Software Infrastructures for ad-hoc networks oriented to difficult environments

WP1D3

DISI - Universita' di Genova
in cooperation with the other participants

September 2005

Abstract

Da fare

1 - Introduction

Da fare

2 - Project Environments

Here we include from Deliverable WP1D1 the networking situations considered in the project from the very beginning. Then we expand three possible applications scenarios for the project. Notice that the third scenario is already in the deployment phase (with the cooperation of a local company) while the first two scenarios are kept as a reference for possible deployments.

2.1 – Networking situations

Situation A:

In this simple situation, we have only one MANET without any link to the external world. In this situation a working group of people may share common data, for example a data base of territorial information during the emergency situation, or a related set of snapshot at a given sport event. The data may be partially replicated over various MANET nodes. Other possible environments where this situation occurs are those where each group member needs only group communications (no data exchanges to-from outside the group). In this situation communication could be effective only within a limited radius, because of power limitations on mobile radios and because of possible obstacles. A routing protocol specialized to handle these situations may overcome possible disconnection problems.

Hardware requirements: each unit has a mobile device with WiFi connection, possibly also GPS or webcam.

Situation B:

This situation expands the previous one: the MANET is not connected to the outside world, but there could be some Access Point (Base Station) working as bridges in order to improve the connection among units. These Base Stations can be located in relevant places, and may have omnidirectional antennas to cover a wide area, or directional antennas to reach difficult areas with stronger signals.

As a consequence, in this situation we can imagine our network as divided into “zones” with a tree-like organization: the root is a router, its descendants are Base Station in bridge mode or just mobile units, and all links can be considered static ones (bridge-bridge links always active by means of high gain directional antennas, while bridge-mobile device links can be kept by omnidirectional antennas). This organization could guarantee total coverage of interesting areas, without requiring wired connections or outside connections. It is important to remember that Base Stations may have high energy requirements, so they should be equipped with long lasting batteries and solar panels. The network spanning tree must also be very low, in order to reduce the traffic among bridges.

Hardware: each mobile unit has a WiFi connection, possibly with GPS or webcam. We need also Access Points to be used in repeater mode, WiFi routers, long-lasting battery and solar panels. Special care must be taken in the choice of antennas and possibly parabolic ones.

Situation C:

The connection to the outside world of the MANET is achieved by possible non-WiFi technologies. These include: connection to a satellite network (geostationary satellites at low orbiters LEO), cellular networks (GSM, GPRS, UMTS) or fixed connections (LAN or Internet). For example, one of the Base stations can be on a van, and may support such external connections as well. To achieve an end-to-end communication, suitable middleware would select each time the most suitable medium, considering available links and speed-power tradeoffs.

Hardware: each mobile unit has a WiFi connection, possibly with GPS or webcam. We need also Access Points to be used in repeater mode, WiFi routers, long-lasting battery and solar panels. In addition, cellular or satellite communication systems should be installed at least in one Base Station.

Situation D:

As a further expansion of the previous situations, at a certain point some fixed units may be included in the network, such as for example a computer/laptop/tablet connected to a wired link or to a proper power supply network. For example a van containing power generator and servers may arrive later on in the disaster area. We may then consider such a node a special one of the already established MANET, in one of the previous situations.

Hardware: as the above ones, plus the non-mobile unit which, from the point of view of the communication, does not cause further communication problems.

2.2 Actual scenarios for possible applications

The following scenarios have different requirements for the MANET:

- In the first one, a closed group of peers (reduced number of nodes) is moving in a disaster area. No infrastructure is available. Possible disconnections and security issues.
- In the second one, a very large number of peers forms on the fly groups sharing a common interest. There may be an existing infrastructure, but it is certainly under-dimensioned for the event. The peers groups extend wired connectivity, where available. Very few critical information is kept on the MANET.
- In the third one, an existing wireless infrastructure suffers from possible noise and lack of coverage of the interested area, so certain users are connected in a P2P fashion, in a seamless way, to the fixed network. The deployment in a real world application is currently under way by cooperating with a local company (harbour logistics).

2.2.1 - The first scenario: Earthquake at Casal Ballerino

On June 2, 200X an earthquake reaching the 7° degree Richter hits the area around Casal Ballerino. The Civil Protection reaches the area about 2 hours later, and settles the “base camp” 5 km away from the epicentre. The base camp has many wireless devices:

- Satellite or cellular devices for connecting to the wired network (if available)
- Radio connections with the operative teams (located from 5 to 10 km away)

- Radio connections (802.11, ...) for short distance communication

Each rescue team consists on an autonomous group of 5 people; each team has:

- A 4x4 vehicle with transmitters to connect to the base camp and to team members
- A PDA for each team member providing wireless connection to the vehicle and possibly to other team members

- GPS and other devices for each team member, including videocameras to document the situation

The mission of these rescue teams includes

- First aid and transport to a safe place for the locals
- Estimate damages to infrastructures

- Block access to dangerous areas

- Start works for securing the area

For example, areas close to the lakes and rivers are potentially dangerous for the risk of floods. It is urgent to identify the dangerous areas and to send people away from them**.** Due to the fall of electric wires, some small fires have started, in certain buildings of the village as well as in the woods: it is a priority to estinguish them, so teams must coordinate their work with the firemen, to direct their vehicles to the fires.

At the base camp, all coordination activity is taking place:

- By GPS connections the position of all teams and all team members is known
- Orders to the various teams are prepared and sent to the devices of each member (Certain orders may require a distributed commit)

- Communication to the Headquarters in Prefettura and to the press originates from here

Assume now that one of the rescue teams, Team 5, is entering the center of Casal Ballerino (the epicentre of the earthquake) for a first check of the disaster area and to coordinate other rescue teams. Each team member has a detailed map of the area on his PDA, and connecting to the territorial DB he/she may fetch more information on the area. The positioning system allows each member to locate precisely his own position as well as the position of other members.

Each member has to report about the status of the buildings, if necessary with images. The information collected so far are automatically shared among team members and with the base camp. While searching for injured people**:**

- Information about injured people shall be immediately forwarded to Teams 6, 7 and 8, consisting on medical staff, who will be responsible for caregiving
- For extreme urgencies the team can make audio-video calls to show the actual situation and do some first-aid intervention under medical remote supervision

Three team members (Andrea, Laura, Mario) have the specific task to check the status of water and gas pipes. They upload the map of these pipes and start the survey of the respective area.

Andrea Verdi has gone too far from the 4x4 vehicle of Team 5, so he has lost connection with the base camp. He goes on with his survey, and the collected data is stored in his PDA; when he meets Laura Bianchi the information that were collected separately are shared among both, and he sees that all the area east to where he is has already been surveyed by Laura. He also realizes that two other team members have already inspected the south area: this inspection was done while he was out of reach (and Laura was in reach). So Andrea decides to go west, and Laura goes north, back to the vehicle of the team.

As soon as Laura approaches the vehicle, all data collected by her and by Andrea (until he met Laura) are uploaded to the base camp and shared with other team members.

Mario's PDA runs out of power while he is disconnected from the others (except for Andrea, who was near him). Mario goes back to the 4x4 vehicle and takes another PDA; when he fetches all data available: hesees that he has lost the data he surveyed after his last disconnection. But the data is just temporarily unavailable since it was backupped by Andrea: when Andrea connects again, his data contains the last updates, including Mario's: nothing is lost!

In this scenario, there are two important requirements, security and priority:

- The system must show certain confidentiality degrees (the press should not be aware of everything!)
- The system must be protected from "external" attacks and device malfunctions, which may corrupt the information

- Certain messages have higher priority than others (commands, alerts, etc.)

Less crucial, but important features, could be:

- Use the cellular network
- Use sensor networks

- Exploit helicopters, when available, to:

    - Cover mid-distance connections

    - Synchronize over the time different MANETs

- Exploit other wireless connections (e.g. military)

- Streaming video

2.2.2 - The second scenario: Torino 2006 Winter Olympics.

Most of the information collected hereafter has been derived from the website http://www.torino2006.org/. This event is attracting a large number of people, and we may assume that many of them shall be equipped with a mobile device. There shall certainly be a number of WiFi hot spots around the area, but they shall be hardly dimensioned to cover the actual number of people attending the Olympics. The opportunity for P2P communication, with or without networking support, is to be considered important for the entertainment of attendees as well as for those working in the organization.

Here is some data about the event:

- 17 days of competitions
- 15 disciplines

- 7 locations (3 Olympic Villages)

- 2500 athletes and approx. twice as many official delegates

- 1,500,000 tickets issued

Tickets are sold also through a website. It would not be unrealistic to assume that "registered" people in the website shall be allowed to downolad a special purpose software to enable them to exploit wireless facilities while at the Olympics.

Sports are distributed across the Turin area and over time: for example:

- Hockey: located at Torino Esposizioni (downtown), from February 11 to 26
- Bobsleigh: located at Cesana, 90 km from Torino, from feb.18 to 25
- Alpine Ski: located at Sestriere,from feb. 12 to 25
- Freestyle: located at Sauze d'Oulx, 6 days between Feb 12 and 24
- Ski Jumping: located at Pragelato, 6 days between Feb.12 and 24

and the ticketing system allows to buy multiple tickets for one or two consecutive days: people shall move from one location to another, mostly with public transportation. Since each event attracts thousands of fans from anywhere in the world, there shall be a large number of people "on the move" with a mobile device. The actual numbers of tickets sold are:

- Bobsleigh: 7450 people,
- Hockey: 6450 seats,
- Freestyle: 9020 people,
- Alpine skiing: around 9300 people

Consider now what could be the day of one sport fan, Alice, at Torino Olympics on 14 feb 2006. In the morning she is in Sestriere to watch the alpine skiing: she gets to a position where she can watch a very exciting jump, so she takes digital pictures to many competitors with her Java mobile phone. Alice would like to share them with other people so *she has to "post" an announcement.* Bob has digital pictures taken from another side, he reads Alice's post and wants to exchange the images, so *they have to exchange files (directory content and select individual files) in read-only way*. Charlie has only a PDA, he cannot take pictures, and would like to pick up some images from both; Dave is far from Alice's seat, he has a laptop, and he can fetch Charlie's pictures. *File sharing among the four of them is then possible, by routing the communication and the files*.

After the competition is over, Alice is hungry, she asks for a good restaurant in the area. She joins another group of peers, with potentially hundreds of people, for chat and messaging. Later on, she goes back to Torino, because she has an evening ticket for hockey. She takes a bus, and since the trip lasts for more than one hour, she would like to listen to some country music in digital form. Another group of peers can be formed on the bus while moving to share a file stream.

At the same time, the technical staff of the Austrian national team is exchanging information about slope condition: *they want to encrypt the communication, to be hidden from other teams and from reporters*. The organization team is collecting data on traffic in the valley and is discussing weather forecasts: a snowfall is likely, shall we block the private traffic or could it be only a very light snowfall? *This information too is encrypted.*

Alice reaches Torino Palasport to see the hockey match: she is a fan of the Canadian team, tonight playing vs Russia. *Groups of peers in Torino Palasport may be supported by access points; with hundreds of potential connections will it work? Shall we assume a "cloud" of peers extending network connectivity outside AP visibility?*

2.2.3 – The third scenario: networking and logistics on harbour piers

This section presents a case study of one of the situations described in deliverable WP1D1, namely Situation B, shown above. This scenario has several applications, such as the ones presented above, the Torino 2006 Winter Olympic Games, or civil protection operations in disaster areas. Other examples are:

- Buildings and areas of great historical/cultural value, where a fixed network cannot be installed for its invasiveness
- Large industrial areas where movement of heavy trucks requires great attention in the installation of underground cables. Even when it is possible to place an underground cable, this would require heavy construction works, implying that the plant must be shut down for some time
- Phone / Internet connection services in sparse rural areas, where the cost of optical fiber connections is higher than that of amplifying nodes with long distance antennas.

In the following we describe some of the problems encountered during the deployment of a MANET in an environment where large trucks are always moving at low speed, with a lot of noise due to interference and reflexions. The study was a part of a graduation thesis (Laurea in Informatica vecchio ordinamento) at the University of Genova, to be discussed in Fall 2005, developed in cooperation with an italian company.

The final MANET will be composed by a number of fixed wireless nodes, connected to a Main Base Station through secundary Stations, by means of ad-hoc routing protocol (without Access Points), as shown in Figure 1. A clouds of wireless hosts, assembled on moving means of transport, can benefit of such an ad-hoc network to communicate with the central office, where the traffic is coordinated.

Presently, after the analysis of the state of the art (as described in deliverable WP1D1), a suitable routing algorithm has been tested on the field, using two fixed and one mobile nodes, with two different antennas, in a noisy environment.

More complex scenarios will be studied in the future, in order to provide the other project participants a set of results useful to tune their applications.

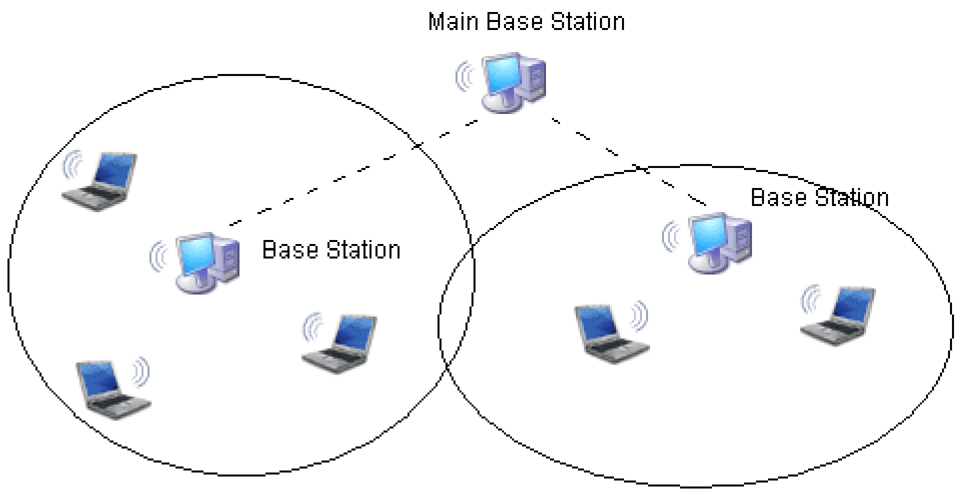

Figure 1: A MANET with 2 base stations connected to a main base station

2.3.3.1 Optimized Link State Routing - OLSR

OLSR is the algorithm that has been selected for dynamic routing implementation. This subsection describes the reasons for such a choice. To avoid misunderstandings, in the following we shall refer to the algorithm as OLSR, and we shall refer to the particular implementation as olsrd .

The algorithm

OLSR is formalized in RFC 3626 proposed by the Hypercomm project, developed by INRIA[1].

At present it still is considered inside an experimental phase, but the protocol core is well-defined and the remaining discussion deals with extensions still in experimental phase. The presence of an official formalization allows the different implementations to converge to a common set of base functions, and to provide additional features while keeping interoperability.

Such a convergency is slowly taking place, yet at pesent not all existing implementations are actually interoperable with one another.

In the last two years there have been two *OLSR Interop & Workshop*, the first one in the US, the second one in Paris, sponsored by various well known companies, aimed at easing the integration process and increasing cohesion among the user/developers community of OLSR.

OLSR is a proactive algorithm, and at startup each node cooperates with its neighbours to build a routing table containing all paths to all existing nodes. After startup, nodes periodically exchange messages to keep their own topological data updated and consistent.

As the algorithm name suggests, OLSR broadcasts information about the status of all known connections, so that each node can reconstruct network topology. These broadcasts however must be optimized in order to limit bandwidth usage, by use of a system called *MultiPoint Relaying* (MPR).

[1] *Institut National de Recherche en Informatique*

Implementation

At present there are several OLSR implementations, at dufferent stages of maturity. INRIA is developing OLSR in C++, the Naval Research Laboratories in the US are developing NROLSR, the Canadian Communications Research Center has been busy (until 2003) working on CRCOLSR, and finally the *Laboratoire de Recherche en Informatique* of the University of Paris-South is active in implementing QOLSR, where emphasis is given to respecting *Quality of Service* criteria. In addition to these, the most mature implementation appears to be olsrd, developed within a PhD thesis at the University of Oslo by Andreas Tønnesen, and still being maintained.

olsrd has been tested in a mixed network (wireless/fixed) consisting on approximately thirty nodes by the author during the conference Wizard of Operating Systems (Berlin, 2004). It works entirely in user space, and it employs standard mechanisms to update kernel routing tables. In addition, it has a plugin system which allows to include extensions without modifications to the source code of its core. There are several useful plugins already available, providing additional services like battery status communications, or node authentication. A further strength is the licensing policy, an *Open Source* license, which shall guarantee project continuing updates even if the author will eventually quit the project.

At present olsrd is being used at Lille, France for the project Lille Sans Fil[2] aimed at providing wireless coverage to the whole city (in the future to all the region) with dynamic routing. olsrd is also a part of the Freifunk[3] project for the diffusion of free networks in German-speaking areas.

There are available several versions, for Linux, Windows (2000 and XP), Os X and Linux Familiar (for handheld devices). This feature is important to allow the interoperability of the current implementation with other implementations by competitors.

2.3.3.2 Reasons for the choice

The decision to choose OLSR and olsrd rather than other algorithms or implementations has been based on the existence of an active community around it, which guarantees continuing support in the future. In contrast, websites of other projects appear to be quiescent, and have not been updated recently.

This consideration overrides technical aspects, such as for example certain peculiarities of our environment. All communications, in fact, take place among two fixed subsets of nodes, with intermediate nodes acting only as repeaters. This pattern would suggest a preference for a reactive algorithm like AODV, where reachability information are kept only for relevant nodes.

Another advantage in the use of olsrd for network routing emerges from the consideration that VoIP applications on a MANET could be implemented in the next future, substituting radio communication. These applications would make a more extensive use of networking, and would take advantage of a proactive algorithm like OLSR.

Here is a list of pros and cons taken into account when selecting OLSR and olsrd.

<u>Pros:</u>

- Very active developers team
- Large scale successful tests

[2] *http://www.lillesansfil.org*

[3] *http://www.freifunk.net*

- User space implementation
- Expandibility with plugins
- No delay when new routes are sought

Cons:

- Large bandwidth used to keep track of paths which might never be used
- Large resource expenditure (memory and processor) to keep information which might never be used
- Startup time, quantified between 2 and some tens of seconds, in accordance to the number of nodes and their topology.

2.3.3.3 Latency in routing with OLSR

We designed a set of tests in order to measure the influence of routing in network latency.

Since we do not foresee the need for paths longer than three-four hops, we made a test with four devices. The devices were connected through an Ethernet switch, in order to be able to issue commands and data by telnet without polluting the wireless network with more packets than those in the tests.

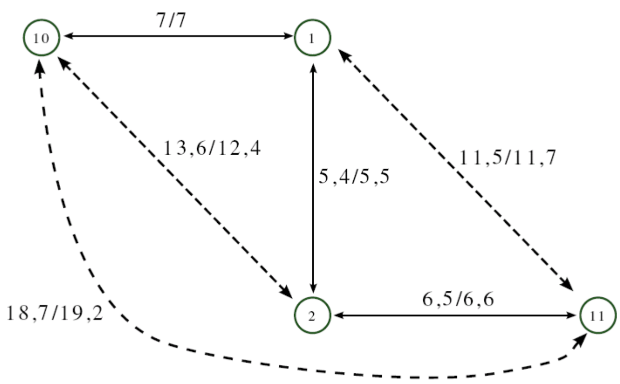

Figure 2: Graph of the network used for testing latency; data has been derived by averaging 40 measures taken with ICMP echo request reply packets

This test, rather than measuring latency introduced by the use of olsrd, serves as a testbed of the performance of the routing code inside the Linux kernel. In fact, once olsrd has established a path to a given destination, it adds a *static route* to the kernel tables, which from that moment behaves as if the route had been manually added. This test is static, and it is started once olsrd has reached a stable configuration, shown in the following:

```
--- 08:42:41.12 ---------------------------------- LINKS


IP address        hyst   LQ     lost    total  NLQ    ETX
192.168.128.1     1.000  0.000  0       0      0.000  0.00
192.168.128.2     1.000  0.000  0       0      0.000  0.00
192.168.128.11    1.000  0.000  0       0      0.000  0.00


--- 08:42:41.12 ------------------------------ NEIGHBORS
```

```
IP address        LQ      NLQ     SYM    MPR    MPRS   will
192.168.128.1     0.000   0.000   YES    NO     NO     3
192.168.128.2     0.000   0.000   YES    NO     YES    3
192.168.128.11    0.000   0.000   YES    NO     NO     3


--- 08:42:41.12 ------------------------------ TOPOLOGY


Source IP addr    Dest IP addr      LQ      ILQ     ETX
192.168.128.1     192.168.128.10    0.000   0.000   0.00
192.168.128.1     192.168.128.2     0.000   0.000   0.00
192.168.128.2     192.168.128.1     0.000   0.000   0.00
192.168.128.2     192.168.128.11    0.000   0.000   0.00
192.168.128.10    192.168.128.1     0.000   0.000   0.00
192.168.128.11    192.168.128.2     0.000   0.000   0.00
```

Portions named LINKS and NEIGHBORS show information on neighbours of the node that displayed such an output. On the other hand, the section named TOPOLOGY shows network topology, displaying IP addresses as source/destination pairs where there is a link. A graphic representation of the network is shown in Figure 2, where we show also latency times of 64-byte packets.

Note that time taken to reach the last node (numbered 11) from the first one (numbered 10) is almost the same that the sum of the various hops. The small difference is explained as the delay introduced at each node to take a routing decision, which is not greater than one millisecond.

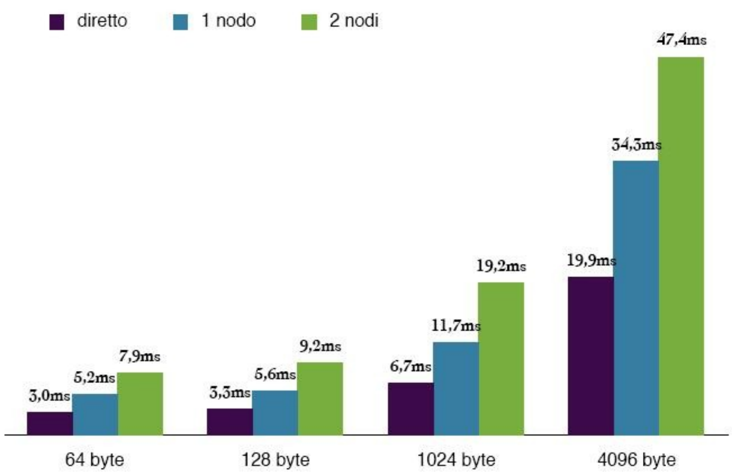

Figure 3: Latency of the OLSR network by varying the number of intermediate nodes and packet size. Data represents the average of 40 measures taken with ICMP echo request/reply packets

With the same data we derive the graph shown in Figure 3, where we can see the linear increase of latency as the number of intermediate nodes in the routing path. In fact each hop introduces a constant delay, due to the time required to:

- read the frame from the network interface
- wait for the arrival of next frames if the packet has been fragmented

- choose a route by consulting routing tables[4]
- send the packet to the network interface, possibly after fragmenting it

Measures were taken also with packets of a greater size than acceptable by the 802.11 protocol (1500 bytes). In this case the IP level has to handle more frames (of smaller size), first reassembling and then fragmenting again the packets at each node.

As expected, routing fragmented packets does not cause significant delays with respect to non-fragmented packets.

2.3.3.4 Performance of two different antennas and use of VNC

This test measures the performances of two antennas, to be used on fixed base station, on top of high buildings or towers.

The antennas were placed on a building, approx. 20 metres high. A first node is at the first floor of that building, and it is connected with an Ethernet cable to a second node, on the terrace of the building. Here the antennas may reach the third node, placed on a car which moved along the red course in Figure 4.

The test was repeated twice, first using one antenna, then using the second antenna; the graphics in Figures 5 and 6 cannot be perfectly synchronized since the actual traffic conditions during the course were slightly different and influenced the measures.

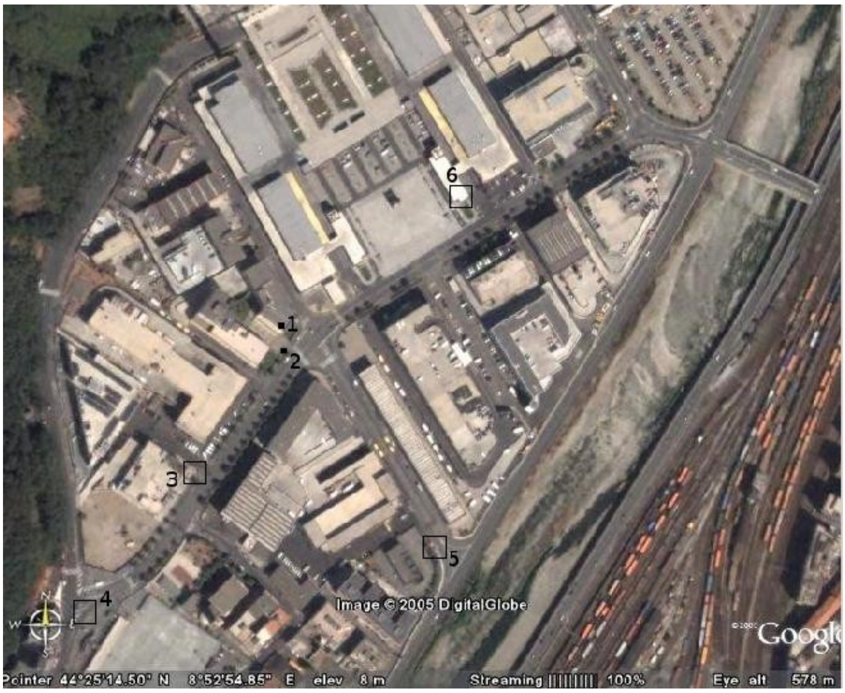

Figure 4: The course taken to evaluate signal quality. Numbered points correspond to positions in the graphics in Figures 5 and 6. Fixed antennas were placed in point 1. Distances between points 1,4 1,5 and 1,6 is approx 200m.

[4] *In Linux this step takes O(log(n)) where n is the number of paths in the table*

In numbered points we stopped to record signal quality, and to use VNC, connecting to a remote desktop session to the computer in the office. In this way we tested the possibility to run such an application over a MANET.

2.3.3.5 Antennas comparison

The comparison of the two antennas has shown that their features can be roughly comparable, so that a choice may be based on other parameters, such as costs and availability of a distributor. The differences are mostly due to the different shape of propagation.

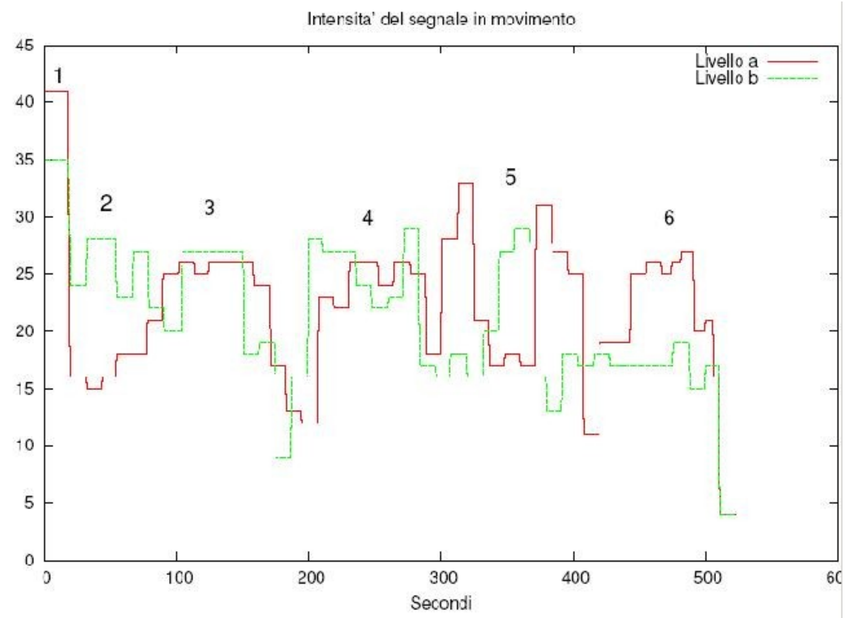

*Figure 5: Signal strength for the two tested antennas. Antenna **a** is manufactured by Cisco, antenna **b** by Huber+Sunher.*

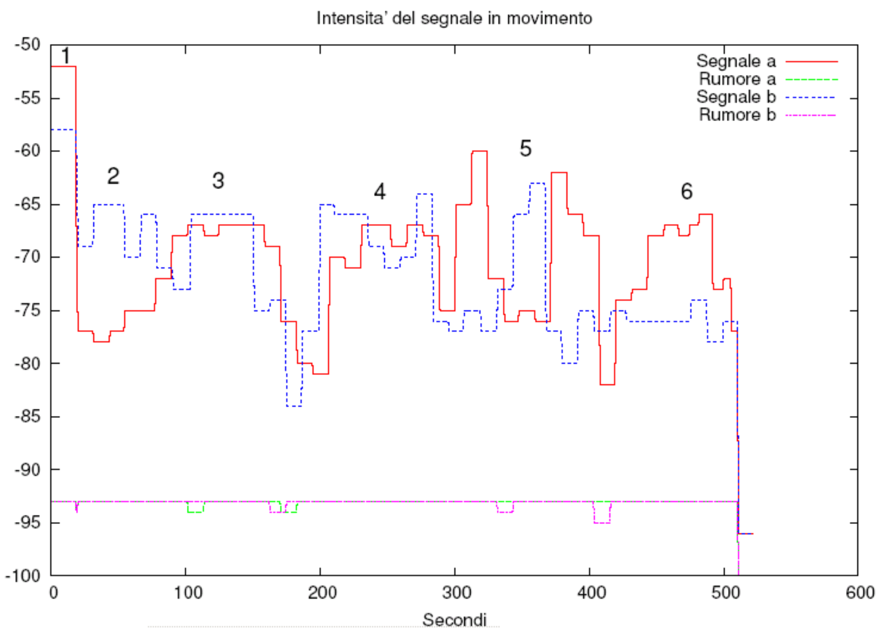

Figure 6: Graphic of the signal and of the noise received by the mobile antenna, where the a and b antennas are the same as in Figure 5

The Cisco antenna gives a stronger signal to a receiver placed at the same height, and this is an interesting advantage for the connection of the two fixed stations, to be placed at the same height. Mobile nodes shall be in the visibility range of two antennas for most of the time.

The good signal to noise ratio at a distance of 200 m makes us foresee that actual communication range may be greater, and longer than the distance between fixed nodes.

Conclusions - VNC

VNC use proved to be fluent, with depth and color resolution higher than needed by the application. The problem we experienced was the loss of the connection by the server when some packet loss is experienced, thus closing the client. A simple reconnect operation was sufficient to restore the working environment.

This problem can be solved by increasing the server timeout or by modifying the client to automatically reconnect when some problem occurs.

A second test, similar to this one, to be completed soon, will be done to test VNC functioning with two intermediate nodes. This is a very important test for our environment, since static tests in the lab showed that the available bandwidth is halved at each hop in between.

Conclusive tests with 4 nodes and 3 hops

This test was done to check the stability of dynamic routing with olsrd, that is to see if a continuous use of the network is possible, and if the bandwidth is sufficiently large to allow two VNC sessions at the same time.

This test allowed to establish new values for hysteresis and for HELLOValidityTime (two fundamental parameters in configuring olsrd), better suited in a slow mobility network as the one we foresee (maximum node speed shall be 30Km/h).

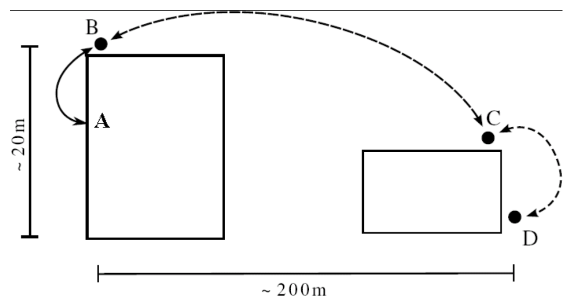

Figure 7: Node placement during the test. Nodes A and B were connected by an Ethernet cable, other nodes connect by 802.11. All routing is handled by olsrd

The test has showed how much the WiFi signal can be influenced by the environment, e.g. reflections on metallic surfaces. During the test a TIR moving in the neighborhood caused as much reflections as to "convince" olsrd to establish a new route, thus causing a consistent packet loss.

Node placement is shown in Figure 7: **A** is a laptop with Windows 2000 and a VNC server; it is a demo version of an application actually running in the production environment of interest. **B** and **C** are WiFi routers, where software was slightly modified to allow dynamic routing as well. Finally, **D** is another Linux laptop, with a logging system especially developed for this test, a ftp client and a VNC visualizer.

FTP has been used to measure the available bandwidth between the first and the last node, transferring a 2 MB file. At the same time, the ftp traffic simulates a second VNC connection, to test the effect on the already established VNC connection.

3 - Software Architectures

In MANETs such as the above described ones, all problems of wireless networks can be present, such as energy saving and bandwidth optimization; besides, there are also other problems typical of the ad-hoc environment, like multi-hop routing and frequent network topology changes.

The consequences of these problems not only affect the network layer, but also other levels in the ISO-OSI stack can be affected, to handle problems like security handling, mobile node positioning, resource discovery and so on.

For these reasons the classical level separation in the ISO-OSI stack can be violated because a need for cooperation among levels may arise, in order to share common information and to send signals when system stayus is changed, as shown in the following figure.

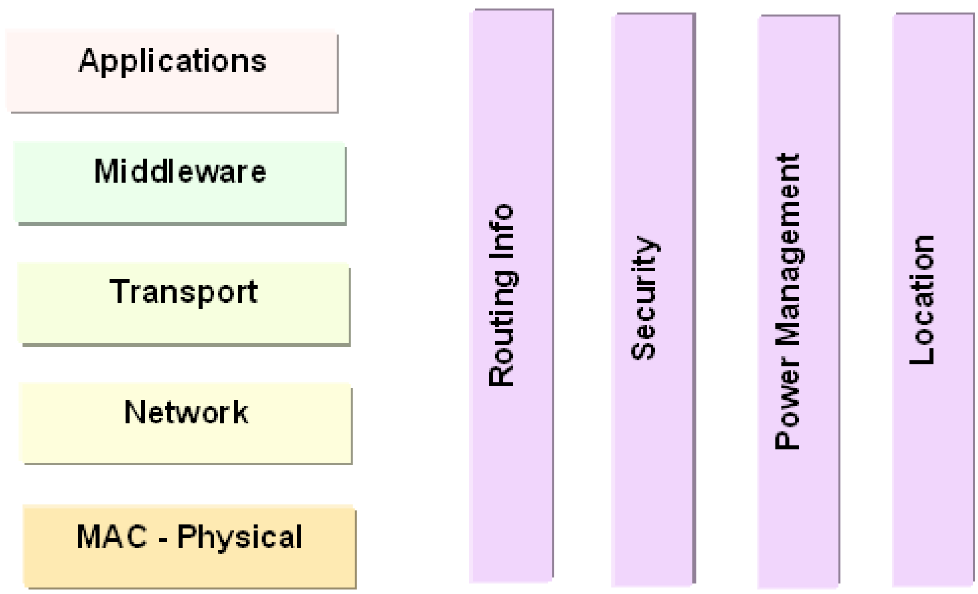

Figure 8. ISO-OSI stack and intra layer relationships

As outlined also in WP4 Deliverables, it is not accidental that one of the question at the state-of-the-art for the MANET community is to what extent developers must modify the pure layered approach by introducing stricter cooperation among MANET-specific solutions belonging to different layers. At the one end, some approaches use strict layering to maintain compatibility and solve interdependencies between different protocols and different solutions. A full cross-layer middleware design represents the other extreme, often present in the current MANET literature.

The MANET research community recognizes that cross-layering can provide significant performance benefits, but also observes that a layered design provides a key element in the success and proliferation of a technology. Strict layering guarantees controlled interaction among layers because developing and maintaining single layers takes place independently of the rest of the stack. On the other hand, an unbridled crosslayer design can produce spaghetti-like code that is impossible to maintain efficiently because every modification must be propagated across all layers. Further, cross-layer design can produce unintended interactions among protocols, such as adaptation loops, which may result in performance degradation.

The software architecture aims at providing context-aware and location-aware services which may strongly depend on locality and on any possible contextual and deployment information. These services should be available on a wide variety of devices, from workstations to servers, laptops, smart cell phones, wireless sensors, PDAs, and RFID tags, and with various kinds of wireless connections: Bluetooth, WiFi, satellite links. In a layered architecture, it is convenient to distinguish among base services and advanced services, the latter being built on top of the former. In turn, applications are built on top of advanced services.

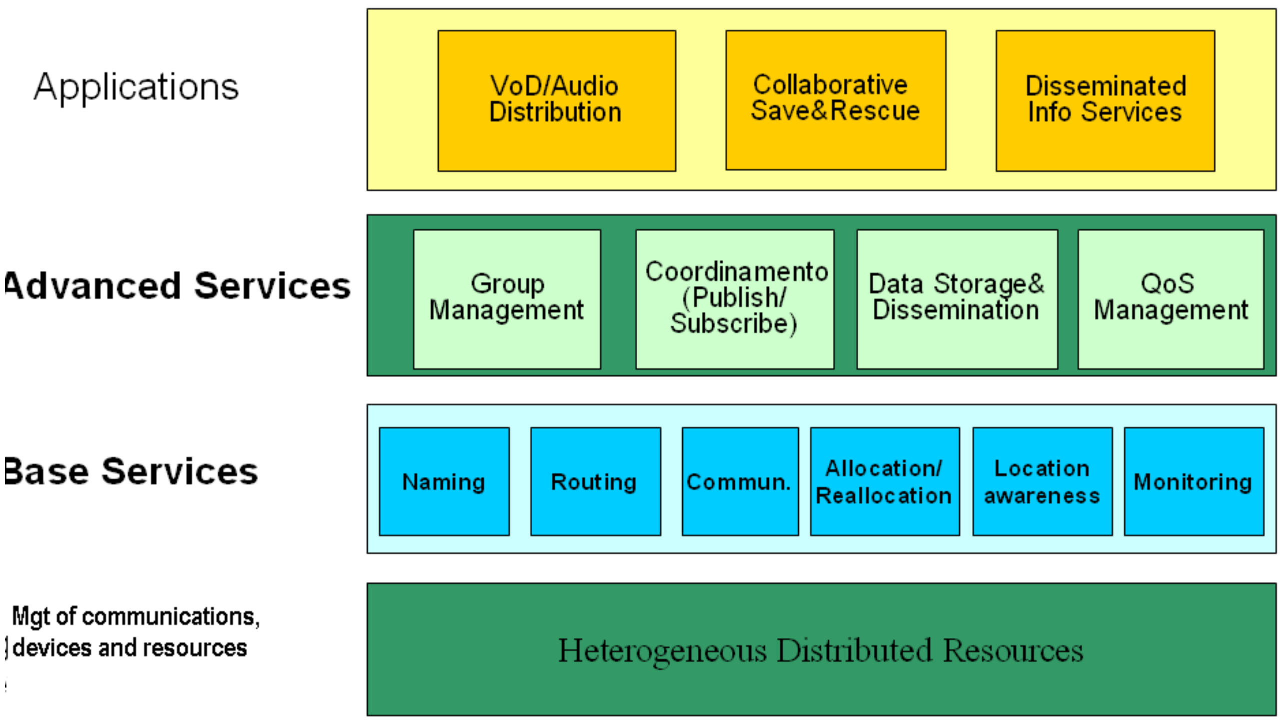

Figure 9. Software architecture

Base services provide fundamentals for any kind of service in any scenario. We consider the following components:

- Routing
- Naming
- Communication
- Allocation and Reallocation

- Location awareness
- Monitoring

These services are especially designed for overcoming limited resource availability and frequent disconnections/topology changes.

Routing is the best algorithmically studied component, and may provide unicast as well as multicast or broadcast services. It is typically a reactive task, but it may use some proactive organization of tabular information as well.

Communication may be used to transfer information about entities and resources. The available API usually support point-to-point connections, and sometimes anycast, multicast, broadcast, based on routing tools and tables.

Naming may identify resources and entities, employing traditional naming schemes, or more sophisticated ones, based on requirement based or system based policies.

Allocation and *Reallocation* components are capable of surviving to insertion and deletion of network hosts, and may implement advertisement policies with dynamical cost adaptation.

Location awareness components must support and maintain locality, both in active support nodes and in passive nodes, in varying density conditions, supporting role changing and election protocols to reconfigure the network after node insertion or deletion.

Monitoring components are in charge of system status identification and control. It is the base for advanced QoS management components.

Advanced services add semantics to the basic operations by coordinating policies and improving management functionalities. Their separation from base services often is a logical one, rather than an implementative one, especially when systems tend to overcome layering barriers. The need for cross layer implementations is typical of systems with stringent requirements and it is documented by several examples in the real world.

We have the following services:

- Group management
- Coordination (event-based, Publish-subscribe)
- Data storage and dissemination
- QoS management

The top three support a unified application view, while the existence of a well-recognizable QoS component is sometimes still an open question: there are several implementations where the QoS issue is dispersed among all the other components.

Group Management components allow to create groups and afford intra- and inter- group management issues. Groups can be created and maintained (members or resources incoming and outgoing), they can be related to one another (context), geographically located, anc can adapt to current environment. Groups may merge and create relationships with different objectives and requirements.

Coordination components usually employ a range of specific techniques, like pure event signalling, publish/subscribe systems, message passing systems, tuple space systems, and more. Each technique has different range of operations available, in any case delivery of interesting events to all interested parties is supported at various fault tolerance levels and different ordering relationships. Reconciliation procedures may be used to cope with inconsistent views; in the literature self-organizing and self-adapting systems are being proposed.

Data storage and dissemination components must be able to overcome unavailability of a specific resource by suitable guarantees on replication. Functions for replica control must be provided, such as for example

- Knowing the number of current replicas
- Maintaining a given number of replicas
- Disseminating information on replica placement

These operations must be implemented in accordance with applications requirements, and without imposing excessive overhead on resource consumption.

At the **application** level, a given set of techniques may be privileged over others because of specificities, but the whole set of problems appear of interest in all considered scenarios (those above and possibly others like healthcare, streaming, distributed gaming etc.

4 – Hardware and networking

The wireless mobile ad hoc network installed at the ISTI exploits the IEEE 802.11b technology. It is composed by the following components:

- Six laptops (Pentium III, 128 MB RAM), equipped with Debian Linux and Prism Linksys PCMCIA wireless network interface;
- One palmtop Sharp Zaurus (Pentium III, 128 MB RAM), equipped with Linux and D-Link DC650W wireless network interface;
- Two palmtop Compaq Ipaq (Pentium III, 128 MB RAM), equipped with Windows CE and Linksys WPC11 wireless network interface;

At the moment all the laptops and the Zaurus palmtop have been installed with OLSR routing protocol, to set up a multihop MANET. The DS^2 system has been developed within the activity of Workpackage 4 of the IS-MANET project by the ISTI U.O. It is a middleware layer providing dependable and secure storage of files in a MANET. The tests performed on the multihop MANET have been aimed at the evaluation of the performance experienced by a DS^2 users in the operations of file creation and removal and read.
All the experiments have been run on a three-host MANET based on the laptops and the OLSR routing protocol. The laptops form a logical two-hop MANET (a host in the middle connects the other hosts). In practice this configuration has been obtained by physically disporting the laptops at 2 meter from each other and hiding two laptops from each other (by a suitable configuration of the iptables).
On each laptop we installed the DS^2 server-process and we performed all the operation of creation, removal and read on one of the laptops on the side of the network. The applications have been used to perform repeated tests of file creations and removal over files of sizes ranging from 125KB and 2,5MB, and read blocks of size ranging from 0 to 64KB at random positions of the file.

From the point of view of satellite systems, MANETs are interesting because in difficult environments such the ones described above a satellite may be the only way that a MANET can be connected to the rest of the world. To this end, at least one of the MANET nodes must be equipped with a satellite modem. Such a connection raises some interesting problems. In fact, both the MANET and the satellite wireless links are different from the more common cabled networking environment: they are error-prone and offer limited and time-varying bandwidth. Unfortunately, the serialisation of the satellite and MANET environments brings the worst of both worlds as far as the link characteristics are concerned.
Applications have been divided into two categories: TCP applications and streaming (multimedia) applications. Both have been studied as typical of the intended scenarios.
The satellite link has been deployed with a receiving station at ISTI, which is connected via Internet to the transmitting station of Toulouse (F). In practice, the MANET can receive data from the satellite Hot Bird VI, and can send back possibile feedbacks only on the terrestrial link. At Toulouse, all traffic originating from ISTI is sent to the satellite and hence broadcasted, together with all other data relevant to other stations. This situation represents only the best possibile approximation to a disaster recovery situation, where the terrestrial link is not available, nevertheless it is possibile to evaluate networking parameters. A transmitting station at ISTI could be installed in the next future.

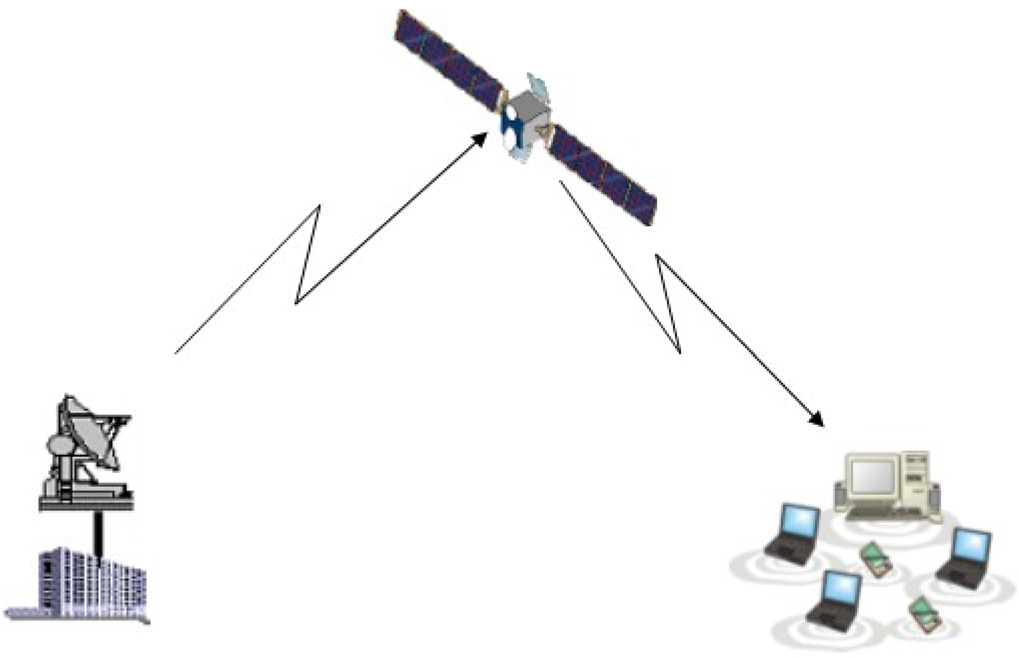

Figure 10. The MANET-satellite connection: the transmission is possible by Internet connection from ISTI to Toulouse

5. Mapping the developed prototypes to the target scenarios

From bottom layers up, the **networking** aspects of developed prototypes are mostly centered around MANETs made up bi *WiFi* connections (IEEE 802.11). However as explained in Section 4 of this document, the problems of a *satellite* connection have been studied by the ISTI-CNIT unit. As an alternative to WiFi, connections among peers can be made with *Bluetooth*: this activity has been explored by DEIS-BO in the activity connected to the JSR82ext prototype, as an extension of ubiQoS middleware.

At the base levels, there is a general agreement among project participants that the **routing** problem can effectively be solved by a mature implementation of OLSR, which has already been used for MANET deployment. Such protocol in fact has been used both in the third scenario presented above, and in the ISTI MANET: in both cases there are sound motivations for the choice. However some participants have investigated other directions by designing new routing algorithms, or by using also AODV.

The **naming** problem is especially significant when existing P2P networks merge due to their moving close. This problem has been investigated by the Messina group, where self configuring IP protocols have been developed and tested.

At the Advanced level, the problems arising for **group management** have been studied by the AGAPE middleware, which is a context aware group management system. The software has been developed by the DEIS-BO unit.

The **coordination** problem has been studied by various units, developing complementary solutions. This is the case of the REDS system developed by DEI –MI, a publish/subscribe system especially designed to handle efficiently problems like topology changes and group joins or partitions. To this respect it appears a complementary approach with that of AGAPE, the integration among the two systems is thus foreseen. A different strategy has been exploited by TOTA, developed by the Modena unit, and JmobiPeer, developed at Catania, which have adopted a different underlying coordination model.

Data dissemination has been studied by various units under different perspectives. The DS2 filesystem, developed by the ISTI Pisa unit, has a fragmented approach to file allocation intended to solve also security issues. It is foreseen to integrate it with the MobEYE system, developed by DISI Genova, which is capable of caching files in intermediate nodes along the routing path, thus reducing latency and energy costs. REDMAN (developed by DEIS BO) has another approach to data accessibility in dense MANETs: it selects replicas of important files with locality criteria.

The **Quality of Service** issue has been studied at DEIS BO developing the ubiQoS middleware, intended to potimize streaming applications; the CIP-QoS environment, developed at Messina, implements the DiffServ policy over a WiFi network connected to a wired one.

Many project participants have been using *streaming* as the most demanding **application** in order to evaluate their software. A system especially designed for streaming over a MANET is MuM, developed by DEIS BO, which shall be integrated with the CIP-QoS environment.