Data serialization protocols in IoT: problems and solutions using the ThingsBoard platform as an example

Dmytro I. Shvaika[1,2], Andrii I. Shvaika[1,2] and Volodymyr O. Artemchuk[1,3,4,5]

[1]G.E. Pukhov Institute for Modelling in Energy Engineering of the National Academy of Sciences of Ukraine, 15 General Naumov Str., Kyiv, 03164, Ukraine

[2]ThingsBoard, Inc., 110 Duane Street, Suite 1C, New York, 10007, USA

[3]Center for Information-analytical and Technical Support of Nuclear Power Facilities Monitoring of the NAS of Ukraine, 34a Palladin Ave., Kyiv, 03142, Ukraine

[4]Kyiv National Economic University named after Vadym Hetman, 54/1 Peremohy Ave., Kyiv, 03057, Ukraine

[5]National Aviation University, 1 Liubomyra Huzara Ave., Kyiv, 03058, Ukraine

Abstract

This article delves into the challenges and advancements in data serialization protocols within the Internet of Things (IoT), primarily focusing on dynamic schema compilation in ThingsBoard. A comparative analysis of Protobuf against other serialization protocols like JSON, XML, and PSON highlights Protobuf's efficiency and outlines the necessity for flexible ways of device integration that use Protocol Buffers for data transmission. We identify the limitations of static schema compilation in Protobuf and propose a novel approach for real-time, user-driven schema compilation that enhances flexibility, scalability, and performance in IoT platforms. Our solution addresses critical adaptability issues by enabling seamless device communication and integration using compact Protobuf formats. We emphasize the potential impact of this solution in the scope of edge computing and suggest directions for future research to broaden the applicability of dynamic serialization across various IoT solutions. This work contributes to improving IoT data management and paves the way for more adaptable and efficient IoT ecosystems.

Keywords

IoT Platform, Data Serialization, Protocol Buffers, ThingsBoard

1. Introduction

In the contemporary landscape, where the Internet of Things (IoT) is gaining prominence [1], data processing and transmission effectiveness emerge as a pivotal determinant of technological success. Data serialization protocols play a crucial role in this domain, facilitating the exchange of information among IoT devices in a compact and efficient format. Widely employed protocols like JSON, XML, Protocol Buffers, and others cater to various IoT systems, addressing the demand for swift and dependable communication. Nevertheless, each protocol presents unique challenges and constraints concerning integration and scalability within intricate IoT ecosystems.

The Internet of Things (IoT) is a rapidly evolving field with many applications. Debnath and Chettri [2] and Villamil et al. [3] highlight IoT's diverse applications, including in industry, business, and improving quality of life. Uckelmann et al. [4] emphasizes the potential for IoT to revolutionize business processes and enable a more convenient way of life. Porkodi and Bhuvaneswari [5] provides a detailed overview of the communication-enabling technology standards in IoT, such as RFID tags and sensors. The study by Khang et al. [6] addresses the limitations of single-path communication in hydroponic systems, emphasizing the need for reliable multi-path communication in IoT-based monitoring systems.

However, when it comes to the specific topic of data serialization protocols in IoT, the literature is relatively scarce (Luis et al. [7], Friesel and Spinczyk [8], Domínguez-Bolaño et al. [9], Pustišek

doors-2024: 4th Edge Computing Workshop, April 5, 2024, Zhytomyr, Ukraine

shvaikad@gmail.com (D. I. Shvaika); andrew.shvayka@gmail.com (A. I. Shvaika); ak24avo@gmail.com (V. O. Artemchuk)
https://www.researchgate.net/profile/Volodymyr-Artemchuk-2 (V. O. Artemchuk)
0009-0001-3088-3997 (D. I. Shvaika); 0009-0006-8461-3550 (A. I. Shvaika); 0000-0001-8819-4564 (V. O. Artemchuk)

et al. [10], Delgado [11], Jacoby and Usländer [12], Deniziak et al. [13], Jiang et al. [14], Hou et al. [15], Hasemann et al. [16], Kolbe et al. [17], Kharat et al. [18], Khodadadi and Sinnott [19]).

The ThingsBoard Platform has garnered substantial popularity among researchers, as evidenced by numerous publications dedicated to its utilization. In particular, the following examples highlight its prominence in the academic community (Ilyas et al. [20], Henschke et al. [21], Aghenta and Iqbal [22], De Paolis et al. [23], Casillo et al. [24], Okhovat and Bauer [25], Bestari and Wibowo [26], Sabuncu and Thornton [27], Jang et al. [28], Kadarina and Priambodo [29]).

In this article, we focus on analyzing data serialization protocols within the context of IoT, examining their applications and the challenges they present to developers and engineers. The ThingsBoard platform, recognized as one of the leading open-source IoT platforms, is a practical instrument in this investigation, allowing for a detailed analysis of various facets of data serialization. Its adaptability and scalability in addressing IoT device management and data processing tasks make it an ideal candidate for delving into the intricacies of serialization protocols within IoT environments.

Research object is data serialization protocols in distributed IoT systems, emphasizing utilizing methods and mechanisms for data transfer between devices and the system. *Research subject* is characteristics and performance of serialization protocols, encompassing data size, processing speed, and utilizing ThingsBoard for practical analysis. *Research objective* is to analyze and assess data serialization protocols in the IoT landscape, delineating their advantages and exploring avenues for improvement, specifically focusing on their impact on performance and flexibility across various IoT scenarios.

2. Comparative analysis of data serialization protocols for IoT

An ordinary device transforms into an IoT device upon integration with an IoT platform, functioning through data exchange with fellow IoT devices or cloud servers. This necessitates a standardized data exchange format at the application level. To address this challenge, libraries offering standardized data formats are readily accessible. However, the costs related to data (de)serialization and transmission with these libraries are largely undocumented in the realm of IoT, or documented in limited capacities for specific protocols. The Friesel and Spinczyk [8] study examined JSON JSON [30] encoding efficiency within the IoT framework. This involved a comparative analysis juxtaposing JSON with alternative serialization formats. The results underscored the efficacy of Protocol Buffers, or Protobuf, highlighting their suitability for energy-efficient data serialization in the context of contemporary, high-capacity IoT devices. The Luis et al. [7] study focused on assessing the performance metrics of PSON, comparing it against a spectrum of formats, including Protocol Buffers Google [31]. This comprehensive analysis covered various dimensions, such as serialization/deserialization velocities, binary file dimensions, and encoding sizes. Building upon the findings of these studies, we present a comparative analysis tailored to elucidate the strengths and limitations of these protocols within the context of IoT applications. The table 1 provide key characteristics of leading data serialization protocols, highlighting their respective advantages and constraints.

Table 1
Comparative analysis of data serialization protocols for IoT (based on Luis et al. [7], Friesel and Spinczyk [8]).

Feature	Protobuf	JSON	XML	PSON
Format type	Binary	Text-based	Text-based	Binary
Efficiency (size)	High	Medium	Low	High
Efficiency (speed)	High	Medium	Low	High
Human readable	No	Yes	Yes	No
Language support	High	High	High	Medium
Extensibility	Yes	Yes	Yes	Yes
Versioning support	Yes(Proto2, Proto3)	No	No	No
IoT device compatibility	High	High	Medium	High

It is clear from the benchmarking that Protobuf is the leader in data serialization for IoT due to its high efficiency in both size and speed, wide language support and extensibility. Despite the rapid development and potential advantages of formats such as PSON, the presence of Protobuf and its continued use in various IoT applications reaffirms its importance.

3. Challenge of device integration over Protocol Buffers in IoT platforms

The challenge of integrating devices over Protocol Buffers in IoT platforms is a universal issue, not confined to a specific platform. Consequently, for our analysis, we've chosen ThingsBoard as our research tool. ThingsBoard, Inc. was founded in 2016 by a team of programmers from Ukraine and specializes in the development of software products for the IoT. ThingsBoard [32], with its open-source nature and comprehensive features, provides a robust foundation for exploring these challenges and potential solutions in a detailed and practical manner.

The IoT developers at ThingsBoard opted for schemaless JSON formats for primary serialization in external communication, facilitating data exchange with IoT devices due to their user friendly nature. In the ThingsBoard system, Protocol Buffers is used for inter-component data exchange. This decision is motivated by the need for streamlined processing of substantial data volumes while maintaining superior system performance. The compact nature and rapid serialization/deserialization of Protocol Buffers render it an optimal selection for enhancing internal network efficiency.

Currently, there is a growing interest in utilizing Protocol Buffers directly at the device level. Certain IoT devices transmit data solely through Protocol Buffers, while other users seek ways to transition to this format to enhance efficiency and reduce network load.

The integration of IoT devices that exclusively communicate using Protocol Buffers into IoT platforms exemplifies a pressing challenge, particularly for open-source platforms like ThingsBoard. Protobuf's static nature necessitates additional developer intervention for each new device type, undermining the platform's universality and scalability, especially in cloud deployments. To integrate a new Protobuf-compatible device, developers must manually define and compile the device's schema into the platform's codebase. This process that is both time-consuming and prone to errors.

A notable example is the integration of Efento devices into ThingsBoard using CoAP and Protobuf for seamless connectivity. The Efento [33] describes the interaction between Efento NB-IoT sensors and the ThingsBoard platform. Simultaneously, with device firmware versions in constant evolution, a scenario emerges wherein the platform must continually adapt to support new or updated devices. This interdependence raises questions about the sustainability of the platform in the IoT environment.

This scenario underscores the necessity for IoT platforms to develop more dynamic and versatile data serialization solutions. A mechanism that allows for the real-time, dynamic compilation and loading of Protobuf schemas would revolutionize device integration, enabling seamless adaptation to new devices and data formats without extensive developer intervention or system disruption.

4. Dynamic schema compilation in Protobuf by ThingsBoard

The preference of Protocol Buffers in IoT applications lies in its binary format's efficiency and the reduced load it imposes on network transmission. However, its static nature presents a formidable challenge. Typically, .proto files must be pre-compiled using the Protobuf compiler (protoc), producing source code for the desired programming languages. Any alterations to the schema necessitate a tedious cycle of recompilation and redeployment, impeding the rapid adaptability required in the fluid IoT ecosystems.

Addressing this, we propose a software tool enabling the real-time compilation of user-uploaded Protobuf schemas. This approach departs from traditional methods by allowing dynamic interpretation of Protobuf schema, thus permitting devices to communicate their data in Protobuf without necessitating system downtime or recompilation of the entire codebase. The solution is encapsulated within the

ThingsBoard platform through the concept of Device Profiles [34], which associate devices with their respective data transmission schemas.

In practice, each schema represents a distinct device's communication blueprint. Once a device is authenticated, its linked profile helps identify the pertinent schema for message interpretation. This dynamic process significantly lightens network traffic, as data is transmitted in Protobuf's compact form and only translated into a more verbose format like JSON when user interaction or specific system functions necessitate it.

This approach ensures that as IoT devices evolve or new ones join the network, the system can swiftly accommodate them without extensive manual interventions or halts in operation. It represents a leap toward an adaptable IoT platform capable of keeping pace with the sector's rapid growth and the diverse array of devices it encompasses.

5. Conclusions

This article explored the evolving landscape of data serialization protocols in IoT, with a special focus on the dynamic schema compilation feature within ThingsBoard. We've demonstrated how Protobuf, despite its efficiency and reduced network load, faces challenges in static schema compilation, limiting IoT devices' adaptability. Our findings suggest that the innovative solution of real-time, user-driven schema compilation can significantly enhance IoT platforms' flexibility, scalability, and overall performance. By enabling devices to communicate using compact Protobuf formats while allowing for seamless integration of new or updated devices, this approach addresses key scalability and adaptability challenges.

For future research and development, it would be insightful to delve deeper into how such dynamic data serialization mechanisms can further benefit edge computing scenarios. Specifically, investigating the impact on latency reduction, bandwidth optimization, and overall system responsiveness when deploying IoT devices in edge-centric networks. Additionally, exploring the integration of these serialization techniques with edge computing models could offer novel approaches to managing data flow and processing between edge devices and central systems, ultimately contributing to the scalability and robustness of IoT solutions.

References

[1] Y. B. Shapovalov, Z. I. Bilyk, S. A. Usenko, V. B. Shapovalov, K. H. Postova, S. O. Zhadan, P. D. Antonenko, Harnessing personal smart tools for enhanced STEM education: exploring IoT integration, Educational Technology Quarterly 2023 (2023) 210–232. doi:10.55056/etq.604.

[2] D. Debnath, S. K. Chettri, Internet of Things: Current Research, Challenges, Trends and Applications, in: X.-Z. Gao, R. Kumar, S. Srivastava, B. P. Soni (Eds.), Applications of Artificial Intelligence in Engineering, Algorithms for Intelligent Systems, Springer, Singapore, 2021, pp. 679–694. doi:978-981-33-4604-8_52.

[3] S. Villamil, C. Hernandez, G. Tarazona, An overview of internet of things, TELKOMNIKA (Telecommunication Computing Electronics and Control) 18 (2020) 2320–2327. doi:10.12928/telkomnika.v18i5.15911.

[4] D. Uckelmann, M. Harrison, F. Michahelles (Eds.), Architecting the Internet of Things, Springer, Berlin, Heidelberg, 2011. doi:10.1007/978-3-642-19157-2.

[5] R. Porkodi, V. Bhuvaneswari, The Internet of Things (IoT) Applications and Communication Enabling Technology Standards: An Overview, in: 2014 International Conference on Intelligent Computing Applications, Coimbatore, India, 2014, pp. 324–329. doi:10.1109/ICICA.2014.73.

[6] A. W. Y. Khang, J. A. J. Alsayaydeh, J. A. B. M. Gani, J. B. Pusppanathan, A. A. Teh, A. F. M. F. Ismail, T. K. Geok, Reliable Multi-Path Communication for IoT Based Solar Automated Monitoring as Motivation Towards Multi-Farming Hydroponic, International Journal of Interactive Mobile Technologies 17 (2023) 115–128. doi:10.3991/ijim.v17i21.43555.

[7] Á. Luis, P. Casares, J. J. Cuadrado-Gallego, M. A. Patricio, PSON: A Serialization Format for IoT Sensor Networks, Sensors 21 (2021) 4559. doi:10.3390/s21134559.
[8] D. Friesel, O. Spinczyk, Data Serialization Formats for the Internet of Things, in: Conference on Networked Systems 2021 (NetSys 2021), Electronic Communications of the EASST, Berlin, 2021. URL: https://journal.ub.tu-berlin.de/eceasst/article/view/1134/1078. doi:10.14279/tuj.eceasst.80.
[9] T. Domínguez-Bolaño, O. Campos, V. Barral, C. J. Escudero, J. A. García-Naya, An overview of IoT architectures, technologies, and existing open-source projects, Internet of Things 20 (2022) 100626. doi:10.1016/j.iot.2022.100626.
[10] M. Pustišek, A. Umek, A. Kos, Approaching the communication constraints of ethereum-based decentralized applications, Sensors 19 (2019) 2647. doi:10.3390/s19112647.
[11] J. C. M. Delgado, An Interoperability Framework and Distributed Platform for Fast Data Applications, in: Z. Mahmood (Ed.), Data Science and Big Data Computing: Frameworks and Methodologies, Springer International Publishing, Cham, 2016, pp. 3–39. doi:10.1007/978-3-319-31861-5_1.
[12] M. Jacoby, T. Usländer, Digital twin and internet of things-Current standards landscape, Applied Sciences 10 (2020) 6519. doi:10.3390/APP10186519.
[13] S. Deniziak, M. Płaza, Ł. Arcab, Approach for Designing Real-Time IoT Systems, Electronics 11 (2022) 4120. doi:10.3390/electronics11244120.
[14] T. Jiang, X. Huang, S. Song, C. Wang, J. Wang, R. Li, J. Sun, Non-Blocking Raft for High Throughput IoT Data, in: Proceedings - International Conference on Data Engineering, volume 2023-April, 2023, pp. 1140–1152. doi:10.1109/ICDE55515.2023.00092.
[15] C.-D. Hou, D. Li, J.-F. Qiu, L. Cui, EasiDEF: a horizontal lightweight data exchange protocol for internet of things, Jisuanji Xuebao/Chinese Journal of Computers 38 (2015) 602–613. doi:10.3724/SP.J.1016.2015.00602.
[16] H. Hasemann, A. Kröller, M. Pagel, RDF provisioning for the Internet of Things, in: 2012 3rd IEEE International Conference on the Internet of Things, 2012, pp. 143–150. doi:10.1109/IOT.2012.6402316.
[17] N. Kolbe, J. Robert, S. Kubler, Y. L. Traon, PROFICIENT: Productivity Tool for Semantic Interoperability in an Open IoT Ecosystem, in: Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, MobiQuitous 2017, Association for Computing Machinery, New York, NY, USA, 2017, p. 116–125. doi:10.1145/3144457.3144479.
[18] P. Kharat, V. Chaudhari, H. Maurya, S. Junghare, An inventory management using cloud function and protocol buffer for improved efficiency, IET Conference Proceedings 2023 (2023) 253–260. doi:10.1049/icp.2023.1499.
[19] F. Khodadadi, R. O. Sinnott, A Semantic-aware Framework for Service Definition and Discovery in the Internet of Things Using CoAP, Procedia Computer Science 113 (2017) 146–153. doi:10.1016/j.procs.2017.08.334.
[20] T. F. Ilyas, F. Arkan, R. Kurniawan, T. H. Budianto, G. B. Putra, Thingsboard-based prototype design for measuring depth and ph of kulong waters, IOP Conference Series: Earth and Environmental Science 926 (2021). doi:10.1088/1755-1315/926/1/012025.
[21] M. Henschke, X. Wei, X. Zhang, Data Visualization for Wireless Sensor Networks Using ThingsBoard, in: 2020 29th Wireless and Optical Communications Conference, WOCC 2020, 2020, pp. 1–6. doi:10.1109/WOCC48579.2020.9114929.
[22] L. O. Aghenta, M. T. Iqbal, Design and implementation of a low-cost, open source IoT-based SCADA system using ESP32 with OLED, ThingsBoard and MQTT protocol, AIMS Electronics and Electrical Engineering 4 (2019) 57–86. doi:10.3934/ElectrEng.2020.1.57.
[23] L. T. De Paolis, V. De Luca, R. Paiano, Sensor data collection and analytics with thingsboard and spark streaming, in: EESMS 2018 - Environmental, Energy, and Structural Monitoring Systems, Proceedings, 2018, p. 1 – 6. doi:10.1109/EESMS.2018.8405822.
[24] M. Casillo, F. Colace, M. De Santo, A. Lorusso, R. Mosca, D. Santaniello, VIOT_Lab: A Virtual Remote Laboratory for Internet of Things Based on ThingsBoard Platform, in: 2021 IEEE Frontiers

in Education Conference (FIE), 2021, pp. 1–6. doi:10.1109/FIE49875.2021.9637317.
[25] E. Okhovat, M. Bauer, Monitoring the Smart City Sensor Data Using Thingsboard and Node-Red, in: 2021 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/IOP/SCI), 2021, pp. 425–432. doi:10.1109/SWC50871.2021.00064.
[26] D. N. Bestari, A. Wibowo, An IoT-Based Real-Time Weather Monitoring System Using Telegram Bot and Thingsboard Platform, International Journal of Interactive Mobile Technologies 17 (2023) 4–19. doi:10.3991/ijim.v17i06.34129.
[27] A. C. Sabuncu, K. A. Thornton, Leveraging ThingsBoard IoT Service for Remote Experimentation, in: ASEE Annual Conference and Exposition, Conference Proceedings, American Society for Engineering Education, 2022. URL: https://peer.asee.org/leveraging-thingsboard-iot-service-for-remote-experimentation.pdf.
[28] S. I. Jang, J. Y. Kim, A. Iskakov, M. Fatih Demirci, K. S. Wong, Y. J. Kim, M. H. Kim, Blockchain Based Authentication Method for ThingsBoard, in: J. J. Park, S. J. Fong, Y. Pan, Y. Sung (Eds.), Advances in Computer Science and Ubiquitous Computing, Springer, Singapore, 2021, pp. 471–479. doi:10.1007/978-981-15-9343-7_65.
[29] T. M. Kadarina, R. Priambodo, Monitoring heart rate and SpO2 using Thingsboard IoT platform for mother and child preventive healthcare, IOP Conference Series: Materials Science and Engineering 453 (2018) 012028. doi:10.1088/1757-899X/453/1/012028.
[30] JSON, JSON.org. Introducing JSON, 2001. URL: https://www.json.org/json-en.html.
[31] Google, Protocol Buffers, 2023. URL: https://developers.google.com/protocol-buffers.
[32] ThingsBoard, ThingsBoard IoT Platform, 2016. URL: https://thingsboard.io.
[33] Efento, Efento NB-IoT sensors and ThingsBoard, 2024. URL: https://getefento.com/library/efento-nb-iot-sensors-and-things-board/.
[34] ThingsBoard, Device Profiles, 2023. URL: https://thingsboard.io/docs/user-guide/device-profiles/.