Intelligent Middle-Ware Architecture for Mobile Networks

Rayene Ben Rayana and Jean-Marie Bonnin

Institut TELECOM, TELECOM Bretagne, France,
Université européenne de Bretagne,
{rayene.benrayana,jm.bonnin}@telecom-bretagne.eu

Abstract. Recent advances in electronic and automotive industries as well as in wireless telecommunication technologies have drawn a new picture where each vehicle became "fully networked". Multiple stake-holders (network operators, drivers, car manufacturers, service providers, etc.) will participate in this emerging market, which could grow following various models. To free the market from technical constraints, it is important to return to the basics of the Internet, i.e., providing embarked devices with a fully operational Internet connectivity (IPv6).

A new device, the Mobile Router (MR), will take place in vehicle to manage mobility and take advantages of the surrounding wireless technology diversity to offer seamless IP connectivity to on-board devices. It has to take into account various constraints, in its decision regarding the management of wireless network interfaces and the routing of the flows. These constraints are many-fold. They could be technical, depend on the environment of the MR or on the flow characteristics. They also have to respect usage policies provided by stake-holders.

This leads to the necessity to design a middle-ware able to gather all kind of information and requirements and to provide the routing engine (at the network layer) with a comprehensive set of elementary rules. This article presents a MR architecture and show how such a middle-ware could make decisions which are context aware and policies aware while allowing a comprehensive resource management.

Keywords: Mobile Middleware, Heterogeneous networking, Context awareness, Mobile network, NEMO.

1 Introduction

Public transportation users are more and more interested in being able to use Internet-based applications while they travel. Having continuous connectivity can make their transportation time pleasant (browsing the web), efficient (consulting emails on the way to work) or even opportune (attending at a work session although being stuck in a traffic jam). Modern public transport companies are already providing such services for their waiting time in airports and train stations. Recently, in Europe, several commercial offers that provide such a service appeared. For example, travellers using the high speed train Thalys

C. Giannelli (Ed.): Mobilware, LNICST 0007, pp. 43–57, 2009.

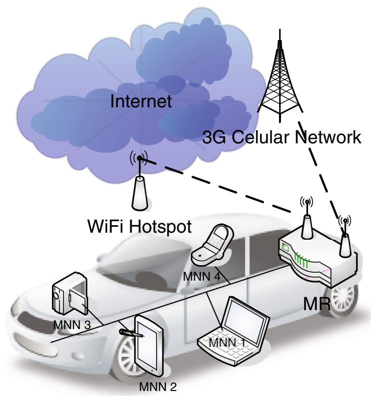

Fig. 1. Example of a Mobile Network

can remain on-line while crossing over the countryside at 350 km/h. The development of IPv6, the growing success of Intelligent Transport Systems (ITS) equipment and the wireless access network diversity will soon take us along to the next step.

As embarked devices and communication expenses become affordable, infotainment applications relying on the Internet connectivity will invade personal vehicles. Additionally, The growing interest in sustainable development issues is another important trigger for developing a "fully networked car"[1]. In fact, organising multi-modal transportation requires a tight coupling among different transportation systems through extensive usage of communications. This would help to provide travellers with up to date information helping them to reduce the overall trip time and global energy consumption.

Nevertheless, current technologies do not allow small-sized devices to have more than one or two wireless interfaces because of energy consumption, size and cost issues. Therefore, these devices would not benefit from the diversity offered by the various network technologies. They also will not be capable of managing mobility and maintaining an ubiquitous access to the Internet. However, considering the problem otherwise, one can notice that these devices are often used in environments such as personal vehicles and public transportation systems. Those environments can manage the ubiquitous access issue for the attached devices and provide them with a stable and easy-to-use access network (e.g., WiFi or Bluetooth).

Several aspects have to be studied to achieve a seamless mobility through multiple access networks. A first step was the design of the NEMO (NEtwork MObility) Basic Support protocol at the IETF. NEMO's approach introduces the Mobile Routers (MR) which will be part of modern vehicles and manage all complexities related to multi-interfaces and seamless mobility management.

[1] It is the name of a workshop organised each year by the ITU at the International Motor Show in Geneva.

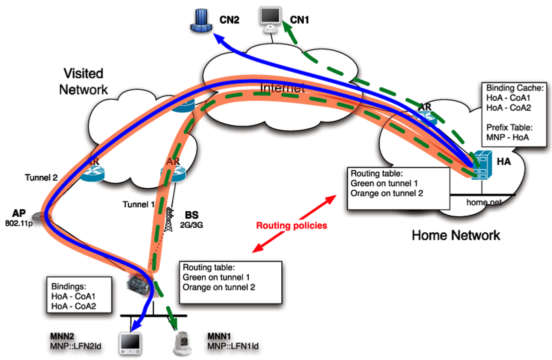

Fig. 2. Example of a NEMO + MCoA Use Case

A heterogeneous network is a combination of several access networks, each of them being optimised for some particular service. Consequently a comprehensive system should deliver each service through the network that is most efficient for that service. Different access networks may also be combined to increase the available capacity.

Many stake-holders, spanning from public authorities to end users via manufacturers, are involved in ITS-related services and most of these services suppose co-operation between them. Interfaces between various devices must be fully open to allow a well-balanced market development. Standardisation bodies are working on the definition of a communication architecture providing, among other features, Internet connectivity for vehicles. A survey of standardisation works leads to several conclusions. First, all the three main architectures (WAVE [12], C2C-CC [5], CALM [21]) design a management plan that has to decide while taking into account high level policies and requirements. It is a common form of cross layer architecture where it is needed to decide considering information coming from several communication layers. This management plan could be implemented as a distributed middle-ware embarked on board devices.

Second, IPv6 and companion protocols have been chosen as a network layer in all architecture. No other choice was reasonable since hundreds of billions of cars will need addresses and it is impossible to provide them in IPv4.

Third, if C2C-CC (Car to Car Communications Consortium) architecture allows the use of WiFi network interfaces along with the dedicated DSRC one, only CALM has been design to fully support multiple heterogeneous wireless access networks (802.11-based, 3G, WiMAX, Millimetre Waves, ...).

In addition to IPv6, ITU[2] has been chosen the NEMO Basic Support protocol [6] to manage mobility in the CALM Mobile router. The later is responsible for updating its position to a location server inside the operator network, namely the Home Agent (HA). All the traffic from, and to the mobile network (the on-board network) is conveyed through a tunnel established with the HA. Moreover,

[2] International Telecommunication Union (www.itu.int).

a recent work in progress (draft MCoA [20]) defines a way to establish simultaneously multiple tunnels. Another work in progress [10] allows to exchange routing policies with the HA to enforce them at both tunnel end-points (see fig.2).

These protocols provide technically the support of multiple interfaces in the TCP/IP architecture, but further works have to be made to develop a fully operational CALM architecture. In fact, the decision process and how it take into account various policies provided by stake-holders and requirements given by application is essential to properly use multiple interfaces.

This article is organised as follows : The next section explores some related works. The third section gives an overview of the proposed architecture. The fourth section focuses on how stake-holders policies are taken into account. Finally, the last section describes how the decision modules refine these policies and produce the corresponding system-level policies.

2 Related Works

The InternetCar Project (1996–2002) is one of the pioneer projects proposing a network mobility solution for vehicles. According to [11], it is based upon two main technologies: *Interface Switching* and Prefix Scope Binding Update (PSBU) which is similar to NEMO Basic Support. The MR uses a policy-based mechanism to choose the most suitable interface at a given moment called *Multiple Network Interface Support by Policy-Based Routing on Mobile IPv6*. The project highlights the importance of multi-homing to ensure continuous connectivity. Unfortunately, it was not possible to use simultaneously several interfaces.

MAR (Mobile Access Router) is a network mobility management framework developed in 2004 [14]. It dynamically instantiates channels based on traffic requests, aggregates the bandwidth and dynamically moves load from poor quality to better quality channels. MAR is based upon the Mobile IPv4 protocol rather than Mobile IPv6. It uses a policy-based mechanism to exploit simultaneously multiple interfaces through the concept of “virtual link”. In this solution the mobile router also takes care of packet losses and disordering to save TCP performances.

The integration of various access technologies in a mobile terminals has also been studied in several other works (e.g., [15,22]). The main idea is to provide a unified platform architecture offering a seamless integration of heterogeneous technologies (e.g., [1,24]). Previously proposed handoff solutions (e.g., [16,23]) try simply to keep the mobile user “Always Best Connected”. However, it is important to use for each running applications the most adapted technology among the currently available ones. At the same time, it is important to provide the best trade-off between bandwidth, bit error rate, one way delay and its induced cost (e.g., [7]).

Various vertical handoff schemes have been proposed recently to offer seamless session continuity. However, there is still a lack of mechanisms allowing a comprehensive network connectivity management while providing means to control some essential parameters such as monetary costs, device energy consumption and service satisfaction.

Ubique is a Mobile IPv6-based framework designed to offer ubiquitous access to mobile users [17,18]. Ubique proposes to generate routing policies dynamically regarding high-level profiles specified by the users, the administrator or the applications [4]. Adaptive applications communicate directly with the framework to inform it about new flows and to specify their requirements. The framework matches the flow requirements with the interfaces and available access networks. It ensures the respect of the administrator limitations and conveys the flow through the most suitable interface. Ubique has been partially adapted to be implemented in a mobile router in [9]. It profited from the ability to manage multiple tunnel toward the Home Agent (i.e., MCoA) to allow simultaneous use of multiple interfaces. It has been designed to cope with various preferences. Although few of them depend on the current environment, they are mainly given during the configuration process and do not allow the operator to control how resources are used. We also showed in [4] that it is possible to influence the behaviour of a fleet of terminals just modifying an administrator-given preference profile.

This quick survey of mobile router architectures points out the remaining need for a comprehensive middle-ware able to take care of multiple sets of policies potentially contradictory. Moreover, it has to take into account the current context to decide which part of policies apply in the considered situation.

3 Architecture Overview

A high-level mobility management framework has been developed in the context of the REMORA project[3]. As fig.3 suggests, the framework produces and enforces three kinds of policies: a flow routing policy, an interface management policy and an application management policy. To obtain them, decision modules process the preferences given by different stake-holders and try to maximise their satisfaction using input from monitoring modules to adapt the decisions to the context the network is evolving in. The result is then fed to enforcement modules to be applied.

The main purpose of this paper is to show how high-level stake-holders policies can be combined to obtain regular system-level policies and rules to be enforced at the network layer. The motivations of this work and the protocols used in the REMORA framework are extensively described in [2,3].

A simplified view of the REMORA framework appears in fig.4. It presents the three types of modules: Monitoring, Decision and Enforcement. Arrows schematise inter-module interaction. As you can see, these modules co-operate to achieve the three major tasks of the middle-ware: Flow Routing, Interface Management and Application Management.

3.1 Interface Management

The Interface Activation Policy Processor is in charge of continuously looking for access networks that fits better with application requirements and stake-holders

[3] This project is a collaborative project supported by the ANR (French government).

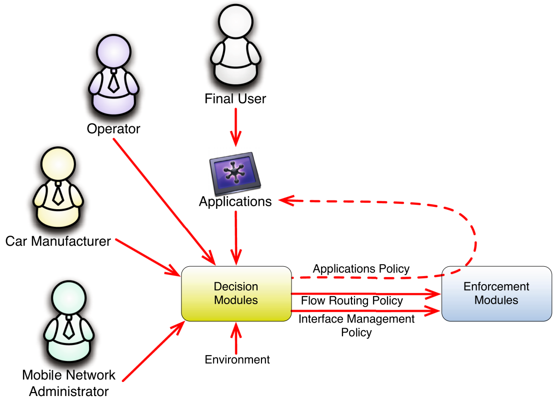

Fig. 3. Overview of the REMORA Mobile Router architecture

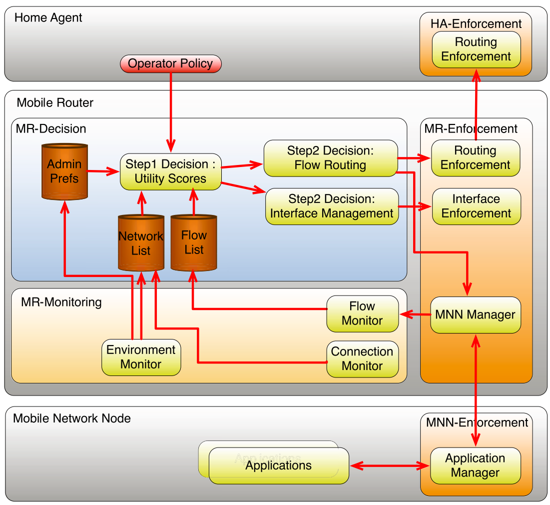

Fig. 4. Modular view of the REMORA architecture

policies. Additionally, this module is responsible for cost and power management. It shut-downs interfaces when they are not needed. This happens when there is no critical flows that require the interface and when the non-critical flows can be dispatched on the other interfaces. The best combination is then chosen and the module generates a policy that reflects this decision. Of course, hysteresis is necessary to avoid redirecting flows back and forth on the interfaces.

The result obtained is a list of interface-network associations. The indicated interfaces should be activated if needed and connected to the corresponding network. The other interfaces have to be deactivated. Once this policy enforced, the Flow Routing Decision Module will adapt itself and redirect the flows conveniently.

3.2 Flow Management

The Flow List contains the characteristics of the flows to be conveyed. For each flow it specifies the manner to recognise belonging packets (port numbers, IP addresses, . . .). The Flow Monitor adds information about whether the flow is alive or not and evaluates the throughput of each flow.

The Interface Management module manages to distribute the available resources regarding flow priorities and requirements. For example, when resource level declines, the MR can choose to drop FTP packets in favour of videoconference packets.

A list of flow-tunnel associations is produced. It specifies the flows that have to be sent (or received) through the corresponding tunnels. Packets that do not match any rule are simply discarded. The enforcement uses standard packet filtering of the underlying operating system to be highly portable (e.g., netfilter on Unix/Linux systems).

3.3 Application Management

It is essential to let applications take part in the mobility management as stated in [2]. Applications requirements vary through time and may depend on available resources. Some applications can adapt their behaviour to the network conditions experienced by the mobile router. For example, a video conference application can reduce video quality if the MR experiences a disconnection of one of its interfaces and have to reduce its throughput. Applications can also announce their requirement to feed the decision process. With these notifications, the MR will be able to dimension the overall requirements and decide if additional interfaces should be waked up or, on the contrary, shut down.

The CALM architecture [21] proposes such interactions between application and a management plan, which have to be present in MNNs and in MRs. Reference [2] shows that a simple middle-ware present in MNNs could free applications to implement complex adaptation policies while allowing advanced management of mobile network resources. It is also possible to consider the networking environment as a context and to use context awareness framework to exchange this information with the applications. It can also be interesting to use a CMS (Context Management System) [19] to share context-related information among nodes.

4 Stake-Holders Policies Awareness

As stated before, the matter of this paper is to highlight the ability of the proposed architecture to take into account high level considerations to produce the corresponding system-level policies. This considerations are described through policies given by various stake-holders (see fig.5). Three sets of rules are produced to manage interfaces, to route traffic and to allow application adaptation.

This decision could have a substantial economic impact on stake-holders. First, an operator can be interested in privileging the choice of networks that belong to it or that belong to its commercial partners. While the Mobile Network administrator can be tempted, on the contrary, to take advantage of the competition between several operators. In the case of mobile routers sold with vehicles, car manufacturers can also propose to customers to update their equipment with a configuration that corresponds to their commercial partnerships. Finally, the user using applications would be interested to influence the decision in a way to serve its interests. For these reasons, the architecture proposes means to be deal with several, and potentially contradictory, high level sets of policies.

4.1 Car Manufacturers and Operators Policy Awareness

It is expected that, in the near future, a customer could get a mobile router in two ways. It could be either provided built-in the car (built-in) or purchased a part on the market. In both cases, the manufacturer may propose a subscription to its policy update system which guarantee the costumer to take full advantages of the commercial partnerships of the manufacturer. The later could then have a certain control over the device.

Nothing have been specified in the CALM architecture to allow external policies to be taken into account in the decision process nor to exchange such policies.

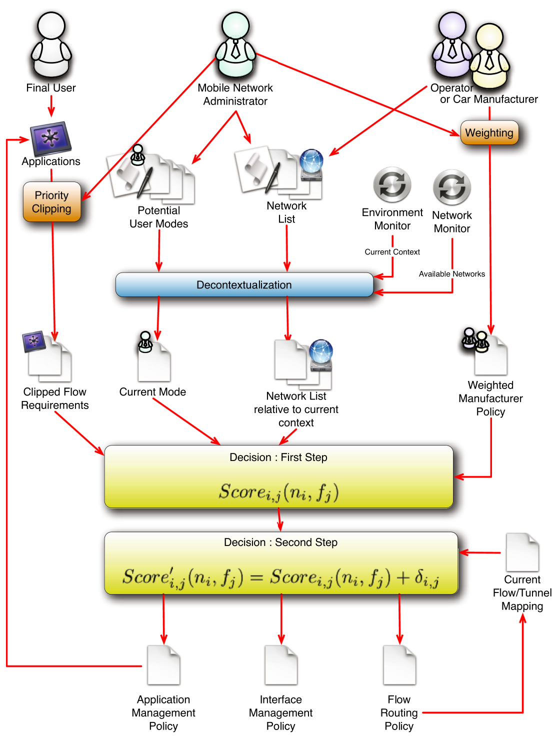

Fig. 5. From Stake-holders Policies to System Policies

Anyway, there is a policy description language that could be used to exchange policies [8].

The control concerns mainly the networks that the user will be allowed to connect to. The influence of the manufacturer on the decision can be handled in several ways. An interesting solution is given by [8]. The document specifies the syntax of the policy as well as the policy exchange scheme. It suggests that the manufacturer policy will provide a way to associate each (port number, network) combination with one of the following directives : (mandatory, optional, not mentioned, unadvised, forbidden). When a “mandatory” directive is associated with a (port, network) tuple, the network, if available, has to be used for the transfer of flows having this port number. Whereas, “optional” directive is just a hint given to the decision module to privilege a network over others having a “not specified” directive. This can be useful for load balancing over different access networks (i.e., privilege load-free networks).

In practice, most mobile network administrators will follow the manufacturer recommendations but more experienced users may want to limit the effect of the manufacturer on the decision to fit better with their interests. And they will be able to do that using a simple weighting mechanism (see fig.5).

4.2 Applications and Final Users Requirements Awareness

A flow notification consists in declaring the flow’s minimum requirements under which the flow cannot be properly sent. When two (or more) tunnels that fulfil these requirements are available, it is interesting for an application to choose a tunnel that is more adapted to its needs. For example, an FTP client will choose a tunnel with more bandwidth while a VoIP application would privilege security and steadiness. These are declared through the declaration of weights that will influence the decision algorithm. The flow notification and application adaptation mechanism is extensively described in [2].

```
TCP  20 Outbound                # FTP Downloading
    Priority = 2;               # The flow priority
    Privilege = download ;      #  The significant flow direction
    Download Requirements
        Cost <= 0.3 Euro/ MB;
        LossRate <= 10\% ;
    Upload  Requirements
         Cost <= 0.3 Euro / MB; # don't download if expensive
         LossRate <= 10\% ;     # don't download if too much loss
    Weights
        Cost = 60;              # cost is important
        LossRate = 20;
        ConnectionStability = 20 ;
        Security = 20;
        Jittter = 20;
        Bandwidth = 80          # bandwidth is important too
```

The example above is a very simplified version of what a flow declaration looks like. It illustrates an FTP client that refuses to operate if the cost is high or if the loss rate is important. In addition, it tells the decision algorithm that if it had to choose among several tunnels, it would prefer the ones with a higher bandwidth and a lower cost.

Each application associates a priority with its flows. A Mobile Network administrator can clip the priority of certain flows. With priority clipping, he can deny high priority flow declarations to "basic users" while granting access to "premium users" who will pay more for this privilege. A car driver can also refuse high priority flows requests coming from his children's game console to ensure that his own flows will be sent in better conditions.

4.3 Mobile Network Administrator Policy Awareness

Usually administrator preferences consist in static routing entries. This means that the system's administrator decides for each type of flow (based upon port numbers, IP addresses, etc.) the tunnels in which it will be routed. Nevertheless, using static rules is nor simple nor efficient. First, it is very hard for a human being to translate his high level considerations into routing rules. The result is rarely what is expected and the complexity grows with the number of users and interfaces. Second, the permanence of static rules contrasts with the changing network conditions, the changing stake-holders requirements, the changing applications needs and the changing context. It is by far more interesting for an administrator to express his high level objectives in a more natural way. For example, instead of telling the system to avoid sending FTP(port 20/21) flows into a 3G based tunnel because it is too expensive, he just tells the system to reduce the overall cost. Given the high-level objectives, the problem can be expressed as an optimisation problem and a comprehensive intermediary module is responsible to convert high level policies into system level rules. An example of an administrator policies description is given bellow.

```
Minimize Cost;
Minimize PowerConsumption;
Maximize Bandwidth;
Maximize ConnectionStability;
Maximize Security;
Minimize Jitter;
```

An administrator could specify various objectives. Anyway the system should have the capability to evaluate their fulfilment. For example, if an objective is to minimise the error rate, the system should integrate tools to evaluate (even roughly) the error rate of each route/tunnel.

If objectives rarely vary through the time, their relative importance could. For example, power consumption is less important when the engine is on than it is when the engine is stopped. The security (encryption level) of the link is more important than cost during working hours and vice versa.

To formalise that, the administrator specifies a weight for each objective that relativize its importance. This leads to the definition of several Operating Modes. The administrator must provide an operating mode corresponding to the behaviour of the system in a particular situation as seen in the example below:

```
Mode: Work                      Mode: CostEconomy               Mode: PowerEconomy
    Cost = 10                       Cost = 99                       Cost = 50
    PowerConsumption = 0            PowerConsumption = 0            PowerConsumption = 99
    Bandwidth = 70                  Bandwidth = 20                  Bandwidth = 20
    ConnectionStability = 80        ConnectionStability = 10        ConnectionStability = 20
    Security = 99                   Security = 0                    Security = 100
```

In this example, the administrator declares three modes. The first mode gives more importance to security and QoS than the other modes. It is planned to be used during working hours. The second gives more importance to the cost and can be used during weekends and holidays. Finally, the last mode can be used when the battery level becomes low.

Only one mode can be active at the same time. Of course, this mode can be selected by the administrator, but, this requires an interaction with the driver, which can be annoying and dangerous.

We prefer by far the idea of a fully autonomous system that is configured using solely high-level parameters that allow the system to react in function of the current context and to select the appropriate operating mode.

5 Context Awareness

As stated before, the administrator needs depend closely on the context. When environment changes, weights change and thus, the relative importance of objectives changes. Therefore, the system configuration has to be decontextualised. In other words, instead of the operating mode to be used, the administrator should rather tell the system under which conditions he would have set this mode. To achieve this, it could use a pseudo-algorithm-based description, which allows to express conditional statements in a human-like language. Below, you can find an example of an administrator configuration file.

```
VehicleLocation in ('France')
    BatteryLevel in [20..100]
        DayOfWeek in ('Saturday', 'Sunday')
            RETURN CostEconomy
        DayOfWeek in other
            HourOfDay in [8..12, 14..18];
                    RETURN Work
            HourOfDay in other;
                RETURN CostEconomy
    BatteryLevel in [0..20];
        RETURN PowerEconomy
VehicleLocation in other;
        RETURN CostEconomy
```

In this example, the user chose to give priority to cost when he is abroad. When he is at home (i.e., France), the user privileges low energy consumption when battery level is low. As the company is charged during work hours, the user privileges quality over cost during the day and the cost reduction otherwise.

The pseudo algorithm-based description allows to express the administrator needs in a natural way. It allows an infinite number of configurations that fit the needs of every user.

You will also notice that this mechanism requires to be fed with up-to-date values for the time, the battery level and the geographic position. A module called “Environment Monitor” is responsible for fetching this information which used to decontextualise the policy description. The system is able to provide the user mode that would have been set by the administrator in the current context. The selected mode is then fed to the decision process to influence its output in a way that serves administrator’s interests.

Decontextualisation can be very useful to automate the selection of the mode. But, user modes are not the only parameter that changes depending on environment. Access network characteristics also depend on the context. For example, the current date and the time of the day give are necessary to choose the cheaper networks if some of them have complex billing schemes (e.g., UMTS is cheaper from 8 *p.m.* to 6 *a.m.* and during weekends, roaming is expensive, etc.).

Vehicle speed could be used to privilege WiFi networks when the vehicle is stationary and 3G networks when it is on the move. The example bellow shows two network entries with parameters decontextualised using the policy description illustrating the examples above.

```
Network UMTS_1
    ConnectionScript = "pppd ..."
    ConnectionStability = high
    Security = High
    Download, Upload
        Cost  = CASE VehicleLocation in ('France')
                    CASE HourOfDay in [6..20]
                        RETURN 0.3 Euro/MB
                    ELSE RETURN 0.1 Euro/MB
                ELSE VehicleLocation RETURN 1 Euro/MB
        Bandwidth = 600kb/s
        Jitter = DEFAULT
        BitErrorRate = DEFAULT
        Delay = DEFAULT
```

```
Network WiFi_1
    ConnectionScript = "iwconfig ..."
    ConnectionStability =   CASE VehicleSpeed  [0..20] RETURN high
                            CASE VehicleSpeed [20..50] RETURN medium
                            ELSE RETURN low
    Security = Low
    Download, Upload
        Cost  =  0
        Bandwidth = 2000kb/s
        Jitter = DEFAULT
        BitErrorRate = DEFAULT
        Delay = DEFAULT
```

6 Processing Policies and Producing Rules

Once the current mode has been set, applications requirements have been filtered and the network list updated regarding the current context, the decision process can be launched. The output of the decision modules are, as stated before (see fig. 5), three system policies: the flow routing policy, the interface management policy and the application management policy.

This decision process have to take into account high level considerations such as operator preferences, all available networks and all running flows. Such important input could result in a heavy process that could not be triggered each time a significant event happens. Especially, in a mobility context, where networks events (handover, disconnection, etc.) are frequent and require fast reaction that cannot be afforded while taking into account such high level consideration. To overcome this, the decision process is split in two phases.

The first step consists in evaluating the matching degree of flows with the networks stored in the Network List while respecting the restrictions and the preferences of the manufacturer, the administrator and the flows. To achieve this, a utility score is calculated for each network-flow tuple ($Score_{i,j}(n_i, f_j)$) where n_i is an entry in the network list and f_j is an entry in the flow list. Of course, the greater the score is, the more compatible the tuple is. Having a score of zero means that the flow f_j must not be conveyed through the network n_i. The score of a network-flow tuple is weighted with the manufacturer policy associated with this tuple. For example, if the manufacturer policy tells the system to prevent a flow f_j from being sent through a network n_i and if the network administrator gives a full control to the manufacturer, the obtained $Score_{i,j}$ will be set to 0 even if the tuple is highly compatible. In this first step, each flow is considered on its own, as if it is the only flow to be treated in the system. The output, and therefore the input, of this first step, does not change very often, therefore it is not executed very often.

The second step considers flows as a whole and tries to apportion the available resources to them regarding their $Score_{i,j}$, their priority and the remaining network resources. In practice, this can be assimilated to the addition of a correction parameter called $\delta_{i,j}$ which is high for high priority flows that require abundant resource and is negative for low priority flows. This could require to send it over a heavily loaded link since if the overall score becomes negative for a link, the flow must not be sent over it. If a flow does not obtain a positive score for at least one candidate network, it is simply discarded. At the end of this second step, the MR translates the obtained scores into the flow routing policy and applications are made aware of the changes if any. Additionally, a utility score is computed for each network to find out which are the less used, respectively the much solicited, ones. This is used to decide to shutdown, respectively to wake up and configure, corresponding interfaces.

7 Conclusion

Taking advantage of the Internet flexibility is for sure the right way to simplify the development of various ITS services (from security to infotainment). This paper deals with the network diversity (multi-homing) management and gives an overview of various works done at standardisation bodies and in the academic world. It identifies the missing part in current standardisation proposals of a Mobile Router architecture such as the CALM architecture designed at ISO TC 204 WG 16 (see [4]). It also sketches up the architecture designed in the REMORA project. In addition, it shows how context awareness is integrated in the multi-interfaces management middle-ware. Moreover, it is tightly coupled with what we call policies awareness to feed a comprehensive decision process which is able to take high level consideration to route the flow.

Further works are necessary to design a fully operational CALM implementation and more generally a fully operational NEMO Mobile Router. It is first necessary to provide a way to distribute decision and routing operation when several mobile routers are present in the mobile network. Moreover, the management plan should be able to interact smoothly with routing optimisation [13] and non-IP networking layers devoted to V2V communications such as CALM FAST and Geo-casting.

This works has also been the occasion to work on the relation between the MR and applications running on-board [2]. The REMORA architecture has been implemented as a proof-of-concept with which experiments and demonstrations have been successfully conducted.

Additionally, we are developing a network emulator to study distributed versus centralised resource management approaches. This will allow us to study carefully the relation between decision mechanisms embedded into adaptive applications and the one built-in the mobile router middle-ware.

Remark: First results will certainly be ready by the time to deliver the camera-ready version of this article. They will be presented during the conference. These first results will give a functional-proof of the design in various ITS scenarios and compare the behaviour of the REMORA architecture with the one of a basic NEMO mobile router statically configured.

References

1. André, F., Bonnin, J.-M., Deniaud, B., Guillouard, K., Montavont, N., Noël, T., Suciu, L.: Optimized support of multiple wireless interfaces within an ipv6 end-terminal. In: SOC 2003 (Smart Objects Conference), Grenoble, France (May 2003)
2. Ben Rayana, R., Bonnin, J.M.: Mobility aware application manager for mobile networks. In: ITST 2008 (Intelligent Transportation Systems and Telecommunication), Phuket, Thailand (October 2008)
3. Bonnin, J.-M., Rayana, R.B.: Wireless Technologies for Intelligent Transportation Systems. In: Heterogeneous Networks and Mobility Management for ITS (2009)
4. Bonnin, J.-M., Lassoued, I., Hamouda, Z.B.: Automatic multi-interface management through profile handling. In: Mobile networks and applications, special issue on Mobile Middleware. Springer, Heidelberg (2008) (online)
5. C2C Communications Consortium. Car-2-car communication consortium - manifesto (2007)
6. Devarapalli, V., Wakikawa, R., Petrescu, A., Thubert, P.: Network Mobility (NEMO) Basic Support Protocol. RFC 3963 (Proposed Standard) (January 2005)
7. Gustafsson, E., Jonsson, A.: Always best connected. IEEE Wireless Communications (February 2003)
8. Larson, C., Eriksson, M., Mitsuya, K., Tasaka, K., Kuntz, R.: Flow distribution rule language for multi-access nodes. IETF working draft (work in progress) (July 2008)
9. Mahamat-faki, M., Bonnin, J.-M., Ben Rayana, R.: Traffic flow control on multi-tunneled mobile router. In: short paper in WNEPT2006 (Workshop on Networking in Public Transport), Waterloo, Canada (August 2006)
10. Mitsuya, K., Tsakada, K., Wakikawa, R., Kuntz, R.: A policy data set for flow distribution. Draft IETF (Work in progress) draft-mitsuya-monami6-flow-distribution-policy-04.txt (August 2007)
11. Mitsuya, K., Uehara, K., Murai, J.: The In-vehicle Router System to support Network Mobility. In: International Conference on Information Networking (ICOIN), South Korea (February 2003)
12. IEEE P1609. Ieee p1609.1 standard for wireless access in vehicular environment (wave) - multi-channel operation. IEEE draft Standard (2006)
13. Perkins, C.E., Johnson, D.B.: Route Optimization for Mobile IP. Cluster Computing 1(2), 161–176 (1998)
14. Rodriguez, P., Chakravorty, R., Chesterfield, J., Pratt, I., Banerjee, S.: Mar: a commuter router infrastructure for the mobile internet. In: MobiSys 2004: Proceedings of the 2nd international conference on Mobile systems, applications, and services, pp. 217–230. ACM Press, New York (2004)
15. Siddiqui, F., Zeadally, S.: Mobility management across hybrid wireless networks: Trends and challenges. Computer Communications 29(9), 1363–1385 (2006)
16. Stevens-Navarro, E., Wong, V.W.S.: Comparison between vertical handoff decision algorithms for heterogeneous wireless networks. In: IEEE Vehicular Technology Conference (VTC), Melbourne/Australia (May 2006)

17. Suciu, L., Bonnin, J.-M., Guillouard, K., Stévant, B.: Achieving "always best connected" through extensive profile management. In: Niemegeers, I.G.M.M., de Groot, S.H. (eds.) PWC 2004. LNCS, vol. 3260, pp. 421–430. Springer, Heidelberg (2004)
18. Suciu, L., Bonnin, J.-M., Guillouard, K., Stévant, B.: Towards a highly adaptable user-centric terminal architecture. In: The 7th Intl. Symposium on Wireless Personal Multimedia Communications (WPMC 2004), Abano, Italie (September 2004)
19. van Kranenburg, H., Bargh, S., Iacob, M.S., Peddemors, A.: A context management framework for supporting context-aware distributed applications. pp. 67–74 (August 2006)
20. Wakikawa, R., Uehara, K., Ernst, T., Nagamix, K.: Multiple Care-of Address Registration. Draft IETF (Work in progress) draft-wakikawa-mobileip-multiplecoa-05.txt (February 2006)
21. Williams, B.: The calm handbook: Continuous communications with and between vehicles. ISO/TC204/WG16 (work in progess v2.0) (November 2005).
22. Williams, M.: Directions in media independent handover. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences - Special Section on Multi-dimensional Mobile Information Networks (July 2005)
23. Xing, B., Venkatasubramanian, N.: Multi-constraint dynamic access selection in always best connected networks. In: IEEE MobiQuitous 2005, pp. 56–64, San Diego/CA/USA (July 2005)
24. Zhang, W., Jaehnert, J., Dolzer, K.: Design and evaluation of a handover decision strategy for 4th generation mobile networks. In: 57th Semiannual Vehicular Technology Conference (VTC 2003-Spring) (April 2003)