OLC: An On-Line Consulting System for UNIX†

Thomas J. Coppeto
Beth L. Anderson
Daniel E. Geer, Jr.

Project Athena
Massachusetts Institute of Technology
Cambridge, MA 02139
{tjcoppet,beth,geer}@ATHENA.MIT.EDU

G. Winfield Treese

Cambridge Research Laboratory
Digital Equipment Corporation
Cambridge, MA 02139
treese@CRL.DEC.COM

ABSTRACT

Helping users learn the intricacies of UNIX, particularly in a custom environment, is always a challenge. Helping thousands of users in an environment that is distributed both geographically and computationally is especially difficult. Project Athena has developed an ‘‘On-Line Consulting’’ system (OLC) that enables users to ask questions of consultants located ‘‘somewhere on the network.’’ OLC allows a staff on the order of twenty students to handle the questions and problems of over 8000 users on a network of more than 900 workstations. This paper describes the motives and design goals for OLC, its implementation, and some of the results of its three years of operation.

1. Background on Project Athena

Project Athena was originally conceived as a five-year experiment in the uses of computers in undergraduate education at the Massachusetts Institute of Technology. With support from Digital Equipment Corporation and International Business Machines Corp., M.I.T. embarked on the construction of a distributed computing environment built on scientific/engineering workstations (VAXstation II and RT/PC). In addition to this environment, Athena has funded over one hundred projects for developing educational software for use in the M.I.T. curriculum. In 1988, Project Athena was extended by M.I.T, Digital, and IBM for three years. An overview of the Project can be found in ref[1].

The Project Athena computing environment is built on a network services model of computation. Virtually every resource used during a login session is supplied by a remote server: software, personal files, mail, message-of-the-day, printing, *etc.* Athena currently has installed over eight hundred workstations and seventy server machines yet the operations staff numbers approximately eight. A central requirement of all services is that they permit easy upward growth in numbers. The Athena model of computation and building such a distributed environment is described more fully in a series of papers presented at the

† UNIX is a Trademark of Bell Laboratories.

Winter 1988 USENIX Conference.

The computing experience of Athena users covers a wide spectrum, from those who have never used a computer to advanced UNIX hackers. Even experienced UNIX users often have questions, especially since Athena has augmented the basic system with several additional software packages. Currently, there are approximately 10,000 registered users of Project Athena.

2. The Problem of Consulting

Each September, more than 1,000 new undergraduates arrive at M.I.T. Many of them are eager to start using Athena workstations at once while others will wait until required to do so by enrolling in a class that uses the Athena system. In either case most students must learn to use the computing environment during their study at M.I.T.

UNIX has never been regarded as an easy system to learn, and the process is complicated by the network server configuration and the addition of other software packages (*e.g.,* the X Window System.) This problem was recognized early in the Project, and a consulting staff was established to assist users by answering their questions about the system. Most consultants have been undergraduates who work for Athena part-time. In addition to the paid consultants, many students have volunteered their time to assist other users with the use of the Athena system. These OLC volunteers were originally Athena staff and members of the Student Information Processing Board (SIPB) at M.I.T., but the group has expanded to include many other users.

From the beginning, it has been impossible to provide consulting coverage in all locations twenty-four hours a day, seven days a week. In 1984 the need to provide some kind of on-line support for consulting became clear and work on OLC began.

3. The On-Line Consulting System

OLC is designed to be the rendezvous service between users and user consultants. Transactions between users and consultants are immediate. If there are no consultants available at the time a user asks a question, the question is held until the question is answered, even if the user logs out. A user may be helped by only one consultant at a time, however, her question may be passed from consultant to consultant or remain in a queue until the question is resolved. OLC is also designed to take advantage of other services in the Athena service environment in a way that does not compromise the basic reliability constraint.

The complete OLC system consists of three pieces: a user program *(olc),* a consultant's response program *(olcr),* and the coordinating daemon *(olcd).* In addition to these programs, there is a collection of ''stock answers'' or prepared responses to common questions. Athena has also established certain management procedures for using OLC which are described later in this paper.

3.1. Design Goals

Some aspects of the design of OLC are those common to the design of any network service. OLC must be:

Scalable:	One of the goals of Project Athena is to provide a large-scale computing environment with minimal operational overhead. OLC must be able to quickly process and maintain large numbers of requests and to meet increased demand gracefully.
Reliable:	The service should be available whenever any user requires it (all the time).
Recoverable:	If the system crashes, it should recover with minimal human intervention, preferably with none at all. OLC should restore its state such that all current conversations are preserved.
Usable:	In some ways, OLC provides the equivalent of a ''help'' system. When a user understands nothing else, he should be able to easily use OLC to contact a consultant without first asking how to use the program.
Inexpensive:	As the user population grows, it is inevitable that OLC is to become a service provided by users, for users. OLC must therefore be simple to manage and maintain.

-- --

3.2. The OLC Daemon

This daemon provides a central rendezvous for user questions such that consultants can read and answer them. All user-consultant transactions are mediated by the daemon so it can handle proper notification, logging, and scheduling of the questions and transactions.

3.2.1. Connections

The daemon maintains the connections between users and consultants. A user may be connected to only one consultant and vice-versa. Connections are merely symbolic pointers within the daemon. OLC clients do not interact directly with each other. This has the advantage of maintaining reliable, current logs of all transactions and state data.

3.2.2. Types

Three different classifications of users are currently defined in the OLC daemon:

1. User who is asking a question
2. Staff consultant
3. OLC volunteer

Any user, including registered consultants, may ask a question. Active questions entered by consultants do not interfere with their role as consultants. Consultants come in two flavors but their functions are identical. The distinction is made merely to allow staff consultants extra flexibility within the system (see below). A consultant is registered in the OLC database by the OLC manager.

3.2.3. Specialties & Priorities (Auto-connection)

Users are best served when their questions are tended to promptly. Consultants may *sign on* in OLC so that the daemon will automatically connect them to waiting users. The daemon will connect the consultant to the first user in the queue whose question topic matches one of the consultant's declared specialties. A specialty is a topic in which the consultant has a strong background, indicating a routing preference for questions. The specialty is declared in the OLC database by the OLC manager.

The daemon will connect new questions with consultants currently signed on. This is determined by the level at which the consultant signs on and by his specialty list. A staff consultant may sign on at one of three levels (in order of priority):

on duty

on

on urgent

The duty level is the front line; consultants signed on at this level are connected to users ahead of any other consultants signed on. The urgent level is useful when someone wishes to help users, but only if no other consultants are available.

The OLC volunteer may only sign on one level. If there is more than one consultant on a given level, a new question is given to one of those consultants with a specialty in that question topic. In addition, volunteers will not automatically connect to questions with topics outside of their specialty list. This feature allows for volunteers who are knowledgeable in a few areas of the system, yet may know little about other areas.

This design not only serves users promptly, but makes technical matches between users and consultants.

3.2.4. Logging

All transactions between a user and a consultant are stored in a log which resides on the server. This feature is necessary so that questions can be transferred among consultants.

Upon the resolution of a question the daemon forwards the conversation log to a third party archive service. The logs are categorized by the topic of the question asked. The daemon is configured to work with the *notes* conferencing system developed at the University of Illinois (ref[3]), or the *discuss* networked conferencing system developed by the Student Information Processing Board at M.I.T. (ref[4]). These logs are kept as an official record of all transactions over OLC. Often a consultant will need to refer to a past conversation for reference or another consultant may scan the logs for accuracy and polite interaction. The OLC logs are a useful mechanism for Project Athena to evaluate itself by noting common problems and questions about the system.

The consulting staff uses this information to prepare itself for commonly asked questions by creating stock answers (see section 3.5). The logs are also a useful pulse of the community.

3.2.5. Notification

The daemon will notify users and consultants of any status changes to their conversation such as a new connection or message (See section 4.4).

```
% olc
Welcome to OLC, Project Athena's On-Line Consulting system.
Copyright (c) 1988 by the Massachusetts Institute of Technology.

If you need help with this program, type 'help' at the "olc>" prompt.
There are 2 busy consultants, with 4 waiting users.

There is still one machine in building 4 down due to hardware problems.
This should not affect most people, although some workstations in
buildings 16, 4, and 2 may not work correctly. Repairs may extend over
the weekend.
                -- 8:00PM, September 30

Please type a one-word topic for your question.  Type ? for a list of
available topics or ^D to exit.

Topic: workstations

Please enter your question. End with a ^D or '.' on a line by itself.
How do I get a clock on my screen?
^D

There is no specialist currently available for topic 'workstations'. Your
request will be forwarded to the first available consultant.
olc>
```

Figure 1: The olc startup screen (with typical contents)

3.2.6. Host Requirements

The OLC daemon runs on a private secure host to avoid tampering with user conversations or logs. The host must be accessible via TCP/IP connections within the distributed environment. In addition it must have enough accessible disk space to house the active conversations and the resolved logs.

3.3. User Client: *olc*

3.3.1. The startup screen

When a user starts olc, the client displays three pieces of information (see Figure 1):

1. How to get help within the program: every command may be explained by 'help <command>'. These explanations are straight forward and simple to avoid generating additional questions.

2. The status of the OLC queue: the queue itself is hidden from the user. However users are generally interested in knowing about how long it may be until their questions are seen by a consultant. This status provides the number of consultants and the number of active users in the queue. The user may request this information during an OLC session.

3. A short message of the day: Occasionally an event may occur within the system which causes many users to ask the same question. To ease the load on consultants in such an event, *olc* displays a short message which may account for down systems, etc. This message can be used to display OLC down times, system problems, consulting hours, or even new ways to get help on the system.

3.3.2. Use of olc

After *olc* displays the message of the day, the user is then asked to supply a topic from a predefined set and the text of his question. This is all that the user is required to do until she is connected to a consultant. During the time the user waits, she may send additional messages which will be held for the consultants in the user's log, query the status of the queue, or exit *olc* and do other computing in the meantime. When the user re-enters *olc,* the client will skip the initial query for a topic and question and place the user at the OLC prompt. At this point the user may send new messages or replay the conversation so far.

The two commands the user must know are *show* and *send. Send* will take additional input from the user which will be appended to the log on the server. If the user is connected to a consultant, then the server will notify the consultant of the user's new message. *Show* will display new messages from the consultant (see figure 4 for an example).

3.4. Consultant reply program: olcr

Olcr is essentially *olc* with additional commands to manipulate connections with users. Consultants have the ability to connect to users, send messages to the connected user, forward or resolve the user questions (see *Figure 5* for an example).

3.4.1. The OLC queue

The OLC queue seen by the consultant is shown in *Figure 2.* The list contains all the essential information in OLC. The two left columns display the user name, location, and status. The status may be one of *unseen, pending, active,* or *logout.* The next column lists the consultant's username. A consultant listed on the same line as the user indicates that he is connected to that user. The last columns display the number of consultants who have been connected to that user and the topic of his question. It is possible to configure the daemon so that it will forward the question to a log when a certain number of consultants have been connected to the question but have failed to resolve it. This feature prevents the ever increasing size of the queue and bouncing of unanswerable or difficult questions inside the queue. The OLC Manager (see *section 5*) can take care these forwarded logs at a later time.

3.4.2. When a user logs out

Olcr has the ability to send mail to a user in the event the user should log out while having a pending question in the queue. This feature enables consultants to eliminate already answered questions in the queue and allows the user to see consultant responses.

In the Athena system, Mail is forwarded to a central mail hub. The Athena mail hub is a separate service which delivers mail to the appropriate address. The olcr client will notify the consultant if the user is unknown to this service. Specific user addresses may also be specified for use with more traditional systems.

3.5. Stock Answers

After OLC was first put into service, it was discovered that users frequently asked the same, or very similar, questions. For example, ‘‘How can I print my paper?’’ is a common question. There were also a number of less common but somewhat technical questions, such as ‘‘How can I link my C program with a FORTRAN library?’’ It was tedious for consultants to retype answers to common questions, and difficult to write answers quickly for more technical topics. Over time, consultants began to keep individual caches of answers, and occasionally these caches would be shared with other consultants.

Once the problem was recognized, a solution was obvious. The consultants gathered a collection of these ‘‘stock answers,’’ and a simple browser program was written to enable consultants to find an appropriate stock answer quickly. Once found, the answer could be tailored to the specific question and sent to the user.

Stock answers are organized in a hierarchy, with the categories and placement of answers determined by the OLC manager. The browser interface is similar to that of *notes* (see *Figure 3*).

Single-letter commands are executed immediately with no carriage return required; an entry is selected by typing its number as displayed in a menu on the screen. The browser has full support for saving entries to files and for adding new entries. Some experience with the interface is required to use it easily, but it works very quickly once it has been mastered. Currently, the browser is also available to users as a supplementary set of on-line documentation.

olcr> *list*

User	Status	Consultant	Topic
hyland@M66-080-2.MIT.EDU	(pending)		(2) emacs
jamarroq@HAWAII.MIT.EDU	(logout)	geer@E40-342F-2.MIT.EDU	(5) fortran
mcampos@GRILL.MIT.EDU	(logout)	beth@KLEE.MIT.EDU	(2) xwindowsystem
gmbelaus@BECKS.MIT.EDU	(logout)		(1) scribe
twleung@E40-358D-1.MIT.EDU	(logout)		(1) mail
enprahba@RENOIR.MIT.EDU	(pending)		(3) unix
annette@GRUMPY.MIT.EDU	(active)	elsj@M4-035-10.MIT.EDU	(1) other
carla@SNOOPY.MIT.EDU	(active)	vanharen@FRIES.MIT.EDU	(1) accounts
tjcoppet@PICASSO.MIT.EDU	(pending)		(1) lisp
not connected		treese@CIROCCO.MIT.EDU	
not connected		tjcoppet@PICASSO.MIT.EDU	

olcr>

Figure 2: The OLC queue as displayed in olcr

Currently, an interface for the X Window System (ref [5]) is under development. Advantages of this version include the ability to view a directory of entries and a particular entry simultaneously as well as a ‘‘friendlier’’ method of finding desired entries. It is possible that a more general ‘‘help’’ system can be based on this interface.

4. Interaction with Other Athena Services

4.1. Athena mail hub

To control the flow of mail within the Athena network, as well as in and out of Athena, all mail is sent to a central mail server which has the information to forward the mail to the appropriate location. These locations are known as Post Offices, ref[6], which are essentially mail holding servers. The user can use any one of the mail clients available on Athena to retrieve and read their mail messages without knowing about the existence of these servers.

4.2. Hesiod

Hesiod, ref[7], provides name service for Project Athena. The clients query the Hesiod service for the location of the OLC server. This location may be changed at any time without altering the any of the OLC programs.

4.3. Kerberos

In an environment of more than 800 public workstations on an insecure network it is necessary to confirm the identity of the consultant when he is assisting users within the OLC system. Services at Project Athena use a third party service called Kerberos, ref[8], to perform this authentication. As a policy, the OLC daemon refuses connections from consultants who fail this authentication mechanism. Currently no authentication is required to ask a question because a user may have a question about this authentication system and turn to OLC to find the answer.

4.4. Zephyr

The OLC daemon sends all notifications directly to the user. Normally this would occur via a system *write* message to the user's terminal or workstation, however, this requires the OLC daemon know which machine the user is logged into at all times. This is not possible when a user logs out and logs in again at another machine, and does not make any requests from olc which would update his location. To solve the problem of finding a user on the network, Project Athena has developed a location and notification service known as *Zephyr*, ref[9]. The OLC daemon sends notifications via this service, and, by default, the user will receive the notification at the tty or inside a *windowgram* which is essentially a small window containing the notice OLC has sent. This is however user configurable and if this system fails, the user will receive the message directly from the UNIX write function.

```
            On-Line Consulting Browser
      /usr/athena/lib/olc/stock/stock_answers/emacs


       1  Controlling the emacs window size
       2  Using the mouse buttons in Emacs
       3  Moving the cursor without the mouse
       4  Finding what line you are on
       5  How to set a mark
       6  Line number mode in Emacs
       7  How to set auto-fill by default
       8  How to recover Emacs autosave files
       9  Customizing Emacs using a .emacs file
      10  How to use Emacs with scheme
      11  How to use Emacs with a newer scheme
      12  Using TAGS in Emacs
      13  Rebound keys for VT240

               ** End of Index **

olc_browser>
```

Figure 3: The OLC Stock Answer browser (a typical display)

4.5. Network dependencies

The first requirement for a functional OLC system is that the OLC server be operational. If the OLC daemon is running on a working machine the clients simply need to establish a TCP/IP connection. Generally if the user can successfully log into a workstation, he can successfully use OLC. The only third party service requirements are *Hesiod* for the name service, *Kerberos* for authentication, and of course, the user must be able to obtain the client. At Athena, the client is delivered via a cluster library server which provides most of the essential software to workstations in a particular location. If any one of these should fail, the workstation is unusable.

Interaction with other services like *Zephyr* is non-essential since the daemon is capable of sending notifications directly to the user's workstation.

5. Managing OLC

5.1. OLC database

The database resides on the daemon host. It contains a list of usernames allowed to act as consultants along with the topics in which they are specialists. The usernames may be marked to indicate if the consultant is a volunteer. This file is stored in ASCII text so it may be edited easily.

The OLC topics are listed in a separate file and also stored in ASCII text. Since the daemon will file conversations by topic, it is necessary that the logging mechanism be updated for any changes to this topic file. Either file may be edited without halting the daemon.

5.2. The OLC queue

Some difficult questions may linger in the queue, especially if the user logs out since logged out users will not be automatically connected to a consultant. It is advantageous to keep the OLC queue short. Large queues often contain many questions from users who logged out and did not provide sufficient information so that a consultant could send a definite answer.

In general, every question is answerable by at least one member of the Athena staff. It is ultimately the job of the OLC manager to find the answer to difficult questions. In rare cases, such as requests about network addresses of remote hosts or debugging of large programs, the user is given information on how to help himself. Often this is a reference to materials off-line.

5.3. Orientation

The OLC manager is also responsible for orienting new consultants and volunteers with the proper use of the *olcr* client. An olcr manual exists but orientation is geared toward handling difficult situations with users that might lead to misunderstandings or arguments. Since OLC is one of the major user interfaces to Project Athena, it is necessary to ensure smooth communication between users and consultants.

6. Problems and Future Directions

Soon OLC will allow consultants to answer multiple questions simultaneously. The advantage is that other users may be served while a consultant is waiting for one user to respond, i.e. consultants will be more efficiently utilized.

As more OLC volunteers are added, OLC approaches the model of a system of users consultation provided by users. These volunteers are largely responsible for the success of OLC. The number of volunteers will gradually increase to fill the need.

An invaluable source of information about the functionality of the system and consultant performance can be found in the OLC logs. However, the man-power required to read over 6000 questions each term is greater than what Athena is able to provide.

The limits of OLC scale are dependent upon the limits of the OLC server (an estimated 300 questions). With some work, the OLC system can be expanded to utilize multiple servers simultaneously. Each OLC server may be responsible for some area of specialty or area of administration. The limits on the type of scale is dependent upon the limits of Athena's authentication, notification, and name services across internet regions.

7. Conclusion

A team of 10 consultants and 20 volunteers provide help to users 24 hours each day. Even when a particular consultant may not know the answer to a question or problem, it is reassuring to the user to know that he is able to contact another person on the network. OLC has proven to be the solution for providing user assistance on a broadly distributed network system.

8. Acknowledgments

The original design of Project Athena's OLC (based on an on line consulting system provided by Information Services at M.I.T.) was done by Bill Saphir. Dan Morgan and G. Winfield Treese of Project Athena continued the implementation through to a usable system.

We would also like to thank Marc Campos, Theodore Leung, and Gregory Belaus for maintaining and providing new ideas for the OLC system; Carla Fermann, Chris VanHaren, and Linda Kim for reviewing early drafts of this paper.

9. Sample OLC sessions

The following pages contain sample *olc* and *olcr* sessions. The text contained in the boxes are asynchronous notifications sent by the daemon via *Zephyr* or */bin/write*.

(continued from Figure 1)

Message from OLC daemon...
You are connected to consultant Tom Coppeto (tjcoppet@PICASSO.MIT.EDU)
Tom has received your question.

Message from OLC daemon...
Tom has sent a reply.
To read it, use the 'show' command from olc.

olc> *show*
time: 88/09/28 03:07:23

Hi Joe,

To get a clock on your screen type:
xclock &
When your cursor changes to an upside down L, click the left mouse button where you want the clock to appear.

- Tom

olc> *send*

Enter your message. End with a ˆD or '.' on a line by itself.
Thank you.
-Joe
ˆD

What now? *send*
Message sent.

olc> *done*

Using this command means that the consultant has satisfactorily answered your question. If this is not the case, you can exit using the 'quit' command, and OLC will save your question until a consultant can answer it. If you wish to withdraw your question, use the 'cancel' command.
Really done? [y/n] *y*

Goodbye from OLC. Glad to be of service!
%

Message from OLC daemon...
Tom has marked your question resolved.

Figure 4: An example olc session with joe user.

Message from OLC daemon on MATISSE.MIT.EDU... You are connected to user Joe User (jruser@SOMEWHERE.MIT.EDU)

olcr> *replay*

Log Initiated for user Joe User (jruser@SOMEWHERE.MIT.EDU)
(88/09/28 03:14:45)

Topic: workstations

Question:
How do a get a clock on my screen?

--- Question grabbed by consultant tjcoppet@PICASSO.MIT.EDU.
[88/09/28 03:15:03]

--- Connected to consultant tjcoppet@PICASSO.MIT.EDU
[88/09/28 03:15:04]

olcr> *chtopic xwindowsystem*
Topic changed to 'xwindowsystem'

olcr> *send*
Enter your message. End with a ˆD or '.' on a line by itself.
Hi Joe,

To get a clock on your screen type:
xclock &
When your cursor changes to an upside down L, click the left mouse button where you want the clock to appear.

- Tom
ˆD

What now? *send*
Message sent.
olcr>

Message from OLC daemon... OLC message sent from user.

olcr> *show*

time: 88/09/28 03:08:46
Thank you.
-Joe
olcr>

Message from OLC daemon... User is done with question.

olcr> *done*
Enter a title for this conversation: *how to get an xclock*
Question resolved.
olcr>

Figure 5: Example olcr session

10. References

1. E. Balkovich, S.R. Lerman, and R.P. Parmelee, "Computing in Higher Education: The Athena Experience," *Communications of the ACM* **28** (11), pp 1214-1224, ACM (November 1985).

2. G.W. Treese, "Berkeley Unix on 1000 Workstations: Athena Changes to 4.3BSD," pp. 175-182 in *USENIX Conference Proceedings,* Dallas, Texas (February 1988).

3 R.B. Essick IV and R. Kolstad, "Notesfile Reference Manual," *UNIX User's Supplementary Documents,* February 1983.

4. K. Raeburn, J. Rochlis, W. Sommerfeld, and S. Zanarotti, *Discuss:* An Electronic Conferencing System for a Distributed Computing Environment," in *USENIX Association Winter Conference 1989 Proceedings,* February 1989.

5. R.W. Scheifler and J. Gettys, "The X Window System," *ACM Transactions on Graphics* **5** (2), pp. 79-109 (April 1987).

6 M.T. Rose, "Post Office Protocol (revised)," University of Delaware, 1985.

7. Stephen P. Dyer, "The *Hesiod* name server," In *USENIX Association Winter Conference 1988 Proceedings,* pp. 183-190, February 1988.

8. J.G. Steiner, B.C. Neuman, and J.I. Schiller, "Kerberos: An Authentication Service for Open Network Systems," pp. 191-202 in *USENIX Conference Proceedings* Dallas, Texas (February 1988).

9. C. Anthony DellaFera, "The *Zephyr* notification service," In *USENIX Association Winter Conference 1988 Proceedings,* pp. 213-220, February 1988. *Project Athena Technical Plan: Section E.4.1: Zephyr Notification Service,* M.I.T. Project Athena, Cambridge, Massachusetts (December 21, 1987).