

Towards using Microservices for Transportation Management: The New TMS Development Trend

Sabah Mohammed ¹, Jinan Fiaidhi ², and Mincong Tang ²

¹Lakehead University

²Affiliation not available

October 30, 2023

Abstract

This is a pre-print for the submission to IEEE LISS 2020 Conference, Budapest, Hungary.

Towards using Microservices for Transportation Management: The New TMS Development Trend

Sabah Mohammed

Department of Computer Science
Lakehead University
Thunder Bay, CANADA
sabah.mohammed@lakeheadu.ca

Jinan Fiaidhi

Department of Computer Science
Lakehead University
Thunder Bay, CANADA
jfiaidhi@lakeheadu.ca

Mincong Tang

School of Economic and
Management
Beijing Jiaotong University
Beijing, China
mincong@bjtu.edu.cn

Abstract— The growth of today's cities and the increased population mobility are providing great challenge to manage vehicles on the roads. This challenge led to the need for new and innovative traffic management, including the mitigation of road congestion, accidents, and air pollution as well as many business oriented demands. Over the last decade, researchers have been focusing their efforts on leveraging the recent advances in Web Services and Multi-Agents to design new road traffic management systems (TMS) for resolving these important challenges in the future transportation. However, these new solutions are still be insufficient and complex to construct TMS systems that are capable of handling the anticipated influx of the population, vehicles and changing transportation scenarios. This paper is pointing to a new and emerging technology that can solve these challenges and develop more flexible TMS systems based on the notion of microservices offered by web frameworks like IFTTT, Zapier, Node-Red and WoTKit.

Keywords— TMS, Web Services, Microservices, Node-red, IFTTT, Zapier, Node-Red, WoTKit

I. THINKING OUT OF THE TMS BOX

Transportation management systems (TMS) are one of the core technologies used in supply chain management (SCM). Traditionally, these systems are available as stand-alone software or as modules within enterprise resource planning (ERP) suites. TMS serves both shippers, carriers and other logistics providers including distributors, wholesalers and retailers. TMSes have gained traction over the past decade as an enabler of global trade and logistics. Gartner, in its March 2019 Magic Quadrant report¹, predicted the global TMS market will grow at an accelerated rate, reaching \$1.94 billion by 2022. The use of TMS systems reduce transportation complexity dramatically by integrating fleet and logistics management throughout the transportation network. Table 1 lists some of the notable TMS systems currently in use.

Table 1: Popular TMS Systems.

TMS System	Website
3Gtms	https://www.3gtms.com/
CEVA Logistics	https://www.cevalogistics.com/
BluJay	https://www.blujaysolutions.com/

Cloud Logistics	https://www.gocloudlogistics.com/
Descartes	https://www.descartes.com/
JDA	https://jda.com/
Kuebix	https://www.kuebix.com/
Manhattan	https://www.manh.com/
MercuryGate	https://www.manh.com/
Oracle	https://www.oracle.com
SAP	https://www.sap.com
TMC	https://www.mytmc.com/
TMW	https://www.tmwsystems.com/
Transplace	https://www.transplace.com/

In principle TMS can benefit companies in numerous ways, including the following:

- Improved Cash Flow
- Minimal Paperwork
- Inventory Management
- Supply Chain Visibility
- Optimized Routing through Pool Distribution
- Tracking Drivers En-Route
- Accurate Order Fulfillment
- Improving Customer Experience
- Reduced distribution and warehouse
- Reduced administrative costs and invoicing errors

However, TMS ecosystem is changing and there is a timely need to incorporate further advancements and updates to the current systems. TMS needs to adapt to the new ecosystem changes to ensure that a shipper is able to succeed in the next decade. Among the new trends are the followings:

- ❖ Adoption of cloud-based system,
- ❖ More load optimization automation

¹ <https://www.gartner.com/en/documents/3905867/magic-quadrant-for-transportation-management-systems>

- ❖ Adoption of open-sourced platforms
- ❖ Enforcement of connected truck using more Internet of Things technologies
- ❖ Adoption of blockchain-Driven Software to encourage partnership
- ❖ Enhancement of Real-Time Supply Chain Management for greater visibility
- ❖ Intuitive Management and Analytics capabilities via integrating more Machine Learning and AI components
- ❖ Enhanced Web Portals

These trends present huge challenges that shippers face in today's hectic freight transportation environment which advocate for more integration, flexibility and high responsiveness. The present remedy to outsource the added value components that cannot be captured by the enterprise TMS to third party vendors and use sound third party logistics (3PL)² software to manage dealing with the third party partners. The 3PL software is required to have the ability to integrate with the major e-commerce platforms as well as satisfying the following constraints:

- Manage contracts and service level agreements for SCM outsourcing
- Provide features for multi-warehouse and inventory management
- Deliver functionality for transportation management and shipping
- Track the costs (estimate and actual) of all outsourced supply chain activities
- Maintain a repository of providers of supply chain operations
- Allow customers to access information such as inventory availability
- Include metrics that customers can use to track performance

However, selecting a 3PL software to compliment or integrate with the shipper TMS requires careful thought and great expertise. Integrating the 3PL with TMS/ERP systems can bring enormous efficiencies. If these integrations are out-of-the-box, they can be implemented quickly so that we can see the return of investment (ROI) immediately. By integrating the 3PL and TMS/ERP, we'll reduce the need to re-enter information between systems and facilitate the rapid creation of shipments with correct documentation.

Moreover, this integration should also be able to scale with the business when it needs change. All the above constraints increase the complexity of having modern transportation ecosystem that is inherently a network-based business process. It involves an ecosystem of different parties — a community of shippers, carriers, consignees, brokers, and others that need to communicate and collaborate with each other in order to transport products and utilize assets and labor as efficiently as possible. This complexity and fragmentation makes it challenging to quickly and efficiently match transportation demand with available vendor capacity of whatever size. This growing need in the market for better matching of supply and demand, coupled with the rise of cloud computing, web services, software intelligent agents, software-as-a-service (SaaS), modern application

programming interfaces (APIs), and other emerging technologies including microservices, is driving the next evolution of transportation management systems [1]. This ample demand for technology in supply chain management and rapidly growing supply of TMS/3PL tools aiming to improve the industry, have failed to connect the two in a way which truly adds value. This paper aims to guide logistics professionals open to innovation driven by technology but eager to avoid common pitfalls.

II. INNOVATION IN THE TRANSPORT LOGISTICS

Innovation has historically played a vital role in increasing efficiency. However, while other industry sectors have experienced rapid growth of productivity, the transport logistics industry has seen relatively small improvement in terms of efficiency [2]. Increasing transport logistics efficiency through identifying the right innovation technologies can save an enormous amount of money, hence is worthy of study. Notable attempts based on important innovations were based on *web services* [3] to develop a freight brokering system. Web services are self-contained, Web-enabled applications capable not only of performing business activities on their own, but also possessing the ability to engage other Web services in order to complete higher-order business transactions. Developing freight brokering system is an attempt to match the logistic capabilities of transportation providers with the needs of merchandise owners. From a service science perspective, the clients and transporters interacting through the matchmaking system create mutual value by minimizing client costs, maximizing business opportunities, increasing competition, and reducing operational risks. Shifting the dominant thinking of supply chain management toward the concepts of web services ecosystems opens up many research opportunities and strategies for improved organizational performance [4]. However, the key to solving the problem of efficient management of logistics information is centered on building a smart logistics services supply system. In this direction several researchers proposed a *multi-agent system* for smart brokering of logistics web services (e.g. allowing shippers to collaborate during the logistics process, thus improving the overall system intelligence; the use of agent-based negotiation for logistics management). As an example, in [5] the authors have proposed composing web services using coalitions of agents performing tasks for service requestors.

Although the use of web services and multi-agent technologies in designing smart brokering logistics is motivated by the interactivity, interoperability, responsiveness and social characteristics of this domain, these technologies use complex development frameworks that use an Extensible Markup Language (XML)-based service description for the description of each web service. Such a description covers all the necessary details needed in order to interact with the services, including message formats, transport protocols, and methods. Web services share Universal Description and Ontologies, Discovery and Integration (UDDI), which is a centralized service directory, for its service discovery. Multi-agents technology on the other hand provide capabilities of autonomy, social ability, reactivity, and the ability to be proactive. Developing a multi-agent system requires that each functional unit (e.g., the firm)

² <https://www.g2.com/categories/3pl>

to be populated by a number of heterogeneous intelligent agents with diverse goals and capabilities. Each agent, then, is designed to represent a specific functional unit. The requirements for the action strategies and policies to be used may be entered into the agent beforehand. Different levels of collaborative requirement can be easily incorporated into the agent as different goals are made based on different types of scenario. Multi-agents are very effective in addressing both coordination and conflicts among the firms. Thus multi-agents requires to have demand responsive components that provide a share transportation services with flexible routes and focus on optimizing of economic values. The design of such demand responsive components are still in its early stage to become part of the smart TMS systems [6]. Figure 1 illustrates the innovative technologies that has been suggested so far for designing a smart transportation system.

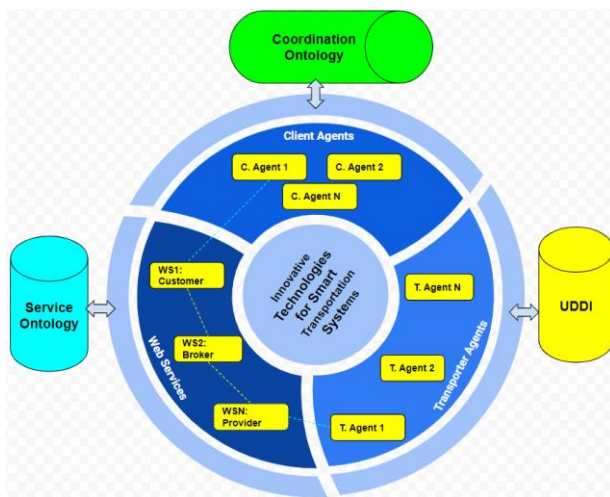


Figure 1: Innovative Technologies for Designing Logistic Transport Brokering System

But the picture is changing rapidly related to the emerging innovations in transport technologies. Recently, new opportunities for developing lightweight transport brokering systems are arising, as Web of Things (WoT) platforms such as IFTTT [7] and Zapier [8] are emerging. These platforms came into service to support flexible composition of applications with various things connected to the Web. A shipper can easily select an application component from a pool of building blocks such as sensor information, actuation functions and data services to create and deploy personalized transport applications. We can reasonably expect more Web applications to be created through such WoT platforms because of the ease of development. These emerging technologies are based on using microservices [9]. Microservices architecture involves breaking down a software application into its smaller components, rather than just having one large software application. Typically, this involves splitting up a software application into smaller workflow components. These workflows can then integrate to each other via an interface.

III. TOWARDS MICROSERVICE BASED TRASPORT SYSTEMS

The execution pattern of a microservice application can be described as a time sequence of Web service invocations. Each web service is categorized into either a trigger or an action in WoT architecture [10]. A trigger is either a publication of some information or a signal that an action

(actuation) took place. An action is a task to be executed whenever a trigger is fired. In this approach, instead of producing Apps as a complete runtime artifact, developers can design intermediary artifacts, as accessible controls parts for typical transportation tasks. Using these controls, transporters are enabled to construct their own tasks autonomously by using resources of their own choice from across the WoT. The microservice based transportation system translates the time sequence of trigger and action executions to a time sequence of network flows. A network flow is a traffic information between two communicating endpoints. Zapier and IFTTT or many other alternatives³ makes this task easy. It lets transporters to integrate everyday apps (e.g. existing legacy app or a newly created workflow) and automate the transportation business processes. Figure 2 illustrates the idea behind using WoT platforms.

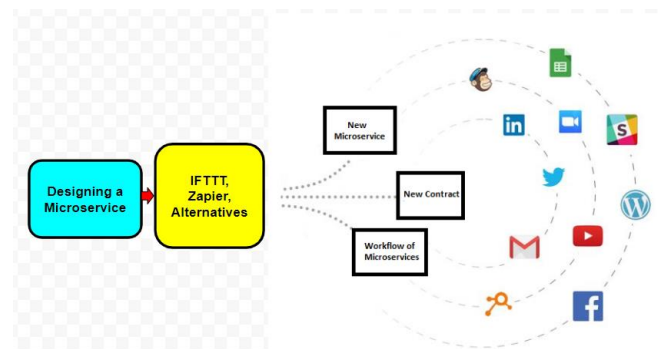
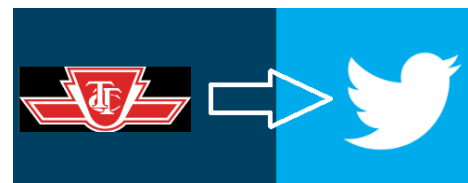


Figure 2: Developing Microservices Based on WoT platforms.

WoT platforms constructs a new microservice by combining Triggers and Actions via completing an action in one app when a trigger occurs in another app. These combos—called “Zaps” according to Zapier terminology or “Applets or Recipes” according to IFTTT terminology— will complete the workflow tasks automatically. Using the IFTTT platform, the "IF THIS" keyword is used to identify the triggers and "THEN THAT" is used to identify the actions part. The trigger is activated by changes that occur within other web services such as TTC⁴ (Toronto Transit Alert). Upon activation you may assign an action like pushing a notification about the transportation alert to your twitter account:



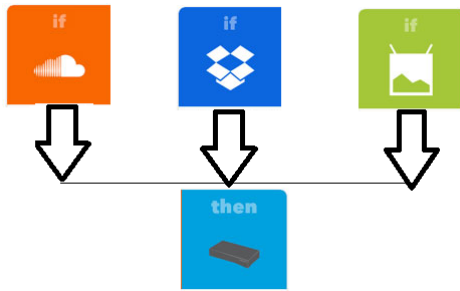
WoT platform may create a new transportation contract upon firing a new microservice. For example log your completed Uber trips in a spreadsheet:

³ <https://www.gadgetgeek.com/zapier-alternatives-open-source/>

⁴ https://www.ttc.ca/Service_Advisories/all_service_alerts.jsp



Moreover, we can add branching logic to microservices to run different actions based on different conditions (e.g. if A happens in your trigger app, then do X. If B happens, then do Y, and so on). Tools like Paths⁵ can be used with Zapier to add such branching logic and produce a workflow of microservices (e.g. Case downloads are either songs, normal files or Android images Then save to the Seagate Personal Cloud):



WoT platforms like IFTTT or Zapier, however, may represent decent workflow composition tool to get started on simple point-to-point integrations but what should you do when you want to grow your organization and tackle the complexity that often arises when doing so? Fast-growing transportation companies and enterprises need to incorporate higher level frameworks to accommodate the various complex scenarios. In this direction we can use more sophisticated workflow wrapper like the WoTKit⁶ and Node-RED⁷. With these wrappers one can produce fully customizable workflows with flexible connector operators such as loops, data storage, array mapping, branching, and if/then conditionals and many more. Node-RED is a programming tool used for wiring together hardware devices, APIs and online services by representing them as nodes. It has been recently used in designing a smart TMS system [11]. The advantages that render the uniqueness and innovation of Node-RED, as a programming tool, is its ability to amalgamate a widely-used programming language under a new flow-based programming model along with a large list of software and hardware components, retaining at the same time, an open-source character. In addition, the Node-RED UI offers a user-friendly way for creating and inserting new nodes with drag and drop – like methods. As a result, adding and connecting new components becomes trivial. It's obvious that an already implemented node can be reused in the same or another flow or even saved for future usage. Flows can be open-sourced or sold, enabling developers this way to focus on the innovative part of their task saving time on developing already implemented flows. For example if we want a Node-RED TMS system to have the capability to track the location of their transportation vehicles, then we can transmit the driver mobile phone location via the OwnTracks⁸ microservice to the Node-RED server. We can also visualize the vehicle position using

worldmap⁹ microservice. We can connect also ingest data from the OwnTracks microservice with geofence¹⁰ microservice to create geofence based triggers. This added microservice is to create automated actions that get triggered when trackers leave or enter areas (for example the vehicle leaving the town while the driver phone is at home). Figure 3 illustrate the composite Node-Red workflow of this application [12].

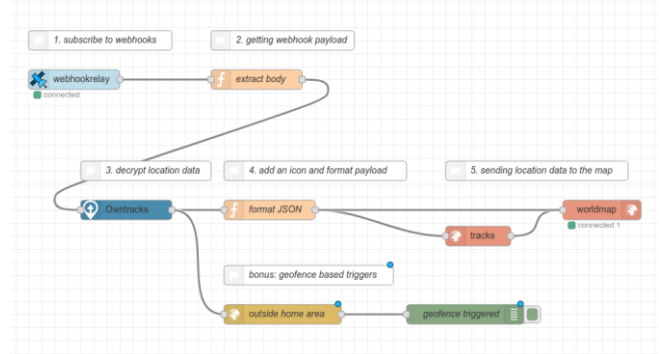


Figure 3: Node-Red Microservices Workflow for Trucking Drivers and Vehicles.

The overall architecture of TMS systems based on the notion of microservices can be represented in Figure 4.

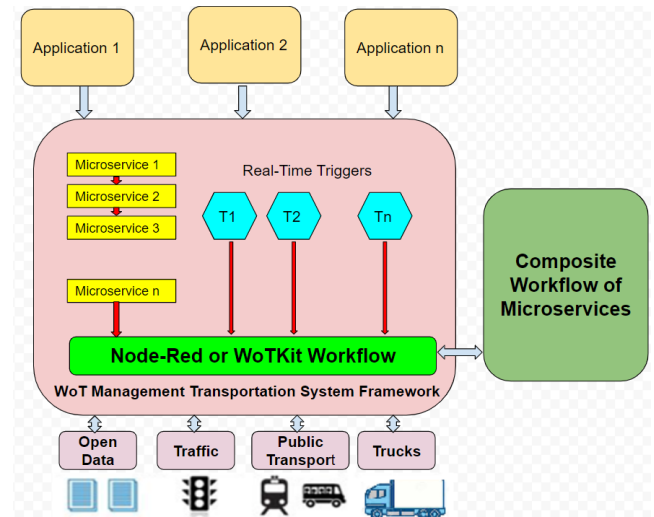


Figure 4: The Envisioned WoT MTS Architecture.

IV. CONCLUSIONS

It is becoming more and more pervasive, modern web applications and platforms to consume or provide data and new functionalities as web services through APIs which you can access with simple calls, like an HTTP request. WoT frameworks, following this trend in much of its architecture, consists of autonomous components which are connected with each other by exposing their services in this way. So, instead of presenting raw data and other specialized services, a WoT platform is flexible enough to allow 3rd parties to develop complex transport microservice applications through the provision of frameworks like IFTTT, Zapier, Node-RED or WoTKit, can be a great asset for the modern MTS infrastructure. This paper illustrate this new trend in developing transportation management system based on the

⁵ <https://zapier.com/paths>

⁶ <https://wotkit.readthedocs.io/en/latest/>

⁷ <https://nodered.org/>

⁸ <https://owntracks.org/>

⁹ <https://flows.nodered.org/node/node-red-contrib-web-worldmap>

¹⁰ <https://flows.nodered.org/node/node-red-node-geofence>

notion of microservices. This research is an ongoing research work in progress to develop innovative transportation system that are flexible enough to respond to the current demand and technology change.

ACKNOWLEDGMENT

This research is kind of extracts from the NSERC Discovery Grants 2020 Proposals of the first two authors. It has been submitted to the IEEE LISS 2020 July 25-28, 2020 Budapest, Hungary.

REFERENCES

- [1] Harris, Irina, Yingli Wang, and Haiyang Wang. "ICT in multimodal transport and technological trends: Unleashing potential for the future." *International Journal of Production Economics* 159 (2015): 88-103.
- [2] Ciprés, David, Lorena Polo, and Alberto Capella. "Innovation in Transport Logistics—Best Practices from the EU Project LOGINN." In *Dynamics in Logistics*, pp. 599-608. Springer, Cham, 2016.
- [3] Ji, C., Li, M. and Li, L., 2004, September. Freight transportation system based on Web service. In *IEEE International Conference on Services Computing*, 2004.(SCC 2004). Proceedings. 2004 (pp. 567-570).
- [4] Lusch, R.F., 2011. Reframing supply chain management: a service - dominant logic perspective. *Journal of supply chain management*, 47(1), pp.14-18.
- [5] Luncean, Lucian, Costin Bădică, and Amelia Bădică. "Agent-based system for brokering of logistics services—initial report." In *Asian Conference on Intelligent Information and Database Systems*, pp. 485-494. Springer, Cham, 2014.
- [6] Satunin, Sergey, and Eduard Babkin. "A multi-agent approach to Intelligent Transportation Systems modeling with combinatorial auctions." *Expert Systems with Applications* 41, no. 15 (2014): 6622-6633.
- [7] Ovadia S. Automate the internet with "if this then that"(IFTTT). *Behavioral & Social Sciences Librarian*. 2014;33(4):208–211.
- [8] Finch M. Using Zapier with Trello for electronic resources troubleshooting Workflow. *The Code4Lib Journal*. 2014;26.
- [9] Newman, Sam. Building microservices: designing fine-grained systems. "O'Reilly Media, Inc.", 2015.
- [10] Yoon, Young, Hyunwoo Jung, and Hana Lee. "Abnormal network flow detection based on application execution patterns from Web of Things (WoT) platforms." *PloS one* 13, no. 1 (2018): e0191083.
- [11] Sicari, Sabrina, Alessandra Rizzardi, and Alberto Coen - Porisini. "Smart transport and logistics: A Node - RED implementation." *Internet Technology Letters* 2, no. 2 (2019): e88.

Jinan Fiaidhi



Dr. Fiaidhi is a full professor of Computer Science and the Graduate Coordinator of the Biotechnology PhD program at Lakehead University. She is an adjunct research professor at the University of Western Ontario and the editor in chief of IGI Global International Journal of Extreme Automation and Connectivity in Healthcare. She is also the chair of Big Data for eHealth with the IEEE ComSoc. Contact her at jfiaidhi@lakeheadu.ca.

Sabah Mohammed



Dr. Mohammed is a full professor of Computer Science and Professional Software Engineer. He is an adjunct research professor at the University of Western Ontario and the editor in chief of IGI Global International Journal of Extreme Automation and Connectivity in Healthcare. He is also the chair of Smart and Connected Health with the IEEE ComSoc. Contact him at Sabah.mohammed@lakeheadu.ca.

Mincong Tang



Dr. Mincong Tang graduated his MS from Beijing Jiaotong University, China (2005) and PhD degree in Management Information Systems at The Chinese University of Hong Kong (2011). He had been the Executive Director of International Center for Informatics Research (ICIR) from October 2012 to present at Beijing Jiaotong University. He is also a research fellow in the International Center for Informatics Research (ICIR) of the same university from November 2012 to present.

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.