I-Gaia: an Information Processing Layer for the DIET Platform

A. Gallardo-Antolín,
A. Navia-Vázquez
[gallardo,navia]@tsc.uc3m.es
H.Y. Molina-Bulla
h.molina@ieee.org
Dpto. Teoría de la Señal y Comunicaciones,
Universidad Carlos III de Madrid
Avda. Universidad, 30.
Leganés 28911 (Madrid) SPAIN

A.B. Rodríguez-González,
F.J. Valverde-Albacete,
J. Cid-Sueiro, and
A.R. Figueiras-Vidal
[abr,fva,jcid,arfv]@tsc.uc3m.es
Dpto. Teoría de la Señal y Comunicaciones,
Universidad Carlos III de Madrid
Avda. Universidad, 30.
Leganés 28911 (Madrid) SPAIN

T.Koutris, C.Xirouhaki,
M. Koubarakis
[koutris,xiruhaki,
manolis@intelligence.tuc.gr]
Dep. of Electronic and Computer Engineering
Technical University of Crete
UNIVERSITY CAMPUS KOUPIDIANA
GR-73100 (Chania, Crete) GREECE

ABSTRACT

In this paper we introduce one application layer for information processing in the DIET platform, a MAS development platform. This application layer is basically formed of three types of agents, here called "infocytes", designed to cater for the information needs of information providers, requesters and brokers. We have also defined and implemented under I-Gaia two separate tasks using the well-known Reuters text-classification corpus: an information-pull and an information-push task, mainly to validate the ability of such layer to effectively retrieve information on demand or spontaneously route it to users. We have also analysed the performance achieved using measurements based on precision/recall. The results show that the I-Gaia environment is completely operative and that can achieve this type of tasks without loosing performance with respect to other centralised, non-agent-based technologies with full information about the task

Categories and Subject Descriptors

H.3.4 [Information Storage and retrieval]: Systems and Software; H.3.3 [Information Storage and retrieval]: Information Search and retrieval --- *Clustering, Information filtering, Query formulation, Retrieval models, Search process, Selection process*; I.2.11 [Artificial intelligence]: Distributed artificial intelligence --- *Coherence and coordination, Intelligent agents, multiagent systems*; I.5.1 [Pattern recognition]: Models --- *statistical*; I.5.2 [Pattern recognition]: Design methodology --- *Classifier design and evaluation*; I.6.4. [Simulation and modeling]: Model validation and analysis; I.6.5. [Simulation and modeling]: Model development.

General Terms

Theory, Design, Algorithms, Performance, Experimentation, Measurement.

Keywords

- Multi-agent systems (MAS). Layered systems. Coordinating multiple agents and multiple activities. Agent societies. – Agent based software engineering. Task validation. Information Retrieval, Information Routing.

AAMAS '02, July 15-19, 2002, Bologna, Italy.

1. INTRODUCTION

The pervasiveness of personalised information needs coupled with the overpopulation of documents in electronic form has brought about the need to modulate the flow of documents between information producers and consumers.

The space information-aware agents inhabit, Infospace, is just a number of repositories to obtain documents from as seen from the information consumer side: we refer to this as the "information-pull" direction of information distribution. On the other hand, infospace is just a number of information sinks into which documents can be sent as seen from the information producer side thus we refer to this as the "information-push" direction of information distribution. Note that the information flow is always from producers to consumers, but the initiative in the flow marks whether it is an information-pull or information-push dynamics.

The complementary character of both information-pull and push directions in the information flow has already been acknowledged [2][16], but not much taken into consideration in the design of

information processing systems. These either consider information-pull in its traditional "information retrieval" guise or information-push in its well understood "information routing" flavour [1]. Other, well-known techniques and tasks, such as text categorisation, summarisation or parameterisation can be understood as facilitating subtasks for the main tasks in information-pull and push.

In this paper we introduce I-Gaia, an information processing application layer in the DIET architecture [11], which is a MAS-developing platform that addresses scalability, robustness and adaptability of applications with special emphasis in distributed deployment and operation. I-Gaia is one possible incarnation of an information-processing layer in DIET enjoying at present both information-pull and push capabilities. In the following, we validate this layer in two controlled tasks built around the Reuters news collection - one for information retrieval and one for news alerts - and give some results about them.

One preliminary note, from now on we will refer to whatever type of information bearing device in electronic form as a "(multimedia) record", although for the present work we are only dealing with texts.

2. THE DIET WAY

According to one approach, an *information ecosystem* is a complex web of interactions arising between information producers and consumers where information is interpreted in its widest sense [3]. The phrase "information ecosystem" is used by analogy with natural ecosystems. In this context, the DIET project (Decentralised Information Ecosystem Technologies), is a European collaborative research project focused on the implementation of ecologically inspired interactions in computational architectures in order to complete particular tasks, considering in particular the use of multi-agent systems to build information ecosystems.

The goals of the DIET project are the following:

? To design and implement a novel agent[1] framework via a substantially bottom-up and ecosystem-oriented approach leading to an open, robust, adaptive and scalable software platform.

? To validate and demonstrate the usefulness of the platform via four tasks/applications: information retrieval, information alert, information mining, and information trading.

? To research into the effects of alternative forms of interaction among different types of agents, under ecologically inspired software models.

The DIET software platform is designed to form the base for information management applications. To be useful in practise, the framework needs to support applications that are:

[1] Agents may be called infohabitants in DIET. The terminology has its roots in the call for proposals from the European Commission.

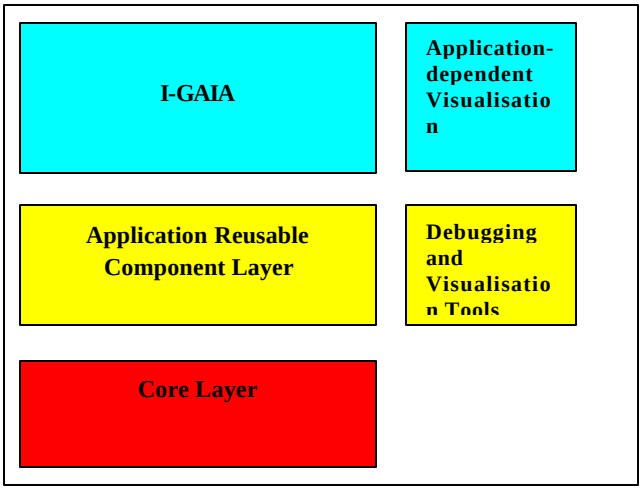

Figure 1: Instantiated DIET architecture for information-

? *Adaptive*: Information gets updated constantly, and new information is generated. Users of the information, and their preferences, as well as the system load and infrastructure, can also change. To operate efficiently, information management applications have to adapt to these changes.

? *Scalable*: There is a massive amount of information available in the real world, consider for example the World Wide Web. For an information management system to be useful, it needs to be built without any implicit limits on its size.

? *Robust*: Failures are inevitable in large-scale, dynamic, distributed systems. So the system needs to be able to cope with them. It needs to handle failing hardware, as well as cope with high system load. Performance should gracefully degrade when parts of the system fail.

? *Decentralised*: A lot of information is located in a distributed form, as the World Wide Web demonstrates. Decentralisation also helps to enhance scalability, by avoiding critical bottlenecks, and robustness, as it reduces the reliance on particular parts of the system.

The net result of this policy is the layered architecture seen in figure1 in which lower levels provide abstract services and objects which upper levels specialise into the appropriate ones for the applications to be supported.

3. AN APPLICATION-BUILDING LAYER IN DIET: I-GAIA

I-Gaia is just one instantiation of the application layer of the DIET platform dealing with information processing tasks, namely information-pull and push. Figure 1 places I-Gaia in the broadest context of the whole architecture being built within the DIET project as described in [11].

The context of use for I-Gaia is supplied by:

? *Querying users*, which submit queries to be answered by lists of documents, and

? *Publishing users*, which publish documents hopefully to reach querying users interested in them.

We suppose that query submission (or "query posting") and document publishing (or "document posting") events exactly delimit the interaction of users with the system.

3.1 The Description of I-Gaia

We next introduce agents to do the querying and publishing on behalf of the users, as well as brokering agents to adapt the flows of queries and published records. Such agents will model qualitatively and temporally the interests of their users or publishers; as well, they will take care of maximising the user/publisher utility (precision and recall for information retrieval, for instance). Furthermore, we demand that they adaptively follow any drift of those interests.

At present, however, agents in I-Gaia will not concerned with information fusion or providing decision-support evidence for their users. Such concerns will later be partially addressed by endowing them with data mining capabilities.

In order to do so, the general concept of "infohabitant" in DIET is

Table 1. Graphical representation of resources in I-Gaia

Origin of the resource	Information resource	Graphical Representation
Document	*Raw Documents*	
	Summaries	
	Parameterised documents (p-infolith)	
User query	*Query*	
	Parameterised query (s-infolith)	

further specialised (and restricted) to the three main living species at I-Gaia[2]:

? S(earch)-Infocytes (SI): Agents that search for information. These are agents which deal with the queries submitted by users and start up the query processing protocol in information-pull interactions, but also exist in information-push interactions, to receive documents on behalf of users.

? M(emory)-Infocytes (MI): Agents that publish and advertise information. These are agents which deal with the documents posted by users and start up the document publishing protocol in information-push interactions. As well, they receive queries in information-pull interactions.

? T(ransfer)-Infocytes (TI): Agents that act as mediators between Search, Memory and other Transfer infocytes by adequately forwarding requests for searching queries or publishing documents.

These infocytes only "know" about the resources depicted in table 2, a sort of ontology for I-Gaia (non-introspective in its present state).

Information resources are in I-Gaia either:

? A *published document*, to be eventually delivered to users as the real repositories of information.

? A *query (document)*, to be used as query-by-example in requesting similar (published) documents.

? A *s-infolith* ('search infolith'), a parameterised version of a query.

? A *p-infolith* ('publication infolith'), a parameterised version of a published document.

? Optionally, for efficiency matters, a *summary* of one document, which is any (human readable) digest of a published document considered sufficient to issue a relevance judgement of its implied document.

Concerning the Agent Communication Language (ACL) used in I-Gaia, it is remarkable that, due to the very generic character of the protocols we are using, the inventory of communication primitives remains very short, mainly inspired the FIPA ACL [4] and assuming semantics related to that of the FIPA interaction protocols [5].

4. VALIDATION THROUGH TASKS

We now introduce the concept of a *validation task*, which is an application instance controlled in the data it receives ([15], 13.1). Each of these validation tasks will help us highlight some particular capabilities of I-Gaia. Also, we have artificially restricted the number of infocytes and their interactions to build two different *scenarios* used for running the tasks. We will deal with several querying S-Infocytes and several publishing M-Infocytes, but a single routing T-Infocyte, since the goal here will be to validate the underlying information retrieval and routing technology and not the problem of mediation in a network of interconnected TI's. In task-driven validation, this would have needed another, information-brokering validation task.

Both tasks described in this section have been created out of the Reuters corpus [6], which was not initially intended for such

[2] This should be considered as a proposal for a menagerie of living species at the model's initial stage. In the future, evolution within I-Gaia may suggest the possibility of including new species or specialising some of those mentioned above.

purposes, but targeted at a text classification task. It is a large collection of news wires from the well-known news agency, consisting of roughly some 10000 texts, labelled according to 118 predefined, overlapping categories. However, it is common to use only the 10 most populated categories to run experiments in text classification. We have selected the well-known ModApte split for the training and test sets, and we have represented every text as a vector count of 600 features (previously selected under a mutual information criterion but without any specific task-oriented optimisation.) The training records are always used to place the system in a certain state of operation (learning of

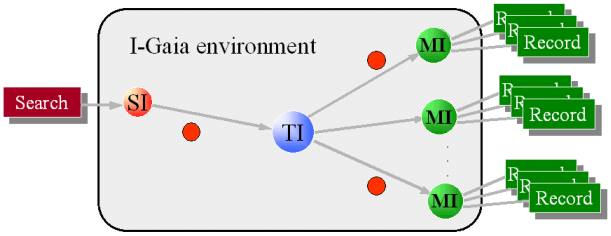

Figure 2. Information-pull scenario: single querying user, single TI and multiple record repositories. S-infoliths are sent.

variable parameters, corresponding to the 'past history' of the information exchange), so that it is then ready to handle the flow of new records drawn from the test set.

In this way, objective performance measurements can be carried out, and also compared with some other reference experiments using different set-up or technology. Performance will be evaluated using the well-known precision/recall curves, with special meanings or interpretations to be discussed below.

Finally, it is worth noting that, under "real-life" operation both pull and push tasks would be very intermingled in a dynamic scenario of simultaneously emitted queries and published records. Analysing such behaviour exceeds the scope of this paper.

4.1 An information-pull task and scenario

As an information-pull task we have decided to create an "ad-hoc" information retrieval one. In this task, before submitting any of the queries, the platform is seeded with some thousand "published" documents thus modelling the publishing behaviour of a population of users: the training set is divided in 100 subsets, each one of them associated to an M-Infocyte. This gives roughly a number of 65 records per repository (this number is higher than in the pull case, since it seems reasonable that a publishing entity will have a wider variety and amount of texts associated to it than queries can have a single user. See below.) We have depicted this scenario in Figure 2.

After this, an application that operates on behalf of the querying user is given queries that consist of one test document each, following a query-by-example strategy. For every query the task is to retrieve documents that are "similar" to it, and after retrieval documents are judged relevant if they share with the query any of their multiple categories.

The following assumptions about the information-pull task and scenario are made:

? Documents in the stream are published once and stay published.

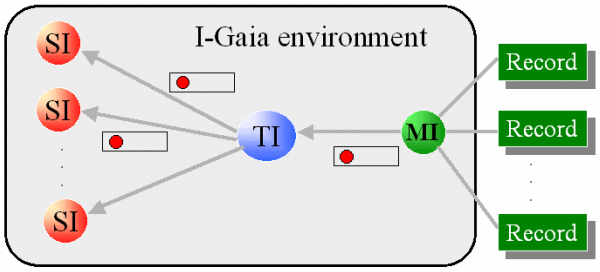

Figure 3. Information-push scenario: multiple potentially interested users, single TI and single publishing repository. P-infoliths are sent.

? The queries always appear later than the answers to the queries.

? Although the queries are also submitted once and for all, which suggests long-term standing queries, they are considered in fact short-term queries with duration of 1000 ms.

Thus, each document-posting event must be generated before each query submission event the posted document is relevant to. In Section 5 we will describe the specific measurements to be carried out on this task.

4.2 An information-push task and scenario

The task for information-push is reminiscent of information routing [1]: the purpose is to make published texts reach all the users possibly interested in them.

The training subcollection is considered to model the queries of the population of users and their interests and the test subcollection is used to validate the task. We assume that documents arrive in the chronological order indicated in the collection, and we assign the 6490 training records considered as queries to 1606 users (each represented by an S-Infocyte). The relevance judgements are also based in the sharing of categories between pushed documents and the queries characterising each user. Figure 3 shows such a scenario.

This behaviour reminds of the standard batch filtering task proposed in TREC-8 [8], the most important differences being the distribution of the relevance judgements and users and the fact that the routing technology can in I-Gaia contribute to decrease the performance of the application as measured by precision and recall (more about this later.)

To control the dynamics of the interactions for this task, the following assumptions are made about the different external human agents and resources in the task:

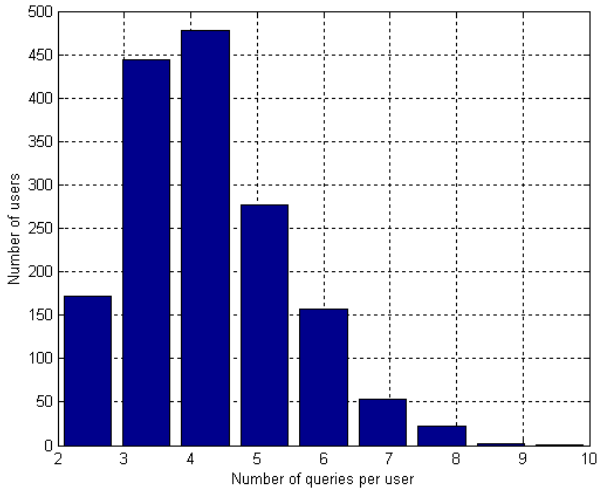

Figure 4. Distribution of texts as queries among users.

- User interests do not change over time. Their profiles remain static.
- Documents in the Reuters stream are published once and stay published.
- Documents are published later than the queries of the users that are interested in them.

This means that document publishing events must come later that query posting events of the users that are interested in such documents. In the following section we will describe the specific measurements to be carried out on this task.

4.3 Profiling

In record-space, the set of all possible records to be queried about or published infocytes that stand in for any users are regarded as *profiles*, sets of records that define them.

For the Push experiment, we have to assume that every querying user's profile can be somehow determined by his/her submitted queries up to a certain point before the experiment.

To "simulate" the process of profiling a number of users through the analysis of their queries, we have randomly split the training set into subsets of variable length, following a Poisson distribution with ?=2, (on average, 4 queries per user), to obtain a total of 1606 users in this scenario. These texts are interpreted as queries-by-example posted by that user, such that they define their interests. Figure 4 describes such distribution.

However, for the Pull scenario and task, each repository's profile will be determined on the basis of the set of documents its publishing user has previously emitted, using the same probabilistic models, but with 100 documents each approximately, which gives rather heterogeneous content for each profile.

4.4 Routing technology at Transfer-Infocytes

The basic routing mechanism implemented in Transfer-Infocytes is completely based on vector space models [1] and Estimation/Decision Theory [14]. Figure 5 depicts the s-infolith and p-infolith routing ends in a T-Infocyte.

Based on incoming s-infoliths (queries as vectors) $\underline{q}_{ij}$ arriving at infocyte *j* through a link with an infocyte *i*, a probability density function $f_i(\underline{x})$ is learned for each link (*a*), such that they model the source of incoming data, *a querying connection profile*, through that link, whatever is origin. Once learning accomplished, scoring values can be computed for any record $\underline{x}_k$ to be published and routing of it may follow *(b)*.

Something analogous can be said with respect to the learning of *publishing connection profiles* from the repositories' side: incoming p-infoliths (documents) $\underline{x}_k$ are used to estimate profiles $g_j(\underline{x})$ *(c)* an then forwarding of new queries is possible *(d)*, following a mechanism related to one above.

To decide routing either s- or p-infoliths, we estimate the matching score between that infolith and the data distribution or profile associated to each outgoing connection. Actually we are dealing with likelihood criterion–based rules in the form "route p-infolith $\underline{x}_k$ through connection ***i*** if $f_i(\underline{x}) ? t$", where ***t*** is a threshold value, to be discussed later, and similarly for s-infoliths.

Furthermore, since we are dealing with probabilistic routing, decisions can also be taken in a stochastic way, which breaks determinism in the system and may improve any optimisation process run under this scheme (this last aspect is not further investigated in the present work).

Although separately run in the experiments, processes shown in Figure 5, (a)-(d) will usually coexist in normal operation, leading to a feasible solution for dynamic scenarios.

The particular model assumed in this paper for the probability

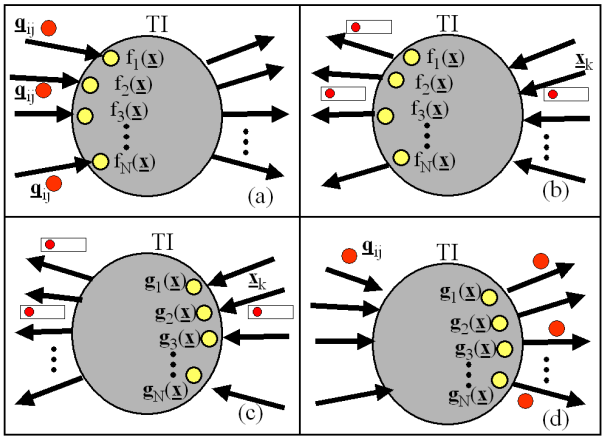

Figure 5: Estimation of probability distribution functions based on either queries (a) or published records (c). Routing of published records (b) and queries (d) can then proceed

density functions is a Gaussian distribution so that parameter estimation is simplified. In spite of this relatively simple model,

good performance is achieved since, as noted in [7], the problem of text processing is usually solved in a very high dimensional vector space, and simple models are flexible enough to cope with high dimensionality. Nevertheless, more complicate schemes (Gaussian mixtures, for instance) are also completely compatible with this scheme, and they may prove useful in modelling multimedia record repositories. Investigating them exceeds the scope of this paper, however.

In spite of its simplicity, the technology proposed here presents great advantages:

? These statistical models are well known and understood,

? Estimation of parameters is straightforward because explicit formulae exist for them,

? Only local information is used, thus extension to distributed environments is made easy,

? Adaptive versions are already available,

? Stochastic decisions are possible, thereby avoiding problems associated to deterministic decisions (greater advantages are expected in the multiple T-Infocyte case, not dealt with in this paper.)

Obviously, the above-described technology is not intended to be an original contribution (indeed other approaches are being simultaneously evaluated), but rather the means to empower the I-Gaia platform with the mechanisms to perform information-pull and -push experiments.

Under "real-life" operation both pull and push tasks would be co-occurring in a dynamic scenario of simultaneously emitted queries and published records. Analysing such behaviour also exceeds the scope of this paper.

5. PUSH AND PULL TASKS SET-UP AND EXPERIMENTAL RESULTS

5.1 The information-push experiment

In this scenario, the publishing user is single and will emit the whole set of document posting events associated to the test part of the task. Once the querying user profiles have been acquired by the system with the training part of the task corpus, the publishing phase (actual push) may begin.

Two sets of measurements are possible in this scenario, each one characterising the system from a different point of view:

? First of all, it is important to measure the average user satisfaction with respect to the information he/she is receiving (i.e.: Am I, as a S-Infocyte, receiving all the information I am interested in, and not anything else?).

? Secondly, it is also important to measure the average publication impact of the records (i.e.: Am I, a M-Infocyte, reaching all of the potentially interested users in every document, and no more?). We will call this second aspect *visibility (of a document)*.

A trade-off between both measurements exists and, since the system can only operate at one point, a reasonable balance has to be found (ideally, this is an aspect to be changed once a valid utility function for agent performance can be objectively

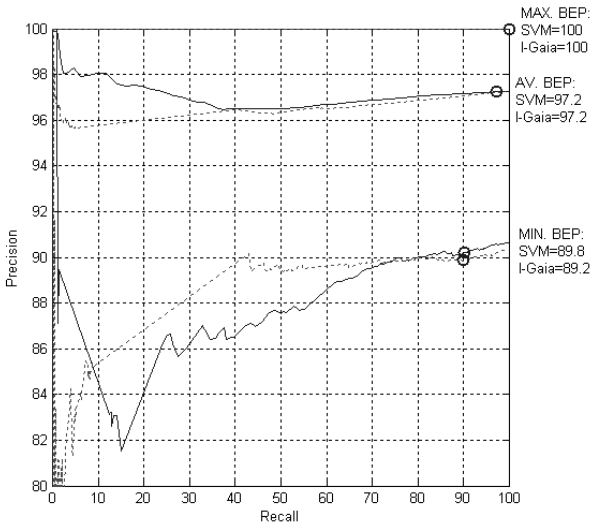

Figure 6: Technology comparison (pull case) between SVMs (dashed line) and profile-based measurements used at present in I-Gaia (continuous). Three precision/recall curves have been represented in every case: average, best and worst

optimised).

5.2 The information-pull experiment

In this information-pull case, both type of measurements, discussed in the previous section, also arise, reinforcing once more the symmetry in the pull-push scenario, already pointed out previously:

? The first measure is, again, the user satisfaction with respect to the information he/she is receiving as an answer to his/her query.

? The second one can be described as a *reachability*, which questions whether or not a query reaches all the potentially interesting record repositories (and none else). Again, flooding the system with record announcements is not an appropriate solution since, although not accounted for in this case, resource economy and accuracy criteria must be taken into account.

5.3 Reference experiment

First of all, to evaluate the average performance achieved for this tasks under I-Gaia we have defined a "reference experiment" conducted in ideal conditions (centralised solution, good pattern processing technology, etc.), and solved it outside I-Gaia. We have used Support Vector Machines technology [3] to build an "optimal"

[3] We have resorted to standard software to ease further comparisons: the SVMlight implementation by T. Joachims, available at http://ais.gmd.de/~thorsten/svm_light.

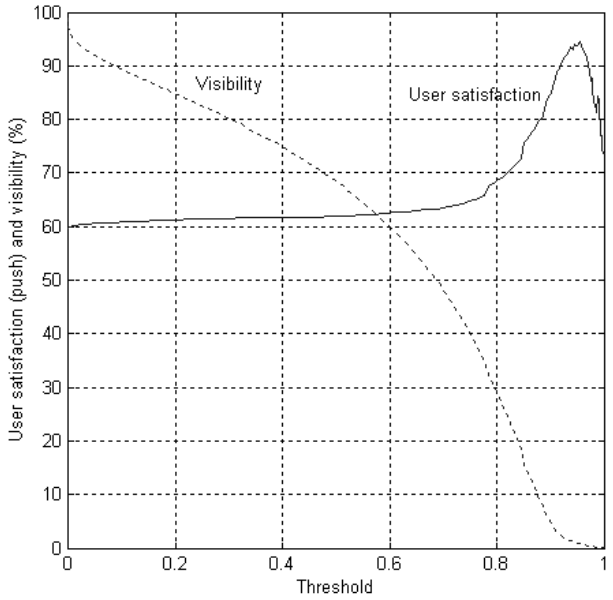

Figure 7: Trade-off between user satisfaction and visibility to users of published documents in the push scenario.

Transfer-Infocyte, since they have proved to be very efficient at solving this task [9][10] [12] [13], which will give us a reference value of the expected performance, and the difference will be the price to pay for operating with local information in I-Gaia. When I-Gaia gets distributed the influence of this decision will be analysable in terms of performance as well.

The results of this technology comparison can be observed in Figure 6, illustrated for the information-pull task, although analogous results have been obtained in the push task, since the same mechanism is used in both.

We notice that both technologies provide roughly the same performance:

? The maximum breakeven-point (BEP), -attained in the best-conditioned case, is 100% in both cases,

? The worst case yields BEP = 89.8 for the perfectly informed case and BEP = 89.2 for the I-Gaia case,

? The average BEP values also being almost identical (97.2).

This reinforces the results presented in [7], and validates the technological choices of section 4.4 for our T-Infocytes.

5.4 Performance evaluation

To give an overall performance of the system we have decided to evaluate the performance of the central T-Infocyte with regard to precision and recall for each task. As a working point has to be defined for every direction of information flow (pull and push) at the T-Infocyte, we have opted for representing performance parameters as a function of the decision threshold used in every case (we have only depicted the pull scenario here, though). A relevance decision in the range (0,1) with a threshold can be obtained using many alternative technologies, so this score represents a way of unifying comparison between results in very different scenarios. The curves corresponding to the experiment performed in I-Gaia have been represented in Figures 7 and 8 for, respectively, the push and pull experiments.

In Figure 7 the trade-off between user's satisfaction and the publishing user's visibility, two opposite goals, can be observed: a publishing repository wants to reach all potentially interested users in its records (threshold around zero), while a user wants to receive only those texts relevant for him (threshold around 0.9, in this case). In this case, "optimal" publishing would imply flooding the network with p-infoliths (recall once again that no penalty is taxing information flow), while high user satisfaction would imply low recall, i.e., many relevant documents would never be presented to that user.

Presently, the adequate threshold value needs to be tuned and hardwired before operation, but in future work we intend to adaptively estimate it to maximise a utility function (for instance using a genetic algorithm), still to be defined.

In figure 8 we have depicted the analogous measurements as described before, but from the information-pull point of view. In this case, the more familiar trade-off is evident, querying ever more potentially interesting repositories has a price to pay: many irrelevant documents will be retrieved.

6. DISCUSSION

We have introduced and presented some experiments in yet another information-processing multiagent system, I-Gaia. With the two separate task scenarios we have devised I-Gaia has proven successful at running non-overlapping information-pull and –push tasks. We remain confident that, if the host's computational power allows it, I-Gaia will be able to run both tasks simultaneously with comparable success. Given the sparseness and lack of specialisation of the entities inhabiting I-Gaia, this is

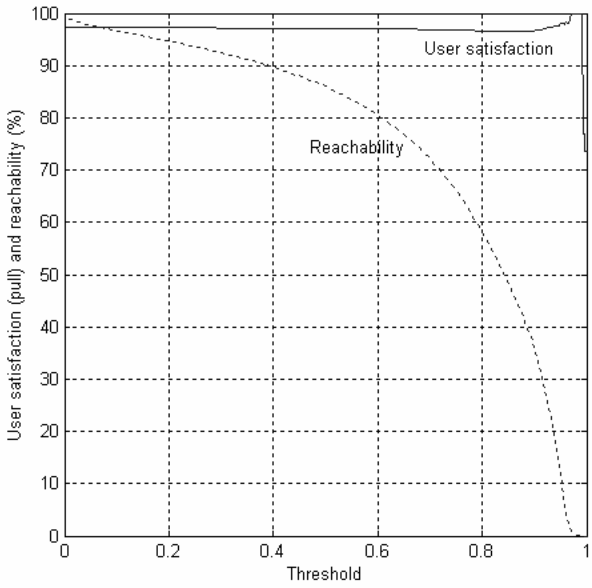

Figure 8: Trade-off between user satisfaction and reachability of repositories by queries in the pull scenario.

encouraging, because it promises and ability to successfully model other tasks needing richer functionality.

We fear that the (minimal) brokering scheme based upon a single intermediary will not scale well for millions, even for thousands of users. This is the next step to validate: whether multiple intermediaries will collaborate or struggle for servicing both types of users, and whether either mode of operation will positively affect the performance seen by users.

Also, the ability of ÞGaia to host information-push and –pull processes bids well for developing an encompassing model where all information retrieval related tasks such as filtering, routing, recommending and retrieval proper can coexist.

Adaptive models of information processing seem to be a must, too, if deployment of applications based on this platform is to be undertaken. More interestingly perhaps, adaptability should pivot crucially around the utility function at T-Infocytes we have been mentioning: a measure of objectively defined performance, which may possibly not be feasible due to the warring compromises in T-Infocytes regarding querying and publishing users.

Finally, the measures of performance for parallel push and pull as proposed in this paper need to be compared to other, better understood ones.

Compared to other platforms [17][18], our agents – as befits the light-weight, scalable approach of the DIET project in which our work fits – are more reactive and less capable of acting and planning to extraneous events, more capable of adapting to changing workloads and communication demands. On the other hand, their communication primitives and choice of performatives are rather limited and their ontology poor, even lacking reflection capabilities. The idea is that somehow, the whole community of agents should be adaptable to extraneous events, the and develop behaviours which in previous systems where expected of individual agents, an idea to be pursued in future work.

7. ACKNOWLEDGMENTS

Part of this work was developed under contract IST-1999-10088 with the European Commission within the Future and Emerging Technologies (FET) initiative of the IST Program. We would also like to acknowledge other members of the DIET consortium: Intelligent Systems Laboratory, BTexact Technologies, and the Intelligent and Simulation Systems, DFKI, for their suggestions and comments.

We appreciate the advice of the anonymous reviewers for improving this paper.

8. REFERENCES

[1] Baeza-Yates, R, and Ribeiro-Nieto, B. *Modern Information Retrieval*, Addison-Wesley/ACM Press, New York (USA), 1999

[2] Belkin, N.J., and Croft, W.B. “Information filtering and Information Retrieval: Two Sides of the Same Coin?”, Comm. Of the ACM, Vol. 35, num 12. Dec, 1992

[3] European Commission (IST Future and Emerging Technologies), Universal Information Ecosystems Proactive Initiative, http://www.cordis.lu/fetuie.htm, 1999

[4] FIPA, Foundation for Intelligent Physical Agents. *FIPA Communicative Act Library Specification*, Geneva, 2000. Visit: http://www.fipa.org

[5] FIPA, Foundation for Intelligent Physical Agents. *FIPA Interaction Protocol Library Specification*,. Geneva, 2000. Visit http://www.fipa.org

[6] http://www.research.att.com/~lewis/reuters21578.html

[7] Han, E.H, and Karypis, G., “Centroid-Based Document Classification: Analysis and Experimental results”. Technical Report TR-00-017, Department of Computer Science, University of Minnesota, Minneapolis, 2000. Available on the WWW at URL http://www.cs.umn.edu/karypis

[8] Hull, D.A., and Robertson, S. *The TREC-8 Filtering Track Final Report*, NIST Special Publication 500-246: Proceedings of the Eighth Text Retrieval Conference (TREC-8), 2000. Electronic version available at http://www.nist.gov/pubs/trec8/t8_proceedings.html

[9] Joachims. T., “Text categorization with support vector machines: Learning with many relevant features”. In C. Nédellec and C. Rouveirol, editors, Proceedings of the European Conference on Machine Learning, pages 137-142, Berlin, 1998. Springer.

[10] Joachims, T., “Transductive inference for text classification using support vector machines”. In International Conference on Machine Learning (ICML), Bled, Slovenia, 1999.

[11] Marrow, P., et al., “Agents in Decentralised Information Ecosystems: the DIET Approach”, Proceedings of the AISB’01 Symposium on Information Agents for Electronic Commerce, York, UK, 2001, pp.109-117

[12] Platt, J. “Fast training of support vector machines using sequential minimal optimization”, in B. Schölkopf, C.J.C. Burges, and A.J. Smola, editors, Advances in Kernel Methods --- Support Vector Learning, pages 185-208, Cambridge, MA, 1999. MIT Press.

[13] Tong, S., and Koller, D. “Support vector machine active learning with applications to text classification”. In Proceedings of the Seventeenth International Conference on Machine Learning, 2000.

[14] Van Trees, H.L.: Detection, Estimation, and Modulation Theory (vol. 1). New York: Wiley; 1968.

[15] Varile G.B, and Zampolli, A., managing eds. *Survey of the state of the art in human language technology*, Cambridge [etc.]: Cambridge University Press; Pisa: Giardini , 1997

[16] Wondergem, B.C.M., van Bommel, P., van der Weide, T.P. "Cumulative Duality in Designing Information Brokers", CSI-R9808, Computing Science Institute, Catholic University Nijmegen (The Netherlands), March 1998.

[17] Decker, K., et al. "Designing Behaviours for Information Agents", in Proc. of the 1st International Conference on Autonomous Agents (Agents'97), University of Delaware, Delaware: 1997

[18] Bayardo et al. "InfoSleuth: Agent-Based Semantic Integration of Information in Open and Dynamic Environments", Proceedings of the ACM SIGMOD International Conference on Management of Data, 1997, pp. 195-206