

AUTOMATIC GENERATION OF COBIE DATA FROM REVIT

*J. Lee, W. Jeong, V. Faghihi, and J. Kang
Texas A&M University
3137 TAMU
Langford Building A, Room 424
College Station, Texas 77843-3137
United States
(*Corresponding author: jeffyi@tamu.edu)

AUTOMATIC GENERATION OF COBIE DATA FROM REVIT

ABSTRACT

As the usage of BIM increased in the construction industry, some construction owners started exploring the use of BIM for maintaining their facilities. Construction Operations Building Information Exchange (COBIE), a spreadsheet data format suggested in 2007 to manipulate building information, is considered as one of the solutions that would facilitate the process of transferring information from BIM to applications for facilities management. Several BIM and Computer Aided Facilities Management (CAFM) applications offer functions to export or import the COBIE data. To facilitate the process of generating the COBIE data out of Building Information Model, some add-in applications, such as COBIE Toolkit for Revit, also were developed. However, most of these applications handle a limited amount of COBIE data. Professionals using these add-in applications therefore sometimes end up spending a significant amount of time to add additional data manually to the COBIE data file. To automatically get a data-rich COBIE file, we developed a prototype application that extracts information from Revit, which is one of the BIM authoring tools, stores them in an external database, and then produces the COBIE file using information stored in the database. We applied our application to a pilot project and produced 8 COBIE worksheets automatically, which demonstrated a significant amount of time saving. This paper presents how our prototype application produces 8 COBIE data sets automatically. It also addresses our attempt to handle more COBIE data sets.

KEYWORDS

BIM, COBIE, automatic generation, Revit, Facilities Management

INTRODUCTION

As Building Information Modeling (BIM) has been actively used for productivity improvement in design and construction (D&C) (Sacks, 2004; Lee et al., 2008), many construction professionals and building owners begin to be serious about using information generated in the course of D&C for operating and maintaining the facilities. To better maintain facilities, professionals already have been using Computer Aided Facilities Management (CAFM) solutions, which is basically helping them make proactive decisions using information stored in the database system. Knowing that both BIM and CAFM handle building information, logical speculation would be whether we could use information stored in BIM for CAFM system, so that we do not have to punch the same information in the CAFM system again.

Construction Operations Building Information Exchange (COBIE), a spreadsheet data format suggested in 2007 (East, 2007) to manipulate building information, has been considered as one of the solutions that would facilitate the process of transferring information from BIM systems to CAFM systems. Several BIM and CAFM systems in fact offer functions to export/import information to/from the COBIE data. To facilitate the process of generating the COBIE data out of Building Information Model, some add-in applications, such as COBIE Toolkit for Revit, also have been developed.

Most add-in applications developed for BIM solutions, however, handle a limited amount of COBIE data. For instance, COBIE Toolkit for Revit produces only 5 COBIE data sets out of 12 commonly used and 16 entire COBIE data sets (Autodesk, 2013). Therefore, professionals using these add-in applications sometimes end up spending a significant amount of time in order to add additional data manually to the COBIE data file. Furthermore, it is a manual process. End users have 11 stages to finally produce COBIE files from the Revit BIM model.

We, therefore, investigate how to produce more a data-rich COBIE file from design tools automatically. We develop a prototype application and also address effort to handle more COBIE data sets from Revit BIM models.

OVERVIEW OF COBIE

COBIE has been considered as one of the solutions that would facilitate the process of transferring information from BIM systems to CAFM systems. Its purpose is improving the way to capture information in D&C phases, and providing for operation and management phase. (East, 2007; 2012) The merit of COBIE is that it can be viewed not only in several BIM applications but also in simple spreadsheets. “This versatility enables COBIE to be used regardless of size and technological sophistication.” (East, 2013)

COBIE file contains data from the entire life cycle of buildings or facilities. While there were 29 individual worksheets in every COBIE file the very first time (Table. 1), 16 worksheets are used in these days and 12 worksheets are commonly used in real projects (Table. 2). There are four, three, three, and six worksheets for design, construction, O&M, and common area, respectively. Each worksheet includes a variety of columns which have more specific data. For instance, contract worksheet is included in contract set. Contract worksheet includes specific data like ‘Email’, ‘CreatedBy’, ‘CreatedOn’, ‘Category’, ‘Company’, ‘Phone’, ‘ExternalSystem’, ‘Department’, and so on.

Table 1 – 29 Worksheets of COBIE

Set	29 Worksheets
Contract	Contract
Design	Facility, Floor, Space, System, Register, Component, Attribute, and Coordinate
Submittal	Schedule, Document, Transmittal, and Action
Installation	Installation, Manual, Warranty, and Spare
Commissioning	Instruction, Test, and Certification
Resource	Material, Tool, and Training
Job plan	PM, Safety, Trouble, Start-up, Shut-Down, and Emergency

Table 2 – 16 Worksheets of COBIE

Set	16 Worksheets
Design	Facility, Floor, Space, Zone
Construction	Type, Component, System
O&M	Job, Resource, Spare
Common	Contact, Document, Issue, Coordinate, Attribute, Connection

The specification of COBIE provides neutral and open standard format. It is delivered in the Spreadsheet XML 2003 format at each stage of building’s life-cycle. Some of worksheets are required fields, while others are optional. Likewise, some columns in a worksheet are fundamental, while others are optional. Commercial BIM applications showed two approaches to create COBIE files. Some of applications like Autodesk Revit and Bentley Architecture use IFC as a media to transfer data. The other applications like ARCHIBUS, Inc ARCHIBUS and AssetWORKS Aim produce directly from the application (East, 2013).

RELATED BACKGROUND RESEARCH

Many construction professionals have begun to study information handling for operating and maintaining phases of facilities (Lee, 2007; Yi et al., 2008; Sabol, 2008; Riddell, 2013). For the better maintenance and management of facilities, Motamedi and Hammad (2009) suggested the use of RFID (Radio frequency identification) with BIM concept. They showed a technical way to utilized BIM into the practical projects. Kasprzak and Dubler (2012), Lucas (2012), and Becerik-Gerber et al. (2012) studied BIM frameworks or BIM data handling to support facility management (FM). Lucas applied BIM

framework particularly to the FM of health care facility. While many researchers have shown the effort to struggle with BIM and FM, COBIE is out of research scopes. Some researchers tend to consider COBIE just a medium to transfer data from D&C phases to O&M phase. While COBIE is not a complete implement to deliver data at each stage, researchers have studied on how to apply COBIE itself to a real project (Anderson et al., 2012).

STATE-OF-THE-ART OF REVIT COBIE TOOLKIT

While Revit is one of the most popular BIM design tools in North America, it also supports API (Application Programming Interface) to expand its potential towards the other life cycle phases of buildings or facilities. Revit COBIE Toolkit is a good sample of it. This toolkit expands its ability into FM area by producing COBIE data file from the BIM design application.

However, according to Autodesk (2013), the add-in application does not generate the COBIE file fully and automatically, and it just assists end users to produce COBIE deliverables in conjunction with their Revit model. In terms of data, it just produce 5 data sheets out of 16 sheets; ‘04 – Space’, ‘05 – Zone’, ‘06 – Type’, ‘07 - Component’, and ‘08 – System’. The data in these sheets contains design information, thus it is quite possible to be produced from BIM design application directly. While 3 other sheets such as ‘01 – Contact’, ‘02 – Facility’, and ‘03 – Floor’ also contain design information, Revit COBIE Toolkit is not able to produce them. Furthermore, professionals using these add-in applications sometimes end up spending a significant amount of time in order to add additional data manually like ‘09 – Spare’, ‘10 – Resource’, ‘11- Job’, ‘12 – Document’, ‘13 – Attribute’, ‘14 – Coordinate’, ‘15 – Connection’, and ‘16 – Issue’ like Figure 1. Thus, it is required to advance the technology.

COBIE Toolkit is not an automated API. To produce a COBIE file, we needed to proceed with 11 steps. First of all, add-in is set up in the Revit project and is deployed by loading COBIE template. Then, users make systems and zones schedules. Next, it includes the families and types to be connected into the COBIE file through the schedule view of COBie2-ComponentsToSchedule. It shows additional properties of family in the property window, which will be used in the COBIE file. After inputting all data into that window, COBIE project level parameter field is populated by running an add-in application within Revit. Eventually, exports and then imports schedule data into Microsoft Excel and copies and pastes all data into the COBIE template worksheet one by one.

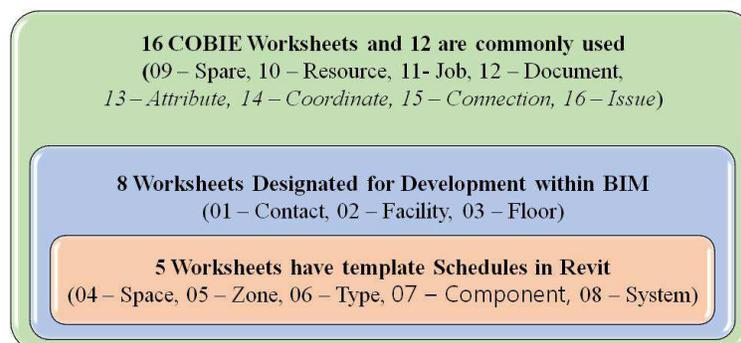


Figure 1 – COBIE spreadsheet; only 5 worksheets can be extracted from Revit project

METHODOLOGY

To produce a COBIE file, we need to export BIM model to external database, and then generate a COBIE file from DB like Figure 2. For the first step, we used ‘export to ODBC’ function to produce a DB for a project. Figure 3 shows the brief data structure of Revit. Before exporting the entire data, we set up the essential requirement to export to ODBC (Open DataBase Connectivity). Next, it is exported into a Microsoft SQL server. The reason for using DB is that “dynamic information repository (e.g., a DB management system) can store and return any information item generated or used in a process” (Lee et al.

2007). In the whole life cycle of facilities, huge amount of data should be stored and used very frequently. So, it is essential to utilize the merit of DB system. Additionally, it is easier both to remove irregularities and redundancies in a data model and to define a well-organized data structure.

In the second step, we generate a COBIE file from external DB by using a module which connects DB into COBIE. This module needs a mapping relationship between DB and COBIE data structure. We figure out what kind of tables and columns are embedded in Revit through a prototype BIM model. Table 3, for instance, shows the relationship between worksheet #2 ‘CONTACT’ in COBIE, which was not able to be produced by using Revit Toolkit, and DB. Most of the columns are filled with the data from DB, while some of them come from other worksheets. At the first column of ‘CONTACT’ worksheet, there is ‘Email’ which is a primary key. It is possible to get data from ‘ProjectAddress’ column of ProjectInformation’ table. It is not exactly the same, but we can match similar things to transfer data. It means that we have to input email address of architects or engineers into ProjectAddress field. Likewise, we match each data field like Table 3.

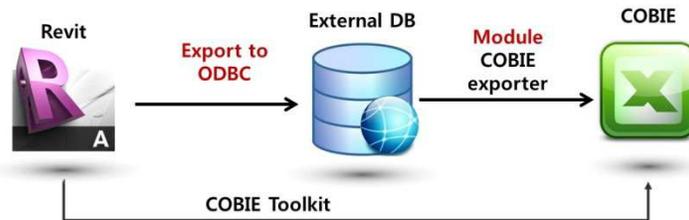


Figure 2 – Flow of data from Revit model to COBIE

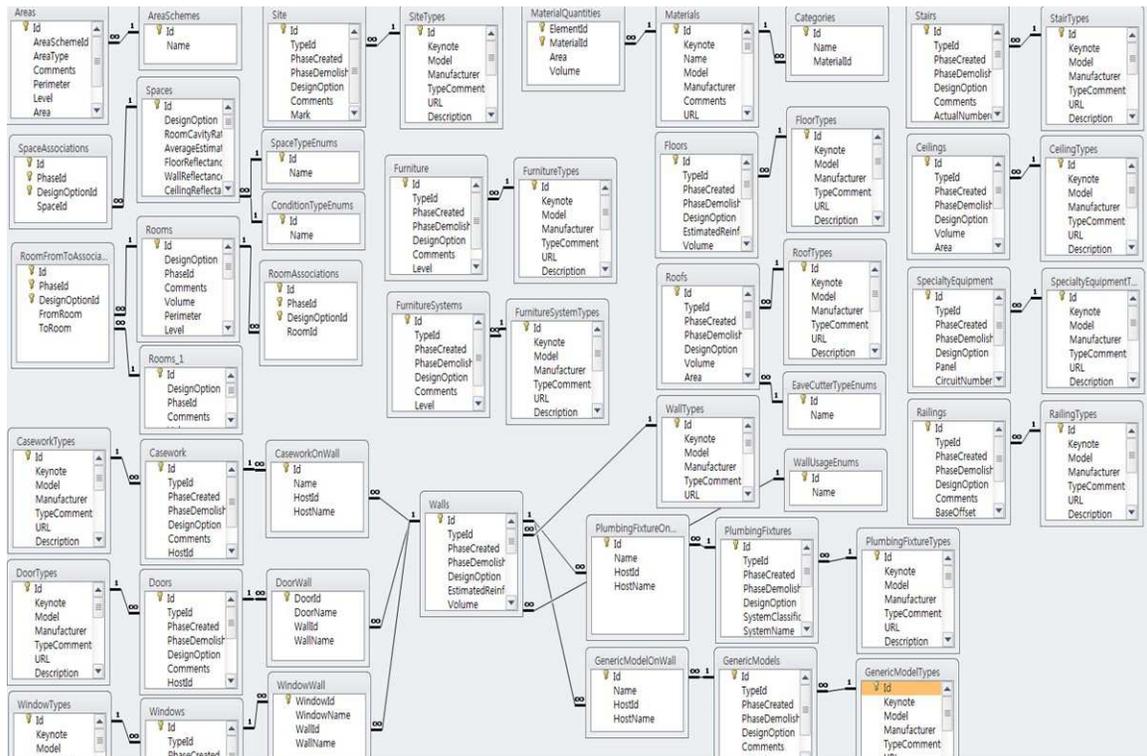


Figure 3 – DB structure of Revit

Table 3 –Mapping Relationship between External DB and COBIE data structure

COBIE		Database
Col A	Email	Primary Key: ProjectInformation - ProjectAddress
Col B	CreatedBy	*Contact – Email
Col C ~ D	CreatedOn Category Company Phone	ProjectInformation - ClientName *PickList - Category-Role ProjectInformation - ProjectName ProjectInformation - ProjectNumber
Col E ~ G	ExtSystem ExtObject ExtIdentifier	Name of SW. Fixed (Ex. Revit2013) *PickList – objContact ProejectInformation - ID
Col J ~ S	Department OrganizationCode GivenName FamilyName Street PostalBox Town StateRegion PostalCode Country	Profile – TypeName New field New field Profile – FamilyName ProjectInformation - ProjectInformation - New field New field New field New field

Now, based on the mapping relationship, we use the following query command to export data from SQL 2008 server to MS Excel. We use COBIE template file which has only the titles of each column and other data fields are empty. In this command, we locate the COBIE template file in C root folder. Following commands query data of ProjectAddress column in ProjectInformation table and transfer into Email column of CONTACT worksheet in COBIE template file.

```
insert into OPENROWSET('Microsoft.Jet.OLEDB.4.0',
'Excel 8.0;Database=C:\COBIE_Template.xls;',
'SELECT Email FROM Contact') select ProjectAddress from ProjectInformation
```

ExtSystem is unique data field and it contains the name of system. In this study, it should be 'Autodesk Revit' through the entire COBIE file. It is unique column and the value is not changeable. Thus, we input it as an invariable value. Likewise, we suggest mapping relationship for other COBIE worksheets: '01 – Contact', '02 – Facility', '03 – Floor', '04 – Space', '05 – Zone', '06 – Type', '07 - Component', and '08 – System'. With this mapping relationship, we generate COBIE file through the developed module from external DB.

DISCUSSION

Figure 4 shows an example of COBIE file produced from this study. It contains the information of 'Contract' to 'Space' worksheets which is not included in COBIE file created by Revit COBIE toolkit. By automatically producing COBIE file which contains more data than Revit COBIE toolkit product, we are able to reduce the total time to generate the final version of COBIE file. In this study, our object building consists of 2 stories and 4 offices including 14 spaces, 68 zones, 20 types, 50 components, and 50 systems. If we consider it takes the same time to input data into 'Common' set of COBIE worksheets in both ways, we can conclude that we save time as much as we input data into '01 – Contact', '02 – Facility', and '03 – Floor'. In this study, it is about 2.5 hours because we have to check the primary key, CreatedBy', at each worksheet as well as compare with other data field. In real projects, the more complex the building is, the more we can save time.

Additionally, transferring data directly from BIM modeling tool, it is not required to repeat and spending time by inputting the same D&C data at O&M phases. It enables both to reduce redundancy and reduce the chance to loss data.

CONTACT		B	C	D	E	F	G	H
1	Email	CreatedBy	CreatedOn	Category	Company	Phone	ExternalSystem	ExternalObject
2	jeffvi@tamu.edu	jeffvi@tamu.edu	2013-01-02T09:38:21	34-25 21 00 Architect	n/a	979-492-8644	Revit 2013	UID
3	jeffvi1221@gmail.com	jeffvi@tamu.edu	2013-01-02T09:38:21	31 11 00: Manufacturers	Texas A&M University	(999)999-9999	Revit 2013	UID
4	ihvunsabina@gmail.com	jeffvi@tamu.edu	2013-01-02T09:38:21	34-31 11: Manufacturer	Texas A&M University	(999) 999-9999	Revit 2013	UID
5	jeffvi@nate.com	jeffvi@tamu.edu	2013-01-02T09:38:21	34-65 11 11: Corporation	Texas A&M University	(999) 999-9999	Revit 2013	UID
6								
7								

FACILITY		C	D	E	F	G	H	I	J	
1	Name	CreatedBy	CreatedOn	Category	ProjectName	SiteName	LinearUnits	AreaUnits	VolumeUnits	CurrencyUnits
2	Williams	jeffvi@tamu.edu	2013-01-02T09:38:21	11-16 21 11 00: Campus	FM	Williams	Feet	SquareFeet	CubicFeet	\$
3										

FLOOR		C	D	E	F	G	H	I	J	
1	Name	CreatedBy	CreatedOn	Category	ExtSystem	ExtObject	ExtIdentifier	Description	Elevation	Height
2	Erdgeschoss	jeffvi@tamu.edu	2013-01-02T09:38:21	Floor	Revit 2013	Floor	3ZVjwdGuHzb84xOEUqVovl	n/a	0.00	3.10
3	Obergeschoss	jeffvi@tamu.edu	2013-01-02T09:38:21	Floor	Revit 2013	Floor	0Z4ZKrq4 LHqNivXTWVh7u	n/a	3.10	2.80
4	Dach	jeffvi@tamu.edu	2013-01-02T09:38:21	Roof	Revit 2013	Floor	0\$wTwYdc1OkYjk253hRjeK	n/a	5.90	1.00
5										

SPACE		C	D	E	F	
1	Name	CreatedBy	CreatedOn	Category	FloorName	Description
2	EG-101	jeffvi@tamu.edu	2013-01-02T09:38:21	13-51 24 11 00: General Residential Space	Erdgeschoss	Lounge
3	EG-201	jeffvi@tamu.edu	2013-01-02T09:38:21	13-51 24 11 00: General Residential Space	Erdgeschoss	Lounge
4	EG-202	jeffvi@tamu.edu	2013-01-02T09:38:21	13-11 19 11 11: Kitchen	Erdgeschoss	Kitchen-Diner
5	EG-204	jeffvi@tamu.edu	2013-01-02T09:38:21	13-85 31 11 00: Entry Vestibule	Erdgeschoss	Entrance
6	EG-104	jeffvi@tamu.edu	2013-01-02T09:38:21	13-85 31 11 00: Entry Vestibule	Erdgeschoss	Entrance
7	EG-102	jeffvi@tamu.edu	2013-01-02T09:38:21	13-11 19 11 11: Kitchen	Erdgeschoss	Kitchen-Diner
8	OG1-108	jeffvi@tamu.edu	2013-01-02T09:38:21	13-85 11 11 00: Corridor	Obergeschoss	Hallway
9	OG1-106	jeffvi@tamu.edu	2013-01-02T09:38:21	13-51 21 11 00: Bedroom	Obergeschoss	Office1
10	OG1-205	jeffvi@tamu.edu	2013-01-02T09:38:21	13-51 21 11 00: Bedroom	Obergeschoss	Office2
11	OG1-105	jeffvi@tamu.edu	2013-01-02T09:38:21	13-51 21 11 00: Bedroom	Obergeschoss	Office3
12	OG1-206	jeffvi@tamu.edu	2013-01-02T09:38:21	13-51 21 11 00: Bedroom	Obergeschoss	Office4
13	OG1-207	jeffvi@tamu.edu	2013-01-02T09:38:21	13-41 11 14 11: Bathroom	Obergeschoss	Bathroom
14	OG1-107	jeffvi@tamu.edu	2013-01-02T09:38:21	13-41 11 14 11: Bathroom	Obergeschoss	Bathroom
15	OG1-208	jeffvi@tamu.edu	2013-01-02T09:38:21	13-85 11 11 00: Corridor	Obergeschoss	Hallway
16						

Figure 4 – Example of COBIE file

CONCLUSION

In order to get more information transferred from BIM solutions to CAFM systems automatically, we developed a prototype application that extracts information from Revit, which is one of BIM authoring tools, stores extracted information into an external DB, and then produces the COBIE data file using information stored in the DB. We applied our application to prototype projects and demonstrated that it

produced 8 COBIE data sets automatically, which helped facilities managers save a significant amount of time.

Utilization of these modules reduced the time required to produce a COBIE file from design information. It also increased interoperability by directly delivering data from the design tool - Autodesk Revit in this research – and decreased the loss of information at each phase of the building project. It is expected that 12 data sheets of COBIE can be produced through additional investigation. Furthermore, using a private external DB has the advantage of vendor independency because software versions of BIM are no longer an issue.

REFERENCES

- Anne Anderson, Andrew Marsters, Carrie Sturts Dossick, and Gina Neff (2012) “Construction to Operations Exchange: Challenges of Implementing COBie and BIM in a Large Owner Organization”, *Construction Research Congress 2012*, 688-697.
- Ali Motamedi and Amin Hammad (2009). “RFID-Assisted Lifecycle Management of Building Components Using BIM Data”, *26th International Symposium on Automation and Robotics in Construction (ISARC 2009)*, 109-116.
- Autodesk (2013). “Using the COBie Add-in Demonstration Application for Autodesk Revit 2013”, Autodesk.
- Bill East (2013). “Construction Operations Building Information Exchange (COBIE)”, Retrieved from buildingSMART website:
<http://www.buildingsmartalliance.org/index.php/projects/activeprojects/25>
- Burcin Becerik-Gerber, Farrokh Jazizadeh, Nan Li, and Gulben Calis (2012) “Application Areas and Data Requirements for BIM-Enabled Facilities Management”, *Journal of Construction Engineering and Management*, 138 (3), 431–442.
- Christopher Riddell (2013). “BIM: Facility Management Tool of the Future”, Retrieved from website:
http://christopherriddell.com/Riddell_BIM_FM.pdf
- Colleen Kasprzak and Craig Dubler (2012) ‘Aligning BIM with FM: streamlining the process for future projects’, *Australasian Journal of Construction Economics and Building*, 12 (4) 68-77.
- Ghang Lee, Rafael Sacks, Charles Eastman (2007). “Product data modeling using GTPPM — A case study”, *Automation in Construction*, 16, 392–407.
- Jason Lucas (2012). *An Integrated BIM Framework to Support Facility Management in Healthcare Environments* (Doctor’s dissertation). Virginia Polytechnic Institute and State University, Virginia, US.
- JinHoon Lee, Ji-Hoon Lee, Hyo-Jin Kim, Sang-Ho Lee (2008). “An Extended Data Model based on the IFC for Representing Detailed Design Information of Steel Bridge Members”, *Journal of Computational Structural Engineering Institute of Korea*, 21 (2), 253-263. (ISSN: 1225-1569)
- Jin-Hoon Yi, HyunJung An, Hyo-Jin Kim & Sang-Ho Lee (2008). “Interoperable Information Model based on IFC for the Cable-Stayed Bridge Monitoring System”, *Proceedings of the Fourth International IABMAS Conference*, 352, Taylor & Francis 2008, (ISBN: 978-0-415-46844-2)
- Louise Sabol (2008). “Building Information Modeling & Facility Management”, IFMA World Workplace.
- Rafael Sacks, Charles Eastman, and Ghang Lee (2004). "Parametric 3D modeling in building construction with examples from precast concrete". *Automation in Construction*, 13 (3), 291-312.
- William East (2007). “Construction Operations Building Information Exchange (COBIE): Requirements Definition and Pilot Implementation Standard”, US Army Corps of Engineering. Retrieved from <http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA491932%26amp;Lo>
- William East (2012). "Construction Operation Building Information Exchange". USACE ERDC. Retrieved from Whole Building Design Guide: <http://www.wbdg.org/resources/cobie.php>
- William East (2013) “Construction Operations Building Information Exchange (COBIE): Means and Methods”. Retrieved from buildingSMART Alliance:
<http://www.buildingsmartalliance.org/index.php/projects/cobie>
-