KMi Planet: A Web Based News Server

John Domingue and Peter Scott
Knowledge Media Institute
The Open University
Milton Keynes, MK7 6AA, U.K.
http://kmi.open.ac.uk/people/domingue/
j.b.domingue@open.ac.uk

Abstract

In this paper we describe KMi Planet, a web based news server which facilitates communication within a our laboratory and allows the wider community to access lab-related items of interest. Unlike most news-of-the-future projects, which focus on personalised news feeds, we have concentrated on replicating some of the functionality provided by newsrooms. Our goal was to reduce the article submission overheads for the members of our lab, by allowing them to send in articles using plain email text, whilst creating high quality news pages.

1. Introduction

In this paper we describe KMi Planet, a web based news server which facilitates communication within a our laboratory and allows the wider community to access KMi-related items of interest.

Most news-of-the-future projects, particularly those based upon the concept of intelligent agency focus on the idea of personalised news feeds [2] [5]. In these systems the reader *delegates* the activity of finding interesting stories, possibly from a wide variety of sources such as the web and usenet newsgroups, to an agent which 'knows' the reader's interests. Within the KMi Planet project we have investigated how a web-based news server could aid in the task of creating a high quality web based news page from widely accessible input. Our aim was to enable 'journalists' to send in stories from the 'front line' with minimal hardware constraints.

The Planet architecture is based upon the LispWeb server [10]. Within LispWeb, HTTP requests are handled by arbitrary Lisp function calls. The Planet then is implemented as a set of Lisp modules replicating some of the activities found within news agencies.

2. Communication within an Organisation

Organisations need to keep members informed. Most organisations of a significant size do this via an internal publication. The major problems with internal publications is the effort involved in getting a story from a member of the organisation into the publication, and the delay in getting the publication to all the members. The hurdles include:

- transcription - transcription software is not readily available when the story is 'hot'. There follows a time delay after which members may suffer from motivation and recollection problems,
- distribution - paper based techniques necessarily have a time delay,
- archiving, and
- searching.

Within some organisations internal mail lists are used to keep members informed. Whilst alleviating the problems of distribution and archiving, messages within a mail list constitute a low quality medium (single font and style used, graphics can be attached but are not in place or laid out).

Many organisations are now deploying electronic and networked news delivery to facilitate organisational communication. Distribution techniques vary between email lists, sites in HTML, to the use of commercial news feed systems like PointCast™ [8], Headliner™ [4], and WebCast™ [11]. A number of organisations (e.g. Apple and EuroParc) use large screen displays in social areas to increase employee awareness. Even with commercial news feed systems, which can provide the news very effectively to the news-user, the construction of high quality news articles still represent a substantial editorial and journalistic effort that can take up a great deal of time.

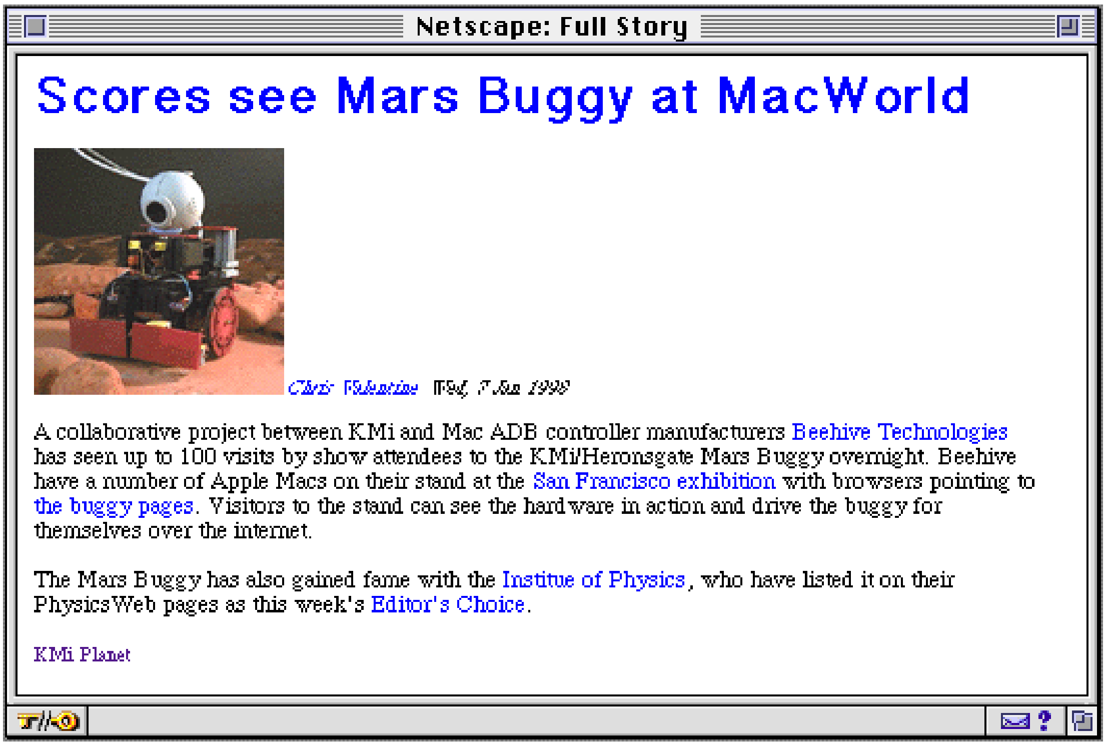


Figure 1. A screen snapshot of a full story displayed in KMi Planet.

Figure 2. A screen snapshot of the KMi Planet front-page.

3. Our Approach - A Web Based News Room Server

Our approach to alleviating the above problem is to provide a news server which accepts stories submitted in the lowest-common-denominator-medium - an email message - yet is able to create a high quality web page. The key to our rationale is that the submission must be as simple and easy as can be arranged so that it will be used. The system should take the pressure off the journalist and editors by making their contributions entirely lightweight - by having a web based news server taking on much of the work.

In our model the Planet is given stories directly by researchers, who in this context play the role of journalists, and takes on the responsibility of editing these and presenting them to the research community as news. In this simplest communication model a journalist sends an email message to the KMi Planet story account. The subject line of the mail message becomes the headline of the story. The body of the message becomes the text of the story. If they attach an image to their mail message, it is added into the story in an appropriate place. In figure 1 we see a single story web page rendered from an email message. The message header is at the top, the attached image follows beneath and then the name of the "journalist" who sent the message (linked to their home page) and the date of the submission. The message body is shown below, with web links appropriately anchored to their destinations. In the current Planet view this is what single stories look like. Although the human journalist is responsible for sending in a good story, all the rest of the process is handled by the software.

Given that we have a database of such stories then we could present them in a number of different ways. The most obvious is the newspaper-like multiple story page where the submitted story is presented in the context of other stories. A snapshot of a KMi Planet multi-story news page is shown in figure 2. Here we see the story from figure 1 taking the top half of the page with three story columns beneath it. The top story image has been scaled to fit into this page design as an illustrative graphic and the body text has been truncated to fit into the cell. Finally, a button has been added to lead one back to the full story (of figure 1). In the columns beneath we see two further truncated story views - in the left two columns we see less of the body of the story, with no illustration, whilst in the rightmost column are only some headlines and authors names. In the current Planet view, this is what the Planet front page looks like.

Finally, in the screen snapshot of figure 3 we see a very simple headline listing of a large set of stories - listed in this case by date. In the current Planet view, this is what the story archive looks like.

Figure 3. The KMi Planet archive of stories.

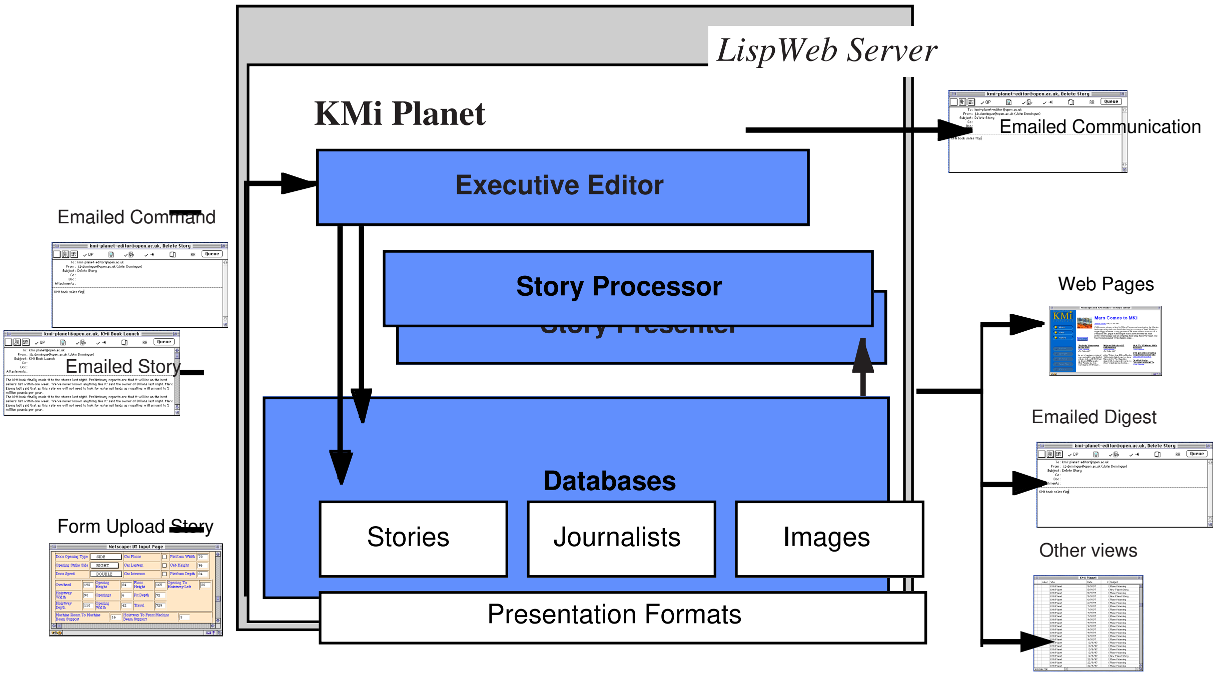


Figure 4. The KMi Planet Architecture.

As can be seen from figure 4 the KMi Planet architecture, built on top of the LispWeb server [10] has four main components:

- **An executive editor** - which has overall control, and can respond to commands from the human editors and generate requests for the other modules or human collaborators.
- **A story processor** - which can take story submissions and maintain the database entries for them.
- **A story presenter** - which presents the Planet stories to the client.
- **A set of databases** - which include the stories that make up the Planet's news; the journalists who may contribute the stories; the images that story presenter may use to illustrate the stories; and a small database of presentation formats.

3.1. The Planet Databases

The stories database contains the text of the stories that have been provided to Planet by the journalists. The stories are stored in a simple text form, and have been processed by the story processor to identify unusual text characters.

The journalists database is maintained by the executive editor and is a record of all the people who are registered to contribute to the publication. In the case of the KMi Planet, all researchers in the Knowledge Media Institute are registered as "journalists" with this system. The database holds each journalist's the email address and the address of his or her home web page and a picture of the journalist. The picture is used in some presentation formats, particularly for a detailed web page in which no suitable other image can be found.

The images database is a collection of the images that can be used to illustrate stories. The main source of these images is from journalists who add GIF attachments to their emailed stories.

The presentation formats database is a set of templates, and instructions that can be used by the story presenter to format stories for output. So, for example, the web front-page format is a HTML framework into which stories may be added when the story presenter receives a request for the Planet front page. Other formats include the web page archive view, the web single story view, the Apple PlainTalk™ [7] speech output interface (described below) and even the plain text view. The templates are easy to change and to therefore make a very different style of presentation in each of these views.

3.2. The Executive Editor

As its name implies, this component is in overall charge of the system. As can be seen from figure 4 it takes input from various sources (usually email) and can respond using email. It can write directly to any of the Planet databases or through the story processor

At the core of the executive editor is an email processor - as this module is responsible for maintaining a number of email accounts that relate to the work of the Planet. The executive editor reads mail in these accounts and acts appropriately. The message could cause the module to change a database entry directly, to generate an email reply or even be passed on to another module. At the moment there are 3 main email accounts: a demo account, a story account and a command account.

The first is the easiest to deal with, as messages to the demo account are deemed insecure and for demonstration purposes only and so are passed on directly to the story processor with the injunction that they not be added permanently to any database. These entries are ephemeral and only used to test out the Planet - to see for example, how a story might look if it were to be submitted to the Planet properly - or to show a visitor a temporary example of the use of the system. For both the other accounts the module first checks with the journalists database that the mail is verified as coming from a registered source. If this is not the case then the executive editor generates a polite refusal message inviting the contributor to contact the human editorial team. Only registered journalists can contribute to the second account and entries from these are assumed to be story contributions which are passed on to the story processing module. Some journalists are members of the human editorial team and allowed to contribute to the command account.

Through this channel they are allowed to mail editorial commands. The human editors can instruct the executive editor to perform simple actions. For example, simple commands at present include:

- add, delete or edit stories in the story database,
- add or delete registered journalists, and
- change the layout of the web news page, for example change the top story or change the number of stories displayed.

The executive editor can use email to respond to submissions, for example, it acknowledges the receipt of all stories. However, it can also use email proactively. For example it can use simple heuristic measures to chase up stories from journalists. At the moment, the Planet notices when the top stories fall below simple thresholds - set on command, by the human editors. When these thresholds are exceeded (e.g.. the latest story is embarrassingly old) the executive mails a message to this effect to the editorial team. It can then be instructed to chase up errant journalists for fresh copy.

3.3. The Story Processor

The executive editor feeds stories from registered journalists to a story processor which is primarily responsible for maintaining the image and story databases.

On receipt of an emailed story an entry in the story database containing the email messages's author, date, and subject line, as well as the body of the message and a pointer to any attached elements is created. If the attachment is a GIF image then it is decoded before adding it into the picture database. If any problem is detected in processing the story, the journalist is emailed directly about the problem. Most typically, this will involve a problem processing the image. The journalist can then send the image again. If the story is sent as an upload from an HTML page the procedure is much the same.

3.4. The Story Presenter

Whenever an HTTP request comes from a web browser client the LispWeb server passes the request on to the story presenter. This module uses the formats database (described above) to generate an appropriate response. So, for example, if requested for the front-page (using a front-page multi-story format) the story presenter will create a page of stories where each displayed story is composed of the following:

- *a headline* - taken from the subject line of the mail message,
- *an author* - taken from the author of the mail message,
- *a date* - taken from the date of the mail message,
- *a story* - taken from the body of the mail message, and
- *an associated image* - a GIF image either sent as an attachment with the story or an image of the author taken from the journalist's database.

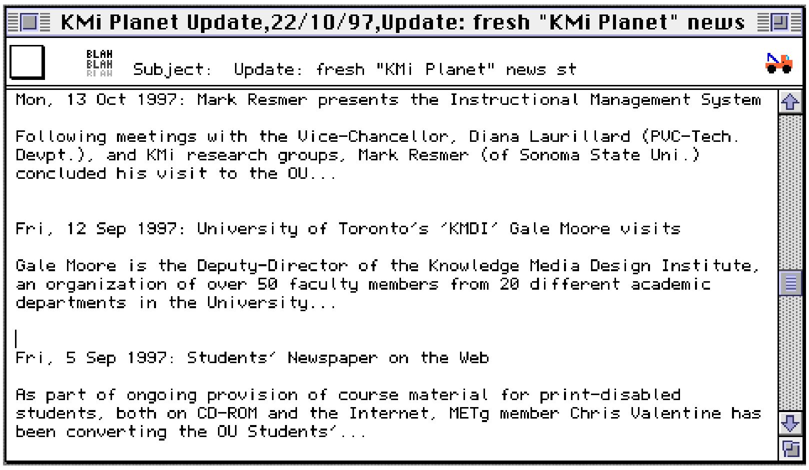
Note that in the front-page format shown in figure 2 the image is only used for the story that is presented as the top-story of the page.

In addition to providing web based news pages the KMi Planet can also serve its news via multiple other routes as generating most text based formats is no trickier than generating HTML.

One of the more proactive formats is the mailing list. The executive editor sets the parameters for the mailing list format via simple flags in the relevant format database entry. So for example, in the current mailing list format, a mail-out threshold is set to six submissions. When six new submissions have been received the story presenter will construct a condensed text digest of these stories and email them to a subscription based mailing list. A screen snapshot of part of a mailed digest is shown in figure 5.

Another format which we currently employ is text to be read by a text-to-speech processor. So, for instance, one of the formats in the database used by the story presenter allows it to perform some simple formatting to assist the Apple PlainTalk™ software that benefits from some elementary phonetic pre-processing help. For example, at the moment, PlainTalk™ reads our unprocessed logo (KMi) as "ke-mee" rather than the "kay-em-eye" which we would prefer. Similarly, the version we are using has problems reading HTML text anchors and email addresses correctly. We have defined a set of formats and filters to avoid some of these problems. One use of text-to-speech systems like PlainTalk™ is to 'can' arbitrary text as speech for presentation in a number of interesting ways. In the Planet experiment we have shown how this may be applied to a VoiceMail interface to the system.

In figure 6 we present a small segment from the KMi Planet VoiceMail interface. This program is written using the Bing Software PhonePro™ software [1] for Macintosh. The software uses touch-tone phone controls to allow audio access to the KMi Planet stories. Essentially, all that the phone system does is to lead you quickly and effectively to any story that you want to hear, that is then read to you over the phone using Apple PlainTalk™. The usefulness of the Planet VoiceMail system is derived from its simplicity and the use of the story presenter to give the phone interface the appropriate files from the Planet story database in the best form to read.



KMi Planet Update,22/10/97,Update: fresh "KMi Planet" news

Subject: Update: fresh "KMi Planet" news st

Mon, 13 Oct 1997: Mark Resmer presents the Instructional Management System

Following meetings with the Vice-Chancellor, Diana Laurillard (PVC-Tech. Devpt.), and KMi research groups, Mark Resmer (of Sonoma State Uni.) concluded his visit to the OU...

Fri, 12 Sep 1997: University of Toronto's 'KMDI' Gale Moore visits

Gale Moore is the Deputy-Director of the Knowledge Media Design Institute, an organization of over 50 faculty members from 20 different academic departments in the University...

Fri, 5 Sep 1997: Students' Newspaper on the Web

As part of ongoing provision of course material for print-disabled students, both on CD-ROM and the Internet, METg member Chris Valentine has been converting the OU Students'...

Figure 5. A screen snapshot of the news digest sent out by Planet.

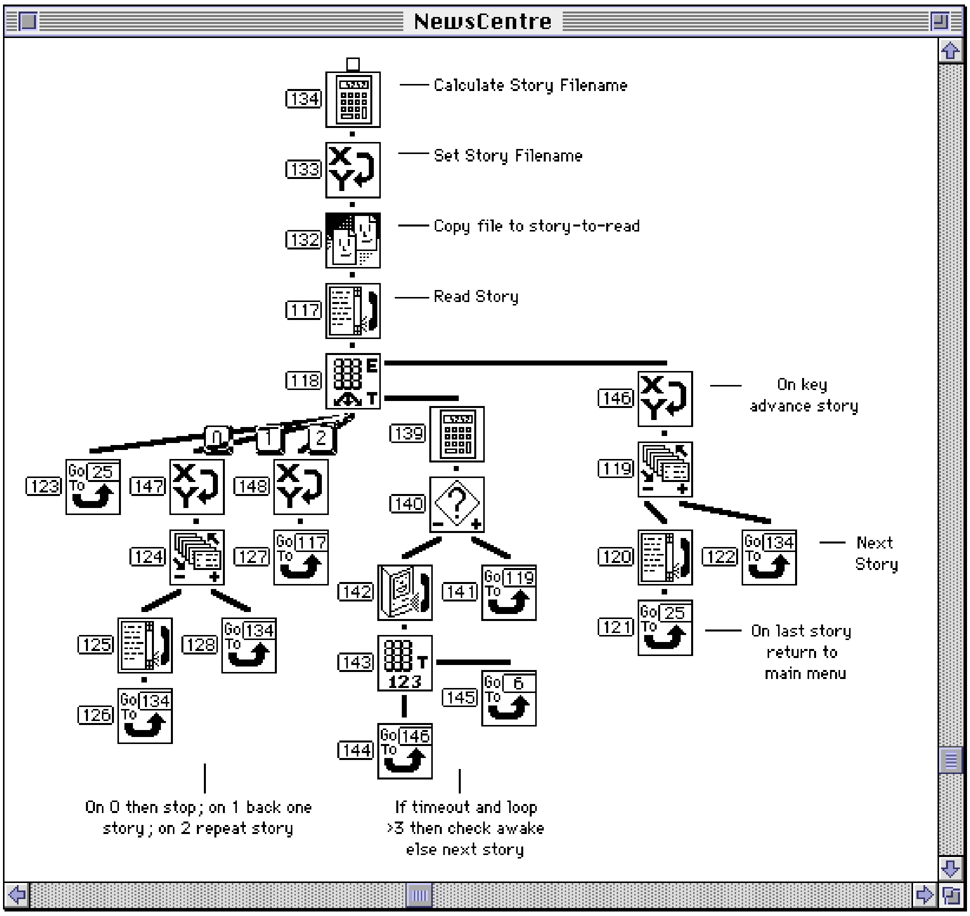


Figure 6. A segment from the Planet VoiceMail system.

4. Ongoing Work

We are currently working on a number of different developments of the KMi Planet project. At the simplest level we are extending the system to handle different image encodings, we are adding in new formats for presentation and also looking at increasing the support Planet can give to the communication between human journalists and editors. Let us consider ongoing innovations in each part of the system in turn:

Innovation for the Executive Editor: most of the commands that the executive editor responds to are simple flag settings and database manipulations. Our goal for is for the executive editor to be able to manage the team of journalists, setters and printers. One obvious development is in the proactive email correspondence with the editors and journalists. Current use of email is crudely based around very simple settings and heuristics which should be much more flexible and sensitive. For example it might notice a story (perhaps through the tagging below) that would naturally benefit from a follow up and invite the journalist to do so, if the submission rate falls.

Innovation for the Story Processor: We plan to hook the story processor to a KMi story ontology which we are currently developing in our web accessible knowledge modelling language [6] [3]. We envisage that editors would tag incoming stories using the ontology. This enriched story repository could then be used to facilitate semantic searches and individualised presentations. We will also investigate the extent to which ontologically enriched stories could provide an organisational memory for KMi and promote organisational learning. Thus far we have created the first 3 levels of the ontology from an analysis of the current 50 stories. The current top level classes are: trip report, use of KMi technology, KMi visitor, and donation to KMi.

We are also extending the VoiceMail interface - to add voice notes to the news web stories. We have already prototyped a simple interface for journalists to dial in to a VoiceMail server and leave voice messages that can be linked to individual news items. The voice note is converted to RealAudio™ [9] format, copied to the server and a suitable link is added.

Innovation for the Story Presenter: we have already indicated that it is possible for the story presenter to write out stories from the database in any format. We are currently adding to the number of presentation formats that it can write to: e.g. push technology and other delivery mechanisms such as HeadLiner, PointCast and WebCast. Each of these supports a database format - that in principle - the story presenter could write to.

We also aim to increase the system's adaptability in recognising the differing capabilities of different browsers and even the differing needs of different end-users through the use of ontologies. We intend to investigate how the use of ontologies can support the automatic selection of pictures to illustrate stories and aid in page layout.

Innovation in services: As well as serving the KMi research lab and its community, a version of the Planet is currently also being developed for the Open University Business School. We need to consider how we may use this architecture to serve other news and publishing services that share these problems. Our ideal would be to provide a simple direct manipulation authoring environment for creating your own Planet so that the system can work for you with the minimum of coding on your part.

KMi Planet is currently available at http://kmi.open.ac.uk/planet/.

References

1. Bing Software PhonePro (1998) http://www.bingsoftware.com.

2. Chesnais, P. R., Mucklo, M. J. and Sheena, J. A., (1998) The Fishwrap Personalized News System. http://fishwrap-docs.www.media.mit.edu/docs/dev.

3. Domingue J. (1998) Tadzebao and WebOnto: Discussing, Browsing, and Editing Ontologies on the Web. Proceedings of the 11th Banff Knowledge Acquisition Workshop, Banff, Alberta, Canada, April 18-23, 1998.

4. Headliner (1998) http://www.headliner.com/.

5. Kamba, Tomonari, Sakagami, Hidekazu and Koseki, Yoshiyuki (1997) ANATAGONOMY: a personalised newspaper on the world wide web. International Journal of Human-Computer Studies, 46, 789-803.

6. Motta E. (1998) Reusable Components for Knowledge Models. PhD Thesis. Knowledge Media Institute. The Open University, UK.

7. PlainTalk (1998) http://www.speech.apple.com/speech/ptk/.

8. PointCast (1998) http://www.pointcast.com/.

9. Progressive Networks (1998) RealAudio http://www.realaudio.com/.

10. Riva, A. and Ramoni, M. (1996) LispWeb: a Specialized HTTP Server for Distributed AI Applications, Computer Networks and ISDN Systems, 28 (7-11), 953-961.

11. WebCast (1998) Astound Incorporated. http://www.astoundinc.com/products/webcast/webcast.html.

Acknowledgements

We would like to thank all of KMi, in particular Stuart Watt, Simon Buckingham Shum, Marco Ramoni, Mike Lewis and Marc Eisenstadt. Also we would like to acknowledge Mark Miller, Stephanie Houde and Rachel Bellamy of whose work at Apple Computer inspired our efforts.