Section 4: Week 7: Smart Cities

Nate Bachmeier

TIM-7010: Computer Networking and Mobile Computing

August 11th, 2019

North Central University

# Table of Contents

[Smart Cities 2](#_Toc16712008)

[Problem Statement 2](#_Toc16712009)

[Goals 3](#_Toc16712010)

[Relevance and Significance 3](#_Toc16712011)

[Literature Review 4](#_Toc16712012)

[Reasoning about Smart Cities (2018) 4](#_Toc16712013)

[IoT Smart City Architectures (2018) 4](#_Toc16712014)

[The success of IoT in Smart Cities of India (2018) 5](#_Toc16712015)

[Relationship Between Smart Cities, Policing and Criminal Investigation (2018) 5](#_Toc16712016)

[Smart Cities: an IoT-centric Approach (2014) 5](#_Toc16712017)

[NDN IoT Content Distribution Model (2016) 6](#_Toc16712018)

[IoT over SDN for Smart City Applications (2016) 7](#_Toc16712019)

[Software-Defined Networks: State of the Art and Research Challenges (2014) 7](#_Toc16712020)

[Using PBR for Security Protections 8](#_Toc16712021)

[DURE: An Energy- and Resource-Efficient TCAM Architecture (2019) 9](#_Toc16712022)

[Approach 10](#_Toc16712023)

[References 12](#_Toc16712024)

# Smart Cities

The Internet of Things (IoT) attempts to widen the interconnectivity of computers to include interconnectivity of objects (Commission of the European Communities, 2009). These objects expose sensors that can aggregate into personalized data feeds.

These objects can share a contextual domain, such as a home, warehouse, or city to form smart spaces. As the scope of these smart spaces grows, so does the number of user scenarios that can be enhanced.

Developers can harness that data emitted from those scenarios to make intelligent recommendations and provide guidance around optimizations and safety decisions. These capabilities delight the inhabitants and encourage them to interact with more objects, continuing the cycle.

## Problem Statement

Are the core of these user scenarios are policy-based routing, implemented on OpenFlow switches. These policies control the behavior of people and devices as they move around the physical environment. They also project a virtual environment where core aspects, such as identity, data discovery, and QoS can exist.

As the scope of the smart space increases, so does the number of devices and the permutations of their interactions. OpenFlow policy tables store the authorization and routing decisions for those interactions. Because of this design choice, the innovation of future smart cities cannot exceed the limits of the OpenFlow policy tables.

OpenFlow policy tables are dependent on Ternary Content-Addressable Memory (TCAM), a specialized type of computer memory. While Random Access Memory (RAM) translates an address to content, TCAM has the opposite requirement.

Without efficient alternatives, excessive hardware is needed and can make the adoption of smart cities prohibitively expensive. Locations that are willing to accept these costs are limited in the granularity of their policies, which directly stifles innovation.

## Goals

The state of the art alternative to TCAM is the use of Field-Programmable Gate Array (FPGA) circuits (Ullah, Ullah, Afzaal, & Lee, 2019). Ullah et al. propose a mechanism to emulate TCAM by dynamically modifying the logic gates on FPGA chips.

The next steps are to reproduce the results and their feasibility on a larger scale. Those simulations are likely to highlight changes to their reprogramming algorithm. Their paper points out that specific changes to the logic gates can have enormous performance penalities. During these update operations, sections of the circuit become locked and cannot process traffic until those changes are committed.

An investigation into mechanisms for reducing the need for updates and their associated locks is critical for this alternative solution.

## Relevance and Significance

Several metropolitan areas across the globe have already begun the transition toward smart cities. On that journey, the users consistently discover more insights using more devices improves their quality of life. These improvements inspire developers to implement cross-cutting concerns, such as identity and augmented reality aspects, for their cities.

These aspects promote exponential growth in connectivity as developers build upon one another’s work. Administrators need to manage that secure environment for that trend to continue. That continuation relies on the routing subsystems supporting granular policies.

# Literature Review

## Reasoning about Smart Cities (2018)

Balducci et al. define a smart city as a system of systems, including instances of Cyber-Physical Systems (CPS) and the Internet of Things (Balduccini et al., 2018). They make the argument that the problem space is too ample for a single platform to address every facet of the required design. Because of this characteristic, all smart cities are naturally heterogeneous and must consider interoperability through open standards.

Cross-cutting concerns can be extracted from the problem and separated into distinct aspects (Kiczales, 1997). For instance, the notion of identity can become a centralized service that many devices and users jointly share and trust. The complexity to continue extending the smart city decreases, as more aspects of the system become available.

## IoT Smart City Architectures (2018)

Fahmideh and Zowghi build on this idea that smart cities are collections of connected services and devices. They reviewed nine different reference architectures and looked for commonalities between them. Key differences can be categorized based on which aspects were considered necessary to their designers.

For instance, British designs focus heavily on environmental sustainability solutions versus American Big Tech desired business integration scenarios. Groups, like Open Geospatial Consortium, have deep roots into academia and have extensive capabilities for machine-to-machine yet minimal functionality for users (Fahmideh & Zowghi, 2018).

The diverse collections of aspects provided by the system demonstrate that one size cannot fit all. That uniqueness makes sense, as a *smart city* is an extension of the *physical city* that contains it. Within those physical cities are diverse cultural expectations.

## The success of IoT in Smart Cities of India (2018)

Starting around 2015, the government of India pledged the equivalent of fifteen billion US dollars toward smart cities. Their goal was to purchase one hundred Information and Communication Technology (ICT) locations. They acknowledged that many people were abandoning the villages and moving into urban areas. Those new inhabitants would need access to the Internet and a transition toward purely digital lives (Chatterjee, Kar, & Gupta, 2018).

Chatterjee et al. describe a cycle where people interacted with physical and virtual objects, which in turn generated lots of data. Artificial Intelligence (AI) systems mine the data and propose recommendations. City planning and legislation decisions leverage those recommendations to customize the area to the needs of the people. More virtual and physical objects fill those needs resulting in even more data.

## Relationship Between Smart Cities, Policing and Criminal Investigation (2018)

Users of the smart city have specific roles within their community and need data that improve their effectiveness, and efficiency — police, medical, and fire & rescue teams are prime examples of this scenario. In traditional cities, police officers need to rely on eye witness encounters that might be racially biased (Kaja & Bostjan, 2018).

Instead, safety officials can deploy sensors into high-risk environments and record evidence as the crime unfolds. Systems such as ShotsSpotter can detect gunshots and report the incident to an emergency hotline. Unfortunately, these systems are still aways out, but the technology is progressing (Drange, 2016).

## Smart Cities: an IoT-centric Approach (2014)

As technology progresses, it generates an ever-growing volume of data that needs to be efficiently indexed. Smart cars will need to consume data feeds from traffic lights, road sensors, accident reports, and construction schedules, to name a few. One solution is to decompose smart cities into smart city *hubs*, where a hub is a logical unit such as a shopping district or residential neighborhood (Lea & Blackstock, 2014).

Lea and Blackstock examined the implementation of two smart cities, one in the United Kingdom the other in Canada. With both locations, an efficient data catalog was critical to the project’s success. They attributed this to reducing the learning curve, resulting in more developers creating more applications available.

Over the last five years, several American cities, such as Seattle and New York, have created open data platforms. The easy access to data has enabled dozens of high-quality applications; however, they are specific to that location.

Small development studios are willing to build dedicated apps for a given city, but more prominent corporations are unwilling to enter the space. Their userbase spans the entire country and needs it to be usable on a national level. If they must implement a data adaptor of every metropolitan area, the development cost will be too great to justify. Until businesses, city planners, and hardware manufacturers agree on open standards; there will be challenges sharing the data and making it fully discoverable.

## NDN IoT Content Distribution Model (2016)

An alternative solution to making content discoverable is through Name Data Networks (NDN). Under NDN, publishers create signed content under a namespace, and then consumers can request those resources through standardized names. Metadata about those resources can be cached and served directly from OpenFlow routing tables (Shilton, Burke, Caffy, & Zhang, 2016).

The requested names can be relative and can leverage the shared context of the entities logical and physical features. For example, a smart car can ask for the status of the next stoplight by querying name /local/route/stoplight/2. The software-defined network can response with route forwarding policy to the associated RESTful endpoint.

In effect, the query name functions as a small internal domain-specific language. Integrating legacy systems is straight forward as the NDN maintains metadata and pointers. The caller could also encode their desired format, through additional child path annotations (e.g., stoplight/2/foo+xml).

## IoT over SDN for Smart City Applications (2016)

In addition to the Name Data Network scenarios, software-defined networking is a requirement for smart cities as a mechanism to manage the IoT ecosystem. Users and devices need to move around the space, and they expect the network to adapt to this movement (Ogrodowczyk & Belter, 2016).

These dynamic networking environments cannot rely on simple static routes and instead use Policy-Based Routing.

## Software-Defined Networks: State of the Art and Research Challenges (2014)

The de-facto protocol used for policy-based networking is called OpenFlow (OFP) (Jammal, Singh, Shami, Asal, & Li, 2014). Within OFP network traffic is partitioned into *flows*. A flow can have an arbitrary level of granularity such as a simple web request or all messages destined to a database cluster.

An *OpenFlow Switch* has one or more *flow tables* with each entry in the table mapping *match criteria* with zero or more *actions.* A group table can also exist to associate match criteria with a collection of related flows. The *OpenFlow Channel* is responsible for distributing policy notifications from the *OpenFlow* *Controller* to the *OpenFlow Switch*. This distribution takes place across the *OpenFlow Protocol* (OFP), which standardizes the messages used to describe the policy operations.

## Using PBR for Security Protections

For instance, a policy could specify that incoming traffic for a virtual IP is forward to one or more physical switch adapters. Perhaps one adapter goes to a backend service and the other to an Intrusion Detection and Prevention System (IDS/IPS).

If the traffic is determined to be malicious or misconfigured, then more policy can be created to block that device or provision additional virtual network functions (Lopez, Caraguary, Villalba, & Lopez, 2015).

Akamai Technologies is responsible for the management of Global Content Delivery Networks (CDN); have reported an annualized 60% increase in network attacks (Singh, Singh, & Kumar, 2017). Their figures suggest that network administrators should expect more policy in the future for IDS/IPS scenarios.

**Survey of Concepts for QoS improvements via SDN (2015)**

Several existing systems, such as OpenQoS and FlowQoS, encode the Quality of Service decisions, into policy-based routing tables. For instance, a video streaming service might reserve capacity to remove jittering. Another everyday use case is specifying how traffic to offline hosts needs to be rerouted (Mirchev, 2015).

**From Raw Data to Smart Manufacturing: Things for Industry 4.0 (2018)**

The rise of Industrial Internet of Things (IIoT) and 5G wireless are expected to cause a 1000-fold increase in the number of connected devices (Petel, Ali, & Sheth, 2018) (Frodigh, 2018).This multiplication effect will further grow the size of the policy-based routing tables, introducing additional challenges for scaling software-defined networking systems.

Today, businesses address these limitations through the use of extra hardware. However, as the problem grows to the scale of large enterprises or smart cities, the costs become prohibitively expensive (Jain et al., 2013).

**SDWN Opportunities and Challenges for IoT (2006)**

The success of smart cities relies on capabilities exposed through policy-based routing technologies, such as though provided by Software-Defined Wireless Networking. When running at scale, those needs will use enormous amounts of policy entries in the OpenFlow switches.

An open research area within software-defined networks are mechanisms to increase the supportable size of Policy-Based Routing (PBR) in the Open Flow Tables. Ternary Content-Addressable Memory (TCAM) introduce these limits, as there is a finite amount on each physical networking device (Shood, Yu, & Xiang, 2006).

## DURE: An Energy- and Resource-Efficient TCAM Architecture (2019)

A commodity workstation uses Random Access Memory (RAM) and requires the application to provide an *address* to retrieve the *content*. Network packets entering into the OpenFlow switch have the opposite requirement; the destination’s virtual IP (content) needs mapping to a virtual switch port (address).

Ternary Content-Addressable Memory (TCAM) addresses this requirement by allowing each bit in the content to represent states (a) on, (b) off, or (c) doesn’t care about ‘x.’ In a single clock cycle, these wildcards are applied and queried across the entire routing table (Ullah, Ullah, Afzaal, & Lee, 2019).

A limited amount of TCAM memory is available on each device due to (1) the chips are expensive to produce; (2) requires significant power for complex circuits; (3) the power consumption emits large amounts of heat; and (4) the complex circuitry reduces amount of memory that can be placed per square centimeter.

According to Ullah et al., a typical chip contains on the order of 1000 x 144-bit words. The number of words does not directly map to the number of supported devices. Vendors can implement filter policies using multiple ‘allow’ and ‘drop’ actions requiring additional Flow Table entries.

To partially mitigate the scenario, vendors have introduced the notion of ‘Flow Groups’ as a mechanism to group multiple flows into the same policy entry. However, the needs for more fine-grained control reduces the size of each group. Eventually, this size limitation will impede innovation within smart cities.

# Approach

Ullah et al. propose a TCAM replacement solution based on Field-Programmable Gate Array circuits. They emulate the behavior of TCAM by manipulating the logic gates to produce hardware speeds from a software-based algorithm. This approach shows a lot of promise but needs to confirmation on a more complex environment.

CityGML, IndoorGML, and related open standards exist for describing smart cities and the supported interactions within them (Fahmideh & Zowghi, 2018). A simulation environment could consume these files and then perform (1) Monte Carlo, (2) Genetic Algorithms, and (3) Reinforcement Learning searches. These algorithms would be able to hunt down the most efficient combinations for manipulating the FPGA circuits.

For instance, a reinforcement algorithm could attempt to minimize the number of clock cycles required by randomly taking actions from a predefined list. The training of the model results in a policy map, that defines the expected reward from making decisions based on the current state. After sufficient iterations, the model will converge and is capable of producing very accurate predictions (Fridman, 2019) (Lapan, 2018).

NOX, POX, NodeFlow, and Floodlight are software-defined network controllers that are available as Open Source Solutions (OSS) (Azodolmolky, 2013). Researchers can rapidly build prototypes on these platforms, as there is a clear separation of duties. This separation removes the need to recreate boilerplate code, and instead focus on the task at hand. The reinforcement learning model can be exposed as an OpenSwitch extension on one of these platforms.

Standardized benchmarks exist for MinInet, a utility capable of simulating entire virtual networks on a single laptop. These benchmarks can validate the performance against existing solutions. It would also be advantageous to find or create a test set, similar to LEACH-MEEC that explicitly focuses on short-lived mobile entities (Ahmad, Li, Khan, Khurshid, & Ahmad, 2018).

# References

Ahmad, M., Li, T., Khan, Z., Khurshid, F., & Ahmad, M. (2018). A Novel Connectivity-Based LEACH-MEEC Routing Protocol for Mobile Wireless Sensor Network. *Sensors, Volume 18, Issue 12*, 4278.

Azodolmolky. (2013). *Software-Defined Networking with OpenFlow.*

Balduccini et al. (2018). Reasoning about Smart City. *2018 IEEE International Conference on Smart Computing*, 381-386.

Chatterjee, S., Kar, A., & Gupta, M. (2018). Success of IoT in Smart Cities of India: An empirical analysis. *Government Information Quarterly Volume 35, Issue 3, September 2018*, 349-361.

Commission of the European Communities. (2009). *Internet of Things — An action plan for Europe.* Retrieved from http://eurlex.europa.eu/LexUriServ/site/en/com/2009/com2009\_0278en01.pdf

Drange, M. (2016). Shots Misfired: How Silicon Valley's Answer To Gun Violence Missed The Mark. By: Drange, Matt. *Forbes.com*.

Fahmideh, M., & Zowghi, D. (2018). IoT Smart City Architectures: an Analytical. 709-715.

Fridman, L. (2019, January 24). *MIT 6.S091: Introduction to Deep Reinforcement Learning.* Retrieved from YouTube: https://www.youtube.com/watch?v=zR11FLZ-O9M

Frodigh, M. (2018, May 30). *Live from ICSE: Conference Opening + Keynote from Magnus Frodigh*. Retrieved from YouTube: https://www.youtube.com/watch?v=cpeMmMh7Syk

Jain et al. (2013). B4: Experience with a Globally-Deployed Software-Defined WAN. *SIGCOMM’13, August 12–16, 2013, Hong Kong, China.*

Jammal, M., Singh, t., Shami, A., Asal, R., & Li, Y. (2014). Software-Defined Networking: State of the Art and Research Challenges.

Kaja, P., & Bostjan, S. (2018). Analysis of the Relationship Between Smart Cities, Policing, and Criminal Investigation. *Varstvoslovje: Journal of Criminal Justice & Security. 2018, Vol. 20, Issue 4, p389-413. 25p.*, 389-413.

Kiczales, G. (1997). Aspect-Oriented Programming.

Lapan, M. (2018). *Deep Reinforcement Learning Hands-on.* Packet Pub.

Lea, R., & Blackstock, M. (2014). Smart Cities: an IoT-centric Approach. *IWWISS '14, September 01 - 02 2014, Saint Etienne, France*.

Lopez, L., Caraguary, A., Villalba, L., & Lopez, D. (2015). Trends on virtualization with software-defined networking and network function virtualization. *IET Journals*.

Mirchev, A. (2015). Survey of Concepts for QoS improvements via SDN.

Ogrodowczyk, L., & Belter, B. (2016). IoT ecosystem over programmable SDN. *2016 Fifth European Workshop on Software-Defined Networks*, 49-51.

Petel, P., Ali, M., & Sheth, A. (2018). From Raw Data to Smart Manufacturing: AI and Semantic Web of Things for Industry 4.0. *IEEE INTELLIGENT SYSTEMS July/August*, 79-85.

Shilton, K., Burke, J., Caffy, K., & Zhang, L. (2016). Anticipating Policy and Social Implications of Named Data Networking. *COMMUNICATIONS OF THE ACM | DECEMBER 2016 | VOL. 59 | NO. 12*.

Shood, K., Yu, S., & Xiang, Y. (2006). Software-Defined Wireless Networking Opportunities and Challenges for Internet-of-Things: A Review. *IEEE INTERNET OF THINGS JOURNAL, VOL. 3, NO. 4, AUGUST*, 453-463.

Singh, K., Singh, P., & Kumar, K. (2017). Application layer HTTP-GET floods DDoS attacks: Research landscape and challenges. *Computers & Security 65*, 344-372.

Ullah, I., Ullah, Z., Afzaal, U., & Lee, J. (2019). DURE: An Energy- and Resource-Efficient TCAM Architecture for FPGAs With Dynamic Updates. *IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 27, NO. 6, JUNE 2019*, 1298-1307.