolean combination of keywords appearing in the total rule set's rule antecedents. The goal-directed special condition analyses are interleaved with the forward-direction inferencing by procedurally attaching these analyses to logical conditions in the rule antecedents.

Actions with side effects, e.g., forwarding, also are interleaved with the forward-direction inferencing. These actions are invoked by procedural attachment to logical conditions in the rule consequents. In the currently deployed version, actions include:

- folder: where to keep the retrieved append in the user's personal information management database; e.g., with what pathname in the user's hierarchical file system

- color: the color with which to display each retrieved append for visual presentation. This color is used to indicate attentional prioritization: e.g., red for new and important.

- **assignment**: to whom (and whether) the append

has been assigned for question answering, in the collaborative process of the customer service support application task.

Enhancements in the New Version using RAISE

Next, we describe how the currently in-development version of the intelligent agent aspect of Globenet differs from and enhances that in the currently deployed version.

The deployed version of Globenet has only relatively limited rule-based inferencing and representation capabilities (e.g., no chaining), and controls only a handful of actions and special condition analyses.

The in-development version (e.g., the currently running public demonstration prototype) is built using RAISE. (Recall, "RAISE" is mnemonic for Reusable Agent Intelligence Software Environment.) RAISE is an object-oriented C++ class library of agent intelligence capabilities for reasoning and communication. It is being developed by our group.

Using RAISE has a number of architectural and functional advantages. First, the reasoning components are more powerful and cleaner overall: standard knowledge representation and inferencing techniques are employed. This enables chaining of inferences. And it enables predicates with multiple arguments. The deployed version lacks chaining and such predicates. Also, it improves extensibility, reusability of code and knowledge bases for other applications, and the ability to communicate with other intelligent agents, e.g., other users' instances of ForAgent (Globenet agents).

Second, the rule-based reasoning is more easily integrated with other software components: notably, with the attached procedures for special condition analyses and for actions. RAISE includes simple yet powerful mechanisms for procedural attachment. This facilitates using the rules to control many more special condition analyses and actions than in the deployed version.

The in-development version of Globenet includes several additional very useful special condition analyses and actions, e.g.:

- special condition analysis: more complex free text NLP than just keywords and phrases

- action: dissemination / forwarding of appends, e.g., to other people

- action: spatial position (sequential ordering) with which to display each post-filtering append for visual presentation

Field Experience

In the deployed application: the IBM OS/2 customer service support organization wrote their own rules. Currently, there are dozens of users. The number of users is anticipated to grow to perhaps hundreds later this year. The number of rules per user is relatively modest: dozens at most.

An example rule schema useful in the customer service support application is:

"If an append is from one of a list of newsgroups [i.e., list of pairs $\langle source, newsgroupname \rangle$] that I/we monitor
and it contains a question
and it's unassigned
and it contains a cnf [conjunctive normal form] expression in keywords/phrase strings, e.g.: ('OS/2' or 'AIX') and ("bug" or 'problem') [representing a topic of my expertise or responsibility]
then keep it in folder Me-To-Answer [i.e., I'm interested because it's up my alley as a staffer]."

Major productivity gains of over 30% have been reported for the electronic customer service support task in the IBM OS/2 organization, as the result of using Globenet. Primarily, this is manifested in the lower amount of staffer **labor** time required on average to answer each customer question.

Discussion and Future Work

Globenet has a number of novel and interesting features:

- the fielded customer service support application of newsgroup information retrieval and handling; and, especially in this context:

- the division of retrieval into remote foraging and local further filtering

- the mix of handling with retrieval, controlled by a common knowledge base of rules

- the reformulation of the rule set to make remote foraging queries

- the mix of structured and unstructured information / attributes; in particular, the mix of NLP free text information retrieval techniques with rule-based representation and inferencing

We are pursuing several directions for future work. These include:

- How to ease the burden of **authoring** of rules, especially by "lay" end users who are not programmers. This is critical to the practicality of applying Globenet and RAISE widely to the more general generic task of newsgroup filtering. Our approach to end user authoring has several different elements:

  1. organizing schematized rules by groups into libraries. A user can then select, merge, and parametrize to create a personal rule set.

  2. GUI aspects of the rule authoring, including menu pull down and drop-and-drag indication of logical flow within a rule antecedent

  3. common sense style for rule specification, including natural ways to handle exceptions and conflict via techniques in default reasoning

- To explore relatively similar **other applications** (for the Globenet and RAISE techniques) in which retrieved information has a similar combination of structured and unstructured attributes, e.g., retrieval and handling of e-mail or textual articles, including customized electronic news services.

- How to incorporate probabilistic-flavor reasoning (including fuzzy and expected-utility decision analysis) and inductive learning techniques, including to more powerfully incorporate content analysis techniques for free text NLP and multimedia. These content analysis techniques typically yield numerically-flavored scores rather than yes/no cf. logical rules.

- How to exploit richer remote capabilities for search / filtering as they become available on the sources, especially: more complex queries and/or more subscription-style initiative by the remote source (as opposed to polling).

- More generally, how to tradeoff between remote versus local search effort and smarts, and how to appropriately distribute processing, e.g., to exploit commonalities among a local *group* of Globenet users.

RAISE is especially important to our pursuit of these directions.

Related Work

John S. Breese from Microsoft Research (Redmond, WA) demonstrated a prototype information retrieval system for assistance of customer service support staffers at a AAAI-94 plenary session.[3] The system retrieves previous questions and answers (textual items) by reasoning probabilistically. Each retrieved item is thus assigned a probabilistic score which indicates closeness or relevancy to a given question item. They reported that it had been fielded successfully in a part of Microsoft's customer service support organization.

Computer conferencing on newsgroups and IBM's use of it are discussed more in (Foulger 1991).

The 1994 AAAI Spring Symposium on Software Agents and the 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments included a number of papers on the use of intelligent agents for information retrieval, but none specific to customer service support applications nor even to newsgroup information. The emphases of the work there are distinctly different than in our work on Globenet. Globenet does *not* form / operationalize a topic / query by learning what is interesting, e.g., from feedback or from a community's experience (Maes 1994) (Armstrong *et al.* 1995) (Balabanovic and Shoham 1995) (Krulwich 1995). Nor does Globenet+RAISE utilize meta-information about sources, e.g., to select relevant sources or to generate a query plan (Knoblock and Arens 1994) (Knoblock 1995) (Levy *et al.* 1994) (Kirk *et al.* 1995). Nor does Globenet use a script or plan overall (Voorhees 1994) (Etzioni *et al.* 1994).

---

[3] The demonstration was part of the session with Steve Ballmer of Microsoft as main speaker, at the National Conference on Artificial Intelligence, held July 31–Aug. 4, 1994, Seattle, WA; sponsored by the American Association for Artificial Intelligence, Menlo Park, CA; WWW http://www.aaai.org . See WWW http://www.research.microsoft.com/dtg/ for more about Breese *et al*'s related work.

Rather, Globenet takes / assumes as a given: the specification of what are, in effect, operational topics. Globenet then uses rule-based handling to generate actions, similar to a production system, rather than any more complicated kind of planning. Globenet does not address the issues of deep semantics of sources and queries. Overall, Globenet does provide a relatively convenient, simple way to instruct an agent and have it do information retrieval and handling tasks that appear to be quite useful in its areas of application.

In the future, Globenet's approach might be extended, however, to exploit some of the other techniques above, especially learning and source selection, and especially to aid the more general task of newsgroup filtering.

Summary of Paper: See the Abstract.

References

Robert Armstrong, Dayne Freitag, Thorsten Joachims, and Tom Mitchell. Webwatcher: A learning apprentice for the world wide web. In Craig Knoblock and Alon Levy, editors, *Working Notes of the 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Menlo Park, CA; http://www.aaai.org, 1995. American Association for Artificial Intelligence. Working Notes to be available as a AAAI Technical Report.

Marko Balabanovic and Yoav Shoham. Learning information retrieval agents: Experiments with automated web browsing. In Craig Knoblock and Alon Levy, editors, *Working Notes of the 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Menlo Park, CA; http://www.aaai.org, 1995. American Association for Artificial Intelligence. Working Notes to be available as a AAAI Technical Report.

Oren Etzioni, Neal Lesh, and Richard Segal. Building softbots for unix (preliminary report). In Oren Etzioni, editor, *Working Notes of the 1994 AAAI Spring Symposium on Software Agents*, Menlo Park, CA; http://www.aaai.org, 1994. American Association for Artificial Intelligence. Working Notes are available as a AAAI Technical Report.

Davis A. Foulger. *Medium as Process: The structure, use and practice of computer conferencing on IBM's IBMPC computer conferencing facility*. PhD

thesis, Dept. of Communications, Temple University, Philadelphia, PA, 1991. Published by University Microfilms, Inc., Ann Arbor, MI.

Thomas Kirk, Alon Y. Levy, Yehoshua Sagiv, and Divesh Srivastava. The information manifold. In Craig Knoblock and Alon Levy, editors, *Working Notes of the 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Menlo Park, CA; http://www.aaai.org, 1995. American Association for Artificial Intelligence. Working Notes to be available as a AAAI Technical Report.

Craig Knoblock and Yigal Arens. An architecture for information retrieval agents. In Oren Etzioni, editor, *Working Notes of the 1994 AAAI Spring Symposium on Software Agents*, Menlo Park, CA; http://www.aaai.org, 1994. American Association for Artificial Intelligence. Working Notes are available as a AAAI Technical Report.

Craig Knoblock. Integrating planning and execution for information gathering. In Craig Knoblock and Alon Levy, editors, *Working Notes of the 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Menlo Park, CA; http://www.aaai.org, 1995. American Association for Artificial Intelligence. Working Notes to be available as a AAAI Technical Report.

Bruce Krulwich. Learning user interests across heterogeneous document databases. In Craig Knoblock and Alon Levy, editors, *Working Notes of the 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Menlo Park, CA; http://www.aaai.org, 1995. American Association for Artificial Intelligence. Working Notes to be available as a AAAI Technical Report.

Alon Y. Levy, Yehoshua Sagiv, and Divesh Srivastava. Towards efficient information gathering agents. In Oren Etzioni, editor, *Working Notes of the 1994 AAAI Spring Symposium on Software Agents*, Menlo Park, CA; http://www.aaai.org, 1994. American Association for Artificial Intelligence. Working Notes are available as a AAAI Technical Report.

Pattie Maes. Social interface agents: Acquiring competence by learning from users and other agents. In Oren Etzioni, editor, *Working Notes of the 1994 AAAI Spring Symposium on Software Agents*, Menlo Park, CA; http://www.aaai.org, 1994. American Association for Artificial Intelligence. Working Notes are available as a AAAI Technical Report.

Ellen Voorhees. Software agents for information retrieval. In Oren Etzioni, editor, *Working Notes of the 1994 AAAI Spring Symposium on Software Agents*, Menlo Park, CA; http://www.aaai.org, 1994. American Association for Artificial Intelligence. Working Notes are available as a AAAI Technical Report.