Data Naming in Vehicle-to-Vehicle Communications

Lucas Wang[1], Ryuji Wakikawa[2], Romain Kuntz[2], Rama Vuyyuru[2], and Lixia Zhang[1]

[1]Computer Science Department, University of California, Los Angeles
[2]Toyota InfoTechnology Center, USA Mountain View, CA

***Abstract*—Vehicular networking is becoming reality. Today vehicles use TCP/IP to communicate with centralized servers through cellular networks. However many vehicular applications, such as information sharing for safety and real time traffic purposes, desire direct V2V communications which is difficult to achieve using the existing solutions. This paper explores the named-data approach to address this challenge. We use case studies to identify the design requirements and put forth a strawman proposal for the data name design to understand its advantages and limitations.**

I. INTRODUCTION

Toward safety and comfortable driving experience, the automotive industry has discussed various new concepts such as Intelligent Transportation Systems (ITS) and telematic services over the last 10 years. Technologies are now ready for the implementation and deployment of these concepts. Today, many vehicles are connected to the network and can run a variety of applications on the head unit system, in addition to the usual navigation software.

Beside these services offered through communication with infrastructure servers, the automotive industry has another important network deployment plan: the Vehicle-to-Vehicle (V2V) and Vehicle-to-Road side unit (RSU, small static stations deployed along the roads) networking for safety purposes (e.g. collision avoidance and adaptive cruise control). To that end, the automotive industry has standardized multiple radio and medium access technologies, as we will introduce in Section II-A.

Once vehicles are equipped with these dedicated wireless systems, they will also be able to exchange various application data beyond safety information with neighboring cars and with road-side units. For example, the driving experience could be much improved if all the cars could share traffic information directly among themselves (e.g. to get around accidents). However direct V2V communications bring stringent requirements to the higher level network protocols, such as highly ad hoc connectivity, scaling to large number of vehicles, and independence from infrastructure support. Unfortunately as we will explain in Section II-A, the current TCP/IP implementations are a poor fit to such direct V2V communications.

This paper explores a new approach of using named-data to support direct V2V communications. We investigate the applicability of the Named Data Networking (NDN, aka CCN [1][2]), a newly proposed architecture for the future Internet, to V2V networking in replacement of IP. NDN replaces IP's end-to-end communication model with a request/reply model to retrieve data by application data names directly, potentially eliminating the mobility management, session management and service discovery that are needed in TCP/IP based V2V and V2R solutions. However as we will discuss in Section II-B, applying the NDN concepts to V2V networking raises multiple challenges.

We tackle the above research challenges by first articulating the design requirements through use case studies (Section III). We identify that data naming is the most important and challenging issue when designing NDN-based applications. In Section IV, we sketch out a strawman name structure design for V2V traffic information exchanges, and discuss its advantages and limitations. Finally, Section V discusses a number of important considerations that we have identified.

The contribution of this paper is 3-fold: we first show how V2V communications could take advantage of the NDN scheme, we then identify three design changes to the base NDN data forwarding module as needed to meet design goals, and we also sketch out the data naming design for a traffic information dissemination application. We emphasize that the specific data naming proposal presented in this paper represents the first attempt, rather than the final words, on data naming in vehicle networks; we use the traffic application as a design exercise to explore the new direction of data name-based networking and to identify related research challenges.

II. BACKGROUND

A. V2V communication technologies

Among the variety of wireless technologies for ITS applications [3][4], DSRC (Dedicated Short Range Communications [5], aka IEEE 802.11p [6]) and IEEE 1609 (aka WAVE: Wireless Access in Vehicular Environments) are the emerging ones for direct V2V communications.

Operating on the 5.9Ghz band, DSRC/WAVE can achieve a transmission rage of 100 to 500 meters depending on the radio frequency on which it operates. In this range, a car can reach from 10 to a few hundreds other neighbor vehicles within one single DSRC hop. This technology has proven to be ready: experiments done in [7] show that vehicles traveling at normal speed (e.g. 60 mph) can exchange at least 10 messages per second, with 3K bits in each message. The Japanese government plans to start operating the technology from 2012 and the US government will decide by 2013 whether DSRC/WAVE would be mandated in every car.

Safety applications that are foreseen on top of DSRC/WAVE usually use the broadcast capability of the wireless channel to

978-1-4673-1017-8/12/$31.00

propagate information. In the case of non-safety applications, such as infotainment systems in the vehicle, TCP/IP protocols have been proposed to run on top of DSRC/WAVE for data exchanges among vehicles. For example, IEEE 802.11p has a reference to IPv6 as an alternative to the WAVE Short Message Protocol (IEEE 1609.3) network stack for non-safety data. However, several technical difficulties arise when one attempts to run IP over IEEE 802.11p [8]. Although there has been a rich literature in the context of ad-hoc networking over IP and various routing protocols have been proposed [9], a fundamental limitation in their deployment is the infrastructure support requirement as needed for global IP address allocations. In ad hoc scenarios of V2V and V2R communications, it is infeasible to assume the availability or dependency on infrastructure support. Moreover, constant connectivity changes in a vehicular environment also demand that routes be re-calculated and sessions be re-established at rather high frequency which are also deemed infeasible.

In contrast, a request/reply model of communications using data names requires no allocation of addresses, no setup of data delivery paths, nor session establishment before data delivery can happen. Thus NDN appears as an attractive direction to achieve effective and efficient communications over V2V. However, bringing this conceptually simple model to reality remains a great research challenge, and data naming resides at the center of this challenge.

B. *Challenges of the NDN naming*

Communication in NDN is driven by the receiving end, *i.e.*, the data consumer. To receive data, a consumer sends out an ***Interest*** packet, which carries a name that identifies the desired data and (optionally) a set of rules to determine which piece of content to return if more than one data object matches the Interest. Retrieving data by name is a much better match to the need of ad hoc vehicular networking, as compared to retrieving data from a specific location as today's Internet Protocol does: among moving vehicles, applications running in a car know what information they desire, but not which other vehicles may have it.

Different vehicular applications may desire to sort data into different categories, or with different granularities. For example some car wants to know the expected total driving time from San Jose to San Francisco along Highway 101 and 280, while another car wants to know whether there is any congestion between San Jose and Mountain View along Highway 101 only. Achieving the above functions requires well defined application naming conventions that is understood by all the vehicles, and that is flexible enough to allow vehicles to express exactly what kinds of data they may desire.

As the first step to address the above challenge, this paper depicts a specific use case and discusses the corresponding requirements in the next section.

III. V2V COMMUNICATION VIA NAMED DATA

This section first describes a simple automotive use-case that will be used as an illustration throughout the remainder of the paper. We then explain how we foresee the future automotive telematics platform as well as how NDN can serve this architecture. We will especially detail the underlying NDN technology as well as list the resulting requirements.

A. *Use case: dissemination of traffic information*

Various events on the road (e.g. road work, accident, flooding, etc.) can have a major impact on the traffic conditions. Disseminating information about such events to all the vehicles heading to the event direction may help them decide whether rerouting is necessary.

Some automotive manufacturers operate a cellular-based system to collect and report the traffic information (such as the speed) using centralized servers. This is known as "floating car data" or "probe system". However, letting all vehicles communicate directly with centralized servers raises scalability challenges. Beside, communicating with a remote server is likely to take longer delays than retrieving data directly from nearby vehicles. Instead we envision a traffic information distribution system that is composed by all the cars traveling on the roads and that generate, disseminate, and consume traffic information among themselves (in addition to providing the information to centralized servers). For example, vehicles driving by a road event can generate and store a variety of data about it, such as:

- The timestamp when the data was produced, e.g. a UNIX timestamp 1323201889 (which means *Tuesday, December 6th 2011, 12:04:49*),
- Location information such as latitude and longitude of the car (collected by the GPS) where the data is generated (e.g. *37.446346,-122.120183*), and/or the road name on which the car is traveling (e.g. *Highway 101, between exit 402 and 403*), as well as in which direction the car is traveling, e.g. *Northbound*, and
- The associated measured data such as the traveling speed of the car, e.g. *20mph*.

Such traffic data could be generated either once (e.g. upon sudden break) or periodically (e.g. the average speed over the last 5 minutes).

To demonstrate how we could use NDN to achieve the functions from the above use case, Sections III-B will first describe the in-vehicle system architecture, and Section III-C will explain how the data could be disseminated among the vehicles.

B. *In-vehicle system assumptions*

Figure 1 depicts an overview of our proposed future in-vehicle telematic architecture. The primary goal of our system is to serve the driver with valuable information through the head unit of the car (or through any other popular device that the driver may bring to the vehicle, such as a smartphone or tablet). When the navigation application running on the head unit requests new traffic information, for example traffic speed toward a specific destination, it could fetch from surrounding vehicles in addition to any data it has locally.

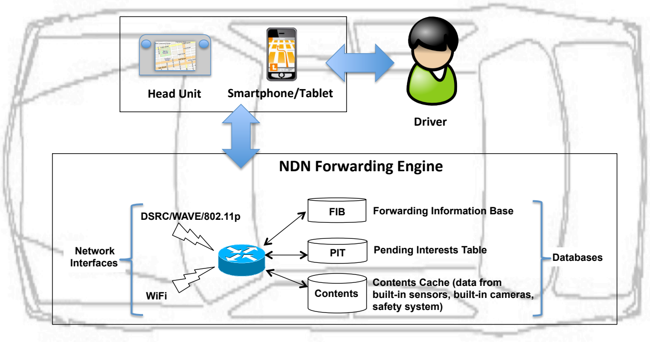

Fig. 1. Overview of the proposed in-vehicle telematic architecture. The head unit provides traffic information (e.g. traffic information) to the driver. The information can be fetched from surrounding vehicles in a V2V fashion by using the NDN forwarding engine embedded in the vehicle.

For that purpose, we assume that each vehicle embeds a NDN forwarding engine (later referred to as NDN module) that manages one or multiple network interfaces, such as DSRC/WAVE and WiFi. In our example, when the navigation application residing on the head unit instructs the NDN module to retrieve the desired information, the NDN module can then use WiFi in ad-hoc mode and/or DSRC/WAVE to send out Interest packets to other vehicles through one-hop broadcast, without any network infrastructure support. Similarly, it could also retrieve information from the road-side units which also run a NDN module.

In addition to retrieving needed data, as we mentioned earlier each vehicle also generates various data, either periodically or in an event-driven fashion, from its built-in sensors, cameras and safety system (such as collision avoidance). These locally generated data would be recorded in a large local data store. Compared to other mobile communications, vehicle networking possesses one distinct advantage, i.e. each vehicle can have sufficient memory space and transmission power.

Another unique feature of vehicle networks is the rapid movement – cars on the road are busy driving towards different locations. Together with the large storage they can afford to possess, they can serve as data mules in addition to being data producers and consumers. Since one cannot predict what information may be needed by other vehicles, being a good data mule means that each vehicle should try to collect as much information as feasible.

With the above functions in mind, let us consider the NDN module residing in the head unit. According to [1], an NDN module contains three major databases: a Forwarding Information Table (FIB), a Pending Interest Table (PIT), and a Content Store (data cache). In applying NDN to V2V networking, we identify the following few adjustments. As the first step in investigating the name-oriented V2V communications, in this paper we focus on single-hop Interest packet forwarding only, which does not use the FIB. Instead we examine the simplest case where Interest packets that are broadcast by one vehicle can be received by single-hop neighboring cars and RSUs; Section V provides further discussions regarding multihop Interest forwarding.

Second, in a highly dynamic environment where vehicles move at high speed, the probability of an unanswered Interest can be significantly higher than in wired Internet where each Interest packet is expected to retrieve a data packet with high probability. Thus our design must consider different schemes in the PIT management.

A third departure from the general NDN module concerns the data acceptance. According to the description in [1], a NDN router would accept a data packet only if it has a corresponding entry in its PIT table. However in V2V communication, given the broadcast nature of DSRC/WAVE and WiFi channels, a vehicle can often hear data packets in responses to other vehicles' requests. In order to serve as an more effective data mule, one should take the opportunity to cache all received data packets even when they are requested by others, rather than ignoring them.

C. System operation

Based on the above discussion, we define three different roles played by vehicles and RSUs in our system as depicted in Figure 2: data publisher, data mule and data consumer[1].

A data publisher produces data and stores it in its cache (Figure 2, step ①) and make it available for distribution. The publisher may receive, at anytime, some Interests from other vehicles. The data publisher would then determine if any data that it carries matches the Interest. If so, the data publisher would send out a response packet with the matched data.

A data mule is a vehicle which collects data from other vehicles in addition to its own data, and provides the data to other interested parties upon request[2]. A data mule sends Interest packet periodically to retrieve data that can be potentially useful for other vehicles, continuously listens to the medium and cache the data it hears (step ②), as well as responds to other vehicles requests when it finds matching data from its cache. The data mule can thus collect a large amount of data that could not only serve its own purpose, but also transport that data from one point to another to further redistribute it. *Data muling* is a critical function in NDN-based mobile networks and a departure from the NDN operations in wired Internet. In our design a data mule serves two critical roles: not only it helps disseminate the traffic information to vehicles in many different locations, it also keeps the information available even when the original publisher has gone off the system (e.g. reached its destination and turned off).

A data consumer sends Interest packets to retrieve data from publishers and mules (step ③). In NDN, each data packet has a unique name and data consumers can explicitly spell out their desired data name in the Interest packets. For example, whenever the navigation application wants to know the traffic condition towards a given destination, the NDN module would broadcast Interest packets to reach the surrounding cars and RSUs, in expectation of receiving replies from any cars and/or RSU that may have the matching data.

---

[1] Of course RSUs can only cache data but not physically move it.

[2] A RSU can also serve as a stationary data mule to pass data between passing vehicles.

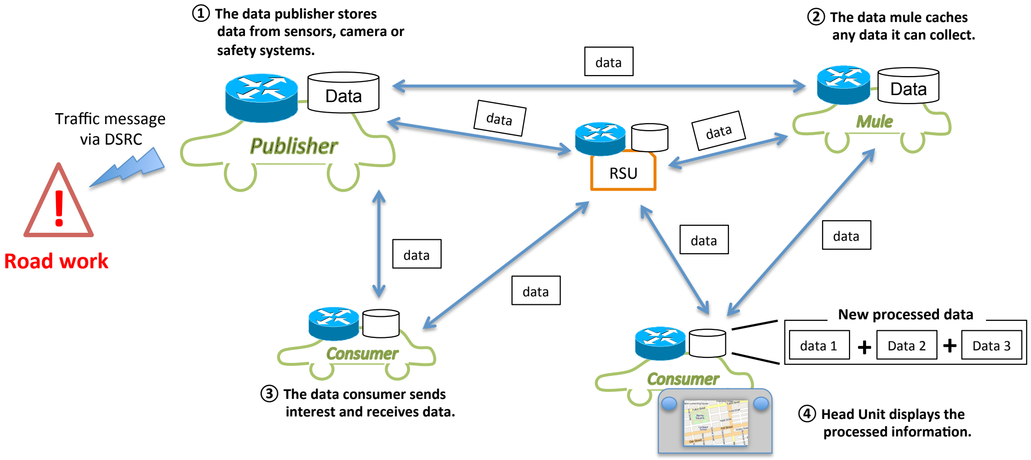

Fig. 2. Data publishers, mules and data consumers. The publishers produce and store data about specific events (e.g. road work ①). The mules cache all the data they can collect ② for a later redistribution. Consumers request specific data using Interest packets ③ targeting the publishers or the mules. Multiple data can be processed into a new one which carries a different information. That processed data can be fetched by consumers e.g. for head unit applications ④.

A data consumer may collect multiple pieces of data from different cars regarding the same traffic event and perform further processing to increase the accuracy of the obtained traffic information. For example, by collecting the speed information from cars that pass by the same road event, a consumer can compute the average speed limit at the event location to estimate what would be the required time to pass that event. Such processed data may be directly used by head unit applications (step ④) to inform the driver about the traffic information towards his final destination. The consumer may also publish this processed result to make it available to other interested vehicles.

D. Requirements on data naming

The functionality described above can be supported only if we have a good data naming design that allows data publishers to describe precisely what they have, and data consumers to express what information they need. In this section, we summarize the identified requirements on the data naming design from the above use cases:

- *Geographical Scoping*: a car must be able to request traffic data surrounding specific geographical area, e.g. a specific section of highway 101; the range can be either small or big.
- *Temporal Scoping*: a car must be able to request data for a specified time period. One may desire either data that is fresh enough to be useful for one's own driving, e.g. any road event occurred within the last 30min between one's current location and the destination, or data with a longer history, e.g. over the last day, as a reference[3].
- *Duplication detection*: An Interest packet may potentially match the data stored in a number of neighboring cars within the DSRC/WAVE transmission range. As vehicles move rapidly on the road and periodically collect updated traffic information, it is important to identify the replies of the same data by the same producer. Furthermore, data packets from multiple publishers that describe the same event should also be easily identified by the consumer.
- *Data accuracy*: It is important for the consumer to obtain accurate traffic information. However certain percentage of the collected data can be inaccurate or even contain false value due to malfunctioning sensors or the highly dynamic environment. One way to increase data accuracy is to leverage on redundancy, i.e. based on the number of vehicles advertising data for the same road event as we discussed in Section III-C.
  It is also possible that some vehicles may publish false data on purpose. This malicious producer problem can be addressed through cryptographic-based security measures, as we discuss in Section V.
- *Application Type*: we derived the above requirements from the traffic information dissemination application. However different applications may have different requirements on data naming. Thus the naming structure should be able to accommodate different data naming structures for individual applications.

Based on this list of requirements, we sketch out a strawman data name structure in the next section.

IV. DATA NAMES FOR TRAFFIC APPLICATION

In this section, we perform a design exercise for data names that can meet the requirements we identified from the use case. In Section V we discuss the usability and limitations of the proposed name structure.

A. Naming Design

We propose to use the following structure of data names for V2V traffic information exchanges: **/traffic/geolocation/timestamp/data type/nonce**.

---

[3] Such historical traffic data may not be available from current vehicles on the road but can be stored in infrastructure servers, and our design must allow its retrieval by the same data name structure.

The first component *traffic* serves as the application id, and makes the following naming structure design to be application dependent. The *geolocation* component uses the format of *road ID/direction/section number*. The *road ID* component represents the road name which must be unique. Inter-state highways all have unique IDs. Current navigation systems will use a unique link ID to identify the road segment they are traveling on. If needed (e.g. for local highways and streets that may have the same road names in different cities and states), we can use a combination road ID and link ID to ensure uniqueness. The *direction* component represents the traffic direction. The *section number* component can be the exit numbers of highways. A pair of section numbers is used to represent a region on the highway.

The *timestamp* component uses the same UNIX timestamp format that is used for the data. Similar to the geolocation component, we use a pair of timestamps to represent a time period. When a consumer receives a data packet, the application module can convert the timestamp into a more human-readable format, e.g. *Tuesday, December 6th 2011, 12:00:00-13:00:00*.

The *data type* component indicates the meaning of the data itself, e.g. closed lane, vehicle speed, etc. The trailing component *nonce* is a large random number generated by the publisher. It is possible that multiple publishers generate data with identical values for the above components in the data name, and use the *nonce* component to distinguish data generated by different producers.

Following the above explanation, here is a specific example of a name: */traffic/Highway 101/north/{400,410}/{1323201600,1323205200}/speed/19375887*. The brackets in the section number and timestamp components specify a desired range and allows to request for data that was generated geographically between section 400 and 410 on Highway 101 north, and temporally between 12:00 and 13:00 on December 6th, 2011.

B. Data Retrieval Example

When a consumer wants to retrieve data packets, it expresses its Interests following the above data naming convention. Our naming design is flexible enough that it can handle data retrieval at different granularities. The following are several examples:

- A car can use the interest */traffic/Highway 101/north/{400,410}/{1323201600,1323205200}/speed/* to request speed data within the specified 10-section and 1-hour scope. If a car wants to request data in a scope that goes across the unit boundaries, say between section 405 and 415, it can use multiple interests, e.g. */traffic/Highway 101/north/{400,410}/{1323201600,1323205200}/speed* and */traffic/Highway 101/north/{410,420}/{1323201600, 1323205200}/speed.*
- The interest */traffic/Highway 101/north/{400,410}/* means the requester would like to receive traffic data in the specified region generated at any time and regardless of the data type.
- An Interest carrying the name */traffic/Highway 101/* requests all traffic information on highway 101 regardless of the regions and time.

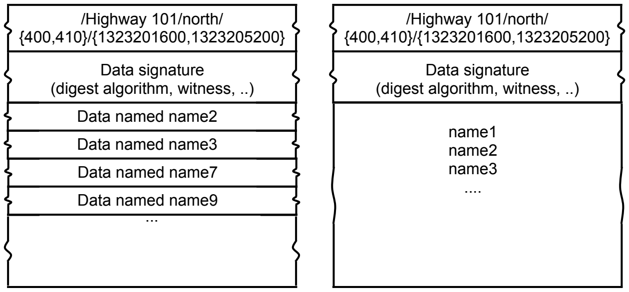

Fig. 3. The left side is a sample of reply packet containing several small pieces of data; the right side is a packet containing a list of data names rather than actual data.

When a vehicle C is within the wireless broadcast range of an Interest and finds matching data in its own cache, it generates a response packet with the data. Note that although the name carried in an Interest packet may indicate a wide region and a time period, each piece of the data is generated at a specific location with a specific timestamp. One design option is to put the specific location and timestamp values inside the data; another option is to use the specific location and timestamp, instead of a pair of values, in the data name[4].

There can be cases where C may have multiple pieces of data that all match the received Interest, but not all of them can fit into one reply packet. In this case, we are currently considering the following two options.

- The first option is to only list the names of the available data pieces[5], rather than actual data. For instance, if the data named $name_1, name_2, ..., name_{10}$ all match an Interest but cannot fit into one reply, C can send a reply that contains name $name_1, ..., name_{10}$, as shown on the right in figure 3. The consumer can then issue subsequent Interests with specific names to retrieve any of them as it wishes.
- Considering the short time window within which vehicles must be able to exchange data, the second option is to send a reply with randomly selected pieces of the matching data. Using the same example as above, if C can only put 4 of the 10 pieces of matching data in a data packet, it may randomly select 4 pieces in reply. If more than one vehicle replies to the same Interest, there is a good chance that each of them selected different pieces.

C. Meeting system requirements

In the above we showed how our proposed name design incorporated the temporal and geographical scoping as well as application type into the name. Below we explain how our design addresses the remaining requirements.

---

[4] However this latter option leads to the need to interpret data name to find the matching Interest, instead of using binary bit string matching.

[5] This option requires that data names carry specific location and timestamp values.

Duplication detection and elimination: Since vehicles may broadcast Interests periodically to collect traffic data, one wishes to avoid receiving the same data piece generated by the same producer repeatedly. This can be achieved through the use of an *exclusion filter* [10]. Recall that the last component in each data name is a random nonce, a requester can simply include the nonces of the data that match the name carried in the Interest but have already been received.

Data accuracy: a piece of data (e.g. average speed) may contain input data names (e.g. data names representing individual car speed) that are used to generate itself. The input data names inside may serve two purposes. First receivers of the result data can get an estimate of how accurate it is based on the number of data inputs. Second, receivers who want to further improve data quality need to avoid reusing the same input data for further processing.

V. DISCUSSION

We have taken an initial step towards understanding data naming designs in V2V networks by examining a specific use case, the dissemination of traffic information through one-hop V2V and V2R data exchanges. Although our investigation is still far from finish even for this simple case, we have identified several design issues and new insights. We discuss three of them below.

First, we believe it is likely to be the common case that data names for vehicular applications need to contain *both* location and time information, and with flexible data granularities along each of the two dimensions. A data name consists of a linear sequence of components and can easily represents data of different granularity along one dimension [1][2]. How to design a name structure that can support flexible granularity along multiple dimensions remains an open issue. The previous section showed two design options to address this issue.

Second, this work used road ID plus section number to represent geolocations. Another option is to use a combination of road ID with latitude and longitude values to represent a location region, which we plan to evaluate in the next step.

Third, the highly dynamic environment of vehicle networks defeats the conventional routing protocols, and we would further raise a bold question on whether a routing protocol is necessary for peer-to-peer based V2V communications as described in this paper. As one can see from Section III-C's description, although Interest packets are forwarded through one-hop broadcast only, vehicles can indeed receive traffic information retrieved from far away and across multiple hops, thanks to data muling by all the vehicles. Consider a car X near San Jose generates some data D for event E, car Y traveling north may request and carry D over distances and forward to car Z in Palo Alto when Z requests the data. Z in turn may further forward data to car W. As one can see from this example, although individual Interest packets only propagate one-hop, data on the other hand can travel multiple hops to reach the parties who desire it. We plan to use simulations to examine the effectiveness of such multihop data propagation in meeting user information needs.

Another important issue that has not been discussed so far is providing data authenticity while protecting vehicle privacy at the same time. We refer interested readers to our ongoing work [8] about using group signature as a solution to this problem.

VI. CONCLUSION

In this paper we used a simple case study to investigate data name designs in vehicle networking. In the process we identified three different roles vehicles may play in traffic data dissemination. We also identified three changes to the NDN forwarding module to make it more suited in a vehicle network.

To the best of our knowledge, we believe this paper represents the first exploration of applying named data approach to V2V and V2R communications. Instead of using IP addresses, our design allows vehicles to collect and disseminate traffic information among themselves using data names that are defined a priori during application development and understood by all vehicles. The use of such data names enables request-reply model of data exchanges that are best suited in the ad hoc environment characterized by rapid changing and short durations of wireless connectivities among vehicles.

Clearly, the effectiveness and flexibility of this solution largely rely on the proper selection of the data name structure that can be used by vehicles to express what data they desire under different situations and when they play different roles in the overall system. What is the best way to design the data name structure remains an open challenge. Rather than asserting any specific results, this paper serves as our invitation to the broader community to join the exploration of named-data approach to vehicular networking.

REFERENCES

[1] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking Named Content," in *CoNEXT: International Conference on Emerging Networking Experiments and Technologies*. Rome, Italy: ACM, December 2009.

[2] L. Zhang *et al.*, "Named Data Networking (NDN) Project," in *PARC Technical Report NDN-0001*, 2009. [Online]. Available: http://www.named-data.net/

[3] K. Dar, M. Bakhouya, J. Gaber, and M. Wack, "Wireless communication technologies for its applications," in *IEEE Communications Magazine*, vol. 48 issue5, May 2010, pp. 156–162.

[4] J. Eriksson, H. Balakrishnan, and S. Madden, "Cabernet: Vehicular content delivery using wifi," *MOBICOM*, 2008.

[5] J. B. Kenney, "Dedicated Short-Range Communications (DSRC) Standards in the United States," in *Proceedings of the IEEE*, vol. 99, no. 7, July 2011, pp. 1162–1182.

[6] IEEE Standard for Information Technology, "Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements; Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications; Amendment 6: Wireless Access in Vehicular Environments," in *IEEE Std. 802.11p*, July 2010.

[7] Ahmed-Zaid, F. et al., "Vehicle Safety Communications – Applications (VSC-A) Final Report," *DOT HS 811 492A*, 2011.

[8] R. Wakikawa, R. Kuntz, R. Vuyyuru, L. Wang, and L. Zhang, "Vehicular Networking via Named Data," *Submitted to IEEE Communications Magazine special issue on Information-Centric Networking*.

[9] J. Chennikara-Varghese, W. Chen, O. Altintas, and S. Cai, "Survey of Routing Protocols for Inter-Vehicle Communications," in *MobiQuitous: Conference on Mobile and Ubiquitous Systems*, July 2006.

[10] "http://www.ccnx.org/releases/latest/doc/technical/InterestMessage.html."