New Models and Methods for Programming Cyber-Physical Systems (Keynote)

Rajesh K. Gupta
Halicioglu Data Science Institute and
Computer Science & Engineering
University of California, San Diego
California, United States
rgupta@ucsd.edu

Jason Koh
Computer Science & Engineering
University of California, San Diego
California, United States
jbkoh@eng.ucsd.edu

Dezhi Hong
Halicioglu Data Science Institute and
Computer Science & Engineering
University of California, San Diego
California, United States
dehong@eng.ucsd.edu

Abstract

Emerging cyber-physical systems are distributed systems in constant interaction with their physical environments through sensing and actuation at network edges. Over the past decade, the embedded and control systems community have vigorously pursued a vision of coupled feedback-controlled systems with a broad range of real-life applications from transportation, smart buildings to human health. These efforts have continued to push intelligent processing to edge and near-edge devices, provide new capabilities for improved sensing with high quality timing information, and establish limits on the quality of time and its impact on the stability of control algorithms etc.

It is now time to put these capabilities to use through the emerging "stack" of capabilities, software and systems for emerging applications such as interactive spaces, buildings, smart cities, etc. In this talk I will review our efforts related to pushing intelligent processing to edge or near-edge devices, our strategies to lighten the computational and memory demands of recognition tasks, and strategies to ensure high quality of timing information. I will focus on detailing our vision of how we can treat physical spaces and built environments as consisting of sensing, actuation, processing, and communication resources that are dynamically discovered and put to use through emerging meta-data schema and methods.

CCS Concepts • **Information systems → Mediators and data integration**; • **Computer systems organization → Sensor networks**; • **Software and its engineering → Distributed systems organizing principles**.

Keywords Cyber-Physical Systems, Metadata, Built Environments, Programming Platforms

1 Background

The Internet of Things is proliferating and the number of connected devices is expected to reach 75 billion by 2025 [7]. These devices are rapidly turning physical systems into cyber-physical systems (CPS). The CPS are by nature distributed systems in constant interaction with their physical environments through sensing and actuation at network edges. Thus coupled feedback-controlled systems have found a broad range of real-life applications from transportation, smart buildings to human health.

These developments have also continued to push intelligent processing of sensory data to the devices at the edge (such as camera, motion sensors) and near the edge (such as smart phones) of the network. While cloud computing provides a flexible resource pool for CPS components, it entails timing uncertainty through the stack because of the network latency, task scheduling, hardware instability, etc. By contrast, near-edge computing benefits from high quality timing information that is available near sensory interfaces, and so enables more complex applications such as real-time controllers and object recognition [3]. Further, near-edge processing can establish limits on the quality of time and its impact on the stability of control algorithms as timing certainty directly translates into parametric certainty of the feedback control loops. However, to realize the full potential of these capabilities, we would need a new "stack" of software and systems for emerging applications such as interactive spaces, buildings, smart cities, etc. This is particularly important for CPS applications used in our daily lives that must be resilient to changes in operating conditions, user and contextual uncertainties.
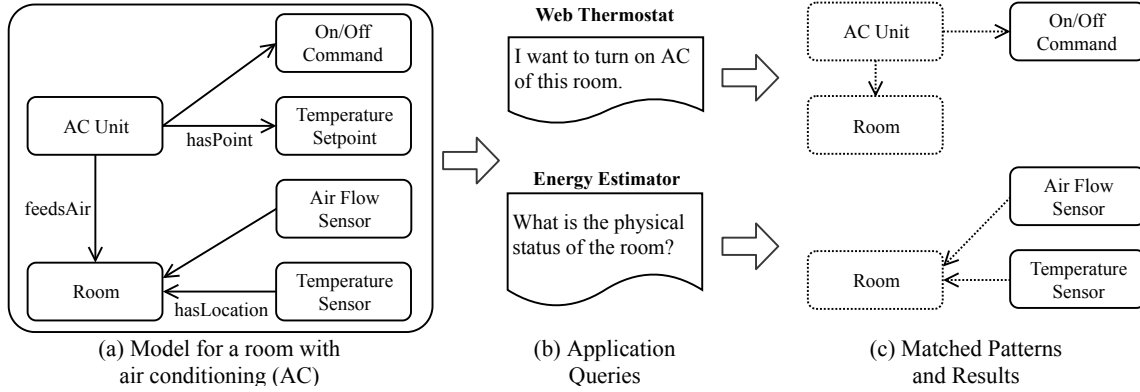
(a) Model for a room with air conditioning (AC) (b) Application Queries (c) Matched Patterns and Results

Figure 1. With a unified representation of the data and relevant resources in a room, different applications can exploit customizable patterns to find the right resources.

```
1  PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2  PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
3  PREFIX brick: <http://brickschema.org/schema/Brick#>
4
5  # Web Thermostat Query
6  SELECT ?room ?cmd WHERE {
7    ?room        rdf:type                 brick:Room.
8    ?ac_unit     brick:feedsAir           ?room.
9    ?ac_unit     brick:hasPoint           ?cmd.
10   ?cmd         rdf:type/rdfs:subClassOf*  brick:Command.
11 }
12 # Energy Estimator Query
13 SELECT ?room ?sensor WHERE {
14   ?room        rdf:type                 brick:Room.
15   ?sensor      brick:hasLocation        ?room.
16   ?sensor      rdf:type/rdfs:subClassOf*  brick:Sensor.
17 }
```

Figure 2. Example SPARQL queries corresponding to the applications in Fig. 1.(b).

2 Understanding Metadata

Real-time sensing, learning, and decision making for resilient CPS will very much depend upon our ability to automatically fuse and navigate through multiple data streams, where incoming individual data streams may not be as important in isolation as the relationship of data to the context. Furthermore, important information may be absent from a data stream but may exist elsewhere in the data as a whole.

However, sensory data is scattered both spatially as well as logically among various layers of data processing pipeline. Processing such distributed data while incorporating its context is a difficult challenge. Architecturally, we have already seen how representations and meanings attached to the *address* of data within the computing system, memory, storage, and networks have led to advances in computers. Whereas the address was constructed by the computer architects with definite structural information passed to the system builders, meta-data desperately needs to go beyond agreed-upon standards to formal models.

While the concept of meta-data has been around in several research communities such as middleware composition

frameworks, often these were a little more than annotations and reified data models. We are now at a tipping point of being able to conceptualize metadata as a means to generalized contextual capture, much the same way as the distinction between address and data has driven the current computing stack. Our recent initiative on the BRICK meta-data schema (for buildings) [1], along with follow-up work on Scrabble [5] and Plaster [6], has demonstrated the possibility of a general methodology to capture the data context. Labeling is a form of meta-data, so is schema generation or its extraction from a given (self-described) data. BRICK schema has shown how a commercial building or group of buildings can become a programming platform with the ability to migrate, compose, and analyze a new class of programs that are essentially spatio-temporal queries for discovery and use of building "resources". More generally, it represents the possibility to embed context with data streams, a perfect convergence opportunity through the creation of foundations and methods, not unlike conceptualization of regular languages and automata theory that led to the creation of compilers. Meta-data models have the potential to revolutionize the creation of new "programming" platforms and applications.

3 Commercial Building as a Programming Platform

Fig. 1 illustrates the concept of programming over BRICK schema. BRICK defines a complete taxonomy of entities and their relationships necessary for developing building applications. Top classes of the taxonomy are (data) Points, Location, and Equipment. Their actual instances and relationships are presented in the model in Fig. 1.(a). In a typical room with an Air Conditioning (AC) unit, there may be control points for the AC and sensors monitoring the room. Applications can flexibly locate the right resources for the model, as a web thermostat may look for control points while an energy estimation app needs to understand the current status of the room. These can be expressed in natural language as in

Fig. 1.(b), and we use SPARQL, a graph query language, to formally represent the patterns. SPARQL returns only matched entities as in Fig. 1.(c). The corresponding SPARQL queries are exemplified in Fig. 2. For resource discovery, different applications apply their own query patterns to the canonical model, enabling real portability of applications over CPS.

4 Conclusions

Having a formal information model is a vital first step towards achieving programmable cyber-physical systems. We have dedicated to building an entire software stack from data organization and federation to a building operating system with a security model for applications. We have explored various methods to efficiently extract contextual data from the raw data in existing systems. Our novel methods, including Zodiac [2], Building Adapter [4], and Scrabble, together deliver an efficient way to normalize existing unstructured metadata into standardized Brick labels, using techniques from natural language processing, deep neural network to transfer learning. Plaster enables the integration of various algorithms and data sources into workflows. The complex procedure finally creates a data ocean with formal context, ready to be used by applications and end-users. Our building operating system, BuildingBase, can host different third-party applications in safe containerized sandboxes while preserving only the right amount of access per application to the infrastructure.

Acknowledgments

References

[1] Bharathan Balaji, Arka Bhattacharya, Gabriel Fierro, Jingkun Gao, Joshua Gluck, Dezhi Hong, Aslak Johansen, Jason Koh, Joern Ploennigs, Yuvraj Agarwal, et al. 2016. Brick: Towards a unified metadata schema for buildings. In *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*. ACM, 41–50.

[2] Bharathan Balaji, Chetan Verma, Balakrishnan Narayanaswamy, and Yuvraj Agarwal. 2015. Zodiac: Organizing large deployment of sensors to create reusable applications for buildings. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. ACM, 13–22.

[3] Zhou Fang, Dezhi Hong, and Rajesh K Gupta. 2019. Serving Deep Neural Networks at the Cloud Edge for Vision Applications on Mobile Platforms. In *ACM Multimedia Systems Conference*.

[4] Dezhi Hong, Hongning Wang, Jorge Ortiz, and Kamin Whitehouse. 2015. The building adapter: Towards quickly applying building analytics at scale. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. ACM, 123–132.

[5] Jason Koh, Bharathan Balaji, Dhiman Sengupta, Julian McAuley, Rajesh Gupta, and Yuvraj Agarwal. 2018. Scrabble: transferrable semi-automated semantic metadata normalization using intermediate representation. In *Proceedings of the 5th ACM International Conference on Systems for Built Environments*. ACM, 11–20.

[6] Jason Koh, Dezhi Hong, Rajesh Gupta, Kamin Whitehouse, Hongning Wang, and Yuvraj Agarwal. 2018. Plaster: An integration, benchmark, and development framework for metadata normalization methods. In *Proceedings of the 5th Conference on Systems for Built Environments*. ACM, 1–10.

[7] Statista. 2019. Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions). .