K-Trek: P2P Knowledge Management in Wireless Mobile Networks

Paolo Busetta[2], Chiara Ghidini[2], Paolo Bouquet[1,2], and Matteo Bonifacio[1,2]

[1] University of Trento – Trento, Italy
{bonifacio,bouquet}@dit.unitn.it
[2] ITC-irst – Povo, Trento, Italy
{busetta,ghidini}@itc.it

Abstract. We introduce a notion of context awareness, derived from work on so-called *distributed knowledge management*, based on data accumulated and categorized by users during an extended period of time and matched against data proposed by local sources of information. This notion is exploited by an architecture, called *K-Trek*, that enables mobile users to *travel across knowledge* distributed over a large geographical area (ranging from large public buildings to a national park). Our aim is providing, distributing, and enriching the environment with location-sensitive information for use by applications on board of mobile and static devices.

1 Introduction

In this paper, we explore an architecture, called *K-Trek*, that supports a form of *context-aware computing* in mobile and wireless networks. K-Trek enables mobile users to *travel across knowledge* distributed over a large geographical area (ranging from large public buildings to a national park). This is obtained by providing, distributing, and enriching the environment with location-sensitive information for use by applications on board of mobile and static devices.

Context-aware computing is an area of active research at the very heart of *pervasive computing* and *ambient intelligence* [10], even if a clear focus has yet to emerge (see for instance the recent [1]). Context-awareness is usually defined as sensitivity to the user's state, the environment where she currently is, and the current physical environment [12]. Distinguishing features of our approach with respect to the known literature are:

- no long-range, permanent wireless networks or sensors of any kind are involved. Instead, we "augment" the environment, as well as mobile devices, with very low cost, easily available hardware for wireless, short range communication. As a consequence the nodes of the network are not guaranteed to be able to connect (directly or indirectly) to all other nodes all the time. Bluetooth [4] is our reference technology, but the architecture can be easily adapted to future standards as they will emerge;

– our definition of *context*, derived by applying the formal framework described in [11] to distributed knowledge management (DKM) issues, is based on data accumulated and categorized by each user during an extended period of time. Information given to and left by users during their movements is filtered or annotated by means of the contexts on board of mobile and static devices. This process effectively subsumes traditional feature-based selections based on user preferences or profiling;
– applications on board of static as well as mobile devices can exploit the knowledge accumulated and categorized in the devices as well as users they get in contact with for exchanging and transporting information to agents they cannot directly reach.

As shown in this paper, these characteristic features justify a knowledge-driven approach both to the definition of the notion of context in which a particular, possibly mobile, node finds itself in, and to the development of techniques for transporting information across the network. The principal communication mechanism in a network which does not rely on any permanent, long-range, connection is necessarily based on peer-to-peer exchange of information in a completely decentralized and distributed manner. Thus, one of the main goals of the work on K-trek is to use and develop novel peer-to-peer techniques for knowledge management able to allow mobile users to travel across knowledge distributed over a large geographical area. While a rich literature exist in the area of peer-to-peer knowledge management, e.g. [2, 8, 13], not much work exists in applying peer-to-peer knowledge management techniques to mobile and wireless networks. Hence the main novelty of the K-Trek approach is the development of an architecture and of peer-to-peer techniques for identifying the context of applications and for propagating information.

Context-awareness will allow applications to get only information relevant to them at that particular time at that particular location – which is to say, a mobile device is context aware as commonly meant [12]. Information propagation is conceived in such a way that messages can delivered even in an environment where no routing is possible, since user movements are not predictable, and only stochastic guarantees are given concerning the actual delivery of a message, its latency, and the geographical area of distribution.

Previous work on K-Trek aimed at understanding its feasibility, behaviour, and performance via simulation [9]. Current work has been focused on the refinement of the general architecture and on designing the context-aware capabilities of K-Trek; this is what we present in this paper. Future work is devoted to the problem of knowledge driven information propagation algorithms, and to the implementation and careful evaluation of the P2P and KM techniques necessary to make K-Trek a reality; here, we describe these aspect of the system only in generic terms, and outline a vision of the potentialities of the K-trek approach and architecture in the P2PKM area.

In this following, we call *agent* any distributed, autonomous, communicating application component [14]. Agents in K-Trek adopts some typical peer-to-peer patterns and techniques for reciprocal discovery; in particular, small peer-to-peer networks are formed on-the-fly and enable localized, context-aware interactions. Users movements are exploited to provide message transport in the larger environment, in a way that reminds query propagation on some well-known peer-to-peer networks. This mechanism is effectively a particular form of *ad hoc* wide area networking that does not need any permanent long-distance communication infrastructure.

Next section motivates and provides an overview of our approach. Section 3 describes the architecture of K-Trek. Context awareness in K-Trek is discussed in more detail in Section 4. We present, in Section 5, a few application scenarios. We conclude with some final remarks (Section 6).

2 Motivating the K-Trek approach

There are several reasons that motivate our approach, and they will be made even clearer in the remaining of the paper. In brief, the justification for a network which allows only for short range communication is the fact that in many application scenarios long-range, permanent wireless networks or sensors are not a realistic or optimal solution, for various reasons. In certain scenarios, such as trade fairs, or congresses where users and infrastructures change their configuration very often, low cost, easily available hardware for short-range wireless communication allow to set up flexible and adaptable networks, easily supporting location-awareness without additional sensors (as explained below) and at the same time cutting down costs. In other scenarios, such as natural parks, mountains, and (remote) environmental landscapes, the low cost hardware for wireless short range communication is the only viable choice. The same for certain industrial scenarios such as train workshops, where engineers are equipped with PDA-like user device and the workshop is enriched with sensors that can communicate with the engineers to guide them during their daily work and to collect data from the engineers. Here, the motivations are not given by geographical or economical reasons but by environmental specificities, such as the arrival of a train carriage: the high power electrical engine of the train interferes with standard radio technology (e.g., WiFi or cell phones), disrupting normal connectivity – alternative mechanisms, supporting interruptions and limited network availability, have to be found.

Focusing on contexts, our choice is to use this concept in two distinct senses: *context as location* and *context as perspective*, which will be better explained in section 4. The important thing to be noted here is that the particular type of network we focus on, where the movements of the mobile devices are not predictable, forces us to discard the possibility of using geographic coordinates to define the "location" where a particular device finds itself in. Thus location will be defined here as the co-presence of other K-Trek devices; from a knowledge management perspective, this can be usefully exploited in many scenarios. Moreover, in order to reach the goal to enable K-Trek devices to exchange only contextually relevant information, we propose to define the context of an application taking into account not only the devices in geographical proximity, but also the perspective (that is the particular focus or viewpoint) of the devices themselves. This following the usage of context in knowledge representation [11, 3] and knowledge management[6, 5].

Focusing on information propagation, we already said that in a K-Trek environment no routing is possible, since user movements are not predictable, and only stochastic guarantees are given concerning the actual delivery of a message, its latency, and the geographical area of distribution. Therefore we propose to utilize users' movements for message transport. It is clear that in order not to saturate the network with useless or redundant messages clever strategies need to be developed in order to choose an "appropriate" carrier for the messages. We propose to enable K-Trek devices to exploit the knowledge accumulated and categorized, for instance about the users they get in contact with, for exchanging and transporting information to agents they cannot directly reach.

3 K-Trek: an overview

K-Trek is an infrastructure based on three main types of device, called K-Trek devices (see Figure 1):

K-Beacon: a static device (such as an embedded system with integrated Bluetooth board) which stores contextual information about a specific location (i.e., the location where it is placed), and can interact in various forms with other K-Trek devices;

K-Voyager: any mobile device – such as a PDA or a last generation mobile phone – with any number of K-Trek applications on board;

K-Plug: any device with a standard network interface that acts as a gateway between K-Trek devices and back-end servers.

K-Trek devices can be connected to each others in two main types of networks:

- *K-Trek micro networks*, i.e. on-the-fly networks that connect a limited number of K-Trek devices in a very small geographical area, and
- *K-Trek Wide Area Network (K-wan)*, i.e. a wide-area, message-based, asynchronous network, where mobile K-Trek devices may be used as temporary bridges between disconnected devices on the same K-wan.

In the next two sections, we describe the functioning of the two types of networks.

3.1 Micro networking with localized resources

The main feature of K-Trek is the ability of setting up "micro networks" on-the-fly, i.e. networks that cover a very small geographical area (no more than a few ten meters) with a limited number of devices and limited bandwidth, without the need for dedicated, static equipment (wires, routers, access points, or other paraphernalia)[3].

[3] As mentioned in the introduction, our reference technology is Bluetooth [4], because it is suitable to very low-cost, low-power, wireless devices, and it is commonly built

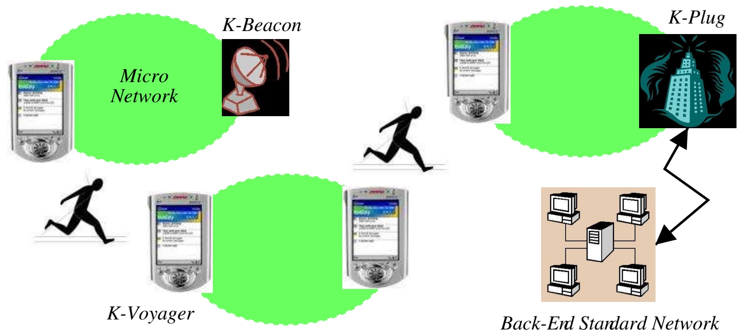

Fig. 1. K-Trek: main components

Communication among K-Trek devices in a micro network is in charge of special light-weight message handling agents. Their tasks include the most basic peer-to-peer interaction, i.e. discovery. To this end, they periodically broadcast *announcements*. For instance, a K-Beacon announcement contains the K-Beacon's contact information and a set of short messages, sent by local application agents and directed to the agents on passing K-Voyagers. The processing of this announcement is discussed later, in Sec. 4.

This discovery-based approach for finding out local resources, inspired by peer-to-peer systems contrasts with location-aware systems based on geographical coordinates, commonly adopted with mobile phones and other wireless networks, for various reasons. First, no location sensor such as a Global Positioning System (GPS) is needed. Second, since there are no coordinates, there is no need for geo-referencing information to be delivered to users, as it is commonly required when central services are involved (typically with mobile phones), or when local applications need to retrieve data already on board of the user's mobile device or to access a centralized directory.

Notice that, in general, communication happens while the user is moving, thus the time of contact between two devices can be short. This, and the limitations on bandwidth, impose strong constraints on the protocols, concerning in particular the frequency of announcements and the amount of data exchanged. However, these constraints may be loosened after a careful study of the characteristics of a specific scenario, which may reveal peculiar user patterns (possibly induced, e.g. by some HCI mechanism such as a sound that invites the user to look at the screen and thus to slow down), or may impose a specific behavior (e.g., stopping whenever the K-Voyager signals that it is in contact with a K-Beacon).

---

into many last-generation mobile phones and PDAs. What we call a micro network is commonly referred to as a *pico network* in Bluetooth; we purposely differentiate our terminology, which refers to a high-level architecture. The definition of K-Trek device is independent from Bluetooth.

3.2 K-wan: a wide-area asynchronous network

The second type of network is what we call K-Trek Wide Area Network (K-wan), designed for certain types of knowledge management applications. A K-wan is a wide-area, message-based, asynchronous network, where messages may be delivered long after being posted, and only stochastic guarantees are given concerning their actual delivery, latency, and the geographical area of distribution. As discussed later, a K-wan exploits the users' movements for message transport, thus no special equipment is required other than what is required to set up micro-networks (that is, Bluetooth boards). A K-wan may remind of a *partial mesh* network, where each node is connected to each other node either directly or by means of other intermediate ones that act as routers. However, in a K-wan no routing is possible, since user movements are not predictable, thus messages are broadcasted, adopting a very different propagation strategy than in mesh networks. This strategy may be improved in future, if some intelligence (for instance, user profiling or other machine learning techniques) is able to predict the future user movements, or in specific domains where user movements are well known (e.g., devices on board of public transport vehicles).

Some micro networking mechanisms are implemented by the message handling agents to support transport within a K-wan. One of them follows the K-Beacon discovery by K-Voyagers mentioned in Sec. 3.1 above, and consists of two complementary actions. The first is downloading any message for the K-Beacon contained in a dedicated K-wan buffer on board of the K-Voyager; in other words, a K-Voyager delivers, to the K-Beacons it gets in contact with, anything for them that was picked up during its trip. Conversely, the second action is uploading on the K-Voyager messages from the K-Beacon directed to agents running remotely.

The second mechanism needed by K-Wan is applied between K-Voyagers. The announcement mechanism of Sec. 3.1 enables K-Voyagers to discover each other; this is followed by the exchange of the contents of their K-wan buffers, in a truly peer-to-peer fashion. At the end of this process, any message addressed to either of the two K-Voyagers is delivered to the appropriate agent and discarded from both buffers (since it reached its destination), while all others are duplicated.

This buffer content exchange happens whenever two users carrying K-Trek devices get close by, without any human involvement. This effectively implies that messages spread around the geographical area covered by moving K-Trek users as a sort of benign – but highly infectious – virus. A number of mechanisms – such as setting expiration dates on messages, maintaining lists of those already delivered, managing buffer overflows – are used to keep things under control. However, a number of questions arise about this transport technique, e.g. what buffer size is required, what is the probability of reaching the destination, which geographical area is covered; the answers are affected by many factors, the most important being the pattern of movement of users. The last section of this paper shows some studies on the suitability of a K-wan to specific scenarios.

The last major micro networking mechanism used by a K-wan involves the third type of K-Trek device, K-Plugs. A K-Plug can be any device (e.g., a personal computer or a Bluetooth *access point*) with a standard network interface that acts as a gateway between devices on a K-wan and back-end servers. To this end, all K-Plugs provide access to a single, centralized mailbox service. When a K-Voyager gets within the range of a K-Plug, a set of peer-to-peer protocols similar to those presented above are used to deposit messages for agents on back-end systems, and to pick up messages addressed to the K-Voyager (or its user) and for other K-Trek devices; the first are immediately delivered to their destination agents, while the others are deposited in the K-wan buffer.

We expect that more than one K-Plug are part of a K-wan. Ideally, they should be located in places where, sooner or later, most if not all users pass by. [4] In situations where the paths followed by users can be predicted, messages for a K-Beacon `K` are distributed only by the K-Plugs along the paths that touch `K`. Since message duplications are likely while delivery cannot be guaranteed, care is taken in the mailbox administration, for instance by making sure that messages for K-Beacons are not removed until expired or requested by their senders (possibly after an application-level handshake). [5]

A K-wan is particularly suited to cases where low-power embedded systems distributed on a large territory need to perform occasional exchanges of non-critical data (e.g., collecting data from sensors detecting animal or tourist movements in a national park). These scenarios currently require either expensive links (such as microwaves or satellite), or people physically going to each device for uploading and downloading data via floppy disks or other media. As shown in the examples in the concluding section of this paper, a careful analysis can predict the performance of a K-wan with some precision. To this end, we have developed analysis tools that can be used to set up a K-wan so that any required level of performance (e.g., maximum time for delivery) is achieved, thus making a K-wan appealing for a large number of application scenarios.

4 Context-sensitive mobile applications in K-Trek

Our first objective is to enable the exchange of contextually relevant information among the K-Trek devices temporarily connected in a micro-network. Context here is used in two distinct senses:

Context as location. This is the more traditional sense of context in context-aware applications. However, K-Trek supports a particular form of location-awareness, where the "location" is determined not by geographic coordinates

[4] For this reason, and to reduce the amount of circulating messages, a K-Trek administrator may configure K-Plugs so that K-Voyagers can pick up messages for themselves and for K-Beacons, but not for other K-Voyagers.

[5] It should be noted that K-wan is a potential source of security problems because of its virus-like message transport mechanism. For instance, if no care is taken, a denial-of-service attack could be easily performed by somebody generating many apparently innocent messages with very long expiration dates. Thus, the K-wan buffer management is a particularly sensitive issue.

but by the co-presence of other K-Trek devices (e.g., a meeting can happen anywhere as long as all the required participants are present)

Context as perspective. Context here is used in the same sense of DKM, and refers to the idea that different people (or groups of people) produce heterogeneous and partial views (called *contexts*) on the information available within an organization, each from their own perspective, and that these views – far from being an obstacle to management and coordinated action – are a potential source of innovation and knowledge creation, if suitably managed.[6]. The idea here is that peers can encode a representation of a domain, or of their neighborhood, which depends on their collection of interests, beliefs, goals, and so on. For example, a K-Voyager specialized and focused on alpine botanic species will classify the information provided by K-Beacons of a natural park concerning the Alpine pine under the scientific name "Pinus cembra", while a K-Voyager only interested in broad descriptions of the alpine trees will classify the same information under "alpine trees". Current work on DKM has adopted this notion of context for developing a peer-to-peer system called KEx (*Knowledge Exchange*) [5], which embodies the functionality for defining one or more local views (contexts) for each so-called K-Peer, and automatically discovering mappings among contexts using a complex algorithm for semantic matching [7]. In KEx, contexts are graphs of concepts that are used to index or annotate document bases, data bases, and Web services in future. The algorithm in [7] works with propositional contexts. Current work is devoted to extend the algorithm for semantic matching to Description Logic based languages. K-Trek supports a use of context similar to the conceptual graphs mentioned for KEx. In the simplest case, a context is nothing more than a set of labels indicating features or user preferences. A context is enriched off-line with linguistic information from thesauruses or data bases; this phase is what we call *semantic explicitation* and is described in detail in [7]. The full context mapping of KEx is replaced by the semantic comparison of selected context nodes described below; this simpler process can be supported within the limitations imposed by moving users.

Whenever a micro-network is established, K-Voyagers discover whatever resources are available on other K-Trek devices, attempt to match the contextual labels attached to those resources against those contained in the contexts they have on board, and act consequently – e.g., they may report on the findings to their users. Context-sensitivity is then achieved by "augmenting" the environment with K-Beacons, with their own contexts on board, representing or annotating local information such as data generated by local sources (typically on embedded systems) or information left by other mobile devices.

Application agents running on a K-Voyager are associated to one or more contexts. By operating on the K-Voyager's GUI, the user decides which applications, and which contexts, to keep active. This means that user gets *only information*

[6] As mentioned in the introduction, this definition of context is a direct derivation of the work on contextual reasoning by Giunchiglia and his group. The interested reader can refer to [3, 11]

relevant to her at that particular time at that particular location – which is to say, a K-Voyager is context-aware as commonly meant [12]. Since the interaction is two-way, also data flowing from K-Voyagers to K-Beacons can be annotated with contextual information (i.e., the labels and their associated formulas obtained by the semantic explicitation), so agents on a static device can get additional information on mobile users and possibly select only the information that is of their interest.

User contexts can be edited by users; this is a typical off-line process, better performed on a more convenient platform than a mobile device (e.g., a desktop computer), and is followed by the semantic explicitation phase. Similarly, contexts on board of K-Beacons are typically edited off-line, semantically explicitated, and downloaded by a system configurator. In the future, it is foreseeable that contexts may be acquired semi-automatically by K-Trek devices themselves, e.g. in a mixed-initiative process where some of the information collected by a K-Voyager during a trip is suggested to the user for addition to her contexts.

The interactions between K-Trek devices follow a common pattern; we illustrate here the case of a K-Voyager in the range of a K-Beacon. When the message handling agent on board of the K-Voyager receives a K-Beacon announcement, it performs a discrimination of its content, then a first type of context-sensitive processing. Application messages addressed to a remote system or to a different K-Voyager are stored in the K-wan buffer; their processing has been discussed earlier. The others (i.e., those addressed to either anybody or specifically to this K-Voyager) are *filtered* against the active user contexts before delivering to their destination agents. Filtering substitutes the full context mapping of KEx. It consists of identifying potential target concepts within the active contexts by searching for all the senses in the formulas associated to the contextual labels attached to the application messages (and transferred with them), followed by a semantic comparison phase, that is a phase in which the mappings between contexts are built.[7] If the result is satisfactory – that is, one or more labels attached to a message are semantically compatible with the active context of its destination agent – the message is delivered.

Typically, application messages are further application-specific announcements or local information to be shown to the user. Apart from those described in Sec. 3.2, further interactions between K-Beacon and K-Voyager are driven by the application agents, for instance to retrieve or deposit data or obtain services from K-Beacon agents. Since a K-Voyager may fall within reach of multiple K-Beacons, application agents must be able to handle simultaneous interactions. As mentioned above, application messages are also annotated with contexual labels (and their associated formulas); these labels are selected by the sending agents either implicitly (e.g., when they can be inferred by the current state) or after explicit request to the user.

[7] It is important to note here that whatever is needed by the semantic comparison is attached to the contexts by the semantic explicitation phase, so that it is not necessary to have thesauruses like WordNet on board of any K-Trek device. For more details on the explicitation and comparison phase see[5]

5 Distributed Knowledge Management applications on K-Trek

Most things that one can imagine doing in the physical world by putting a sign, leaving a mark, depositing a form in a mailbox, attaching a "post-it" card, and so on, can be done electronically with K-Trek, with the exception of those actions that require knowledge of the exact location and direction of the user (e.g., direction-giving relative to the user position, such as "move for 20 meters on your left and you will see the Colosseum", cannot be supported without additional sensors). Looking at K-Trek from a broader knowledge management perspective, its architecture is suitable to situations in which:

- the physical environment is populated by objects whose value can be increased by either delivering to, or collecting information from, other objects or users;
- linking these "informative" objects by means of an information network based on long-distance wireless connections is unfeasible, because of costs or environmental constraints;
- mobile actors in the environment need to locally exchange information either with informative objects or with other actors;
- mobile actors move across the environment along paths that, statistically, connect all the informative objects;
- an environment administrator has an interest in enhancing the environment through the provision of infrastructural services;
- there may be external actors that have an interest in "owning" the informative processes related to one or more objects.

A first example of potential K-Trek enabled environment is natural parks and, in general, geographically dispersed entertainment environments such as archaeological sites. Parks are populated by objects such as natural attractions, routes or historical sites whose value can be enhanced if able to exchange information with users, other objects, the administrator, or the "owner" of the site (an entity that has an interest in updating and collecting the information that belong to the site). For example, a historical site may receive information: from a school of architecture in order to update its description; from a visitor that wants to leave a message to those that will visit the site in future ("virtual post-it"); and, from a member of the maintenance staff that has periodically to asses its status. Conversely, the site can provide: architectural information to a visitor whose contexts show an interest in architecture; maintenance information to inspectors, previously deposited by members of the maintenance staff; and, information about number of visits, type of users and the kind of information they deposit on the site to the park administrator. Visitors and maintainers unintentionally provide the "lazy" communication channel needed to ensure information delivery, update, and collection by K-wan.

Another scenario involves field management activities of geographically distributed industrial settings. Relevant objects are industrial sites or components (power stations, junction boxes, and so on) that generate information about their status and collect information about those maintenance activities that must be performed and assessed in site. Here, since the certainty of information delivery and collection is quite critical, maintenance visits are intentionally scheduled not just as a function of each maintenance task, but also to enable the circulation of information across the overall system. For example, maintainer A that has to visit and asses the status of site 1, has a route that passes in front of site 2 whose maintenance is under the responsibility of the maintainer B. In such case, A deposits his visit report on site 1 and automatically collects the visit report of B done on site 2. The latter will be delivered to the environment administrator whose task is to monitor the overall system.

This scenario provides an example on how K-wan can handle certain levels of information criticality when the administrator is able to exploit the value of predictable "visit paths" in terms of connections that will happen with a known frequency and with a known level of reliability. Another good example is represented, in a urban environment, by mailmen that, in addition to their usual task of mail delivery, might deliver data to and collects updates from those K-Beacons that are positioned on their typical routes.

In summary, the peculiarities of K-wan make it useful for specific – but not at all uncommon – classes of applications, the most important being the non-real time ("lazy") monitoring and control of large territories. For instance, the application domains mentioned above would benefit from the collection of statistics (e.g., on tourists' preferences, on the state of natural resources), of summaries of notes or forms left locally by passing users (e.g., tourist satisfaction forms), of data coming from embedded processors (e.g., usage statistics of industrial equipment). K-wan supports this kind of data collection at a very low cost with respect to more traditional communication mechanisms. Conversely, the lazy distribution of data or configuration parameters (even software) from a centralized place to users or K-Beacons is very cheap with K-Wan.

It is worth to stress again that the annotation of messages with information taken from the originating agent's contexts helps in performing typical knowledge management tasks, varying from the ability to support communities of mobile users to classical data mining processes such as understanding tourists' interests, identifying patterns of visit per user category, and so on.

6 Conclusions

We have introduced an architecture, called K-Trek, that provides both context-sensitive information and "lazy" distribution of information over a wide area network, at very low cost. The concept of context and context awareness, as presented here, derive from work in knowledge management. We have presented a few application domains where K-Trek is applicable. Future work will be directed at practical experimentation in real world cases and at the development of knowledge driven information propagation mechanisms.

References

1. G. D. Abowd, M. Ebling, G. Hunt, H. Lei, and H.W. Gellersen. Context-aware computing. *IEEE Pervasive Computing*, 1(3), 2002.
2. M. Arumugam, A. Sheth, and I. B. Arpinar. Towards p2p semantic web: A distributed environment for sharing semantic knowledge on the web. In *WWW2002 Workshop on Real World RDF and Semantic Web Applications*, May 7 2002.
3. M. Benerecetti, P. Bouquet, and C. Ghidini. Contextual Reasoning Distilled. *Journal of Theoretical and Experimental Artificial Intelligence*, 12(3):279–305, July 2000.
4. Inc. Bluetooth SIG. The Official Bluetooth Wireless Info Site, 2002. `http://www.bluetooth.com/`.
5. M. Bonifacio, P. Bouquet, G. Mameli, and M. Nori. KEx: a peer-to-peer solution for Distributed Knowledge Management. In D. Karagiannis and U. Reimer, editors, *Fourth International Conference on Practical Aspects of Knowledge Management (PAKM-2002)*, Vienna (Austria), 2002.
6. M. Bonifacio, P. Bouquet, and P. Traverso. Enabling distributed knowledge management. managerial and technological implications. *Novatica and Informatik/Informatique*, III(1), 2002.
7. P. Bouquet, B. Magnini, L. Serafini, and S. Zanobini. A SAT-based algorithm for context matching. In P. Blackburn, C. Ghidini, and R. Turner, editors, *Proc. of Fourth International and Interdisciplinary Conference on Modeling and Using Context (Context'03)*, LNAI. Springer Verlag, 2003. To appear.
8. J. Broekstra, M. Ehrig, F. van Harmelen, M. Sabou, R. Siebes, S. Staab, H. Stuckenschmidt, and C. Tempich. Swap: Ontology-based knowledge management with peer-to-peer technology. In *WM2003 - German Conference on Knowledge Management. Workshop on Ontology-based Knowledge Management*, Lecture Notes in Informatics,GI e.V. K ollen Verlag, 2003.
9. P. Busetta, P. Bouquet, G. Adami, M. Bonifacio, and F. Palmieri. K-Trek: A Peer-to-Peer Approach To Distribute Knowledge In Large Environments. In *Second International Workshop on Agents and Peer-to-Peer Computing (AP2PC 2003)*, Melbourne, Australia, July 2003. To be published by Springer, LNCS series - draft version available on-line at http://p2p.ingce.unibo.it/2003.
10. K. Ducatel, M. Bogdanowicz, F. Scapolo, J. Leijten, and J-C. Burgelman. Scenarios for ambient intelligence in 2010. Technical report, Information Society Technologies Programme of the European Union Commission (IST), February 2001. `http://www.cordis.lu/ist/`.
11. C. Ghidini and F. Giunchiglia. Local Models Semantics, or Contextual Reasoning = Locality + Compatibility. *Artificial Intelligence*, 127(2):221–259, April 2001.
12. Bill Schilit, Norman Adams, and Roy Want. Context-aware computing applications. In *IEEE Workshop on Mobile Computing Systems and Applications*, Santa Cruz, CA, US, 1994.
13. L. van Elst, V. Dignum, and A. Abecker. Towards agent-mediated knowledge management. In *Agent Mediated Knowledge Management*, volume LNAI, pages 1–31. Springer,Heidelberg, 2004.
14. Michael J. Wooldridge. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, February 1997.