Modeling Urban Digital Twins over the Cloud-to-Thing Continuum

Nathalie Moreno[1,*], Lorenzo Toro-Gálvez[1], Javier Troya[1] and Carlos Canal[1]

[1]ITIS Software, Universidad de Málaga, Spain

Abstract

The interest in the development of smart cities and the proliferation of digital twins as a powerful information processing technology have lead to the definition of Urban Digital Twins (UDTs). UDTs model city aspects such as transportation, water supply or air pollution, together with their inherent complexity. For addressing this complexity, in this paper we propose the use of models distributed over the Cloud-to-Thing Continuum, in order to take advantage of the flexibility, scalability and computational capacity of digital services through the multiple physical infrastructures offered by this environment. Following this philosophy, we have modelled a distributed digital twin of the public bus service of the city of Malaga and the users who travel on it. We have validated the model against the real system using the USE tool, and used it to perform simulations and inferences.

Keywords

Urban Digital Twin, Cloud-to-thing Continuum, Model-Driven Engineering, Digital Avatar

1. Introduction

Over the last few years there is a increasing interest in the development of smart cities. These are complex entities composed of myriads of elements—infrastructures, humans, computers and other technological components, etc.—interacting among themselves in disparate manners. The goal of smart cities is to use technology for providing useful services to their citizens and to solve urban problems [1]. So far, smart city applications have been mostly developed in ad-hoc manners, duplicating efforts and costs, and hindering reusability and interoperability [2].

At the same time, Digital Twins (DT) have grown as a powerful information processing technology. A DT is a comprehensive digital representation of a system, employing models and data for representing its properties, conditions, and actions. DTs are continuously updated with real-time information about their physical counterparts [3], facilitating two-way feedback between their digital model and the physical entities represented in it, allowing real-time monitoring, adopting adaptation policies, and enhancing decision-making processes.

In the literature we can find many proposals advocating for applying DT technologies to the development of smart city applications [4]. These Urban Digital Twins (UDTs) model specific city aspects such as transportation, water supply, or air pollution. As citizens are the fundamental key to smart city ecosystems, they must be also taken into consideration. However, their role in UDTs has often been neglected, treating them as a crowd and not considering them as individuals. Instead, we advocate for integrating citizens in the UDT picture, exploiting their information, habits and preferences, and allowing them to have full control over their own data, as well as benefiting from better services from the organisations with whom they share them. With this purpose, we introduced the concept of Digital Avatars (DAs) for representing the virtual profile of a person in collaborative and social computing applications [5].

In addition, conventional implementations of DTs are based on monolithic cloud architectures [6], which face limitations related to scalability, latency, and data privacy and security. These drawbacks are particularly important when modelling UDTs, since a huge amount of information has to be transmitted from the end devices that produce it to the cloud deployment of the UDT. To overcome these limitations, the so-called Cloud-to-Things Continuum [7, 8] (the Continuum, in short) can be used for optimizing information transmission and the way resources are deployed. For this reason, we advocate for a distributed, hierarchical, and low-coupled architecture deployed over the Continuum in which the digital counterpart of each element of the system is deployed on the device which is closest to it, improving the scalability, response time, and privacy of the sensed data. This approach takes advantage of the actual computing architecture of smart cities, where the existing services are distributed over the different layers of the Continuum. In order to show the benefits of the proposed architecture, we have applied it to a case study of smart city

MESS'23: 3rd International Workshop on MDE for Smart IoT Systems, July 19, 2023, Leicester, United Kingdom

*Corresponding author.
nmv@uma.es (N. Moreno); lorenzotoro@uma.es (L. Toro-Gálvez); jtroya@uma.es (J. Troya); carloscanal@uma.es (C. Canal)
0000-0001-9470-0283 (N. Moreno); 0009-0000-0234-6898 (L. Toro-Gálvez); 0000-0002-1314-9694 (J. Troya); 0000-0002-8002-0372 (C. Canal)

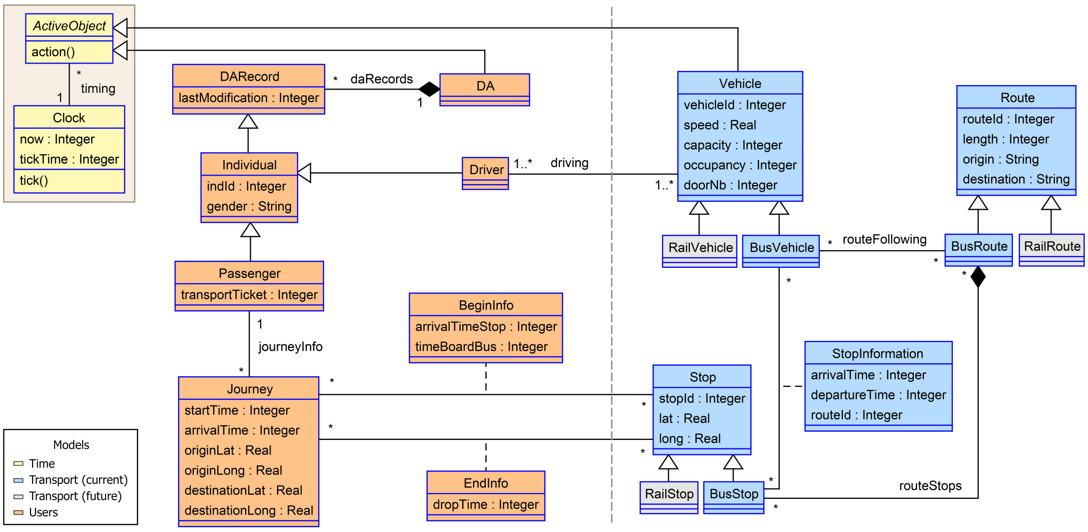

Figure 1: A UML model of a smart city transportation.

transportation in Malaga (Spain), building a UDT which models both its public bus system, and the individuals that make use of it.

The structure of this paper is as follows. After this introduction, section 2 describes how the different elements of the data model of the public transportation system are deployed over the Cloud-to-Thing Continuum, resulting in a loosely coupled and hierarchical architecture of the UDT. Then, section 3 presents the current state of the proposal, describing its implementation and validation, and section 4 describes some related works. Finally, section 5 presents the conclusions of this work, discussing the advantages and limitations of the proposal, and pointing out some future works.

2. Deploying UDTs over the Cloud-to-Thing Continuum

Leveraging the Cloud-to-Thing Continuum infrastructure, we have designed a UDT using high-level UML models. They have allowed us to address the complexity of a urban transportation service, reproducing and simulating at low cost both the behavior of the service and that of its users. The design has been made trying to be scalable in both directions, up and down, so that the system can regulate the workloads that occur at different times.

As shown in Figure 1, the urban transport service of the city of Malaga has a set of buses (`BusVehicle`) that circulate throughout the city. Each of these buses is assigned a specific route (`BusRoute`) to transport passengers (`Passenger`) from a point of `origin` to a point of `destination`. Also, each route consists of a series of stops (`BusStop`) where passengers can board and alight.

In addition to the buses, passengers are another key part of our system. They will have in their smartphone an entity called Digital Avatar (`DA`). This DA is composed of a series of records (`DARecord`) which store information about the activities that a user carries out in the context of their daily life, or in their interaction with certain apps. In our case study, the DA stores the `Passengers`' `Journeys`, recording their `start` and `end` times as well as the GPS coordinates of the `start` and `destination` points of the `Journey`. Passengers take buses in their journeys, and might actually need to take more than one bus to reach their destination. Therefore, the model also collects information about the `Stops` where passengers start (`BeginInfo`) and end (`EndInfo`) the partial legs of their `Journey`. All this information is private but can be shared, with the user's authorization, with other services and devices while preserving the user's identity.

Up to this point, our proposal combines two models: that of the users (in orange in Figure 1), and that of the urban transportation service (in blue). But nothing prevents from extending the model with other data models relevant to the UDT. For example, other transportation systems such as subway (in gray color in Figure 1), or a weather model, as weather conditions have a noticeable influence on the behavior and timing of buses.

Instead of producing a monolithic UDT, we have distributed these models according to the computational capacity of the layers of the Continuum, reducing the problems inherent to a centralized system (delays between recording data and getting results, large storage requirements, etc.). In this distributed design we have brought the data processing as close as possible to the source of the data. Hence, the buses—whose data model and behavior is shown in blue in Figure 1—are located at the fog layer, acting as a gateway between the devices at the edge and cloud layers. This way, we endow each bus with the ability to store, process, and filter relevant data that is sent either to the cloud for further analysis, or to the specific users that may be affected by its behavior (i.e., those riding the bus or waiting at the forthcoming bus stops).

Furthermore, we have deployed the user model on the edge layer. Private data produced by the use of applications or the mobile device itself are consumed by the DA. With the user's prior authorization, some of these data can be shared with other devices and services (at fog and cloud layers) while preserving the user's identity. This involves some interaction between the edge and fog layers as shown in Figure 1. Finally, we have deployed in the mist layer devices with little or no processing and/or storage capacity, but which are also relevant for the proper functioning of the UDT. We will describe them in more detail in the next section.

3. Implementation and Validation

As previously discussed, the focus or our study is public transportation networks in smart cities, particularly in Malaga (Spain). We have at our disposal information updated every minute regarding the city's bus utilisation, including data about routes, stops, timetables, and the GPS location of buses. This information is provided by the City Council as open data[1], and it plays a critical role within the model. The bus routes and stops build the primary infrastructure of the network, acting as the fundamental pillar that sustains the entire system. The GPS locations of the buses, updated every minute, introduce dynamism to the system, offering updates on the buses' positions and movements throughout the city.

Currently, the mobile application provided by the transportation company in Malaga lacks certain information that our UDT can infer. For instance, the company does not know where its passengers will get off, a detail our UDT can infer from the habits stored in the DA. The company also sends mass notifications to all users during incidents, regardless of their relevance to individual users. Our UDT can provide more targeted notifications based on real-time data. To incorporate user data, we rely on smartphones equipped with GPS and Bluetooth, and the DA that represents the user. Bluetooth beacons are installed at stops and on buses, allowing the DA to detect when a user arrives at a stop, gets on, or gets off the bus. The system also verifies that the user is moving as the bus moves, using the smartphone's GPS and the bus location.

We have been collecting data from the transportation system for several months, focusing on a specific bus line (line 11). This line traverses the city, covering 66 stops in total. We currently have over 500,000 documents stored in a MongoDB Atlas database, occupying more than 30 MB, which record the locations of buses within this line every minute.

We have modelled the system with USE [9], which provides a number of tools for analyzing UML models, including model validation, instance generation, and invariant checking. The behavioral aspects of the system are specified in SOIL, an executable language available in USE. The model[2] calculates average times between stops depending on the day of the week and time.

Estimating precisely the stop time is challenging due to the minute-by-minute snapshots of the buses and the fact that a bus does not stop at a station if there are no passengers to get on or off. This variability in travel times can affect the accuracy of the model's predictions, so it is something the model needs to take into account.

To address these challenges, we are currently validating the model. Validation provides insights into the accuracy and reliability of our predictions at a statistical level. It also helps us identify the factors that contribute to significant deviations from the predicted data. Long-term predictions have proven to be more accurate than short-term ones due to various influencing factors, such as traffic jams or road closures that affect short-term predictions more significantly.

Moreover, validation allows us to refine the model by incorporating variables such as traffic patterns, weather conditions, special events, or road closures. These can significantly influence bus movements and arrival times. By continuously validating the model with real-world data and adjusting for these variables, we aim to improve its predictive capabilities and provide more reliable and accurate information to the users of the public transportation service.

4. Related Work

The literature contains numerous proposals that are related to the contribution made in this paper. We discuss them in this section from different perspectives.

4.1. Model-Based Digital Twins

There are mainly two main categories of digital twins: data-based digital twins and model-based digital twins

[1] https://datosabiertos.malaga.eu/group/transporte

[2] https://github.com/atenearesearchgroup/public-transport-system-dt

[10]. The latter aims to represent the dynamic and static behavior of the system based on its known physical characteristics, in addition to reducing the management and operation risk by considering a larger number of uncertainties. Model-based digital twins need to connect the (usually) physical models of the subsystems, which can be a great challenge in complex and multidisciplinary systems.

Current integration efforts by the research community in model-based software engineering (MBSE) and DT development have so far shown that the benefits outweigh their challenges [11, 12]. Most proposals found in the literature advocate the use of a domain-specific modeling languages either in the design phase of the digital twin or in modeling its interaction with the physical system. For example, in [13, 14] the authors present fully model-driven methods for describing the cyber-physical system software, its digital twin information system, and its integration. The integration method in the cited references is based on MontiArc models. Our approach, on the contrary, advocates the use of UML in all phases both in the design of the digital twin and in the simulation of its behavior in the context of the USE tool. High-level UML models can be seamlessly plugged into a DT framework and used not only to specify a DT, but also to define tests for them according to different levels of abstraction.

4.2. Digital Twins over the Cloud-to-Thing Continuum

In the field of digital twin deployment in the continuum, limited research exists, but we consider two proposals here. The authors in [15] propose a distributed digital twin architecture that combines edge and cloud computing for real-time shop floor monitoring. They handle large manufacturing data volumes by filtering and analyzing data at the edge, while achieving long-term storage and collaboration through cloud integration. On the other hand, Constantini et al. [16] introduce IoTwins, a project focusing on distributed DTs in the IoT-edge-cloud continuum. IoTwins enables the development and deployment of DTs at multiple infrastructure levels, including data collection, processing, analytics and ML.

While these works deal with smart manufacturing and distribute different components over separate layers, our distribution, which is specifically focused on the context of public transportation, aims to bring data processing closer to the data source, reduce delays, and address storage requirements. It involves deploying buses at the fog layer as gateways, user models on the edge layer for consuming private data, and devices with little or no processing or storage capacity in the mist layer.

4.3. Transportation Digital Twins

Technologies related to digital twins are maturing to offer intelligent solutions in the transportation sector. In [17] the authors present BODIT, a DT for the urban public bus transportation system in the city of Badalona based on the use of genetic algorithms. BODIT uses the SUMO simulator to realistically reproduce the city traffic. To do this, the simulator requires a large number of configuration parameters, including: the number of vehicles, their type, their origin, their destination, the route they follow or information on the actual and expected arrival times of the buses at each stop. It also incorporates traffic data to make simulations even more realistic. Although the proposal is initially very close to ours, BODIT does not take advantage of a dynamic distributed model deployment approach as we do in the Continuum. Instead, its deployment is centralized and entirely static (decided and fixed at deployment time).

One of the applications of DT technology in transportation is for driving assistance systems. The authors in [18] developed a DT architecture for cloud-based cyber-physical systems, which was used to create a prototype for an advanced driver assistance system. Similarly, Wang et al. [19] also developed a vehicle-to-cloud-based advanced driver assistance system using DT technology. In both cases all the computation is performed on the cloud server, while in our proposal the computation is mostly performed on the same node that generates the data, thus decreasing processing times and data latency.

5. Conclusions

In this paper we have described how UML models and their simulation in the USE tool are suitable for the specification of Urban Digital Twins in order to verify their expected behavior with a very low computational cost. The models thus designed have been deployed in practice by taking advantage of the computational resources offered by each layer of the Continuum. In this way we move from a centralized cloud service (starting point of our case study) to a distributed architecture where data processing is performed close to the node that produces the data.

The distributed deployment we propose allows balancing the workload of the service currently managed by the Malaga bus company: (a) each bus checks if it meets the expectations set by the company (i.e., the bus arrives without delay at each stop at the estimated time); (b) in case of delay or incident, a bus can conveniently notify its passengers, both those on the bus and those waiting at the forthcoming stops. This facilitates personalised communications.

We are currently completing the validation of our model as far as buses are concerned and analyzing the causes that lead to deviations between estimated and actual behaviors. This suggests some lines of future work. First, we plan to feed our model with relevant information that very directly affects the response times of buses (e.g., city holidays, occurrence of major events, weather conditions, etc.). Second, we still need to incorporate user behavior and activity into the simulated model. Then, we will be able to predict data such as the arrival time at a destination or the future occupancy of the buses. Finally, we plan to use the information provided by users' digital avatars to refine predictions and provide personalized recommendations.

Acknowledgments

We would like to thank the reviewers for their insightful comments and constructive suggestions, which helped us to improve the paper. This work was partially funded by the Spanish Government (FEDER/Ministerio de Ciencia e Innovación) under projects PID2021-125527NB-I00 and TED2021-130523B-I00.

References

[1] M. Batty, K. Axhausen, F. Giannotti, et al., Smart cities of the future, The European Physical Journal Special Topics 214 (2012) 481––518. doi:10.1140/epjst/e2012-01703-3.

[2] J. J. Peralta Abadía, C. Walther, A. Osman, K. Smarsly, A systematic survey of internet of things frameworks for smart city applications, Sustainable Cities and Society 83 (2022) 103949. doi:10.1016/j.scs.2022.103949.

[3] A. M. Madni, C. C. Madni, S. D. Lucero, Leveraging Digital Twin Technology in Model-Based Systems Engineering, Systems 7 (2019) 7.

[4] T. Deng, K. Zhang, Z. Shen, A systematic review of a digital twin city: A new pattern of urban governance toward smart cities, Journal of Management Science and Engineering 6 (2021) 125–134.

[5] A. Pérez-Vereda, C. Canal, E. Pimentel, Modelling digital avatars: A tuple space approach, Science of Computer Programming 203 (2021) 102583.

[6] S. Laso, L. Toro-Gálvez, J. Berrocal, C. Canal, J. M. Murillo, Deploying Digital Twins over the Cloud-to-Thing Continuum, (submitted) (2023).

[7] T. Lynn, J. G. Mooney, B. Lee, P. T. Endo, The Cloud-to-Thing Continuum: Opportunities and Challenges in Cloud, Fog and Edge Computing, 2020.

[8] S. Laso, J. Berrocal, P. Fernandez, J. M. García, J. Garcia-Alonso, J. M. Murillo, A. Ruiz-Cortés, S. Dustdar, Elastic data analytics for the cloud-to-things continuum, IEEE IC 26 (2022) 42–49.

[9] M. Gogolla, F. Büttner, M. Richters, Use: A uml-based specification environment for validating uml and ocl, Science of Computer Programming (2007) 27–34.

[10] F. Jaensch, A. Csiszar, C. Scheifele, A. Verl, Digital twins of manufacturing systems as a base for machine learning, in: 2018 25th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), 2018, pp. 1–6. doi:10.1109/M2VIP.2018.8600844.

[11] F. Bordeleau, B. Combemale, R. Eramo, M. Brand, M. Wimmer, Towards Model-Driven Digital Twin Engineering: Current Opportunities and Future Challenges, 2020, pp. 43–54. doi:10.1007/978-3-030-58167-1_4.

[12] A. M. Madni, C. C. Madni, S. D. Lucero, Leveraging digital twin technology in model-based systems engineering, Syst. 7 (2019) 7.

[13] J. C. Kirchhof, J. Michael, B. Rumpe, S. Varga, A. Wortmann, Model-driven digital twin construction: synthesizing the integration of cyber-physical systems with their information systems, Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems (2020).

[14] M. Dalibor, J. Michael, B. Rumpe, S. Varga, A. Wortmann, Towards a model-driven architecture for interactive digital twin cockpits, in: International Conference on Conceptual Modeling, 2020.

[15] N. Ouahabi, A. Chebak, M. Zegrari, K. Kamach, M. Berquedich, A Distributed Digital Twin Architecture for Shop Floor Monitoring Based on Edge-Cloud Collaboration, 2021, pp. 72–78.

[16] A. Costantini, G. Di Modica, J. C. Ahouangonou, D. C. Duma, B. Martelli, M. Galletti, M. Antonacci, D. Nehls, P. Bellavista, C. Delamarre, D. Cesini, IoTwins: Toward Implementation of Distributed Digital Twins in Industry 4.0 Settings, Computers 11 (2022).

[17] P. Ruiz, M. Seredynski, Á. Torné, B. Dorronsoro, A digital twin for bus operation in public urban transportation systems, in: C.-H. Hsu, M. Xu, H. Cao, H. Baghban, A. B. M. Shawkat Ali (Eds.), Big Data Intelligence and Computing, Springer Nature Singapore, Singapore, 2023, pp. 40–52.

[18] K. M. Alam, A. El Saddik, C2ps: A digital twin architecture reference model for the cloud-based cyber-physical systems, IEEE Access 5 (2017) 2050–2062. doi:10.1109/ACCESS.2017.2657006.

[19] Z. Wang, X. Liao, X. Zhao, K. Han, P. Tiwari, M. J. Barth, G. Wu, A digital twin paradigm: Vehicle-to-cloud based advanced driver assistance systems, in: 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), 2020, pp. 1–6. doi:10.1109/VTC2020-Spring48590.2020.9128938.