Title: *Building HVAC Control Knowledge Data Schema – Towards a Unified Representation of Control System Knowledge*

Authors:

Yan Chen^{1,*}, Stephen J. Treado², John I. Messner²

*corresponding author: yan.chen@pnnl.gov

Affiliation:

1 Pacific Northwest National Laboratory

2 Department of Architectural Engineering, The Pennsylvania State University

Present address: 902 Battelle Boulevard, Richland, WA 99352

Abstract:

Building control systems for Heating, Ventilation, and Air Conditioning (HVAC) play a key role in realizing the functionality and operation of building systems and components. Building Control Knowledge (BCK) is the logic and algorithms embedded throughout building control system. There are different methods to represent the BCK. These methods differ in the selection of BCK representing elements and the format of those elements. There is a lack of standard data schema, for storing, retrieving, and reusing structured BCK. In this study, a modular data schema is created for BCK representation. The data schema contains eleven representing elements, i.e., control module name, operation mode, system schematic, control flow diagram, data point, alarm, parameter, control sequence, function, and programming code. Each element is defined with specific attributes. This data schema is evaluated through a case study demonstration. The demonstration shows a new way to represent the BCK with standard formats.

Keywords:

Building control, modularization, knowledge, representation, BIM, HVAC,

1. Introduction

The concept of building control knowledge (BCK) can be defined as information that conveys the functionality, detailed control logics, algorithms, sequences, programming code, and the hierarchical structure of a (number of) controller(s). BCK plays an indispensable role in Building Automation System (BAS), and represents the control decision-making process for building HVAC system operation.

Practically, the development of BCK starts with HVAC and control system designers. At this stage, the BCK is represented in the format of control sequences (also called sequences of operation). For example, the control sequences for a zone temperature set point may be described as “during the occupied mode, the set point shall be 23 °C for cooling and 21 °C for heating; during the unoccupied mode, the set point shall be 29 °C for cooling and 13 °C for heating”. Then, control sequences are interpreted into programming codes, and implemented into BAS by control programmer. After that, the whole HVAC system is tested through building commissioning and used for normal operation. During building operation, control sequences and other relevant BCK documents are used for system testing, fault detection and diagnosis (FDD). Thus, a well-documented BCK representation is important for control system design, control code programming, commissioning, operation, and maintenance.

Efforts have been made for information modeling related to BAS, where the BCK are implemented. For example, the open information exchange standard *ISO 16739:2013 Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries* have adopted information elements to represent BAS as part of their schema [14]. Also, there have been recent attempts to develop IFC model view definitions to document standard exchange elements for control information (e.g., the Building Automation Modeling information exchange (BAMie) Model View Definition (see buildingSMART International [8]). The Building Energy Data Exchange Specification (BEDES) was developed by U.S. Department of Energy and it included a section on “Control and Operation” [20]. Construction-Operations Building Information Exchange (COBie) was developed to support the collection of information during the design, construction and operation stages to increase the data

availability [13]. An object-oriented product model was developed by Lucas, et al. [17] to support healthcare facility information management. Liu, et al. [16] presented extensions to the Information Delivery Manual approach and the results of using it to identify information requirements for performance analysis algorithms of HVAC systems, especially for FDD. Dong, et al. [12] developed a BIM enabled information infrastructure for FDD, which focused on data schema for the information exchange process. Dong's work also notified the current BIM effort lacks of detailed data schema for the control sequences representation. These studies analyzed the building system, and developed information model, or implemented the information model for applications. However, there is still a lack of study focused on BCK representation, although Lee, et al. [15] has identified the technical needs for establishing point-naming conventions, standardization of control sequences. Currently, there are several challenges in the area of BCK representation that require investigation.

First, different elements are used to represent the BCK. There is currently no standard about the selection of these elements, and the definition of their attributes. For example, CtrlSpecBuilder by AutomatedLogic used control sequence, system schematics, data points, and alarms [4]. *Library of System Control Strategies* used control sequence, system schematics, control flow diagrams, data points, and alarms [18]. EquipmentBuilder used control sequences, system schematics, data points, alarms, and programming code [5]. This discrepancy among different methods of selecting BCK representing elements creates inconsistency of the BCK representation. As a result, it adds difficulty in understanding and interpreting the BCK documents. It can be much easier to understand and interpret the BCK if there is a standard list of BCK representing elements.

Second, for the selected BCK representing elements, there is a lack of standard format. The documentation of control sequences can be viewed as an example. Based on *ASHRAE Guideline 13-2007 Specifying Direct Digital Control Systems* [2], there are two styles of organizing control sequences: 1) Operation mode style, and 2) Component style. In the Operation mode style, sequences are grouped by

different operation modes, and within each mode, it describes how each control component behaves. In the Components style, sequences are grouped by different components, and within each component, it describes how this component behaves at different operation modes. Each style has its pros and cons, and mixed uses of the two styles need to be avoided for control sequences documentation. While the ASHRAE Guideline 13 [2] gave examples of control sequences written in each style, there is no vigorous language structure or template to follow. In practice, there are mixed uses of both styles. That means that control sequences are partially written in the Operation mode style, and partially in the Component style. The mixed uses of styles can increase ambiguity issue. Besides, other relevant problems include incomplete control sequences, copying and pasting previously written sequences, sequences contradicting with other supplementary drawings, mistakenly specifying the control sequence, and programming errors [9,22]. These problems affect the normal operation of building HVAC system, which can contribute to building energy waste, indoor air quality problem, and thermal discomfort. Thus, there is also a need to define a standard format for each BCK representing element.

Third, BCK representation of larger control system is usually more complex, and it is difficult to interpret and understand. In the U.S., approximately 90% of the building floor space is contained within medium and larger size commercial building, i.e., medium size (5,000-50,000 ft², 41%) and large size (>50,000 ft², 49%) [21]. Most current documenting resources present the BCK of the entire HVAC system without systematically modularizing the BCK into smaller modules (e.g., in *Sequence of Operation for Common HVAC System* [3]). A few commercial software tools represent their BCK based on some sorts of modularization, e.g., EquipmentBuilder [5] is based on equipment and components for modularization. However, there is no clear definition regarding to the modularization at building level control, system level control, and equipment level controls.

The goal of this paper is to define an open knowledge data schema for storing, retrieving, and reusing structured BCK. Specifically, Section 2 provides a review of existing BCK representing, Section 3

presents a method to standardize BCK representing, Section 4 demonstrates the application of BCK data schema via a case study, Section 5 provides additional comparison and discussion, and Section 6 summarizes the findings and conclusions.

2. Existing representations and formats

To further analyze the existing problems, a number of existing resources are reviewed and compared for BCK representing. These include published guidelines, industrial application tools, and books. The following list provides a brief description for each selected document.

1. *ASHRAE Guideline 13-2007* is a guideline for specifying the direct digital control (DDC) system [2]. It focuses on the DDC system design procedure, but also discusses the formats of control drawings, control sequences, and objects list.
2. CtrlSpecBuilder is a web-based application tool developed by AutomatedLogic [4]. It provides control system design guideline and can generate detailed control sequences for most of the HVAC system types.
3. EquipmentBuilder for Educators 5.5 is a downloadable free software developed by AutomatedLogic [5]. It can generate control sequence, system schematics, and WebCtrl® programming code for a number of pre-defined HVAC systems and equipment.
4. *Library of system control strategies* is a reference book for specifying, developing and configuring control strategies by the Building Services Research and Information Association [18]. It provides control sequences specification, highlighted by its detailed control flow diagram for easier understanding.
5. *ASHRAE Sequence of Operation for Common HVAC System* is a reference CD-ROM for specifying the control sequence [3].

Based on the documents above, a list of BCK representing elements are identified (Table 1). Several elements are frequently used among these documents, including control module name, control objective, operation mode, system schematic, inputs, outputs, trends, and alarms. Other elements are only mentioned in one or two documents, i.e., functions/algorithm, control flow diagram, and programming codes. In order to maintain the consistency of BCK documentation, the selection of representing elements need to be consistent and comprehensive to represent BCK.

Table 1 Comparison of existing published documents for BCK representation.

Representing Elements	1	2	3	4	5
Control module name	-	**	*	**	***
Control objective	*	-	-	***	***
Operation mode	*	*	*	***	*
System schematic	***	***	***	***	***
Control flow diagram	*	-	-	***	-
Object list (inputs/outputs/parameters)	***	***	***	***	***
Alarms	***	***	***	*	***
Control sequence	**	**	**	**	**
Functions/algorithms	*	-	-	-	-
Programming code	*	-	***	-	-
***: listed with standard format, clear and unambiguous					
**: listed with standard format, but not well-organized, and have ambiguity issues					
*: listed without standard format					
-: not listed					

The existing formats of the representing elements are further discussed here. Particularly, there are style and format inconsistency for control sequences documentation due to the lack of standardization [15]. Control sequences or programming codes are often organized in a mixed style (i.e., mixed use of both operation mode style and component style). For example, when using the Automated Logic EquipmentBuilder to generate control sequences and programming codes for VAV terminal unit, the

layout of control sequences document (in Table 2) have mixed uses of *operation modes* (i.e., Zone Optimal Start, Zone Unoccupied Override), *components* (i.e., Reversing Variable Volume Terminal Unit, Reheating Coil Valve), and *other set point configurations* (i.e., Run Conditions, Zone Set point Adjust, Demand Limiting). Similar problem happened for the layout of programming codes. This example represents the format inconsistency issue, which can obstruct the interpretation and implementation of these problematic BCK documents.

Table 2 Layout structure of control sequences and programming codes (a VAV terminal unit example from EquipmentBuilder for Educators).

Control Sequences (Viewed in MS Word: the alarm and data point sections are excluded)	Programming Code (Viewed in EIKON: the alarm and data point sections are excluded)
<ul style="list-style-type: none"> • Run Conditions – Scheduled • Minimum Ventilation on Carbon Dioxide (CO₂) Concentration • Demand Limiting – Zone Set point Optimization • Zone Set point Adjust • Zone Optimal Start • Zone Unoccupied Override • Reversing Variable Volume Terminal Unit – Flow Control <ul style="list-style-type: none"> ○ Occupied ○ Unoccupied • Reheating Coil Valve 	<ul style="list-style-type: none"> • Zone - Run Conditions – Scheduled <ul style="list-style-type: none"> ○ Outgoing Requests to Run Interlocked Equipment <ul style="list-style-type: none"> ▪ Request Heat Source ▪ Request Cool Source ▪ Zone Occupied for - Minutes • Zone CO₂ Control • Airflow Control – Reversing – Internal Actuator <ul style="list-style-type: none"> ○ Outgoing Requests to Run Interlocked Equipment • Heating Control

In summary, a list of BCK representing elements is identified from existing documents. Their associated format issues need to be solved through a standardized and well-defined formatting.

3. A new approach – Modularized BCK representing

A Modularized BCK (M-BCK) data schema is developed to standardize the BCK representing. This approach includes two major parts (see Figure 1). The first part is about modularizing BCK representing. It targets to transform a one-big-complex BCK into a number of small BCK modules. The second part is

about the standardizing of representing elements for a typical control module. Then, standard formats for each element's attributes are provided. The following sub sections discuss each part in details.

Figure 1 Overview of the BCK modularization.

3.1. BCK modularization

Modularization of BCK is intended to reduce system complexity based on the structural or functional similarities in BAS. This transforms a big and complex BCK into a number of small and simple modules organized in a rational structure. Each BCK control module has its own inputs, parameters, embedded control logic, and the outputs. It functions as a complete system, yet its inputs might depend on other control modules, or its outputs might be used as inputs for other control modules. Thus, it becomes natural and feasible to modularize the BCK representation based on the relationship of control variables.

An ontology analysis of the BCK illustrates this rational structured feature (see Figure 2). As can be seen, the control modules are classified into three layers (i.e., building layer, system layer, and local layer). The building layer manages control modules that are functioned to determine the optimal point at building level (e.g., identify optimal point for building energy usage and operational cost). The system layer manages control modules that are functioned to determine the set point at system level, (e.g., modify supply air pressure set point). The local layer manages control modules that functioned to maintain the set point by actuating hardware (e.g., control supply fan speed).

The modularization process is designed based on the determination of control variables via ontology analysis. As shown in Figure 3, the first major step is to identify all the hardware control variables (e.g., supply fan speed). For each hardware control variable, a local control module is created and constructed with its sensor inputs, virtual variable inputs, and parameters. Virtual variables are outputs from system or building control modules (e.g., supply air temperature set point determined in the system control module), and may be used in local or system control modules as their inputs. For each virtual variable identified, a system/building control module is built for this virtual variable. Similar to the local control module construction, sensor inputs, virtual variable inputs, and parameters are identified for this system/building control module. If another virtual variable is identified in this process, then another system/building control module will be created. This modularization process continues to populate control modules until all virtual variables and hardware control variables have their dedicated control modules. In this way, each control module determines its control output as a functional unit. An input variable for one control module might come from the output of another module.

This modularization is control variable oriented, while the modularization in building & HVAC system modeling is typically equipment and component oriented (such as TRNSYS, Modelica Building System library). This modularization approach follows naturally the logic flow of control decision process. Therefore, each control module is independent as a functional unit, yet integrated as the connection of information flow. Implementation of the modularization process is further demonstrated in Section 4.1.

Figure 2 Ontology analysis of building control system structure.

Figure 3 Control module classification process.

3.2. *BCK module elements identification and formatting*

Based on the previous analysis of representing elements in existing resources (see Table 1), the format quality of each representing element are further discussed here. If an existing format is well defined without ambiguity, this format will be used in the M-BCK data schema; otherwise, new format will be created. This section discusses the selection of each element and their attributes (in Table A.1).

3.2.1. Control module name

Control module name element has two attributes, the symbol, and the narratives. The symbol attribute is a unique ID, which can be used to store and manage related information in a database. The narratives attribute uses plain English to describe the controlled components, variables, or systems, and the type of control algorithms.

3.2.2. Control objective

Control objective element is used to explain the functionality of control module. It has two attributes. The narratives attribute describes the control purpose. The controlled system/actuator ID attribute represents the controlled physical component. Each control module is associated with an actuator or system.

3.2.3. Operation mode

Operation mode element is used to describe system operation for any given operating stage. It helps the explanation and documentation of control sequences by the operation mode. The format of Operation mode is mentioned in the context of control sequences as a subtitle (or clause name). There is usually a list of typical operation modes, such as occupied, unoccupied, heating, cooling, optimal start, freeze protection, etc. The formatting of this element followed the style in the *Library of system control strategies* by Martin and Banyard [18], and defined a list of operation mode.

3.2.4. System schematic

The system schematic element is to explain the layout and connection of systems and components configuration. For the modularized BCK representation, it is unnecessary to have separate system schematics for each of them. The format of the drawings needs to follow the ANSI/ASHRAE Standard 134 about the graphic symbols for HVAC systems [1].

3.2.5. Control flow/logic diagram

The control flow diagram, or called control flow chart sometimes, is used to graphically illustrate control decision process. It is more intuitive than narrative sequences, and this usually helps to assist programming and interpretation. It is recommended in the ASHRAE Guideline 13 as “use tables and diagrams where possible to assist in conveying sequence logic” [2]. It was also suggested to be included for control sequence representation by Utterson [22]. However, only the *Library of control system strategies* includes the control flow diagram as one of the BCK representing elements [18]. Our approach includes control flow diagram as a necessary BCK representing element. The format of the drawings shall follow the ANSI/ASHRAE Standard 134 about the graphic symbols for HVAC systems [1].

3.2.6. Control sequence

The control sequence element is the narrative description of control knowledge without detailed calculation algorithms. As mentioned previously, control sequence typically can be organized by operation modes, or by components. Based on the reviewed documents, most of the control sequences have main sections organized by components, and then within each main section, it is organized by operation modes. Our approach uses operation mode based structure within each control module, since the control modules are organized by control functionalities and components through modularization process. For the control sequence language, there are currently no standard languages. Researchers from computer science have studied and developed several formats to better represent general algorithm in

narratives, including *pseudocode*, *prosecode*, *literate code* [24]. Here we select *prosecode* since it can concisely represent the control sequences with a structured way. Prototype and language for the control sequence is created (in Table A.2).

3.2.7. Data point

Data point element is to describe the BCK module related input, output, and parameter. All the listed documents (in Table 1) include this element with a standard format. Our approach adopts the existing format for this element.

3.2.8. Alarm

Alarm element is to describe the list of abnormal status that should be alerted to the BAS and system operators. The existing format of Alarm is in narratives within the context of control sequences, as a separate section. The existing format can be extracted as “Alarm name: triggering condition”. Our approach adds the symbol to represent the alarm in a programming language, and includes “logic” to explain the triggering condition.

3.2.9. Function/algorithm

Function /algorithm element is to provide a detailed description of the logics/or algorithms used to determine control outputs, which can be valuable to the field operators, commissioning engineers, and operational analyst. This element is recommended by the ASHRAE Guideline 13 as “show the formulas in the sequences if they are to be used in calculations” [2], but only one of the reviewed documents includes this element. Our approach includes this Function/algorithm element, because it can provide information that is necessary for control programming, commissioning, FDD, and other system operational analyses. This might bridge the gap between control designers/manufacture and field operational engineer, which can contribute to resolve the deployment difficulty for advanced or optimal control strategies realized by a number of studies [7,11]. The format of this element is created to include

the Symbol, Narratives, Formula Inputs, Formula Outputs, Formula Parameters, Formula, and References.

3.2.10. Programming code

Programming code is an element that represents the BCK in programming language. Reviewed documents don't include this element except the EquipmentBuilder. The benefits of using this element include 1) this can potentially contribute to controller reconfiguration, and self-configuration as discussed in [10]; and 2) if a standardized programming format is defined and provided, it can eliminate programming errors. Therefore, this element is included in the lists of our BCK module elements. The format of programming code generally depends on manufacture and programmer's style. There are three commonly-used types, namely, text-based, graphic-based, and menu-driven based [2]. They are compared in Table A.3. The first two types are more flexible and robust, while the menu-driven depends on pre-programmed sequences. In practice, each type is used by different manufactures (e.g., AutomatedLogic uses graphic-based in WebCtrl, Siemens uses text-based). Thus, the format of Programming code element is not limited to any type, but depends on the users and manufactures. For testing and demonstration purpose, StateFlow (in Simulink) based programming is used in the later demonstration.

As can be seen, a list of representing elements are identified, including Control module name, Control objective, Operation mode, System schematic, Control flow diagram, Data point, Alarm, Control sequence, Function/algorithm, and Programming code.

3.3. *Relation to the overall research framework*

This M-BCK data schema developed in this work is envisioned to have the capability to support the development of a control self-configuration framework (shown in Figure 4). Particularly, this data schema will be used to create a BCK database, which will be designed as the BCK source center for a

building control self-configuration. This building control self-configuration will interact with the BCK database to obtain the applicable control modules for a specific system. This self-configuration process is envisioned with three steps 1) acquire system profile from a specific building HVAC system, 2) query the BCK database to identify the applicable control modules, and 3) implement control modules for the building control system. While M-BCK data schema is the focus of this paper, the rest of the work for building control self-configuration framework will be discussed in the future.

Figure 4 Overview of the self-configuration process interacting with the BCK database.

4. Demonstration through an example system

Here the narrative that describes how a single-duct VAV system operates could increase the appearance of reliability. In this section, an example is used to demonstrate the BCK representation following the M-BCK data schema. This example is based on the control system for single-duct VAV multizone system. This type of system is commonly used in commercial buildings. The major system components include the AHU (heating coil, cooling coil, and supply fan), VAV terminal units (damper and reheating coil), and mixing box. The original control sequences and programming code for this system type are obtained

from the *EquipmentBuilder for Educators 5.5*. This is used as raw data for modularizing and constructing the BCK representing elements based on the M-BCK data schema.

4.1. Modularization

The modularization of the BCK for this example system follows the process described in Figure 3. Based on the given HVAC system and components, a list of hardware control variables are identified. Then, each hardware control variable is further analyzed using the ontology analysis (shown in Figure 2), which determines the related virtual control variable(s). In this example, due to limited space, detailed analysis for the part of VAV terminal unit is presented in Figure 5, and ontology analysis of the whole system is shown in Figure 6. As shown in Figure 5, in order to determine the VAV damper opening, Control module F needs to know the values of four inputs/parameters (i.e., manual override, zone heating/cooling demand, VAV min/max airflow set point, and CO₂ override). Among them, the manual override signal is the sensor measurement or direct manual inputs, while zone heating/cooling demand, VAV min/max airflow set point, and CO₂ override are virtual control variables determined by other control modules. i.e., Control module B determines zone heating/cooling demand; Control module E determines VAV min/max airflow set point; Control module D determines CO₂ override. Then, for Control module B, zone temperature set point is a virtual variable, which is determined by Control module A. Thus, for a VAV terminal unit, a hardware control variable (VAV damper position) depends on at least four virtual control variables (i.e., zone heating/cooling demand, VAV min/max airflow set point, CO₂ override, and zone temperature set point).

Using the same modularization method, seven hardware control variables, and eighteen virtual control variables are identified (as can be seen in Figure 6). Each control variable is the foundation to construct its control module. In this way, the whole system BCK for this demonstration case is modularized into twenty-five individual functional-independent yet strongly related control modules.

Figure 5 Detailed ontology analysis for the VAV terminal unit (part of the whole system).

Figure 6 Ontology analysis of the control variables for a single-duct VAV multizone system (the whole system).

4.2. Construction of the representing elements for control modules

After modularization, the format guideline (see Table A.1) is followed to construct the BCK representing elements for each control module. Each module has its own content, while sharing the same structure (list of representing elements) and formatting style. We present in Appendix B with one control module demonstrating the details, i.e., *VAV #1 damper control module (L-Z1-VAV_D)*.

In *VAV #1 damper control module*, its *Control module name symbol* is designated as L-Z1-VAV_D. This designation indicates Z1 VAV damper position control is a local control module. Its *Control objective* means this control module is to maintain the zone temperature set point. In the *Operation mode* section, there are five operation modes. In the *Data point* section, it indicates that there are five input variables and one output variable. The *Alarm* section indicates there are four alarms associated with this control module. In the *Sequence* section, the sequence is organized by the five operation modes, and it can be further interpreted with the *Control flow diagram* and the *Programming code*. The *Control flow diagram* provides a graphic explanation of how control decision is made based upon the values of input variables. For example, if the manual override is disabled, the CO₂ override is disabled, and cooling demand is larger than 10%, then the control decision will be “enter cooling mode”. After the cooling control action is activated, this control module will cycle again by starting over from the first step to decide what would be the new control action at this cycle. The *Data point* section lists the relevant inputs, parameters, and outputs. The Alarm section and Control sequence section provide the narrative description of the alarm triggering logic and the control decision process at each operation modes. In the *Formula* section, two calculation formulas are identified and represented explicitly. The *Programming code section* shows a graphic control programming developed in MATLAB Stateflow as an example.

The other twenty-four control modules are represented using the same format. In this way, by applying the modularization process and following the defined elements format, the BCK for the example HVAC system is modularized and reformatted into twenty-five control modules. Each of them is packaged as a functional

independent module. Integrated together they realize the functionality of heating, cooling, and ventilation with the choice of optimization for different performance measures.

5. Comparison and discussion

From this demonstration example, it can be seen that this M-BCK data schema appropriately represented building control knowledge. Comparing to the existing approaches previously mentioned in Table 1, the M-BCK data schema has a number of features (shown in Table 3).

Table 3 Comparison of the M-BCK data schema and existing ones.

Existing Approaches	The M-BCK data schema
<ul style="list-style-type: none"> • Lack of standard list of representing elements for BCK. • Use different formats in different BCK documentations. • Lack of high-level relationship analysis. • Usually do not have control flow diagram. • Within each control sequences and programming code file, there are mixed use of operation mode and component styles. No explicit modulation rule is provided. • Usually do not have a detailed description of the calculation formula. 	<ul style="list-style-type: none"> • Identified a standard list of representing elements for BCK. • A standard format is developed for BCK representing elements. • Ontology analysis is provided at the high level of modularization. • Control flow diagram is provided. • A number of control modules are created based on control variables ontology analysis. Each control module is functionally independent. The modulation rule is well-defined. • A function/algorithm element is created to represent the calculation formula.

First, a clearer representation of the control knowledge (as shown in Appendix B) is developed, comparing to existing documentation approach. The newly added representing elements for formula and algorithm provided a platform to describe the calculation in details. In the existing BCK documentation, detailed calculation formulas and algorithms are mostly excluded. This disconnects control algorithm designers and building operators/commissioners, which creates difficulties in understanding the logics and diagnosing faults. Therefore, this new element for formula and algorithm reconnects the algorithm designers with end users. This can improve the field engineer's interpretation the operation of the building HVAC system, which is critical for FDD process. If the control documentation is using the existing format without control

calculation formula, there could be some ambiguous control sequence description. This will affect the interpretation and understanding of the HVAC system operation.

Second, the modularization method simplifies the big and complicated BCK representation into a number of functional independent yet integrated control modules. The functional independent control module can be potentially helpful for control analysis and FDD. Because each module is constructed based on specific control variable, it is then relatively easy to build and verify its control sequences, logics, programming code, for the specific control functionality. For the FDD process, it becomes easier to analyze the operation of each control module with a specific function, rather than to analyze the whole building control system. For example, if a zone temperature control has some issues, it can just focus on the control module of “zone temperature” for the diagnosis.

The benefit of using a modular approach can also potentially support more distributed control algorithm development and enable control self-configuration. For example, the modular approach can support the development of control self-configuration process, if it depends on a database that stores generalized BCK modules. Chen and Treado [10] envisions this type of new self-configuration control system will be different from conventional self-configuration control systems (such as the work by Ryan and Shah [19], Wruck [23], Baldwin, et al. [6]). While conventional self-configuration control systems have been based on preinstalled fixed control algorithms to determine control outputs, the new one would interact with a BCK module database (where stores BCK modules) to select/update the applicable BCK control modules for a specific system. In this way, the self-configuration process is not be limited to fixed control algorithms, but can cover a wider application range and be easier to manage and update. This topic will be further explored in the future study of integrated database for BCK, and self-configuration process development.

Meanwhile, this BCK data schema developed in this work can potentially contribute to the IFC BAMie.

This data schema may be adopted under the Domain schemas -> Building Control Domain ->

IfcBuildingControlsDomain schema, serving as a new entity that document the control sequences, algorithms and logics for determining control outputs. In this way, it could expand the application of IFC Building Automation Modeling for a wider application range at the operational and maintenance stage (e.g., use this information for retrofitting control system, FDD, and on-going commissioning). This data schema may also be integrated into the BACnet Control Device modeling section as a set of new object types to enhance the representation of control knowledge that is used to determine control outputs. This can enrich the information exchange among control devices and benefit applications and services that are based on the information (such as FDD, optimization, and performance analysis).

6. Conclusion

In summary, this study presents an M-BCK data schema to represent generic BCK. First, a modularization process is developed based on individual control variables (both hardware and virtual ones). Second, a list of representing elements is developed to represent modularized BCK. This list provides the most comprehensive BCK representing elements. Comparing to existing approaches, this representation can potentially reduce the format inconsistency. Third, a demonstration case is provided, showing an example of how to implement our approach to represent BCK for a typical control module. This demonstration can be used as a reference for field application. The new BCK representation developed in this work would benefit the processes of building control system design, implementation, operation maintenance, and self-configuration.

To quantify and analyze the benefits of the proposed method, further studies are needed. First, it will be important to analyze the ambiguity level of proposed formats for BCK representing elements. Second, usability test needs to be conducted by implementing the BCK data schema in a database and tested by actual building engineers. This will help identify pros and cons of this data schema and move further towards field applications.

Acknowledgement

This material is based upon work supported by the Energy Efficient Buildings Hub primarily sponsored by the U.S. Department of Energy along with matching funding from The Pennsylvania State University. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

References

- [1] ASHRAE, ANSI/ASHRAE Standard 134-2005 Graphic Symbols for Heating, Ventilating, Air-Conditioning, and Refrigerating Systems, American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc., Atlanta, GA 30329, 2005, p. 79.
- [2] ASHRAE, ASHRAE Guideline 13-2007: Specifying Direct Digital Control Systems, American Society of heating Refrigerating and Air-Conditioning Engineers, 2007.
- [3] ASHRAE, Sequences of Operation for Common HVAC Systems, Multimedia CD American Society of Heating Refrigerating and Air-Conditioning Engineers Atlanta, GA, 2005.
- [4] AutomatedLogic, CtrlSpecBuilder, 2012. <https://www.ctrlspecbuilder.com> [Accessed: Dec. 31, 2012].
- [5] AutomatedLogic, EquipmentBuilder for Educator 5.5, Vol. 5.5, AutomatedLogic, <http://www.automatedlogic.com/eikon-for-educators/%20/>, 2014.
- [6] J.M. Baldwin, R.A. Bishop, W.G. Hansen, P.L. Polley, Self configuring controller, in: U.S. Patent Office (Ed.), Vol. 5,276,630, American Standard Inc, United States, 1994.
- [7] M.R. Brambley, P. Haves, S.C. McDonald, P. Torcellini, D. Hansen, D.R. Holmberg, K.W. Roth, Advanced sensors and controls for building applications: Market assessment and potential R & D pathways, Technical Report Pacific Northwest National Laboratory, Richland, WA, 2005.
- [8] buildingSMART International, Building Automation Modeling, Model View Definition Industry Foundation Classes (IFC 4), 2013.
http://docs.buildingsmartalliance.org/MVD_BAMIE/link/building-service-element.htm [Accessed: Oct. 6, 2014].
- [9] S. Calabrese, Sequences and Control Schemes for Air and Water Systems, 2010.
<http://www.slideshare.net/illinoisashrae/sequences-control-schemes-for-air-water-syst> [Accessed: Mar. 23, 2014].
- [10] Y. Chen, S. Treado, Self-Configuration of Building Control System Using Knowledgebase, American Society of Heating, Refrigerating and Air Conditioning (ASHRAE) Annual Conference, Seattle, WA 2014.
- [11] COOLTOOLS™, Design Guidelines: Chilled Water Plant Design Guide, 2009.
- [12] B. Dong, Z. O'Neill, Z.W. Li, A BIM-enabled information infrastructure for building energy Fault Detection and Diagnostics, Automation in Construction 44 (2014) 197-211. doi: <http://dx.doi.org/10.1016/j.autcon.2014.04.007>.
- [13] E.W. East, Construction Operations Building Information Exchange (COBie), Washington, DC, 2014. <http://www.wbdg.org/resources/cobie.php> [Accessed: Feb. 13, 2015].
- [14] International Organization for Standardization, ISO 16739:2013 Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries 2013.
- [15] J.M. Lee, J. Butler, M.E. Cantabene, H. Fairman, Standards for Fault Detection, Diagnostics, and Optimization in Building Systems, Cimetrics Inc., 2007.
- [16] X. Liu, B. Akinci, M. Bergés, J.H. Garrett Jr, Extending the information delivery manual approach to identify information requirements for performance analysis of HVAC systems, Advanced Engineering Informatics 27 (4) (2013) 496-505. doi: <http://dx.doi.org/10.1016/j.aei.2013.05.003>.

- [17] J. Lucas, T. Bulbul, W. Thabet, An object-oriented model to support healthcare facility information management, *Automation in Construction* 31 (2013) 281-291. doi: <http://dx.doi.org/10.1016/j.autcon.2012.12.014>.
- [18] A.J. Martin, C.P. Banyard, *Library of system control strategies*, Building Services Research and Information Association (BSRIA), Bracknell, UK, 1998.
- [19] J.D. Ryan, R.K. Shah, Self-configuring controls for heating, ventilating and air conditioning systems, in: U.S. Patent Office (Ed.), Vol. US 7,243,004 B2, Carrier Corporation, United States, 2007.
- [20] U.S. DOE, Building Energy Data Exchange Specification (BEDES), BEDES Dictionary 1.0, 2014. <http://energy.gov/eere/buildings/building-energy-data-exchange-specification-bedes> [Accessed: Dec. 27, 2014].
- [21] U.S. DOE, Commercial Sector Characteristics, Building Energy Data Book, 2003. <http://buildingsdatabook.eren.doe.gov/TableView.aspx?table=3.2.5> [Accessed: Oct. 4, 2014].
- [22] E. Utterson, Temperature Control Systems – Why Don't They Work?, *Solutions: The Role of the Consulting Engineer*, 2006. <http://www.slideshare.net/ashrae6/the-consultants-role> [Accessed: Mar. 23, 2014].
- [23] R.A. Wruck, Indoor air quality and economizer control methods and controllers, in: U.S. Patent Office (Ed.), Vol. US 7,434,413 B2, Honeywell International Inc., United States, 2008.
- [24] J. Zobel, *Writing for Computer Science*, 2nd ed., Springer London, UK, 2004.

APPENDIX A: BCK ELEMENTS FORMATTING

Table A.1 Elements formatting for typical control modules.

Elements	Attributes	Data Type	Format (explanation)		
			Local module	System module	Building module
Control module name	Symbol	text	L-#(component)_#(variable)_###(control algorithm)	S-##(system)-_#(variable)_###(control algorithm)	B-_#(variable)_###(control algorithm)
	Narratives	text	Local control module-(component name) (variable) using (control algorithm)	System control module-(system name) (variable) using (control algorithm)	Building control module-(variable) using (control algorithm)
Control objective	Narratives	text	To modulate <i>the controlled variable</i> with the purpose of <i>doing something</i>		
	Controlled system/ actuator ID	text	###-#####		
Operation mode	Symbol	text	English words		
	Narratives	text	Provide (functionality)		
System schematic	Symbol	text	Module ID - sch		
	Diagram	file	Attached diagram		
Control flow diagram	Symbol	text	Module ID - cfd		
	Diagram	file	Attached diagram (see Figure A.1 for the format)		
Control sequences	Symbol	text	### (using three letters to represent operation mode)		
	Mode	text	Use the mode symbol in the Operation mode section		
	Narratives	text	see Table A.2 for the format		
Data point (Inputs /Outputs /Parameter)	Symbol	text	###(variable characteristic 1)-### (variable location)-## (type)		
	Narratives	text	(A detailed description of the variable, including the variable location, measured media)		
	Unit	text	(SI or PI units or Unitless)		
	Source	text	Sensor/Virtual/Manual		
Alarms	Symbol	text	AL - ##(characteristic1) - ##(characteristic2) - ###(location)		
	Narratives	text	Describe the alarm		
	Logic	text	If Xi is in Yi, then triggered the alarm		
Functions / algorithms	Symbol	text	Fcn - ###(algorithm)		
	Narratives	text	Name of the algorithm and functional description of this algorithm		
	F-inputs	text	Algorithm inputs		
	F-outputs	text	Algorithm outputs		
	F-Parameters	text	Algorithm parameters		
	Formula	text	Mathematical description of the algorithm		
	Reference	text	Author-Year (of the related reference for this algorithm)		
Programming code	Symbol	text	Module ID - C - Version #		
	Attachment	file	Attachment (see Table 5 for different types comparison)		

Table A.2 Control sequence organized by component output value vs. by operation mode.

	By components output values	By operation mode
Rules	y_1 SHALL be $Y_1 \dots y_m$ SHALL be Y_m a) WHEN x_1 is X_1 b) OR WHEN x_2 is X_2 AND ,... OR x_n is X_n . where x_1 stands for the operation mode variable $x_2 - x_n$ stand for the other conditions variables that need to satisfy when x_1 is enable. X_1, X_2, X_n , are the linguistic variables of x_1, x_2, x_n , respectively $y_1 - y_m$ stand for control outputs $Y_1 - Y_m$ stand for control outputs values under this rule	IF x_1 is X_1 AND (x_2 is $X_2 \dots$ OR x_n is X_n), THEN y_1 is $Y_1 \dots y_m$ is Y_m . a) IF THEN
Usage	<ul style="list-style-type: none"> • Provide specified programming guide • Useful for trouble shooting and maintenance for specific components • Can provide sequences for multiple variations of a piece of equipment 	<ul style="list-style-type: none"> • Generally easier to understand • Better explain operating concept • Highlight differences between the operation modes

Table A.3 Comparison of programming types.

	Text-based	Graphic-based	Menu-based
Characteristic	Use high-level programming language (BASIC, C, Pascal)	Use graphic “functional blocks” for programming	Select control sequences pre-programmed by manufacturer
Flexibility	High	Medium	Low
Comprehensive	High	Medium	Low
Learning curve (time)	Long (a few months)	Medium (a few days)	Short (a few hours)
Programming time	Medium	Long	Short

Figure A.1 Examples of graphic symbols that should be used in a generalized control flow diagram following the CSI standard.

APPENDIX B: A ZONE VAV BOX DAMPER CONTROL MODULE EXAMPLE

- Control module name

<i>Symbol</i>	<i>Narratives</i>
L-Z1-VAV_D	Local control module - VAV damper position for Zone #1

- Control objective

<i>Narratives</i>
To modulate the VAV damper position with the purpose of realizing different operation scenarios.
<i>Controlled system/actuator ID</i>
Z1-VAV

- Operation mode

<i>Symbol</i>	<i>Narratives</i>
Manual override	Provide manual operation
Ventilation only	Provide ventilation (only) required by code
Cooling	Provide appropriate airflow rate during cooling mode
Heating	Provide appropriate airflow rate during heating mode
CO2 override	Provide CO2 override control to keep indoor CO ₂ level satisfy the requirement

- System schematic

<i>Symbol</i>	<i>Diagram</i>
L-Z1-VAV_D-sch	<p>The diagram illustrates the control system for a Zone VAV Damper. Supply Air (SA) enters the damper, which is actuated by a motor (M). The air then passes through a heating coil (HW) and a temperature sensor (T) labeled 'AI - Discharge Air Temp'. The air then splits into two paths: one goes to the discharge air (DA) duct, and the other goes through a reheat coil (AO - Reheating Valve) controlled by a motor (M). The reheat coil is connected to a heating water supply (HWS) and a heating water return (HWR) line.</p>

- Control flow diagram

- Data point

Symbol	Narratives	Unit	Source
C-Z1O-S	CO ₂ override signal	unitless	virtual
Coo-Z1-V	Zone #1 cooling demand percentage	%	virtual
Hea-Z1-V	Zone #1 heating demand percentage	%	virtual
Z-Z1D-S	Zone #1 VAV damper position	unitless	virtual
Z-Z1D-O	Zone #1 VAV damper position manual override	unitless	manual

- Alarm

<i>Symbol</i>	<i>Narratives</i>
AL-H-T-ZA	High temperature alarm Triggered: if the <u>zone temperature</u> is higher than high threshold value for more than 3 minutes
AL-L-T-ZA	Low temperature alarm Triggered: if the <u>zone temperature</u> is lower than low threshold value for more than 3 minutes
AL-H-h-ZA	High humidity alarm Triggered: if the <u>zone humidity</u> is higher than high threshold value for more than 3 minutes
AL-L-h-ZA	Low humidity alarm Triggered: if the <u>zone humidity</u> is higher low threshold value for more than 3 minutes

- **Control sequence**

<i>Symbol</i>	<i>Mode</i>	<i>Narratives</i>
L-Z1-VAV_D_seq-1	<i>Manual</i>	IF the <i>manual override</i> is activated, THEN the damper is under the Manual operation mode: i.e., the damper position is determined by the manual override signal.
L-Z1-VAV_D_seq-2	<i>Ventilation</i>	IF the <i>manual override</i> is NOT activated AND the <i>CO₂ override</i> is NOT activated, AND cooling demand is smaller than 5%, AND heating demand is smaller than 5%, THEN the damper is under the Ventilation only operation mode: i.e., the damper is controlled to maintain the minimum airflow set point.
L-Z1-VAV_D_seq-3	<i>Heating</i>	IF the <i>manual override</i> is NOT activated AND the <i>CO₂ override</i> is NOT activated, AND cooling demand is smaller than 5%, AND heating demand is larger than 10%, THEN the damper is under the Heating operation mode: i.e., the damper is controlled to maintain the minimum airflow set point (see reheat coil valve for heating temperature control).
L-Z1-VAV_D_seq-4	<i>Cooling</i>	IF the <i>manual override</i> is NOT activated AND the <i>CO₂ override</i> is NOT activated, AND <i>cooling demand</i> is larger than 10%, AND <i>heating demand</i> is smaller than 5%, THEN the damper is under the Cooling operation mode: i.e., the damper is controlled to maintain the lower zone temperature set point by adjusting the damper to modify the cooling airflow rate.
L-Z1-VAV_D_seq-5	<i>CO₂ override</i>	IF the <i>manual override</i> is NOT activated AND the <i>CO₂ override</i> is activated, THEN the damper is under the <i>CO₂ override</i> operation mode: i.e., the damper is gradually increased to lower the indoor CO ₂ concentration to the set point level (default 1000 ppm).

- **Functions/algorithms**

Symbol	Narratives	Formula	Inputs	Outputs	Parameters	Reference
Fcn-Z-VAV-Coo	Damper opening calculation for cooling	$D = \frac{(cooling * (V_{max} - V_{min}) + V_{min})}{V_{max}}$	Cooling demand (<i>cooling</i>); Minimum airflow rate (V_{min})	Damper position (D)	Maximum airflow rate design (V_{max})	Engineering practices
Fcn-Z-VAV-Ven	Damper opening calculation for ventilation and heating	$D = \frac{V_{min}}{V_{max}}$	Minimum airflow rate (V_{min})	Damper position (D)	Maximum airflow rate design (V_{max})	Engineering practices

- Programming code (programmed in MATLAB Stateflow as an example)