2023 European Conference on Computing in Construction
40th International CIB W78 Conference
Heraklion, Crete, Greece
July 10-12, 2023

ENABLING SCALABLE DEPLOYMENT OF DATA-DRIVEN APPLICATIONS ACROSS BUILDING PORTFOLIOS

Dimitris Mavrokapnidis
University College London (UCL), London, UK

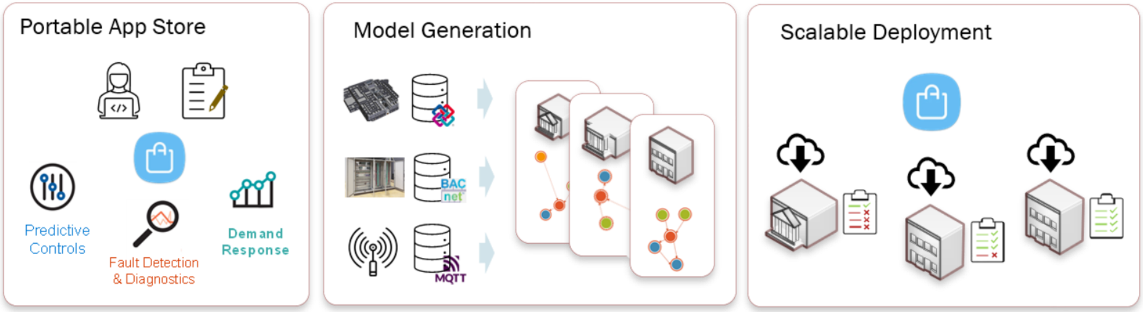

Figure 1: Overview of Research Contributions towards Enabling Scalable Deployment of Data-Driven Applications at Scale.

Abstract

Despite the promising benefits of data-driven applications to improve building performance, they are still developed on an ad-hoc basis, mainly due to the burden of discovering and reusing building data across deployment sites. Recent efforts in semantic data modelling aim to overcome these barriers, though application development remains building-specific, leading to bespoke configurations that cannot scale. This research seeks to establish a new regime of data-driven applications that are developed once and run across multiple buildings — similarly to the applications we download and use in our mobile phones. This research contributes to realising this vision in two ways: introducing (a) a method for automatic building metadata model generation through the lifecycle, and (b) a portable application development paradigm that offsets the burden of configuration in the authoring process. An overview of this vision and contributions are illustrated in Figure 1. These advancements are expected to overcome expertise barriers, reduce time for applications' configuration and deployment, and thus, accelerate their adoption at scale.

Introduction and Motivation

Buildings are producing an ever-growing amount of data throughout their lifecycle. These data can be processed to enable a wide range of applications — such as predictive controls, fault detection and diagnostics, and demand response (Burak Gunay et al. (2019)) — able to improve building performance, increase occupants' comfort, and deliver energy savings without significant capital outlay. Nevertheless, such applications have not been widely adopted mainly because their deployment is still ad-hoc, entailing repetitive efforts and expert knowledge to access, prepare and analyse data in each individual building (Bergmann et al. (2020)). Building data may be available in diverse sources and formats, either digital or analogue, creating silos produced by various stakeholders due to ad hoc data processing approaches and diverse lifecycle requirements (Luo et al. (2021)). As a result, discovering and reusing the necessary information requires a spectrum of skills and excessive manual interventions to process disparate data structures, fill the information gaps and configure applications at different deployment sites. Inevitably, this building-specific application development and deployment practice is challenging to scale-up.

Poor Data Delivery through the Building Lifecycle

Numerous metadata schemas have been devised to make building data more discoverable across the building lifecycle (Luo et al. (2021)). Industry Foundation Classes (IFC) provide 3D representations of building data to support the needs of design and construction phases. Whereas, operational requirements such as the identity of data sources, building assets, and the relationships between them may be represented using Project Haystack [1], Brick Schema (Balaji et al. (2018)), amongst others (Pritoni et al. (2021)). Yet, data acquisition for application development remains challenging. For example, Bhattacharya et al. reported an inability to run three simple diagnostics applications in 10 buildings due to a lack of the required semantic information richness (Bhattacharya et al. (2014)).
To increase data discoverability, more emphasis is required on how to harmonise and reuse metadata representations to deliver sufficient data quality to application developers. However, compared to the growing number of new vocabularies (Pritoni et al. (2021)), the efforts to harmonise existing building data representations, especially between design and operation, and reuse them from the viewpoint of application developers, are only few (Schneider (2019); Fierro et al. (2020)). Overall, it is still unclear what and how design-stage data can be authored, processed and delivered from the design phase to the O&M phases for the configuration of data analytics.

Missing a Portable App Programming paradigm

Recent analytics platforms offer several methods for enabling the *mass-customization* of applications. Building Application Stack (BAS) (Krioukov et al. (2012)) provides a fuzzy query interface that enables application authors to describe the various building entities and their relation-

[1] `https://project-haystack.org/`

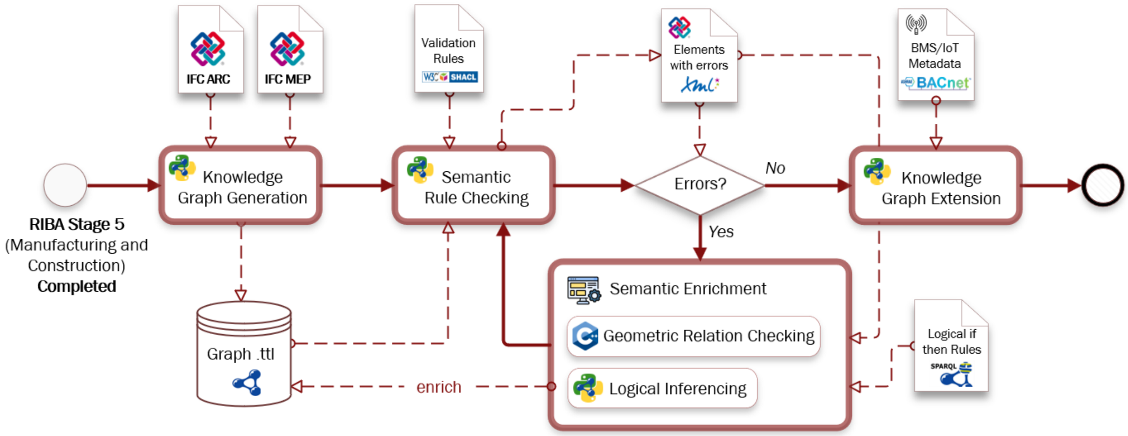

Figure 2: Design-stage Building Metadata Delivery

ships. BuildingDepot (Weng et al. (2013)) restricts user applications to predefined sets of building entities, shifting the manual mapping of those to corresponding template elements. Mortar (Fierro et al. (2019)) requires several lines of code to write queries and application configuration logic while SkyFoundry (`https://skyfoundry.com/`) and Energon (He et al. (2021)) use purpose-built query languages to reduce lines of code, but still require developer-driven reconfiguration between deployment sites. These approaches often address only the relationship between descriptions of the building and the actual telemetry, leaving the implementation of application portability to the developer. Overall, we still lack a paradigm for developing applications once and reusing them in multiple buildings without the need for manual configuration.

Enabling Building Analytics at Scale

To address the aforementioned limitations, this work (a) develops a metadata integration method that aims to deliver high quality metadata from design to operation to support the scalable deployment of data-driven analytics, and (b) proposes the adoption of a programming model for self-configurable building applications by offsetting the configuration costs in the development process.

Automatic Design-Stage Metadata Delivery

To enable information delivery from design to operation, detailed mappings [2] are specified between IFC, a prevalent building data standard in the design stages, and Brick Schema, a popular ontology that can be used for the configuration of building analytics during operational stages.
Figure 2 presents an overview of the proposed method and toolchain. To support the process, the Knowledge Graph Generator (KGG) was developed to extract "IfcDistribution" and "IfcSpatial" elements from IFC, transforms them into a directed graph using BOT [3], FSO [4] and Brick classes and relationships and load them in the knowledge graph database (Mavrokapnidis et al. (2021)). This initial version of the knowledge graph is then validated over predefined validation rules that express the required semantic information richness among the extracted Distribution and Spatial elements. Through this rule-checking mechanism, a list of IFC elements is produced that contain errors or, in other words, do not have the required relationships as defined by the validation rules.

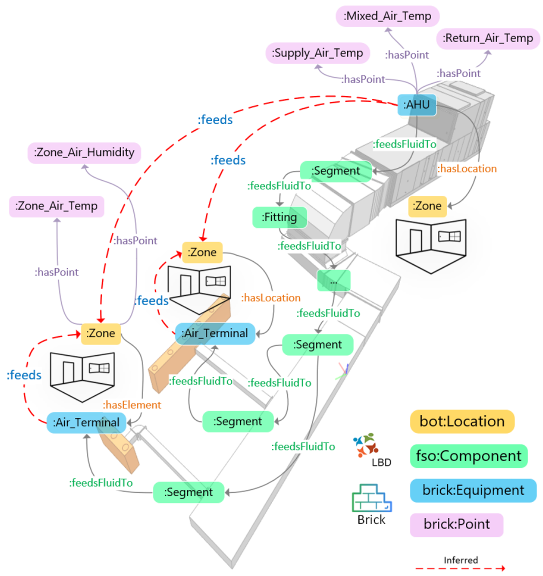

Figure 3: Illustrative example of IFC transformation into a RDF knowledge graph using BOT, FSO and Brick ontologies

To eliminate these errors, the Geometry Relation Checking (GRC) tool is developed to infer automatically missing associations between Distribution and Spatial components from IFC geometry inputs. Furthermore, a logical inferencing engine is used to infer additional semantics by implementing predefined if-then rules, while any unresolved issues can be resolved through manual inspection of the data and user enrichment with the aid of a Graphical User Interface (GUI). Finally, the knowledge graph can then be extended with virtual information (e.g. data points) using other metadata sources such as a BACNet network.
Figure 3 illustrates this data transformation and enrichment process. Overall, initial findings from implementing the method in a real-world project demonstrate its potential to eliminate semantic losses throughout the building lifecycle (Mavrokapnidis et al. (2023b)).

[2]`https://github.com/dimavrok/IFC-to-Brick/`
[3]`https://w3c-lbd-cg.github.io/bot/`
[4]`https://alikucukavci.github.io/FSO/`

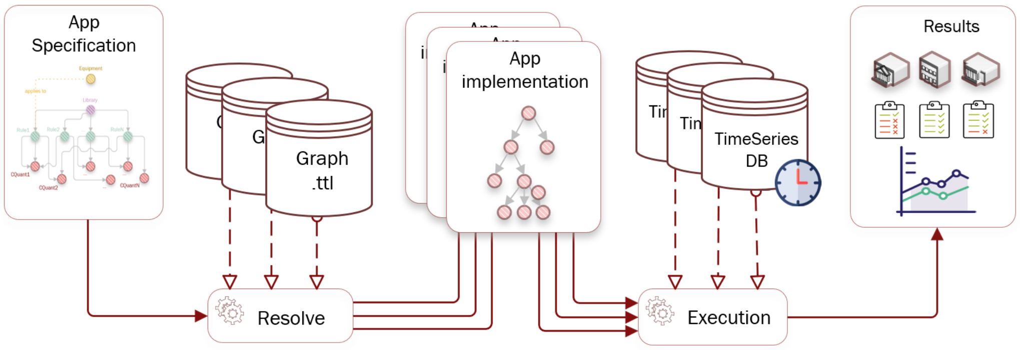

Figure 4: Software Architecture for Portable Analytics

Programming Portable Building Applications

Generating a semantically rich graph model is prerequisite one-time cost for data-driven applications. Still, configuring a data-driven application to run on a given building remains a time-consuming and site-specific process. This section proposes a novel programming model that separates the application logic and the configuration with specific data input and a software system that allows their self-configuration and execution across various building configurations, expressed as Brick models. As illustrated in Figure 4, a building application is specified once and resolved across different buildings (i.e. Brick graphs), producing a site-specific implementation that is then executed over a time-series database.

To enable portability, the burden of configuration is being offset in the app specification process. In mode detail, building applications can be expressed in terms of computational quantities. A computational quantity (CQ) is an abstract numeric quantity that appears in the building, such as the mixed air temperature of an air-handling unit. These quantities are categorised as: (a) *GraphCQ*: resolution observed directly by sensors or other digital I/O points from the building management system, (b) *ComputationalCQ*: computed indirectly through other observations or CQs, or (c) *DefaultCQ*: assumed to be a default value.

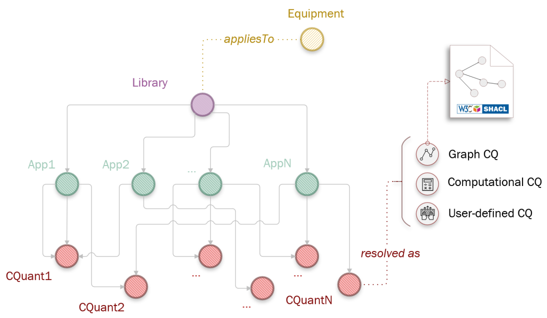

Figure 5: Portable App Specification using CQs

A CQ is expressed as a "decision tree" of possible resolutions, ordered by the accuracy and relevancy of each resolution. Resolving a CQ on a graph involves determining the most preferred resolution for that CQ and returning the corresponding value. All the potential Graph CQ resolutions are implemented as SHACL shapes during the specification of a portable application, incorporating various semantic descriptions of how the CQ may be expressed in a Brick model. The execution of the applications involves resolving each of the CQs on the target graph G by traversing the decision tree associated with each CQ. Figure **??** illustrates a high-level overview of how the proposed system resolves and executes a portable FDD library.

The programming model is employed to express and execute the Air-Handling Unit (AHU) Performance Assessment Rules (APAR) (Schein et al. (2003)) library in 2 buildings and 24 AHUs. Using the proposed model, the developer can express the application logic without accounting for the data source. For example, here is the Python implementation for expressing APAR rule 1 where expressions like Tsa(G) perform the resolution of the supply air temperature CQ on the graph G.:

```
# import provided definitions of common computational quantities
from APAR.quantities import Tsa, Tma, Tsf
def rule1(G: Graph) -> bool:
    return Tsa(G) < Tmat(G) + Tsf(G) - ϵ # ϵ is the error threshold
```

Table 1 includes results from executing the first 4 rules of the APAR ruleset Schein et al. (2003), identifying faulty operations across the various AHUs located in the two buildings. More detailed findings are provided by Mavrokapnidis et al. (Mavrokapnidis et al. (2023a)).

Table 1: Sample results from the execution of APAR Rules

APAR	Building	#AHUs with Fault
Rule1	Bldg1	9/17
	Bldg2	2/7
Rule2	Bldg1	5/17
	Bldg2	4/7
Rule3	Bldg1	11/17
	Bldg2	4/7
Rule4	Bldg1	5/17
	Bldg2	3/7

Conclusions

This research introduced (1) a workflow for the generation, checking and enrichment of design-stage metadata from IFC to Brick models, enabling their reuse by building application developers, and (2) a programming paradigm that eliminates the need to explicitly query metadata models in order to re-configure their applications across different buildings. Overall, this paper summarised findings and demonstrated the potential of these advancements to reduce manual and repetitive tasks during data discovery and application development, thereby accelerating the adoption of data-driven building applications.

Acknowledgments

The research leading to these results has been funded by the CBIM-ETN funded by the European Union's Horizon 2020 programme under the Marie Skłodowska-Curie grant agreement No 860555. I am grateful to Dr. Ivan Korolija and Dr. Dimitrios Rovas for their guidance and support, and Dr. Gabe Fierro, Dr. Pieter Pauwels and Dr. James O' Donnell for inspiring and advising this piece of work.

References

Balaji, B., Bhattacharya, A., Fierro, G., Gao, J., Gluck, J., Hong, D., Johansen, A., Koh, J., Ploennigs, J., Agarwal, Y., Bergés, M., Culler, D., Gupta, R. K., Kjærgaard, M. B., Srivastava, M., and Whitehouse, K. (2018). Brick : Metadata schema for portable smart building applications. Applied Energy, 226:1273–1292.

Bergmann, H., Mosiman, C., Saha, A., Haile, S., Livingood, W., Bushby, S., Fierro, G., Bender, J., Poplawski, M., Granderson, J., and Pritoni, M. (2020). Semantic interoperability to enable smart, grid-interactive efficient buildings. United States. Research Org.: Lawrence Berkeley National Lab. (LBNL), Berkeley, CA (United States).

Bhattacharya, A., Culler, D., Hong, D., Whitehouse, K., and Ortiz, J. (2014). Writing scalable building efficiency applications using normalized metadata: demo abstract. In Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings, BuildSys '14, pages 196–197, New York, NY, USA. Association for Computing Machinery.

Burak Gunay, H., Shen, W., and Newsham, G. (2019). Data analytics to improve building performance: A critical review. Automation in Construction, 97:96–109.

Fierro, G., Prakash, A. K., Mosiman, C., Pritoni, M., Raftery, P., Wetter, M., and Culler, D. E. (2020). Shepherding Metadata Through the Building Lifecycle. In Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, BuildSys '20, pages 70–79, New York, NY, USA. Association for Computing Machinery. event-place: Virtual Event, Japan.

Fierro, G., Pritoni, M., Abdelbaky, M., Lengyel, D., Leyden, J., Prakash, A., Gupta, P., Raftery, P., Peffer, T., Thomson, G., and Culler, D. E. (2019). Mortar: An open testbed for portable building analytics. ACM Trans. Sen. Netw., 16(1).

He, F., Deng, Y., Xu, Y., Xu, C., Hong, D., and Wang, D. (2021). Energon: A data acquisition system for portable building analytics. In Proceedings of the Twelfth ACM International Conference on Future Energy Systems, e-Energy '21, page 15–26, New York, NY, USA. Association for Computing Machinery.

Krioukov, A., Fierro, G., Kitaev, N., and Culler, D. (2012). Building application stack (BAS). In Proceedings of the Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, BuildSys '12, pages 72–79, New York, NY, USA. Association for Computing Machinery.

Luo, N., Pritoni, M., and Hong, T. (2021). An overview of data tools for representing and managing building information and performance data. Renewable and Sustainable Energy Reviews, 147:111224.

Mavrokapnidis, D., Fierro, G., Korolija, I., and Rovas, D. (2023a). A Programming Model for Portable Fault Detection and Diagnosis. In Proceedings of the 14th ACM International Conference on Future Energy Systems (e-Energy '23), June 20–23, 2023, Orlando, FL, USA, e-Energy '23, Orlando, Florida, USA. Association for Computing Machinery.

Mavrokapnidis, D., Katsigarakis, K., Pauwels, P., Petrova, E., Korolija, I., and Rovas, D. (2021). A linked-data paradigm for the integration of static and dynamic building data in digital twins. In Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, BuildSys '21, page 369–372, New York, NY, USA. Association for Computing Machinery.

Mavrokapnidis, D., Lilis, G., Katsigarakis, K., Ivan, K., and Rovas, D. (2023b). Semi-automated extraction of hvac system topology from imperfect building information models.

Pritoni, M., Paine, D., Fierro, G., Mosiman, C., Poplawski, M., Saha, A., Bender, J., and Granderson, J. (2021). Metadata Schemas and Ontologies for Building Energy Applications: A Critical Review and Use Case Analysis. Energies, 14(7):2024. Number: 7 Publisher: Multidisciplinary Digital Publishing Institute.

Schein, J., Bushby, S. T., Castro, N. S., and House, J. M. (2003). Results from field testing of air handling unit and variable air volume box fault detection tools. Technical Report NIST IR 6994, National Institute of Standards and Technology, Gaithersburg, MD.

Schneider, G. (2019). Automated Ontology Matching in the Architecture , Engineering and Construction Domain-A Case Study. Conference: The 7th Linked Data in Architecture and Construction Workshop.

Weng, T., Nwokafor, A., and Agarwal, Y. (2013). BuildingDepot 2.0: An Integrated Management System for Building Analysis and Control. In Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings, BuildSys'13, pages 1–8, New York, NY, USA. Association for Computing Machinery.