Shaping Smart City Systems

RJ Holman

Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA

April 20, 2019

Somnio Studios

Introduction

When is the last time you saw one of these?

Introduction

When is the last time you saw one of these?

Or saw something like this?

Introduction

Or wait a substantial amount of time for this

Img: https://www.strongtowns.org/journal/2016/12/13/best-of-2016-about-those-pesky-pedestrian-crossing-buttons

Introduction

Or wait a substantial amount of time for this

Inconveniences like these are becoming:

Img: https://www.strongtowns.org/journal/2016/12/13/best-of-2016-about-those-pesky-pedestrian-crossing-buttons

Introduction

Or wait a substantial amount of time for this

Inconveniences like these are becoming:

- More prevalent

Img: https://www.strongtowns.org/journal/2016/12/13/best-of-2016-about-those-pesky-pedestrian-crossing-buttons

April 20, 2019

Introduction

Or wait a substantial amount of time for this

Inconveniences like these are becoming:

- More prevalent
- More avoidable

Outline

1. Background

2. The Network (SOXFire)

3. Conflicts

Background

Background: Smart City Basics

Smart City: a city with a network of technology systems that aid in typical city tasks.

Background: Smart City Basics

Smart City: a city with a network of technology systems that aid in typical city tasks.

 Meets the following goals:

Background: Smart City Basics

Smart City: a city with a network of technology systems that aid in typical city tasks.

 Meets the following goals:

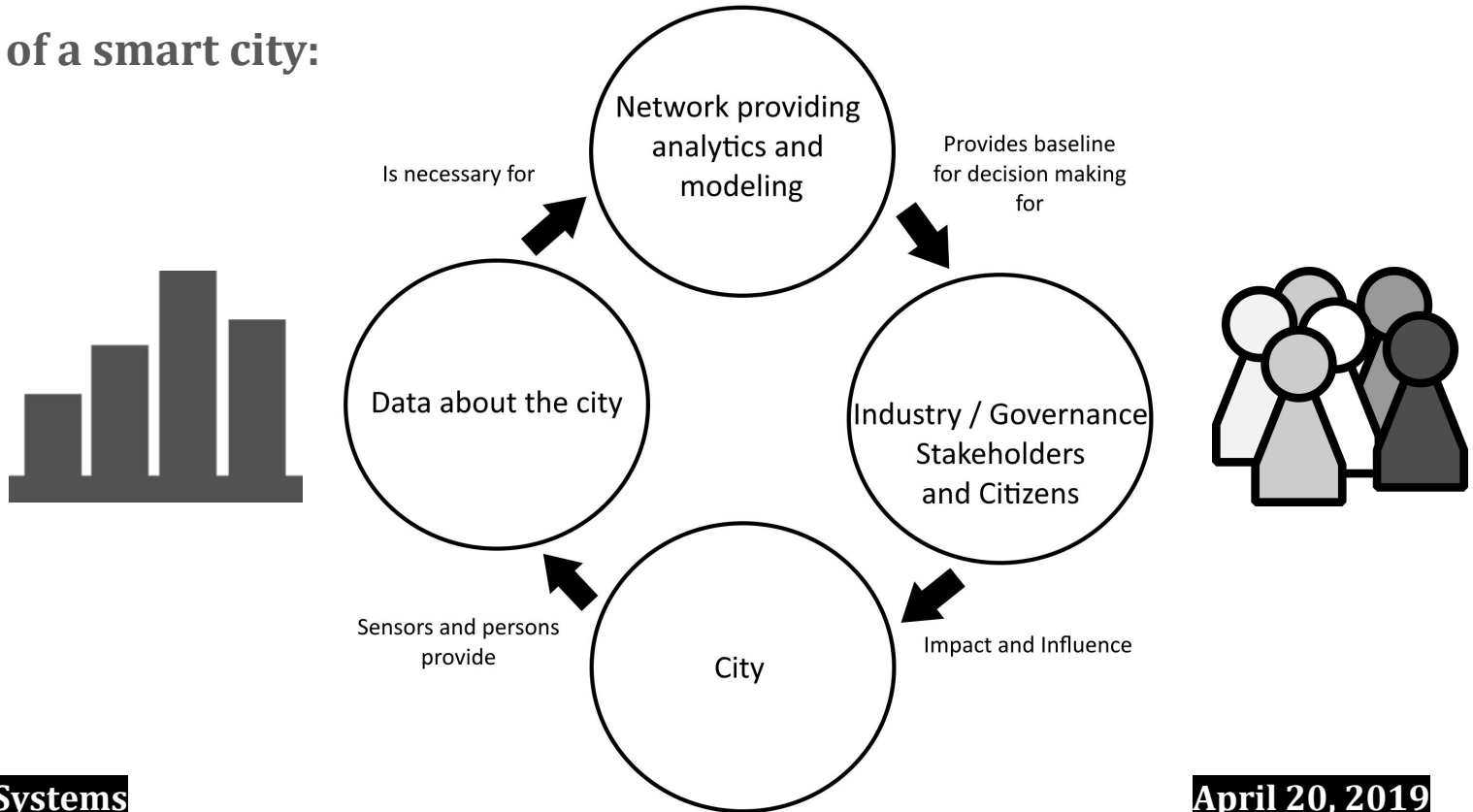- Data collection

Background: Smart City Basics

Smart City: a city with a network of technology systems that aid in typical city tasks.

Meets the following goals:

- Data collection

- Data analysis

Background: Smart City Basics

Smart City: a city with a network of technology systems that aid in typical city tasks.

Meets the following goals:

- Data collection

- Data analysis

- Response to the data

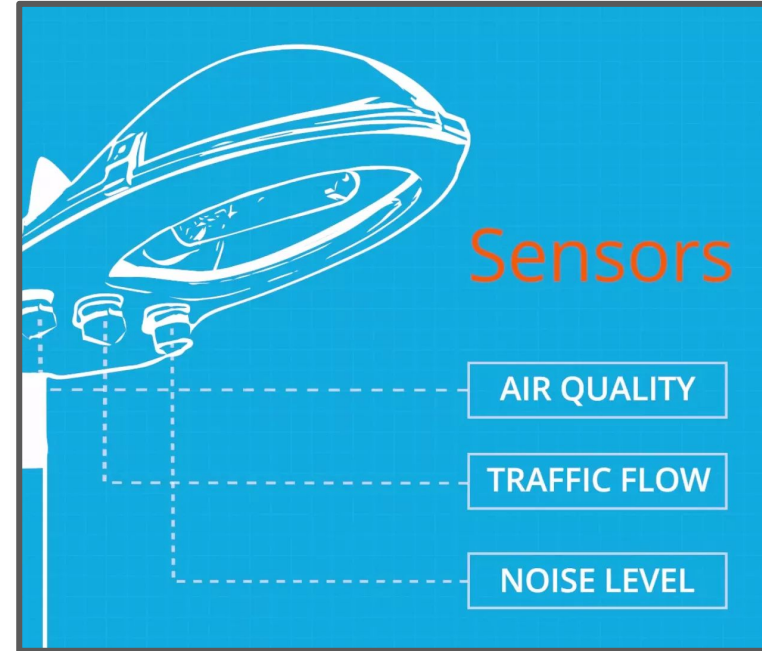# Background: Smart City Basics

Feedback loop of a smart city:

Network providing analytics and modeling

Is necessary for

Provides baseline for decision making for

Data about the city

Industry / Governance Stakeholders and Citizens

Sensors and persons provide

City

Impact and Influence

Background: Sensors

Types of Sensors

Background: Sensors

Types of Sensors

- Fixed Sensors

  - Most Common
  - Quantitative Data
  - Cover small area
  - Expensive

Sensors

AIR QUALITY

TRAFFIC FLOW

NOISE LEVEL

Img: https://enterpriseiotinsights.com/20170619/smart-cities/20170619smart-citiesconnected-lighting-smart-city-platform-tag23

Background: Sensors

Types of Sensors

- Fixed Sensors

  - Quantitative Data

- Vehicle-Mounted Sensors

  - Quantitative data
  - Cover large area
  - Inexpensive
  - Less reliable data

Background: Sensors

Types of Sensors

- Fixed Sensors
  - Quantitative Data

- Vehicle-Mounted Sensors
  - Quantitative Data

- Crowd Sensing
  - Participatory & Expert
  - Qualitative Data

Img: https://www.istockphoto.com/illustrations/crowd-looking-up?sort=mostpopular&mediatype=illustration&phrase=crowd%20looking%20up
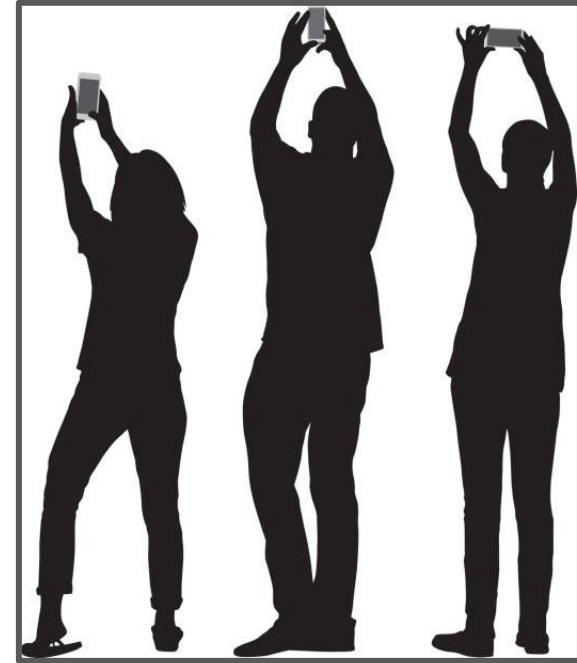
The Network

The Network: Requirements

The Network: Requirements

- Handle heterogeneous sensor data

The Network: Requirements

- Handle heterogeneous sensor data

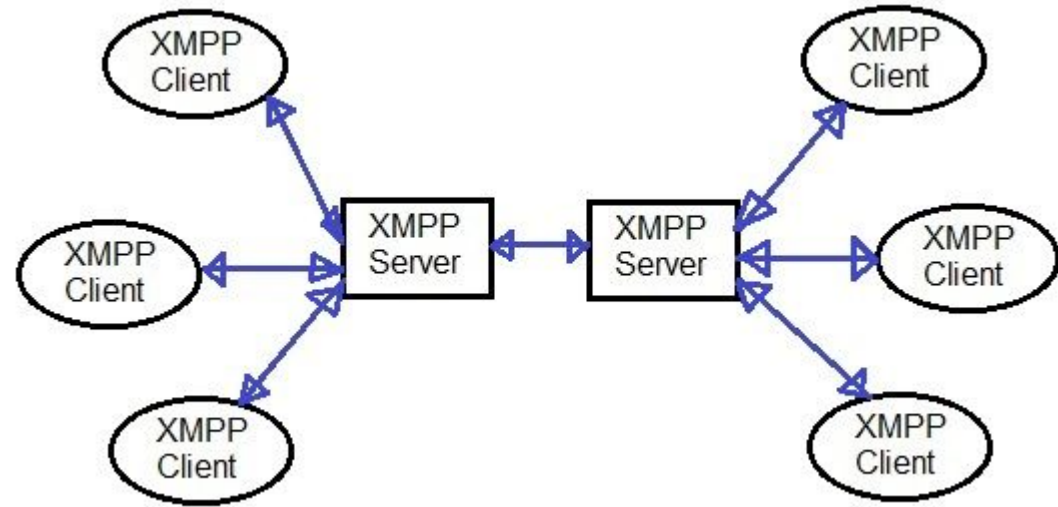- Extendability and Scalability
  - Federation

The Network: Requirements

- Handle heterogeneous sensor data

- Extendability and Scalability
  - Federation
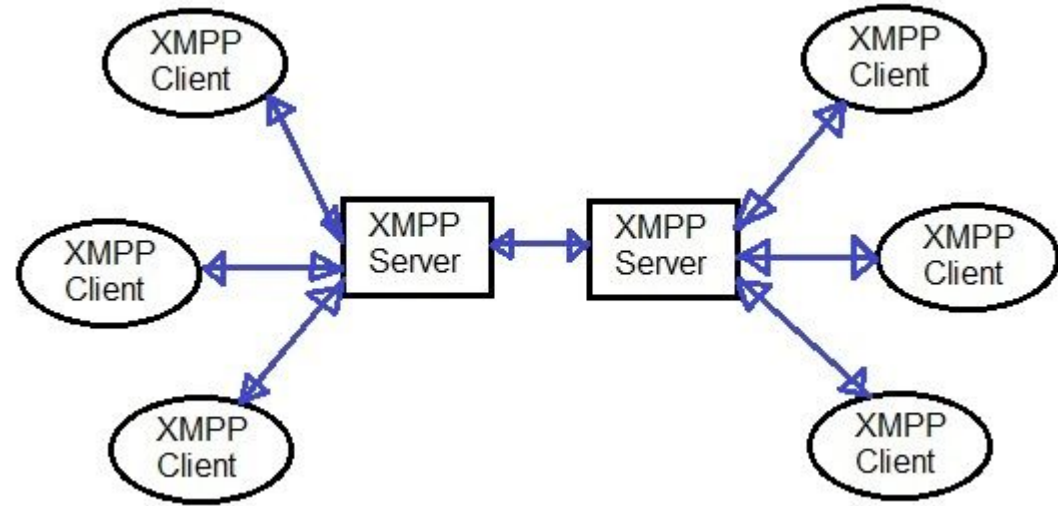
- Security

The Network: SOXFire

The Network: SOXFire

- Built on XMPP
  - Used for online chat
  - Extensibile
  - Decentralized
  - Built in security

Img: http://www.rfwireless-world.com/IoT/XMPP-protocol.html

The Network: SOXFire

- Built on XMPP
  - Used for online chat
  - Extensibile
  - Decentralized
  - Built in security
  - Federation

Img: http://www.rfwireless-world.com/IoT/XMPP-protocol.html

The Network: SOXFire

XMPP Federation

- Decentralization

The Network: SOXFire

XMPP Federation

- Decentralization

- Gateways

The Network: SOXFire

XMPP Federation

- Decentralization
  - Can be run on any domain
  - Possible through unique IDs
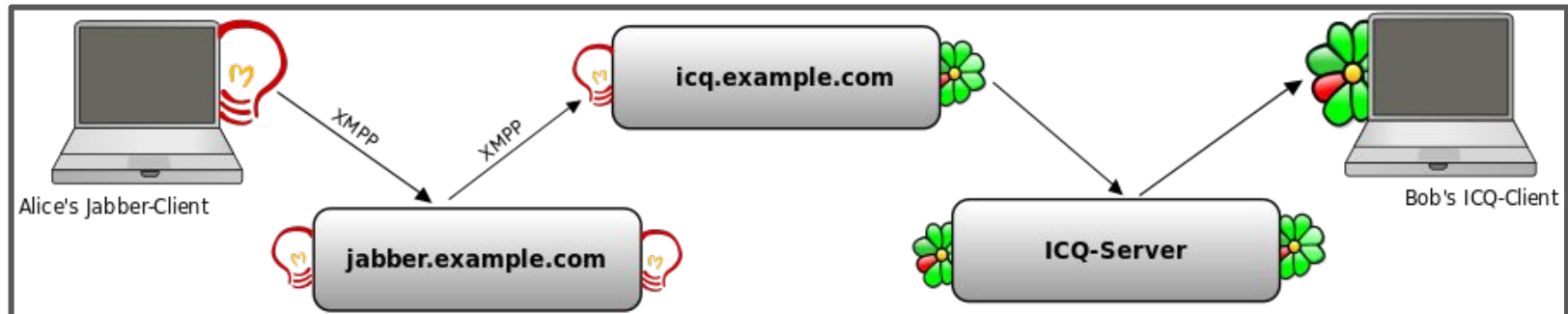- Gateways

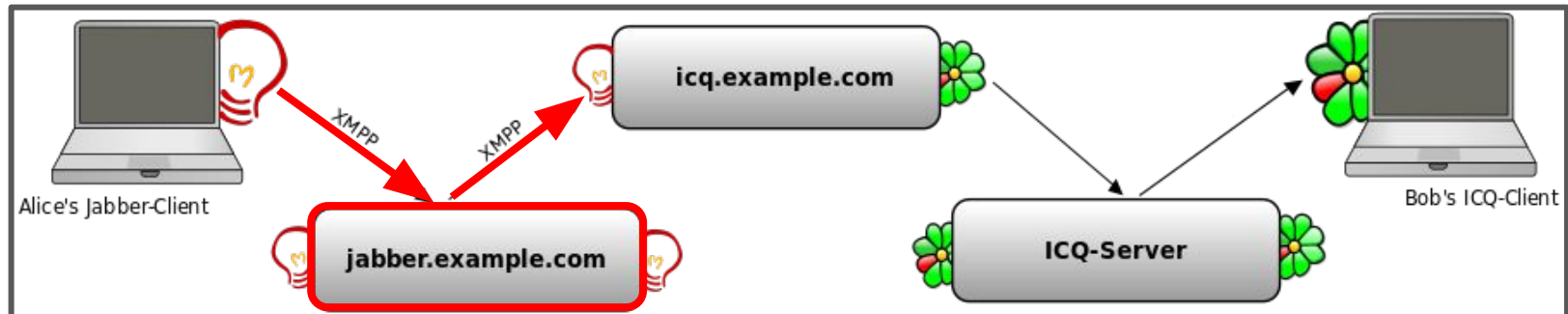# The Network: SOXFire

XMPP Federation

- Decentralization
  - Can be run on any domain
  - Possible through JIDs
- Gateways
  - Allows XMPP network to communicate with another network even if not XMPP
  - Allows auto authentication on users registered to the gateways

The Network: SOXFire
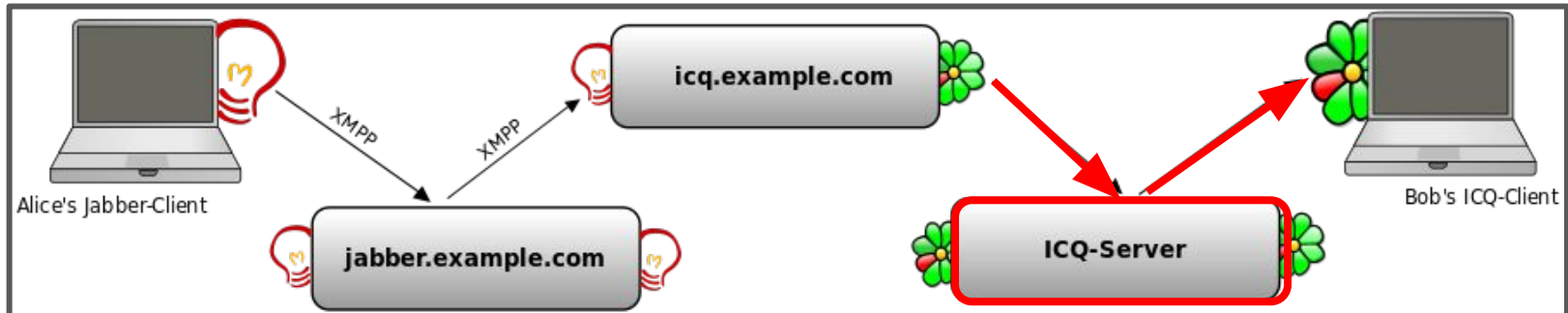
XMPP Federation

- Decentralization
  - Can be run on any domain
  - Possible through JIDs
- Gateways
  - Allows XMPP network to communicate with another network even if not XMPP
  - Allows auto authentication on users registered to the gateways

The Network: SOXFire

XMPP Federation

- Decentralization
  - Can be run on any domain
  - Possible through JIDs
- Gateways
  - Allows XMPP network to communicate with another network even if not XMPP
  - Allows auto authentication on users registered to the gateways

The Network: SOXFire

XMPP Federation

- Decentralization
  - Can be run on any domain
  - Possible through JIDs
- Gateways
  - Allows XMPP network to communicate with another network even if not XMPP
  - Allows auto authentication on users registered to the gateways

Alice's Jabber-Client

XMPP   XMPP   icq.example.com

jabber.example.com   ICQ-Server   Bob's ICQ-Client

The Network: SOXFire

XMPP Federation

- Decentralization
  - Can be run on any domain
  - Possible through JIDs
- Gateways
  - Allows XMPP network to communicate with another network even if not XMPP
  - Allows auto authentication on users registered to the gateways

The Network: SOXFire

What does SOXFire add?

The Network: SOXFire

What does SOXFire add?

- Subscription and publish events *with* federation
    - More scalability for a city scale

The Network: SOXFire

What does SOXFire add?

- Subscription and publish events *with* federation
  - More scalability for a city scale

- Virtual sensor
  - Data node
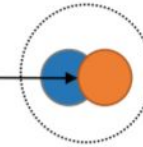 - Metadata node

The Network: SOXFire

What does SOXFire add?

- Subscription and publish events *with* federation
  - More scalability for a city scale

- Virtual sensor
  - Data node
  - Metadata node

1) Publisher : Virtual Sensor = 1 : 1

JID: User A
(owner of the sensor)

Sensor data publish

Access Model: *open*

Publish Model:
publishers
(only user A is allowed)

2) Publisher : Virtual Sensor = n : 1

JID: User B

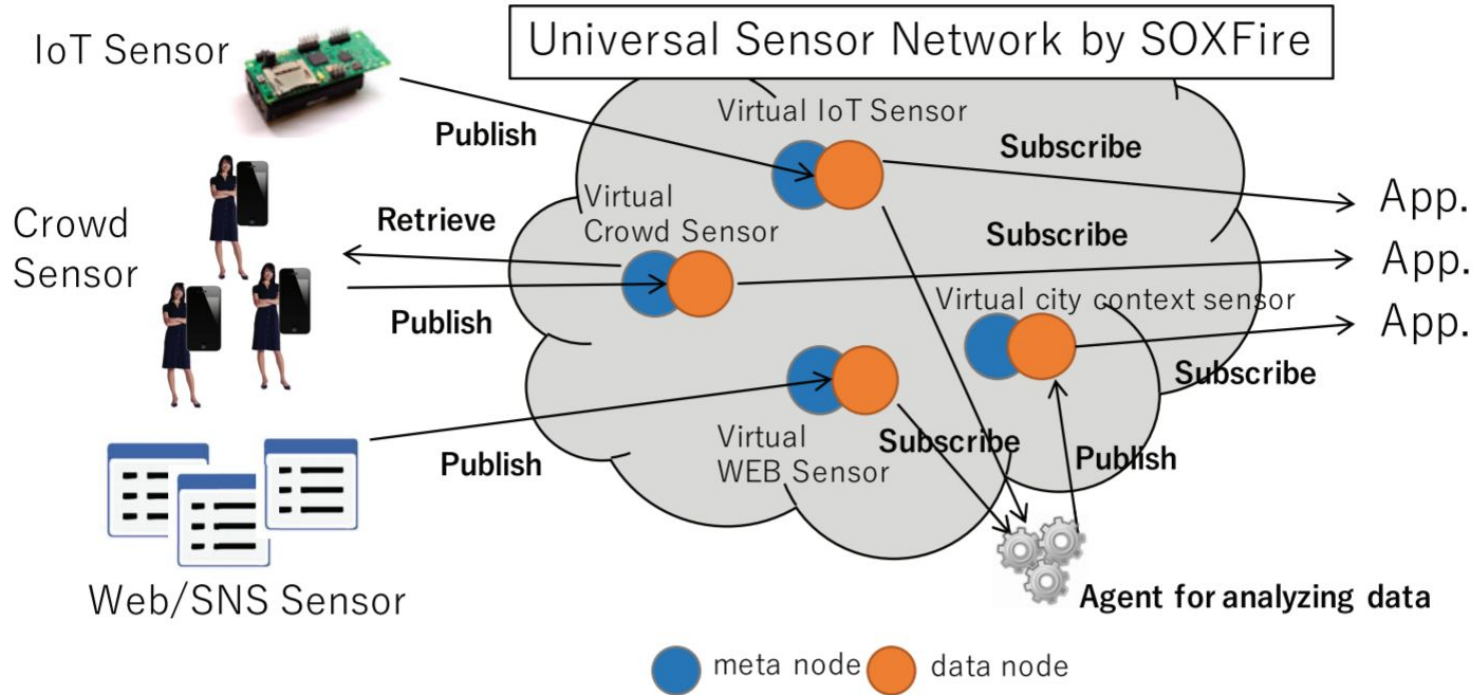JID: User C

Sensor data publish
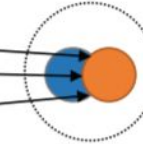
Access Model: *open*

Publish Model: *open*

JID: User D

Img: SOXFire: A Universal Sensor Network System
for Sharing Social Big Sensor Data in Smart Cities

The Network: SOXFire

Universal Sensor Network by SOXFire

Img: SOXFire: A Universal Sensor Network System
for Sharing Social Big Sensor Data in Smart Cities

Visualization Tools

Visualization Tools: SOXFire

SOXFire Dashboard

- A "Widget" Interface
- Provides customizable locations and information

(1) Main menu (select registered dashboard or create new one)

(3) Widget add interface

(2) City dashboard

Img: SOXFire: A Universal Sensor Network System for Sharing Social Big Sensor Data in Smart Cities
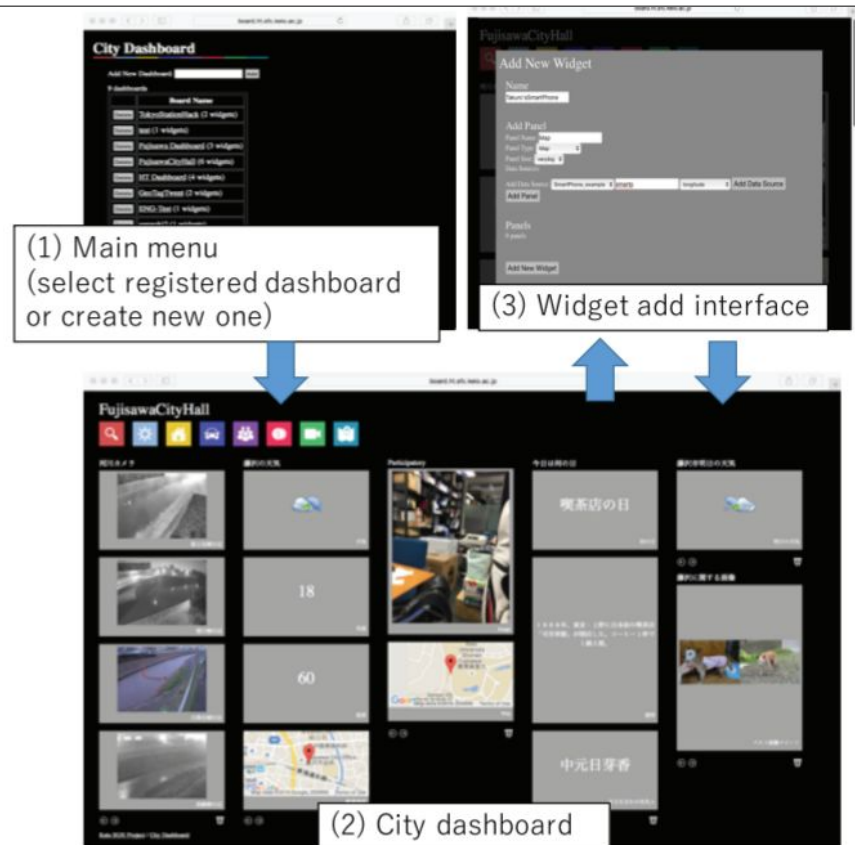
MIT City Scanner
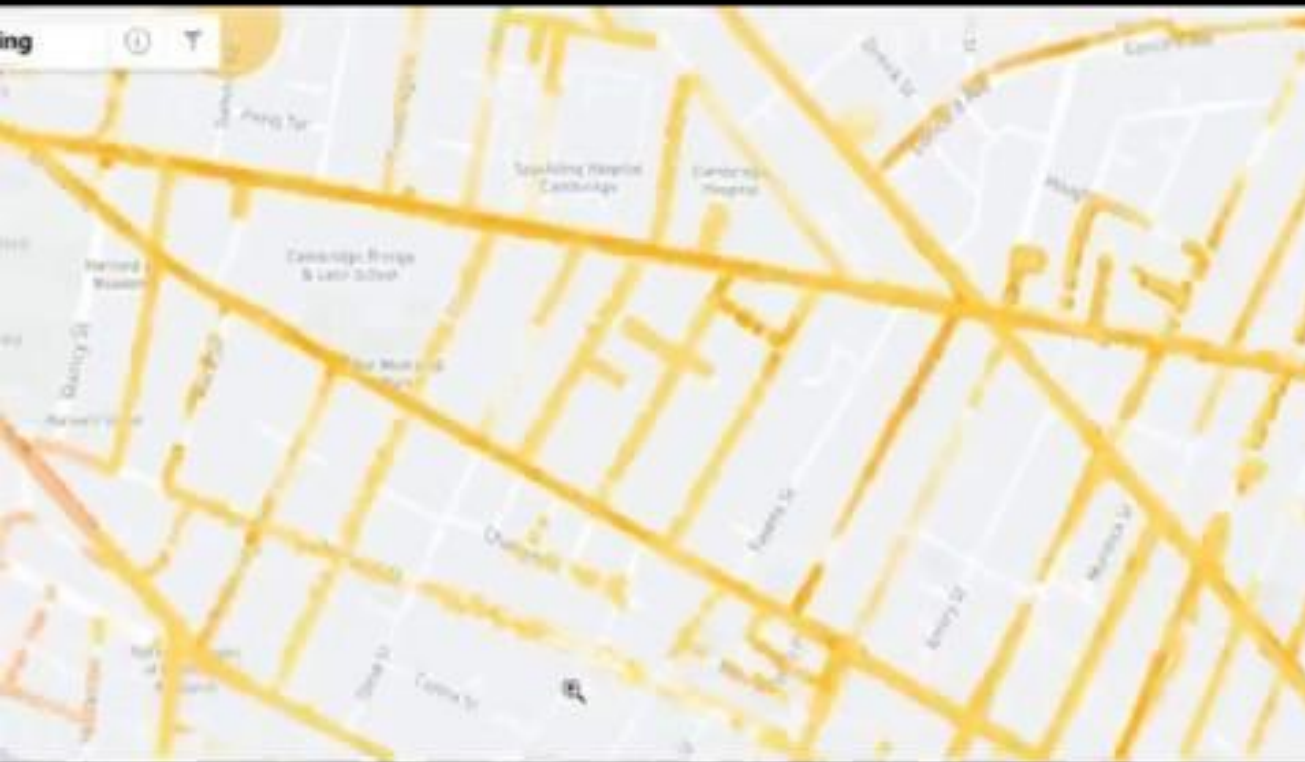
- **"Heatmap" approach**
- **Garbage truck mounted sensors gather data on various environmental statistics within Cambridge, MA**

System Conflicts

System Conflicts: Conflict Types

System Conflicts: Conflict Types

- Device Conflicts
  - Pedestrians vs traffic on same light

System Conflicts: Conflict Types

- Device Conflicts
  - Pedestrians vs traffic on same light
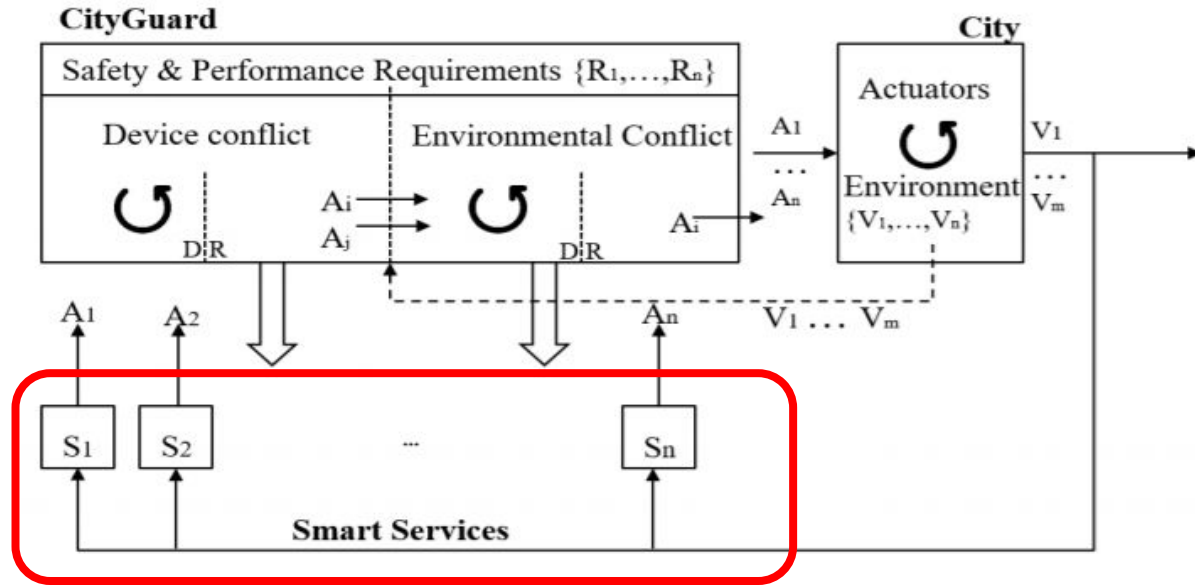
- Environmental Conflicts
  - Congestion service vs emergency

System Conflicts: CityGuard

System Conflicts: CityGuard

- Addresses Conflicts via
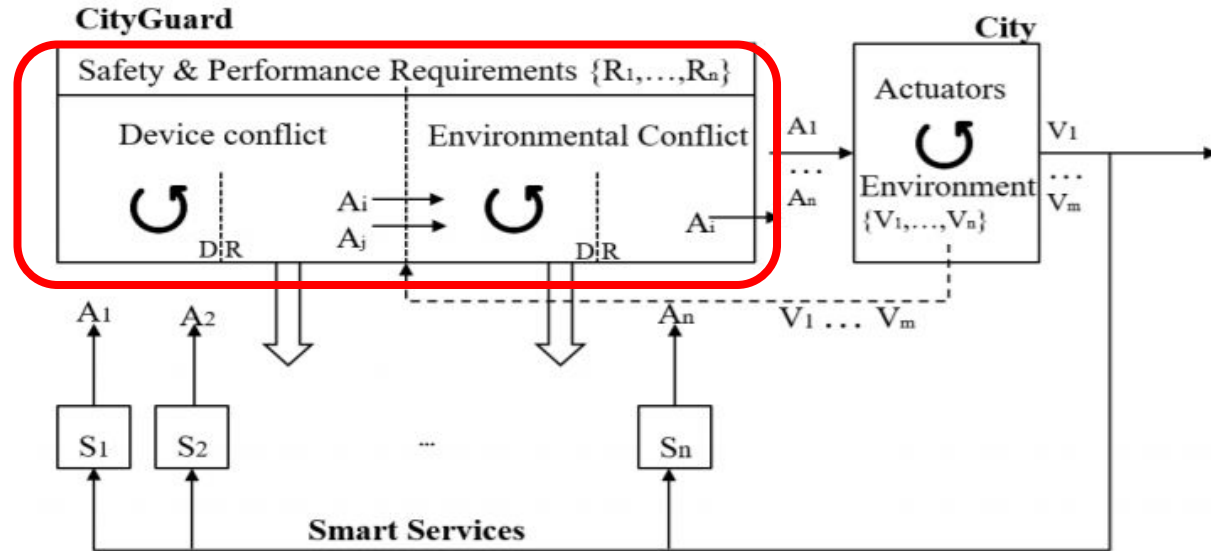 - Feedback loop algorithm
  - Actuators

System Conflicts: CityGuard

- Addresses Conflicts via
  - Feedback loop algorithm
  - Actuators

- Overview
  - Takes in service actions

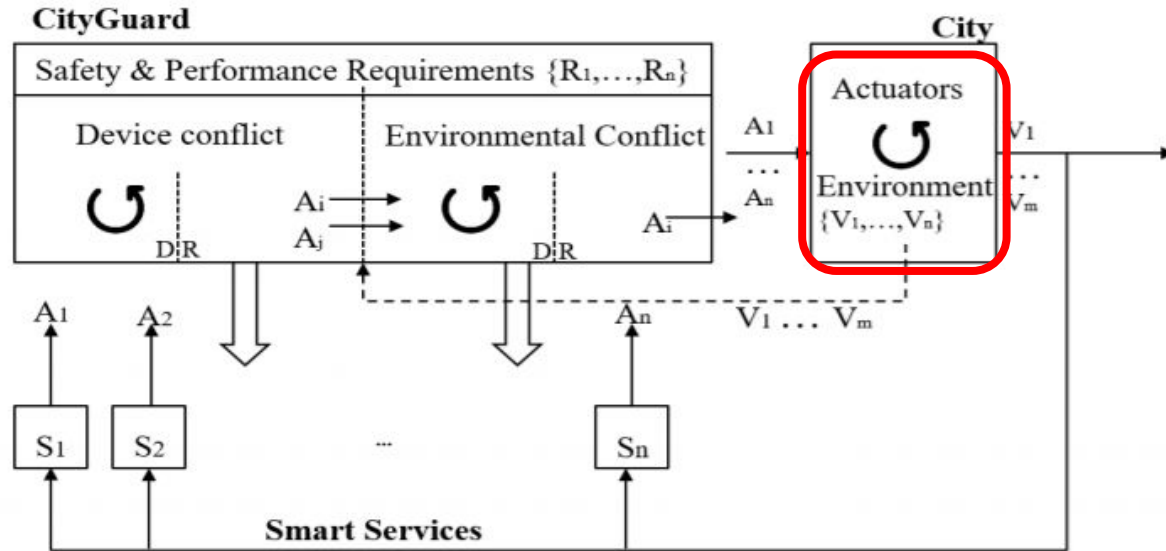**Img:** CityGuard: A Watchdog for Safety-Aware Conflict Detection in Smart Cities
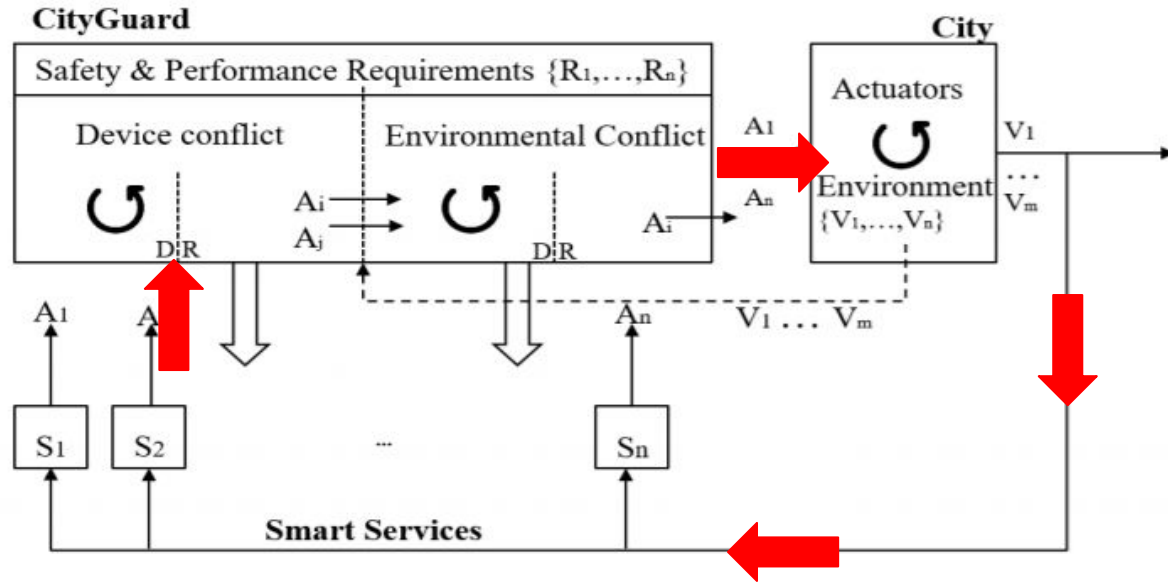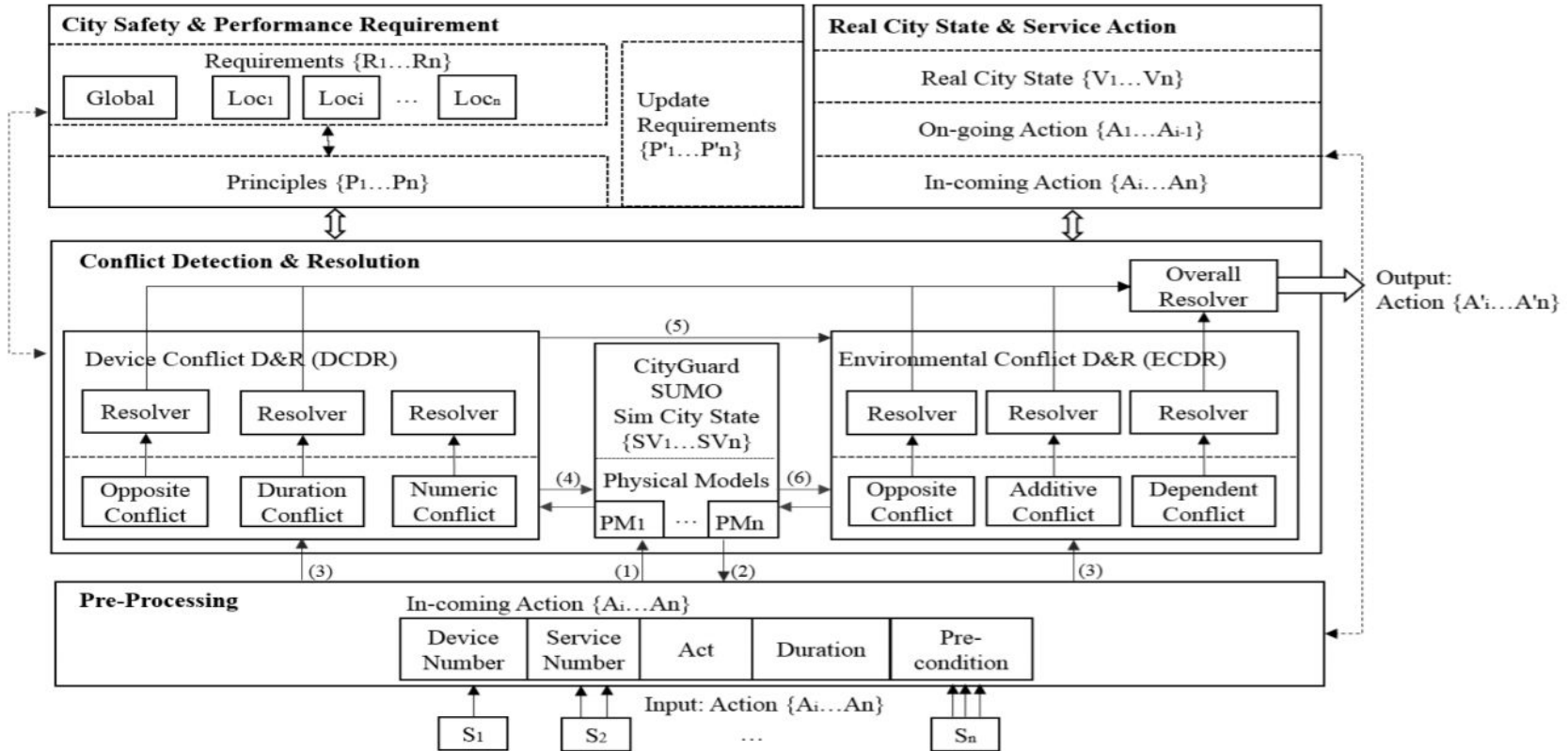
System Conflicts: CityGuard

- Addresses Conflicts via
  - Feedback loop algorithm
  - Actuators

- Overview
  - Takes in service actions
  - Looks for conflicts
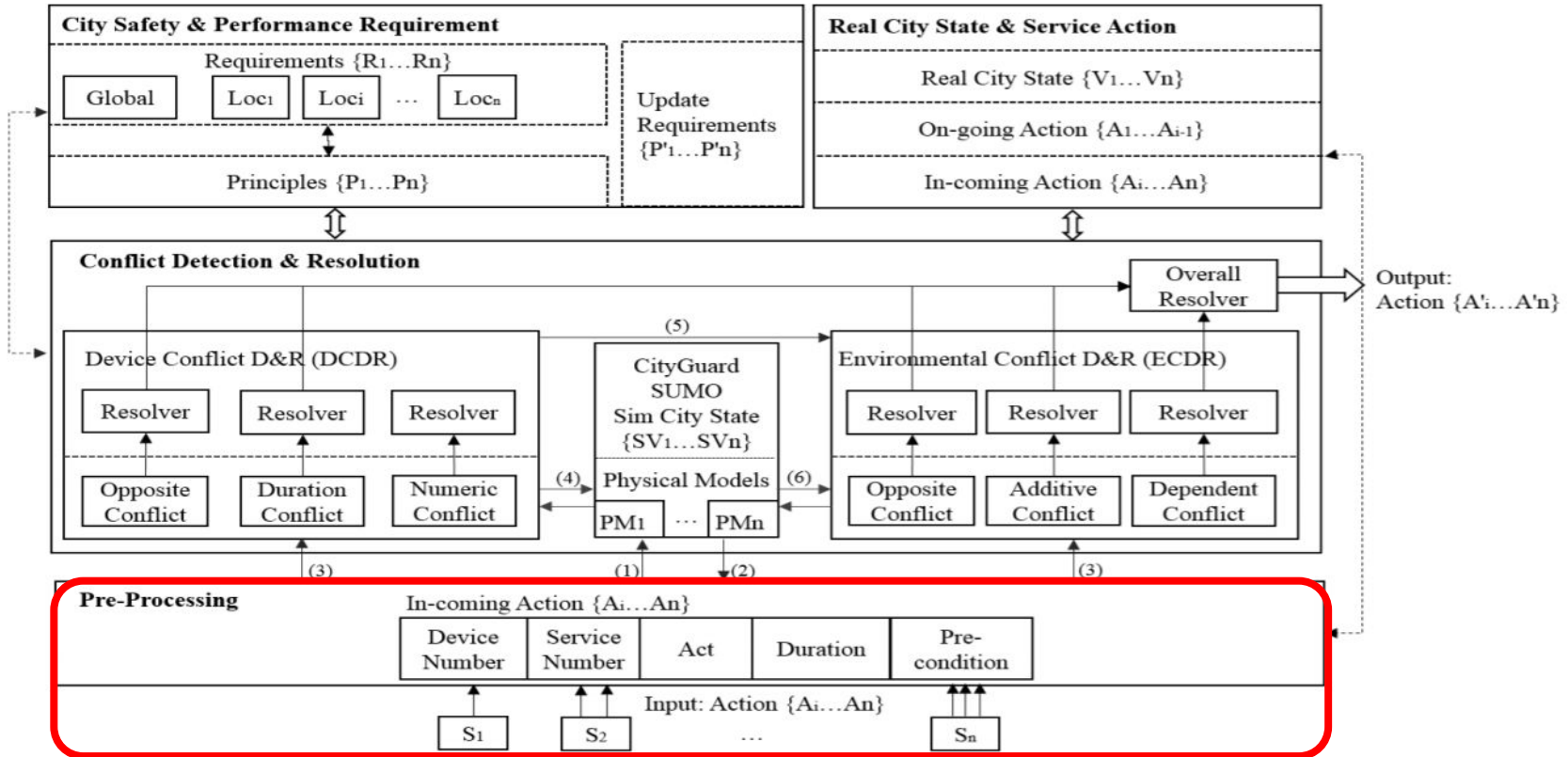
System Conflicts: CityGuard

- Addresses Conflicts via
  - Feedback loop algorithm
  - Actuators

- Overview
  - Takes in service actions
  - Looks for conflicts
  - Actuators automate response

System Conflicts: CityGuard

- Addresses Conflicts via
  - Feedback loop algorithm
  - Actuators

- Overview
  - Takes in service actions
  - Looks for conflicts
  - Actuators automate response
  - Cycle Repeats

System Conflicts: Algorithm

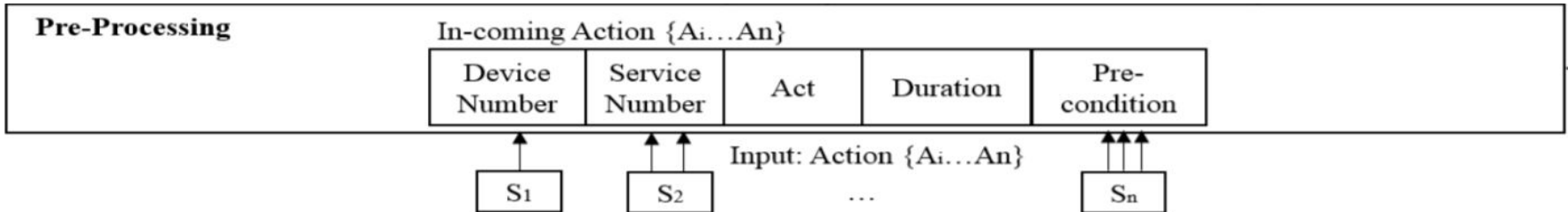**Img:** CityGuard: A Watchdog for Safety-Aware Conflict Detection in Smart Cities

System Conflicts: Algorithm

System Conflicts: Algorithm

Pre-processing

- Takes set of actions
  - $\{A'_i \ldots A'_n\}$

Output:
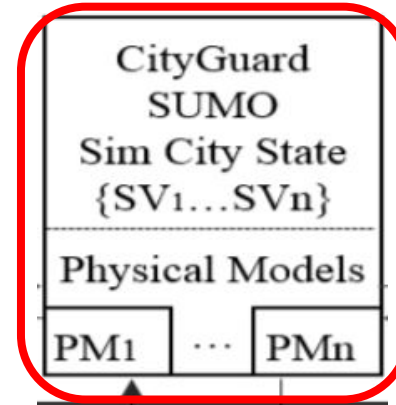Action $\{A'_i \ldots A'_n\}$

Pre-Processing	In-coming Action $\{A_i \ldots A_n\}$				
	Device Number	Service Number	Act	Duration	Pre-condition

Input: Action $\{A_i \ldots A_n\}$

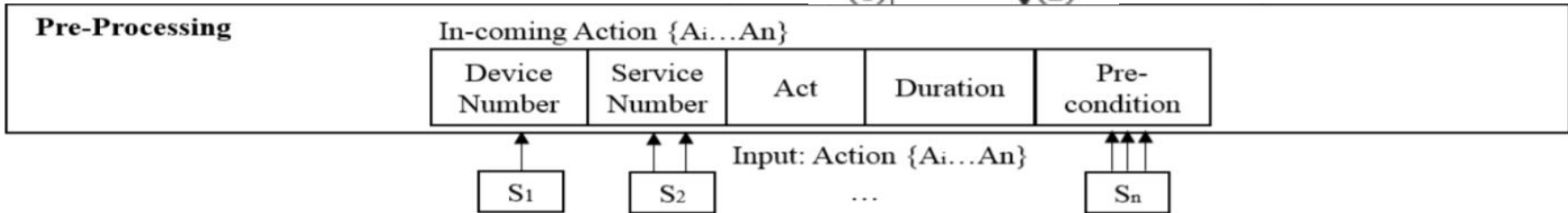S_1 S_2 $\ldots$ S_n
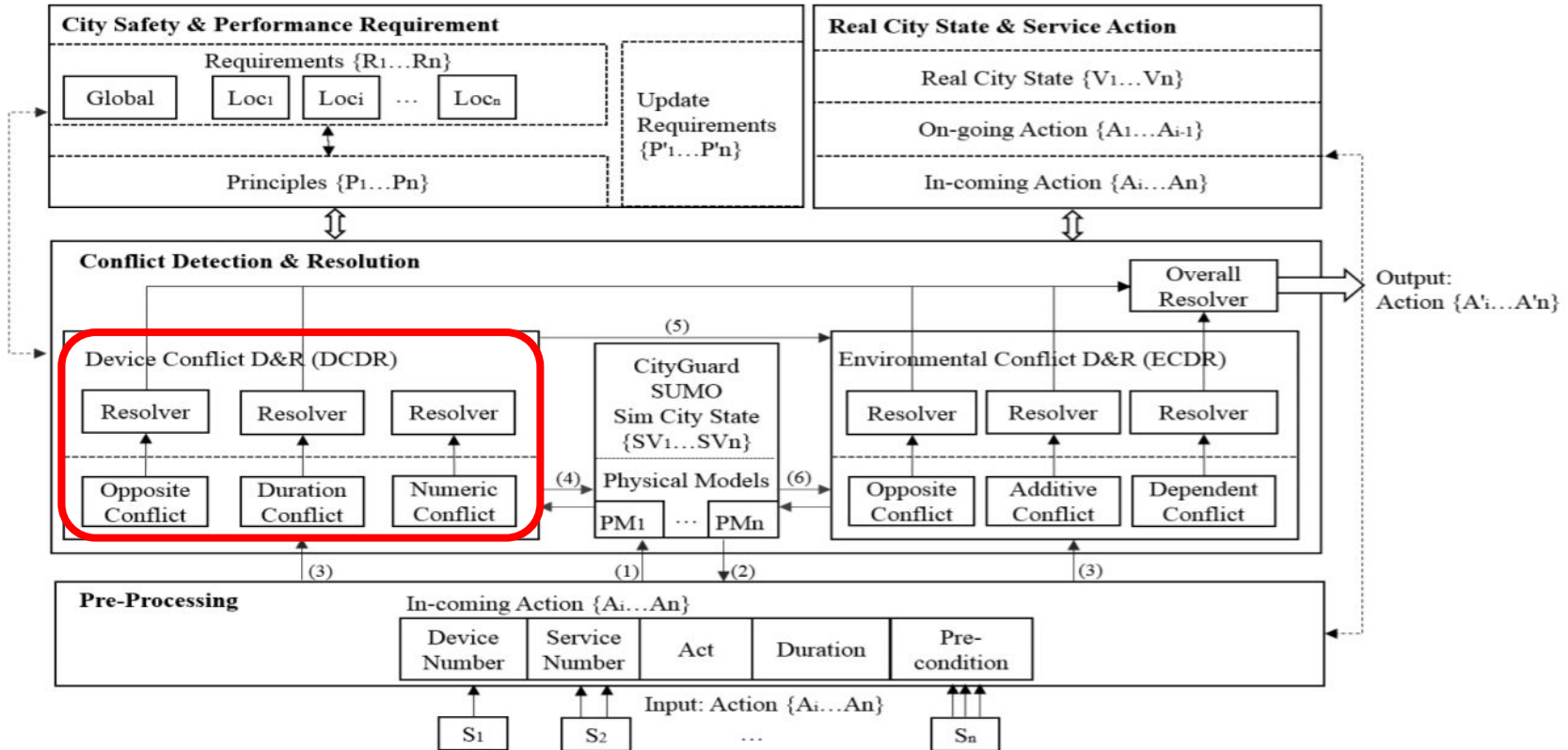
System Conflicts: Algorithm

Pre-processing

- Takes set of actions
  - $\{A'_i \ldots A'_n\}$

- CityGuard SUMO run
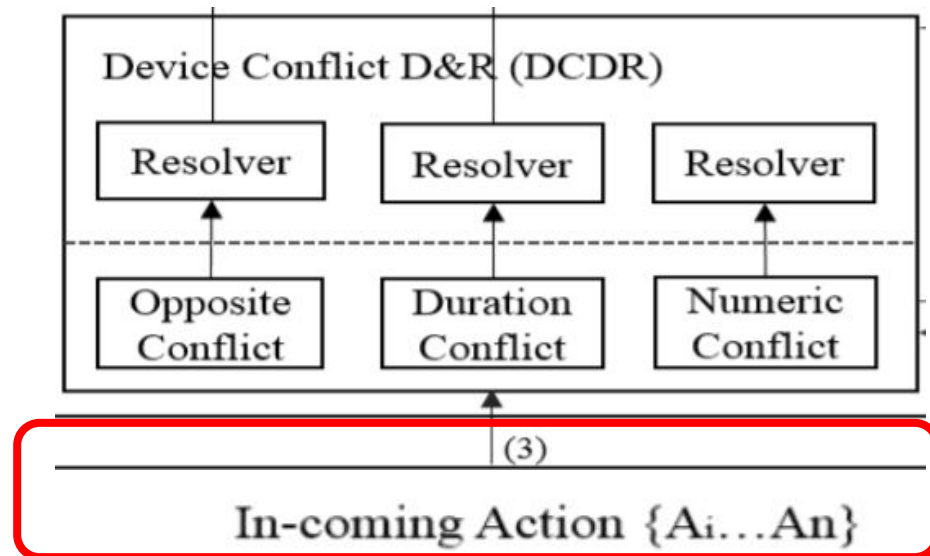 - Returns simulation state: $\{SV_1 \ldots SV_n\}$

System Conflicts: Algorithm

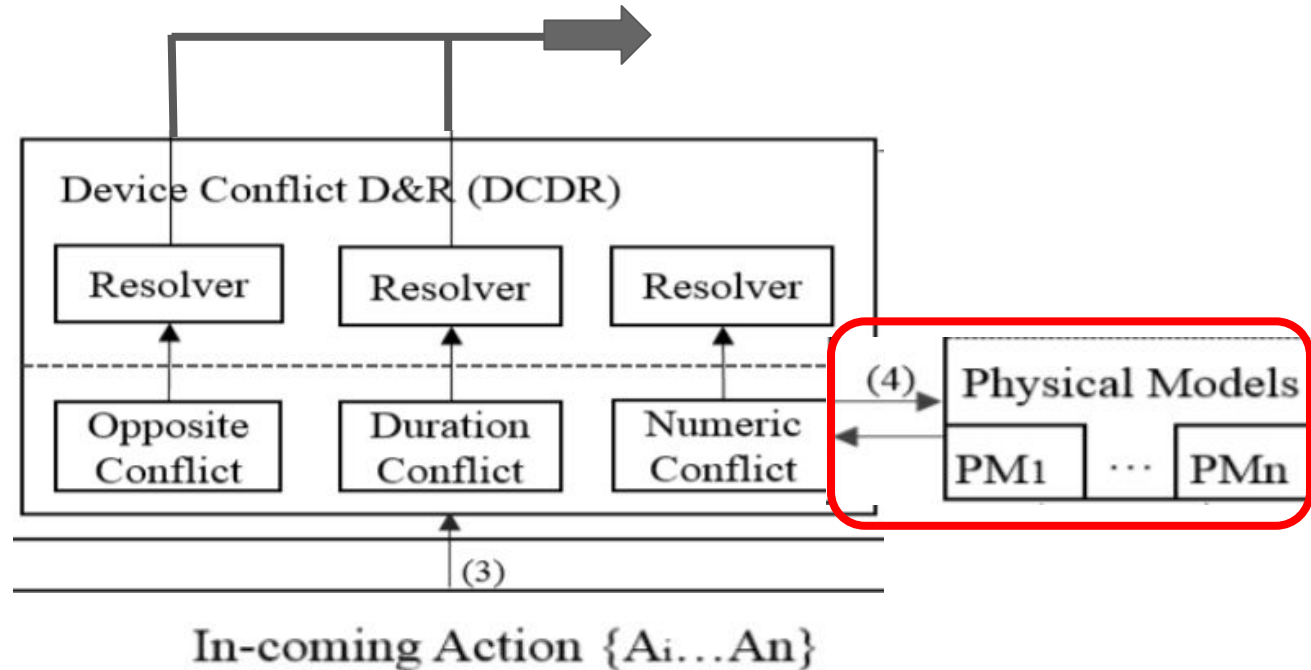# **System Conflicts:** Algorithm

Device Conflict Resolution

- Takes set of actions
  - $\{A_i \ldots A_n\}$
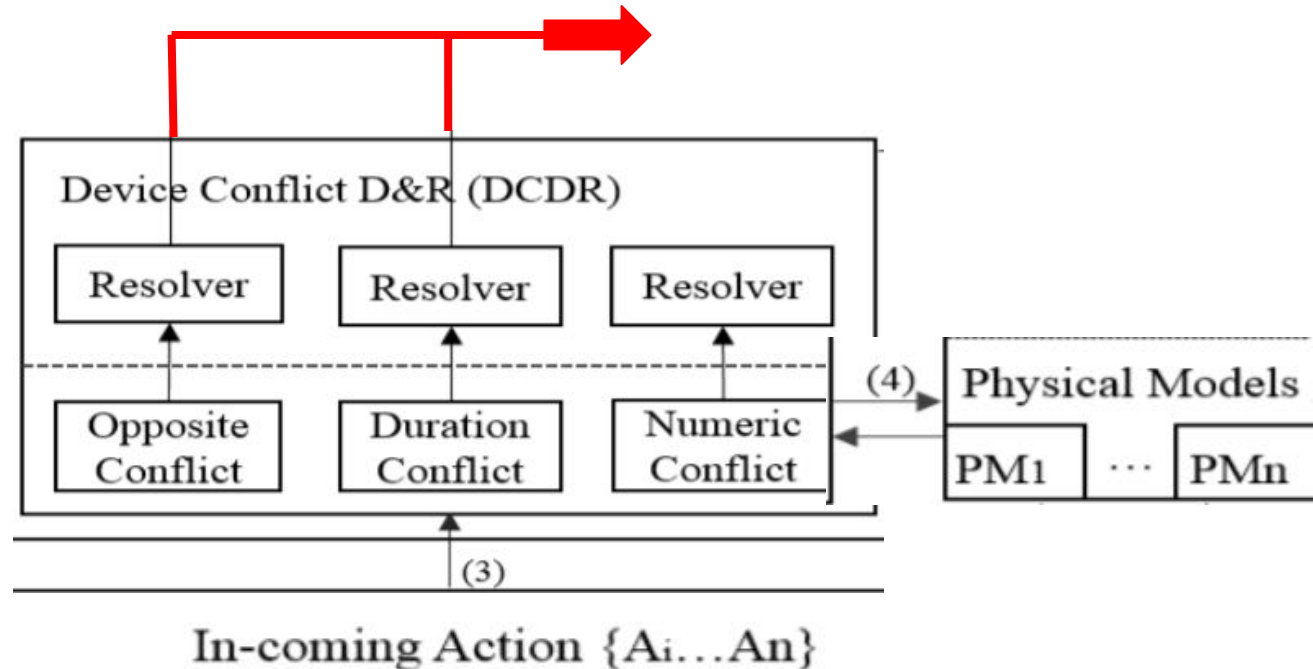
System Conflicts: Algorithm

Device Conflict Resolution

- Takes set of actions
  - $\{A_i \ldots A_n\}$

- Check for Conflict

Device Conflict D&R (DCDR)

| Resolver | Resolver | Resolver |

(4) Physical Models

PM1 ... PMn

| Opposite Conflict | Duration Conflict | Numeric Conflict |

(3)

In-coming Action $\{A_i \ldots A_n\}$
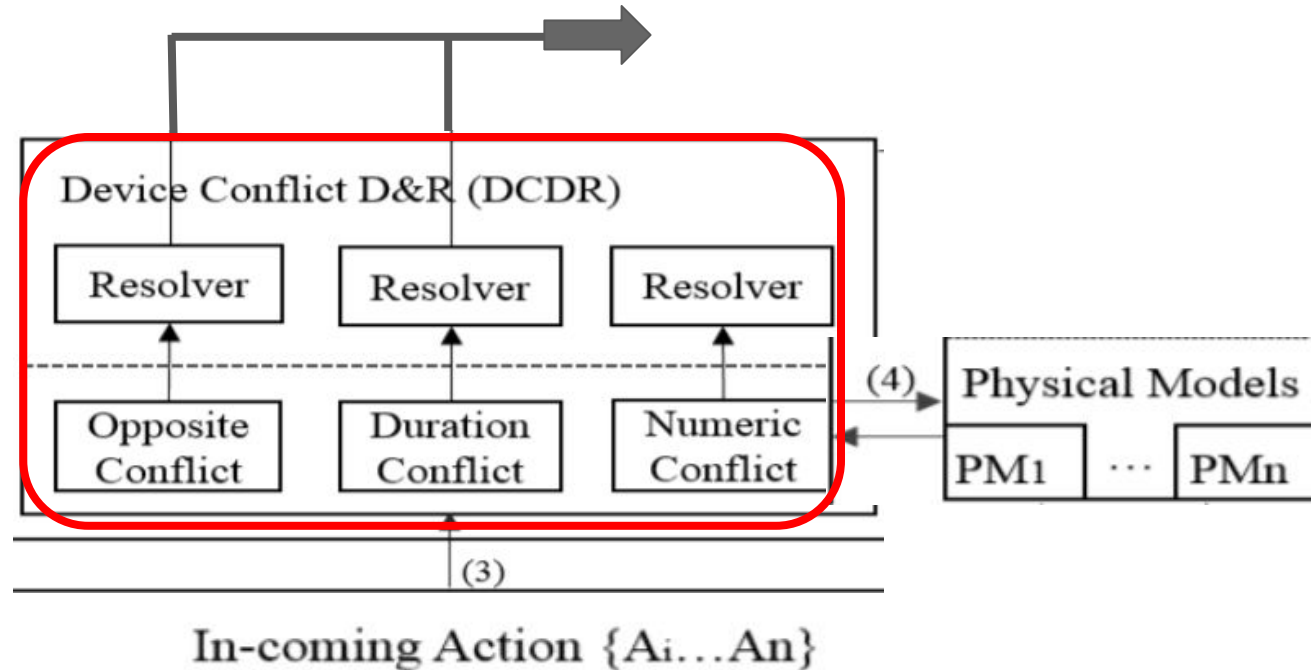
System Conflicts: Algorithm

Device Conflict Resolution

- Takes set of actions
  - $\{A_i \ldots A_n\}$

- Check for Conflict

- If no device conflict
  - Move on

Device Conflict D&R (DCDR)

| Resolver | Resolver | Resolver |

Opposite Conflict | Duration Conflict | Numeric Conflict

(4) Physical Models

PM1 ⋯ PMn

(3)

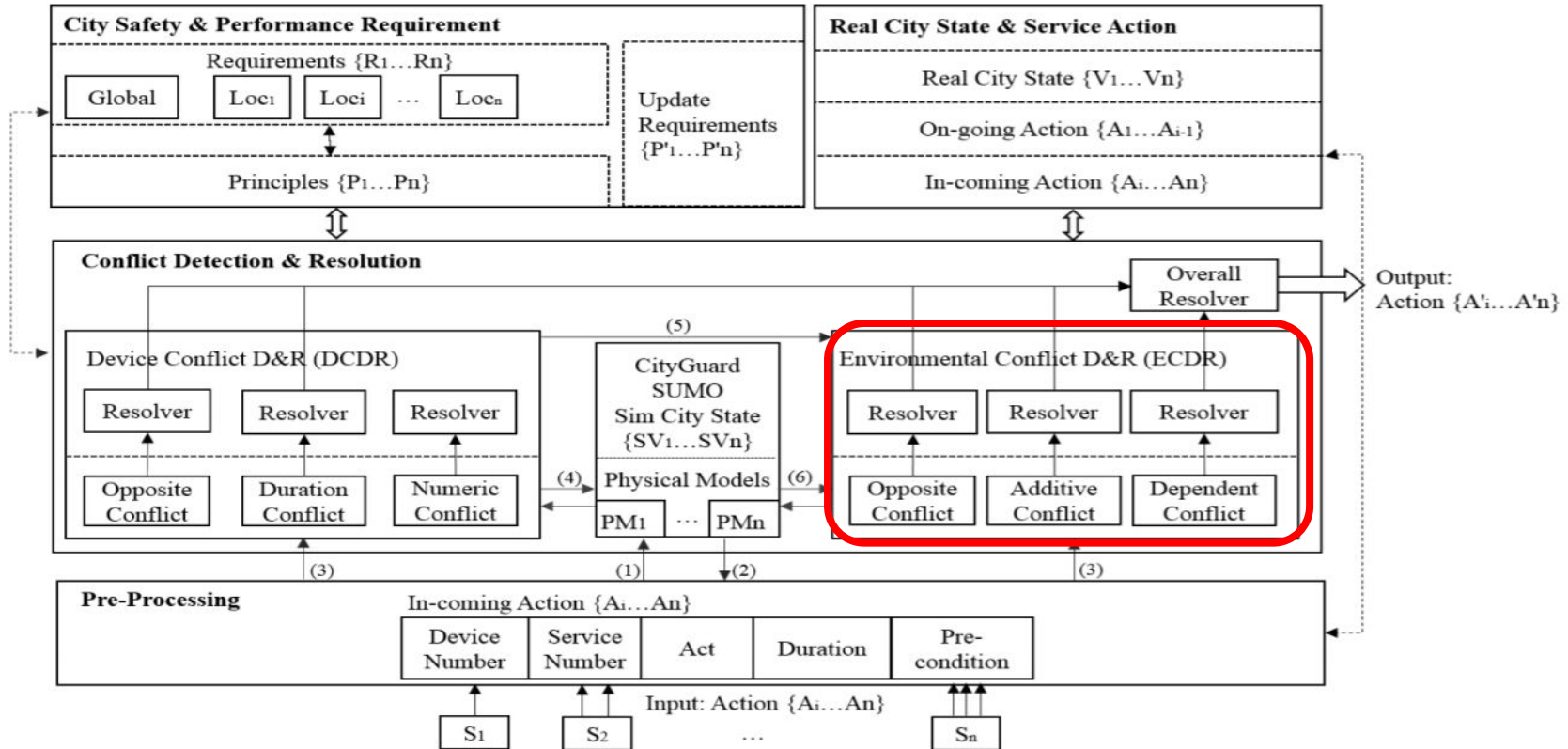In-coming Action $\{A_i \ldots A_n\}$

System Conflicts: Algorithm

Device Conflict Resolution

- Takes set of actions
  - $\{A_i \ldots A_n\}$

- Check for Conflict

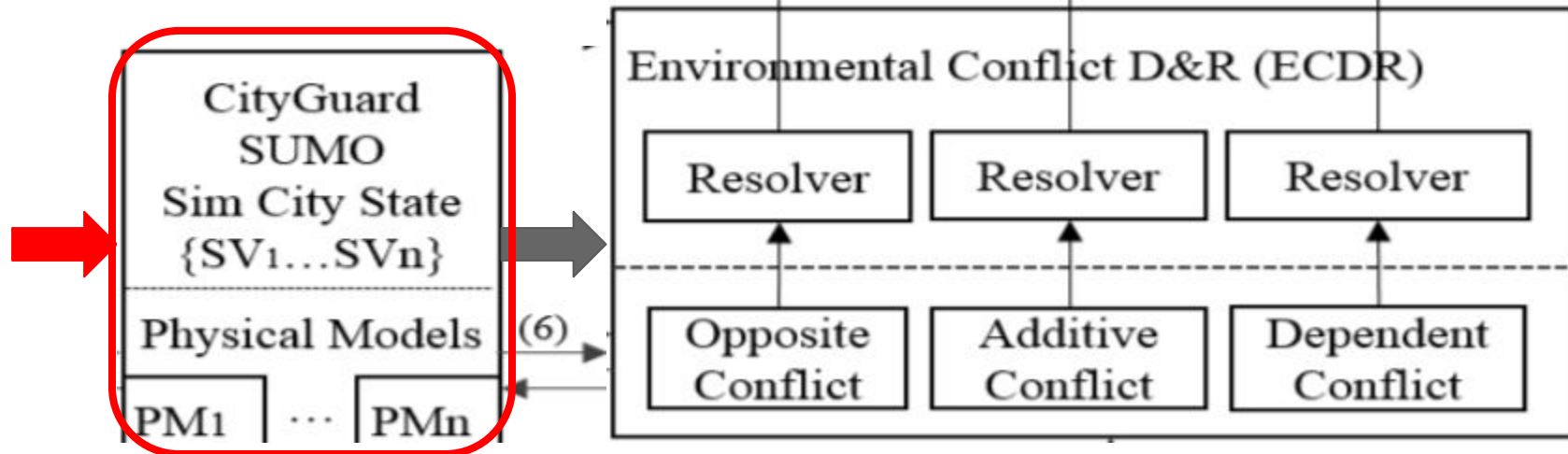- If no device conflict
  - Move on

- Else
  - Categorize
  - Resolve

Device Conflict D&R (DCDR)

| Resolver | Resolver | Resolver |

Physical Models

| Opposite Conflict | Duration Conflict | Numeric Conflict |

(4)

| PM1 | ⋯ | PMn |

(3)

In-coming Action $\{A_i \ldots A_n\}$

System Conflicts: Algorithm

System Conflicts: Algorithm
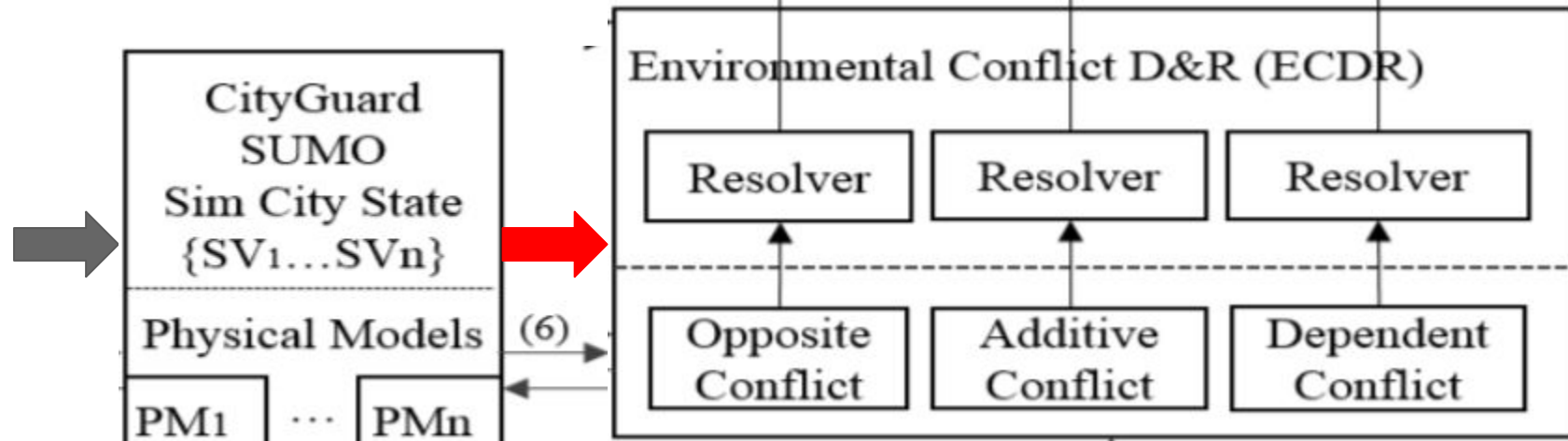
Environmental Conflict Resolution

- resolved DC actions rerun in SUMO

System Conflicts: Algorithm
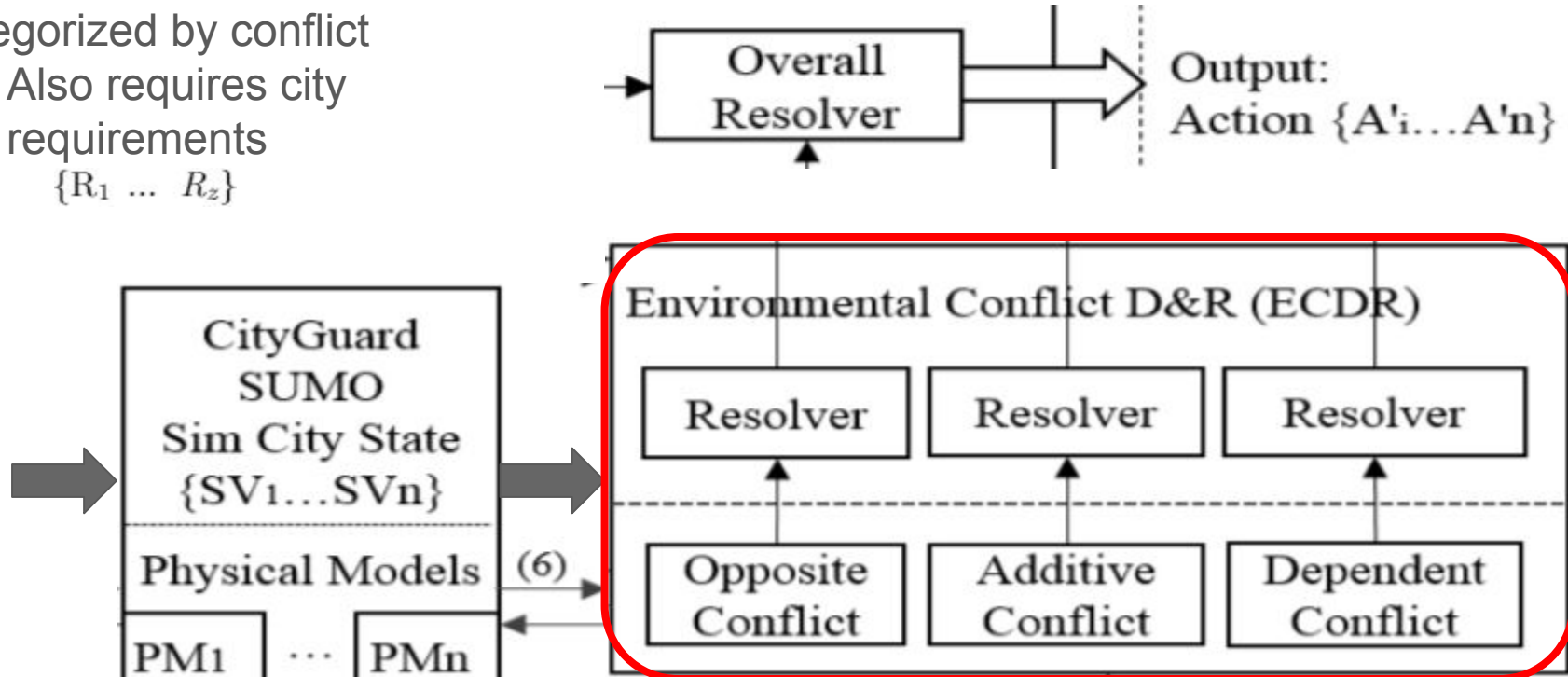
Environmental Conflict Resolution

- Takes SUMO rerun actions
  - $\{A'_i \ldots A'_n\}$

System Conflicts: Algorithm

Environmental Conflict Resolution

- Categorized by conflict
  - Also requires city requirements
    $\{R_1 \ ... \ R_z\}$

Overall Resolver → Output: Action $\{A'_i...A'n\}$

CityGuard SUMO Sim City State $\{SV_1...SVn\}$
Physical Models (6)
PM1 ... PMn

Environmental Conflict D&R (ECDR)
Resolver | Resolver | Resolver
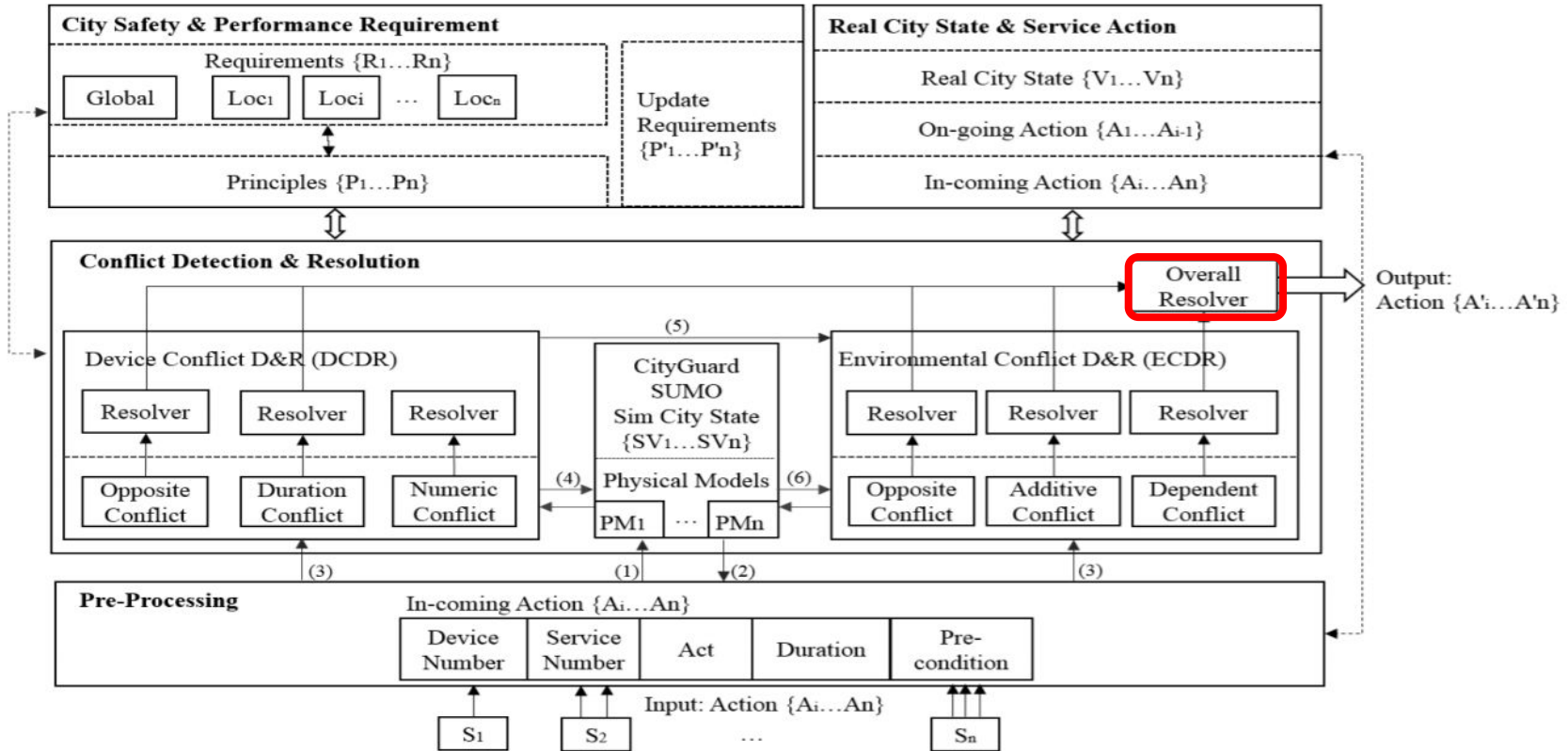Opposite Conflict | Additive Conflict | Dependent Conflict

System Conflicts: Algorithm

Environmental Conflict Resolution

- All results get passed to the overall resolver
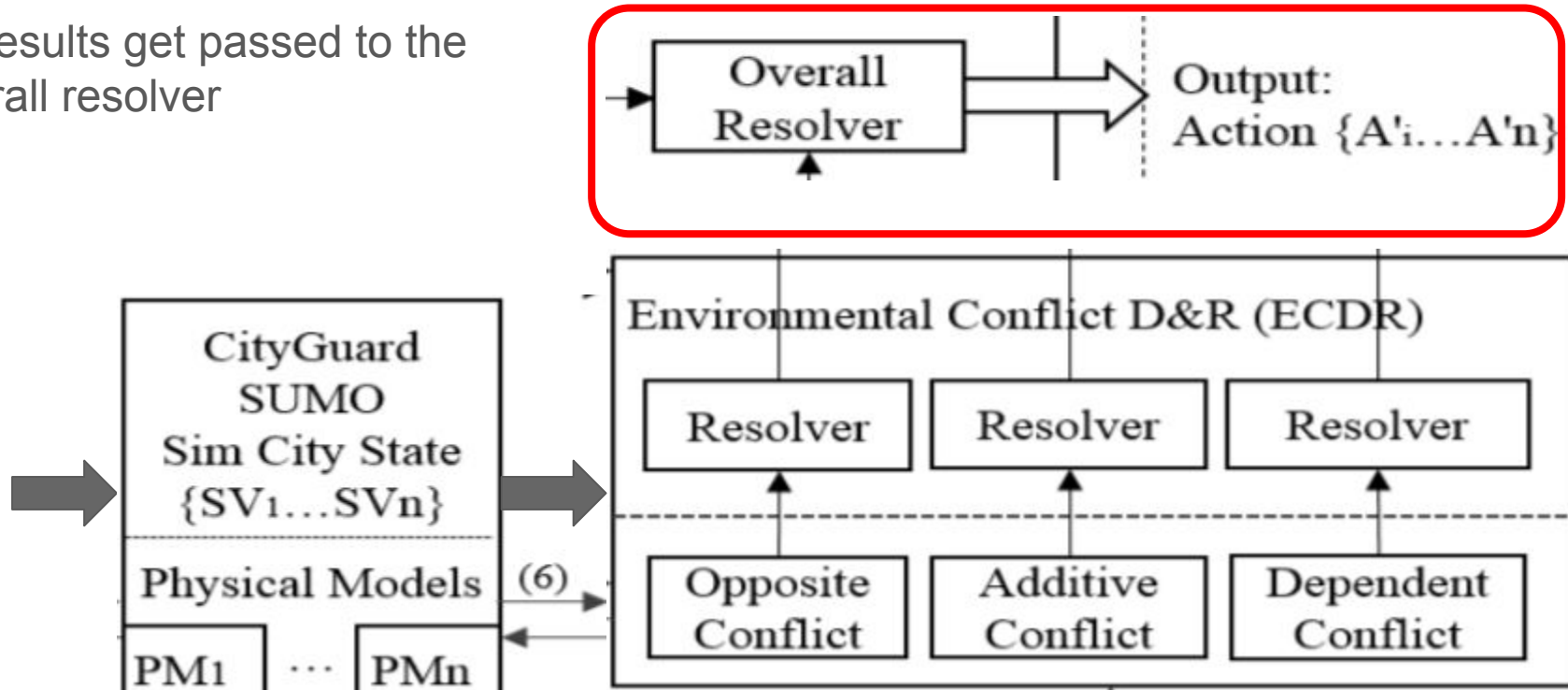
System Conflicts: Algorithm

System Conflicts: Algorithm

Overall Resolver

- Actions deemed safe by both resolvers are okayed
- Actions rejected by the ECRC are rejected
- Actions deemed unsafe by the Device Resolver:
  - Rechecked for possible solutions
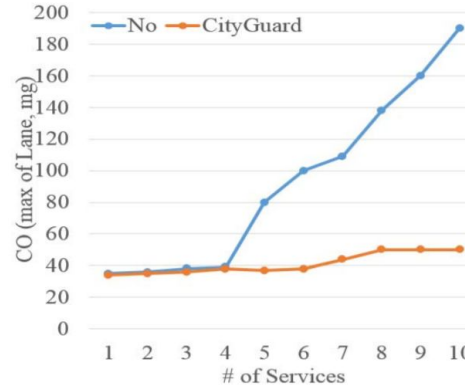 - If none found, it is rejected.

System Conflicts: CityGuard

Results

- Car collisions are eliminated with CityGuard

- Traffic jams greatly reduced

- Carbon Monoxide (air pollution) reduced significantly

- Vehicle waiting time greatly reduced

- Compromise of wait time for normal vehicles increased 2%
  - Still better than without a service though
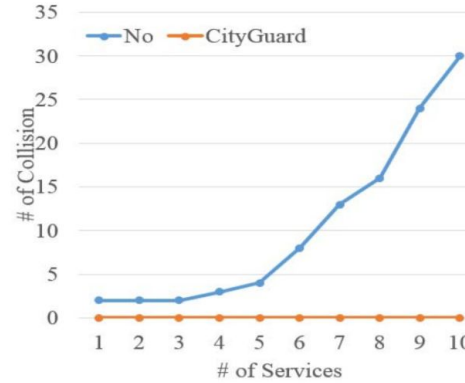
(1) Number of collisions
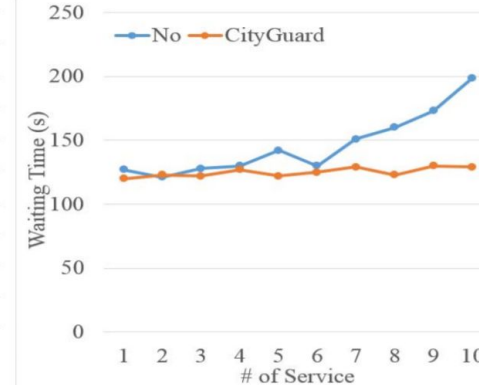
(2) Number of jam

(3) Maximum CO of lane
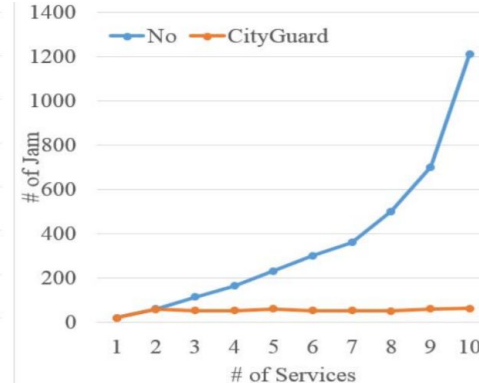
(4) Waiting time

Conclusion

Conclusion:

- Smart city networks must be extendable, decentralized, and seamless

- Conflicts of all kinds can be mitigated on the network with an algorithm.

- Smart city systems have the potential to greatly improve city performance and safety metrics.

Acknowledgements

Thanks to KK Lamberty, and Elena Machkasova for their advice and feedback

Questions

?

References

[1] A. Anjomshoaa, S. Mora, P. Schmitt, and C. Ratti. Challenges of drive-by iot sensing for smart cities: City scanner case study. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, UbiComp '18, pages 1112–1120, New York, NY, USA, 2018. ACM.

[2] N. M. Kumar, S. Goel, and P. K. Mallick. Smart cities in india: Features, policies, current status, and challenges. In *2018 Technologies for Smart-City Energy Security and Power (ICSESP)*, pages 1–4, March 2018.

[3] M. Ma, S. M. Preum, and J. A. Stankovic. Cityguard: A watchdog for safety-aware conflict detection in smart cities. In *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation*, IoTDI '17, pages 259–270, New York, NY, USA, 2017. ACM.

[4] R. Rivera, J. G. Robledo, V. M. Larios, and J. M. Avalos. How digital identity on blockchain can contribute in a smart city environment. In *2017 International Smart Cities Conference (ISC2)*, pages 1–4, Sep. 2017.

References

[5] J. M. Schleicher, M. Vögler, C. Inzinger, and S. Dustdar. Towards the internet of cities: A research roadmap for next-generation smart cities. In *Proceedings of the ACM First International Workshop on Understanding the City with Urban Informatics*, UCUI '15, pages 3–6, New York, NY, USA, 2015. ACM.

[6] Wikipedia. Federation (information technology) — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Federation%20(information%20technology)oldid=866156138, 2019. [Online; accessed 30-March-2019].

[7] Wikipedia. XMPP — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=XMPPoldid=889703389, 2019. [Online; accessed 30-March-2019].

[8] T. Yonezawa, T. Ito, J. Nakazawa, and H. Tokuda. Soxfire: A universal sensor network system for sharing social big sensor data in smart cities. In *Proceedings of the 2Nd International Workshop on Smart*, SmartCities '16, pages 2:1–2:6, New York, NY, USA, 2016. ACM.