

6LoWPAN conform ITS-Station for non safety-critical services and applications

Giovanni Pellerano, Mariano Falcitelli, Matteo Petracca, Paolo Pagano

CNIT - National Laboratory of Photonic Networks

Via Moruzzi, 1 - 56124 Pisa, ITALY

{giovanni.pellerano, mariano.falcitelli, matteo.petracca, paolo.pagano}@cnit.it

Abstract

This paper presents a novel architecture for supporting WSN applications in the field of Intelligent Transportation Systems. Besides the set of the so called CALM ISO-standards for vehicular environments, a low-cost roadside ITS-Station network is proposed. It is based on the Internet of Things paradigm and its applicability at delivering non safety-critical services and applications are evaluated by means of experimental tests on prototypes. Boards with both specific hardware and tailored software have been designed. Then a Wide Area Network testbed has been instantiated to support trials of representative use cases. The performance analysis shows that the network of SEED-EYE boards driven by a 6LoWPAN compliant ITS-Station software suite matches the application requirements in a broad set of domains. Moreover the advantages of the proposed system architecture over existing solutions based on expensive and general purpose technologies have been discussed. This activity has been approved for standardization purposes as work items at ISO TC 204 Working Group 16.

Keywords

Intelligent Transportation Systems, Internet of Things, Wireless Sensor Network, 6LoWPAN, CoAP, CALM.

I. INTRODUCTION

Intelligent Transportation Systems (ITS) are advanced electronics, information and telecommunication technologies applied in the field of road transport, including infrastructure, vehicles and users. They aim to provide innovative services in traffic and mobility management, as well

as for interfaces with other modes of transport. ITS make it possible to imagine a future in which the drivers of vehicles will be able to foresee and avoid collisions, navigate the quickest route to their destination, making use of up-to-the-minute traffic reports, identify the nearest available parking slot and minimize their carbon emissions, etc. ITS can significantly contribute to a cleaner, safer and more efficient transport system; consequently, ITS have become the focus of several policy and legislative initiatives.

Currently, ITS depend on many components including traditional monitoring sensors such as inductive loops, video cameras, ultrasonic sensors, radars, etc. Regardless of sensors used, most of ITS technologies requires costly infrastructures supporting the installation on site. Furthermore most of the existing sensor network systems requires custom gateways and specific tools. These systems are generally power-hungry, expensive to install, maintain and repair: characteristics that rarely match with the scalability requirements. In fact ITS do not usually cover an entire city, but they tend to be installed only along the major roadways. The objective of an effective traffic monitoring through a pervasive diffusion of ITS is still faraway, until the feasibility for an implementation of low-cost roadside networks will not be shown.

The achievement of a seamless ITS architecture may be accelerated with standardization initiatives in which the different stakeholders in the ITS domain can cooperate. The ISO TC204 WG16 (International Organization for Standardization Technical Committee 204 Working Group 16 - Wide area communications/protocols and interfaces) is defining an ITS communication architecture which is better known as CALM for C-ITS (Communications in Cooperative Intelligent Transport Systems) [1]. This architecture, recently included in the more general picture of the Standards for Cooperative ITS [2], covers all OSI layers, from the physical layer up to the application layer and covers all types of communication scenarios (direct and multihop vehicle-to-vehicle, vehicle-to-infrastructure, vehicle-to-Internet) and scopes (uni-cast, multicast and geocast). The architecture effectively supports any physical communication medium (cellular, long-range; medium-range and short-range communication networks) - to be used simultaneously - and all types of applications (ITS safety applications, ITS non-safety applications, and all legacy Internet applications). The networking layer is designed to support IP (IPv6) and non-IP (FAST) types of communications (Fig. 1).

The adoption of IPv6 in conjunction with High Performance and Pervasive Computing is enabling for large scale (potentially EU wide) ITS [3]. An open issue in the CALM context

regards the way Wireless Sensor Network (WSN) can be interconnected with ITS. WSNs are based on a technology which is becoming more mature than in the past, gaining momentum as one of the enabling tools for the IoT (Internet of Things). According to IoT paradigm, Internet is being applied ubiquitously and, there is no reason why the ITS communication network would not be part of the overall Internet [4]. WSNs consist of medium to large networks of inexpensive wireless sensor nodes capable of sensing, processing and distributing information acquired from the environment through the collaborative effort of nodes [5]. They provide significant advantages both in cost as well as in distributed intelligence. On the one hand, installation and maintenance expenses are reduced because of the use of cheap devices which do not require wiring. Furthermore, distributed intelligence enables the development of diverse real-time traffic applications not feasible with centralized solutions and it would allow the interoperability of the ITS communication system with other communication systems such as for example health-care and emergencies.

The natural location for the WSNs inside ITS is the roadside eco-system instead of vehicular environment, because the commonly used IEEE 802.15.4 MAC protocol has still several problems with mobile nodes related to association with coordinators and synchronization [6]. Roadside networks, compliant with CALM, are expected to interact with local access networks to send the information retrieved from sensors to other ITS-Stations. Thus, this scenario would greatly benefit from a smooth integration between WSN and ITS. Nevertheless, the current ISO solution misses an effective fully interoperability between ITS-Stations and WSNs, since the issue is addressed using an adaptation gateway similar to those concerning legacy systems and proprietary devices.

In this paper the feasibility for effective implementation of low-cost roadside WSNs is evaluated as an extension of the actual CALM ITS-Station. The aim is to show that WSNs, driven by a 6LoWPAN [7] compliant ITS-Station software suite, match the application requirements for non safety-critical services and applications.

The paper is organized as follows. In Section II research issues related to WSN communication protocols and IPv6 applicability to ITS are introduced. Some details of the IP connected WSN enabling technologies such as 6LoWPAN and CoAP are provided in Section III. This background will help to understand the architecture proposed in Section IV for large scale roadside networks, so to better clarify its novel characteristics. The features of the 6LoWPAN roadside network are discussed in Section V. The instantiated WAN testbed is shown in Section VI, together with a

performance evaluation. Finally, the concluding remarks are given in Section VII.

II. RELATED WORKS

The design of WSN for ITS, considering not only independent applications, but also their position in heterogeneous systems has been recently surveyed by [5]. The concept and the simulation of an integrated network of roadside sensors and vehicles for driving safety are presented in [8]. As for IPv6 applicability to ITS, standards already consider IPv6 as the new Internet Protocol for ITS. Regarding this, a recently EU funded project called ITSSv6 is developing an open-source IPv6 ITS Station stack freely available to European and national third parties (projects, industry and academia) using input from previous works. As reported by the authors [9], the communication stack - based on IEEE 802.11p - operates correctly, maintaining the in-vehicle network connectivity in all tests and showing performance results that enable the communication stack to be used in many vehicular services. Unless high-quality multimedia transmissions are required, the bandwidth results indicate that the data rate required by most of the traffic efficiency and comfort services can be covered, and, according to latency tests, even non-critical security services, which are not highly dependent on real-time response, could be implemented, such as emergency assistance, variable traffic signaling, etc.

Many technological issues related to the application of WSNs to ITS are covered by the previous papers, including the selection of the suitable communication protocol for the communications among vehicles, as well as in the vehicle to infrastructure scenarios. However the issue of the interoperability between 6LoWPAN and external IPv6 networks is not tackled in any of them.

At now, nobody has tried a specific realization of 6LoWPAN in the ITS context, except for the work presented in [10], where existing IoT routers are incorporated with any ITS station in a coupled architecture. Leveraging the standardization efforts to deploy generic ITS stations, this paper presents an early implementation of some ITS facilities and applications in 6LoWPAN devices.

III. BACKGROUND

In this section some details of 6LoWPAN and CoAP standard protocols for IP connected WSNs are provided. This background is needed to better clarify what was already available when the here proposed solution for large scale ITS roadside networks has been implemented.

A. *6LoWPAN/CoAP Protocol Stack*

The 6LoWPAN protocol was created to facilitate IPv6 communication by embedded Internet devices, using open standards able to scale across large network infrastructures. On one hand, IPv6 can guarantee auto-configuration and stateless operation of networks where every smart object can be connected readily to other peers of the Internet, without the need for intermediate entities like translation gateways or proxies. On the other hand, the limited packet size and other constraints of Low-Power Wireless Personal Area Networks need an adaptation layer for IPv6 to perform header compression, fragmentation and address auto-configuration. 6LoWPAN allows communication via IPv6 address of a sensor node with an adaptation layer above the 802.15.4 link layer for the fragmentation and reassembly of packets. This new standard allows the use of IPv6 networks in Low Power and Lossy Networks (LLN) such as IEEE 802.15.4, which is characterized by short range, low bit rate (250 kbps maximum at the physical layer), high fragmentation, (maximum size of data frame is 127 bytes), low memory usage and low cost. The physical layer of IEEE 802.15.4 specifies how the devices communicate on using one of the 27 channels available, located in different frequency bands. The IPv6 standard defines certain requirements for the link-layers over which it is to be transported. However, the IEEE 802.15.4 MAC layer does not fulfill these requirements in certain points. Hence, the 6LoWPAN specification defines not only the frame format for the transmission of IPv6 packets over IEEE 802.15.4, but also some methods to compress and decompress IPv6 packets. The advantages from utilizing 6LoWPAN includes flattening the naming and addressing hierarchy in addition to simplifying the connectivity model. This means that gateways are not required to translate between proprietary protocols and standard IP, but they can be replaced with much simpler bridges and routers. 6LoWPAN also provides support for stateless address auto-configuration, which means that a host can generate its own addresses using a combination of locally available information and information advertised by routers, without making stateful binding with routers. This represent one of the first interesting compression applied by 6LoWPAN to IPv6 packets; 6LoWPAN defines a compression algorithm that permits to obtain a unique IPv6 address from either, the 16-bit or the 64-bit IEEE 802.15.4 MAC addresses [11]; this algorithm relies on the assumption that a PAN (Personal Area Network) maps to a specific IPv6 link. Even though all 802.15.4 devices have an EUI-64 address, the 16-bit short addresses can be used also for address

auto-configuration. In this case, a pseudo 48-bit address is formed by concatenating 16 zero bits to the 16-bit PAN ID, the resulting 32 bits are concatenated with the 16-bit short address. The interface identifier is formed from this 48-bit address as defined in the "IPv6 over Ethernet" specification [12]. For example the IPv6 link-local address for an IEEE 802.15.4 interface is formed by appending the interface identifier to the prefix FE80::/64.

Another important component of the protocol stack is CoAP, a specialized Web transfer protocol optimized for resource constrained networks typical of IoT and M2M (Machine to Machine) applications. CoAP is similar to HTTP but its goal is not to simply compress HTTP, but to implement a subset of REST [13] operations optimized for M2M interactions and also to include constraints such as statelessness, cache-ability, layered system, uniform interface common in current web protocols. The interaction model is similar to the client/server model of the HTTP protocol. Clients request an action to a resource; then, the server sends a response with a response code, which may include a resource representation. Messages are exchanged asynchronously over the UDP datagram-oriented transport. Unlike HTTP based protocols, CoAP operates over UDP instead of TCP, since the latter is not appropriate for controlling data flow transmission inside a LLN and its overhead is considered too high for short-lived transactions. In addition, TCP does not have multicast support, unlike UDP, and it is rather sensitive to mobility. CoAP operates over UDP and therefore has significantly lower overhead. It employs a simple retransmission mechanism instead of using complex congestion control as used in standard TCP and uses a unique Transaction ID to identify each GET request for retransmission purposes to keep reliability. In addition, the use of UDP enable CoAP at delivering multicast support. CoAP includes many characteristics among the main ones of HTTP: for example URI based resource representation (e.g., `coap://its-s.com/traffic-light`) and support for different payload content types. In addition with respect to HTTP, CoAP supports:

- transmission of large amount of data by splitting the data into blocks called Blockwise Transfer;
- a Resource Observe mechanism built using a publish/subscribe pattern;
- some resource discovery capabilities to allow clients to discover new resources handled by servers.

IV. SYSTEM OVERVIEW

In this section the 6LoWPAN ITS-Station architecture is presented. After a brief introduction to the generic ITS-S architecture, specific characteristics of the CALM Roadside Network are outlined along with a discussion with the open challenges. The focus is finally given to our proposal of mapping of a 6LoWPAN compliant stack onto the CALM reference architecture along with description of typical application use cases.

Fig. 2 shows a general abstraction of the architecture of the CALM ITS-Station (on the left), blowing up the details of the stack modules that are modified in order to accommodate the low cost ITS-Station here proposed (on the right).

The architecture partially follows the principles of the ISO Open Systems Interconnection (OSI) model. It consists of six main parts: in the data plane, it has four layers (three of those corresponding to the OSI 7-layer stack) that perform different tasks; from the bottom to the top: Access (OSI Physical and Data Link), Network & Transport (the same as OSI), Facilities (OSI Session, Presentation and Application) and Application layers. The neighbor layers are connected to each other via a Service Access Point (SAP). In addition to the vertical-layer stack, a Management entity and a Security entity are connected to all the layers via the SAPs in a cross-layer fashion. This design has been originally introduced by ISO TC204 WG 16 and is now clearly represented on the architecture diagram. The cross-layer functions to be offered are ITS layer-links under discussion at the standardization levels.

To obtain seamless design and taking into account the possibility offered by the 6LoWPAN protocol stack, the proposed adjustments of the CALM ITS-Station are suggested as depicted on the right of Fig. 2. In particular the following modules are integrated:

- Network & Transport level: 6LoWPAN and RPL are proposed at Network & Transport level as they permit a seamless IP integration between ITS-Station defined by the actual CALM standard and WSNs based on the IEEE 802.15.4 Access Medium;
- Facility level: the Constrained Application Protocol (CoAP) defined by the IETF CoRE Working Group [14] is adopted; it is an OSI application layer protocol designed to provide a REST-like interface [13], but with a lower cost in terms of bandwidth and implementation complexity than HTTP-based REST interfaces.

Both the protocols are envisaged to export their parameters as SAPs to the Management and

Security cross-layers.

V. THE 6LOWPAN ITS-STATION ROADSIDE NETWORK

In Fig. 3 the conception of a typical ITS-Station Roadside Network for 6LoWPAN devices is shown, together with common use cases. All the specific components of the CALM ITS-Station are implemented, and also the interconnection results conform to the standard:

- specific WSN-based ITS-Station nodes could be implemented for proprietary devices and data bus (up-to layer OSI 7) and could be interconnected to legacy roadside equipment (e.g., line/speed sensors, traffic light and variable message signs);
- a large number of router-only nodes could be widely deployed, due to the low-cost and small-size nature of the nodes, creating a routing topology suited for the application;
- border router nodes could be added to the network enabling a remote ITS Central Station to direct manage and reconfigure specific nodes functionality by uniquely reach it by using global IPv6 addressing.

All the building blocks at Access and Network & Transport layers (referring to CALM model) are common to the nodes. This particular design choice allows to define a P2P mesh topology where all the involved objects in the network participate in the network routing and thus adheres to the CALM ITS-Station Router functionality.

1) The Hardware prototypes: The hardware used for the WSN nodes was designed within the IPERMOB project [15], thanks to which a prototype of such infrastructure has been deployed at the Pisa International Airport. The SEED-EYE board [16] is an advanced WSN node specifically thought for ITS applications [17]. This device is equipped with an 80 MHz Pic32 micro-controller with built-in 128 KB of RAM and 512 KB of ROM. It implements in hardware IrDA, SPI, I2C, UART, USB, and CAN communication protocols easing the connection with external units; the operative voltage of the chip ranges from 2.3V to 3.6V and some power sleeping modes (RUN, IDLE, and SLEEP modes) are allowed, along with multiple switchable clock modes useful for the development of power saving policies. From the point of view of the network layer and radio communications, SEED-EYE offers a platform based on IEEE 802.15.4: the Microchip MRF24J40B transceiver; this transceiver is IEEE 802.15.4 compliant and operates in the 2.4 GHz to 2.4835 GHz ISM unlicensed band. The on-board MRF24J40B radio transceiver along with its internal power amplifier (20dBm) enables the SEED-EYE for long range communication (up

to 100 meters in open space at maximum power). The MRF24j40B also provides a Turbo mode to transmit and receive at 625 kbps (2.5 times larger than nominal bit rate in IEEE 802.15.4 networks) enabling for higher rate proprietary protocols.

2) *The SW prototype*: A peculiar all-encompassing software suite was implemented. Many aspects of the system are involved, they go from the Operating System (OS) of SEED-EYE board to the ITS-Station Application layer [18]. As first step, the Contiki OS [19] has been ported to the micro-controller integrated on the SEED-EYE board. Contiki uses protothreads [20], a programming abstraction for event-driven sensor network OS. By protothreads, programs can perform conditional blocking on top of event-triggered systems with run-to-completion semantics, without the overhead of full multi-threading. Therefore, protothreads simplify implementation of high-level functionality on top of event-driven systems, without significantly increasing the memory requirements. Moreover, the IEEE 802.15.4 standard was implemented on Contiki OS in such a way to supports, at MAC level, the CSMA (Carrier sense multiple access) protocol and a NULL RDC (Radio Duty Cycling) module, which effectively keeps the radio always on. At Network & Transport level, nodes implement 6LoWPAN with its most recent scheme for header compression available [21]. The network stack has been configured with UDP and ICMPv6 enabled, while routing has been handled using the Routing Protocol for LLN (RPL). RPL (as defined in [22]) can be considered as the state-of-the-art routing algorithm developed by the WSN community. RPL has been proposed by the IETF Routing over LLN Working Group (ROLL) as a standard routing protocol for 6LoWPAN, since existing routing protocols do not satisfy all the requirements for LLN.

Additional SW modules were implemented on the specialized nodes to deploy all the needed entities of the CALM Roadside Network. CALM ITS-S Border Router functionality has been developed using a Serial Line IP (SLIP) interface [23], a very simple protocol that frames IP datagrams in order to be sent over serial connections. SLIP interface has been implemented developing an interrupt-driven UART module in the MCU; a UART port has been used to interconnect the SEED-EYE board with a Linux PC acting as IPv6 route forwarder. A border router application for Contiki OS has been customized to address some performance issues. In particular a global IPv6 address resolution has been deployed and tested using SixXS (an IPv6 tunnel broker) and the related software AICCU [24] to build an IPv6 tunnel over an IPv4 network. This has been done to provide a temporary global IPv6 addressability of the PCs

adopted in the trials, in order to emulate a Wide Area Network (WAN) through a remote PC acting as a CALM ITS Central Station. Specifically the Border Router was enabled to distribute the 2001:1418:100:823c::/64 global subnet over the WPAN.

Regarding the CALM ITS-Station Host, some specific functions have been implemented by means of virtualization. For example, traffic light devices have been emulated using simple LEDs, light switch by using on-board buttons and sensors/actuators by using ADC converters. The specific resources of the CALM ITS-Station Host are exported through a CoAP server (e.g. ITS-Station Hosts export analog switch, Received Signal Strength Indicator (RSSI) sensor and some timing information, permitting the monitoring of the operation parameters); The selected Contiki CoAP library used is described in [25].

For evaluating the capabilities of the prototypes in terms of resource efficiency, the Flash memory and the RAM occupancy of the firmware was checked. The larger set of implemented functions produced the maximum percentage of occupancy of 15 % and 42% for Flash memory and RAM respectively.

VI. INSTANTIATED WAN TESTBED

In order to give a characterization of the system and evaluate its performance at delivering non safety-critical applications, a specific testbed involving a WAN connection between two peers was instantiated. The system architecture is formed by two ITS-Station Sub-Systems interconnected through a WAN: a Roadside Network and a Central Station. The roadside network, shown in Fig. 4, was implemented with reference to the CALM Architecture. It represents a typical ITS-Station providing non safety-critical services (i.e. exchanges of simple infotainment messages through CoAP Resource Observe pattern). In particular, it is composed by the following three entities: an ITS-Station Host acting as CoAP server, an ITS-Station device acting as simple ITS-Station Router; an ITS-Station Border Router providing WAN connectivity and global addressability to the entire ITS-Station network. As previously stated in Section V, all the instantiated nodes act as ITS-Station Routers by design choice. The nodes have been deployed in a regular line topology at 10 mt distance one from the other. The transmission power has been set to the minimum that allows the neighbor nodes to communicate among them. To test the proposed architecture and to evaluate the behavior of the components over a Wide Area Network (WAN), a simple Central Station was also configured. It is composed as follows: a generic PC

with Linux Ubuntu OS as base system; the same AICCU software just used to develop the WAN IPv6 interconnection through a SixXS tunnel broker and here used for the same purpose; traditional IP tools for network diagnostics such as ping6 and traceroute6; a Firefox browser enhanced with the Copper plugin [26] to test CoAP facilities and ITS-Station Host functions.

In order to verify the correctness of the implementation and to analyze the capabilities of the system, two kinds of tests have been performed: functional and performance tests.

A. Functional tests

First, the matching of the basic functional requirements of the stack have been verified. Simple ITS-Station Router nodes have been used to build several network topologies and they have been connected to the Hosts using the RPL protocol. Both physical (up to ten) and virtual WSN nodes have been mixed to evaluate the event handling capability of the system, even in large-scale IoT Networks. All nodes have been enabled for IPv6 global networking and addressing by using a node acting as ITS-Station Border Router. An ITS Remote Central Station has been simulated using simple PCs interconnected to an IPv6 tunnel broker and executing some queries to other nodes using CoAP protocol. ITS-Station Hosts have been connected to simple devices (e.g., to a power switch using an ADC) and device functionality has been exported to the network as CoAP resources. Fig. 5 illustrates two examples of requests executed using the Copper Firefox plugin. Tests are based on a ordinary traffic scenario (up to 3000 veh/hour/lane at speeds from 5 to 130 km/h), as the goal is to show the functionalities of the station for non safety-critical applications.

- A Line Sensor notifies the passage of vehicles through an Observable Resource.
- A Client Side Application fills a velocity histogram of the passing vehicles by correlating timing and speed information coming from two Line Sensors [27], [28].

B. Performance tests

Goal of the present architecture is to provide non safety-critical services to drivers. In order to verify that the system is capable to deliver such services, network response in terms of Round Trip Time (RTT) and Packet Loss (PLOSS) metrics was measured. The former is the time it takes for a signal to be sent, plus the time delay of it takes for an acknowledgment of that signal to be received. RTT therefore consists of the sum of transmission times between

two peers of the network. In the context of computer networks, the signal is generally a data packet, and the RTT is also known as the ping time. On the other side PLOSS occurs when one or more data packets traveling across a computer network fail to reach their destination. In case of WSN, the packet loss can be produced at physical and MAC layers, because of the limited capacity for buffering packets, since the transceiver resources and main memory can accommodate few packets at a time. In Fig. 6 RTT and PLOSS are shown according to an increasing ICMP (Internet Control Message Protocol) payload size. The graphic shows an increasing packet loss starting from 74 bytes of ICMP payload. This behavior is caused by IP fragmentation mechanism and is well known in literature; despite this problem, solved by CoAP protocol through the Blockwise Transfer mechanism, the graphic clearly demonstrates that the implemented 6LoWPAN networking stack is working fine and that the experimental values of RTT and PLOSS are small enough to match at least non safety critical applications (e.g information messages exchanges and other comfort services).

VII. CONCLUSION

In this paper a novel architecture for supporting WSN applications in the field of ITS has been presented. The followed approach consisted in integrating IoT standards with a wireless low-cost roadside network compliant with 6LoWPAN. In particular, a new node has been designed and realized together with the needed software suite to implement a working prototype of ITS-Station. Also, how such a network could be connected in a seamless fashion to ITS networks, as defined by the CALM standard, has been shown. As preliminary result, the Contiki porting to a 32 bit micro-controller-based platform along with a 6LoWPAN/CoAP stack has been obtained; the performance analysis has shown that system matches the application requirements in a broad set of domains. At least for non safety-critical applications and services, the proposed system architecture can compete with existing solutions based on expensive general purpose technologies.

After the successful functional verification of the integrated technology hereby proposed, several aspects of the performance need to be further evaluated. It could be interesting to define the design of a new abstraction layer on top of CoAP and permitting the on-line ITS-Station reprogrammability, in the context of general purpose cooperative ITS applications. The implementation of such a service will face new issues; in fact, connecting transportation systems,

electricity grids and other critical infrastructures poses new challenges to security, safety and end-user privacy. CALM architecture, along with its two management and security planes, perfectly fits with these design requirements, leaving space for the deepening of the specification of the standard driven by next studies. Further, focusing on cross layer design, the development of new cross-layer services (not envisaged by the canonical OSI stack) and optimization techniques are allowed. This enhancement can involve new research topics such as context-aware and application-aware optimizations. Meantime, the results achieved in the present paper have been approved for standardization purposes as work items ISO 19079 (6LoWPAN networking), ISO 19080 (CoAP facility) at ISO TC 204 Working Group 16.

REFERENCES

- [1] “Communications in Cooperative Intelligent Transport Systems - CALM for C-ITS,” <http://calm.its-standards.info>.
- [2] *Standards for Cooperative ITS - Release 1*, CEN/ISO Proposal Release 1, 2013. [Online]. Available: http://its-standards.eu/Feeds/Release1_CEN_ISO_ProposalFromExperts.pdf
- [3] P. Pagano, M. Petracca, D. Alessandrelli and C. Salvadori, “Is ICT mature for an EU-wide intelligent transport system?” *Intelligent Transport Systems, IET*, vol. 7, no. 1, pp. 151 – 159, 2013.
- [4] A. Azzarà, D. Alessandrelli, S. Bocchino, M. Petracca and P. Pagano, “Architecture, functional requirements, and early implementation of an instrumentation grid for the IoT,” in *Proceedings of the 14th IEEE International Conference on High Performance Computing and Communications (HPCC)*, Liverpool, UK, Jun. 2012, pp. 320–327.
- [5] F. Losilla, A.-J. Garcia-Sanchez, F. Garcia-Sanchez, J. Garcia-Haro, and Z. Haas, “A Comprehensive Approach to WSN-Based ITS Applications: A Survey,” *Sensors*, vol. 11, pp. 10 220–10 265, 2011.
- [6] K. Zen, D. Habibi, A. Rassau and I. Ahmad, “Performance Evaluation of IEEE 802.15.4 for Mobile Sensor Networks,” in *Proceedings of the 5th IFIP International Conference on Wireless and Optical Communications Networks*, Surabaya, Indonesia, May 2008, pp. 1–5.
- [7] J. Hui and P. Thubert, *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*, RFC 6282 (Proposed Standard), Internet Engineering Task Force (IETF) Std., Sep. 2011. [Online]. Available: <http://www.ietf.org/rfc/rfc6282.txt>
- [8] H. Qin, Z. Li, Y. Wang, X. Lu, G. Wang and W. Zhang, “An integrated network of roadside sensors and vehicles for driving safety: Concept, design and experiments,” in *Proc. of the 2010 IEEE Int. Conf. on Pervasive Computing and Comm. tions (PerCom)*, Germany, pp. 79–87.
- [9] P. J. F. Ruiz, F. B. Hidalgo, J. S. Lozano, and A. F. Skarmeta, *Deploying ITS Scenarios Providing Security and Mobility Services Based on IEEE 802.11p Technology*. InTech, Feb. 2013.
- [10] L. Virág, J. Kovács and A. Edelmayer, “Extension of the its station architecture to low-power pervasive sensor networks,” in *Proceedings of the IEEE 27th International Conference on Advanced Information Networking and Applications Workshops (WAINA2013)*, Barcelona, Catalonia, Spain, Mar. 2013, pp. 1386–1391.
- [11] S. Thomson, T. Narten, and T. Jinmei, “IPv6 Stateless Address Autoconfiguration,” RFC 4862 (Draft Standard), Internet Engineering Task Force, Sep. 2007. [Online]. Available: <http://www.ietf.org/rfc/rfc4862.txt>

- [12] M. Crawford, "Transmission of IPv6 Packets over Ethernet Networks," RFC 2464 (Proposed Standard), Internet Engineering Task Force, Dec. 1998, updated by RFC 6085. [Online]. Available: <http://www.ietf.org/rfc/rfc2464.txt>
- [13] R.T. Fielding and R.N. Taylor, "Principled Design of the Modern Web Architecture," *ACM Transactions on Internet Technology*, vol. 2, no. 2, pp. 115 – 150, 2002.
- [14] Z. Shelby, K. Hartke and C. Bormann, *Constrained Application Protocol (CoAP)*, draft-ietf-core-coap-18, Internet Engineering Task Force (IETF) Std., Jun. 2013. [Online]. Available: <http://tools.ietf.org/html/draft-ietf-core-coap-18>
- [15] "The IPERMOB project," <http://www.ipermob.org>.
- [16] "SEED-EYE: an advanced multimedia Wireless Sensor Network node for ITS applications," <http://rtn.sssup.it/index.php/hardware/seed-eye>.
- [17] D. Alessandrelli, A. Azzarà, M. Petracca, C. Nastasi, and P. Pagano, "ScanTraffic: Smart Camera Network for Traffic Information Collection," in *Proc. of Europ. Conf. on WSN*, 2012, pp. 196–211.
- [18] G. Pellerano, "Master's thesis," 2012. [Online]. Available: <http://rtn.sssup.it/images/theses/doc/tesipellerano.pdf>
- [19] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki-a lightweight and flexible operating system for tiny networked sensors," in *Local Computer Networks. 29th Annual IEEE Int. Conf. on. IEEE*, 2004, pp. 455–462.
- [20] A. Dunkels, O. Schmidt, T. Voigt, and M. Ali, "Protothreads: Simplifying event-driven programming of memory-constrained embedded systems," in *Proceedings of the 4th international conference on Embedded networked sensor systems. Acm*, 2006, pp. 29–42.
- [21] M. Durvy, J. Abeillé, P. Wetterwald, C. O'Flynn, B. Leverett, E. Gnoske, M. Vidales, G. Mulligan, N. Tsiftes, N. Finne *et al.*, "Making sensor networks IPv6 ready," in *Proceedings of the 6th ACM conference on Embedded network sensor systems. ACM*, 2008, pp. 421–422.
- [22] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550 (Proposed Standard), Internet Engineering Task Force, Mar. 2012. [Online]. Available: <http://www.ietf.org/rfc/rfc6550.txt>
- [23] J. Romkey, "Nonstandard for transmission of IP datagrams over serial lines: SLIP," RFC 1055 (INTERNET STANDARD), Internet Engineering Task Force, Jun. 1988. [Online]. Available: <http://www.ietf.org/rfc/rfc1055.txt>
- [24] "AICCU - Automatic IPv6 Connectivity Client Utility," <http://www.sixxs.net/tools/aiccu/>.
- [25] M. Kovatsch, S. Duquennoy, and A. Dunkels, "A low-power coap for contiki," in *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on. IEEE*, 2011, pp. 855–860.
- [26] M. Kovatsch, "Demo abstract: Human-coap interaction with copper," in *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on. IEEE*, 2011, pp. 1–2.
- [27] M. Chitnis, C. Salvadori, M. Petracca, G. Lipari, and P. Pagano, "Traffic related observations by line sensing techniques," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '10. New York, NY, USA: ACM, 2010, pp. 373–374. [Online]. Available: <http://doi.acm.org/10.1145/1869983.1870029>
- [28] M. Chitnis, C. Salvadori, M. Petracca, P. Pagano, G. Lipari, and L. Saltinelli, "Distributed Visual Surveillance with Resource Constrained Embedded Systems." in *Visual Information Processing in Wireless Sensor Networks: Technology, Trends and Applications. L. Ang, & K. Seng (Eds.)*, 2012, pp. 272–272.

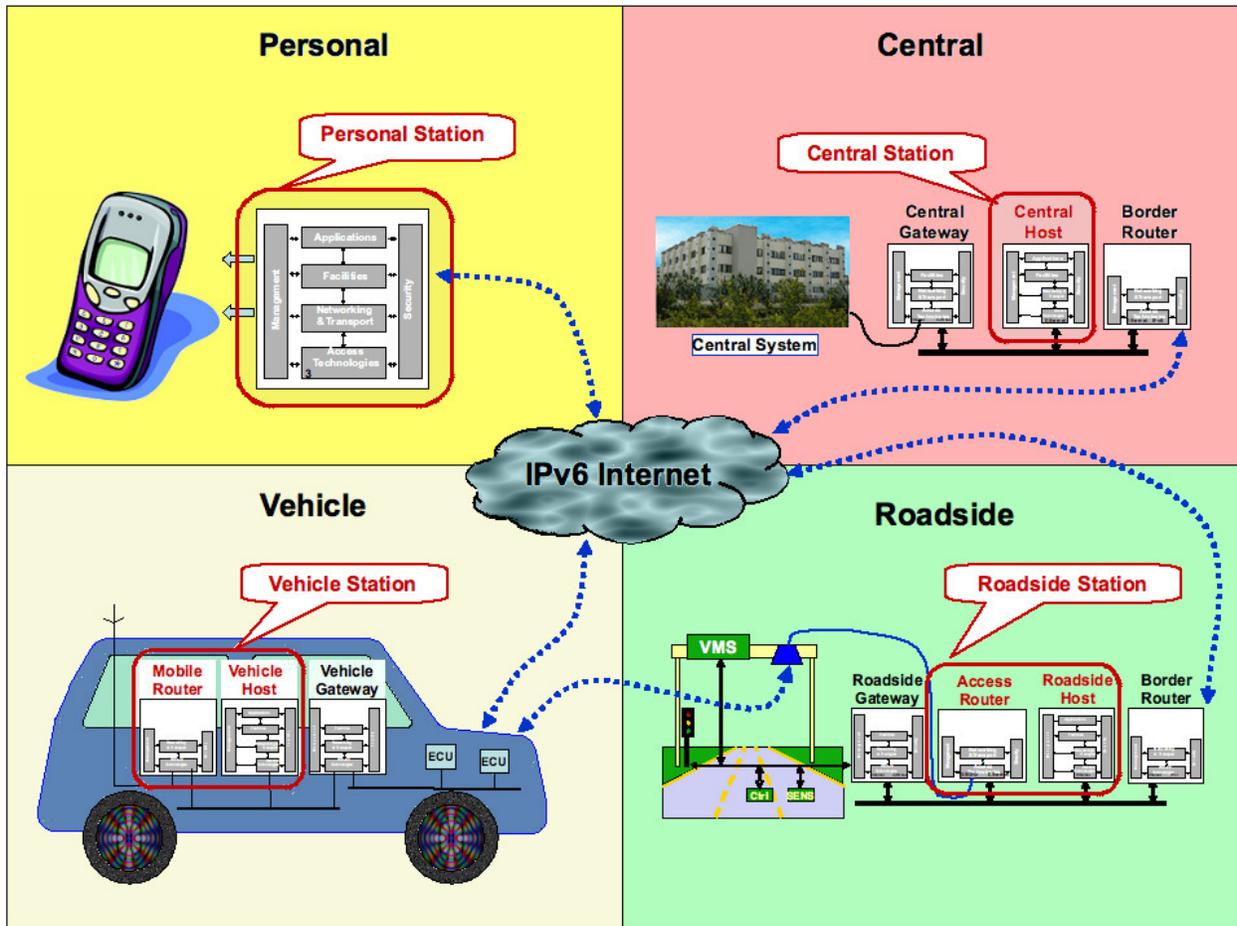


Fig. 1. Overview of the Internet-based European ITS Architecture of Communication.

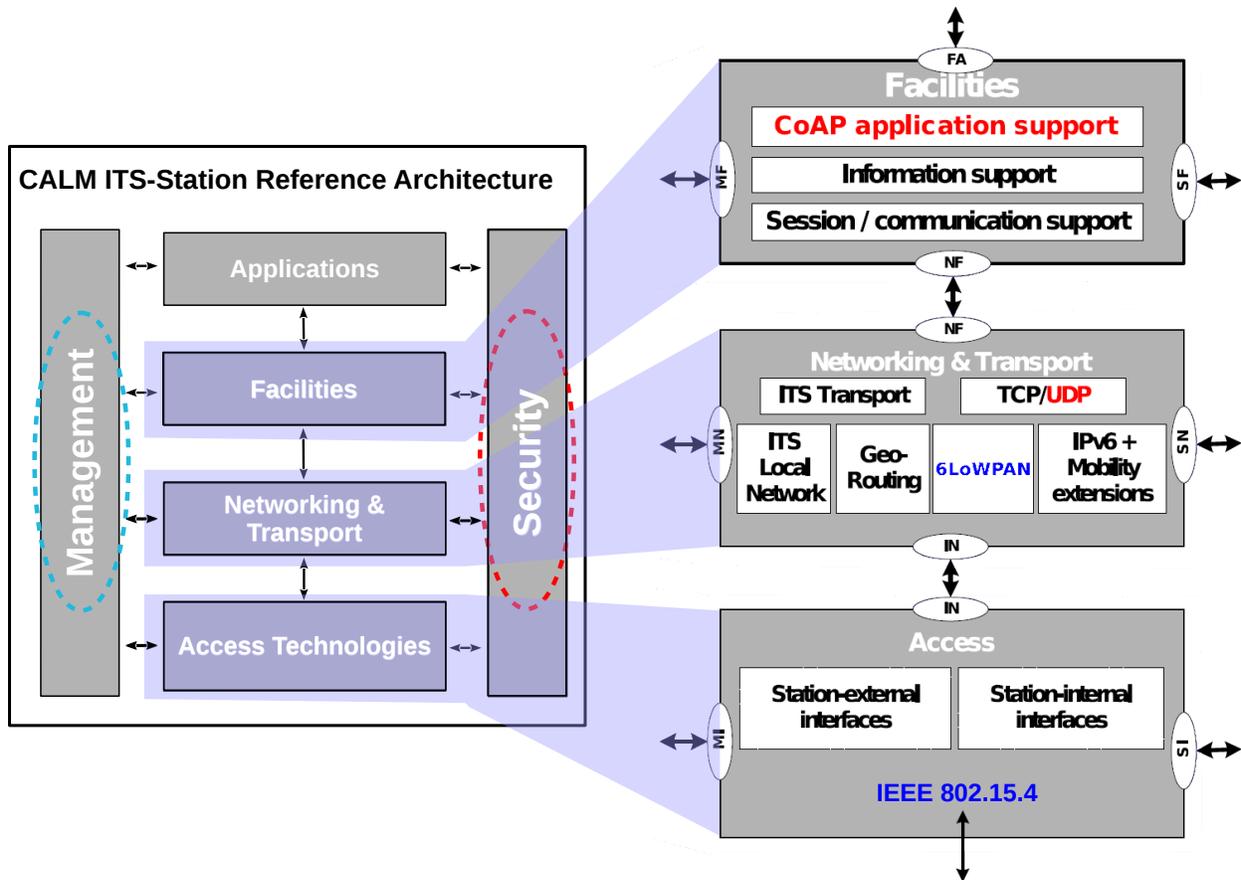


Fig. 2. Proposed architecture of the 6LoWPAN ITS-Station.

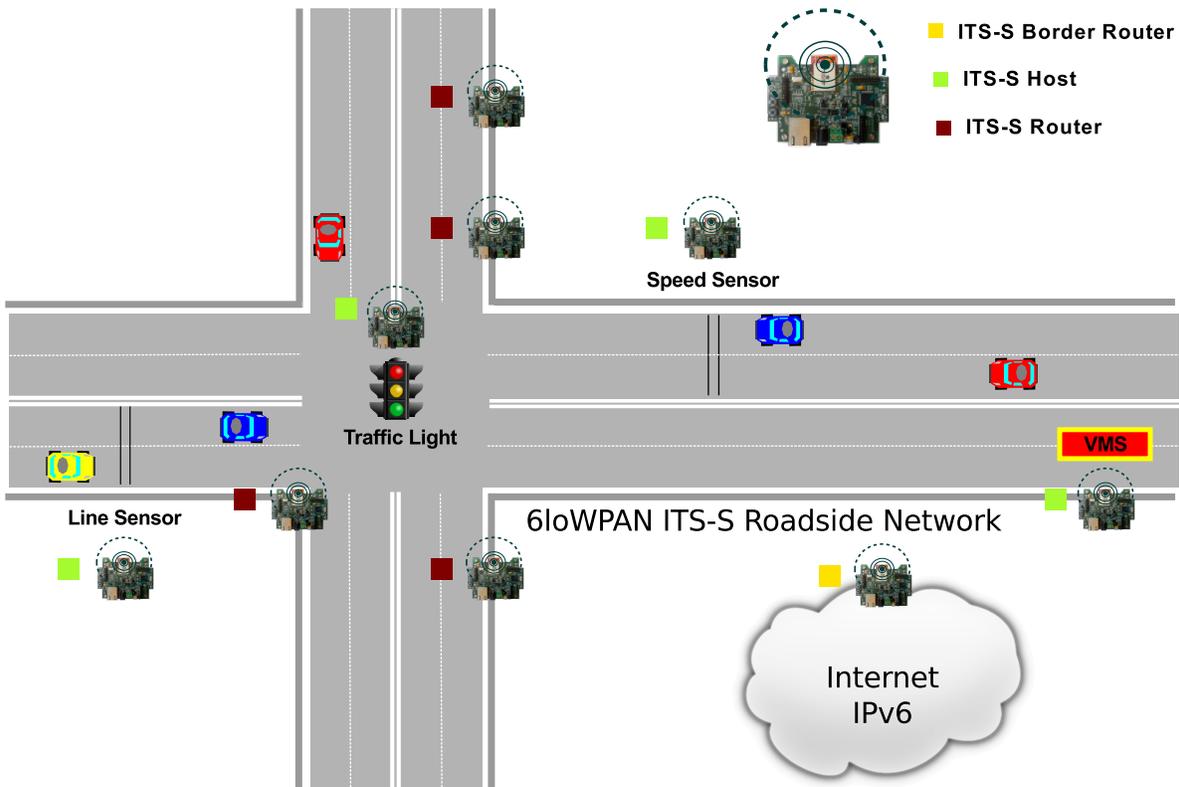
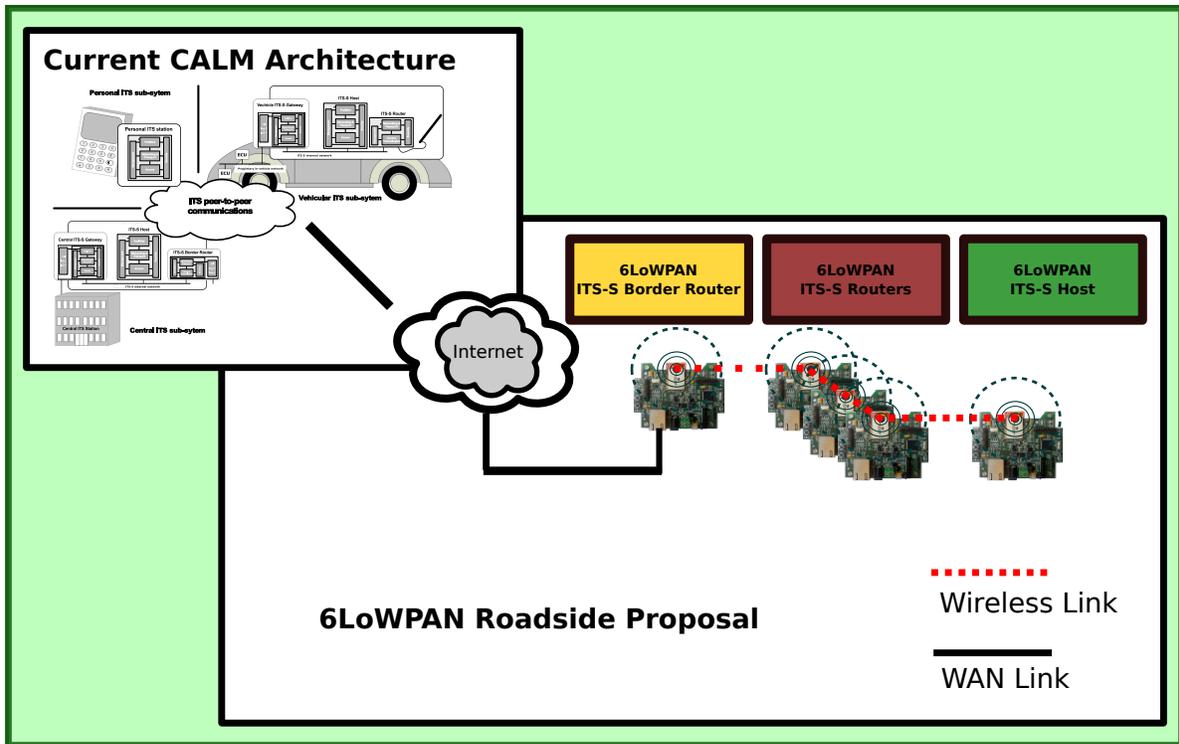


Fig. 3. 6LoWPAN ITS-Station Roadside Network: integration with ISO CALM ITS (up), deployment of common use cases (bottom).

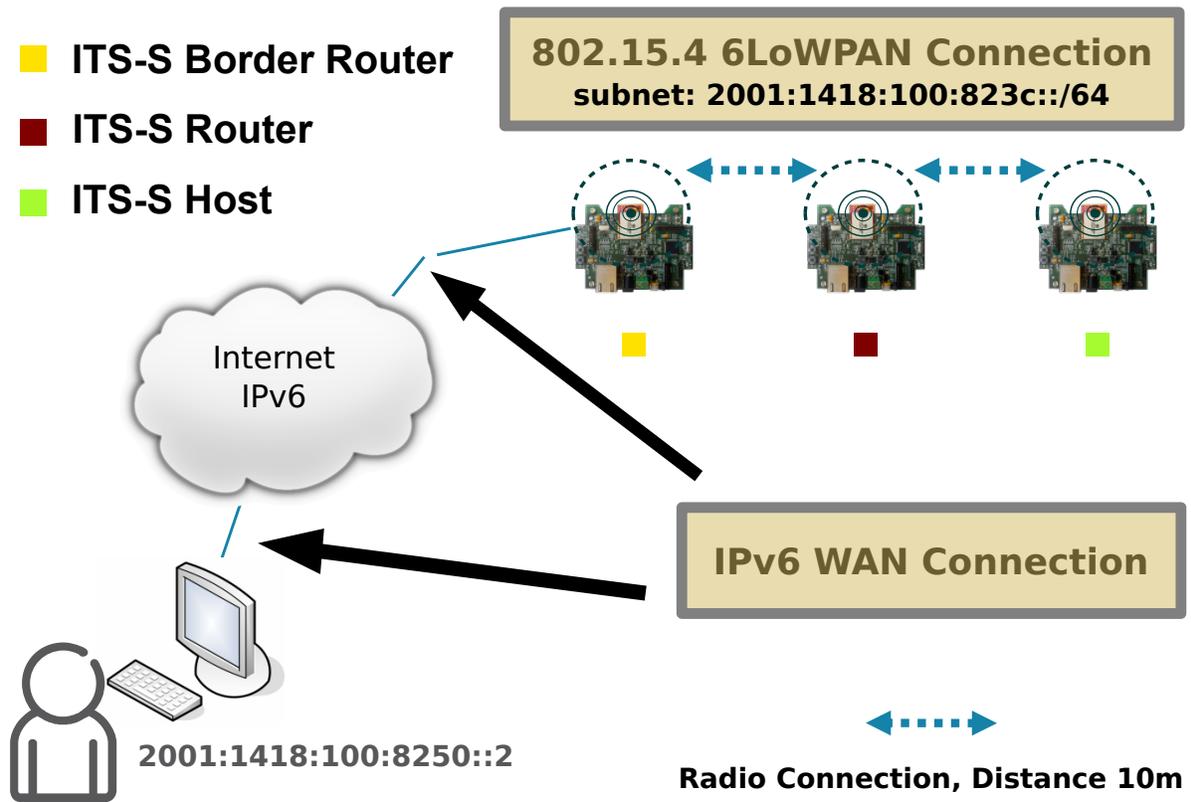


Fig. 4. Instantiated Testbed for the 6LoWPAN ITS-Station.

coap://[2001:1418:823c:205:e257:22b8:6d74]:5683/sensor/line - Mozilla Firefox

File Edit View History Bookmarks Tools Help

coap://[2001:1418:823c:205:e257:22b8:6d74]:5683/sensor/line

Discover GET POST PUT DELETE Cancel Payload Text Behavior CoAP 08

[2001:1418:100:823c:205:e257:22b8:6d74]:5683

2.05 Content (Observing)

Debug options

Accept

[/.well-known/core](#) [/actuators/leds](#) [/actuators/toggle](#) [/sensor/radio](#) **[/sensor/line](#)** [/test/chunks](#) [/test/path](#) [/test/separate](#)

| ... | Value | Option | Value | Info |
|---------|----------------|--------------|-----------|-----------|
| Type | Acknowledgment | Content-Type | text/html | 0 |
| Code | 2.05 Content | Observe | | 0 byte(s) |
| Tran... | 33862 | Token | 0x66BE | 2 byte(s) |
| Opti... | 3 | | | |

Block-Down Block-Up Auto

block no. x block no. x

Token

use hex (0x..) or string x

Observe

use integer x

Payload (9)

Incoming Rendered Outgoing

Vehicles Passed (Event 42)

A Line Sensor notifies vehicles's passage through an Observable Resource

A Client Side Application constructs an histogram of vehicle passage along with their speed by correlating timing information coming from two Line Sensors

Fig. 5. An example of the Resource Observe pattern applied to a Time Event.

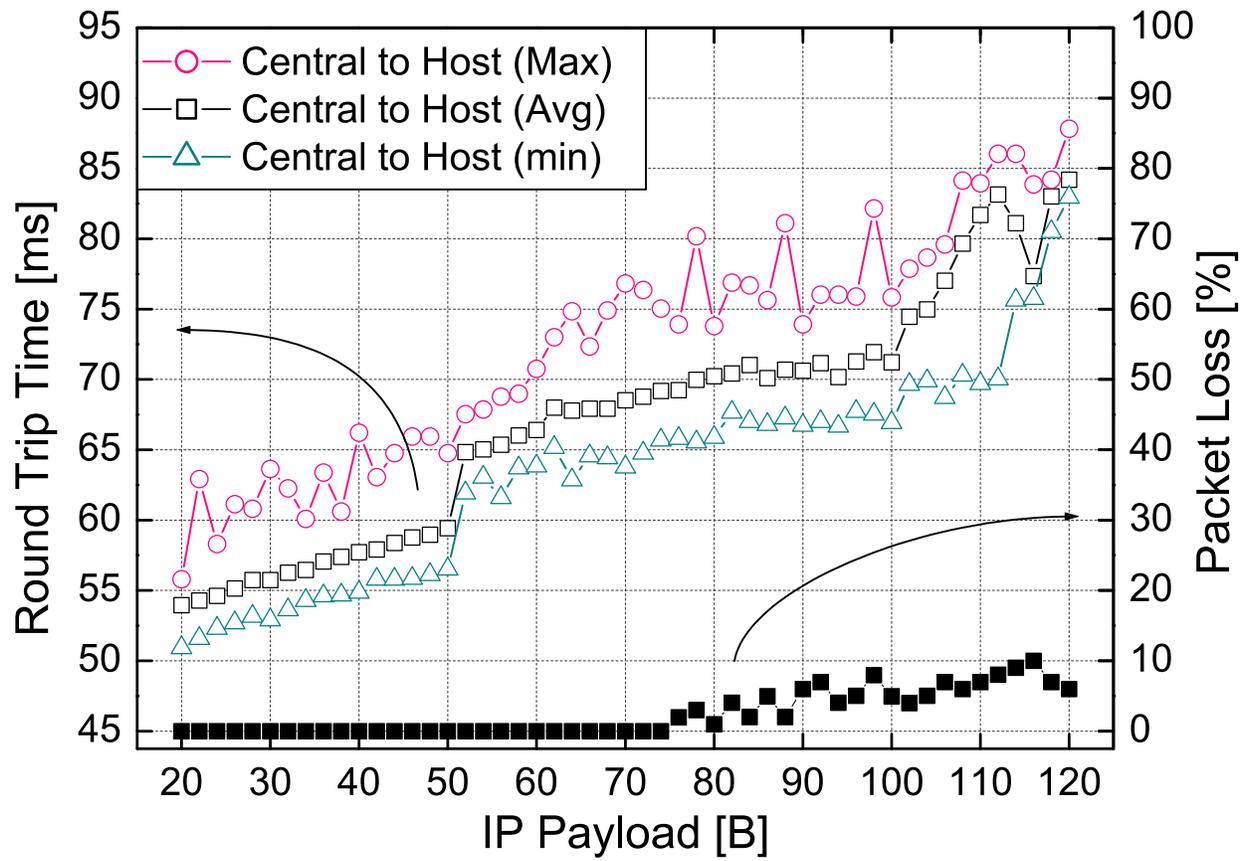


Fig. 6. RTT and PLOSS evolution according to ICMP payload size.