A Comparative Case Study for Using COBie in Project Turnover

Zhulin Wang, zlwang@vt.edu
Virginia Tech, United States

Tanyel Bulbul, tanyel@vt.edu
Virginia Tech, United States

Andrew McCoy, apmccoy@vt.edu
Virginia Tech, United States

Abstract

The planning, design, construction, and commissioning processes of buildings are rich with project information created in various formats and stored in different locations. Such information contains specific project details helpful for understanding, operating and maintaining a facility properly. Yet not all information created during the production stages are in a usable format or even accessible to the owner. Currently, new information needs to be generated and recorded by an owner to track a facility's status. Therefore, the need exists for a mechanism to capture the necessary information at the right time, filter out unnecessary information, and reorganize the information for an owner's use. This paper will focus on COBie as an existing effort to capture and deliver facility information to the owner, in terms of its usability for project turnover and facility management. We provide a comparative analysis between a COBie process and a BIM-supported project turnover process without COBie to evaluate benefits and challenges involved in both.

Keywords: COBie, Project Turnover, Facility Management, Comparison

1 Introduction and Background

The traditional way of storing and transmitting facility information is document-based, either in paper or electronic format. The architecture, engineering and construction (AEC) industry has welcomed benefits from building information modeling (BIM) in visual presentation and conflict detection for more accurate design and construction coordination. On the other hand, BIM's ability to become a repository for project information for owners during the operation of a facility has not been utilized yet. BIM's potential to enable a data-rich facility management has been studied before (Turkaslan-Bulbul et al 2006; Lucas et al 2013; Wang et al 2015) and the benefits to the owner are well documented (Akcamete et al 2010; Arayici et al 2012; Becerik Gerber et al 2011). However, there are barriers for the AEC industry in adopting a process for BIM-based project turnover. The challenges to adoption can be summarized as follows:

- Design and construction teams involve multiple participants. Depending their work scope, they use different terminology, naming conventions and specialized software that are not always interoperable with each other.
- If there is not a specific procedure defined by the design and construction team at the beginning of a project, the quality of the final handover deliverable to the owner is hard to control.
- Multiple sources of information in different formats cost the owner's personnel much time for locating the information they are looking for. This causes the facility management (FM) teams to create their own set of information.

- Owner's FM team does not usually have the software and the expertise to extract the facility information when handed with a BIM based as-built model, causing the populated data in it untapped.
- Even if the owner has the ability to use the model or export the data in the model to an FM system, the benefits from this still depend on the (1) quality of the model, which requires the organizational support by all project participants developing the facility and (2) the interoperability between the BIM systems and FM systems.

The building delivery process needs to have a mechanism to capture the necessary information at the right time along the lifecycle of a facility. Such a mechanism needs to: filter out unnecessary information; reorganize the information for owner's convenience; resolve interoperability problems among different software systems with the assistance of open standard formats and data-based communication; and finally extend the benefits of BIM to a facility's operation. Since the transition to a data-rich process involves reforming both technical and organizational structure, the AEC industry might resist adoption if benefits do not outweigh the effort or challenges involved. In this paper, we will study COBie (the Construction Operations Building Information Exchange, a data schema and a data capturing process) as an existing effort to capture facility information during the design and construction process and deliver it to the owner. In order to gain a better understanding of how using COBie impacts current BIM-supported turnover process and how COBie can be utilized in practice, we have conducted a case study based on the data from a real world turnover process. The COBie based turnover is then compared to the project's actual BIM-supported turnover process regarding the benefits and challenges of each with implications from the comparison summarized.

2 Case Study

The project in this study is real; a 10000-square-foot 3-storie institutional building with sustainability features. The facility contained virtual design and construction processes, which were utilized throughout the project, and serves as an appropriate lens through which to view COBie. Since owner is promised a BIM-based project turnover, the contractor provided a mobile BIM tool, BIManywhere (BIManywhere 2015) during the turnover for as-built building information visualization and inquiry. For this study, we used the same resources (available design models, construction models, submittals, O&M documents, etc.) to populate COBie data for owner's FM. All the steps for both processes are recorded and the differences and similarities are analyzed and evaluated in order to inform for future projects for digital building turnovers.

2.1 The non-COBie-based turnover process

The handover package was completed by the contractor with the subcontractors' involvement in the construction model development, the submittal approval and the O&M documents' collection. The designer's model and drawings were referenced in the process. The following tools were used during this process: Autodesk Revit, Autodesk Navisworks, Autodesk AutoCAD, Microsoft Excel, BIManywhere Navisworks extension, BIManywhere AutoCAD extension, web browsers, Microsoft OneDrive (optional) and Bluebeam (Optional). Figure 1 shows a process map of the non-COBie-based turnover. The process that is conducted by the contractor and, supported and observed by our team for this research is described in the following sections.

2.1.1 Creating the Record Specification Library

1. Using Autodesk Revit, equipment, room and area schedules are exported as spreadsheets from design team's 100% completed models (both architecture and MEP).
2. These extracted schedules are used to create tables for the equipment list. New columns are added to the table to store data for the **tags** (e.g., VCB-1, VCB-2), **types** (e.g. Volume Control Box (VCB), Variable Refrigerant Fan Coil (VRF)), **sub-types** under a certain type (e.g. VRF wall-mounted unit, VRF vertical air handler, and VRF 4-way discharging ceiling-mounted cassette), **location**, and **related specification number** of each piece of equipment that maps them to design drawings.
3. The final construction model is generated by integrating the designer's and subcontractors' models in Navisworks. This model is used to verify the equipment location and catching inconsistencies due to post-design phase project changes.

2.1.2 Selection Set and Saved Viewpoint setup in Navisworks for exporting the model to BIManywhere

1. During the verification of the equipment list and locations in the previous step, the corresponding objects in the construction model are identified and used to create a Navisworks Selection Set. A selection set is created for each piece of equipment on the equipment list. Each selection set is named under a convention that recognizes both its equipment tag and location, and then organized into folders with a 3-level hierarchy (*1. specification number and name, 2. equipment type and abbreviation, 3. equipment subtype under a certain type*) in the Selection Set window. An extra number is added as a suffix to the name of a selection set, to prevent the conflicts if more than one piece of equipment share the same tag on the schedules in the design drawings.
2. A viewpoint is created for each room or area and for each piece of equipment. The room viewpoints are created using a view angle that looks at the room or area from a door or an entrance to the space. If the room or area is large with multiple doors or entrances, multiple viewpoints are created. The room viewpoints are named by "*room number + room name*", and organized in folders with a 2-level hierarchy (*1. "Room Views", 2. Different levels*). The equipment viewpoints are created using a view angle that shows each piece of equipment in the center, but excludes most unrelated equipment. The equipment viewpoints are named by "*equipment tag @ equipment location*" and add the same suffix as the corresponding selection set to differentiate pieces with the same name. The equipment viewpoints are then organized into folders with a 2-level hierarchy (*1. "Equipment Views", 2. Equipment type and abbreviation*).

2.1.3 Document preparation and upload to cloud folder

1. Approved submittals and O&M documents have a distributed association with various equipment types, and a scattered approval process. The electrical submittal review system can be used to track submittal approval status. However, since required submittals under the same specification number are usually approved as a complete package, an optional solution is used to create separate submittal tracking lists to break them down into detailed requirements within a specification. Other documents involved in the process include commissioning reports, as-built drawings, specifications, etc.
2. Each approved submittal is broken down into multiple .pdf files to the level that they only have one type of information (such as a product data sheet, a shop drawing, a certification, etc.). Each file is named using the related "*equipment tag + file type*", such as "CU-1 - Product Data". The aim is to make sure that the contents inside each .pdf file can be quickly decided by its name. The process is repeated for O&M documents, sustainability documents, and commissioning documents.
3. After the break down of these documents, bookmarks to important chapters are added for documents with more than 20 pages for quick inquiry.
4. The documents are organized into 1-level folders using "*specification number + specification name*" as the naming conventions. Then all folders are uploaded to selected web-drives, such as OneDrive, Google Drive, Bluebeam Studio, Dropbox, etc.

2.1.4 Picture map creation

1. Pictures of each piece of equipment are taken in their as-installed condition before the drywall installation.
2. As-built drawings (in .pdf format) are opened in Bluebeam, and a symbol at the location of each piece of equipment is added with markup tools. Corresponding pictures are added to the symbols.
3. Files are saved and uploaded to the web drive.

2.1.5 Final model update

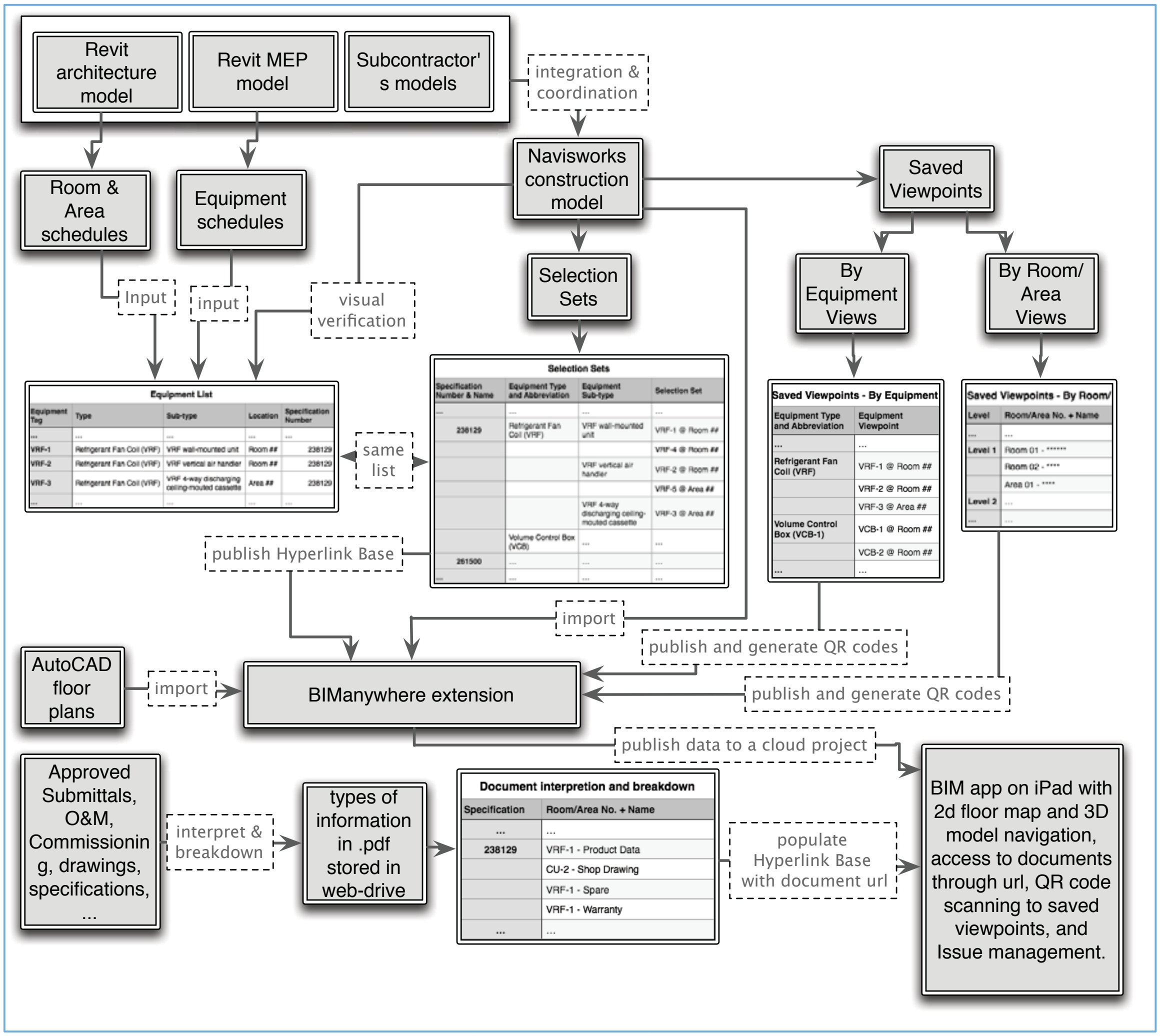
1. After the substantial completion of construction, the construction model is checked against actual installation, and inconsistencies are identified.
2. Inconsistencies are communicated with subcontractors and final model updates are collected.
3. Construction model is updated with the final changes together with selection sets and saved viewpoints in Navisworks.

2.1.6 Document integration

1. Construction model (.nwd) and floor plans (.dwg) are imported into BIManywhere with its Navisworks and AutoCAD extensions.
2. Saved Viewpoints are published to BIManywhere and QR codes are generated for each viewpoint through Navisworks extension. QR codes are printed for BIManywhere app to scan and quickly access a certain viewpoint from the physical building.
3. An empty "Hyperlink Base" for each Selection Set is created and published with BIManywhere Navisworks extension, and each hyperlink base is named the same as the corresponding selection set. The hyperlink bases for each Selection Set are managed as a list of "Assets" in a BIManywhere project and are accessible through a web browser on a computer or tablet device.
4. Properties of each asset/hyperlink base (equipment type, model no, manufacturer, installer, installation date) are edited with the information from available documents.
5. Hyperlinks are inserted to each "Asset" or "Hyperlink Base" and named according to the type of information inside the document name (such as Product Data). The URL of the target document stored in web-drive is added.

2.1.7 Access and information confirmation with iPad

1. The model is navigated and, viewpoint, QR codes and hyperlink bases are confirmed with BIManywhere app on iPad.
2. The project information is handed over to the owner.
3. A brief training is provided to the owner about using the iPad app and retrieving information.


Revit architecture model
Revit MEP model
Subcontractor's models
integration & coordination
Navisworks construction model
Saved Viewpoints
Room & Area schedules
Equipment schedules
Selection Sets
By Equipment Views
By Room/ Area Views
Input
input
visual verification
same list
publish Hyperlink Base
import
publish and generate QR codes
AutoCAD floor plans
import
BIManywhere extension
publish and generate QR codes
publish data to a cloud project
Approved Submittals, O&M, Commissioning, drawings, specifications, ...
interpret & breakdown
types of information in .pdf stored in web-drive
populate Hyperlink Base with document url
BIM app on iPad with 2d floor map and 3D model navigation, access to documents through url, QR code scanning to saved viewpoints, and Issue management.

Equipment List

Equipment Tag	Type	Sub-type	Location	Specification Number
...	...	...	...	...
VRF-1	Refrigerant Fan Coil (VRF)	VRF wall-mounted unit	Room ##	238129
VRF-2	Refrigerant Fan Coil (VRF)	VRF vertical air handler	Room ##	238129
VRF-3	Refrigerant Fan Coil (VRF)	VRF 4-way discharging ceiling-mouted cassette	Area ##	238129
...	...	...	...	...

Selection Sets

Specification Number & Name	Equipment Type and Abbreviation	Equipment Sub-type	Selection Set
...	...	...	...
238129	Refrigerant Fan Coil (VRF)	VRF wall-mounted unit	VRF-1 @ Room ##
			VRF-4 @ Room ##
		VRF vertical air handler	VRF-2 @ Room ##
			VRF-5 @ Area ##
		VRF 4-way discharging ceiling-mouted cassette	VRF-3 @ Area ##
	Volume Control Box (VCB)	...	...
261500	...	...	...
...	...	...	...

Saved Viewpoints - By Equipment

Equipment Type and Abbreviation	Equipment Viewpoint
...	...
Refrigerant Fan Coil (VRF)	VRF-1 @ Room ##
	VRF-2 @ Room ##
	VRF-3 @ Area ##
Volume Control Box (VCB-1)	VCB-1 @ Room ##
	VCB-2 @ Room ##
...	...

Saved Viewpoints - By Room/

Level	Room/Area No. + Name
...	...
Level 1	Room 01 - ******
	Room 02 - ****
	Area 01 - ****
Level 2	...
...	...

Document interpretion and breakdown

Specification	Room/Area No. + Name
...	...
238129	VRF-1 - Product Data
	CU-2 - Shop Drawing
	VRF-1 - Spare
	VRF-1 - Warranty
...	...

Figure 1 Non-COBie Process Map

2.2 COBie-based turnover process

The COBie worksheets were created with the same package of available resources from the contractor. The design models in Revit were used for generating COBie data and transforming it into spreadsheet format with Revit's COBie extension. The tools used were Autodesk Revit, Revit's COBie Extension, and Microsoft Excel. Figure 2 shows a process map of the COBie-based turnover. The process conducted by the research team is also described in the following sections.

2.2.1 COBie Extension - setup (using CADD Microsystems, Inc 2014a)

1. COBie extension is used for Revit to input and save a list of contacts (contributors to the COBie data) with required (e.g. Created By and Email) and optional (e.g. Name, Organization, and Address) input.
2. Default settings for Revit are selected to prepare COBie data, such as locality, classification system, unit, naming, room/space positioning of components, exported attributes, coordinates documentation, and schedule creation for COBie fields editing.
3. The COBie related schedules are created automatically.

2.2.2 COBie Extension - modification (using CADD Microsystems, Inc 2014a)

1. COBie extension is used to recognize all the spaces (rooms and areas) and each is assigned to defined zones.
2. Revit types and components are selected to be included in COBie export, and then all designated COBie fields are updated with mapped Revit data passed to them.

2.2.3 Revit - COBie fields quality control

1. The mapping relationship of Revit data and COBie data (CADD Microsystems, Inc 2014b) is used to ensure that COBie data sources are available and have proper values in Revit extension. COBie data is populated and revised as needed in property window and facility setting.
2. Generated COBie schedules in Revit are checked and updated for incorrect or missing fields.
3. Since Revit is mainly for design use, the naming of different elements may not be intuitive for the contractor and the owner. Therefore, it is important to conduct quality control of COBie fields carefully before exporting. For this case study, the first exporting trials without verifying COBie data quality in Revit have resulted in obscure and confusing spreadsheets with incomplete information.

2.2.4 COBie Extension - export and append/merge

1. After proper data quality control, the extension is used to export a spreadsheet from each design model (in this case an architecture model and a MEP model). Except for error-prone manual merging of files in spreadsheet format, COBie extension offers to append a new export automatically to an existing spreadsheet file. In either way, after merging the exports, files are checked for eliminating information duplicates or conflicts (East & Carrasquillo-Mangual 2012)
2. The exportable data from Revit are for worksheet (1) COBie.Contact, (2) COBie.Facility, (3) COBie.Floor, (4) COBie.Space, (5) COBie.Zone, (6) COBie.Type, (7) COBie.Component, (8) COBie.System, (9) COBie.Attribute, and (10) COBie.Coordinate. Information for worksheets reflecting the physical or logical relationship among different components (COBie.Assembly and COBie.Connection) can not be exported.

2.2.5 Microsoft Excel - Manual population of COBie.Spare, COBie.Resource, COBie.Job, COBie.Impact, COBie.Document, etc

1. Other worksheets in a complete COBie spreadsheet such as (1) COBie.Spare, (2) COBie.Resource, (3) COBie.Job, (4) COBie.Impact, and (5) COBie.Document are more related to the construction and commissioning data to capture spare part and resources information, required preventative maintenance activities, environmental influence, and approved documents of different assets. These are mostly out of designer's scope of work. If the COBie turnover data is to be completed by the contractor without commissioning software exporting COBie compatible data, these worksheets have to be populated manually in spreadsheet format.

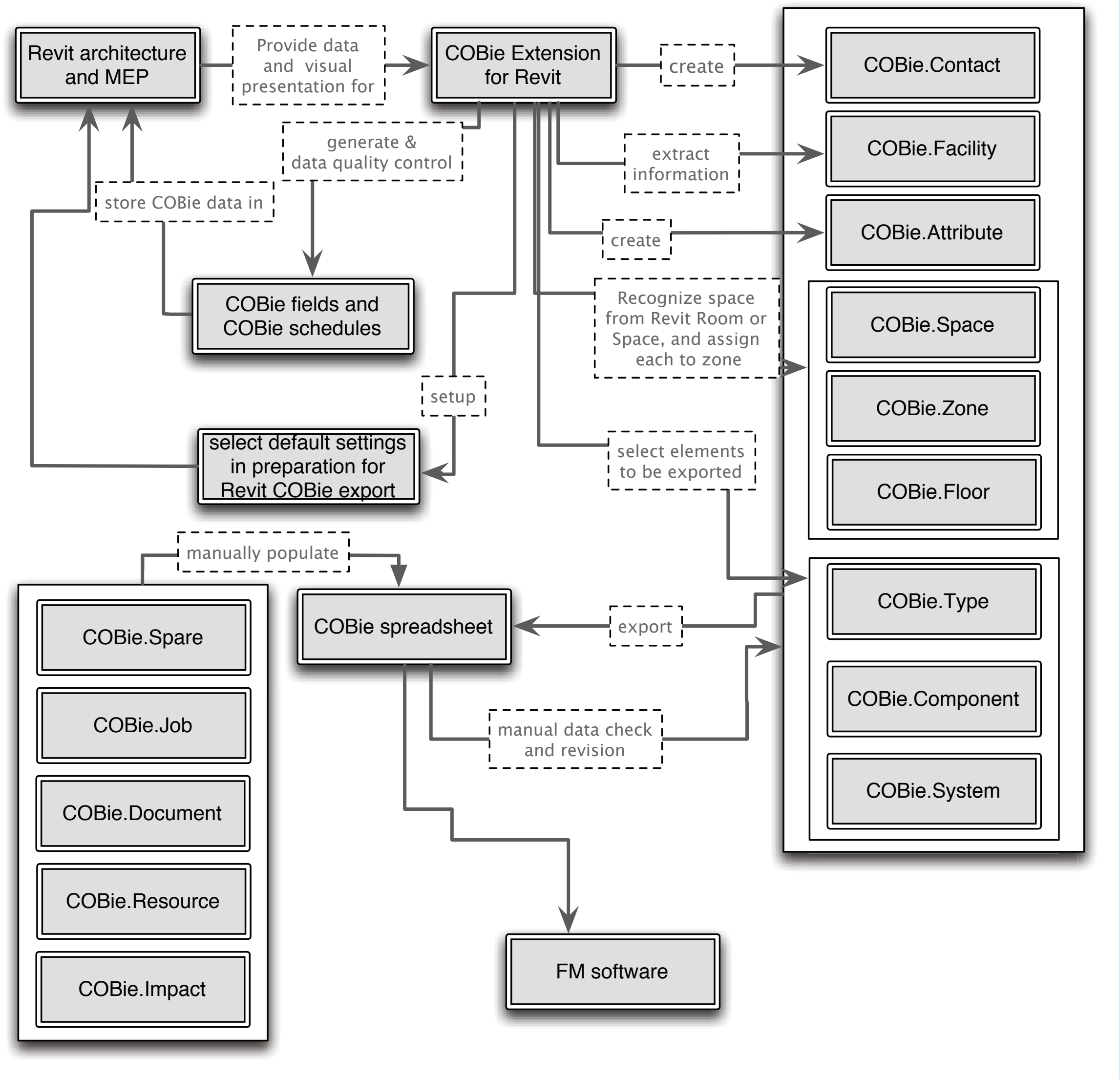

Revit architecture and MEP
Provide data and visual presentation for
COBie Extension for Revit
create
COBie.Contact
extract information
COBie.Facility
create
COBie.Attribute
generate & data quality control
store COBie data in
COBie fields and COBie schedules
Recognize space from Revit Room or Space, and assign each to zone
COBie.Space
COBie.Zone
COBie.Floor
setup
select default settings in preparation for Revit COBie export
select elements to be exported
manually populate
COBie spreadsheet
export
COBie.Type
COBie.Component
COBie.System
COBie.Spare
COBie.Job
COBie.Document
COBie.Resource
COBie.Impact
manual data check and revision
FM software

Figure 2 COBie Process Map

3 Comparison

The similarities and differences between the non-COBie and COBie-based turnover processes for FM use in this case study are analyzed according to the following aspects.

3.1 Model and data consistency with actual installation

The non-COBie-based process uses the construction model to keep track of actual installation changes, makes final updates to eliminate the differences, and captures as-built properties from approved documents and project records. The owner can get a final accurate model with information at hand for FM.

The COBie-based process uses the design model because Navisworks, which is used for developing the construction model, does not have COBie extension or support ifc export directly. Therefore, model changes after completing the design are not reflected from the COBie export and the export have to be manually checked and revised against as-built information. There are third party tools, such as iConstruct that lets the user to map Navisworks Selection and Search Set to the IFC schema (iConstruct 2015) inside Navisworks. However, the quality of the ifc export depends on how the user maps the data, and more importantly how much information is captured in Navisworks when importing models from Revit and other design software. Even when the existing properties in Navisworks are mapped properly, valid COBie data can not be transformed from the ifc file with COBie TOOLKIT (ERDC 2013) if the ifc file does not contain a complete set of facility information needed for COBie, leaving more workload for manual population.

As for model consistency, the COBie-based process requires extra labor and quality control to export information from the as-built model, or have to manually correct the COBie spreadsheet to reflect the changes. It is difficult to integrate COBie with accurate as-built model, because the automatic data (purple areas) in the spreadsheet which recognize related model components (by element ID) can only be mapped with the design (Revit) model (where it is exported from), instead of the as-built model. As for data consistency, both processes have to manually populate as-built properties and related documents.

3.2 Efforts needed

Both processes start with organizing the facility’s equipment list and location information based on types, subtypes, and other system properties. In non-COBie-based process, location is just a property of equipment, while in COBie-based processes, locations (space and zone) are managed assets with their own properties like equipment. The management of equipment and location information in COBie-based process is more convenient than that in non-COBie-based process. Based on the underlying data mapping structure from Revit to COBie, Revit extension’s GUI guides the user to select the equipment and location from a recognized list, while non-COBie process is more labor-intensive without automated steps.

The non-COBie process is a compromise between data-based and document-based systems. In concerns that a thorough data-based system is too cumbersome and the construction process can not guarantee the collection of accurate data, the non-COBie process breaks down traditional construction documents into smaller pieces of information in .pdf (each containing a specific informaiton). It only captures the minimum amount of general data for all types of equipment such as model number, manufacturer, but leaves the specific parameters in documents for manual data extraction. On the other hand, COBie displays all data in cells on multiple related worksheets and allows users to input and check the information. A spreadsheet can be a perfect tool to manage information in small amounts and simple relationships. If the BIManywhere tool in non-COBie process could accept a spreadsheet to enter all equipment data, the input efficiency would be improved. However, when it comes to complex COBie spreadsheets, the scale of involved data inquiry and validation overpasses the benefits of spreadsheet in managing data. Especially for larger and more complex projects, the huge amount of managed assets and properties will result in vey long worksheets, making it difficult to locate a specific piece of information. If COBie is generated from multiple models, it is also difficult to manually eliminate duplicates and conflicts without a good understanding of each COBie spreadsheet.

Although the Revit extension can create COBie fields, retrieve data from Revit, and allow the user to review before export, management of the COBie fields is still time-consuming. Some fields remain empty in the schedule, and some have obscure names. It also remains a question whether changes to COBie fields will influence the original value in Revit model, and whether the revision will remain when updating the COBie fields again through Revit extension. After the export, it takes time to figure out the equipment classification and attributes because everything is listed in rows and columns in spreadsheet format, making it difficult to understand the relationships among data in different worksheets. In order to verify the data, the user needs to check every value. Some of these values require a specific format, and others require a limited input from a pick list. If the data doesn’t meet value and format rules, its import and consumption will be impaired. Moreover, the value seen in COBie schedules can be different if COBie extension is programmed to change some of the value during exporting, thus requiring extra data verification after the export.

During the non-COBie based process, the construction team is aware of how the provided information will be used by the owner’s FM team. The .pdf documents will become buttons hyperlinked to every piece of information and attached to selected equipment sets, the viewpoints are for quick QR code scanning entry, and the input model number and manufacturer, etc will be displayed when opening an equipment set. On the other hand, COBie data is specifically intended for generic FM software. It could be helpful for the design and construction teams to know about the owner's intention of using a specific type of FM software and how the COBie data should be organized. This understanding offers people a clearer vision of the destination of data on each row, column and worksheet and helps them fit COBie data collection and quality control into feasible to-do lists. It would be easier for implementing COBie in the early design stage as suggested, with a well-planned implementation process and contractual arrangement together with the support from designers and subcontractors, rather than putting all the responsibility on owner or contractor during the limited time of turnover.

The extra work of data input, value / format conformance, and data validation may push some users away from adopting a COBie-based process, even when they are aware of the benefits for an owner. Since COBie is supposed to be an intermediary format between BIM and FM, it is acceptable to have a spreadsheet as an intermediary presentation for organizing information. However, if the spreadsheet is also supposed to be used by design and construction professionals, a better user interface for visualization and management of data is needed.

3.3 Learning Curve

Design and construction teams in the non-COBie-based process are mostly familiar with Revit and Navisworks. While using the BIManywhere extensions, all that is required of the user is to follow instructions on the interface and test the final result on a tablet device (such as an iPad). These processes can happen without much exposure to raw data.

For a COBie-based process, the Revit extension outcome being intermediary data, extra work should be done to understand the principles, steps and settings to correctly transfer Revit model files to COBie, and to map relationships of Revit data and COBie data (CADD Microsystems, Inc 2014b). First, the user has to understand the COBie data structure and relationships, and its presentation (worksheets, rows, columns, cells, picklists, naming conventions, formatting requirement, color coding, etc) in spreadsheet format. Second, COBie data quality control is delicate and requires a team's cooperation. In addition to the performance-based standard, a COBie guide providing a framework of basic steps, required properties and customization options should be learned before the user could focus on each decision point when populating COBie data. The learning process is relatively longer than a non-COBie process, though. Attending a training workshop or learning from others' experience instead of searching for resources from multiple places and interpreting them all alone would significantly reduce the learning time. Adoption of AEC innovation often fails with the need to train in new systems, so care should be taken to educate the industry properly (McCoy et al. 2012).

3.4 Extensibility

COBie can be extended by adding columns to the rightmost side of each worksheet as its customized fields. It is convenient to add parameters to an existing and mature data structure, instead of building up a complete new structure from scratch. The only concern is that the customized FM software may not be able to distinguish the customized parameters and map them to the fields in the software if no data-mapping tool supports it. However, with extra programming support, the customized value in COBie structure can expedite development of new functions. The COBie spreadsheet contains information of the element IDs and the BIM system that is used (East 2013). It is possible to link populated COBie information back to the model, which generated it at the beginning.

Unlike other software and COBie that take documents as local attachments, the hyperlinking function in the non-COBie based process can link multiple sources of media stored in the cloud, such as a web page, an online video, or documents from web-based drives. For many web drives, if the file name and directory is not changed, updating the contents or replacing it with another same-name document will not influence the validity of the hyperlinked address. Moreover, the hyperlinking function is advanced to hyperlink information to any selected component (even not under the same type) as a set of equipment instead of adding hyperlinks to every piece of equipment one by one. The only requirement is connection to the Internet or intranet, or if necessary, to sync the web-based drive locally.

Unlike the COBie process which exposes the user to the project data directly and requires measures to standardize the submittal process and quality, the non-COBie process is flexible to embed whatever is submitted and approved in the turnover package. The grouping of equipment, type and subtype is also at user's flexible decision while the COBie process is more based on Revit model's component description. However, to reduce the amount of information validation, reorganization, and recreation, it is still important that the non-COBie process makes provisions for the submittal process and collection of resources for the intended functions at the beginning of the project.

In a word, the COBie process is more extensible in regards to adding new functions with data customization, and mapping with BIM systems and IFC standard, while non-COBie process is more extensible and flexible with dealing the existing practices and functions.

3.5 Data maintenance and update

For both processes, the update of turnover data needs to fit in well with the project's changes and BIM updates.

It is easy to update the COBie file with Revit's COBie extension. The settings are saved along with the Revit model, so that the user does not need to re-do every step of setup and modification for an updated export, although the user does need to select and conduct the update function because the fields are not automatically synced from Revit to COBie. However, after it is exported from Revit, the data is maintained and updated in spreadsheet format, the user can easily lose track of the changes made to it. Therefore, the timing of COBie extension export should not be too early, in order not to lose the benefits of COBie extension's quality control and data mapping. The timing should not be too late either, because there is still a lot of need for data input and validation in the spreadsheet format. For the owner, the information can be edited after COBie data is imported into their FM system, but these changes may not be updated to the original COBie data. Therefore, COBie is primarily maintained in spreadsheet before project turnover, and after that it will get expired if the FM system does not maintain and produce it anymore.

For the non-COBie based process, after the model is published to the cloud, the hyperlinks and general equipment properties in each equipment set (hyperlink base) are accessible and editable by both contractor and owner through a web browser on PC and iPad. The system can also be used for FM problem recording and tracking during the operation and maintenance phase. However, when it comes to building renovation which involves major changes, the viewpoints, selection sets, floor plans and 3D model can be only modified through the Navisworks interface by the contractor as an extra service for the owner. Therefore, the non-COBie process can be maintained easily by the owner, as long as there are no changes in models, floor plans, viewpoints and selection sets.

4 Implications

The implications for a better project turnover process from the comparison in this case study are in areas of user interface development, management support, and the advantage of having standardized data.

First, both the non-COBie and COBie processes can be improved by providing better a user experience (e.g. the non-COBie process can support faster entry of equipment data with online forms to reduce clicks) and reducing the learning curve. Industry innovation has been shown to resist adoption without proper training and a reduction of the learning curve. For example, COBie-based process can reduce spreadsheet usage by having a better UI to visually guide the user on each step, and combining all needed training resource at a single location for easy reference. The user interface can be developed to be compatible with multiple inputs and capable of multiple outputs with consideration of different stages, disciplines and users. Instead of exposing user to every detail, complexities can be hidden behind the user interface with conditional programming.

Second, in both cases, early decision making and provisions for project turnover can lead to a more seamless flow of information. It is important to include the management support and contractual obligations into the BIM execution plan. Project turnover should be seen as a process, not just a task to be completed at the end of the construction.

Third, data standard development is more promising than random user interface development in the long run. Standardized data saves time during modeling and supports seamless information flow between different teams.

5 Conclusions, Limitations and Future Work

In this case-based example, we have implemented COBie on a real project as a pilot experiment to understand the current usage of COBie, the resources and learning curve required, the procedures and encountered difficulties, and the efforts spent. After implementation of COBie with a case project's resources, the COBie process is compared to a non-COBie BIM-supported turnover process that is actually used for turnover of the case project. The compared aspects include model and data consistency, efforts needed, learning curve, extensibility, and data maintenance and update. Implications have been discussed based on the comparison regarding possible improvements in future projects.

In this study, the 100% design model, construction model with final as-built updates, approved submittals, specifications, contractual documents, and drawing sheets have been used or references. However, as the commissioning documents are not fully available, the COBie data collection from commissioning documents may be conducted as needed after the documents are ready. This study mainly focuses on COBie data preparation rather than COBie data consumption. In the future, the resulting COBie data is to be imported to an example of FM software, in order to check how COBie data have been used from the FM side, including the functions supported, and COBie data mapping to FM software's data structure. Software that provides lifecycle management of COBie data is also considered in future investigation.

References

Akcamete, A., Akinci, B., & Garrett, J. H. (2010) Potential utilization of building information models for planning maintenance activities. *Proceedings of the International Conference on Computing in Civil and Building Engineering*. The University of Nottingham, Nottingham, UK, June 30th-July 2nd 2010. pp. 151.

Arayici, Y., Onyenobi, T. C., & Egbu, C. O. (2012) Building information modelling (BIM) for facilities management (FM): The MediaCity case study approach. *International Journal of 3D Information Modelling*. 1(1). pp. 55-73.

Becerik-Gerber, B., Jazizadeh, F., Li, N., & Calis, G. (2011) Application areas and data requirements for BIM-enabled facilities management. *Journal of construction engineering and management*. 138(3). pp. 431-442.

BIManywhere (2015) *What is BIManywhere?* Retrieved Jun 2, 215 from http://bimanywhere.com/

CADD Microsystems, Inc (2014a) *COBie Extension – Documentation*. Retrieved May 21, 2015 from http://www.caddmicrosystems.com/cobieextension/help/help.html

CADD Microsystems, Inc (2014b) *COBie Extension – Documentation*. Retrieved May 21, 2015 from http://www.caddmicrosystems.com/cobieextension/help/help-04-parameters.html

East, B. (2013) *Using COBie. BIM for Facility Managers, 1st Edition.* John Wiley & Sons. New Jersey, United States. pp. 107-143.

East, B., & Carrasquillo-Mangual, M. (2012) The COBie Guide: a commentary to the NBIMSUS COBie standard. *Building SMART Alliance*.

ERDC (2013) COBie TOOLKIT User Guide- Version: BETA 1.4.2. *U.S. Army Engineer Research and Development Center.*

iConstruct (2015) *iConstruct Features – Smart IFC Exporter V2.0*. Retrieved May 21, 2015 from http://www.iconstruct.com/about/features

Lucas, J., Bulbul, T., & Thabet, W. (2013) An object-oriented model to support healthcare facility information management. Automation in Construction, 31, 281-291.

McCoy, A.P., O'Brien, P.J., Novak, V. and Cavell, M. (2012). "Toward Understanding Roles for Education and Training in Improving Green Jobs Skills Development." *ASC International Journal of Construction Research and Education*. Volume: 8, Issue: 03, pages 186 - 203.

Turkaslan-Bulbul, M. T., & Akin, O. (2006) Computational support for building evaluation: Embedded Commissioning Model. *Automation in construction*,*15*(4), 438-447.

Wang, Z., Bulbul, T, & Lucas, J. (2015) A Case Study of BIM-Based Model Adaptation for Facility Management - Information Needs Analysis. *2015 ASCE International Workshop on Computing in Civil Engineering.* Austin, Texas, June 21th-23th, 2015.