

# Life Cycle information exchange (LCie): Product Type Template

by E. William East, PhD, PE, F.ASCE and Danielle Love - Engineer Research and Development Center, U.S Army, Corps of Engineers and Nicholas Nisbet MA (Cantab) DipArch (UNL) - AEC3

## BACKGROUND

Today there is no common minimum specification for BIM objects within designs. This results in significant confusion by those reviewing BIM output from different vendors and in using BIM data during the life of the project. Designers placing building model objects need commonly accepted templates published through the Specifiers Properties information exchange (SPie) project templates. These templates are in the process of being coordinated with industry associations and the current status of each template are provided in the United States. Additional information about the SPie project may be found on the buildingSMART alliance project page.

## BUSINESS CASE

Product manufacturers today are being bombarded with requirements to produce BIM models in a variety of non-backwardly compatible formats. In addition publishers of these models are charging manufacturers for the right to distribute the manufacturers' own information. The business case for software developers is that common methods for delivering product template models can reduce the cost of such functionality for all software developers allowing the software to focus on core-missions instead of having scores of people dedicated to manually loading product data into catalogs. For the designer, contractor, and facility manager the delivery of common product information provides the basis for streamlining the entire life-cycle product logistic chain.

## EXAMPLE CONTRACT CLAUSE

Contract clauses for design products should include the requirement that all scheduled equipment and named products in the model must conform to the common object property templates provided through the Whole Building Design Guide's Product Guide. Software that is allowed for the purpose of design, construction, and facility management should be self-certified to demonstrate the use of these product type templates as is appropriate for the specific software domain. Specification and BIM software should also be linked through the Product Type Template to allow the specification of three or equal products defined as Product Type Candidates (as described in the Product Type Candidate exchange page).

## ORGANIZATION

The buildingSMART international Information Delivery Manual process identifies information exchanges according to the table shown below. Use this table to determine if this information exchange applies to your area of responsibility for a given project.

Code	Phase	Used
0	Portfolio requirements	
1	Conception of need	
2	Outline feasibility	
3	Substantive feasibility	
4	Outline conceptual design	
5	Full conceptual design	
6	Coordinated design and procurement	✓
7	Production information	
8	Construction	
9	Operation and maintenance	

10	Disposal	
----	----------	--

The buildingSMART alliance classifies information exchanges according to a number of different classification tables, called OmniClass, provided by the Construction Specification Institute. In addition to OmniClass references to the subject exchange, the buildingSMART alliance provides an overall business activity diagram node referenced in the table below.

LCie Worksheet	OmniClass Table 31 Phase	OmniClass Table 34 Actors	OmniClass Table 32 Services	Activity Node Tree
Blank object component	<b>31-20 20 24</b> Product Selection Phase  <b>31-20 20 27</b> Material Selection Phase  <b>31-20 20 31</b> Equipment Selection Phase	<b>34-25 41 00</b> Specifier	<b>32-11 45 00</b> Specifying	<b>3.32</b> Coordinate to Find Compatibles

## EXCHANGES

The sections below describe the inputs required to apply this information exchange. The processing that is accomplished to process these inputs, and the resulting outputs that should be expected as a result of performing this information exchange. This information exchange can be characterized as a “transactional” update of the as-built construction building model. A general description of the requirements for transactional exchanges is found in the LCie Overview (URL).

### Inputs

Product Type Template requires the user authentication and project authorization wrappers described in the LCie Overview (URL). In addition, the information below is the minimum data set that will be processed; however, additional worksheets may be provided by the creator of the COBie file. Please note that additional worksheets may be needed to produce a proper ifc file.

- Type Worksheet
  - The following required COBie fields must be present in this file:
    - Type name
    - Category
    - Description
    - Asset type
    - Who created the entry and when
  - The following required COBie fields may not be present in this file:
    - Manufacturer
    - Model number
    - Warranty guarantor-parts
    - Warranty duration-parts
    - Warranty guarantor-labor
    - Warranty duration-labor
    - Warranty duration unit
    - Replacement cost

- Expected life and it's duration unit
- Warranty description
- Attribute worksheet.

The following table summarizes the expected content in the COBie file. Referenced rows are for informational purposes and should not be changed. New rows require the addition of new row items to the designated worksheet. Updated rows require the addition of information to an existing row item. It may also be appropriate to add a new row item to a worksheet as part of an update. Optional rows are not required but will be processed if provided.

Key: Referenced Rows= Y or – (not reqd.)  
 New Rows= Y or – (not reqd.)  
 Updated Rows = Y or – (not reqd.)  
 Optional Rows = Y or – (not reqd.)

Worksheet	Referenced Rows	New Rows	Updated Rows	Optional Rows
Facility	Y	-	-	-
Floor	-	-	-	-
Space	-	-	-	-
Zone	-	-	-	-
Type	-	-	Y	-
Component	-	-	-	-
System	-	-	-	-
Spare	-	-	-	-
Resource	-	-	-	-
Job	-	-	-	-
Document	-	-	-	-
Attribute	-	-	Y	-
Connection	-	-	-	-
Coordinate	-	-	-	-
Issue	-	-	-	-

## Processing

The capture of this transactional information may be seen as a type of building information survey where the appropriate portion of the building information is requested to generate a data entry form, the user completes that form, and the information is returned to update the building information.

### Preparation of building information template

The first stage is the preparation of a template data set from the current building information. Implementation of the template information may be accomplished through specific software solutions using appropriate menus. To create realistic examples, bimServices demonstration scripts automatically create COBie spreadsheets containing the minimum set of information needed to provide the required BIM sub-set.

- Select current project from building information database
- Select specific product type from current project
- Generate product type template data entry form
- Provide product type template data entry form

### Building information capture

The second stage is the capture of the required new information in the data entry form. As with step one this would be expected to be accomplished within proprietary software solutions. To provide a realistic example the COBie file provided in the first step may be used to:

- Access product type template data entry form
- Provide required attribute data

### Building information transmission and processing

The final stage is the transmission and processing of that information by the target information system. For this specific information exchange the following steps are required. To provide a realistic example the completed COBie building information survey form is used as the input artifact that updates the model.

- Access product type template data entry form
- Provide user credential information (if needed)
- Provide required attribute data
- Send information to the building information server

Processing information sent to the building information server will require the following steps.

- User authentication
- User authorization
- Checking file compliance with COBie
- Checking file compliance with LCie exchange requirements
- Checking the requested transaction with targeted information
- Backing-up prior building information
- Identification of matching type
- Updating mandatory attribute information
- Completion of the transaction and reporting

### Output

There are two types of outputs created with this transaction. The first is the creation of the product type template form. This output may be shown on a screen as part of an information system or may be produced as a standalone template file, as is accomplished with the bimServices engine.

The second outputs are files that demonstrate proper processing of the submitted information. The following reports would be expected:

- Incoming file compliance with COBie
- Incoming file compliance with information exchange requirements (identification of optional data)
- Verification of mapping to target model
- Results of completing the transactions
- Comparison of prior and current model states.

### Follow On

The following processes are expected to occur after or concurrently with this process:

- Other product type template reports
- Product Type Candidate

### EXAMPLES

The LCie project has two example projects, a duplex apartment and a medical clinic. For each example project, product type template COBie files have been created. The type and attribute worksheets in

each product type template file have been completed in accordance with the COBie instruction worksheet. Instances of product type template files for both example projects can be found below.

## Duplex Apartment

- Example 1:
  - Input:
    - Prior building model  
(DuplexApartment\_ProductTypeTemplate\_SwingingDoor\_0.750\_before)
    - Exported template for Product Type Template (URL)
    - Completed template for Product Type Template  
(DuplexApartment\_ProductTypeTemplate\_SwingingDoor\_0.750)
  - Output:
    - Incoming file compliance with COBie
    - Incoming file compliance with exchange requirements (identification of optional data)
    - Verification of mapping to target model
    - Results of completing the transactions
    - Comparison of prior and current model states.
  
- Example 2:
  - Input:
    - Prior building model  
(DuplexApartment\_ProductTypeTemplate\_SwingingDoor\_0.885\_before)
    - Exported template for Product Type Template (URL)
    - Completed template for Product Type Template  
(DuplexApartment\_ProductTypeTemplate\_SwingingDoor\_0.885)
  - Output:
    - Incoming file compliance with COBie
    - Incoming file compliance with exchange requirements (identification of optional data)
    - Verification of mapping to target model
    - Results of completing the transactions
    - Comparison of prior and current model states.

## Medical Clinic

- Example 1:
  - Input:
    - Prior building model (MedicalClinic\_ProductTypeTemplate\_DoorTypeA\_before)
    - Exported template for Product Type Template (URL)
    - Completed template for Product Type Template  
(MedicalClinic\_ProductTypeTemplate\_DoorTypeA)
  - Output:
    - Incoming file compliance with COBie
    - Incoming file compliance with exchange requirements (identification of optional data)
    - Verification of mapping to target model
    - Results of completing the transactions
    - Comparison of prior and current model states.
  
- Example 2:
  - Input:

- Prior building model (MedicalClinic\_ProductTypeTemplate\_DoorTypeB\_before)
- Exported template for Product Type Template (URL)
- Completed template for Product Type Template (MedicalClinic\_ProductTypeTemplate\_DoorTypeB)
- Output:
  - Incoming file compliance with COBie
  - Incoming file compliance with exchange requirements (identification of optional data)
  - Verification of mapping to target model
  - Results of completing the transactions
  - Comparison of prior and current model states.

## Software Implementation Guidance

### SCRIPTED PROCESS

To recreate the example files identified in this information exchange documentation the bimServices engine was used based on information from two projects a Duplex Apartment building and a Clinic building. The following batch file was used to process the appropriate files through the bimServices engine.

```
echo off
set a0=ProductTypeTemplate
rem Replace %1 with MedicalClinic or DuplexApartment
set a1=%1
rem Replace %2 with appropriate type name
set a2=%2

call goCl      %a1% %a0%_%a2%
call goMerge2  %a1% %a0% %a2%
call goIC      %a1% %a0%_%a2%

time /t
```

Figure 1 doProductTypeTemplate batch file