Topic 8b: Innovative Building Energy Rating Delivery Models - BuildingSync, gbXML, and Building EQ

Principal Investigator:
Carmel Software Corporation
1050 Northgate Dr., Suite 360
San Rafael, CA 94903 USA
PI: Stephen Roth, sroth@carmesoft.com
Phone: 1.415.225.6030

Research Institution:
Rensselaer Polytechnic Institute
86 34th Street, Suite D-601
Brooklyn, NY 11232-1000
Faculty: Dr. Dennis Shelden, sheldd@rpi.edu
Phone: 347-529-4058

PROBLEM, OPPORTUNITY, AND TECHNICAL APPROACH

Problem

Simulating the energy usage of buildings using sophisticated software has become a key strategy in designing high performance buildings that can better meet the needs of society. Automated exchange of data between the architect's software design tools and the energy consultant's simulation software tools is an important part of the current and future building design process.

Steady progress over the past two (2) decades has led to pervasive impact of computers on the building design industry. Building Information Modeling (BIM) and advances in building energy modeling (BEM) software have resulted in their adoption into the mainstream design process. BIM authoring tools are being adopted by more architects and engineers as these tools improve and become faster and easier to use. The whole premise behind BIM is that it is essentially a "database" where all the building information, including the geometry, is stored.

However, there is a fundamental disconnect between many of the BIM, BEM, building analysis, building asset, and building auditing software tools. Because these tools are developed by 10s, if not 100s, of different software vendors throughout the world, many of these tools do not "talk" with one another despite the fact they many of them require the same information about a building: i.e. – building square footages, wall areas, window areas, occupant densities, plug loads, occupancy schedules, and much more. There are many software tools in the building design and analysis industry that allow engineers, architects, and energy modelers to perform the following types of analysis, including whole building energy use, heating and cooling load analysis, lighting analysis, CFD analysis, solar/shading analysis, life-cycle cost analysis, energy benchmarking, and more.

The fact that many of these tools do not talk with one another discourages wide use of these software tools by energy modelers and other related practitioners. This is where "interoperability" comes into play. Interoperability allows for the sharing of data between different software tools developed by many different vendors. Interoperability is essential for BIM to realize its potential as a transforming technology as opposed to 3D CAD programs that are limited in their use as holistic building design tools. However, interoperability doesn't just apply to BIM to BEM software. It can apply to other types of building design software including building asset information, building audit information, and energy benchmarking software tools. This is where BuildingSync XML (or just BuildingSync) comes into play. BuildingSync is a schema developed by the National Renewable Energy Lab (NREL) that allows for the exchange of building energy audit information such as energy efficiency measures, utility data, and building rating information. This information can be used by other types of software tools including energy benchmarking software (such as ASHRAE Building EQ (http://buildingeq.ashrae.org), energy analysis software, and custom software developed by cities and municipalities to satisfy energy auditing rules and mandates.

While interoperability schemas have been around for twenty (20) years and are integrated into all major BIM and building performance software tools, end-users still struggle with less-than-ideal workflows or workflows that simply do not work. For example, geometric information from one BIM authoring tool is not properly represented in a popular HVAC load calculation software tool developed by a third-party vendor. While users can always manually edit and tweak data in an XML file (fortunately, it is clear text) so that it successfully imports into a consuming software tool, the ideal interoperable workflow should not include any type of human intervention. In fact, the ideal workflow would be the press of a button on the screen that would send information from one software tool to another, and it would all work perfectly. While this may be a utopian vision, there's no reason why the current state-of-the-art cannot be dramatically improved.

This is where benchmark test cases and validation software tools come into play to help improve this entire workflow. For example, geometry benchmark tests (or test cases) help to ensure that, as building geometry produced by building designers becomes more complex, the geometry produced for energy and heating and cooling loads analysis maintains the integrity of information that is required for a proper analysis. These tests are mostly prescriptive and serve as marks of excellence that identify the ability of a technology to translate geometry properly from its native format to XML schemas. For example, NREL's open source energy modeling tool, Open Studio (OS), relies on high quality interoperability with third party software through XML schemas such as Green Building XML (https://www.gbXML.org). In fact, NREL has previously funded development of a test case and validation procedure for tools which author gbXML: A series of 9 known test models were developed using the Open Studio software tool and then exported to gbXML. These exported models were then compared, using a web-based validator, to known truth standards. If the software passed all the validation tests, then it could be certified as producing correct XML schemas.

Opportunity

BTO is requesting that research and development be conducted for innovative delivery models for increasing access to building asset data from tools such as Home Energy Score and Asset Score. For Asset Score's Audit Template tool, one of the ways to increase access to this data by third-party software tools is using an interoperability schema such as NREL's BuildingSync.

BTO is asking that bidders suggest new workflows for either BuildingSync or HPXML. Our proposal will focus on BuildingSync XML since we wish to target the commercial building space. We are suggesting developing a comprehensive web-based portal that will help facilitate the adoption of BuildingSync and other similar interoperability schemas by third-party building analysis, auditing, and benchmarking software tools. In our 20 years of experience developing and managing the popular Green Building XML (gbXML) schema, we have come to realize that schemas do not work for end users unless there is some type of "transport" mechanism and validator that makes sure it will successfully import into a consuming tool without causing headaches and delays for the energy modeler.

Phase I of this proposal will focus on the workflow that involves the U.S. Department of Energy's Asset Score Audit Template to BuildingSync to the ASHRAE Building EQ benchmarking portal. ASHRAE Building EQ (http://buildingeq.ashrae.org) is a web-based software portal that provides a quick energy analysis that benchmarks a building's energy performance. Building EQ assists in the preparation of an ASHRAE Level 1 commercial energy audit (as defined by ASHRAE Standard 211) to identify means to improve a building's energy performance including low-cost, no-cost energy efficiency measures and an indoor environmental quality survey with recorded measurements to provide additional information to assess a building's performance. Carmel Software proposes developing a web-based tool (from this point on, it will be referred to as the "**Schema Server**") that will completely streamline the flow of information from DOE's Asset Score Audit Template into ASHRAE Building EQ, so that all the user has to do is press one button on the producing or consuming tool, and the software will perform quick data checks and test case validation and then seamlessly transfer to the consuming tool (in this case, ASHRAE Building EQ). Once we have proven in Phase I that this is doable for this specific workflow, we will expand to other workflows that involve additional XML and JSON schemas and producing/consuming software tools. However, this does not mean we are solely going to limit Phase I to focusing on the BuildingSync schema. Since Asset Score exports an Open Studio (OS) file, there is the potential for some geometric information to be transferred. In this the case, we will include some gbXML interoperability since OS exports to gbXML and BuildingSync references gbXML geometric elements in Version 2 of the BuildingSync schema.

The Schema Server described above will solve at least four (4) existing problems:

1. The most common complaint by users of the Building EQ portal is the amount of information that needs to be entered to be able to submit for approval to ASHRAE and then receive a final rating score (from 0 to 200, where 0 is net-zero energy). There are over 500 data entry values in the Building EQ portal. The Schema Server will be able to import a BuildingSync file produced by an Asset Score Audit Template project, cleanse the data, validate it, then send it over to the Building EQ portal, thus eliminating the entry of almost 60% the manual data input. This will save a tremendous amount of time for the end user and increase adoption of both software tools.

2. Another issue occurs when updates are made to the original Asset Score project and the updated data must then propagate to other tools that had originally consumed the first iteration of the data. This is often difficult and confusing for end users since end users must keep track of multiple versions of BuildingSync files and make sure the latest data is being consumed. The Schema Server will solve this problem by performing basic “version control” so users can track the differences between multiple versions of files and make sure the latest version is being consumed by a tool like Building EQ. An added benefit would allow users to observe the exact differences between all versions of a project so that they can track the work they have done and audit changes later on.
3. Often, the data that is produced by one software tool is not compatible with one or more consuming software tools. Data validation was discussed above, and a data validator could be used to alert the user that there is data incompatibility with one or more consuming software tools. The validator could then cleanse the data so it could be successfully consumed by the analysis software tool. The Schema Server will include a built-in data validator and data cleanser that ensures the data produces by Audit Template will be compatible with Building EQ and other software tools in future phases.
4. Developing XML or JSON schema integration functionality is often a difficult and tedious process for third-party software developers. The Schema Server will include an application programming interface (API) and a basic schema mapping function that will make it much easier for third-party software developers to import and export BuildingSync, thus increasing adoption.

The overall opportunity presented here is aimed at improving the workflow between disparate software tools whether we are talking BIM, energy benchmarking, building analysis, or building asset software tools. Currently, the workflows are so disjointed that it often prevents the adoption of these schemas and the wide-spread usage of various building design and benchmarking software tools. Ironically, these schemas were developed to facilitate seamless usage of various tools from different vendors in the software-for-built-environment space. Analysis software vendors do not want to develop integration with interoperability schemas without knowing that other tools are going to properly produce these files; BIM authoring tool vendors do not want to develop schema producing functionality without knowing that there is a large ecosystem of consuming software tools. Based upon our experience with working with and developing interoperability schemas (i.e. – gbXML, which integrates with over 50 software tools worldwide), it is imperative that for software developers to integrate a schema, there be an accompanying schema validator that helps facilitate this integration.

Potential Energy Savings

It is difficult to accurately predict total building energy savings across the U.S. when using building energy performance software. However, there are a couple of metrics that we can use to approximate it. If we assume most building performance software tools are attempting to satisfy the ASHRAE Standard 90.1-2016 energy savings requirements, then according to the BTOs Building Energy Codes Program Commercial Determination (https://www.energycodes.gov/development/determinations), there should be 8.2% energy cost savings by following 90.1. In 2016, U.S. commercial buildings used 13.70 quads (4.0^12 kWh) of electricity. An 8.2% energy cost savings represents a 3.3^11 kWh savings per year. If we assume the national average electricity cost is $0.1037/kWh, then this represents a $34 billion/yr in electricity cost savings. Building energy performance software plays a large role in helping architects and energy modelers incorporate ASHRAE 90.1 energy savings into their designs. Therefore, we can assume that software helps with calculating the majority of the 8.2% of energy savings mentioned above. If energy modelers were able to more easily use multiple software tools to calculate building energy cost savings using the software described in this proposal, this 8.2% figure could possibly increase. Even a modest 0.02% increase represents $83 million/yr in cost savings. This translates into 8.0^8 kWh/yr energy savings.

Technical Approach

The research portion of Phase I will be conducted by our partner research organization from the **Center for Architecture Science and Ecology (CASE)** at **Rensselaer Polytechnic Institute (RPI)** and will include the following technical approach:

1. Develop multiple schema test cases that will ensure a smooth workflow between DOE's Asset Score Audit Template tool and the ASHRAE Building EQ portal
2. Convert these schema test cases into "stub" software code to be used in the software development portion of this project
3. Architect basic schema mapping functionality that allow software developers to map XML schema elements to the tables and fields in their consuming software tool databases.

The co-PI, Dr. Dennis Shelden, and a doctoral student from CASE will develop the test cases based upon interviews with stakeholders. They will then convert these test cases to software "stub" code and pass them onto the Carmel team to develop the Schema Server software tool.

The Schema Server software development portion of this project will be conducted primarily by the PI, Stephen Roth, and his team at **Carmel Software** and will include functionality described in the Technical Objectives section below. Initial software prototypes will be developed using open source software development environments and technologies. End user feedback will be incorporated as an agile software development process will be used to develop the production portion of the software. The RPI team will perform additional testing and quality assurance throughout the software development process. They will also help with documentation and training toward the end of Phase I.

BENEFITS TO END-USERS

Designing energy efficient buildings is of utmost importance today due to a wide variety of factors including limited fossil fuel resources, pollution, global climate change, federal and state laws, high energy costs, and a host of other reasons. Buildings use 40% of all energy and a whopping 75% of electricity. If society is going to rely less on fossil fuels, we need to design more energy efficient buildings for both new and existing construction. The first step in designing more energy efficient buildings occurs during the initial design phase which involves running building energy simulation and analysis software that will predict yearly building energy usage. Improving the interoperability workflows discussed above will benefit the following stakeholders:

1. *Energy modelers*: Give them more incentive to use various software tools to design energy efficient buildings since it will be an easier and more seamless process to enter the same data in more than one BIM authoring and building analysis or benchmarking software tool.
2. *Software developers*: Gives them more incentive to integrate interoperability functionality into their tools if there is an easier and less expensive way to do it.
3. *Building owners*: By designing more energy efficient buildings, it will save building owners a significant amount of money in utility and energy costs over the lifetime of the building.
4. *Society as a whole*: Whatever people's political beliefs, there is no arguing that our fossil fuel supplies are finite, we are polluting the earth, and adverse climate changes are occurring all over the world including in our own backyard of California with unprecedented wildfires. Designing energy efficient buildings is just one step toward reducing our reliance of fossil fuels and cleaning the air for future generations.

The software developed as part of this proposal is a web-based portal that could prove valuable to the stakeholders listed above. In fact, it's valuable enough that this could be a viable commercial product with profit potential. We envision implementing the following business model:

1. Charging software developers an “API” fee to integrate with the API discussed in this proposal. This would be an ongoing monthly fee that allows them access to many features
2. Charging energy modelers a premium account fee for features above and beyond basic features of the portal. These premium features would include special reports, unlimited project creation capabilities, premium viewer functionality, and much more.
3. Charging building owners, ESCOs, and others a fee to access reports on all the building data that will eventually be captured by this service. This data could be quite valuable for determining building energy trends, types of energy efficiency measures being implemented, and much more.
4. Eventually, enough building geometry information will be located in the Schema Server database that it could be integrated with Google Maps and Google Street View allowing users to not only view the building from the outside, but also view the interior of the building along with energy efficiency-related information. This could drive additional licensing revenue.

TECHNICAL OBJECTIVES

The technical objectives for Phase I are as follows:

Our partner, RPI, will create validation test cases that involve the workflow between DOE’s Asset Score Audit Template software and the ASHRAE Building EQ portal:

Technical Objective 1: RPI will identify 10+ candidate test cases by interviewing energy modeling practitioners and other related professionals to identify issues with the current building asset data to consuming software tool workflow. After the 10+ test cases have been developed, RPI will stack-rank them (i.e. – prioritize them) from most to least important.

Technical Objective 2: Based upon the stack-rank exercise in Technical Objective 1, RPI, along with Carmel, will select the seven (7) or so most important test cases, then create written documentation for the seven test cases along with sample BuildingSync files that accompany them. The deliverables will include stub-code that can be used by the Schema Server to validate BuildingSync data.

Carmel Software will next develop the software code for the Schema Server:

Technical Objective 3: This objective will involve developing the Schema Server web portal that includes the following functionality:

1. A web-based front end that allow users to access the functionality discussed below.
2. Validation logic for the 7+ test cases selected in Technical Objective 2
3. User authentication and account creation capabilities: The user will be able to register, login, reset passwords, and create user accounts.
4. Project creation: Once the user creates an account, they will be able to create one or more projects that store individual BuildingSync files produced by Asset Score Audit Template. The user will also be able to upload multiple versions of a BuildingSync file produced by the same Audit Template project so different versions of the same BuildingSync file exists (aka – version control).
5. A web-based building geometric viewer: Implement a web-based viewer for any geometry data that is represented within the schema. While this may not be applicable for all Asset Score projects, there are some use cases where this could apply. For example, if an Open Studio file accompanies a BuildingSync file produced by an Asset Score project, and the Open Studio file includes building geometry, then it can export to a gbXML file with the correct 3D building geometry. Since BuildingSync 2.0 files reference gbXML geometric elements, the building

geometry could be displayed in the web-based viewer. We will build upon an existing open-source schema viewer that already exists.

6. A reporting section that displays many types of metrics about the BuildingSync XML file including how many times it is uploaded or downloaded to a consuming tool, which type of software tools are consuming it, and date and time stamps.
7. Implement data analysis functionality that could be performed on all the inputted building data. This functionality will probably be reserved for future phases due to limited budget in Phase I. Also, data analysis will not be useful until a fair number of projects have been created.

Technical Objective 4: Develop a basic schema mapping function for third-party software developers that allow them to map schema elements from BuildingSync data elements to tables and fields in their respective software databases. After the mapping is complete, it will produce skeleton or stub code in a variety of programming languages that allow developers to include XML consuming functionality in their software. The PI has learned through programming enough schema integration software that it is a very time consuming and tedious process to write code the reads an XML or JSON file and then populates the data into the correct tables and fields in the consuming software database. Phase I will include work that develops a very basic prototype of a user interface that allows software designers to more easily map this information, and if it proves successful, Phase II and beyond will develop a more complete schema mapping function along with multiple software language code templates and XML/JSON schemas.

Technical Objective 5: Develop an application programming interface (API) that allows third-party software developers to integrate with the Schema Server. It will allow producing (i.e. – Audit Template) and consuming (i.e. – ASHRAE Building EQ) software tools to communicate with the web portal so that users do not have to manually export and import XML files to and from their file system. The API will also perform basic data and test case validation. One example of how this will work is as follows:

1. The ASHRAE Building EQ web portal calls the Schema Server API looking for some BuildingSync XML project data just created by an Asset Score Audit Template project.
2. The API will run the basic test cases on the XML file to make sure that it will work with Building EQ data structure. If possible, the portal will “cleanse” any data that it can so it will be acceptable to the Building EQ data structure.
3. If not, a report will be produced showing the user what needs to be corrected or is missing from the Audit Template project.
4. If the BuildingSync XML data passes the tests, then it will automatically transfer to the Building EQ project and populate the appropriate data fields, thereby saving the user from manually inputting the data.
5. If the project is approved by ASHRAE personnel, the user will then be able to export the data to customized ASHRAE Standard 211 audit and narrative reports (this functionality already exists in Building EQ).

The difference between Technical Objective 4 and 5 may not be obvious at first glance, but they are quite different. Technical Objective 4 develops a schema mapping tool that will be a “one-time” function for a third-party software developer to create stub or skeleton code to be integrated into their software tool. Technical Objective 5 develops the API which is used as an ongoing feature to transfer data to and from Asset Score (producing tool) to Building EQ (consuming tool).

Technical Objective 6: Create user documentation and training materials so that all end users of the tool will be able to easily adopt the tool. Development of the user documentation and training will be an ongoing process throughout all phases of this project.

FINAL DELIVERABLES

After the above technical objectives have been completed, the following Phase I software deliverables will include:

1. Schema Server: The Schema Server web portal that seamlessly transfers data from Asset Score Audit Template via BuildingSync XML to the ASHRAE Building EQ.
2. Schema Mapping: Software developers will be able to map BuildingSync data to their own software data models.
3. Application Programming Interface (API): Software developers will be able to take advantage of an API that will allow them to more easily create BuildingSync producing and consuming functionality in third-party software.

WORK PLAN

The Phase I work plan will be conducted by Carmel Software and its team at its offices in San Rafael, CA and by Dr. Dennis Shelden and his team at Rensselaer Polytechnic Institute in Brooklyn, NY. The work plan is designed to accomplish the six (6) technical objectives described above.

The work plan will be broken down into four (4) sub-phases:

1. Developing the required test cases that allow for Asset Score to seamlessly transfer building information via BuildingSync into ASHRAE Building EQ. It will also include requirements gathering for developing the Schema Server web portal. This will be performed by Dennis Shelden and his team at RPI.
2. Developing the Schema Server web portal that allows users to register, login, create accounts, create projects, and view building geometry. This will involve both front-end (user interface) and back-end (database) development. It will be built on Microsoft Azure (https://azure.microsoft.com) which is both a scalable and secure cloud platform. It will be developed using open-source software development tools. This work will be performed by the PI and the team at Carmel and testing will be performed by the team at RPI.
3. Develop the open API (application programming interface) and schema mapping functionality that allows third-party software developers to integrate with the web portal and more easily map BuildingSync schema elements to data model elements. This work will be performed by the PI and the team at Carmel and testing will be performed by the team at RPI.
4. Develop the user documentation and training materials that will encourage adoption of the software tool. This work will be performed by both teams.

14 Stage Software Design Process

Completing the above 3 sub-phases will require the following stages of software design:

Pre-design Stage:

1. Interviews and requirements gathering
2. Preparation of functional specifications
3. Project coordination

Design Stage:

4. Detail Specifications
5. Build a prototype
6. Meetings

Development:

7. Coding
8. Testing
9. More coding and testing until complete

Pre-deployment:

10. User documentation
11. Installation

Deployment:

12. Training
13. User and Technical Support
14. Software Maintenance

Task Specifics

This section describes the above tasks in more detail.

Task 1: (Pre-design stage) Interviews and requirements collection - This stage involves collecting requirements from end-users including energy modelers, energy program managers, building owners, architects, and engineers. RPI will interview energy modeling practitioners and other related professionals to identify issues and problems with the current building asset data to energy benchmarking tool (or related) workflow and group these issues in three categories:

1. BuildingSync schema validation
2. BuildingSync data validation
3. BuildingSync data appropriateness for the consuming software tool

RPI will first develop a format of the interview. The interview will be arranged in a face-to-face or video conference format (Zoom or GotoMeeting), and it will consist of a mix of a list of questions and user demonstrations. RPI estimates to conduct about 15 to 20 interviews in this task and each interview will take approximately 30 minutes. RPI will leverage existing customer groups and partners as well as reach out to local modeling organization groups to conduct the interviews. Once the interview process is complete, RPI will identify 10 or more candidate test cases that specifically represent the typical issues discovered in the interview process. The 10 test cases will be summarized following the same documentation format as in ASHRAE 1468-RP – "Development of a Reference Building Information Model (BIM) for Thermal Model Compliance Testing". This was an ASHRAE research project that focused on developing test cases for common thermal features in today's buildings that are assumed to have the greatest impact on a building's energy use.

Lastly, RPI will review and estimate the development time for each defined test case, then optimize the number of test cases that can be developed within the proposed budget. RPI will initially develop 10+ test cases and then stack-rank the list from most appropriate to least and end up utilizing around 7 test cases that will make it into the web portal.

*Interim Deliverables***:** Interview summary report and a list of the 10+ candidate test cases
will be provided along with descriptions. The deliverables will include Microsoft Word documentation and stub-code that can be used in the Schema Server to validate BuildingSync data. A minimum of 7 (maybe more) test cases will be developed that ensure the flow from Asset Score to Building EQ is seamless as possible. It will take about 5 weeks to finish this task.

Task 2: Feasibility analysis and preparation of functional specifications - Following the requirements gathering phase, RPI will do the following

1. Develop a stack-ranked list of the requested functionality that will be implemented into the software. This list and its rankings will be based upon the client/partner discussions in Task 1.
2. Determine the feasibility of developing the requested features

The deliverables will include:

1. Software architecture documents
2. A stack-ranked list of required functionalities for the entire project.

Task 3: Project coordination - The project schedule and costs will be prepared after having some idea of the requirements. This includes the project coordination, administration and management help to control the implementation and deployment of the expected application. The deliverable will be a detailed work plan. Both RPI and Carmel will work on this task together.

Task 4: Detail specifications - All the information is collected from the pre-design stage (tasks 1 – 3) and the software architecture process begins. This involves creating the software workflows and specifying the software "objects" that will accomplish those workflows. Using techniques such as UML (uniform modeling language) and other related software design procedures, these specifications will be converted from ideas on paper into working code. The deliverable from this task will be detailed UML diagrams and stub-classes that will form the basis for the software code.

Task 5 and 6: Build a user interface and solicit feedback from potential end users –A basic user interface of the Schema Server will be developed that will be void of any core functionality, but will be a "mock-up" that can be presented to end-users to solicit feedback. The deliverable resulting from this task will be user interface code that will become the basis for the working software prototype. Carmel Software will work on these tasks, and the PI will coordinate the efforts of these 2 tasks. Carmel software developers will work on coding the user interface and backend database. Once the user interface prototype is developed, RPI will conduct some simple and basic usability tests with potential end users to solicit feedback on the look-and-feel. Any valid feedback will be incorporated into the user interface in future tasks discussed below.

Task 7, 9: Coding - This task involves the development of the core software prototype. It will build upon the deliverables from tasks 1 to 6. The deliverable from this task will be a functional prototype that end users will be able to use to provide feedback on the functionality, itself. In addition, this prototype will be presented to the FOA SBIR contact, if appropriate at this stage. Carmel will perform these tasks using an agile style of software coding.

The PI will coordinate the efforts of these 2 tasks. Carmel software developers will work on coding the user interface and backend database. Carmel will utilize the following development environments and other software management tools:

1. GitHub and SourceTree for software version control and backup
2. JIRA for issue and bug tracking
3. Brackets and PyCharm for software coding in Python
4. Brackets for software coding in Javascript and NodeJS
5. MySQL Workbench 8.0 for database creation and management

Carmel has the qualifications to work on this specific workflow described here (Asset Score to BuildingSync to Building EQ) for several reasons:

1. Carmel developed the ASHRAE Building EQ web portal, so we are very familiar with how it works, how to integrate with it, and what the data model looks like. We also developed a very basic BuildingSync integration into Building EQ, but it lacks any type of data validation and is seldomly used.
2. Carmel is very familiar with the BuildingSync schema. We have worked closely on multiple fronts with the NREL team and BuildingSync:
    a. Since Carmel manage the gbXML schema, Carmel worked with them to help reference the gbXML geometric elements from BuildingSync 2.0.
    b. Carmel worked on the BuildingSync integration into ASHRAE Building EQ so they are familiar with the schema, itself, and the type of data that is populated into the BuildingSync file by Asset Score Audit Template.
    c. The PI is a member of ASHRAE TC 7.6 which is the "Building Data Exchange" technical committee which gathers stakeholders in BuildingSync to help further promote it within ASHRAE Society and elsewhere.

Task 8, 9: Testing – After creating an initial prototype of the web portal, RPI will conduct quality assurance testing both in-house and with stakeholders that were interviewed in the first 2 tasks. Based upon the in-house unit testing and feedback, Carmel will update the code (return to Task 7 above) until the code is robust and usable. Carmel will use JIRA and other unit testing tools for conducting quality assurance.

Tasks 10/11: User Documentation and Deployment – Carmel will develop some limited user documentation for Phase I of this FOA. The bulk of the documentation will be developed in future phases. The Schema Server web application will be deployed to production on a virtual Linux server hosted on Microsoft Azure web application.

Task 12/13/14: Training, User Support, Maintenance – Carmel will perform some limited training and maintenance during Phase I of this FOA. The bulk of the training and maintenance will occur in future phases. User support will be ongoing throughout Phases I, II and III, if applicable.

LINKING THE WORK PLAN TO THE TECHNICAL OBJECTIVES

Technical Objectives 1 and 2 involve identifying and stack-ranking the 10 candidate Asset Score Audit Template to BuildingSync to Building EQ test cases to be incorporated into the Schema Server portal. Tasks 1 and 2 will accomplish these objectives that will be conducted by the team at RPI.

Technical Objectives 3, 4, and 5 involves the multi-step software development process of creating the Schema Server portal, the schema mapping functionality, and the API. Task 3 will cover writing the software development plan, and Task 4 will cover architecting the software modules. Tasks 5 and 6 cover coding the initial user interface and soliciting feedback from end users. The bulk of the work will be performed in Tasks 7, 8, and 9 since these tasks involve software coding, testing, receiving user feedback, re-coding, then re-testing until a production-level and robust software tool has been created. An agile style of software design will be used to accomplish these objectives. The bulk of this work will be performed by Carmel and its software development team. The testing and user feedback will be performed by RPI.

Technical Objective 6 involves developing user documentation and training and support. Tasks 10 to 14 cover user documentation, training, and support tasks which accomplish this objective.

SETTING THE FOUNDATION FOR FUTURE PHASES

The software development in this first phase (Phase I) will set the foundation for future phases that will build upon a comprehensive Schema Server, schema mapping tool, and application programming interface (API). We envision this

Schema Server eventually becoming a central repository for all building data interoperability schemas allowing 10s, if not 100s, of building analysis and BIM software tools to query it and seamlessly import and export information.

The prototype of the schema mapping functionality developed in this phase will become the basis for a more comprehensive schema mapping function that allows third-party software developers to map multiple types of XML and JSON schema dictionaries to their own software data models. Some examples of other data interoperability schemas that could be accommodated by the Schema Server are gbXML, the Industry Foundation Classes (IFCs), and coBIE. After the mapping is complete, it will automatically produce software code in a variety of software languages that could be used in software code to read the XML or JSON file and populate the corresponding fields in the application database.

In future phases, the API will also be able to accommodate a wide variety of XML and JSON schemas.

Once enough building information is stored in the Schema Server database, it will be able to perform complex analysis on this data allowing interested parties to observe trends in building energy efficiency design.

LABOR CATEGORIES

The following table displays a list of all the labor categories and the number of labor hours utilized for tasks 1 to 14. The tasks are not necessarily sequential. The expected duration of this portion of the project is around 10 months.

Labor Category	**Tasks 1/2**	**Task 3**	**Task 4**	**Task 5**	**Task 6**	**Tasks 7/8/9**	**Tasks 10/11**	**Tasks 12/13/14**
Principal Investigator (Carmel)	20	15	28	28	12	78	18	8
Co-PI (RPI)	40	18	0	0	9	52	9	6
Doctorate Student	65	15	0	0	12	78	21	14
Software Developer 1	0	0	0	108	21	286	15	10
Software Developer 2	0	0	0	108	15	260	0	0
Total Hours:	**125**	**48**	**28**	**244**	**69**	**754**	**63**	**38**

PROJECT SCHEDULE

The proposed project duration is 10 calendar months from the start date:

Tasks	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**0**	**1**	**2**
Tasks 1 and 2:												
Task 3:												
Task 4:												
Task 5:												
Task 6:												
Tasks 7, 8, 9:												

Tasks 10, 11:												
Tasks 12, 13, 14:												

EXISTING RESEARCH AND DEVELOPMENT

The PI knows of parallel research being conducted by other organizations that complements the R&D in this proposal. One existing research project is ASHRAE RP-1810, "Development of Reference Building Information Model (BIM) Test Cases for Improving Usage of Software Interoperability Schemas". The work statement for this research project was written by the PI and ASHRAE approved funding for it. The contractor for RP-1810 is BuildSimHub, a software development firm out of Pittsburgh, PA. The objectives of this research project are:

1. To develop 20+ gbXML test case documents and corresponding gbXML files that software vendors can follow to develop their own gbXML files that will then be uploaded to a web-based validator that will provide feedback on the validity of the files;
2. Further develop an existing web-based validator software tool that will validate the 20+ test cases;

The resulting test cases from RP-1810 could be used as test cases for this or future phases.

The BuildingSync website (https://www.buildingsync.net) does have its own schema validator for different use cases. However, it requires the user to manually upload a BuildingSync file to validate it. There is no API, no mapping functionality, and no viewer functionality similar to what is proposed in this project: https://selectiontool.buildingsync.net/validator

There currently does not exist a comprehensive building schema server such as the one proposed here that allows many tools to communicate with one another. There are publicly available APIs from organizations such as DOE, PNNL, and several private companies. However, these APIs fail to achieve the goal of truly promoting the expansion of building schemas and democratizing them so that non-domain experts can more easily integrate their software tools with other tools in the building design software space.

QUALIFICATIONS OF PERSONNEL

This project requires a comprehensive set of Building Information Modeling (BIM) skills including experience in working with information exchange standards based on XML, creation and management of data content based on those schemas, and extensive knowledge of BEM and building asset software. We believe this project team is uniquely qualified to perform the work described in this proposal.

Complete resumes for the following personnel are available in the FOA Workspace attachments. The following abridged resumes include experience that applies to this proposal.

Stephen Roth, Principal Investigator

Stephen (**https://www.linkedin.com/in/stephenbroth/**) will serve as the Principal Investigator for this project. He is owner of Carmel Software, a 25-year old software development firm currently focused on developing mobile and web-based applications that allow architects, engineers, and contractors to design more energy efficient buildings. Currently, Carmel licenses over 200,000 mobile apps to customers worldwide.

Prior to this role, Stephen was the Senior Product Manager of Revit at Autodesk. He oversaw an entire development, quality assurance, and publication team for Autodesk Revit (30+ people) and helped gather requirements and manage the development and release of each subsequent version of the software tool. Stephen sold his previous engineering software company's intellectual property (IP) to Autodesk prior to beginning work there and helped incorporate that IP into the Revit product.

Stephen is a graduate in mechanical engineering and business from the University of Pennsylvania, a registered Professional Engineer with the State of California, and a registered LEED AP with United States Green Building Council.

In addition, Stephen is the President of the Board of Directors for Green Building XML (gbXML) for over 10 years. He has obtained funding, updated the schema, performed other research projects with Autodesk, NREL, US DOE, and Georgia Tech, in additional to working with 50+ software vendors to integrate the gbXML schema.

Stephen is a member of ASHRAE and currently holds, or has held, the following technical committee positions that cover topics related to this FOA:

- Current Chair of ASHRAE SPC-224: The Application of Building Information Modeling
- Prior Chair and Current Voting Member, TC 1.5: Computer Applications
- Current Voting Member, MTG-BIM: Multidisciplinary Task Group on Building Information Modeling Standardizing and Utilizing ASHRAE Online BIM Data Exchange Protocols
- Current Corresponding Member, TC 7.6: Data Interoperability Schemas
- Current Corresponding Member, Building EQ Committee

Dennis R. Shelden, Associate Professor, Rensselaer Polytechnic Institute, NYC

Rensselaer Polytechnic Institute's CASE will be the research organization for this project. The following individual from CASE will be the co-PI:

Dennis Shelden (https://www.linkedin.com/in/dennis-shelden-06b69a6/) is a licensed architect, academic, author, and entrepreneur, whose experience spans architecture, engineering, and computer science applications to professional practice and the built environment.

His three degrees from MIT include a Bachelor of Science in Architectural Design, a Master of Science in Civil and Environmental Engineering, and a Doctor of Philosophy in Design and Computation conducted under the advisement of the late William J. Mitchell. He joined Gehry Partners in 1997 while still in MIT's PhD program, first as Director of Research and Development, then as Director of Computing, and was responsible for the management and strategic direction of the firm's computing program including research, IT infrastructure, and partnerships. His PhD Thesis *Digital Surface Representation and the Constructability of Gehry's Architecture* remains a widely referenced treatise on the motivations and technical underpinnings of the firm's revolutionary applications of digital modeling to the design and delivery of complex buildings. In 2002 he co-founded the spinoff company Gehry Technologies, serving as Chief Technology Officer on the development of several software products and Project Executive on numerous groundbreaking building projects until the firm's acquisition by Trimble in 2014.

He is an Associate Professor of Architecture and Director of Rensselaer Polytechnic Institute's Center for Architecture, Science and Ecology (CASE) in New York City. His work focuses on professional applications of generative and building information modelling to the design, construction and operation of integrated physical / digital building systems, including built environment Digital Twin applications. Prior to leading CASE he was Associate Professor of Architecture, Director of the Digital Building Laboratory, and Director of the School of Architecture's Doctoral program at Georgia Tech, and an Associate Professor of Practice in MIT's Design and Computation Program.

His publications have appeared in journals including *AD: Architectural Design*, *TAD: Technology and Architectural Design*, and the *Journal of Construction Automation*, and he has been interviewed in publications including the *Economist*, *New*

York Times, *Design Intelligence*, and *Engineering News Record*. He is editor for the forthcoming book series *Practical Revolutions: Professional Applications of Disruptive Technologies to Building*, published by Wiley.

Previous Positions:

- Associate Professor of Architecture, Director of the Digital Building Laboratory, and Director of the School of Architecture's Doctoral program at Georgia Tech
- Chief Technology Officer, Director of Global Services, Gehry Technologies, 2002-2016
- Associate Professor of Practice, Visiting Lecturer, MIT Department of Architecture Design & Computation Program, 2005-2016

About CASE

The Center for Architecture Science and Ecology (CASE) is a Rensselaer Polytechnic Institute wide research center addressing the need for radically improved, energy-effective, and sustainable built environment solutions. CASE unites advanced architectural and engineering practices with scientific research through collaborations among multiple academic institutions, manufacturers, and professional offices within the building industry.

REFERENCES

1. ASHRAE, 2012. "Automating Building Thermal Model Creation: Interoperability from BIM to Energy Simulation Software", ASHRAE Research Project 1468.
2. ASHRAE, 2020. ASHRAE Building EQ, https://buildingeq.ashrae.org
3. ASHRAE Standard 211-2018. "Standard for Commercial Building Energy Audits".
4. ASHRAE. 2016. ANSI/ASHRAE/IES Standard 90.1-2016. Energy Standard for Buildings Except Low-Rise Residential Buildings. ASHRAE.
5. Department of Energy Building Energy Codes Program, 2017: https://www.energycodes.gov/development/determinations
6. Department of Energy and Office of EERE's "Energy Savings Analysis: ANSI/ASHRAE/IES Standard 90.1-2016" Report, October 2017. http://Energy.gov/eere/buildings
7. gbXML.org, 2020. The Green Building XML Schema, http://www.gbxml.org/
8. National Institute of Building Sciences, 2018. Construction Operations Building information exchange (COBie), https://www.nibs.org/page/bsa_cobie
9. National Renewable Energy Laboratory (NREL), 2018. BuildingSync, https://buildingsync.net/