International Journal on Advances in Internet Technology

2009 vol. 2 nr. 2&3

The *International Journal on Advances in Internet Technology* is Published by IARIA.
ISSN: 1942-2652
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Sponsored by IARIA
www.iaria.org

International Journal on Advances in Internet Technology
Volume 2, Numbers 2&3, 2009

Communication Theory, QoS and Reliability

- Adrian Andronache, University of Luxembourg, Luxembourg
- Shingo Ata, Osaka City University, Japan
- Eugen Borcoci, University "Politehnica" of Bucharest (UPB), Romania
- Michel Diaz, LAAS, France
- Michael Menth, University of Wuerzburg, Germany
- Michal Pioro, University of Warsaw, Poland
- Joel Rodriques, University of Beira Interior, Portugal
- Zary Segall, University of Maryland, USA

Ubiquitous Systems and Technologies

- Sergey Balandin, Nokia, Finland
- Matthias Bohmer, Munster University of Applied Sciences, Germany
- David Esteban Ines, Nara Institute of Science and Technology, Japan
- Dominic Greenwood, Whitestein Technologies AG, Switzerland
- Arthur Herzog, Technische Universitat Darmstadt, Germany
- Malohat Ibrohimova, Delft University of Technology, The Netherlands
- Reinhard Klemm, Avaya Labs Research-Basking Ridge, USA
- Joseph A. Meloche, University of Wollongong, Australia
- Ali Miri, University of Ottawa, Canada
- Vladimir Stantchev, Berlin Institute of Technology, Germany
- Said Tazi, LAAS-CNRS, Universite Toulouse 1, France

Systems and Network Communications

- Eugen Borcoci, University ‘Politechncia’ Bucharest, Romania
- Anne-Marie Bosneag, Ericsson Ireland Research Centre, Ireland
- Jan de Meer, smartspace®lab.eu GmbH, Germany
- Michel Diaz, LAAS, France
- Tarek El-Bawab, Jackson State University, USA
- Mario Freire, University of Beria Interior, Portugal / IEEE Portugal Chapter
- Sorin Georgescu, Ericsson Research - Montreal, Canada
- Huaqun Guo, Institute for Infocomm Research, A*STAR, Singapore
- Jong-Hyouk Lee, Sungkyunkwan University, Korea
- Wolfgang Leister, Norsk Regnesentral (Norwegian Computing Center), Norway
- Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France
- Sjouke Mauw, University of Luxembourg, Luxembourg
- Reijo Savola, VTT, Finland

Future Internet

- Thomas Michal Bohnert, SAP Research, Switzerland
- Fernando Boronat, Integrated Management Coastal Research Institute, Spain

- Chin-Chen Chang, Feng Chia University - Chiayi, Taiwan
- Bill Grosky, University of Michigan-Dearborn, USA
- Sethuraman (Panch) Panchanathan, Arizona State University - Tempe, USA
- Wei Qu, Siemens Medical Solutions - Hoffman Estates, USA
- Thomas C. Schmidt, University of Applied Sciences – Hamburg, Germany

Challenges in Internet

- Olivier Audouin, Alcatel-Lucent Bell Labs - Nozay, France
- Eugen Borcoci, University “Politehnica” Bucharest, Romania
- Evangelos Kranakis, Carleton University, Canada
- Shawn McKee, University of Michigan, USA
- Yong Man Ro, Information and Communication University - Daejon, South Korea
- Francis Rousseaux, IRCAM, France
- Zhichen Xu, Yahoo! Inc., USA

Advanced P2P Systems

- Nikos Antonopoulos, University of Surrey, UK
- Filip De Turck, Ghent University – IBBT, Belgium
- Anders Fongen, Norwegian Defence Research Establishment, Norway
- Stephen Jarvis, University of Warwick, UK
- Yevgeni Koucheryavy, Tampere University of Technology, Finland
- Maozhen Li, Brunel University, UK
- Jorge Sa Silva, University of Coimbra, Portugal
- Lisandro Zambenedetti Granville, Federal University of Rio Grande do Sul, Brazil

International Journal on Advances in Internet Technology
Volume 2, Numbers 2&3, 2009

CONTENTS

Individual and Social Recommendations for Mobile Semantic Personal Information Management

Wolfgang Woerndl, Georg Groh, Aleksandar Hristov
Institut für Informatik
Technische Universitaet Muenchen (TUM)
85748 Garching, Germany
{woerndl,grohg,hristova}@in.tum.de

***Abstract*— We present an approach for personal information management for mobile devices like PDAs based on the Semantic Desktop. The main objective is to design and realize a recommendation system to identify interesting items (e.g. messages or documents) based on the current context (time and location) and a user's personal ontology. To do so, our algorithm uses an evaluation function to traverse the graph of resources and rank nodes. Relevant resources and other items such as points-of-interests can also be displayed on a map on the mobile device. The ideas have been implemented and (rudimentarily) tested in the "SeMoDesk" application. Furthermore we introduce the extension of this application with the respect of social computing by a social information item filtering approach. The social filtering allows for the integration of other user's information spaces and makes use of a special bootstrapping algorithm for the integration of heterogeneous ontological perspectives.**

Keywords-personal information management, semantic desktop, social filtering, mobile, context

I. Introduction

Personal information management (PIM) is intended to support the activities people perform to organize their daily lives through the acquisition, maintenance, retrieval, and sharing of information [1]. Examples for personal information include documents, (Email) messages, contact data, appointments and also references to other information items. The collection of these items is called the personal space of information (PSI) [2]. The notion of personal information management first appeared in the 1980ies [3]. Since then, PIM tools have been developed and used. Due to increasing information overload of users during the last 15 years, the interest in PIM has been boosted very much.

Organizing data and having access to relevant information is particularly important in a mobile scenario, e.g. field staff meeting customers. To support these tasks, PDAs (personal digital assistants) and other mobile devices are available. However, organizing information on mobile devices is even more difficult when compared to a desktop setting. This is mostly due to the fact that mobile devices have limitations in network bandwidth, storage capacities, displays and input capabilities. For example, users cannot browse though many search results on the small screen of a mobile device. Therefore, it is very important to adapt information access to the current user needs and context in a mobile scenario.

Yet most of current PIM research is not geared towards mobile and ubiquitous information access. Therefore, the goal of this work is to support the user in mobile personal information management. More precisely, we want to design and implement a recommendation system to recommend resources to a user that are of current interest to her in a given context. For that purpose, the rest of the paper is organized as follows. First, we describe the background of our work, namely the Semantic Desktop. In Section III, we explain the main ideas behind our recommender. We also present the user interface of the prototype and give some implementation details. We also explain how to use the infrastructure to improve the context- and location-awareness of mobile PIM. In Section IV, we present the extension of the application to allow for including information spaces of other users in the personal social network (possibly in the near vicinity of a user's own mobile device) into the context sensitive item filtering. In Section V, we discuss related work. Finally, we conclude with a brief summary and outlook.

II. Background: The Semantic Desktop

One solution to deal with mobile PIM is building onto the *Semantic Desktop*, an approach aiming to integrate desktop applications and the data managed on desktop computers using semantic web technologies [4]. The main idea is to assign meta data to all data objects that a user uses on her computer. Thereby, relations between resources can be defined with the goal to integrate desktop applications and enhance finding relevant information.

Semantic Desktop approaches rely on ontologies to formalize relationships between resources and define a concept hierarchy that can be utilized for information retrieval. For the Gnowsis project, the "Personal Information Model" (PIMO) ontology was designed [5]. We have based our application on the PIMO ontology. The overall goal of PIMO is to define a concept hierarchy allowing a single user to formulate her view on tasks, contacts, projects, files and other resources.

In PIMO, one basic idea is to distinguish between "Thing" and "ResourceManifestation". "Thing" is a superclass of abstract concepts and physical objects, with the goal of representing them on a conceptual level. "ResourceManifestation" is a class to represent the actual documents on a computer system [5]. All objects in PIMO can be connected to each other using relationships, which we explain in more detail in Section III.

While there are Semantic Desktop implementations and related systems like the aforementioned Gnowsis available for desktop computer use, there is little for mobile environments. Therefore, we have designed and implemented SeMoDesk, which is a realization of the Semantic Desktop idea for PDAs [6]. The main design goals were to account for the restricted resources of PDAs, to build a stand-alone application (i.e. not a client of a Semantic Desktop server solution), because of possible network limitations, and adaptation to and usability on the mobile device. For example, phone calls and SMS messages are integrated which is not the case in related, desktop based approaches.

To assist the personal information management, the classes and instances of the personal ontology can be browsed in SeMoDesk. For example, all messages or calls with one person, or all resources such as document or appointments that are associated with a project, can be displayed with one tap on the touchscreen of the mobile device. However, browsing the ontology is not enough, as the following example illustrates. If a user is in a meeting right now, she might not only be interested in documents that are directly related to this meeting, but also messages that are related to a project or a person that is related to the meeting, or contacts that are concerned with a relevant topic, and so on. The goal of this work was to design and realize this kind of recommendation method which will be explained in the next section.

III. INDIVIDUAL RECOMMENDATION OF ITEMS

In this section, we first briefly describe how to manage the ontology, and then we explain the details of the recommender system that proposes resources to users based on the current context (location and time).

A. Managing the personal ontology

After starting up SeMoDesk, the user has the options to manage and browse her ontology, recommend resources of current interest, or display items on a map (cf. chapter III.C.). For the first task, users can and ought to define concepts such as projects, topics or subclasses of a "person" concept (e.g. "personal", "work") based on the PIMO ontology. Fig. 1 shows the top level of the ontology for browsing.

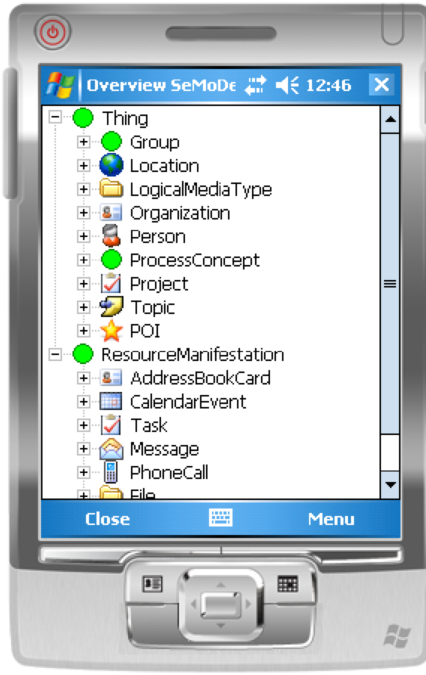

Figure 1. Browsing the ontology

Fig. 2 is example of the details of a location instance. Showing and editing items is possible by either taping on an item in the ontology (a context menu will then open), or by using the "Menu" soft key.

Figure 2. Displaying details of an instance

Afterwards, users can define relationships between the concepts and resources on the mobile device, for example stating that an appointment is related to a project. SeMoDesk was designed to assist the user in this task. In the example of a phone call (Fig. 3), if the caller's phone number can be found in the address book, a relationship between the phone call resource and the person concept is created automatically. The relations can then be easily retrieved. Fig. 4 shows the direct relations for a person.

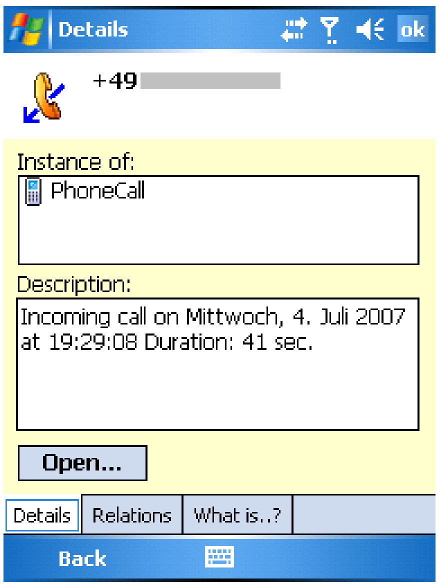

Figure 3. An incoming phone call

Figure 4. Displaying existing relations of an instance

The part of SeMoDesk application for managing the ontology from a user perspective, and the underlying system design, are explained in more detail in [6].

B. *Recommender algorithm*

1) Overview: The personal ontology we explained above forms a graph with a Thing and a ResourceManifestation as the root nodes. To find recommended resources in this graph, we first have to find appropriate starting nodes for our search. Thus, our basic idea for the recommender consists of the following two steps:

1. Finding current resources (Fig. 5), i.e. resources that are of interest for the user right now
2. Recommending other items, starting from the instances found in step 1

Figure 5. Finding current resources

For step one, our system offers different options. First of all, the user can manually select concepts or resources. To do so, the user can browse the "Concept" or "Resources" parts of the ontology (see the taps in the lower part of Fig. 5). For example, the user can select a topic or a document that is of current interest to her. In addition, the system proposes items in this first step based on the current date and time (button "On Schedule" in Fig. 5, and location (button "In Area"). For location-awareness, the user can define locations (e.g. an address) and assign appointments or other resources to these locations (Fig. 6).

Figure 6. Assigning appointments to locations

As a result of this first step, the system displays a list of resources which are of current interest to the user (middle part in Fig. 5). This list may already contain relevant resources, but the nodes in the list mainly serve as possible starting nodes for the more advanced search in step 2. The user can now select one node and start the recommendation process. The second step, finding additional relevant resources from the specified starting node, will be explained in more detail later in this section.

We like to underline the fact that the search algorithm only uses the relations between entities the user has defined. The system does not analyze the items themselves. However, relations between entities in SeMoDesk could be proposed by the system based on content analyzed, e.g. matching text in documents. Our algorithm searches for resources (e.g. appointments, documents, messages), not concepts. However all nodes and relations in the graph including concepts are used in the inference process.

2) Relations and relation types: As explained above, the personal ontology represents a directed graph. Thereby, it is necessary to distinguish between different relation types. We are using the following types as an extension and refinement of PIMO:

- Related: entities having any (weak or strong) relationship
- HasSubClass / IsSubClassOf: a class that extends the given class or is an extension of a super class
- HasPart / IsPartOf: for example, a person can be part of a project
- HasInstance / IsInstanceOf: any object is an instance of a class
- HasOccurrence / IsOccurrenceOf: if a person is mentioned in an article, for example, we say that this person has an occurrence in the document
- HasMade / IsMadeBy: a person has made a phone call, or a phone call is made by a person
- HasTopic / IsTopicOf: a given document or a project for instance can be assigned a topic
- HasCard / IsCardOf and HasBusinessCard / IsBusinessCardOf: denotes that a person or an organization has an address card

3) Finding a meaningful path: The main difficulty in our concept is how to traverse the directed graph of concepts and resources from a selected starting node. For doing so, it can be observed that not all edges are equally meaningful. Therefore it is important to apply heuristics to find out interesting edges to follow. The problem is illustrated in the example depicted in Fig. 7. In this example, we have two appointments which are instances of the CalenderEvent concept. We start with appointment 1 (“Meeting with Alex”) and seek related nodes. “Meeting with Alex” is related to a project that is also “occurrence” of a topic. This topic is discussed in our second appointment “Meeting with Vladimir” in the example as well. Because of this path between appointment 1 and 2, the system should infer that both appointments are related and could recommend appointment 2 if the user is currently in appointment 1.

But there is also a shorter, but less significant path from appointment 1 to appointment 2 because both are instances of CalenderEvent. We don’t want the system to follow this path because then all appointments would be related to each other and a recommendation is likely to be rather pointless. Therefore we have to put different weights on different relations (cf. III.B.5) thus penalizing the ones which are expected to be less meaningful (In this case the Instance-Of edges).

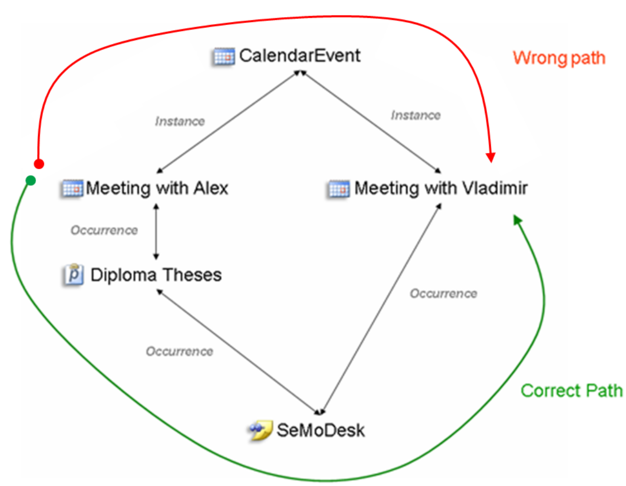

Figure 7. Finding a meaningful path

4) Search algorithm: For the purpose of finding an efficient search algorithm, we have examined common algorithms used in graph theory and Artificial Intelligence research [7]. As is well known, the algorithms can be roughly grouped into informed and uninformed (or blind) approaches. The latter just traverse the whole graph until satisfying results are found, e.g. depth or breadth first search. Informed algorithms apply heuristics to predict the distance to the target node e.g. A* algorithms. For reasons discussed in the previous paragraph we introduced different weights for edges. But for weighted graphs, simple BFS does not yield shortest paths. We could use a "semi-informed" Dijkstra algorithm (time complexity O(n log n+ m)) to find the shortest weighted path form a starting node to a target node that is currently beeing taken into consideration but the complexity is critical for a mobile application if the information space becomes larger.

As a compromise due to the limited resources on the mobile device and potentially many nodes in the ontology graph, we have decided to nevertheless use breadth first search (O(n+m)) in combination with an evaluation function to rate the quality of the expanded nodes. By doing so, we can avoid putting too much effort in following rather meaningless nodes as motivated above in III.B.3 by using BFS to traverse the graph and examining the weight of each node with our evaluation function. Our BFS does not expand nodes whose evaluation value is below the threshold and thus implements an approximation to the target behavior described in III.B.3. The evaluation function allows for incorporating further heuristics than just relation weights (see below).

The evaluation function returns the goodness on a node, in relation to the starting node. That's why our approach relies on specifying one starting node to trigger the search. Our algorithm terminates after all relevant nodes have been analyzed. Neighbors of "bad" nodes – i.e. nodes with an evaluation result below a threshold – are ignored (bad nodes are not expanded) and therefore the amount of expanded nodes is significantly reduced. If search time is a crucial criterion however, the algorithm could easily be modified to use an iterative deepening approach, which would allow termination after a maximal search time duration has elapsed.

5) Evaluation function: Our evaluation function is not only looking at the node that is being analyzed and its neighboring relations, but also at the shortest unweighted path from the starting node found by BFS. We have defined the evaluation function f for this analysis as follows:

$$f = a * depth + b * concept + c * relation$$

a, b and c are parameters to weigh the three factors:

- depth: is computed from the length of the (shortest) unweighted path from the starting node to the current node by weighting each edge by a fall-off coefficient. The fall-off coefficient can be configured in the system, for example we start with 1 and divide by 2 in every further step. Thus depth would be 1 * 0.5 * 0.25 for a path with three edges.
- concept: weight of the node itself, depending on the type of the concept or resource, e.g. less common resources have a higher weight
- relation: Summed up weight of the edges of the shortest unweighted path to the node, where different relation types (corresponding to the edges) can have different weights, as motivated above in chapter III.B.3.

The higher a node is evaluated by the function f, the more likely this node is relevant in the current context. All parameters can be configured in the user interface of our prototype implementation. However, an ordinary user is not supposed to configure the parameters herself, but use a predefined set of reasonable parameters based on the application scenario. For example, in a scenario with a lot of messages but fewer other items, message nodes may have a lower weight.

The algorithm starts examining the nodes with an evaluation value of 1. For each node, the algorithm loops through the whole path back to the starting node, and updates the value of the evaluation using the coefficients for each level. The result of this calculation is then returned to the traversing algorithm, which decides whether to put the node in the result set and whether to expand the node at all.

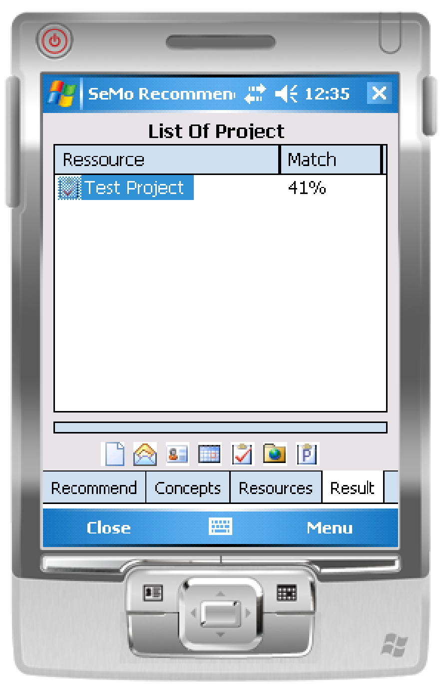

Figure 8. Search result

6) Results and their explanation: After searching the item space as explained above, a result list is shown to the user (Fig. 8). The result set is ranked by how relevant the items are, according to the search process and evaluation function ("match"). Only items with a match above a configurable threshold are given as results. In the example in Fig. 8, only one resource with a match of 41% is recommended.

In the lower half of the result screen, there are seven icons to toggle the type of items a user is currently looking for. The available options in our implementation are: Documents, Messages, Contacts, Calendar, Events, Tasks, Bookmarks and Projects. This selection is in addition to the different weights that nodes have in the evaluation function (see III.B.4). While messages may have been assigned a lower weight in the evaluation function in general, the user is still able to search for relevant messages, for example.

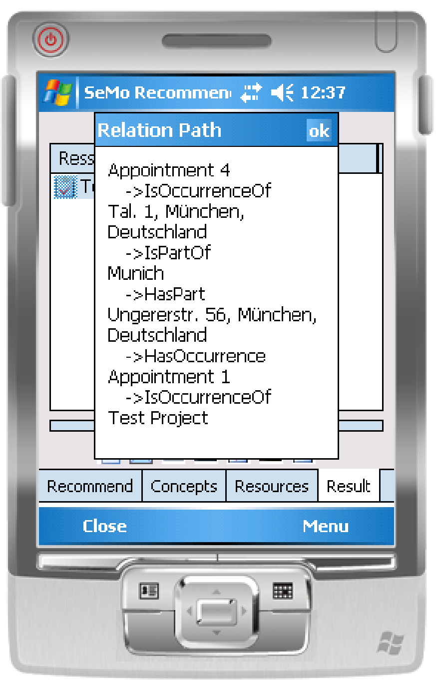

Figure 9. Relation path

The user has the option to display the relation path by taping on a result item (Fig. 9). Thus, the user can comprehend why this particular item was recommended. We consider this explanation an important part of the user interface. In general, it is desirable to explain results to the user in personalization and recommender systems, as studies have shown (e.g. [8]).

C. Using the personal ontology to recommend additional items based on location and time

Until now, the explained concepts allow searching the item space in the personal ontology. But the ontology can also be utilized to recommend additional resources, items that are not explicitly managed by the user. The application scenario is that users are looking for points-of-interests (POIs) in the current geographic vicinity to perform certain tasks. For this purpose, we have extended the PIMO ontology by a POI concept with sub concepts such as "cinema", "restaurants", "shop" etc. The user can then relate tasks or any other resources to POI types, as shown in Fig. 10. In addition, appointments (or any other resources, in theory) can be related to addresses (see above Fig. 6). When the user starts the mapping feature of SeMoDesk, information about relevant POIs are shown on a map (Fig. 11), together with the location of upcoming appointments.

Figure 10. Associating a task with a POI concept

There are several possibilities to retrieve the current user position to select the appropriate map segment and center the map on the user position. More and more mobile devices are equipped with GPS, for example. While the rest of SeMoDesk runs as stand-alone application on the PDA without any server, an Internet connection is required for the mapping feature.

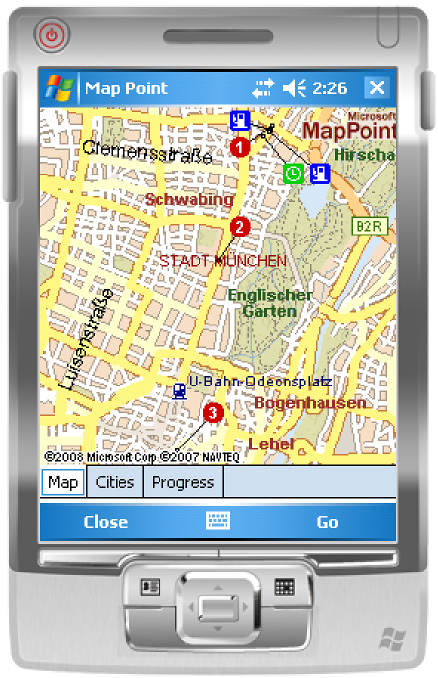

Figure 11. Displaying the map

D. *Indoor location-awareness using RFID infrastructure*

Another goal is to improve the indoor context-sensitivity of the system. We have worked with a RFID infrastructure to be able to locate the user indoors more precisely (e.g. displaying relevant resources when she enters a meeting room) [9].

To accurately model a user's position, a system requires a detailed location model where even small places can be distinguished. This results in the demand for a location model with a diversified granularity. As an extension to the ontology, we hierarchically designed the following classes for the location ontology:

- Country
- City
- Area
- Street
- Place
- Building
- PartOfBuilding
- Level
- Floor

This wide range of granular diversity allows us to model huge areas as well as small spots inside a building. The location taxonomy can be implemented by using HasPart – IsPartOf relations. Additional relation types like IsNearby or IsOnThe-RightSideOf/IsOnTheLeftSideOf can easily be added in the future. In addition, we needed to model location sensors and receivers respectively identification tags. To do so, we have designed a superclass "SensorTags" and subclasses, depending on the technologies used for positioning [9]:

- Sensor: Represents a stationary sensor (e.g. RFID)
- IP: IP entities are linked to Sensor entities and hold an attribute with the IP address to reach a specific sensor on a network (e.g. WLAN).
- RFID Tag: RFID tags represent the devices or objects they are linked to in the ontology. They hold a numerical ID.
- Bluetooth Tag: Bluetooth combines sender and receiver in a single unit that is built-in in electronic devices. Unique numerical tags identify those units.
- GPS Coordinates: Entities of this class hold coordinates for a certain place and are linked to either a Place or a Building entity.

A user is identified by a device with an RFID tag in our model. For reasons of simplicity we decided not to separate user and device positioning. All sensor tags shall be set in direct relation to the user whose devices they are attached to.

The button "In Area" (Fig. 5) is the user interface to search for a certain area. This integrates our location system with the resource recommender that was explained in chapter III.B. Hence, the recommendation system is extended with location awareness. When the user clicks on "In Area", a list of currently available sensors is shown to user. The user can then select one of them, for example a sensor assigned to a meeting room she is interested in right now. What the system does is provide a list of tags, a list of persons in a certain area and also a list of all the resources related to those persons. How this infrastructure can be used to find resources related to a position will be shown below in an example use case (cf. III, F.).

E. *Implementation details*

Our implementation was done using Microsoft's IDE Visual Studio 2008. The programming language is C# and the runtime environment is .NET Compact Framework 3.5. The application was tested using a HTC P3600 PDA phone and some other similar devices. SeMoDesk should run on any Windows Mobile 5 or 6 PDA with a touchscreen interface. For the POI search and mapping feature (chapter III.C.) we are using Microsoft's MapPoint. Other similar services could be integrated easily.

Fig. 12 gives an overview over the main layers of the system design [10]. To store all data, our application uses a SQL Server Mobile Edition as backend database. The main parts of SeMoDesk are components needed for the graphical user interface (GUI), a representation of the PIMO objects and corresponding data provider classes for the database access. Every item in our approach has a GetRelations() method that retrieves all corresponding relations in an efficient manner, for example. The "AI" package contains all the classes of the search and recommendation algorithm as explained above.

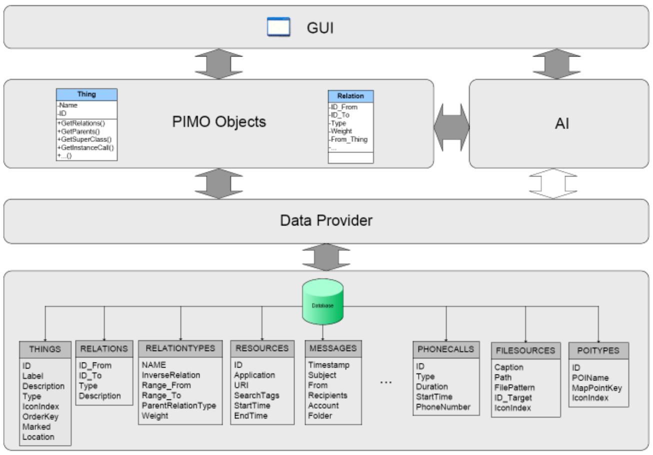

Figure 12. Architecture overview

F. Use Case: Displaying resources that are related to a room

Finally, we want to show how the explained infrastructure including the RFID part can be used to put the following use case into practice: “A user enters a meeting room and want to find resources that are related to this room”. We have used the short-range HF RFID reader Tricon Starter Kit 100 to test this scenario [9].

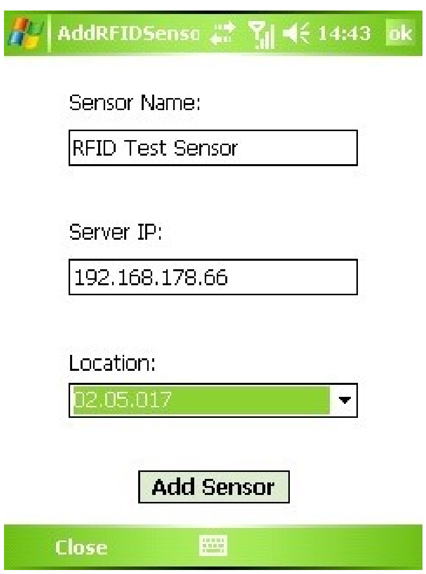

Figure 13. Setting up a new RFID sensor (left)

For this purpose, we set up our ontology to model the sample hierarchy of our university. An RFID sensor was added (Fig. 13) and associated with room 02.05.017.

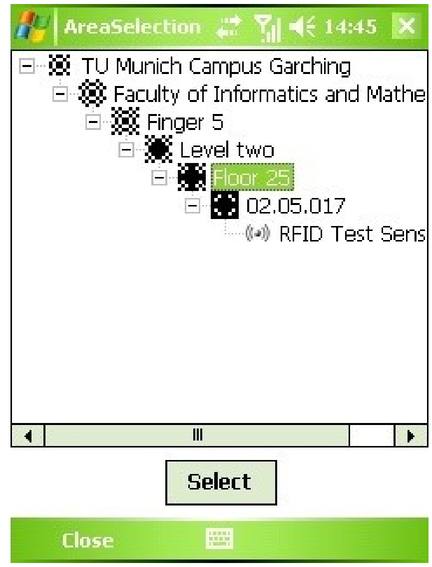

Figure 14. Selecting a floor in the location hierarchy (right)

Then, we started the location-based recommender using the aforementioned “In Area” button. In the appearing AreaSelection window (Fig. 14) we chose the higher location node “Floor 25” which included all sub nodes, i.e. our room with the RFID reader. SeMoDesk connected to the server linked to that reader and received a list of tags in its range (Fig. 15). The one tag that was transmitted turned out to be associated with contact Diane in our ontology. By traversing the relation graph the recommender then determined all entities directly related to the contact (Fig. 16), namely the task “Thesis” and also the associated RFID tag.

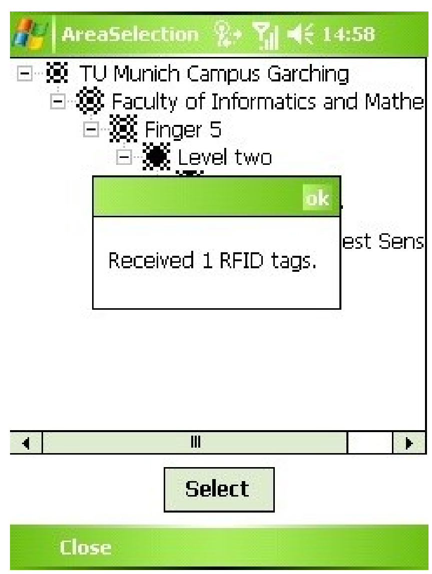

Figure 15. Receiving RFID tags (left)

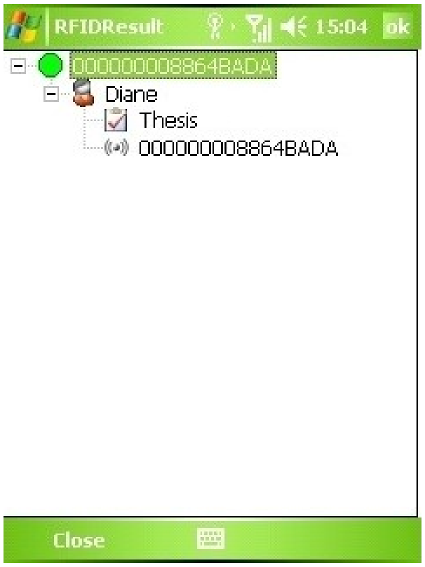

Figure 16. displaying recommended entities (right)

IV. SOCIAL RECOMMENDATION

In this Section IV we explain the extension of our recommendation function with social filtering.

A. *Overview*

Modeling social relations and using the resulting network models for social computing services has become a major trend in Web 2.0 [10]. Although several attempts have been made to apply the social networking paradigm to mobile interaction scenarios, truly convincing applications in this field are still largely missing. We aim at contributing to the research for better mobile social networking applications by investigating possible extensions to our existing mobile personal information management application. One promising option is to extend SeMoDesk with social item filtering, towards realizing the Social Semantic Desktop idea [11][12].

The meaning of the term Social Recommender can be twofold: On the one hand, it can designate systems that recommend social structures that maybe useful for a user. This includes friend-recommenders or team-recommenders [13]. On the other hand, it can designate approaches where the neighborhood from which recommendations are generated is not chosen by e.g. selecting the n other users with the most similar rating behavior, but rather chosen based on the social network of a user [14]. Groh and Ehmig found that social recommendations in the latter sense worked better than state of the art collaborative recommendations in taste-related domains [14]. Follow-up experiments indicated that for more fact-related domains the approach might not work as well. However, in both cases standard cross–validation evaluation methods for recommenders were used that are not able to value one key advantage of social recommenders: Horizon broadening recommendations [14]. By that we mean recommendations made by the close social network of a user, that the user might not like at first sight but that e.g. may help him in “complying to tendencies in his peer group“ or that he may like “on second sight” considering the social relations to those people whose ratings have mainly contributed to this recommendation having been made. These horizon broadening effects can make those recommendations useful.

The key idea of our social filtering approach is to extend the information space, which is subject to the information item filtering, by including parts of the information spaces of other users. To do so, we use the social network composed of the community’s individual’s contacts, already present in the basic application. The resulting network thus has directed edges, which can optionally be weighted (automatically or via an additional module that assigns weights by counting the relative number of communication acts with the respective person).

The overall process of information sharing, which can be mediated by all available network infrastructures (Bluetooth, WiFi Ad-Hoc-Mode etc.) is then very simple: An “inquirer” asks for related nodes to a specified node A from his own information space. While we generally limit the possible set of “inquired” persons to those that have a mutual social relation to the inquirer, we provide two more special modes for the determination of the actual set of “inquired” persons (besides the option to ask all mutual contacts): The first mode inquires all mutual contacts that are additionally located in the physical neighborhood. The idea behind limiting the possible set of to-be-inquired nodes to the immediate physical neighborhood is that this mode supports an additional social control about who might be inquired or who a user is inquired by. In the second mode all persons with which the mutual relationship is either of type business relation or personal relation are inquired. Depending on the type of current network access only a subset of the three basic types of inquiry may be supported.

In order to realize the incorporation of foreign information spaces we extend the PIMO ontology. While the problem of providing or collaboratively constructing an agreeable ontology of social relations is still at least partly open, we propose a basic intermediate solution by extending the PIMO ontology by introducing sub-concepts of “contact”, namely “business contact” and “private contact”. Thus by effectively assigning these relation-types to the corresponding edges we create a partition of the set of contacts that is applicable to the vast majority of social relations. It is nevertheless planned for later versions to further enhance the portfolio of relation types.

A second extension of the ontology regards the question, which elements of the personal information space are made “publically” available. In order to allow the user to control this option in more detail, we introduce Boolean attributes “socializable” (with sub-attributes “business socializable” and “privately socializable”) for every element of the personal information space with a default setting of FALSE.

The filtering (or recommendation) process delivering related items to a specified item from the personal information space can be managed by the inquirer by additionally specifying whether she wants to include other user’s information spaces at all and, if so, whether to include only business-contacts or only private contacts or both. Furthermore it can be specified whether only the local physical neighborhood is inquired or whether all contacts are inquired.

Since it cannot be generally assumed that many inquired users are willing to open their information spaces to the general public, we assume in our first version, as has been explained before, that the opening is confined to direct contacts. A more elaborate alternative would be to allow each user (in addition to the per-item-“socializable” attributes) to specify as a policy whether the opening is confined to the set of direct contacts (“restricted”), to the general public (“public”), or to a network of path length at most n away from her (“intermediate”(n)). In order to implement the last policy we have to include information about the social network path into the inquiry-element of the agent interaction protocol.

B. Bootstrap approach

After having determined the set INQ of inquired persons, the social extension of the filtering or recommendation process then follows a “bootstrap” approach: Assume that person X seeks items related to her own item A in the information spaces of the persons in INQ. Assume further that person Y is in INQ . Then the sequence is as follows:

1. If Y does not turn down X’s request, on request of agent X, Y virtually includes X’s item A into his own information space. Virtual inclusion encompasses all “agreeable” semantic item relations from the common PIMO ontology that are present for A in X’s information space. “Agreeable” relations have targets and types that are present in both information spaces.

2. Y then computes a set of related items from his information space with the algorithm described in the previous section with (virtual) start node A. With that step we find related items to A from Y’s “perspective”. After the computation, Y deletes the virtual node from her information space.

3. Y communicates the result set (with those parts of paths of the results which are agreeable) back to X

4. X virtually includes the result items from Y and other agents into her own information space and runs another instance of the filtering/ recommendation algorithm (restricted to those “foreign virtual nodes”) with start node A. This step yields related items to node A from X’s “perspective”. Overall we thus realize a common “perspective” of X on the one side and Y and the other agents on the other side.

Step 4 of the algorithm is optional and ensures that really a common perspective of relevance is established. The Application of Person X can also be configured to omit this step and trust Y’s perspective.

C. Example

The approach is illustrated with an example in Fig. 17. Person X seeks for related items to his “Eigner”-node. Shown in red are Root concepts from PIMO, shown in Blue are our PIMO Extensions. Red edges denote “is-a”-relations, blue edges denote “isInstanceOf” relations. In the example we assume that Y is organizationally related to X and has the same project “OSMOZIS” in his graph. After the virtual insertion, Eigner’s agreeable relations (in Fig. 17 one such relation is indicated as a dashed edge) are virtually included into Y’s information space. Then Y’s application applies the recommendation algorithm and sends the results to X’s application. These results represent Y’s view on what relates to “Eigner” in his information Space and should be recommended. For example this could be the “Palin” node of a person also working in his part of the “OSMOZIS-project”. “Palin” is sent together with the path <“Eigner” → “Project OSMOZIS” ← “Palin”>. If X’s application is configured to perform step 4, Person X’ application will then insert “Palin” under his “Project OSMOZIS” and re-run the recommendation process, which might or might not yield “Palin” as a relevant node depending on the structure of X’s information space.

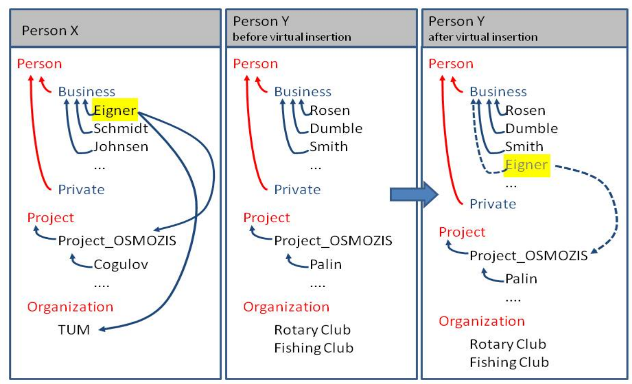

Figure 17. Example for the virtual insertion procedure

Actual access to foreign information items resp. nodes is subsequently implemented by a separate process, if demanded by user X.

In the first version, we use the unaltered local filtering / recommendation algorithm with local evaluation function described in the previous section, thus treating foreign virtual nodes and respective relations and own nodes and relations on equal footing. An enhanced version will include an additional relevance score-element, which differentiates between own and foreign nodes and configurably may give nodes from users a higher weight if the relation to that user is strong or a higher weight if the relation is weak, following Granovetter’s paradigm on the importance of weak ties and deduced importance of information items from the “social fringe”.

D. Implementation

In order to implement the variant of the service where only those mutual contacts are inquired which are in the physical neighborhood of the inquiring user, we rely on Bluetooth for detecting those users. GPS would be less suitable for this task because we assume that a substantial fraction if not the majority of such interactions take place indoors which renders GPS useless. Furthermore, besides its function for detecting nearby other nodes (users, devices), Bluetooth obviously can also serve as a channel to handle the communication between devices. In order to handle the problems associated with all various Bluetooth stacks we use a serial port emulation.

V. RELATED WORK

In this section, we discuss related work. One example for a Semantic Desktop implementation on a desktop computer is the already mentioned Gnowsis system [4]. Gnowsis consists of two parts, the Gnowsis server which performs the data processing, storage and interaction with native applications; and the graphical user interface (GUI) part, implemented as Swing GUI and Web-based interfaces. External applications such as Microsoft Outlook or Web browsers are integrated using standardized interfaces. There are other similar systems for personal computers or servers such as IRIS [15]. However, a Semantic Desktop approach tailored towards mobile devices comparable to SeMoDesk does not exist, as far as we know.

Integrating SeMoDesk with an existing desktop application would be possible rather easily, because the data model of SeMoDesk is based on the PIMO model, but is out of the current scope of our work. A reasonable real life scenario is that the user defines her PIM ontology on the desktop computer and imports and manages resources on the mobile device, while being able to browse and search items, and occasionally add sub concepts on the PDA.

Other earlier related work in PIM research includes the Haystack project which aims at connecting application data and let people manage their information using personalization [16]. However the Haystack client is a rather complicated and extensive application that is not usable for mobile devices. “Stuff I’ve seen” is another interesting desktop application [17]. It allows creating an index of content, including Microsoft Outlook resources, files, and Web pages in the browser cache.

Niu and Kay present a recent approach to utilize a personal con-text ontology called PECO [18]. The ideas are similar to the Semantic Desktop idea with respect to formalizing a user’s personal view on things. PECO is created semi-automatically and then applied for personalization. Their focus is solely on location concepts such as buildings and rooms though. The approach could possibly be integrated into our system to provide a more sophisticated model of location in PIMO.

The idea of the Social Semantic Desktop paradigm was introduced by Stefan Decker which also aims at integrating Social Computing and Semantic Web [11][12]. Gruber aims at the same integration, explaining promising approaches and techniques using the example of a collaborative travel information space [19]. Our system is another example for the integration of social computing and semantic desktop application with the added aspect that it is aimed at mobile social interaction. Völkl et al. formulate requirements for personal knowledge management, reviews some existing approaches and introduces an approach for adding semantic richness to personal knowledge management, which is clearly related to the personal semantic desktop paradigm [20]. Agosto et al. emphasize the decentralized P2P nature of social information exchange on the Web, by introducing an example application [21].

The special role of data-management in ubiquitous computing environments is emphasized in [22]. Two applications for social data exchange are discussed which show that social data exchange and sharing is a key application in mobile environments. Mobile social data sharing is also a natural application in e-learning environments [23].

While there a lots of reviews on recommender systems and information item filtering, Peis et al. attempt to review the activities in the field of semantic recommender systems, a field, which our application is also contributing to [24]. In the area of mobile recommender systems, [25] is an example for a decentralized system for recommending images on PDAs. The approach utilizes item-based collaborative filtering and also incorporates public shared displays for group recommendations.

Finally, there is work on semantics based context reasoning in mobile domains (see e.g. [26]). A more sophisticated model of context in addition to location and time could be integrated in our approach.

VI. CONCLUSION

We have presented an approach for mobile personal information management based on the Semantic Desktop idea. Thereby, users can define and manage a personal ontology to structure their information space. This ontology can then be used to recommend items based on the current user context. We have explained the reasoning behind the recommendation process in this paper.

In order to evaluate our approach, we conducted small scale qualitative evaluations in our lab where users where presented the application with a small information space and asked to perform several recommendation-processes. The feedback was very positive. A systematic evaluation design involves a set of users with their own information spaces. In a first step the individual recommendation process would have to be tested by each individual by rating the relevance and usefulness of the first n of the recommended items. We can then compute a precision estimation of the approach as an average over these ratings. The social recommendation process should then be tested in the same way.

Planned future improvements also include learning of relationships based on the content of resources content (e.g. finding similar documents), and also learning relations based on user behavior. For example, if the user works on a certain document when/after interacting with another user, a relationship between the document and the user could be inferred and proposed for addition to the ontology.

REFERENCES

[1] W.Woerndl and A. Hristov, “Recommending Resources in Mobile Personal Information Management”, Proc. IARIA International Conference on Digital Society (ICDS 2009), Cancun, Mexico, 2009.

[2] J. Teevan and W. Jones, Personal Information Management. Combined Academic Publ., 2008.

[3] M. Lansdale, “The Psychology of Personal Information Management”, Applied Ergonomics, vol. 19, no. 1, pp. 55-66, 1988.

[4] L.Sauermann, G.A.Grimnes, M. Kiesel, C. Fluit, H. Maus, D. Heim, D. Nadeem, B. Horak and A. Dengel, “Semantic Desktop 2.0: The Gnowsis Experience”, Proc. 5th International Semantic Web Conference, Springer LNCS 4273, 2006.

[5] L. Sauermann, L. van Elst and A. Dengel, “PIMO – a Framework for Representing Personal Information Models”, Proc. of I-Semantics' 07, 2007.

[6] W. Woerndl and M. Woehrl, “SeMoDesk: Towards a Mobile Semantic Desktop”. Proc. Personal Information Management (PIM) Workshop, CHI 2008 Conference, Florence, Italy, 2008.

[7] S. Russel and P. Norvig, Artificial Intelligence: A Modern Approach, 2nd edition, Prentice Hall, 2002.

[8] J.L. Herlocker, J.A. Konstan and J. Riedl, “Explaining collaborative filtering recommendations”, Proc. of the 2000 Conference on Computer Supported Cooperative Work, pp. 241–250, 2000.

[9] F. Schulze, Context- and Location-aware Retrieval Methods for Personal Information Management on Mobile Devices, Bachelor Thesis, Department for Informatics, Technische Universitaet Muenchen, Germany, 2009.

[10] A. Hristov, A Context-Sensitive Recommendation System for a Mobile Semantic Desktop Application, Diploma Thesis, Institut fuer Informatik, Technische Universitaet Muenchen, 2008.

[11] S. Decker, “The social semantic desktop: Next generation collaboration infrastructure”. Information Services & Use, vol. 26, no. 2, pp. 139-144, 2006.

[12] S. Decker and M. Frank, The Social Semantic Desktop. DERI Technical Report 2004-05-02, Digital Enterprise Research Institute (DERI), Galway and Innsbruck, 2004.

[13] M. Brocco and G. Groh, “Team Recommendation in Open Innovation Networks”, Proc. RecSys09, New York, USA, Oct. 2009.

[14] G. Groh and C. Ehmig, “Recommendations in Taste Related Domains: Collaborative Filtering vs. Social Filtering”, Proc. Group07, Sanibel Island, USA, Nov. 2007.

[15] A. Cheyer, J. Park and R. Giuli, “IRIS: Integrate. Relate. Infer. Share”, Proc. 1st Workshop on the Semantic Desktop, ISWC Conference, Galway, Ireland, 2005.

[16] D.R. Karger, K. Bakshi, K., D. Huynh, D. Quan and V. Sinha, “Haystack: A General-Purpose Information Management Tool for End Users Based on Semistructured Data”. Proc. Conference on Innovative Data Systems Research (CIDR), Asilomar, CA, 2005.

[17] S.T. Dumais, E. Cutrell, J.J. Cadiz, G. Jancke, R. Sarin and D. Robbins, “Stuff I've Seen: A system for personal information retrieval and re-use”, Proc. SIGIR, 2003.

[18] W. Niu and J. Kay, “Pervasive Personalisation of Location Information: Personalised Context Ontology”, Proc. of the International Conference on Adaptive Hypermedia, Springer, 2008.

[19] T. Gruber, “Collective Knowledge Systems: Where the Social Web meets the Semantic Web” Keynote Presentation, 5th International Semantic Web Conference, November 7, 2006, http://tomgruber.org/writing/CollectiveKnowledgeSystems.pdf.

[20] M. Völkel, S. Schaffert and E. Oren, „Personal Knowledge Management with Semantic Technologies“ In Rech / Decker / Ras, Emerging Technologies for Semantic Work Environments: Techniques, Methods, and Applications, chapter 9. Information Science Reference, June 2008.

[21] L. Agosto, M. Plu, L. Vignollet and P. Bellec, “SoMeONe: A cooperative system for personalized information exchange”, in: Enterprise Information Systems V, Springer, 2005.

[22] M. Franklin, "Challenges in Ubiquitous Data Management”, Lecture Notes in Computer Science 2000, Springer, 2001.

[23] G. Singh, L. Denoue and A. Das, “Collaborative Note Taking“, Proc 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education, 2004. pp. 163- 167.

[24] E. Peis, J. Morales-del-Castillo and J. Delgado-López, ”Semantic Recommender Systems. Analysis of the state of the topic”, Hipertext.net, num. 6, 2008. ISSN 1695-5498, http://www.hipertext.net.

[25] W. Woerndl, H. Muehe and V. Prinz, “Decentral Item-based Collaborative Filtering for Recommending Images on Mobile Devices”, Proc. Workshop on Mobile Media Retrieval (MMR'09), MDM 2009 Conference, Taipeh, Taiwan, May 2009.

[26] A. Bikakis, T. Patkos, G. Antoniou and D. Plexousakis, “A Survey of Semantics-Based Approaches for Context Reasoning in Ambient Intelligence”, In: Constructing Ambient Intelligence, Communications in Computer and Information Science series Vol. 11, Springer, 2008.

A Long-Tail Model of Mobile Application Usage

HANNU VERKASALO
Helsinki University of Technology
Helsinki, Finland
hannu.verkasalo@tkk.fi

Abstract–**In management research, the long tail phenomenon is typically linked to the long-tail of product demand distribution, particularly under electronic distribution, storing and consumption of content. This article discusses the role of open mobile software platforms in creating a market for niche mobile applications. Open software platforms of smartphones facilitate innovation around new applications. This study makes a hypothesis that open software platforms are boosting the use of niche applications. Empirical data on smartphone usage is collected over three consequent years in Finland. The dataset of 1 145 smartphone users is analyzed in studying whether the long-tail phenomenon is evident in the demand for mobile applications. The analysis of usage-level data reveals that the application demand is more heterogenic in the newest panel study than in the earlier studies. In other words, though the top 5% of applications typically represent more than 90% of total application usage, the bottom 80% of applications (the long-tail part) already represent 2.10% of total observed application usage in 2007, whereas this tail is only 1.39% in 2006 and 0.89% in 2005. Average usage activity of niche applications has increased. The analysis reveals a U-relationship between the number of users and usage frequency of applications, meaning that many niche applications are being used actively by those who installed them, suggesting that the value of add-on applications is high.**

*Keywords-**long-tail; mobile applications; smartphones; application stores***

I. INTRODUCTION

Chris Anderson introduces the concept of long-tail in his articles [1] [2], suggesting that though typically only few top hits or instances (e.g., top movies, books, search words) dominate the rankings, particularly the digital means of content distribution and consumption have opened the doors for niche products that face demand from only few people. These products, though outside of the top rankings, are significant in number, and together form the long-tail. The value of this long-tail can be significant, because of the mere number of titles in the long-tail. Particularly in situations where the supply and demand of products is potentially infinite (huge variety), the long-tail is evident. In addition to Anderson, e.g., Kilkki [13] discusses the practical applications and mathematical modeling of the long-tail concept.

Figure 1 presents the basic logic of the long-tail phenomenon. Few titles receive very high popularity (e.g., sales), but the potentially infinite tail of the distribution can cumulatively represent a significant share of total value and volume of the market.

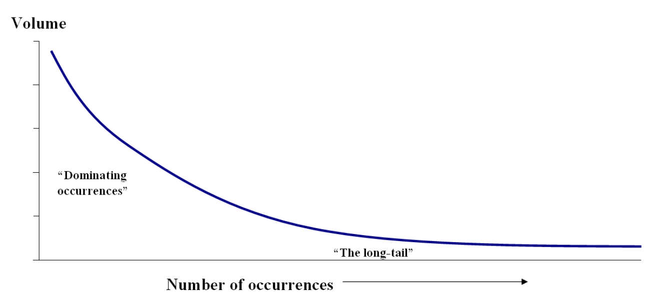

FIGURE 1 - THE LONG-TAIL PHENOMENON

The mobile industry is undergoing a major transformation, as it is converging with the Internet, media and computer industries [22]. From the perspective of this paper particularly the increasing penetration of open mobile software platforms is of importance [16], transforming mobile phones into multi-purpose smartphones. The birth of the mobile application market is the consequence of this trend. Independently of handset vendors or mobile operators, mobile software developers (3rd party companies and individuals) can create their own solutions on top of Symbian, Windows Mobile, Google Android or any other software platform. In essence, mobile phones have become programmable handheld computers, which have Internet connectivity, computing power and open APIs (application programming interfaces), providing prospective platforms for an infinite set of new mobile services and applications.

The assumption of this study is that the creation and evolution of the mobile application market, which is constantly being induced by the penetration of open mobile software platforms, is changing the way how end-users use mobile phones. The key hypotheses of the thesis include:

1. A long-tail of mobile applications is emerging, and the application demand is distributed over an increasing number of applications
2. The mobile application market is more fragmented than earlier, there is more variety both in supply and demand
3. Niche products can achieve high usage among the few who adopt them

The research problem of this study is to find out whether empirical metrics of smartphone usage over time reflect these hypotheses. New empirical modeling approaches are developed in solving the problem [24].

The article first introduces the concept of long-tail, and then proceeds to a summary of the current state of the mobile industry, particularly with regards to smartphones and add-on applications. After that, the research setting and dataset of this study are explained, and then rigorous analysis practices are applied in studying the distribution of add-on application usage activity. Finally, the article summarizes the main findings of analysis.

II. BACKGROUND

A. Long-tail

The long-tail phenomenon is first introduced by Anderson (see [1] and [2]). Anderson realizes that many businesses of today (e.g., Amazon) generate significant revenue by selling a high number of items in small quantities. Despite the market including a small number of dominating titles that sell in huge numbers, the digital age provides cost-efficient distribution and storing mechanisms to economically sell practically an infinite number of items, each potentially selling only a handful of copies. However, all together these low-selling items represent a significant amount of total volume.

Statistically the concept of the long-tail has been known for ages. Many distributions, such as power law and Pareto distributions, experience a long-tail. Statistically the long-tail means a low-frequency part of the distribution following a high-frequency part. This low-frequency part of the distribution asymptotically tails off. Many business cases account for this phenomenon, as it relates to many things from sales volume to productivity of employees. For example, McKinsey is using its 80/20 rule typically in communicating various business-related findings [8].

What Anderson [1] [2] and Shirky [18] contribute to the existing literature, is the suggestions that the digital economy makes both storing and distribution of products (e.g., content, applications, software, products) cheaper, thus making it economically viable to provide much more heterogenic portfolios of products available for sale. In other words, the supply of products (in terms of number and variety of items) goes up. Given the heterogenic preferences of people, there will be a creation of markets for niche products, selling only few copies. These niche products contrast with the bestseller hits that dominate the rankings. However, due to the changing economies of supply and demand, the relative total volume and value of these niche products (making up the long-tail) is much higher than in traditional markets.

In addition to Anderson and Shirky, Ken McCarthy [15] points out the impact of the Internet (and openness) and the potential emergence of the long tail phenomenon. The assumption is that all people have individual preferences, and there is (some) demand for a high number of products, given they can be economically provided for sale. For example, digital online stores, such as Amazon, boost the size of the market and variance of products sold, creating consumer surplus by simply changing the mode of product delivery [3]. In a later article [2] it is also suggested that demand side dynamics, such as search engines and recommendation engines, help customers to find niche products and to induce the long tail effect. The long tail has been discussed under many topics, from competition [12] to user-driven innovation [10], and from science fiction novels [9] to contrary effects of the Internet [7].

B. Mobile software platforms

This paper follows the definition of Webodia [25] for a *mobile operating system*, which is defined as an operating system for mobile devices, meaning essentially a *software platform* on top of which applications can run. Software platform is used as a synonym to operating system in this paper, highlighting the platform functionality of operating systems.

The key contribution of a mobile software platform is its programmability: in addition to default programs embedded in the system also new applications can be installed and used. The PC industry is known for its modular technical design, in which openness and the role of operating systems as platforms is critical. The emergence of mobile operating systems, such as Symbian, Windows Mobile, Apple iPhone and Google Android (forthcoming) are transforming the mobile industry towards more PC like evolution. Symbian is a market leader of platforms at Q4/2008 (65% market share), followed by Windows Mobile (12%) and RIM (11%). Symbian is mainly being boosted by Nokia (see Table 1) with its massive sales volume of converged devices.

TABLE 1 - SALES OF CONVERGED MOBILE DEVICES Q4/2007 [6]

Vendor	Q4 2007 shipments	% share	Q4 2006 shipments	% share	Growth Q4'07/Q4'06
Total	**35,522,360**	**100.0%**	**20,667,200**	**100.0%**	**71.9%**
Nokia	18,802,480	52.9%	11,114,630	53.8%	69.2%
RIM	4,046,860	11.4%	1,829,260	8.9%	121.2%
Apple	2,320,840	6.5%	-	0.0%	NA
Motorola	2,301,260	6.5%	1,463,090	7.1%	57.3%
Others	8,050,920	22.7%	6,260,220	30.3%	28.6%

Smartphones are here defined as pocket-sized computer devices that provide at least cellular circuit and packet switched connectivity, and run a mobile operating system. Smartphones can also be called as multimedia computers [17] or converged devices [5]. Smartphones access wireless networks through various radio-access technologies, such as WiFi, 3G and EDGE. Processing power and memory capacity of smartphones support advanced services, from games to office applications. Smartphones effectively combine traditional offline functions, such as personal information management or office applications, with online services such as person-to-person communications or Internet browsing. [11] Effectively mobile phones are migrating from communication devices towards computers. According to [11], today's smartphones hold the highest potential in becoming multi-purpose devices supporting everything from communications to digital wallets and from personal data assistants to authentication/authorization devices

Mobile software platforms are the key cornerstone in the transformation of the mobile industry [22]. As the user-driven innovation, open APIs, and connectivity to the Internet are the key factors of this evolution, smartphones together with open mobile software platforms serve as catalysts of the evolution. This paper assumes that the open mobile software platforms are creating new supply of services and applications. Independent and numerous developers all over the world can build new applications, and for example players of Internet and computer industries can easily port their existing solutions to mobile phones. Along with the increasing supply also increasing demand should realize, as the potential demand can better be fulfilled. The emerging variety of available mobile applications is defined here as the *mobile application market*. The mobile application market is created partially by the increasing number of applications shipping with new devices, but more importantly because of the programmability of devices and availability of add-on applications.

The creation of the mobile application market could have two profound effects on the usage of applications:

- First, it can shift demand to low-frequency and niche applications.
- Second, it can create new demand by providing solutions that did not exist earlier.

The purpose of the empirical part of the paper is to collect and analyze data particularly with regards to the first implied effect. The second effect is more difficult to study, as the size and length of the panel studies used for data collection in this study are different, and therefore the absolute number of identified applications does not necessarily communicate the absolute domain of demand. In addition, the micro-level analysis perspective of this study is suitable particularly for relative comparisons (the first implied effect above).

III. ANALYSIS

A. Research setting

A handset-based end-user research method is used in the empirical analysis of this paper [20]. This research method is developed over the years to study the behavior of mobile end-users using smartphones. The research method includes a handset-based client, that observes usage-level events (e.g., application sessions), and transmits this data to centralized research servers at predefined intervals. In addition, various web-based surveys can be deployed through the research platform.

The handset-based end-user research is structured as panel studies, which typically include a few hundred consumers from each geographical market for a period of 1-2 months. The advantages of the platform are the combination of subjective survey and objective usage-level data, observations of real end-user behavior, and accuracy as well as variety of data points available. The shortcomings include the effort of arranging panel studies, and adverse selection of panelists (only early-adopters can be studied because of the requirement of owning a smartphone).

The dataset of this study include three panel studies, arranged in a similar manner in three consequent years in Finland 2005-2007 (for the reports of the studies, see [14]; [21] and [23]). Only panelists with a self-purchased device are included in the dataset. This is because handset bundling in Finland is bringing smartphones to more mass-market oriented people, which can be hypothesized to form a different type of market (in terms of tech-savvy nature) from smartphone self-purchasers [22] [19]. 500, 369 and 276 Finnish early-adopter consumers (equipped with Symbian smartphones) are studied in years 2005, 2006 and 2007, respectively. In the analysis of the data, each application usage session of the panelists is identified, and analyzed with standardized data mining methods. For each application several metrics are calculated, ranging from number of trial users to number of active users, and from average time spent per day to average number of weekly application activations. All in all, 359 744, 251 749 and 138 636 application activations are observed in the panels of 2005, 2006 and 2007, respectively.

B. Usage of mobile applications

Usage-level data (applications usage) is available from three years. In addition, a special add-on application survey is conducted during the smartphone panel study of 2007. Panelists are asked several questions regarding add-on application installation and usage. Appendix A provides the results of the questionnaire. All in all, 84% of panelists have installed add-on applications to the device, and 35% claim to install applications frequently. The most typical way to install applications is to download from the Internet with computer (70% of those who installed applications), and to install then the application from computer to mobile phone via USB (66%). 57% of those who have installed applications, have downloaded applications directly with a mobile phone. The Internet is the best source of information when looking for applications (85% of those who have installed applications browse the Internet with computer when looking for information). 42% of those who installed applications have heard of new applications from friends or family. The most typical reason for not installing more applications is the lack of interesting applications in the market (62% of those who have installed applications blame this). 31% blame the prices, but only less than 20% blame the difficulty of installation or search. 76% of panelists have a positive attitude for an advertising-based delivery of content and applications. Handset vendors and operators are still considered as the most important actors among the producers of applications (31%, 26%, 10% and 6% consider vendors, operators, Internet companies and media companies as very important actors in mobile service delivery, respectively). See Appendix A for details.

The usage-based dataset of three years is first processed with standardized data mining processes. The raw data consists of accurate traces of each application usage session of each panelist over the panel period, and the data is available from all of the three annual panel studies. Voice calls are not studied here, and the focus is solely on smartphone applications. In the data mining process, average activation times per day and average usage frequencies (share of days when used) are calculated for each application and for each user. This data is further linked to separate application mapping files (see [20]) that map each application into a distinct functional category. Based on the data also the number of users for each application is calculated. Table 2 illustrates some of the key descriptive statistics of the dataset. All in all, the number of panelists without a bundled subscription (the requirement for the panelist to be included in the dataset) is not that high for the newest panel than earlier. However, significant amount of data is collected each year. The average usage activity (in activations per day) and distribution of application activations among the different types of applications do not experience significant changes. PIM (*personal information management*) corresponds to the use of phonebook, calendar and other daily applications.

TABLE 2 - DESCRIPTIVE STATISTICS OF APPLICATION GROUPS

Means	Group A			Group B			Group C		
	Usage Frequency	Usage Intensity	User Rate	Usage Frequency	Usage Intensity	User Rate	Usage Frequency	Usage Intensity	User Rate
2005	22 %	0,55	77 %	5 %	0,09	22 %	7 %	0,14	3 %
2006	25 %	0,61	81 %	6 %	0,11	23 %	7 %	0,20	3 %
2007	26 %	0,65	79 %	7 %	0,13	25 %	6 %	0,12	4 %

This article studies the patterns of application usage and the structure of realized demand. Regarding the use of applications, Figure 2 plots the share of panelists who have at least once tried applications. All the applications are plotted in descending order of number of users (x-axis as the percentile). The figure communicates the diversity of applications that are used by end-users. All in all, 820, 752 and 404 different applications are observed for the years 2005, 2006 and 2007. The number of distinct applications is decreasing over time, because in the most recent panels the amount of usage data collected is lower than earlier.

It is notable that the more recent panels experience wider use of mobile applications. In other words, for example the 10% percentiles of user rates (in share of panelists) are 3%, 5% and 10% for 2005, 2006 and 2007. The share of panelists, therefore, adopting rare applications (outside of top 5%), is higher in newer panels. In other words, an increasing number of mobile applications are achieving high penetration rates among early-adopters.

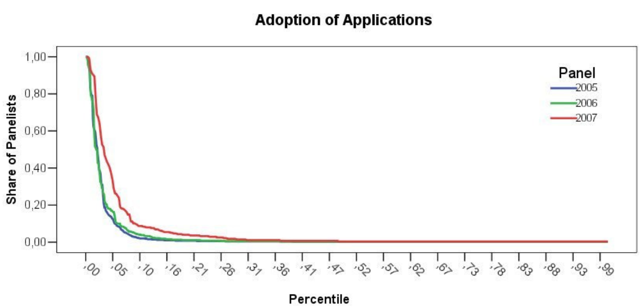

FIGURE 2 - ADOPTION OF MOBILE APPLICATIONS

Next, more accurate usage-level profiles of applications are studied. Usage frequencies are plotted across user rates for all applications that have at least 1% user rate (meaning that at least 1% of all panelists have used that particular application). Usage frequencies communicate the average frequency of usage in percents (the share of all panel days when the application is used), and user rates communicate the share of all panelists who have tried the application. Figure 3 plots the exemplary results for the applications observed in the panel of 2007.

It is interesting to differentiate between three main types of applications. The widely adopted applications (group A; having a user rate higher than 50%; all embedded applications), less widely adopted applications (group B; having a user rate between 10% and 50%; almost totally embedded application that do not achieve high success), and niche applications (group C; having a user rate lower than 10%; almost totally 3rd party applications).

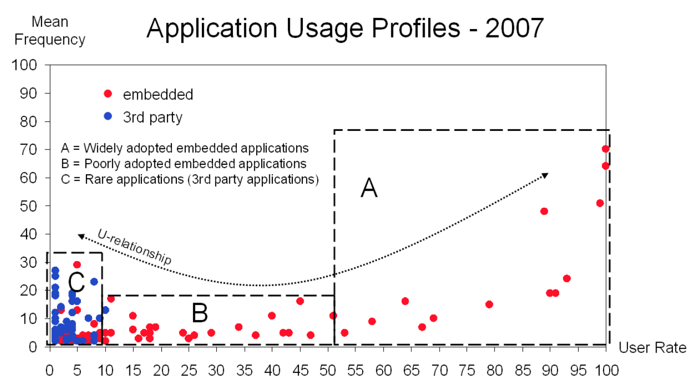

FIGURE 3 - THE U-RELATIONSHIP BETWEEN MEAN FREQUENCIES AND USER RATES

Though it can be expected that only few applications make it to the group A, it is interesting that many of the embedded applications in today's smartphones (for example embedded calculators and notes applications) do not achieve the 50% user rate, and are instead categorized into less widely adopted applications. The most significant observation is the high number of applications existing in the group C, meaning that a long-tail of applications exist - including many niche applications that do not achieve a high user rate.

Panel	Panelists	Different Applications Observed	Total Amount of Panelist-Days	Mean Number of Application Activations per Day per User	Share of Browsing	Share of Multimedia	Share of Messaging	Share of PIM
2005	500	820	32 749	10,90	2 %	11 %	26 %	45 %
2006	369	752	24 630	10,11	3 %	12 %	26 %	44 %
2007	276	404	14 431	9,85	4 %	12 %	27 %	43 %

Many of the applications in group C receive high mean usage frequencies, meaning that those few panelists, who use them, use them actively. This means that though these niche applications receive a low number of users, these niche applications can still generate a lot of value to the end-user, assuming that high usage frequency corresponds to high perceived value. This inverted relationship between user rates and mean usage frequencies is here called as the U-relationship between the number of users and usage activity. This U-relationship holds irrespective of the panel (see Figure 4).

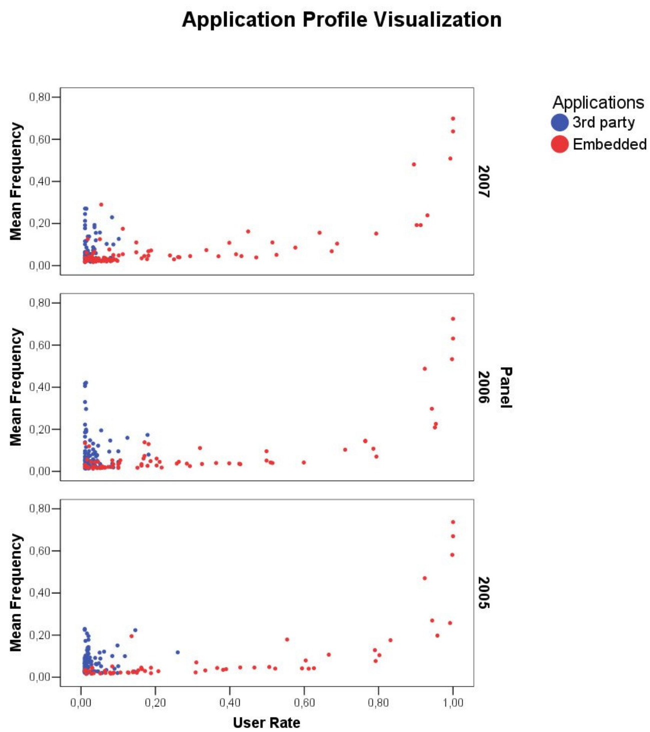

FIGURE 4 - ADOPTION OF MOBILE APPLICATIONS

Table 3 highlights the identified patterns in each of the application groups. Usage intensity is calculated as the average number of activations per day per user. Expectedly the usage intensities and frequencies are highest in the group A of applications. However, there are no significant differences between applications of group B and C. In the panels of 2006 and 2005 the mean usage frequencies and intensities are higher for the applications in the group C than in the group B. The number of applications that achieve high usage frequencies (>20%) is higher in the category C than in the category B, this holding in all panels. However, the number of applications is also higher in the category C, which forces the arithmetic means quite low.

The descriptive analysis of this chapter reflects the diversity of application usage. In addition, it emphasizes the U-relationship between application usage frequency and number of users. In other words, the study suggests that though the application might be a niche item in terms of number of users, the realized usage activity might still be quite high. This gives support regarding the hypothesis of the existence of the long-tail of mobile application usage.

C. Long-tail of mobile applications

The hypothesis of this paper is that the usage of mobile applications is more heterogenic today than earlier. In other words, the distribution of total application usage should be flatter than earlier, due to increasing usage activity of add-on applications. Figure 5 plots the total amount of usage (daily usage intensities) over the users of the applications. The applications are sorted in the descending order of total usage (in number of launches per day).

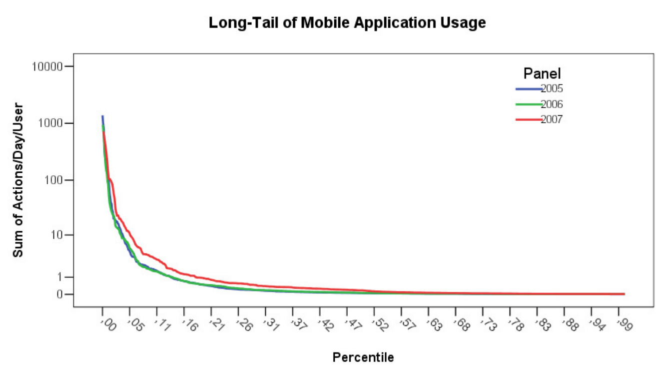

FIGURE 5 - TOTAL AMOUNT OF MOBILE APPLICATION USAGE (LOGISTIC SCALE)

The figure reveals that the end of the usage distribution is very flat (the logistic scale is used for the purposes of illustration). Demand exists for a number of applications. Additionally the hypothesis of the study can be confirmed true, as the distribution of application usage for 2007 is flatter than in 2006 or 2005. In other words, the value of the tail of the distribution is higher in 2007 than earlier, though absolute numbers are used in the figure (and fewer panelists are included in the dataset of 2007 than in 2006 or 2005).

Table 4 presents the analysis for each panel study. Generally, the top applications catch a significant share of total application usage. This is not surprising. However, the share taken by these top applications is decreasing over time. In 2005 the top 3% of applications (ranked based on total usage activity) represent 92.18% of total usage, in 2006 only 89.64% and in 2007 85.76%. The estimation of the total volume of usage in the long-tail part of the distribution (the bottom 80% of applications) reveals that in 2007 application usage patterns are more heterogenic than in 2006 or 2005, mainly because of increasing usage activity of add-on, niche applications. The total share of usage in the bottom 80% of the distribution is 0.89% in 2005, 1.39% in 2006 and 2.10% in 2007.

TABLE 4 - LONG-TAIL STATISTICS OF MOBILE APPLICATION USAGE

	2005	2006	2007
Cumulative volume of top 1% of titles	79,35 %	75,95 %	65,64 %
Cumulative volume of top 3% of titles	92,18 %	89,64 %	85,76 %
Cumulative volume of top 5% of titles	95,51 %	93,82 %	90,87 %
Cumulative volume of top 10% of titles	97,76 %	96,82 %	95,38 %
Cumulative volume of top 20% of titles	99,11 %	98,61 %	97,90 %
Total volume of the bottom 80% of titles	0,89 %	1,39 %	2,10 %
Rule	4.8%/95.2%	5.6%/94.4%	6.9%/93.1%

Figure 6 illustrates graphically the cumulative usage of mobile applications. As can be seen, in 2007 the cumulative distribution line is beneath the lines of 2006 and 2005, indicating that a relatively higher number of applications are responsible of the total demand for mobile applications.

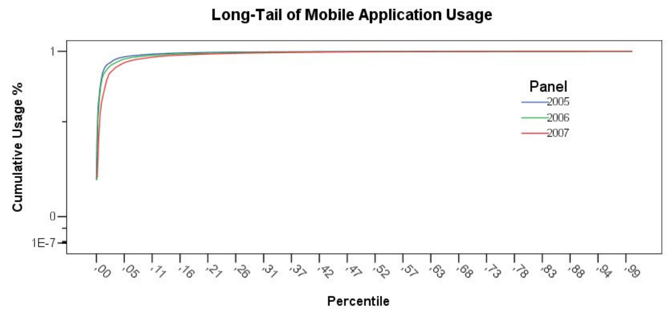

FIGURE 6 - CUMULATIVE USAGE OF MOBILE APPLICATIONS (LOGISTIC SCALE)

The statistics above support the hypothesis, that the long-tail of the mobile application market is not insignificant. In fact, it has been growing in volume over the years. In general, instead of the typical 20%/80%, a modified rule of 6.9%/93.1% holds in the panel study of 2007, for example. Top 6.9% of applications, ranked by total usage activity, are responsible for 93.1% of total smartphone usage observed in the panel study of 2007. The analysis confirms that average usage activities of niche applications have risen, and the bottom part of the distribution represents more of the usage in 2007 than in 2006 or 2005. It is not possible to analyze the absolute size (length) of the long-tail in this study, as the panel studies are of different length and size (which affects the likelihood of observing applications). This relative comparison, however, confirms that the balance in usage between default platform and niche add-on applications is shrinking.

Figure 7 explores the usage patterns of mobile applications. Though the total number of usage sessions in the panel is not that high for the bottom 80% of applications (deriving from the rules of ranking), the mean absolute usage activity (sessions per day per user) is still quite high for many applications, as can be seen in Figure 7. The usage activity in sessions (application activation and consequent usage) per day per application user can be considered as a proxy for the value of the service, as it reflects the extent of application usage among those who have really adopted the application. Also alternative metrics for usage activity exist, such as absolute face time per day per user, but different applications experience high variety in this variable due to their inherent nature (consider e.g., music players against calendars). Therefore usage sessions were chosen as the key metric in this article. Although the total panel-wide usage is not always so extensive, the value of the application to an individual user might be high. This corresponds to the finding of a U-relationship between the number of users and mean usage frequencies. In essence, the value of niche applications can be significant to the ones who adopt them, and this leads to the long-tail phenomenon.

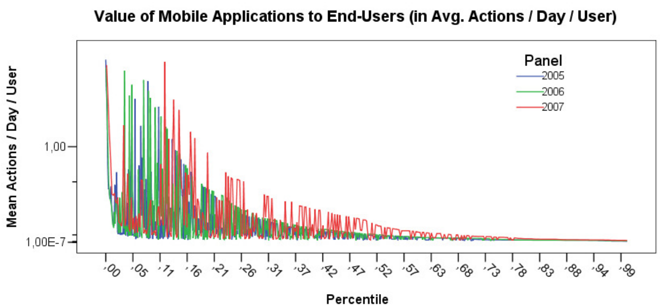

FIGURE 7 - VALUE OF MOBILE APPLICATIONS TO END-USERS (LOGISTIC SCALE)

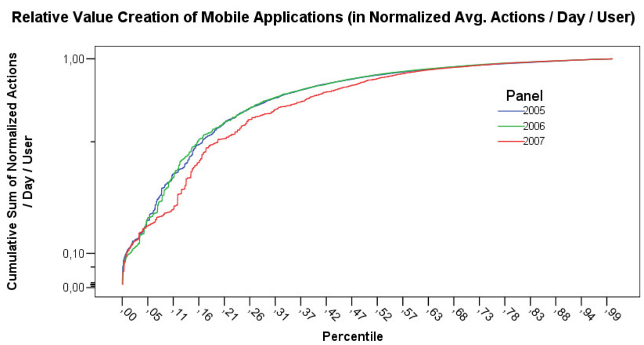

FIGURE 8 - CUMULATIVE RELATIVE VALUE OF MOBILE APPLICATIONS TO END-USERS (LOGISTIC SCALE)

Figure 8 reflects the value-creation of mobile applications. An average number of usage sessions per application per day per user are normalized against the total number of sessions observed in the panel, and then the cumulative sum of these values (assuming that observed usage sessions per day per user reflect the value of applications) are plotted against percentiles. The figure illustrates that the value creation of mobile applications is not that steep in 2007 as in 2006 or 2005. This suggests that the value of mobile application usage increasingly derives from niche applications, the finding that is done already earlier in this paper with the U-relationship of application user rate and usage frequency, and with the long-tail analysis of Table 4.

CONCLUSION

According to a survey study conducted in 2007, 84% of panelists have installed add-on applications to their mobile devices, and 35% claim to install applications frequently. 76% of panelists have a positive attitude for an advertising-based delivery of content and applications. Handset vendors and operators are still considered as the most important actors among the producers of applications (31%, 26%, 10% and 6% considered vendors, operators, Internet companies and media companies as very important actors in mobile service delivery, respectively).

The study covering real empirical usage-level data from three consequent years 2005-2007 in Finland reveals that a U-relationship exists between the number of users and average usage frequency of applications. In other words, the most widely adopted applications also experience high-frequency usage from end-users, meaning that they are valuable to end-users. The middle group of applications, including applications that are used by many panelists because they are typically embedded in smartphones, do not experience very active usage on average. However, the niche applications, receiving only a handful of panelists, experience very active usage inside their narrow user domains, increasingly so over time. These niche applications therefore generate significant value to the particular end-users who adopt them.

The further analysis of usage-level data reveals that indeed the demand for mobile applications is more heterogenic in the newest panel study than earlier panels. In other words, though the top 5% of applications typically represent 91-96% of all application usage, the bottom 80% of applications (the long-tail part) already represent 2.10% of total observed application usage in 2007, whereas this tail was only 1.39% in 2006 and 0.89% in 2005. This change is due to the increasing usage activity of add-on applications. In the newest dataset from 2007, 6.9% of top applications represent 93.1% of total smartphone usage.

The article finds that indeed the mobile application market is fragmenting, and end-users increasingly derive value from niche applications. This holds albeit still the top applications represent a significant volume of total smartphone usage. Value plots of the paper are based on the assumption that the observed usage activity per panelist reflects the value of the application to end-users. Based on this value analysis, it seems that the value is created increasingly in the long-tail of the application market. In this regards the mobile industry is moving towards the PC and Internet industries, where wide consumer choice is prevailing. The implication of this research is that there is a clear business case for developers who are targeting a selected group of subscribers, and not selling in high numbers. The value of the application (and therefore willingness to pay) might be very high for the selected, niche, subscriber segment.

Though the relative plots of application usage patterns confirm the hypothesis regarding the existence and relative size of the long-tail of the mobile application market, the absolute growth of mobile application market cannot be studied with the obtained data. This mainly results from the fact that that the size of the panel studies differs from each other and the length of the studies is different. Future research should attempt to collect data that is easier to control and compare against other available datasets. In addition, a more macro-level instead of a micro-level study setting (meaning more representative, bigger panels) should be established in order to follow the trends on the market. Future research should attempt to understand if the findings are a result of shifting demand or advancing technology.

References

[1] Anderson, C. (2004). The Long Tail. *Wired*, volume 12, number 10 (October), at http://www.wired.com/wired/archive/12.10/tail.html, Referred 4.7.2008.

[2] Anderson, C. (2006). *The Long Tail*. London: Random House.

[3] Brynjolfsson, E. & Hu, JY. & Smith, MD. (2003). Consumer Surplus in the Digital Economy: Estimating the Value of Increased Product Variety at Online Booksellers. MIT Sloan Working Paper No. 4305-03.

[4] Brynjolfsson, E. & Hu, JY. & Smith, MD. (2006). From Niches to Riches: Anatomy of the Long Tail. *Sloan Management Review, Vol. 47, No. 4, pp. 67-71, Summer 2006.*

[5] Canalys. (2007). 64 million smartphones shipped worldwide in 2006. Press release, February 12, 2007. http://www.canalys.com/pr/2007/r2007024.htm. Referred 12.3.2008.

[6] Canalys. (2008). Mobile device market 2007. Available at: http://www.canalys.com/pr/2008/r2008021.htm. Referred 20.7.2008.

[7] Elbers, A. (2005). Should You Invest in the Long Tail*? Harvard Business Review, July, 2008.*

[8] Ethan, R. (1999). The McKinsey Way: Using the Techniques of the World's Top Strategic Consultants to Help You and Your Business. McGraw-Hill Professional.

[9] Gibson, W. (2003). *Pattern Recognition*. G.P. Putnam's Sons, February 2003.

[10] Hippel, E. (2006). *Democratizing Innovation*. MIT Press, Cambridge, MA.

[11] Iftode, L. & Borcea, C. & Ravi, N. & Kang, P. & Zhou, P. (2004). Proceedings of the 10th International Workshop on Future Trends in Distributed Computing Systems, FTDCS 2004.

[12] Jackson, D. (2005). Competition and the Long Tail. Seeking Alpha, March 15, 2005. http://seekingalpha.com/article/1175-competition-and-the-long-tail. Referred 3.8.2008.

[13] Kilkki, K. (2007). A practical model for analyzing long tails. *First Monday, volume 12, number 5 (May 2007).* http://firstmonday.org/issues/issue12_5/kilkki/index.html. Referred 2.7.2007.

[14] Kivi A. (2006). Mobile Internet Usage Measurements: Case Finland. Master's Thesis Series. Networking Laboratory. Department of Electrical and Telecommunications Engineering. Helsinki University of Technology, Espoo, Finland, May 2006.

[15] McCarthy, K. (1994). Presentation: "The first conference on web commercialization". Available at http://video.google.com/videoplay?docid=-5046297730700144952. Ken McCarthy, Mark Graham, Marc Andreessen, San Francisco: November, 1994. Referred 1.7.2008.

[16] Medford, C. (2008). Apple, Nokia Battle for Mobile Startups. Red Herring, 10 July, 2008. http://www.redherring.com/Home/24459. Referred 12.7.2008.

[17] Nokia. (2006). Sharing discoveries - Nokia Nseries delivers the latest in mobile multimedia. Nokia Press Release, September 26, 2006. http://www.nokia.com/A4136002?newsid=1077763. Referred 12.3.2007.

[18] Shirky, C. (2003). Power Laws, Weblogs, and Inequality. Writings About the Internet. http://shirky.com/writings/powerlaw_weblog.html. Referred on 1.7.2008.

[19] Tallberg, M. & Hämmäinen, H. & Töyli, J. & Kamppari, S. & Kivi, A. (2007). Impacts of handset bundling on mobile data usage: The case of Finland. *Telecommunications Policy, vol. 31, pp. 648-659, 2007.*

[20] Verkasalo, H. & Hämmäinen, H. (2007). A Handset-Based Platform for Measuring Mobile Service Usage. *INFO: The Journal of Policy, Regulation and Strategy. Vol 9 No 1, 2007*.

[21] Verkasalo, H. (2007). Handset-Based Measurement of Smartphone Service Evolution in Finland. Journal of Targeting, Measurement and Analysis in Marketing; Special Issue; December 2007.

[22] Verkasalo, H. (2008a). Framework for the Strategic Analysis of the Mobile Internet Business. Presented in EuroCPR 2008, in Seville, Spain, March 30 - April 1, 2008.

[23] Verkasalo, H. (2008b). Summary of the Annual Finnish Smartphone Study 2007. ComNet / Helsinki University of Technology.

[24] Verkasalo, H. (2009). Open Mobile Platforms: Modeling the Long-Tail of Application Usage. Presented at ICIW 2009. The Fourth International Conference on Internet and Web Applications and Service. May 24-28, 2009, Venice/Mestre, Italy.

[25] Webodia. (2008). Mobile operating system. Accessible: http://www.webopedia.com/TERM/M/mobile_operating_system.html. Referred 13.7.2008.

Appendix A - Add-on application survey results

N: 606 Data: Finnish smartphone panel study 2007

Have you installed applications to the phone?

Yes, frequently	35 %
Yes, a couple of times	49 %
No	16 %

I used the following methods in installing applications to the phone? (multiple answers)
(from those who installed applications)

Downloaded from the Internet with computer	70 %
Transmitted via USB	66 %
Downloaded from the Internet with phone	57 %
Transmitted via Bluetooth	48 %
Device application market (Download!)	23 %
Operator provided	18 %
From friends, workmates, or family	18 %

How did you learn about the applications that you installed? (multiple answers)
(from those who installed applications)

Browser the Internet with computer	85 %
From friends, workmates, or family	42 %
Browser the Internet with phone	29 %
Device application market (Download!)	23 %
From operator	21 %

Why haven't you installed more applications to the phone? (multiple answers)
(from those who installed applications)

I have not found more interesting applications	62 %
Add-on applications are expensive	31 %
I did not know, what other kinds of applications are available	28 %
I am afraid of viruses	18 %
Finding of applications is difficult	18 %
Installation is difficult	10 %
Other reason	7 %

Would you be willing to receive advertising, if you got free applications and content in exchange?

Yes	24 %
Maybe	53 %
No	24 %

I consider the following actors important in the delivery of mobile services…

	Consider very important	Consider important
Mobile phone vendors (e.g., Nokia)	31 %	79 %
Telecom operators	26 %	75 %
Internet companies (e.g., Google)	10 %	45 %
Media companies (e.g., MTV3)	6 %	33 %

Discovering Points of Interests Through A Web 2.0, Location-based Architecture

Stefano Ferretti, Vittorio Ghini
Department of Computer Science
University of Bologna
Mura Anteo Zamboni 7, 40127, Bologna Italy
{sferrett, ghini}@cs.unibo.it

Abstract

We propose a distributed architecture for the support and the deployment of location-based, discovery services in mobile contexts. Web 2.0 technologies are utilized to locate point of interests for the mobile user. Such discovery is based on a mashup of services that queries different yellowpages-like sites, filters and merges the returned items, and exports the final list to be sent to the mobile client. A cross-layer networking technique is employed on the wireless leg. This offers to the user a seamless navigation, while moving through different and heterogeneous wireless networks. Importantly, our approach also balances the network traffic over multiple wireless networks. Results from a real experimental assessment confirm the efficacy of the proposed approach.

Keywords: Cross-layers architectures, Web 2.0, context-aware applications.

1. Introduction

Discovering Internet services through mobile devices while on the move is becoming a frequent occurrence for many users. The idea is to query context-aware services, by exploiting the ability of current devices of tracking the geographical location of the user [2],[3],[4],[5],[6]. The most prominent example of these types of applications consists in locating nearby places of interest, such as, for instance, restaurants, pubs, hotels, banking cash machines, shops.

The fact is that if one wants to make such applications really reliable and able to seamlessy operate, they should be accompanied with networking solutions able to offer always-connected services. In particular, due to the many available networking technologies, the diverse coverage of these networks, the different performances these may offer, the different costs to connect and exploit them, strategies would be needed to actively manage all available networks and dynamically create connections using only those that may optimize the communication (in terms of throughput, access fees, reliability, etc.).

Novel applications have been recently developed which allow to retrieve information on the geographical area where the user is currently located. This increasing market is further boosting the demand for novel location-based technologies able to provide detailed geographical data. Each of these exploits different localization techniques, ranging from the use of classic GPS, methods exploiting information on the reachable cells of cellular networks, Wi-Fi positioning systems, as well as hybrid solutions.

Google Mobile is probably the most prominent example of location-based application, which offers to users the possibility to interact with the Maps service of Google on mobile terminals [7]. Users may be located using EOTD (Enhanced Observed Time Difference) and TDOA (Time Difference Of Arrival) technologies. Other examples are various and simple applications on sale for cell-phones, e.g., AroundMe for iPhone [8]. Many of these applications, especially those specifically built for iPhone, primarily use the Wi-Fi Positioning System developed by Skyhook Wireless [9]. Using the information on MAC addresses of nearby access points, this system is able to locate the user with an error of 20-30 meters. The key of the mechanism is probably the database Skyhook manages, which includes more than 100 million Wi-Fi access points, with their related geographical position [10].

All these approaches offer viable solutions, which are exploited in real applications. There are some limitations, however. First, these applications must be installed on the device (i.e., they are not Web-based).

Second, they usually refer to a single repository of information to locate point of interests (e.g., Google

Maps), without resorting to the wide amount of information available on different Web services. In substance, these applications do not exploit at all Web 2.0 technologies and the mashup of Web services, which are instead widely utilized by applications offered to traditional (static) users, and have become fundamental to discover data and news in the wired Internet [11].

Third, the operating system and its implementation of networking protocols on mobile terminals usually impose the use of a single network for the communication. Hence, the typical nomadism of the user is not supported by an adaptive and seamless exploitation of different networks, in order to guarantee an always best connected service.

Based on these considerations, we have developed a cross-layer system for the dynamic discovery and localization of points of interest near the actual position of a given user [1]. The proposed approach is a mixture of Web technologies and cross-layer networking techniques. The user simply exploits a browser. A Web service mashup has been developed that, by exploiting Web 2.0 based techniques, queries different yellopages-like Web sites, which in turn are able to identify places of a certain type together with their addresses. The mashup is based on Dapper [12], Yahoo Pipes [13], Google Maps [14] and Reverse Geocoding [15].

We assume that the user has a Mobile Device (MD) equipped with multiple heterogeneous wireless network interfaces to connect to different access networks. This assumption is quite reasonable, having in mind the technical characteristics of current cell-phones and PDAs. In particular, in the presented scenario we assume that the MDs incorporate two or more Wi-Fi network adapters (IEEE802.11a/b/g/n) that offer connectivity on a local geographical scale, and one wireless network adapter that offers connectivity on a large geographical scale by using a long-range communication technology (e.g., GPRS/EDGE, UMTS, 3G/HDSPA, Mobile WiMAX).

When multiple network interfaces are available at a given mobile terminal, there are several important considerations worth of mention. On one hand, with respect to Wi-Fi local-range communication technologies, those that offer connectivity on a large geographical scale, such as GPRS/EDGE, UMTS, 3G/HDSPA, Mobile WiMAX and others, share the following disadvantages: i) a lower available bandwidth, ii) a larger energy consumption (especially during transmissions) and, iii) generally, an higher access fee.

On the other hand, the Wi-Fi technology provides connectivity only within a limited coverage range (150-300 m); hence, it is likely that a nomadic user that walks around the city will enter and leave different areas where different IEEE802.11a/b/g/n Access Points (APs) are active. As a consequence, a Wi-Fi network interface card may perform several handoffs. This continuous association and de-association to different APs may impose that the terminal changes its IP address. A main consequence of this undesired situation is that all previous running applications must be re-started or reconfigured.

The approach we have developed allows to overcome these limitations. It is based on the idea of concurrently exploiting different, heterogeneous and possibly independently managed wireless network infrastructures. This is accomplished through the use of a user-driven cross-layer architecture, that provides a simultaneous use of available wireless network interfaces on the mobile device. This allows to meet application QoS (quality of service) requirements (i.e., responsiveness, reliability, availability, continuity of the communications) and guarantee the viability of services offered by our mashup.

In view of the considerations above, the heuristics we use consists in exploiting the long-range communication technology only when strictly necessary, thus preferring short-range communication technologies, so as to i) limit the electromagnetic energy to which the user is exposed, ii) guarantee a longer battery duration, iii) limit the costs of the communications, and iv) exploit the overall bandwidth provided by the available APs, without lack of communications continuity.

It is important to notice that the use of an underlying service able to maintain a connection active, even in presence of horizontal or vertical handoffs, is of paramount importance for the success of the application we present in this work. Indeed, the mashup takes a while (20-30 secs) to query all different Web services, retrieve the information requested by the user from these different services, based on the user's geographical position, merge such data and encode it as a list of items based on the JavaScript Object Notation (JSON). During such time period, the connection between the client and the mashup service must remain open. Results from an experimental assessment we performed confirm the viability and the efficacy of the presented proposal.

This paper extends the work presented in [1] in several ways. A complete background section has been added that discusses host mobility and location-based services issues. A deep description of the design and the implementation of the proposed architecture is provided. Moreover, a set of experimental results that validate our proposal is reported with a related discussion.

The remainder of this paper is organized as follows.

Section 2 reports a state of the art of the existing systems which share the same final objectives of our approach. Section 3 describes the system architecture we devised. Section 4 discusses details related to the Web mashup built to dynamically discover points of interest near a given user. In Section 5 we analyze details on the cross-layer architecture. Section 6 reports on the experimental evaluation of the proposed system we performed. Finally, Section 7 provides some concluding remarks.

2. Background and State of the Art

In this Section, we review the state of the art with respect to location-based services that exploit features of Web 2.0, thus highlighting the possibilities which result from the use of location-based services into novel Web applications. Then, we survey the basic design principles shared among the most important architectures and protocols devoted to the mobility management. When describing all these proposals, a distinction is made to locate them in the ISO/OSI network stack (see Figure 1).

2.1. Web 2.0 Location-based Services

Location-based services represent the new frontier for novel Web 2.0 applications. There are several examples worthy of mention. We already cited the classic Web 2.0 applications that provide maps and related information on the geographical area where the user is placed. Google Maps, MapQuest, Yahoo! Maps are just few examples. The interesting thing is that these services allow 3rd party developers to exploit their features and contents, through the use of open APIs. This facilitates the creation of novel services.

When location-based systems are integrated with social applications, novel services can be provided that allow to track not only the position of the single user, but also that of his/her friends. Loopt, Pelago, Moximity, BrightKite, GyPSii, Citysense, Whrrl, Plazes are just few examples of applications. It is worth mentioning that these applications have not received the same interest that users show for the traditional social applications, e.g. Facebook and Twitter. Maybe, this is due to the fact a general user would prefer not divulgating his/her actual position to all his/her contacts.

Another interesting software technology is represented by GeoRSS and Geotagging techniques. GeoRSS exploit Really Simple Syndications (RSSs) for identifying a geographical data as a RSS feed. More specifically, GeoRSSs extend existing feeds with geographic information. This would allow, for example, to post an information in a blog about some point of interest and map directly the mentioned locations [16].

Geotags are used to add geographical information (e.g., longitude, latitude) to websites, images, RSS feeds, videos and more. The idea is that Web services, applications and users then can query this information to obtain directions, for instance. Geotags differ from a simple address in that they usually are encoded in metadata and are not visible as part of the Web page.

2.2. Mobility Architectures

The aim of a mobility architecture is to ensure a MD may move across access networks seamlessly accessing network services. Ideally, a mobility architecture is responsible for i) unequivocally identifying each given MD, ii) allowing each MD be reachable from its correspondent nodes, iii) monitoring the QoS channels to predict the need of a handoff, and iv) performing the handoff seamlessly, thus ensuring the continuity of the communications.

Almost all of the mobility management approaches follows the Always Best Connected (ABC) model [17] that suggests the selection of a preferred Network Interface Card (NIC) to be used as single point of access to the Internet, until the performance degrades too much. A two-steps configuration procedure is used: firstly, each MD's NIC detects the best network access point, from those available, and connects to it; in the second step, the MD elects its preferred NIC from those available and, from this instant, the MD uses the wireless network connected with the preferred NIC as its single point of access to the Internet. When the performance of the preferred NIC degrades, the MD detects a new preferred NIC (and the wireless network connected with it) that replaces the previous one, by means of a handover. This is the idea of networks integration that is at the basis of the so-called fourth-generation (4G) networks. This approach prevents the simultaneous utilization of all the MD's NICs and is therefore unable to take advantage, in terms of QoS and costs, of their different characteristics.

In IP-based communications, the MD's IP address plays the twofold role of MD's identifier, as it distinguishes uniquely the MD, and MD's locator, as it identifies the position of the MD in the network. When a MD moves in a different network, it cannot maintain its IP address; thus, the MD acquires a new IP address from the new network, loosing its identity; hence, it is forced to inform its correspondent nodes (CNs) of its new identity and location. More generally, a mobility management architecture must allow two hosts communicate between them even when both may move and change simultaneously their addresses. In fact, in such a situation, each MD would be unable to contact its CN as it does not know the CN's new IP address.

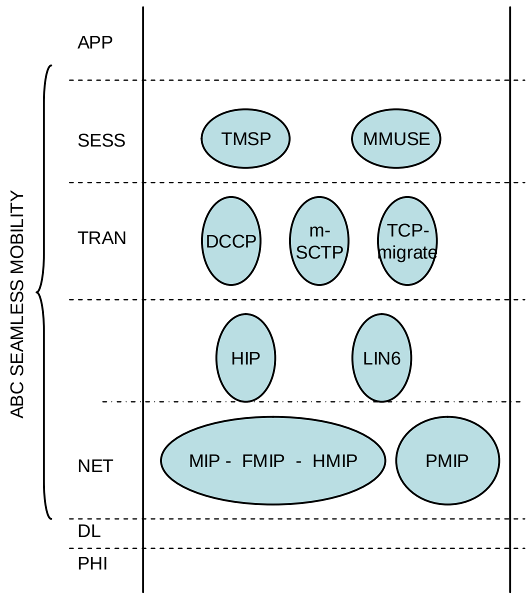

Figure 1: Architectures for mobility management

To solve this problem, all the mobility management architectures adopt, at different architectural layers, a general solution based on two principles: i) defining a unique MD's identifier, independent from the MD location, and ii) providing a localization service, always reachable from the CN, that maintains a mapping between MD identifier and MD current locator, even when the MD moves. Such a location service is usually composed of a sort of Location Registry (LR) working on a server having an IP address which is static, public and known. When the MD changes its IP address, it informs the LR with a registration phase. This way, since a CN knows the MD's unique identifier, when it wants to initiate a new communication with the MD or it wants to continue a communication with the MD that has chanced its address, the CN starts a lookup phase and asks the location registry for the MD's current address. After that, CN may use the obtained MD's address to directly contact it.

Network-layer mobility architectures can be classified in two categories, host-based and network-based, depending on who is the entity that manages the signaling. Host-based architectures, such as Mobile IP version 6 (MIPv6) [18] and its optimizations Fast Mobile IPv6 (FMIP) [19] and Hierarchical MIPv6 (HMIP) [20] add an entity in the architecture, referred as the Home Agent. Such agent acts inside the access network from which the MD belongs, and basically plays the role of LR. A mobility protocol stack, implemented within the MD, is used to generate and manage the mobility session. Moreover, these MIPv6-based approaches impose that both end-systems have IPv6 capabilities, in order to insert in the IP datagrams some extension headers that transport the MD's identifier (i.e., the home address) and the current MD address.

Differently, network-based mobility architectures such as Proxy MIPv6 (PMIP) [21] do not require that the MD actively participates to the mobility support signaling. Usually, these solutions specifically work in a local domain. An entity called local mobility anchor (LMA) analogous to the Home Agent plays the role of LR. In addition, a mobility access gateway (MAG), at the edge of each access network inside the local domain, performs the mobility support signaling instead of the MD. However, if the MD leaves this domain, the established sessions break because the MN would not be anchored in the local topology anymore.

Of course, all these network-layer solutions require that the network infrastructures support IPv6 capabilities. Moreover, the MIPv6 specification does not allow the simultaneous use of the multiple MD's NIC. For each given MD, the address of one NIC is only registered to the Home Agent. In addition, as demonstrated in [22], the handover latency of a MIPv6-based solution typically results higher than the handover latency of a transport-layer solution, due to the numerous authentication messages necessary in the MIPv6 registration phase (binding update).

In the architectures located between the network and the transport layers, such as Host Identity Protocol (HIP) [23] and Location Independent Addressing for IPv6 (LIN6) [24], the location registry is a DNS-like mapping function that operates as a service outside the access networks and associates host identifiers and host locators. HIP, LIN6 and similar solutions insert an intermediate layer between the network and transport layers on both the MD and CN. However, the modification on the CN appears not eligible because it may be a fixed node that could be not interested in supporting the mobility of the MD.

The common approaches working at the transport-layer protocols, such as the datagram-oriented Datagram Congestion Control Protocol (DCCP) [25], the stream-oriented mobile Stream Control Transmission Protocol (m-SCTP) [26] and the TCP enhancement TCP-migrate are very different: each given end-system plays the role of a proactive location registry that informs the CN of its configuration changes. Unfortunately, this approach fails when both the end-systems change simultaneously their IP configuration, because the two end-systems become mutually unreachable. The transport-layer solutions require that the applications are modified on both the MD and CN to invoke the services of the novel transport layer and to implement the suitable recovery policies. This prevents the reuse of the existing applications.

The most important session-layer mobility management architectures make use of an external Session Initiation Protocol (SIP) server. SIP [27] is a session-layer text protocol that uses a message/response handshake for signaling purposes. In particular, it is used to establish or change communication parameters such as IP addresses, protocol ports and audio/video codec between the end-systems. The SIP specification allows to add application-defined fields to the SIP messages. Unrecognized extension fields will be simply discarded. When a given user U1 is available for communications, a REGISTER SIP message is sent to a SIP server that will maintain the address of the user U1 on the node N1. To initiate a call, a user U2 on the correspondent node N2 asks the SIP server, with a INVITE message, for the address of U1 and for some communication parameters. After this exchange, the two nodes may communicate directly. A REINVITE message may be used when communication parameters (such as IP address) change. The SIP protocol allows the presence of SIP proxies that can be transparent to the application (proxy agent) or can masquerade the end systems (back-to-back agent) working as an opaque relay.

Session-layer architectures, such as Terminal Mobility Support Protocol (TMSP) [28], use an auxiliary SIP server, outside the access networks, as location registry that maps a user's URI (e.g., ghini@cs.unibo.it) to the current user's location (the IP address of the user's MD). Each MD uses a SIP user agent that sends REGISTER messages to the SIP server in order to update its current location on the SIP server and INVITE messages to establish communications with the other nodes. The session-layer solutions seem to be not efficient because when an IP reconfiguration happens they require the invocation of an external localization service. In particular, the SIP-based services introduce an additional delay. This is due to the fact that when a reconfiguration of the communication parameters happens, the MD interrupts the communication, sends a SIP signaling message to the CN and waits for the response before to resume the transmission.

The MMUSE [29] approach proposes significant differences: it requires that an auxiliary SIP server (namely, the Session Border Controller, SBC) is located at the edge of the autonomous system inside which the MD will move. This autonomous system may be composed of several subnets using heterogeneous network technologies. While the MD moves across the subnets, each subnet provides the MD with a different IP address. The SBC aggregates the functionalities of SIP and Real Time Protocol (RTP) proxy, firewall and network address translation (NAT) systems, and intercepts the communications that enter and leave the network, in particular the SIP messages between the MD and its CN outside the network edge. Based on the outgoing SIP messages, the SBC sets the firewall rules to allow the subsequent SIP, RTP and Real Time Control Protocol (RTCP) communications. Moreover, when the MD moves to a different subnet and changes its IP address, the SBC modifies the outgoing datagram in order to hide to the CN the current location of the MD.

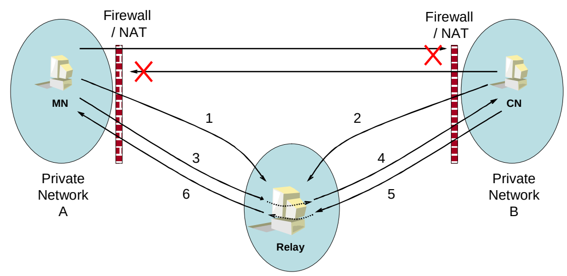

Figure 2: The data relay allows the undirected communication between two nodes behind different firewalls or NAT systems.

An important consideration is that, generally, all the described mobility management solutions do not take into account the possible presence of firewalls and NAT systems, and rely on external STUN (Simple Traversal of UDP Through Network Address Translators) [30] or TURN (Traversal Using Relay NAT) [31] servers (see Figure 2). The main related problem is that this solution introduces additional communication latencies. As we will show in the next Sections, the use of proxies in our system creates a tunnel that easily surmounts this problem.

Finally, it is worth to point out that all the described solutions do not enable the simultaneous utilization of all the NICs available on the MD. Conversely, our approach is able to simultaneously utilizing multiple wireless networks. Therefore, besides allowing a seamless communication, our approach balances also the network traffic over multiple wireless networks.

3. The Architecture

Our architecture is composed of three main modules (see Figure 3), i.e., i) a Web 2.0 mashup which offers the Web functionalities to query, discover points of interest and show them on a map, ii) a geo-localization

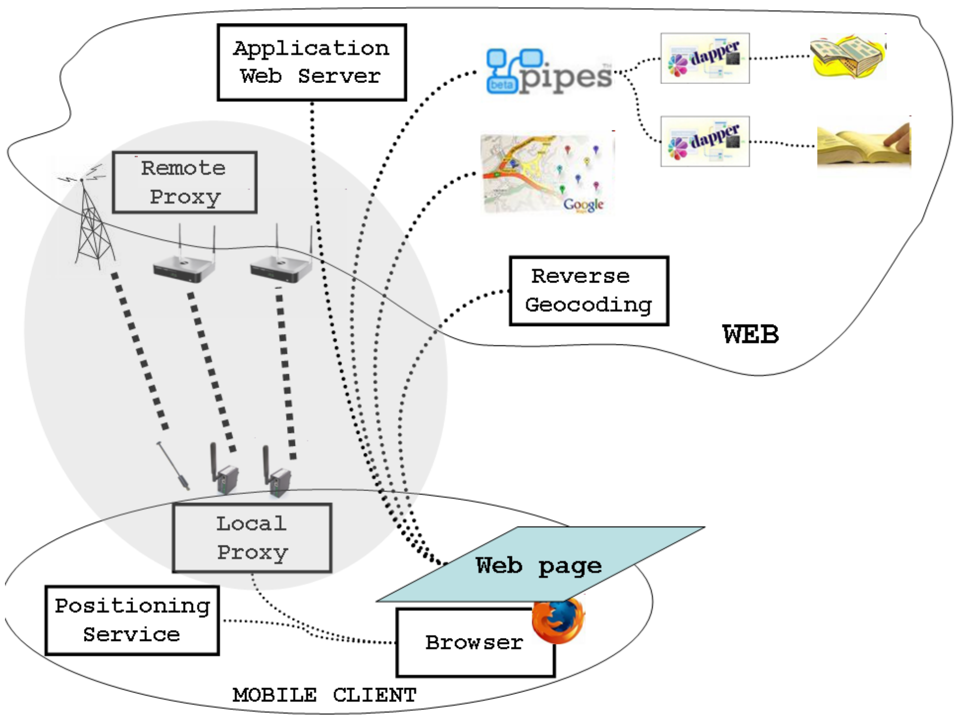

Figure 3: System Architecture

scheme, iii) a cross-layer networking service which offers a seamless communication between the mobile client and the rest of the world.

A Web service has been built that acts as the orchestrator of the mashup. As mentioned, our application is based on Web technologies, and runs on a browser. Javascript is thus used as the language to develop procedures to be run on the browser.

To enable the localization of the user, different existing options are possible. Browser plugins have been developed which allow to interact with GPSs, or Wi-Fi interfaces, in order to retrieve the position of the user. An examples is the Geode plugin recently developed at Mozilla Labs [32]. Instead, Loki is the Skyhook's browser plugin that uses Wi-Fi positioning system to enable 3rd party websites to integrate auto-location using javascript APIs. Other approaches are the Google APIs developed for Android which allow to access the data returned by a built-in GPS receiver, the system presented in [33], or more simply the use of some local Web service installed on the client terminal (if the user agrees), in charge of interacting with the GPS device and able to export such data. Alternatively, if the user prefers not to install such a kind of services on the client machine, he/she will be asked to insert his/her location manually.

A local proxy is installed on the client device, in charge of communicating with a remote proxy for ensuring a seamless navigation. It exploits several wireless network interfaces and a cross-layer networking scheme which guarantees communication even in front of horizontal or vertical handoffs.

Based on these modules, the interaction of the user is as follows. (The whole interaction flow among the distributed system modules of our architecture is depicted in Figure 4. In that Figure, MD represents the mobile device, WS represents the Web Server, RG represents the Reverse Geocoding service, Gmaps stands for Google Maps, while YP_i are the contacted yellopages-like sites.)

1. The client activates the local proxy, which connects to the remote proxy.
2. If some positioning system is available on the mobile terminal, this module should be ready for execution at the client.
3. Once the user wants to discover some points of interest, he connects to the Web application we developed (i.e., interaction between MD and WS in Figure 4). The communication and the whole interaction is performed using the cross-layer networking scheme as the underlying communication service.
4. Through the Web page retrieved at the application Web service, the user can select the typology of

places he is interested in (e.g., pubs, music shops, museums).

5. Once the selection is made, a Javascript procedure verifies if a local positioning mechanism can be utilized. If it is available, the geographical position of the user is retrieved; otherwise, the user is asked to insert such position manually.
6. If GPS coordinates are returned by the local positioning system, a reverse geocoding service [15] is queried to identify the address (i.e., street, city, country) corresponding to the geographical position (i.e., interaction between MD and RG in Figure 4).
7. Based on the type of points of interest and on the current location of the user, a mashup is activated that queries different Web services and analyzes the obtained results, which are finally shown to the user in a map, taken from Google Maps [14]. (In Figure 4, more details are depicted which describe how the mashup works; a related and more detailed description is provided in the next Section.)
8. Upon selection of a specific item by the user, a path from the current position to the selected point is shown (again, using the Google Maps service).

The following sections are devoted to present in deeper detail the Web 2.0 mashup and the cross-layer networking system, which represent the main technical novelties of our proposal.

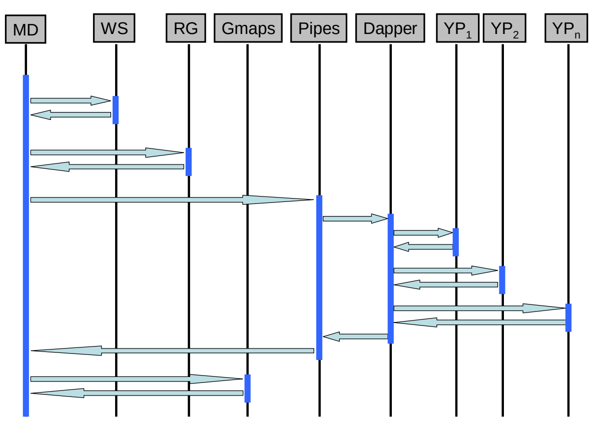

Figure 4: Interactions Flow, Application Level

4. The Web 2.0 Mashup

The developed Web 2.0 mashup module exploits HTML, Javascript RSS and JSON technologies to let the distributed entities interoperate and exchange data on the points of interests sought by the user. Four services to create the mashup have been utilized, i.e., Dapper, Yahoo Pipes, Google Maps, and a reverse geo-coding service. As mentioned, services are orchestrated by the javascript procedures implemented within the Web page retrieved by the client, and run by the browser, as shown in Figure 3.

The reverse geo-coding service is simply exploited to perform a conversion from the geographical position of the user (encoded as GPS coordinates) to a common address composed of street, city, country. This service is locally called by the browser through a Javascript procedure in the Web page retrieved by the client.

Dapper is a Web-based service that enables to extract information from any Web site and create an interactive feed from it. Data can be extracted by creating a Dapp, i.e., a Web site needs to be selected, types of data in the Web pages that should be extracted needs to be specified, together with its output format. Once properly instructed on a specific Web site, the Dapp extracts data based on the site's HTML structure. This allows to automatically query a yellowpages site and export data based on different encoding formats, e.g., RSS, XML, JSON. The automatic query can be done by building a proper URL containing the parameters of the query.

Take, for instance, a template of a query passed to the server of the Italian yellowpages (i.e., `www.paginegialle.it`). Such query corresponds to an URL such as

`http://www.paginegialle.it/pgol/4-`***`[TYPOLOGY]`***`/3-`***`[CITY]`***`?ind=`***`[ADDRESS]`***

where ***`[TYPOLOGY]`*** corresponds to the type of point of interest, ***`[CITY]`*** corresponds to the city and ***`[ADDRESS]`*** to the address where the user is currently located. This means that if one wants to look for a pub in Bologna, near the address "via Rizzoli", it is sufficient to pass to the Dapp a URL which looks like

`http://www.paginegialle.it/pgol/4-`***`pub`***`/3-`***`Bologna`***`?ind=`***`Rizzoli`***.

Then the Dapp will analyze the Web page returned by the server `www.paginegialle.it`, extracts data and encode them as an RSS feed.

In our prototype, we created two different Dapps, i.e., one to query the Italian version of the yellowpages site (`www.paginegialle.it`) , and another which queries a popular Web site to find night clubs and entertainment sites (`www.2night.it`). Both Dapps export the list of items as RSS feeds, which are passed to the next mashup module, created using Yahoo Pipes.

Yahoo Pipes is Web 2.0 service which allows to create a mashup of services by composing them as one or more pipes. During the development of the pipe, a graphical interface is provided to orchestrate these different services. By exploiting it, a mashup has been created to call different Dapps, retrieve output RSSs coming from them, filter and merge these feeds into a unique list, and finally export them as a JSON-encoded document which is passed to the client. Each element of the list is a point of interest with its related address.

Based on this list, the Javascript executed at the browser creates a map (using Google Maps) with all points of interest displayed in it.

5. The Cross-Layer Architecture

The cross-layer networking service (depicted in Figure 5) offers a seamless communication. It is composed of two main entities, a local proxy installed on the MD and a remote proxy, installed in a fixed host, with a static IP address. The local proxy is composed of a Load Balancer Client (LBC) and of a Monitor, that dynamically configures the wireless network interfaces of the MD by selecting the best access points. The remote proxy is composed of a Load Balancer Server (LBS) and a HTTP proxy.

The Monitor and the two load balancers are general-purpose components we have developed (in a context completely different from that described in this paper). An accurate and deep description of the design and implementation of these components can be found in our previous works, in particular the architecture of both Monitor and Load Balancers is described in [36], the latest release of the load balancer algorithm is evaluated in [35] and the mechanism dedicated to the bandwidth estimation of the Wi-Fi access points is detailed in [37]. In this context, instead, we aim in highlighting the interactions among these components and the overall architecture.

Moreover, it is important to point out that the seamless communication service provided by the cross-layer architecture operates as a separate session-layer service implemented in the two end-systems only, the MD and the fixed host. The service operates on top of (and as a complement of) the existing IP-based network infrastructures. Thus, it can coexist with other architectures such as, for instance, 3GPP.

Basically, the service maintains the abstraction of an always available TCP connection between a Web browser in the MD and the HTTP proxy in the remote fixed host, despite of handoffs and changes of MD IP addresses. The service also balances the load among the different MD network adapters and recovers connection faults, but using the long-range communication technology only when strictly necessary (i.e., when no Wi-Fi APs are available).

The Web browser is simply configured to use the LBC as its proxy, so as to deliver/receive the HTTP request/response to/from the LBC through a TCP connection. For each connection with the Web browser, the LBC maintains a session-layer multi-path communication channel (MPCC), through which the HTTP messages are transmitted in both directions, as illustrated in Figure 4. This session-layer MPCC consists of a dynamic set of Secure Socket Layer (SSL) [34] connections, one for each working MD wireless adapter, that guarantee the privacy of the transmitted data. Each SSL connection is supported by an exclusive TCP connection, which binds its local IP address to the address of one of the working MD wireless network adapters, and sends and receives TCP segments through this wireless adapter only. For each MPCC between the LBC and the LBS, the LBS maintains a TCP connection with the HTTP proxy.

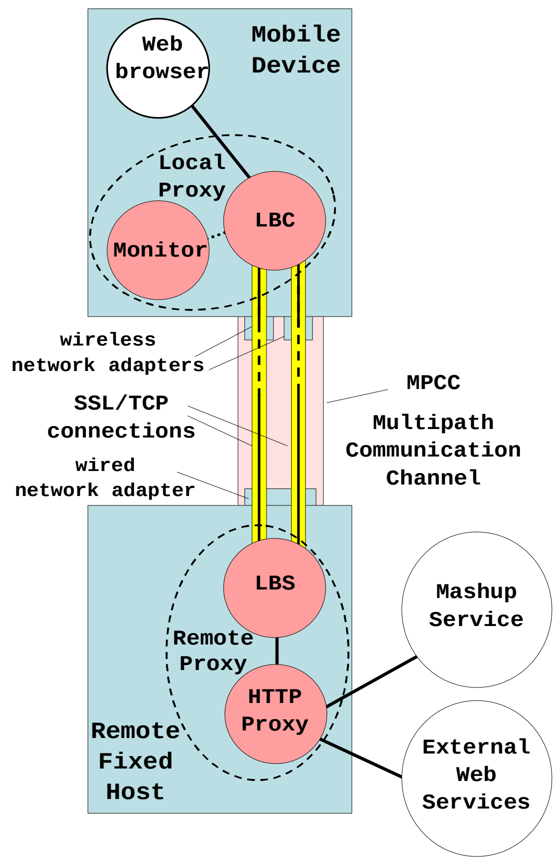

Figure 5: Architecture for Seamless Navigation

The HTTP requests produced by the Web browser reach the LBC as a TCP flow; the LBC splits that flow and delivers the pieces through the MPCC, retransmitting the pieces when necessary. The algorithm to balance the network traffic is described in full details in [35]. In substance, the load balancing algorithm estimates the performance of each MPCC's TCP connection bound to each NIC. The

LBS receives the pieces, reconstructs the original TCP flow and delivers it to the HTTP proxy that forwards the request to the destination. Collected HTTP responses follow the same path but in the opposite direction.

The Monitor at the MD is responsible for the dynamic configuration of the datalink, network and session layers of the MD (see [36] and the next subsection for more details), in particular for the Wi-Fi NIC. Using cross-layer information, the Monitor detects when a NIC leaves the coverage area of the associated AP. In this case, the Monitor informs the LBC that closes the SSL/TCP connection (of the MPCC) that uses that NIC. Then, the Monitor drives the NIC in scanning the Wi-Fi channels to detect the available APs and to estimate the traffic they are subject to, so as to exclude the APs that provide marginal signal strength. Following a model described in [37] that take into account both the traffic and the signal strength, the Monitor estimates the bandwidth each AP may provide to the NIC, excludes those APs that are associated with the other NICs of the MD itself, and selects the AP that provides the higher bandwidth (the interested reader may refer to [37] for a detailed description of the model). Finally, the Monitor associates the NIC to the selected AP, configures the IP address and the routing rules for the NIC, and informs the LBC of the newly available NIC. The LBC creates a new SSL/TCP socket bound to that NIC, connects the socket to the LBS, and inserts in the session-layer MPCC the new SSL/TCP connection between LBC and LBS.

The described procedure guarantees that the MPCC always maintains and uses, for load balancing purposes, an SSL/TCP connection for each working wireless network adapter of the MD.

All the described software entities (LBC, LBS, HTTP proxy) have been implemented at the application layer and do not need particular operating system features. The only exception is the implementation of the Monitor, that is composed of several separates applications, and relies on some features and APIs of the GNU Linux operating system, in particular those that enable the communication with (and the configuration of) the datalink and network layers of the MD. For instance, the Linux Wireless Extensions API for the traffic analysis and the datagram-oriented Netlink socket to manage the dynamic routing tables [38].

The next subsection will describe in more detail how the monitor works.

5.1. Monitor Implementation

The MD is governed by the Linux operating system, and its software has been designed, for backward compatibility purposes, so as to allow one to use old-style Linux kernels, such as those of the versions 2.4.x (the most recent version is 2.6.30, at the time of this writing). This backward compatibility enables the implementation of the MD even using PC boards and handheld devices running old Linux kernel versions. Moreover, the Linux kernel can be configured so as to manage multiple routing tables at the network layer. Such a configuration allows this kernel to perform dynamic routing of IP packets based not only on the packet destination address but also on its source address. Note that one such a configuration can be generated easily by simply compiling the Linux kernel source code with a specific flag appropriately set in the makefile.

The software running in the MD is structured as depicted in Figure 6. The shaded boxes in this Figure represent the MD software components that we have designed. At the application layer, the LBC receives the data flow through an HTTP/TCP connection with the Web browser and delivers it to the LBS through the MPCC.

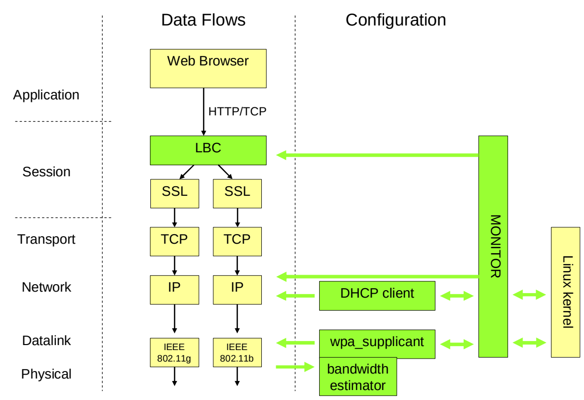

Figure 6: Mobile Device Software Architecture

The Monitor communicates with the Linux kernel by means of a datagram-oriented Netlink socket: when a network adapter changes its status, the kernel informs the Monitor, which in turn activates the adapter reconfiguration procedures. Moreover, the Monitor communicates with the session-layer LBC (see below) to inform it that a wireless network adapter has been completely configured and can be used.

The lowest software layer, i.e., the data link layer, implements the operations that allow each wireless network adapter in the MD to detect the presence of an access point and to configure its own data link layer so as to communicate with that access point. In addition, the data link layer is responsible for managing the network adapter that looses the access point carrier:

that adapter cannot communicate and thus must be disabled.

All these operations have been implemented as a separate process, reusing a widely used, free and open-source application called wpa_supplicant. Wpa_supplicant is not dependent on the wireless technologies being deployed as it makes use of services provided by the wireless adapter driver. For this reason, the use of wpa_supplicant allows the MD to accommodate network adapters of wireless technologies different from IEEE802.11.

As concerns authentication and security of wireless networks, wpa_supplicant encloses all functionalities able to cope with them. As to the authentication functionality, it requires the use of a configuration file in the MD, which contains the information necessary to enable wpa_supplicant authentication in each visited network. This information depends on the kind of authentication a visited network requires; for instance, WEP is the simplest (and less secure) system; it is based on the knowledge of a key shared among all users. The access point asks wpa_supplicant for the name of the wireless network (the so-called Extended Service Set Identifier – ESSID) and the shared 128 bit WEP key. In contrast, other systems require external authentication servers, such as those based on Robust Security Network (IEEE802.11i [39]), IEEE802.1X [40], Extensible Authentication Protocol (EAP) [41] and Remote Authentication Dial In User Service (Radius) [42].

Obviously, the user MD may access only to the access networks of which the user possesses the necessary permissions. The user is responsible to acquire the required authentication information from each access network's administrator and to write the configuration file required by the wpa_supplicant.

As to the security functionality, wpa_supplicant hides the complexity of different security mechanisms. Simply stated, in the IEEE 802.11 scenario, wpa_supplicant drives the wireless adapter in scanning the radio channels to detect the access points of the wireless networks of which it knows the authentication information. Actually, when more access points are available, the original wpa_supplicant selects the one that provides the best radio signal, and executes the authentication procedure with that access point. Hence, we have implemented two minor modifications of the wpa_supplicant access point selection process. The first one consists in a preliminary bandwidth estimation phase to identify the access point that is able to provide the MD with the higher bandwidth. This phase is implemented using the Linux Wireless extensions API and is based on a initial traffic monitoring and on a traffic model [35] that take into account traffic, error rate and signal strength. The second modification ensures that additional network adapters of a given MD select unique access points to provide the MD with multiple independent wireless paths.

The network layer implements those operations that configure the IP address, the netmask and the gateway IP address of the wireless network adapters, and sets up the rules for the dynamic routing of the IP datagram via those adapters. These operations are executed after wpa_supplicant has configured the data link layer of a wireless adapter, by associating it to a given access point. In fact, the kernel informs the Monitor as soon as one such an association is established; then the Monitor starts a Dynamic Host Configuration Protocol (DHCP) client. The DHCP client requests an IP address, a netmask and an IP gateway to the DHCP server of the visited network. When the DHCP client terminates successfully the configuration of the network layer adapter, the Monitor sets up a new rule and a new routing table (invoking the "ip rule" and "ip route" commands) in order to enable the dynamic routing via this adapter. These rule and routing table impose that the IP datagrams having the same source IP address as this adapter be routed through it, regardless of their IP destination address. As this adapter is configured, it can be used by the session layer. In this way, the IP datagram of the TCP connection bound, by the LBC, to a given adapter are routed and transmitted through that adapter.

When the network adapter looses the access point carrier, the adapter's driver interrupts the kernel. This contacts the Monitor to disable that adapter and delete the routing rule for this adapter. Hence, the adapter will be unavailable until it is reconfigured.

6. Experimental Evaluation

In this section we study the behaviour of our developed application, based on a qualitative and quantitative evaluation. Hence, first, we show some screenshoots obtained as a result of the use of the application. Second, we report on measurements we obtained to access the system when the user passes through different networks coverage.

Figure 7 shows the result of a query process. In particular, the position of the user is shown with the flag in the map, thank to the use of Google Maps. The block reported below the image shows a list of points of interest, which are located near the user, found thank to the mashup. Figure 8 shows the result obtained when the user selects one of the items reported in the list. The path to reach the interest point is shown.

The objective of the experimental evaluation was to assess the ability of our architecture to both guarantee the continuity of the communication and deliver the application data within an acceptable level of interactivity, in spite of MD movements and changes of access networks. To this end, the experiments concerned the performances of the MPCC that connect a given MD with the Remote Fixed Host (RFH), in particular the elapsed time necessary to deliver, from the MD to the RFH, each application frame packet (1126 Bytes) with a frequency of 25 frames/second.

Figure 7: A screenshot of the result of a query

Figure 8: Path to reach the selected point of interest

The experimental scenario was as follows. A human operator, carrying an MD, has covered a route of 80 meters, as depicted in Figure 9. The MD is equipped with two IEEE 802.11g wireless interfaces. Along that route two IEEE 802.11g access points were available, indicated with AP1 and AP2 in Figure 9. The access point AP1 worked on channel 1, and was located near the beginning of the MD path, whereas the access point AP2 worked on the channel 6 and was located close to the end of that path. The transmission signal of these two APs was strongly and abruptly obscured by walls. Figure 10a illustrates the signal power received from the two APs along the path, due to the obstacles; it is worth to point out that below a threshold of -80dBm the AP becomes practically unavailable. In other words, the access point AP2 was unavailable at the beginning of the MD path whereas the access point AP1 was unavailable at the end of the path.

Moreover, both the access points were affected by an additional background traffic, of approximately 5Mbps, generated by two others static devices, and there were other three APs to cause inter-channel interferences.

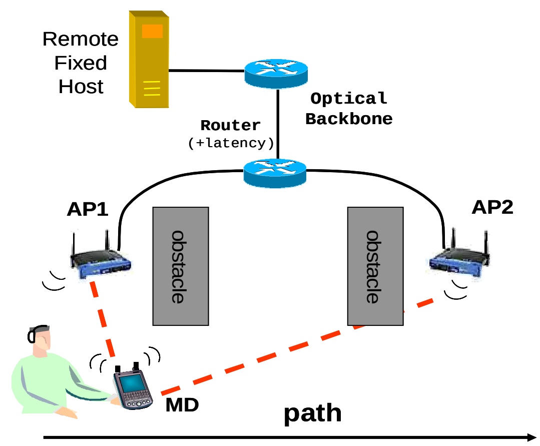

Figure 9: Experimental scenario

The two APs (AP1 and AP2) were connected to the RFH through a 1000Mbps Gigabit Ethernet network and an optical fibre backbone, and a Linux-based router was set to introduce an artificial latency of 50 ms, with a tolerance of 7%. This choice allows us to analyze the system behaviour in a realistic scenario.

The frame transmission has been carried out, in the following three modes, while moving along the described path: in the first two modes we have used the access point AP1 and access point AP2 in isolation, respectively, in order to show the performance of each single access point along the route of our experimental scenario. In the third mode, our system exploits both the access points for the frame transmission.

Figures 10b, 10c and 10d illustrate the frame delivery time we have obtained using AP1, AP2 and both together with our system, respectively.

In these three figures the x axis represents the frames sent from the MD while the y axis is the delivery time of each given frame. Close to the end of the MD path, the access point AP1 firstly introduces considerable delays and then it becomes unavailable, thus the frames do not reach the RFH and there are no delivery times, as

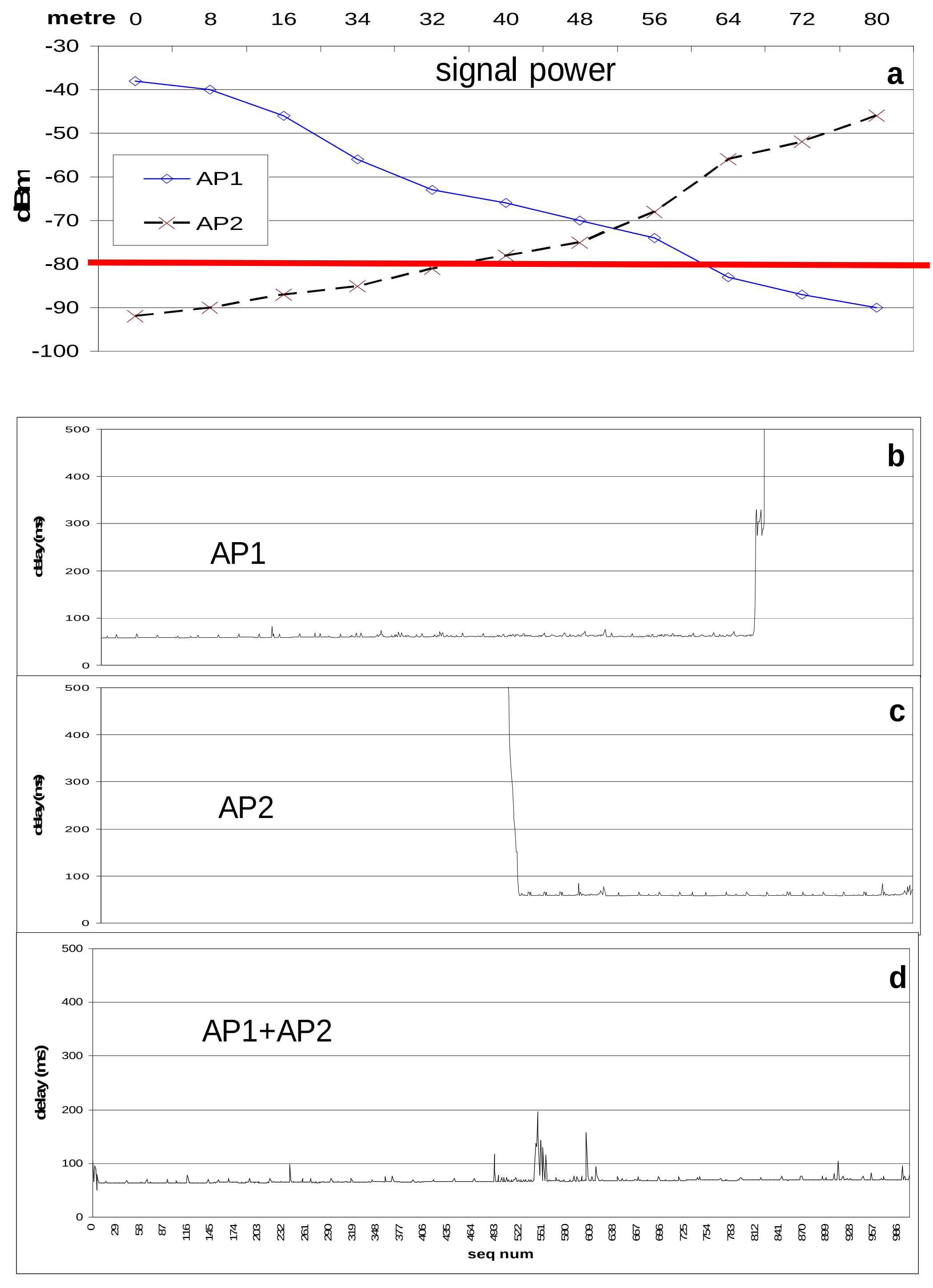

Figure 10: Experimental scenario and measures

illustrated in Figure 10b.

On the contrary, the access point AP2 (see Figure 10b) is unavailable at the beginning of the MD path and introduces small delays at the end of the route. Instead, our proposed system selects the suitable access point along the overall MS route, delivering all the frames to the destination. As depicted in Figure 10d, due to the retransmission, there are some peaks of the frame delivery time, but almost all the frames do not overcome the threshold of 150 ms, and only few (2 over 1000 frames) reach the destination after 150 ms but before 200 ms. Our system introduces just a little overhead: in fact, the amount of bytes retransmitted is small, i.e. 23 frames on a total of 1,000 frames. These results confirm that the proposed system provides responsiveness, reliability, continuity of the communications, and guarantees the viability of services offered by our mashup service.

It is worth to point out that performances do not worsen even when a number *n* (*n*>2) of access points are available in the route. In fact, when a given NIC loses the wireless carrier, the Monitor uses the bandwidth estimator described in [37] to estimate the bandwidth of each available APs. Then, it associates that NIC to the AP with the highest available bandwidth. The same procedure is executed every time the NIC leaves the coverage area of its associated AP.

Finally, the heterogeneity of the NICs on the MD does not represent a difficulty. In fact, the two load balancers abstract from the underlying network technologies and treat each communication passing through a specific NIC as a standard TCP communication channel between them. Moreover, the two load balancers estimate the performances of each TCP connection in terms of latency and bandwidth. This allows to balance the traffic workload among all the available NIC independently of the NIC technologies.

7. Conclusions

We have presented a system for a location-based discovery of points of interest. We have demonstrated that the concurrent use of Web 2.0 technologies and cross-layer networking schemes can guarantee a seamless use of sophisticated Web services, while moving through different and heterogeneous wireless networks.

The key for offering a seamless interaction among the mobile terminal and the Web services deployed over the Internet is the use of a couple of proxies. One is locally installed on the mobile terminal, while the other is made available on the Internet to all wireless devices. This solution enables wireless devices to exploit different networking technologies while being involved in a single session. The experimental results we obtained confirm such claim.

8. References

[1] S. Ferretti, V. Ghini, "A Web 2.0, Location-Based Architecture for a Seamless Discovery of Points of Interests", Telecommunications, 2009. AICT '09. Fifth Advanced International Conference on , pp.226-231, 24-28 May 2009.

[2] R. Lin, Y. Zhao, H. Zou, F. Yang, "An Enhanced Location Service with Social Feature", Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on 12-14 Oct. 2008, 1 - 4.

[3] J.C. Kim, T.W. Heo, J.W. Kim, J.H. Park, "Ubiquitous location based service", Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE, Sept. 2005, 841 - 845.

[4] D. Mohapatra. S.B. Suma, "Survey of location based wireless services", Personal Wireless Communications, 2005. ICPWC 2005. 2005 IEEE International Conference on 23-25 Jan. 2005 358 - 362.

[5] C.T. Wu, H. Mei, "Location-based-service roaming based on Web services", Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on, March 2005, Volume: 2, 277- 280.

[6] P. Bellavista, A. Küpper, S. Helal, "Location-Based Services: Back to the Future", *IEEE Pervasive Computing* 7, 2 (Apr. 2008), 85-89.

[7] Google Mobile, November 2008, http://www.google.com/mobile/index.html.

[8] Around me application, http://www.tweakersoft.com/mobile/aroundme.html, 2008.

[9] Skyhook Wireless Web Site: http://www.skyhookwireless.com/

[10] "Skyhook powers Apple's new location apps", The Journal of New England Technology, January 2008, (last checked, July 2009), available at: http://www.masshightech.com/stories/2008/01/14/daily20-Skyhook-powers-Apples-new-location-apps.html

[11] S. Ferretti, P. Salomoni, M. Roccetti, S. Mirri & L.A. Muratori, "At the Crossroads of Web and Interactive Multimedia: an Approach to Merge the Two Realms", Proc. CCNC 2009, Las Vegas (USA), IEEE, January 2009.

[12] Dapper, November 2008, http://www.dapper.net/.

[13] Yahoo Pipes, November 2008, http://pipes.yahoo.com/pipes/

[14] Google Maps, November 2008, http://maps.google.com/

[15] Reverse Geocoding Webservice, November 2008, http://www.geonames.org/export/reverse-geocoding.html

[16] GeoRSS Web site: http://georss.org/

[17] E. Gustafsson and A. Jonsson, "Always Best Connected", IEEE Comm. Mag., vol. 10, no. 1, Feb. 2003, pp. 49–55.

[18] D. Johnson, C. Perkins, J. Arkko, "Mobility support in IPv6", RFC 3775, June 2004.

[19] R. Koodli, "Fast Handover for Mobile IPv6", IETF RFC 4068, July 2005.

[20] H. Soliman, C. Castelluccia, K. ElMalki, L. Bellier, "Hierachical Mobile IPv6 (HMIPv6) Mobility Management", IETF RFC 5380, Oct. 2008.
[21] S. Gundavelli, V. Devarapalli, K. Chowdhury, B. Patil, "Proxy Mobile IPv6", RFC 5213, August 2008.
[22] Dong Phil Kim, Seok Joo Koh, "Analysis of Handover Latency for Mobile IPv6 and mSCTP", Communications Workshops, 2008. ICC Workshops '08. IEEE International Conference on , vol., no., pp.420-424, 19-23 May 2008.
[23] R. Moskowitz, P. Nikander, "Host Identity Protocol (HIP) Architecture," IETF RFC 4423, May 2006.
[24] F. Teraoka, M. Ishiyama , M. Kunishi ,"LIN6: A Solution to Multihoming and Mobility in IPv6", IETF Internet Draft, draft-teraoka-multi6-lin6-00.txt, Dec. 2003.
[25] E. Kooler et al., "Datagram Congestion Control Protocol (DCCP)", IETF RFC 4340, March 2006.
[26] M. Riegel, M. Tuexen, "Mobile SCTP," IETF Internet draft, draft-riegel-tuexen-mobile-sctp-09.txt, Nov. 2007.
[27] J. Rosemberg et al., "SIP: session initiation protocol", IETF RFC 3261, June 2002.
[28] Teck Meng Lim, Chai Kiat Yeo, Francis Bu Sung Lee, Quang Vinh Le, "TMSP: Terminal Mobility Support Protocol," IEEE Transactions on Mobile Computing, vol. 8, no. 6, pp. 849-863, June 2009.
[29] S. Salsano et al., "SIP-based mobility management in next generation networks", IEEE Wireless Communications, pp. 92-99, April 2008.
[30] J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", RFC 3489, March 2003.
[31] J. Rosenberg, R. Mahy, C. Huitema, "Traversal Using Relay NAT (TURN)", Internet-Draft, draft-rosenberg-midcom-turn-08, September 2005.
[32] Mozilla Labs, Geode, http://labs.mozilla.com/2008/10/introducing-geode/.
[33] D. Carboni, A. Piras, S. Sanna, S. Giroux, "The web around the corner: augmenting the browser with gps", in *Proceedings of the 13th international World Wide Web Conference on Alternate Track Papers &Amp; Posters* (New York, NY, USA, May 19 - 21, 2004). WWW Alt. '04. ACM, New York, NY, 318-319.
[34] A.O. Freier, P. Karlton, P.C. Kocher, "The SSL protocol version 3.0" Internet draft, draft-ietf-tls-ssl-version3-00.txt, November 1996.
[35] V. Ghini, G. Lodi, S. Cacciaguerra, F. Panzieri, "Meeting Interactivity Requirements in Mobile E-Witness: an Experimental Study", Wireless Personal Communications, Springer, accepted for publication, July 2008. Available online at http://dx.doi.org/10.1007/s11277-008-9630-y
[36] V. Ghini, G. Lodi, F. Panzieri, "Mobile E-Witness", Multimedia Tools and Applications (Springer), vol. 37, num. 3, pp.293-318, May 2008.
[37] V. Ghini, S. Cacciaguerra, G. Lodi, F. Panzieri, "Enhancing Mobile E-Witness with Access Point Selection Policies", Proc. of IEEE Information Technology: New Generations (ITNG 2008), Las Vegas, NV, USA, April 2008.
[38] Linux wireless extensions api. http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html. 2008.
[39] IEEE Std. 802.11i-2004, "Medium Access Control (MAC) Security Enhancements," July, 2004.
[40] IEEE Std. 802.1X-2001, "IEEE Standard for Local and metropolitan area networks - Port-Based Network Access Control," June, 2001.
[41] B. Aboba et al., "Extensible Authentication Protocol (EAP)", RFC 3748, 2004.
[42] B. Aboba, P. Calhoun, "RADIUS (Remote Authentication Dial In User Service) support for Extensible Authentication Protocol (EAP)", RFC 3579, September 2003.

www.iariajournals.org

International Journal On Advances in Intelligent Systems
ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS
issn: 1942-2679

International Journal On Advances in Internet Technology
ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING
issn: 1942-2652

International Journal On Advances in Life Sciences
eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO
issn: 1942-2660

International Journal On Advances in Networks and Services
ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION
issn: 1942-2644

International Journal On Advances in Security
ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS
issn: 1942-2636

International Journal On Advances in Software
ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS
issn: 1942-2628

International Journal On Advances in Systems and Measurements
ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL
issn: 1942-261x

International Journal On Advances in Telecommunications
AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA
issn: 1942-2601