Transport over Heterogeneous Networks Using the RINA Architecture

Eleni Trouva, Eduard Grasa, John Day, Ibrahim Matta, Lubomir T. Chitkushev, Steve Bunch, Miguel Leon, Patrick Phelan, Xavier Hesselbach-Serra

▶ **To cite this version:**

Eleni Trouva, Eduard Grasa, John Day, Ibrahim Matta, Lubomir T. Chitkushev, et al.. Transport over Heterogeneous Networks Using the RINA Architecture. 9th Wired/Wireless Internet Communications (WWIC), Jun 2011, Vilanova i la Geltrú, Spain. pp.297-308, 10.1007/978-3-642-21560-5_25 . hal-01583665

HAL Id: hal-01583665
https://inria.hal.science/hal-01583665

Submitted on 7 Sep 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Transport over Heterogeneous Networks Using the RINA Architecture

Eleni Trouva, Eduard Grasa[1]
John Day, Ibrahim Matta, Lubomir T. Chitkushev[2]
Steve Bunch[3]
Miguel Ponce de Leon, and Patrick Phelan[4]
Xavier Hesselbach-Serra[5]

[1] i2CAT Foundation, Jordi Girona, Barcelona, Spain
[2] Computer Science, Boston University, Massachusetts, USA
[3] TRIA Network Systems, LLC, Illinois, USA
[4] TSSG, Waterford Institute of Technology, Ireland
[5] Technical University of Catalonia, UPC, Barcelona, Spain
{eleni.trouva,eduard.grasa}@i2cat.net
{day,matta,ltc}@bu.edu
steve.bunch@ieee.org
{miguelpdl,pphelan}@tssg.org
xavierh@entel.upc.edu

Abstract. The evolution of various wireless technologies has greatly increased the interest in heterogeneous networks, in which the mobile users can enjoy services while roaming between different networks. The current Internet architecture does not seem to cope with the modern networking trends and the growing application demands for performance, stability and efficiency, as the integration of different technologies faces many problems. In this paper, we focus on the issues raised when attempting to provide seamless mobility over a hybrid environment. We highlight the shortcomings of the current architecture, discuss some of the proposed solutions and try to identify the key choices that lead to failure. Finally, we introduce RINA (Recursive Inter-Network Architecture), a newly-proposed network architecture that achieves to integrate networks of different characteristics inherently and show a simple example that demonstrates this feature.

Keywords: Recursive Inter-Network Architecture (RINA), heterogeneous networks, data transport, mobility, hand-offs

1 Introduction

The Internet originated from and still is primarily a wired network. However, with the years several wireless technologies evolved to meet the growing need for mobility. Users have come to expect access from anyplace, anywhere. In order to make this vision a reality, in the future users will have a greater choice between a wide range of services from various access networks, wired and wireless, giving

a new meaning to the Internet as an heterogeneous network. In the effort to migrate the current Internet architecture to a hybrid environment, the research community has encountered many problems. One of the main difficulties arose in providing seamless mobility, while achieving the best utilization of the resources.

As transport over heterogeneous networks must be one of the main features of the future network, it should be faced as a chance to re-evaluate the current Internet architecture. In this paper, we review the challenges encountered, examine the proposed solutions and analyze their inability to provide a global solution that addresses all the problems. Next, we explore how these issues are handled when based on fundamental principles as in the Recursive Inter-Network Architecture (RINA), a clean-slate architecture and present a realistic example of RINA over different physical media.

2 Challenges in Mobility over Heterogeneous Networks

2.1 Characteristics of the Hybrid Paths

Hybrid paths exhibit varied behavior across different physical media and technologies as we move from one end of the network to the other. A plethora of available hardware technologies that carry the signal exist and can be classified into guided (wired) or unguided (wireless). In wireless communications, many physical phenomena, such as fading, shadowing, path loss, affect transmission and put impairments in the propagation of the electromagnetic waves. As a result, wireless channels are characterized by higher bit error rates (BER), lower bandwidth and longer round trip times (often variable) comparing to the wired ones.

TCP's performance over hybrid environments has been extensively studied and proven inefficient with the wireless media [2, 5, 8–10]. Specifically, the congestion control mechanisms of TCP are the main reason why TCP fails to adapt to an heterogeneous environment. As already mentioned, BER is high in wireless links, whilst negligible in wired. In addition, wireless links are characterized by interrupted connectivity and frequent hand-offs. Media related losses that happen in wireless links are falsely interpreted by TCP for losses due to congestion in the network and trigger congestion control mechanisms. Each "false alarm" forces a reduction in the TCP window size, which results in a severe drop in throughput. This assumption leads to degradation of performance. In essence, TCP has network congestion control, rather than Internet congestion control.

Another issue occurs from the default behavior of TCP to increase the window size proportionally to the rate of the incoming acknowledgments (ACKs). In hybrid paths, the wireless nature imposes high delays, which implies a slower rate in opening the window compared to pure wired links, although the channel might have the capacity for higher throughput. Most of the operations performed from a mobile terminal (e.g. e-mail, web browsing) require the transmission of small amounts of data on short-lived flows. In TCP a sender starts to transmit data in slow start phase and gradually increases the window and thus, the

throughput. For short-lived flows, there is high possibility that the operation will be completed while the sender is still in slow start. This is another case in which the native TCP congestion control mechanisms lead to poor use of the available resources.

To alleviate the problems of traditional TCP new implementations have considered additional mechanisms for congestion control: Congestion Avoidance (Tahoe, Reno, Vegas), Fast Retransmit (Tahoe, New Reno), Fast Recovery (Reno, New Reno) and Selective Acknowledgment (SACK) are some of the algorithms introduced. Nashiry et al. [6] and Henna [7] show results of performance measurements of the TCP variants. However, the problem of opening the window according to the received ACKs rate has remained. Moreover, many TCP modifications were proposed to address the problem, such as TCP SAK, TCP FACK and TCP Santa Cruz.

Apart from these proposals, other approaches tried to address the unsatisfactory performance in heterogeneous environments. One of the ideas proposed was to split a connection over a hybrid path to wired and wireless part, for example at the base station. In this way, the base station has the role of acknowledging every packet back to the source. This solution breaks the end-to-end reliability. It is possible that an ACK arrives to the source without the packet actually has arrived to the destination. Another approach is to perform actions in the data link layer and hide it from the transport layer in order to avoid the problems with TCP. Modifications outside TCP were also proposed. Explicit Congestion Notification (ECN) is an example, in which explicit notification for congestion is signaled by the routers. In addition, some new transport layer protocols were designed with the wired/wireless environment in consideration (e.g. WTCP). Pentikousis [5] provides an extended review of the proposed solutions.

Many of the proposals mentioned above managed to improve the performance of TCP over heterogeneous networks. Yet, none of them manages to address all the problems caused by the characteristics of the hybrid paths, as TCP was not originally designed for such an environment. However, some of the proposed solutions provide some insight on what is the root of the problem. Splitting the hybrid path to wired and wireless for example, underlines the necessity of having a way to treat parts of the network differently, depending on their characteristics, e.g. wired versus wireless links. TCP does not provide us with this functionality as it acts in a global scope, not allowing any further configuration. Taking action in the data link layer and hiding the problem from the transport layer points to the radical idea of explicitly recognizing in its structure that the Internet is a network of networks and points to an architecture of layers that perform the same functions over different scopes. For example, a layer could manage a wired network, another layer could manage a wireless network, and a layer on top of them managing the end to end inter-network.

2.2 Mobility and Hand-offs

As a user moves away from the Access Point (AP) that attaches him to the network, the link quality drops gradually, but with luck another AP is getting

closer and can attach to it. Mobility over heterogeneous networks involves hand-offs, as usually the last part of the path before the users terminal is a wireless link. The whole hand-off procedure can be divided in phases: a) discovery, b) selection, b) de-attachment, c) authentication (if required), d) re-attachment. Whether the de-attachment phase precedes or follows the re-attachment to the new AP is a matter of policy or availability. In the case that the de-attachment follows the re-attachment, the mobile node will be multi-homed by two APs for a period of time.

There are two kinds of hand-offs, horizontal and vertical. In an horizontal hand-off, the mobile terminal performs a switch between two wireless APs that belong to the same technology. In a vertical hand-off, APs of different technologies are involved. The switch from one Point of Attachment (PoA) to another might happen while transferring data to another node of the network. The challenge is to allow the user to roam freely between different networks without interruptions or degradation of performance due to hand-offs. The current Internet architecture does not provide a way to address mobility that involves vertical hand-offs. The problem originates from the choice to provide names only for PoAs (IP addresses) and not for the nodes in the naming schema. If the new AP, to which the mobile node attaches, belongs to the same domain with the previous AP, the same IP address may be used. But if the APs belong to different domains (usually the case in vertical hand-offs), the mobile node will be assigned a new IP address and the connection will break due to the infamous TCP pseudo header, which contains the source and the destination IP addresses. The inevitable change of IP address means loss of the nodes identity. Another problem is that during a switch from a fast network to a slow network (and vice versa), the TCP sender does not have a fast way to adapt to the new characteristics of the channel and update accordingly parameters such as the transmission rate, the congestion window and the RTO (retransmission timeout) estimate in order to achieve the best utilization of the resources. Adaptation to the parameters takes place slowly, after several round trips, as the transport layer learns the characteristics of the new channel implicitly by probing it.

TCP was not designed with hand-offs in mind nor is it clear that it should have been. Again, several solutions have been proposed to deal with the problems occurring during a hand-off. The Internet Engineering Task Forces (IETF) Mobile IP uses a foreign agent to update the location of the mobile node at its home agent, thus allowing the mobile node to keep its IP address even when moving. When a mobile node leaves its home router and is attached to a foreign router, the foreign router informs the home router of a care-of address (CoA) to which all packets with destination the mobile node should be sent. An IP tunnel is created between the home agent and the foreign agent with a direct connection to the mobile node. When PDUs are sent to the mobile node with the old IP address, they are routed to the home agent. The home agent knows the mobile node has changed its location. The PDUs are encapsulated and sent down the IP tunnel to the foreign network. There the PDUs are forwarded on the CoA address that connects the mobile node to the foreign agent. Mobile IP

achieves mobility, although several issues arise from the use of tunnels. IP tunnels are hard to configure, create various security problems and represent single points of failure (if the home or the foreign agent fail, communication is lost). Traffic going through the home agent distorts the RTT estimates. In addition, extra cost and overhead is added due to the necessary location update messages and registration messages to the home agent. Propagation of the new position of the mobile node will increase the delay perceived by the user and during this location update, some packets may get dropped. As TCP cannot be aware that the loses are a consequence of the hand-off, it will handle the dropped packets as loses due to congestion. A decrease in the TCP window will be forced, resulting in a drop in throughput. The sender will have to wait for the retransmission timeout to expire, to be able to send packets again. The probability of packets re-ordering at the destination will increase.

Several modifications and enhancements to the TCP protocol have been proposed [5, 12] in an attempt to deal with the packet losses and the performance degradation of TCP during a hand-off. One of the ideas is to perform a make-before-break hand-off [13], in which the connection to the old AP breaks after the connection to the new AP has been established to avoid losses. Another proposal is to make TCP aware of the path change due to a vertical hand-off. This information is signaled to the TCP sender from the lower layers of the stack by setting a new TCP option. After the hand-off the sender sets parameters such as the congestion window, RTO as specified for a new connection. To avoid the long delay caused by waiting for the retransmission timeout to expire, the Fast Retransmit algorithm [14] has been introduced. A similar proposal is the use of a TCP retransmission trigger option, which notifies the TCP sender to retransmit packets immediately after the connection to the new AP has been established. Another solution relies on the fact that the TCP receiver on the mobile node explicitly notifies the TCP sender that a hand-off has just taken place and thus, the receiver can take appropriate actions. Packet buffering at the base station [11] is another approach to avoid losses (split-connection protocol, snoop). Again many of the proposed solutions achieve improvement in TCP hand-off performance. However, it seems that once again we miss the root of the problem that needs to be addressed, which is the incomplete naming schema of the current Internet architecture.

3 Recursive Inter-Network Architecture (RINA) over Heterogeneous Networks

Reviewing the proposals made, it is clear that mobility over heterogeneous networks is a very active research area. However, each proposal offers an improvement to one or some of the existing issues but does not address all the problematic points. Several of the solutions work well under specific circumstances and under-perform when the parameters change. None attempts to solve the fundamental problem. As it was already implied, mobility over hybrid environments is incompatible with the current Internet architecture. Trying to stretch

the capabilities of the current architecture, does not seem to provide an answer. In the following sections of the paper we present RINA, a clean-slate network architecture that inherently manages to integrate networks of different physical media.

3.1 Main Principles of RINA

RINA is a network architecture proposed by John Day in his book 'Patterns in Network Architecture: A Return to Fundamentals" [1]. RINA is based on the view that networking is only inter-process communication (IPC). The core element of the architecture is a Distributed IPC Facility (DIF). A DIF is a distributed application that provides IPC for a given range of QoS. Any two application processes[6] in different systems are able to communicate using the services provided by a DIF. DIFs of higher levels are able to use the services provided by DIFs of lower levels, forming a recursive structure. A DIF is a layer, but not in the same sense as in the current Internet architecture[7]. In RINA all layers use the same protocols to perform a coordinated set of policy managed functions and achieve the desired IPC service. A detailed description of the RINA model, as well as its features, can be found in [1, 4]. In this paper we intend to explore the way RINA deals with transport over heterogeneous networks by examining how mobility and hand-offs are handled.

3.2 Hybrid Paths in RINA

RINA provides a recursive architecture formed by different layers stacked one on top of the other. Each layer is self-contained, and operates a complete set of mechanisms to provide data transport services to the client layers, the layers above. These mechanisms include data transport, relaying and multiplexing of data packets, transmission control, flow control, congestion control, authentication and others. A key feature of RINA is that the protocols implementing these mechanisms are the same in each layer, but operate under different configurations (policies) in order to make their operation optimal for certain conditions. The structure provided by RINA perfectly suits the requirements of transport over heterogeneous paths, as with RINA there is no need for mechanisms or algorithms that work well for all types of physical media. Each physical medium has its own dedicated DIF, tailored to the specific medium characteristics. In fact, the configuration of the lower level DIFs aims to match the best way possible the QoS requirements set by the higher level DIFs. Applications that use the transport services provided by DIFs still can have end-to-end mechanisms, such as end-to-end error, flow, and congestion control, as they typically use a DIF that is sitting on top of multiple lower level DIFs, which in turn operate on top

[6] For clarity, we refer to application processes that are members of a DIF as IPC Processes and users of a DIF as application processes or simply applications.

[7] To compare to the Internet, (MAC, LLC) are one layer; Network and Transport are one layer. They are complete units of IPC.

of different physical media. Examples of this recursive structure are illustrated in Fig(s). 2, 3 and 4.

3.3 Mobility and Hand-offs in RINA

RINA provides names for applications/nodes rather than interfaces (PoAs). This choice permits routing on the node level and not on the interface level, as in the current Internet architecture and thus, multihoming is supported. In fact, it is provided as a consequence of the structure without the need of any special protocols.

Mobility can be seen as a dynamic version of multihoming with "expected failures". As the mobile node moves, it acquires new points of attachments that connect it to the network. In RINA each new PoA means that the mobile system joins a new DIF. Thus, while moving, the mobile node joins new DIFs and drops its participation in the old ones, a procedure that is equivalent to the attachment and de-attachment to APs during a hand-off. At times, the mobile node might be a member of more than one DIF (multi-homed), which is analogous to a mobile terminal that is in range of several APs and is connected to the network by more than one of them.

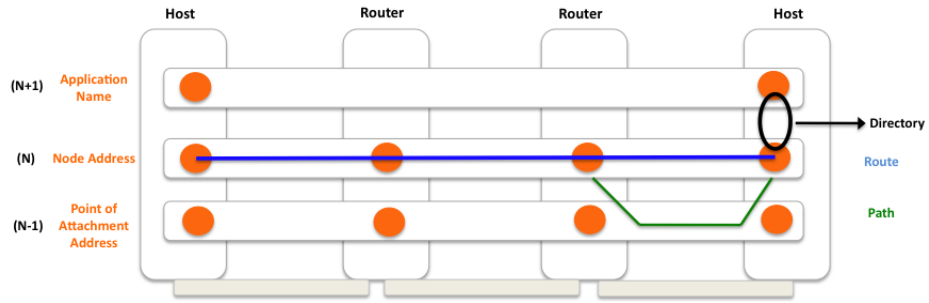

Fig. 1. Translating a route to a path in RINA naming and addressing schema.

To better understand how mobility works, a closer look to the naming and addressing schema used in RINA is required. Figure 1 shows an example of the RINA architecture with two levels of DIFs. A DIF maps the source and the destination application names to node addresses (IPC processes) through the directory function. Then, the route from a source application to a destination application is computed as a sequence of (N)-addresses (the blue line in Fig.1), where the next hop is a node address. Each IPC process knows how to map (N)-addresses to (N-1)-addresses for all nearest neighbors to determine the path to the next hop. Having the addressing schema of RINA in mind, multihoming could now be expressed as an IPC process having more than one (N-1) mapping and mobility as acquiring new (N-1) mappings to use and maybe losing

some others. Thus, in any hand-off scenario, vertical or horizontal, indifferent of the technology of the physical media, connecting to a new AP means that an IPC process acquires a new (N-1) binding. For the incoming flows, this new binding is the last binding in the translation of a route to a path connecting a source IPC process to a destination IPC process, while for the outgoing flows is the first binding. The cost for updating this new binding is small, since the update is done locally during the route to path translation. Ishakian et al. [3] provide further analysis on this cost. Acquiring or dropping (N-1) bindings may modify the graph (topology) at the (N)-layer. These changes are disseminated through routing updates, which will trigger the recalculation of the local forwarding tables of the affected IPC-processes of the N-layer. It is important to note that routing updates carry information about changes in the graph only regarding IPC process addresses ((N)-addresses). The bindings of (N)-addresses to (N-1)-address are a local matter of each IPC process.

4 An Example of Hand-off in RINA Architecture

In this section, we present a simple hand-off scenario in order to explain how transport over heterogeneous networks is handled in RINA architecture. Our scenario involves a user moving around with a mobile terminal causing hand-offs (Fig. 2). The mobile terminal has two different interfaces active (e.g. Wi-Fi and GPRS). An application process running on the user's terminal has opened a connection with an application process running on a server and data transfer is taking place (e.g. the user is streaming live video). While the connection is open and packets are in transit, the user moves physically, forcing a hand-off from the one AP to the other. In our example this is vertical hand-off, as the first AP is a WLAN router (e.g. Wi-Fi) and the second AP is a router of some technology belonging to the Long Term Evolution (LTE) standard (e.g. GPRS, WiMAX).

Having both APs connected to the same router that the server is connected to is unlike to happen in reality, but we are pointing to a simplified scenario that can be easily pictured and understood by the reader. However, this choice does not spoil the generality, since the number of the possible intermediates does not affect the way mobility and hand-offs are treated in RINA.

Figure 2 depicts the case in which the user is in the range only of the Wi-Fi and the terminal is attached to the network through the first AP. In the lower part of figure we can see the RINA architecture for this case. We illustrate as "source" the system of the server that shares content to the mobile user and as "destination" the system of the mobile terminal. "Intermediate" is an intermediate router that connects through a wired link the server from the one side and the two APs (WLAN and LTE routers) from the other. The oval shapes are the IPC application processes residing in each system. In different colors and numbered from 1 to 7 are the DIFs formed, which are the layers of the RINA architecture. The DIFs numbered 1 - 6 are lower level DIFs (0-level or 0-DIFs), while the larger DIF, numbered 7, is an upper level DIF (1-level or 1-DIF).

We can see that the whole architecture involves only application processes that communicate with each other (IPC).

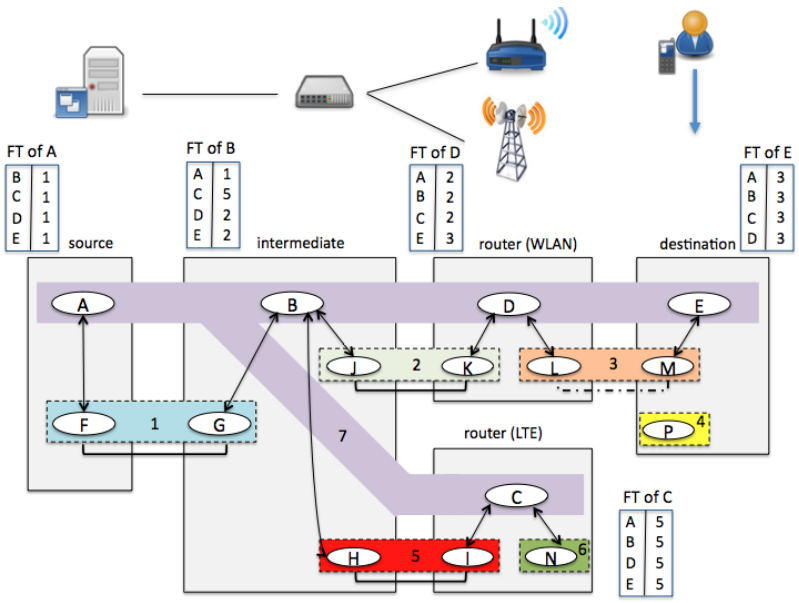

Fig. 2. The mobile terminal is connected to the first AP.

Although it is possible to have multiple layers of DIFs to achieve further configuration through policies, in our example we have chosen to describe a 2-layer RINA architecture for simplicity. The reader should keep in mind that a 0-level DIF for each media AP with a 1-level DIF for networks of a given type, i.e. technology dependent and a 2-level DIF for the technology independent service would be the norm.

The letter in each IPC process denotes its address. Every IPC process has a forwarding table with entries that map addresses of IPC processes to lower level DIFs, so that an IPC process knows where to forward an incoming PDU with a specific destination address. At the lower parts of the 0-DIFs we can see the physical links. The wireless links are denoted with dotted lines, while the wired ones are with continuous lines. As we can see in Fig.2, 0-level DIFs are sitting on top of every physical link, wired or wireless. These 0-level DIFs allow to perform different configurations through policies analogous to the characteristics of each medium. In this way the 0-level DIFs, although configuring different kind of media, provide the 1-level DIF service with certain properties (e.g. loss, delay), so that, transport at the N-DIF can operate more effectively. The 1-DIF can be seen as an "overlay" to which the source, the intermediate router, the Wi-Fi router, the LTE router and the destination have subscribed and it can be used

by the applications of the source and the destination systems to request IPC services. The DIF numbered 4, which consists only of one IPC process, denotes that a second interface is operative in the user's mobile terminal but without a connection to an AP.

The upper DIF maps the source and destination application process names (applications running on the server and the mobile terminal respectively) to IPC process addresses (A and E). For the data transfer between the server and the mobile terminal, the route from A to E is computed as a sequence of intermediate addresses of the 1-level DIF (A, B, D, E). Using its forwarding table, each IPC process has mappings of (N)-DIF addresses to DIFs of the (N-1)-layer, in order to determine the next hop. For example, for a PDU to reach to destination IPC process E, A will forward it to DIF 1, B to DIF 2, etc.

The next figure (Fig. 3) illustrates the changes that happen when the user enters in the range of the second AP and connects to it through an other interface of his terminal. At this moment the mobile node is multi-homed, having the two interfaces on his terminal connected to the two different APs. The IPC processes P and N form a new DIF numbered 4, which is configured according to the characteristics of the wireless link connecting the LTE router and the mobile terminal. A new binding that points to the 0-DIF is added to the forwarding table of IPC process E. This new binding causes changes to the topology of the 1-layer, since the IPC processes A, B and D are now reachable from E through C. These changes are disseminated using routing updates, which trigger re-calculation of the forwarding tables in each IPC process of the 1-layer (DIF 7). Fig. 3 shows the entries of the forwarding tables after the end of this re-calculation process.

Figure 4 describes what happens when the user moves out of the range of the first AP, but remains connected to the second one, completing the hand-off process. The IPC process M departs from DIF 3 and the (N-1)-binding between the processes E and M is dropped. This removal will cause changes to the graph of the 1-DIF, which will be advertised to the rest 1-layer processes through routing updates. In turn, re-calculations of the forwarding tables will take effect to reflect the current bindings.

5 Conclusions, On-going and Future Work

Despite all the work carried out in many years, transport over heterogeneous networks is still a challenge today. In this paper we indicated that the architectural choices in the current Internet architecture lie at the root of the problem. Specifically, we observed two flaws: The first one is that the current Internet architecture does not recognize that it is operating over more than one network. Although, the network and transport layers operate on top of networks with very different characteristics, the functions they provide apply on a global scope. Trying to perform the same function over different physical networks will always lead to inefficiencies, as it allows for a single configuration, which makes it impossible to adapt to the specific requirements each network imposes. A structure that allows different instances of the same functions to operate independently over

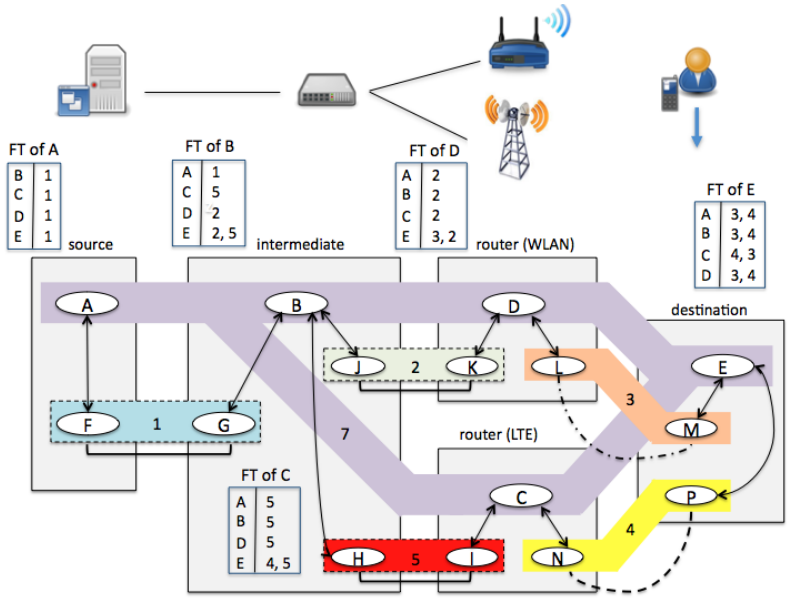

Fig. 3. As the user moves, gets in the range of the second AP and connects to it through the second interface on his terminal (multi-homing).

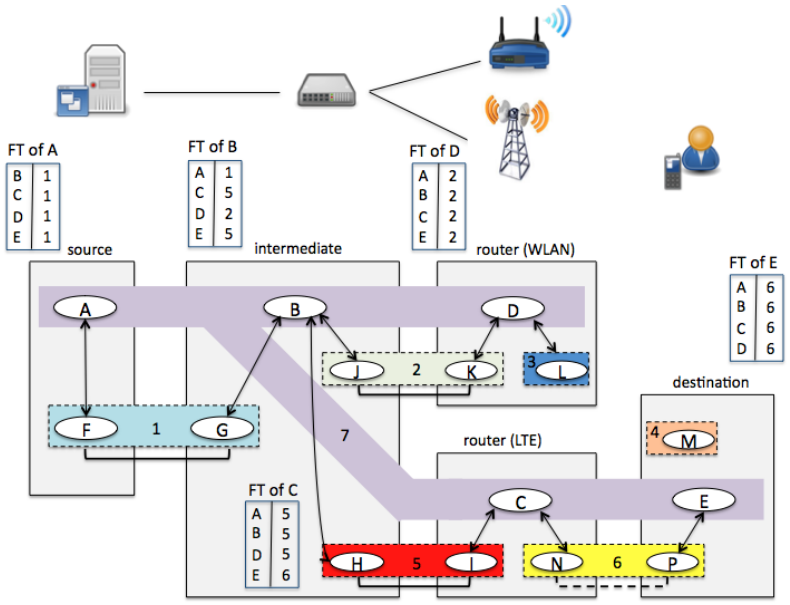

Fig. 4. The user gets out of the range of the first AP and remains connected only through the second one.

different scopes will provide a better framework to achieve transport over hybrid networks. The second flaw we noted is that the current Internet architecture has an incomplete naming and addressing schema, which makes mobility hard, expensive and inefficient. An architecture that provides different names for its basic entities (nodes, PoAs and applications) supports mobility and multi-homing inherently. Following the above observations, we presented RINA, a Recursive Inter-Network Architecture that provides the two features missing from the current architecture, along with many more (support for QoS, multicast, enhanced security). Adopting RINA as the architecture for future networks would make it much easier to support transport over heterogeneous networks.

Currently we are developing a prototype that implements the RINA. Future work includes experimentation with the developed prototype over networks of different physical media and further research, measurements and performance comparisons of RINA versus the current Internet architecture.

References

1. Day, J.: Patterns in Network Architecture: A Return to Fundamentals. (2008)
2. Tsaoussidis, V., Matta I.: Open issues on TCP for mobile computing. In: Wireless Communications and Mobile Computing, vol. 2, Issue 1, 3–20 (2002)
3. Ishakian, V., Matta, I., Akinwumi, J.: On the cost of supporting mobility and multihoming. In: GLOBECOM Workshops, IEEE, pp. 310–314 (2010)
4. Day, J., Matta, I., Mattar, K.: Networking is IPC: A Guiding Principle to a Better Internet. In: Proc. of ReArch08, Madrid, SPAIN (2008)
5. Pentikousis, K.: TCP in wired-cum-wireless environments. In: Communications Surveys and Tutorials, IEEE , vol.3, pp. 2–14 (2000)
6. Nashiry, M.A., Abdullah-Al-Mahmud, M., Rahman, M.M., Anjum, M.N.: Evaluation of TCP performance over mobile IP wired-cum-wireless networks. In: ICCIT, 11th International Conf., pp. 150–155 (2008)
7. Henna, S.: A Throughput Analysis of TCP Variants in Mobile Wireless Networks. In: NGMAST Third International Conf. , pp. 279–284, 15-18 (2009)
8. Balakrishnan, H., Seshan, S., Amir, E., Katz, R.H: Improving TCP/IP Performance over Wireless Networks. In: Proc. of 1st ACM Conf. on Mobile Computing and Networking (1995)
9. Caceres, R., Iftode, L.: The Effects of Mobility on Reliable Transport Protocols. In: Proc. 14th Conf. Distributed Computing Systems, pp. 12–20 (1994)
10. DeSimone, A., Chuah, M.C.,Yue, O.C.: Throughput Performance of Transport-Layer Protocols over Wireless LANs. In: Proc. GLOBECOM (1993)
11. Eom, D.S., Lee, H.S., Sugano, M., Murata, M., Miyahara, H.: Improving TCP handoff performance in Mobile IP based networks. In: Computer Communications, pp. 635–646 (2002)
12. Ben-Othman, J., Nait-Abdesselam, F., Mokdad, L., Ramirez, O.: Performance evaluation of TCP handoffs over mobile IP connections. In: Proc. of the 2008 IEEE/ACS (2008).
13. Daniel, L., Kojo, M.: Adapting TCP for Vertical Handoffs in Wireless Networks. In: Proc. 31st IEEE LCN, pp 151–158 (2006)
14. Caceres, R., Iftode, L.: Improving the performance of reliable transport protocols in mobile computing environments. In: IEEE Journal on In Selected Areas in Communications, Vol. 13, No. 5., pp. 850–857 (1995)