# A Matching-Algorithm based on the Cloud and Positioning Systems to Improve Carpooling

S. Di Martino, R. Galiero, C. Giorio
University of Naples “Federico II”
80127, Naples, Italy
sergio.dimartino@unina.it

F. Ferrucci, F.Sarro
University of Salerno
84084, Fisciano (SA), Italy
{fferrucci | fsarro}@unisa.it

***Abstract* – Sustainable Mobility is a key goal to reduce pollution and improve quality of life. Information Technology can highly support Sustainable Mobility in many ways, such as highlighting to commuters the most suited means of transport for a given trip, or optimizing the vehicle’s occupancy rates by dynamically arranging carpools. In this paper, we present a solution for a Cloud computing-based platform, meant to improve Sustainable Mobility. In particular, the proposed system helps users in choosing a transport solution according to its ecological footprint, matching his/her needs, preferences, and actual location. Moreover, it exploits a geosocial network to improve the users’ confidence in the rides arranged with other passengers. From a technological point of view, the platform takes advantage of the Cloud infrastructure, resulting in a Software plus Services solution. In particular, the route calculation algorithm is based on a composition of different services available in the Cloud.**

## I. INTRODUCTION

The reduction of the global emissions of carbon dioxide ($CO_2$) represents a crucial goal of our society [17]. According to the International Energy Agency, $CO_2$ emissions released by the fossil fuel combustion was 30 billion metric tons in 2007, with an increase of 3.2% on 2006 [8]. Among these, 23% of the $CO_2$ emissions refer to the transport sector that is also the main cause of oil dependency worldwide. *Sustainable Mobility* can represent a valuable approach to address the above goal. The term has been introduced to refer to any means of transport with low impact on the environment. Among the most common means for Sustainable Mobility we can find Bike- and Car-sharing, i.e. rental from a shared fleet of bikes or cars respectively, and carpooling. Carpooling (also known as car-sharing, ride-sharing, lift-sharing and *covoiturage*) is one of the most simple and effective enhancements that can be done on environmentally friendly transport solutions. It is based on the observation that if some travellers have similar departure and destination points they could share vehicles. Carpooling allows for improving the vehicles occupancy rate, that was only 1.06 persons per car in 2005 [16] with significant benefits in reducing environmental impact, decrease in traffic, reducing moving costs, promoting social relationships, and reducing the stress due to driving.

Despite the above benefits and the many efforts and initiatives to support it, presently carpooling has reached only a partial success. This is due to both technological and social issues. Many web portals are being developed that provide basic features such as the possibility to ask for a ride in a given date/time, Nevertheless, to effectively support carpooling it is necessary to define tools able to coordinate “on-the-fly” ride offerings and requests, taking into account users’ positions and preferences [9]. To this aim, several technological-computational issues need to be addressed, such as:

1. Heuristics should be defined to address computational issues. Indeed the long term scheduling of carpooling has been proven to be an NP-complete problem[19]. Some heuristics have been defined for constrained versions of the problem [3], but in any case if the community of users becomes wider, there are still significant computational issues to be addressed.
2. Location awareness should be adequately employed to improve the service, also exploiting the GPS sensors in the smartphones. To this aim, integration within mobile devices should be considered as a priority to improve the diffusion of the service. Indeed, a platform working only on the internet is very limited [16], being able to serve in real-time only a fraction of potential carpoolers.

Moreover, there are also several not trivial social challenges that need to be adequately addresses. Among these, it is worth mentioning:

1. Lack of security for participants [9]. Sharing the car with unknown participants may discourage people to use these systems.
2. Participants may highly vary in terms of cultural, educational, and financial backgrounds. People with different profiles may feel uncomfortable in a ride arranged based only on their location and trip details.
3. Lack of reliability for pre-arranged rides. Often existing systems require that passengers arrange the ride and do not provide location-based features to check if a ride is on time, or if there are any problems.

In this paper, we describe a solution to effectively support *Sustainable Mobility* by means of carpooling, able to address part of the above issues. The proposal is based on the idea to coherently integrate, within a Cloud-based architecture, a set of sustainable means of transport with typical social network concepts. The use of a Cloud-based architecture is meant to address the computational issues. In particular, the route calculation algorithm is based on a composition of different services available in the Cloud. The use of several means of transport aims to increase the number of solutions that can be suggested to users while some social challenges are addressed by social network concepts. Finally, to improve the capillarity of the information distribution and the quality of the provided location-based service the services are accessible thanks to three different types of clients (desktop-based, mobile, and in-car telematics). The back-end of the described platform has been developed upon the Microsoft Azure solution, while the clients exploit the Microsoft Silverlight technology.

The paper is organized as follows. Section 2 provides an overview of existing proposals. Section 3 illustrates the ideas behind the proposed approach. Then, in Section 4, an overview of the employed technologies is provided, while in Sections 5 and 6 the route calculation algorithm and the multichannel clients are described. Some final remarks and future work conclude the paper.

## II. A PLATFORM FOR CARPOOLING

In this section, the platform, we named *Lift4U*, is described illustrating its services and features, while technical details are provided in the next one.

### A. The offered features

As described in the previous section, presently several utilities supporting users to arrange a trip are available on the web. Nevertheless, usually these applications deal with a single modal choice. The result is that, in order to find out, compare, and choose the most viable plan, user has to spend a lot of time, gathering fragmentary information and then “squeeze out” the solution. Thus, in our opinion, there is a critical lack of centralized software applications able to compare various transportation solutions and propose the one that will fit better some given user’s needs and location. To this aim, the platform is proposed to support users in arranging environmentally sustainable trips that best fit their profiles, by identifying and comparing different transportation modalities. Indeed, the innovation in the proposal is due to the use of a platform that embraces many modal choices with a low environmental impact (e.g., carpooling, car sharing, bike sharing, walking, mass transit, and taxi sharing) and is able to compute the ecological and economic impact of each solution. In particular, once user set departure and destination points, Lift4U gives as output several mobility solutions among those fitting user preferences, characterized by different estimated costs, commuting time, and $CO_2$ emissions. Moreover, Lift4U is able to arrange carpools, where passengers are chosen taking into account also social networks, in order to mitigate social issues. Moreover, many different channels can be adopted to access the Lift4U platform, such as a web browser, a smartphone, or an in-car telematics system. Each channel has its peculiarities, allowing users to exploit a different range of services. As an example, a web browser will support user in specifying his/her preferences, while a GPS-equipped smartphone will be useful to provide a real-time location-based transportation solution. Thus, the platform interacts with a set of external data sources to get information about non-private transport solutions and many different external social networks are involved to get information about users’ connections. The GIS services are handled by an external map service (in our case the combination of Microsoft Bing Maps and MapPoint). The core of the platform consists of three main modules running on the Cloud, namely a handler able to dispatch requests and to compute the different solutions (i.e., the *Sustainable Mobility Module*), a *GIS Module* which limits the interaction towards the external data sources to only those geographically involved in a given trip, and a *Carpool Module* exploited to arrange the pools of passengers. A remote database is also employed by the platform to store some information about users. Finally, a set of different intended clients is made available to improve the spreading of the platform.

### B. Exploiting GeoSocial Network Information

To date the spreading of car sharing and pooling is limited by some social issues. Among them the lack of trust in unknown people that could share a ride is probably the biggest one. A key point of the Lift4U proposal is to take leverage on the concept of “community”. Indeed, we believe that exploiting the social links established among users in a community is crucial to increase confidence in other users, by means of:

1. a friendship mechanism, where *friends,* or *friends of friends*, are preferred in the arrangement of a carpool or other mobility solutions that involve the sharing of a vehicle with other people;
2. a feedback mechanism, such as the one included in *eBay*, checking the behavior of a user in past rides. Indeed, at the end of each trip, passengers can rate each other and add also comments that will be valuable for future carpoolers. This concept holds also for other services, such as public transport, car sharing or bike-sharing, where users can leave ratings and comments on their satisfaction.

Since there are available on the web many global-coverage social networks with a huge number of users, it makes no sense to define from the scratch a new community for our platform. Rather, the Lift4U community is based on other diffused social networks to highly increase the potential catchment area of the platform and disseminate the benefits of sustainable mobility. Indeed, Lift4U can exploit social networks, such as Facebook, MSN Live, or Twitter, to access users’ profile information. As a consequence, a GeoSocial Network is considered to take into account not only the user profile and preferences, but also his/her actual location, to provide more tailored location-based services, as described in the following section.

### C. The Location-Based Services

The platform can be exploited through two main different channels: (i) a Rich Internet Application, declined for both a nomadic or a stationary users, or (ii) a web service, planned for any enterprise or public institution willing to provide sustainable mobility solutions within its own web portal. To overcome some computational issues and to scale accordingly to the number of users both these solutions are based on a Cloud infrastructure, thus we can talk of a Rich Cloud Application (or RCA) and of a Software-as-a-Service (or

SaaS). Details on these paradigms are provided in Section 4. Regarding the Rich Internet Application, to date it has been implemented in two different versions, namely a standard desktop application and a lighter mobile solution, able to exploit the positioning sensors of newer mobile phones, to provide Location-based services. The latter offers new interesting scenarios, leading to a conceptual shift from the old *travel planning* to a newer real-time dynamic arrangement of sustainable solution, optimizing users' mobility, by knowing his/her current position and preferences. As an example, if the transport solution is the carpool, the platform will automatically find "compatible" people from both spatial and user profile points of view. Once a user proposes him/herself as driver, he/she specifies the departure and destination points, the number of available seats, and a maximum allowed distance of detour to pick-up other passengers. Then, the platform creates an area of interest (or a *tunnel*) around the original route and checks the list of waiting passengers, looking for all the departure and arrivals points falling within this tunnel. This allows identifying not only the passengers making the same trip but also those who share only a part of the journey. Once all the involved people have agreed on the arranged trip, the mobile clients of the passengers will be notified of the carpool position, alerting their owners for the estimated pick-up time or for delays. These location-based services exploit the Hybrid Positioning Systems (XPS) technology that complements satellite navigation systems, such as the GPS, with Local Positioning Systems (LPS), exploiting other technologies to understand user position. In our case, a lookup mechanism on the IP of the client is used to infer its location. The combination of these technologies helps us in overcoming the limitations of common Assisted-GPS (A-GPS), limiting the "urban canyon effect" between tall buildings and working also indoors with an acceptable precision in the calculation of position. In the future we envision a third version of the client (which is under development), intended as a plug-in to be installed in Automotive Telematics Devices equipped with Microsoft Auto. In this way, a driver interested in creating a carpool could post the availability of seats on the Lift4U platform in a straightforward way, since some information are automatically inferred by the vehicle and in-car navigator status.

## III. ARCHITECTURE AND INTERFACE OF LIFT4U

In this section the employed architecture of Lift4U is presented, then we illustrate how the integration with external services was realized. Finally, two main usage scenarios of the proposed system are described.

### 1) *The Lift4U Architecture*

The main design criteria underlying the definition of the Lift4U architecture were to get an agile application infrastructure able to:

1. easily integrate newer sustainable solutions, as soon as they become available;
2. scale up as the number of users increments;
3. be reliable and highly available.

To address these non-functional requirements and cope with the heavy computational needs of Lift4U's adaptive route matching algorithms induced us to realize the back-end on a Cloud Computing infrastructure. Cloud Computing has been defined as "[…] *a model for enabling convenient, on-demand network access to a shared pool of configurable resource (e.g. networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*" [12]. Indeed, outsourcing IT commodities to a Cloud Computing provider brings many benefits such as a higher availability of computational power offered by data centers and an excellent scalability due to the pay-per-use concept, where additional computational resources are easily allocated on demand. Also from an ecological point of view, the Cloud solution can bring many benefits, since it allows different enterprises to share computing power as well as data storage, consequently improving energy efficiency. In particular, to realize the Lift4U back-end, the Microsoft Azure technology was exploited; it is a Platform-as-a-Service (PaaS) [12] which provides a distributed operating system (mainly based on Windows 2008 Server) useful to develop on top custom Software-as-a-Service (SaaS) [12]. Other than server computation, the necessity to provide Location Based Services, obtained exploiting the Hybrid Positioning System (XPS), and in general an appealing user experience, requires that some logic runs on the client devices. To this end the *Software plus Services* (S+S) paradigm was employed [12], since it combines the appealing features of the server-side *SaaS* model with the advantages of software running on clients. Indeed, S+S is one of the most advanced architectural patterns since the back-end is implemented by a Service Oriented Architecture (SOA) running in the Cloud to offer scalable computational power, and front-end is provided by a Rich Internet Application (RIA) [1], able to fully exploit the potentialities of newer client devices, such as positioning sensors, accelerometers, and so on. More specifically, when dealing with the Microsoft technology stack, an S+S system is known as *Rich Cloud Application* (RCA), where basically the back-end is based on Microsoft Azure and the front-end on Silverlight. Concerning the clients, they are based on the Microsoft Silverlight technology, a development platform proposed by Microsoft which offers interactive user experience. Among the features offered by this technology, one of the most appealing for our purpose is the possibility to run Silverlight interface on several platforms (i.e., Web, desktop, embedded, automotive and mobile applications), with a reduced effort due to the porting between them.

### 2) *Integration with External Services*

The integration with the external data sources, providing information on public transport was a challenging part. The main issue is that to date, to the best of our knowledge, there is no widely accepted standard to specify and exchange public transport information, such as timetables, status, and so on. So the definition of a new interchange standard could be of great utility to improve Sustainable Mobility. Our solution to afford this problem was the adoption of the *Adapter* Design Pattern

[7], which allowed us to specify our own interface for the specification of public transport information. Then, we had to realize a wrapper -for each available web service we found - which is able to convert its actual signature into our defined interchange interface. For each provider, the system keeps track of the area the service covered. (e.g., the information that a specific bike-sharing service *BS1* covers the urban area of Rome, while *BS2* the area of Milan). This is crucial information to reduce the number of providers to involve in computing a solution. In a similar way it was also realized the integration with the social networks that offer an API and allow communications over a REST XML interface, such as Facebook, MSN Live, or Twitter. As an example, the integration with Facebook APIs allows users to post on their walls ride offers, as well as to the system to access profile information. It is worth noting that thanks to the integration with other social networks, only application-dependent user information, (e.g., preferences, feedbacks, comments, etc…) are stored in our database (i.e., Microsoft SQL Azure). Thus, leaving the most of user profile onto the main social networks' infrastructures allowed us to keep track of a very small amount of information per user. The traceability of a profile is handled by means of *OpenID* [14], a decentralized standard with over one billion enabled user accounts and over 50,000 websites accepting it. In this way a user does not need to create and maintain multiple credentials, accelerating also the SignUp process. This brings also the advantage that a user's password is not stored in our platform. Finally, the geospatial and route mapping services are orchestrated with the Representational State Transfer (REST) paradigm, described by Web Services Description Language 2.0 (WSDL). In the platform we developed, the *BingMaps* and the *MapPoint* Web Services provided by Microsoft were exploited to manage the geographical information, to get data about route directions and route mapping, and to render maps onto the web clients.

### 3) *Main usage scenarios*

In the following two main usage scenarios of the proposed system are described, i.e. when the solution is based on public transport and when a carpool is arranged. These two different tasks require involving different modules and 3rd party services. In Figure 1 it is reported a simplified schema showing the modules involved in a trip based only on external mobility solutions. The process is triggered by the user that specifies his/her departure and destination points through a client. This textual information is georeferred by querying an external map service provider, such as *BingMaps*. Exploiting this spatial information, the GIS module discards all the external service providers whose coverage area is too far from the requested locations. Then, each potential service is queried to retrieve plausible transport solutions. The retrieved solutions are merged and forwarded to the geospatial mapping service, now responsible to render them onto a map. Then, a ranking mechanism, based on user's profile, ecological and cost factors is applied to put in the highest positions the solutions that should best fit user preferences. Finally, the core module will transmit the set of solutions to client that requested the service.

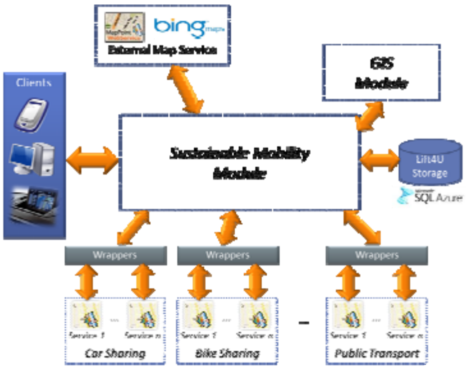

Figure 1: The modules involved in a trip based only on external mobility solutions

When dealing with carpooling, the schema slightly differs. Rather than querying external data sources, the platform searches for all the users waiting for a pick-up, whose path falls within the "tunnel", computed around the carpool driver's route. Then, a ranking mechanism promotes passengers that are in the driver's network of contacts, or those whose profile is most similar to the driver's one. Once the ride is arranged, information about status, pick-up time and location, and others are broadcasted to all the involved passenger clients.

## IV. THE ROUTE CALCULATION ALGORITHM

The route calculation algorithm is one of the key challenges of the proposed solution. Indeed, its optimal solution is an NP problem, requiring to compute all the possible arrangements of rides involving all the friends in the social network. As a consequence, some heuristics are required to make the problem addressable.

The basic idea was to exploit the Cloud to search for different solutions, each of them uses a greedy approach on a different starting item. In particular, the approach explores up to 50 different solutions, differing in the people involved in the computed ride. Each solution is elaborated on a different *Worker* in the Cloud. Once all of them have computed the total distance to cover, the best solution is picked. More in details, the resulting route calculation algorithm is based on the following five steps:

1. We start by computing the route for the user who set-up the trip, by invoking the Bing Maps Web Service on his/her starting and ending destinations. The service returns the path, intended as a sequence of manoeuvres. In  it is represented as the blue line.
2. We define a circle around the starting point of the route, whose radius is defined according to the user preference about the maximum allowance for a detour (the green circle in Figure 2).
3. We look for friends (and then friends of friends) within the social network connections, whose departure point fall within the above defined circle (yellow points in Figure 2). If there is a candidate, we compute the new route to match friend's destination. If the detour is bigger

than driver's preferences, the friend is discarded, and another solution is searched. If more than one candidate is found, we compute different solutions on different "workers" on the Cloud, beginning from the closer one. We limit our search to the 50 candidates whose starting points are close to the driver’s one. Among all the found solutions, we look for the one which minimizes the detour.

4. If the computed detour is shorter than the maximum allowed by the driver, we consider the next potential travel-mate, moving the centre on the friend's starting point and reducing the radius of the research consistently. This is shown in Figure 3, where the green circle has been moved to the closer mate, and its radius has been reduced.
5. We iterate, until (I) the maximum number of passenger is reached, or (II) the maximum detour distance has been reached, or (III) no feasible solution is found. In the latter case, the system asks the user if he/she is interested in involving unknown people in the search.

Further heuristics are included in the algorithm, suited to prefer friends rather than friends of friends in the arrangement of the ride.

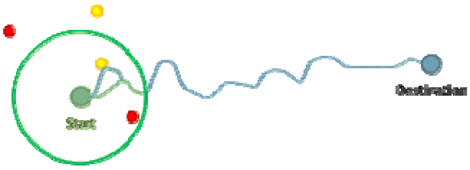

Figure 2: The Route Calculation algorithm at step 3

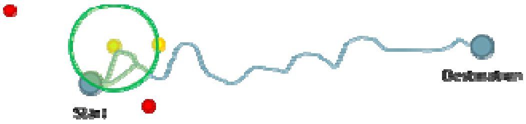

Figure 3: The Route Calculation algorithm at step 4

## V. THE DEVELOPED CLIENTS

In this section the user interfaces (web and mobile) developed for the Lift4U platform are illustrated. A scenario-based description is adopted to present the interfaces, showing a real example of use. User Bob lives in Milan and needs to travel to Warsaw for a business trip in a given date. Bob accesses the login page of Lift4U website and inserts his credentials. Once authorized, he can specify his source and destination addresses. As previously described, different sustainable means of transport will be considered by the system. As a first step, the system queries the map service, to get a total distance of the trip. This information is useful to prune some commuting modalities. For instance, since the distance between Milan and Warsaw is about 1,000 miles, of course walking and bike sharing are not considered. In particular, for such a long distance, only carpooling is proposed. Once selected carpooling, Bob specifies he agrees to drive a carpool. In order to match user’s convenience, the system asks the maximum distance in miles Bob is willing to spend for “pick-up” detours, the number of available seats, an estimated starting period, and the type of vehicle fueling, to compute the trip costs and $CO_2$ emissions. Once all this information has been gathered, Lift4U starts searching available traveling companions in the “tunnel” whose size is given by the defined maximum distance of detour. This is the most computationally expensive part of the process which is realized in the Cloud by querying the Bing Maps route calculation service. Companions are searched initially in the Bob’s network of friends. If no match is found, other potential passengers are identified, ranked by similarity between profiles. At the end of this step, Bob can browse the list of potential passengers on that route (Figure 4), checking their personal details, feedbacks and comments left by previous carpoolers who have already traveled with these users.

Km 1641,11 Km
Cost 174,21 Euro

Route

1 - Depart Piazza Castello toward Via Luca Beltrami
2 - Turn left onto Viale Gerolamo Gadio
3 - Turn right onto Viale Emilio Alemagna
4 - Keep left onto Viale Molière
5 - Bear left onto Viale Giovanni Milton
6 - Turn left onto Via Mario Pagano
7 - Keep straight onto Largo Quinto Alpini
8 - Turn right onto Via Vincenzo Monti
9 - Turn right onto Piazza Sei Febbraio
10 - Turn right onto Viale Severino Boezio
11 - Turn left onto Largo Domodossola
12 - Road name changes to Viale Duilio
13 - Road name changes to Piazzale Carlo Magno
14 - Bear left onto Via Gattamelata
15 - Turn left onto Via Bartolomeo Colleoni
16 - Turn right onto Viale Lodovico Scarampo
17 - Road name changes to Viale Alcide de Gasperi
18 - Bear right onto Cavalcavia Del Ghisallo
19 - Road name changes to A8 / Cavalcavia Del Ghisallo / Autostrada dei Laghi
20 - Keep straight onto E35 / E62 / A8 / Autostrada

| name | surname | departure | arrive |
|---|---|---|---|
| Alan | Turing | Via Legnano, Milano, Italy | AM wall, Chemnitz, Germany |
| Antonio | Brambilla | Viale Varese, Como, Italy | AM Graben, Norimberga, Germany |
| Karl | Weiss | Rainoltweg, Feldkirch, Austria | Ulica Owsiana, Breslavia, Poland |
| George | Boole | Messering, Dresda, Germany | Zwierzyniecka, Warsaw, Poland |

Figure 4: Selection of passengers and trip details

Once he feels confident about the reliability of a passenger, Bob can add him/her to the carpool. The system updates the trip preview on the map, with the driving directions and the travel cost per head, to reach Bob’s destination. Then, the final route is previewed, its length with detours and cost effectiveness are displayed and Bob can check the travel savings and the statistics about $CO_2$ reduction thanks to carpooling. Alan and other selected passengers will receive a notification via the channel specified in their preferences, namely a text message, an e-mail, or an alert when activating the Lift4U mobile client. In the latter case the platform notify Alan through his mobile device that Bob chose him as companion for the Milan-Warsaw trip. Alan is free to confirm or reject the trip plan, also considering Bob’s feedbacks and comments.

## VI. RELATED WORK

Many solutions concerning Sustainable Mobility issues have been proposed in the literature and on the market, but at the best of our knowledge, none of them is able to provide the set of services we are proposing. As a matter of fact, the majority of the existing works take into account only one means of transport (e.g., comparing different car rental rates) or a specific category of intended users with a repetitive transport pattern (e.g., employees, students) [9]. Among them, it is worth mentioning the online community of *NuRide* [13], where users earn rewards from sponsors as they arrange rides exploiting the system. Other interesting proposals are *eRideShare* [5] and the one by the Italian Highway agency [2]. Nevertheless, we believe that two weaknesses arise from an IT solution limited to a single means of transport: (i) user cannot compare various transportation solutions and choose the most suitable for his/her aims; (ii) user is not made aware of the existence of other Sustainable Mobility solutions, other the one suggested by the employed IT solution.

Concerning carpooling, several works tackling the problem from an algorithmic point of view mainly aiming at optimizing the selection of passengers and the route for their pick up. As an example, [18] proposed an optimization algorithm to reduce the idle time for carpoolers, while minimizing the number of vehicles on the road. Unfortunately, these works do not consider the social issues previously described. Integration between carpooling and social networks was proposed in *TravelRole* [15], but this solution does not provide support for mobile users, neither exploits location-based services.

Other interesting approaches have been presented in the domain of the Intelligent Transportation Systems, such as [20]. However, to the best of our knowledge, none of these solutions merge social network benefits with a smart route planning.

Finally, many carmakers are moving toward Sustainable Mobility solutions, such as the Daimler *car2gether* system [4], but they handle only car sharing/pooling and are often restricted to small geographical areas.

## VII. CONCLUSIONS

Reducing the ecological footprint due to transportation is one of the key goals to cut the emissions of carbon dioxide. Advanced information technologies can heavily support this task, by suggesting to the users "smarter" mobility solutions among the various means of *sustainable* transportation.

In this paper, the Lif4U platform has been described. It is meant to overcome the limitations of current sustainable mobility solutions, by coherently integrating a wide range of new technologies. Among them, the Microsoft Azure platform allowed us to develop a Cloud-based back-end, where the computation of route arrangements can be done exploiting the massive elaboration power of the Cloud infrastructure. As for the services, given a user's departure and destination addresses, together with other preferences, the proposed framework computes the transport solution with the lowest ecological footprint, considering various mobility alternatives, namely carpooling, car sharing, bike sharing, walking, mass transit, and taxi sharing. As for carpooling, the use of a Geosocial network is included, in order to overcome some social issues that can arise when sharing a trip with unknown people. Indeed, the system tries to match users' profiles and preferences when combining a ride, taking into account locations and social links among users. The current location of the participants, obtained through Hybrid Positioning System, is taken into account to improve the affordability of the proposed arrangement. Many other issues have to be still addressed. First of all, an usability assessment is planned in order to evaluate the effectiveness of the proposed interfaces. Then, as any community-based solution, there are strong issues about the privacy of the information, especially when dealing with tracking user movements. Moreover, to date the use of the Cloud helped us in overcoming problems in computing the most suited arrangement for carpooling, but other heuristics should be defined, for instance to allow for mixing different means of transport within the same trip, that presently is too much computationally onerous. In particular, it could be very interesting to investigate the use of Search-based approaches, such as Genetic Algorithm or Ant Colony Optimization, for finding preference-based shortest path [6][11].

## REFERENCES

[1] Allaire J., Macromedia Flash MX – A next-generation rich client; 2002.pp 1-4

[2] Autostrade Carpooling, available at http://www.autostradecarpooling.it/

[3] Coppersmith D., Nowicki T., Paleologo G., Tresser C., Wu C., *The Optimality of the On-line greedy algorithm in carpool and chairman assignment problems*, IBM Research Report, (2005)

[4] Daimler car2gether, available at http://www.car2gether.com/

[5] eRideShare, available at http://www.erideshare.com/

[6] Fu, L. Sun, D., Rilett, L.R., Heuristic shortest path algorithms for transportation applications: State of the art. Computers & OR 33(11): 3324-3343 (2006)

[7] Gamma E., Helm R., Johnson R., Vlissides J., *Design Patterns: Elements of Reusable Object-Oriented Software*, ISBN 0-201-63361-2, Addison-Wesley, 1994

[8] International Energy Agency, *CO2 Emissions from Fuel Combustion – Highlights*, 2009 Edition, available at http://www.iea.org/co2highlights/CO2highlights.pdf

[9] Lalos P., Korres A., Datsikas C., Tombras G., Peppas K., *A Framework for Dynamic Car and Taxi Pools with the Use of Positioning Systems*, computation world, pp.385-391, 2009 Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, 2009

[10] Lift4U website, available at: http://www.lift4ucarpooling.com/

[11] S. Ok, W.J. Seo1, J.-H. Ahn, S. Kang, B. Moon, An Ant Colony Optimization Approach for the Preference-Based Shortest Path Search, in Proceeding of FGCN/ACN 2009

[12] P. Mell and T. Grance, Draft NIST working definition of cloud computing, v15, 21 Aug 2009

[13] NuRide, available at http://www.nuride.com

[14] OpenID, available at http://www.openid.net

[15] Selker, T., Saphir, P.H., *TravelRole: A carpooling / physical social network creator*, in Proceedings of the 2010 International Symposium on Collaborative Technologies and Systems (CTS), pp 629 – 634.

[16] Steger-Vonmetz, D.C., *Improving modal choice and transport efficiency with the virtual ridesharing agency*, in Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE, pp 994 - 999

[17] United Nation Millenium Goal website: http://www.un.org/millenniumgoals/ . Last visited on September 2010.

[18] Vargas, M. A., Sefair, J., Walteros, J. L., Medaglia, A. L., Rivera, L. *Car pooling optimization: a case study in Strasbourg (France)*. In Proceedings of the 2008 IEEE Systems and Information Engineering Design Symposium, pp. 89-94

[19] Varrentapp K., et al. The Long Term Car Pooling Problem – On the soundness of the problem formulation and proof of NP-completeness, November 2002

[20] Winter, S. Intelligent Self-Organizing Transport. Knstliche Intelligenz, 08 (3): 25-28