การศึกษา “Industrial Foundation Classes” สำหรับสร้างแบบจำลองอาคารด้วยโปรแกรม FreeCAD

A Study of “Industrial Foundation Classes” for Create Building Model using FreeCAD Program.

จักรี ติยะวงศ์สุวรรณ[1]*, จิรัฐิติ์ บรรจงศิริ

สาขาวิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเอเชียอาคเนย์ กรุงเทพมหานคร 10160

*E-mail: chakkreet@sau.ac.th

บทคัดย่อ

ปัจจุบันได้เริ่มมีการนำ BIM มาประยุกต์ใช้งานกับโครงการต่างๆ ในการก่อสร้างอาคาร โดยใช้รูปของไฟล์กลางในการแลกเปลี่ยนข้อมูล คือ ไฟล์ IFC โดยสามารถสร้างหรือส่งออกได้ด้วยโปรแกรมใช้งานต่างๆ วัตถุประสงค์ในการศึกษานี้คือ เพื่อสร้างไฟล์ IFC จากโปรแกรมรหัสเปิด (Open Source) หรือโปรแกรมที่ไม่คิดมูลค่า(Freeware) โดยจะทำการศึกษาถึงโครงสร้างข้อมูลของไฟล์ IFC ที่สร้างผ่านโปรแกรม FreeCAD และ ifcOpenShell และค้นหาแนวทางการจัดการข้อมูลไฟล์ IFC ด้วยโปรแกรมสำหรับดูข้อมูล เช่น Solibri Model Viewer หรือ BIM Vision เป็นต้น จากการศึกษาเบื้องต้นพบว่าสามารถส่งออกชิ้นส่วนอาคารเป็นไฟล์ IFC ได้ในระดับเบื้องต้น

คำสำคัญ : แบบจำลองสารสนเทศอาคาร, หมวดข้อมูลงานพื้นฐานอุตสาหกรรมก่อสร้าง, ฟรีแคด

Abstract

BIM has been being applied to the construction project by using an IFC file. The data exchange in the form of an IFC file can be created or exported by using various applications. The purpose of this study is to create an IFC file on open source program, or freeware. We have studied the structure of IFC files that are generated by FreeCAD and ifcOpenShell. Then we have sought the approach for IFC files management on Solibri Viewer or BIM Vision. In the basic level, the results show that parts of building can be exported as an IFC file.

Keywords : BIM, IFC, FreeCAD, ifcOpenShell

1. ที่มาและความสำคัญ

ในปัจจุบันเทคโนโลยีทางด้านคอมพิวเตอร์สำหรับการจัดทำแบบก่อสร้างได้พัฒนาก้าวหน้าขึ้นมาก ได้มีการพัฒนาเทคโนโลยีใหม่เข้ามาใช้งาน โดยในขบวนการออกแบบอาคารจะเริ่มต้นจากการกำหนดแบบจำลอง(Model) แต่ในการสร้างแบบอาจจะมีผู้ที่มีส่วนร่วมในการออกแบบหลายส่วนงาน หลายองค์กร แต่ละองค์กรใช้งานโปรแกรมที่แตกต่างกันไป ทำให้มีปัญหาในการแลกเปลี่ยนข้อมูล ในอดีตการทำงานจะสร้างแบบก่อสร้างชนิดสองมิติไฟล์ที่ใช้แลกเปลี่ยนจะใช้ DXF เป็นไฟล์กลางในการแลกเปลี่ยน แต่หลังจากที่ต้องมีการแลกเปลี่ยนข้อมูลที่มีลักษณะของแบบจำลองสามมิติพร้อมข้อมูล ทางบริษัท AutoDesk จึงได้ร่วมกันกับบริษัทอื่นๆ ก่อตั้ง International Alliance for Interoperability(IAI) มีวัตถุประสงค์ที่จะร่วมกันปรับปรุงข้อมูลกลางสำหรับแลกเปลี่ยนระหว่างโปรแกรมในอุตสาหกรรมก่อสร้าง[1] และพัฒนาเป็น Industry Foundation Classes(IFC) ซึ่งต่อมา IAI ได้เปลี่ยนชื่อเป็น buildingSMART

IFC ข้อมูลมาตรฐานเปิด เป็นไฟล์ข้อความที่มนุษย์อ่านและตีความได้ กำหนดตาม ISO 10303-21 ("STEP-File") ปัจจุบันได้ประกาศ IFC4 เป็นรุ่นล่าสุด แต่ยังหาโปรแกรมมารองรับการใช้งานได้น้อย ในรุ่น IFC2x3TC1 จะมีโปรแกรมรองรับการแสดงผล แต่ในปัจจุบันการสร้างไฟล์ข้อมูล IFC จะต้องใช้โปรแกรมเชิงพาณิชย์ที่มีราคาสูงในการสร้าง ดังนั้นโครงสร้างหรืองานออกแบบที่มีขนาดเล็กอาจจะไม่สามารถมีเงินลงทุนที่จะจัดหาโปรแกรมเหล่านั้น ดังนั้นจึงควรทำการศึกษาหาโปรแกรมที่สามารถใช้งานทดแทนได้

2. วัตถุประสงค์

1. ศึกษาโครงสร้างไฟล์ IFC เพื่อใช้เป็นตัวกลางในการแลกเปลี่ยนสำหรับขบวนการ BIM
2. ศึกษาการนำเข้าและส่งออก IFC เข้าสู่ FreeCAD ด้วยโปรแกรม FreeCAD ตามความสามารถในปัจจุบัน
3. ศึกษาการสร้าง Element ด้วยโมดูล IfcOpenShell-Python
4. ศึกษาการดูไฟล์ IFC ด้วยโปรแกรม IFC Viewer ต่าง ๆ เพื่อตรวจสอบความสามารถในการใช้งาน

3. ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

3.1. หมวดข้อมูลงานพื้นฐานอุตสาหกรรมก่อสร้าง (Industrial Foundation Classes)

หมวดข้อมูลงานพื้นฐานอุตสาหกรรมก่อสร้าง (Industrial Foundation Classes) เป็นข้อมูลกลางสำหรับการแลกเปลี่ยนข้อมูลสำหรับโปรแกรมที่ใช้ในแบบจำลองสารสนเทศอาคาร (Building Information Models) เพื่อให้แต่ละโปรแกรมสามารถทำงานร่วมกันได้ (Interoperability) โดยจะมีการออกแบบข้อมูลในส่วนงานต่างๆ เช่น สถาปัตยกรรม, ออกแบบโครงสร้าง, งานระบบท่อ, งานควบคุมอาคารเป็นต้น

ไฟล์ IFC จะอยู่ในรูปแบบของไฟล์ข้อความ โดยจะแบ่งข้อมูลออกเป็นสองส่วนหลัก คือ ส่วนแรกคือส่วนหัวไฟล์ ดังแสดงตัวอย่างใน
รูปที่ **1** (ก) จะเก็บข้อมูลผู้สร้าง(ชื่อบริษัท+ชื่อผู้รับผิดชอบ) โปรแกรมที่ใช้สร้าง วันที่สร้างไฟล์ และรูปแบบของไฟล์ IFC ในตัวอย่างจะเป็นไฟล์สกีมารุ่น IFC2x2_Final และส่วนที่สองคือส่วนเก็บข้อมูลตัวอย่างใน

รูปที่ **1** (ข) จะเก็บข้อมูลอาคารของโครงการไว้ทั้งหมดซึ่งจะอยู่ในรูปแบบของอิลีเมนต์ข้อมูลที่กำหนดด้วยหมายเลขที่ใช้ในการอ้างอิง อิลีเมนต์จะเป็นชนิดข้อมูลต่างๆ เช่น ข้อมูลจุดพิกัด (IfcCartesianPoint) ระบุด้วยชื่อของอิลีเมนต์ และภายในวงเล็บจะเป็นข้อมูลคุณสมบัติต่างๆ ของอิลีเมนต์นั้นๆ จะถูกกำหนดโดยมาตรฐานของ buildingSMART

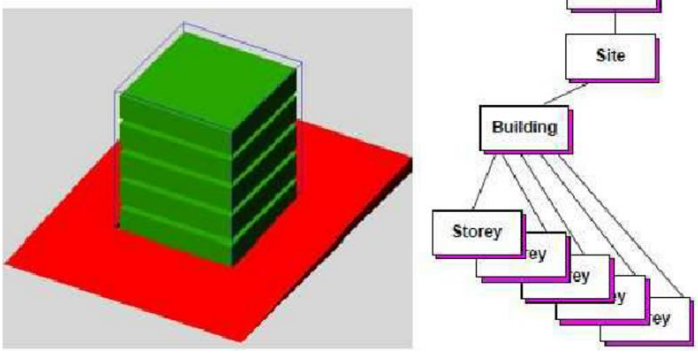
```
HEADER;
FILE_DESCRIPTION(('IFC2x platform'),'2;1');
FILE_NAME(
'Example.dwg',
'2005-09-02T14:48:42',
('The User'),
('The Company'),
'The name and version of the IFC preprocessor',
'The name of the originating software system',
'The name of the authorizing person');
FILE_SCHEMA(('IFC2X2_FINAL'));
ENDSEC;
```

(ก) ข้อมูลในส่วนหัวไฟล์

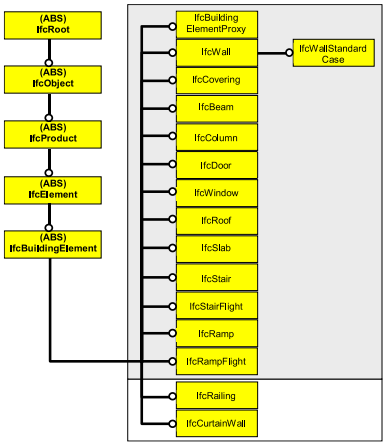
```
DATA;
#7=IFCCARTESIANPOINT((0.,0.,0.));
#8=IFCDIRECTION((0.,0.,1.));
#9=IFCDIRECTION((1.,0.,0.));
#10=IFCAXIS2PLACEMENT3D(#7,#8,#9);
ENDSEC;
```

(ข) ข้อความในส่วนเก็บข้อมูล

รูปที่ 1: ส่วนประกอบของไฟล์ข้อมูล IFC [2]



(ก) โครงสร้างข้อมูลเชิงพื้นที่ของอาคารในโครงการ



(ข) แผนภูมิลำดับชั้นขององค์ประกอบอาคาร

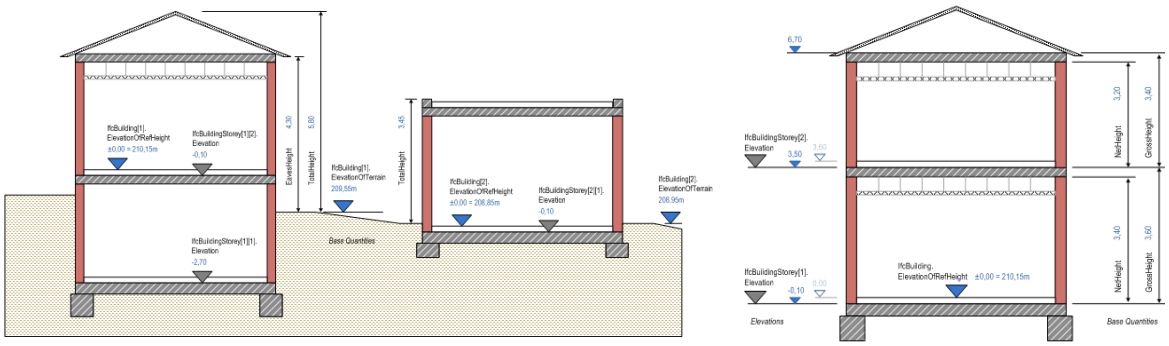
รูปที่ 2: แผนภาพของข้อมูลและความสัมพันธ์ชิ้นส่วนอาคารใน Industrial Foundation Classes [3]

ฐานข้อมูลในไฟล์ IFCหนึ่งไฟล์จะมีข้อมูลของ *IfcProject* อยู่หนึ่งชุดเท่านั้นภายในเนื้อหาจะมีข้อมูลของ *ชื่อ*, *คำอธิบาย รายละเอียด*, *ชนิดวัตถุ* (Object Type) , *ชื่อแบบยาว* และ *ระยะของโครงการ* (Phase) ในแต่ละโครงการจะต้องมีข้อมูลการแสดงผล (representation) ทั้งหมด ไม่ว่าจะเป็น หน่วย (Unit) ที่ใช้ในโครงการ พิกัดของโครงการและข้อมูลอื่นที่ใช้ในมุมมอง(View) ดังแสดงตัวอย่างด้านล่าง

```
#1=IFCPROJECT('2mifZkraLA4xt6Lfl2GhCQ', #2, 'Shopping mall',
  'New development project in the suburban district', 'Development project',
  'Shopping mall at High Street', 'Design', (#3), #4);
```

การกำหนดผู้เป็นเจ้าของโครงงานหรือผู้รับผิดชอบไฟล์ IFC จะกำหนดผ่าน *IfcOwnerHistory* ซึ่งจะประกอบได้ข้อมูลของบุคคลผู้รับผิดชอบ(IfcPerson),องค์กร(IfcOrganization) และโปรแกรมที่ใช้ในการสร้าง(IfcApplication) เป็นต้น ดังแสดงตัวอย่างด้านล่าง

```
#1=IFCBUILDING('WJ7yGd42kMW08n7Cp432Kr',#2,'Sample Building',$,$,$,$,$,$,$,$,$);
#2=IFCOWNERHISTORY(#3,#6,.READWRITE.,.ADDED.,87763554,#3,#6,87763554);
#3=IFCPERSONANDORGANIZATION(#4,#5,());
#4=IFCPERSON('Public','Jane','Q.',(),(),(),(),());
#5=IFCORGANIZATION($,'Architecture by Jane Q. Public, Inc.',$,(),());
#6=IFCAPPLICATION(#7,'Version 1.0','Building Architecture Toolkit','BAT1.0');
#7=IFCORGANIZATION($,'Creating Instance Software, Inc.',$,(),());
```



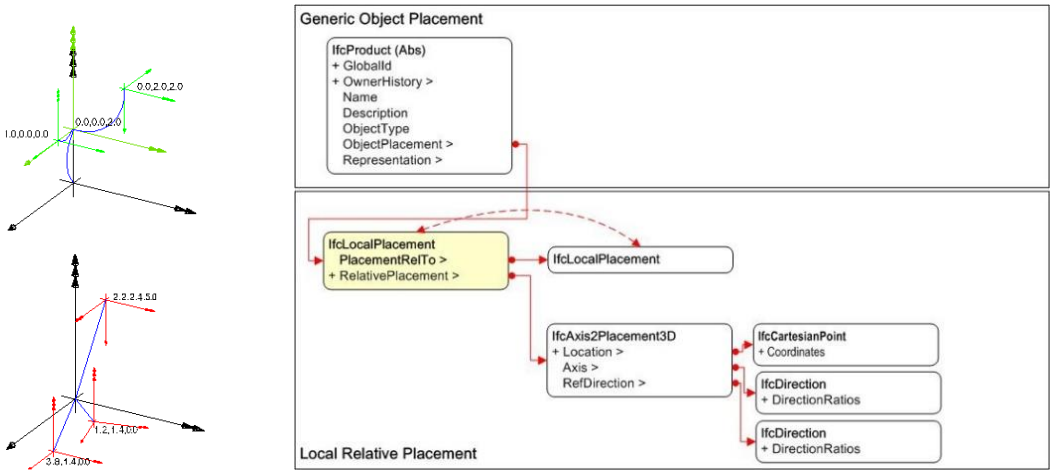
รูปที่ 3: ข้อมูล IfcBuildingStorey [2,3]

ในการกำหนดข้อมูลของอาคารจะกำหนดผ่าน *IfcBuilding* โดยจะอ้างอิงกับข้อมูลของหน้างาน(IfcSite) และข้อมูลแต่ละชั้นของอาคารจะกำหนดด้วย *IfcBuildingStorey* ซึ่งแต่ละข้อมูลจะอ้างอิงกับตำแหน่งผ่านข้อมูล *IfcLocalPlacement* และตำแหน่งจะใช้ข้อมูลจุดพิกัด(IfcCartesianPoint) และทิศทางในการกำหนดอ้างอิง(IfcDirection) ดังแสดงข้อมูลใน

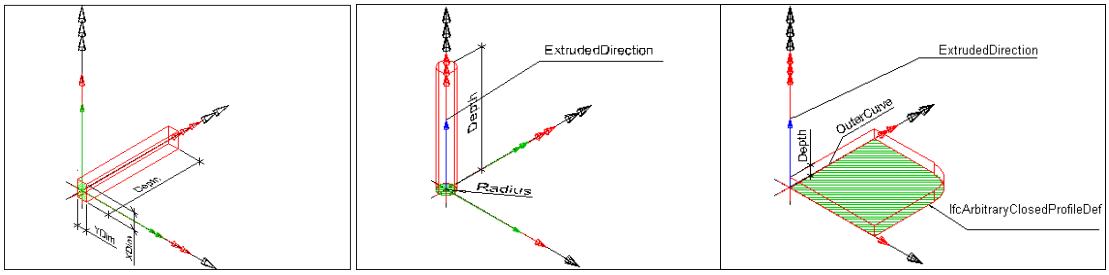
รูปที่ **3** และรูปที่ 4

สำหรับการแสดงชิ้นส่วนอาคารจะมีชิ้นส่วนต่างๆ เช่น IfcBeam, IfcColum, IfcSlab และ IfcStair เป็นต้น โดยกำหนดข้อมูลจะเป็นไปตามข้อมูลของ BuildingSMART เช่น XDim, YDim หรือ Depth เป็นต้นดังแสดงใน

รูปที่ **5**



รูปที่ 4: ระบบการอ้างอิงตำแหน่งของ IfcLocalPlacement [2,3]



รูปที่ 5: ชิ้นสวน IfcBeam, IfcColumn และ IfcSlab [2,3]

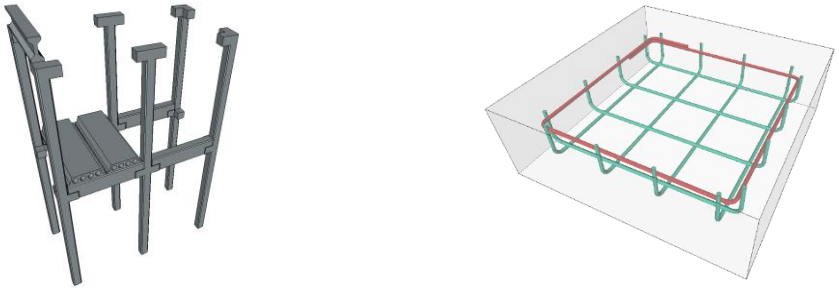
3.2. โปรแกรม FreeCAD

เป็นโปรแกรมที่ใช้ในการสร้างแบบจำลองสามมิติแบบปรับค่าได้ เป็นโปรแกรมฟรีแวร์ที่มีนักพัฒนาเข้าร่วมกันพัฒนาหนึ่งในทีมนักพัฒนาที่อยู่ในงานสถาปัตยกรรมได้พัฒนาส่วนงาน Arch Workbench[4, 5] สำหรับใช้งานสร้างชิ้นอาคาร ประตู หน้าต่าง กำแพง เป็นต้น และสามารถส่งออกข้อมูลของอาคารออกเป็นไฟล์ IFC ได้ ถึงแม้ว่าในปัจจุบันโปรแกรมจะมีความสามารถไม่สูงมากเมื่อเทียบกับโปรแกรมจัดการ BIM ตามท้องตลาด แต่จัดได้ว่าเป็นโปรแกรมที่อยู่ระหว่างการพัฒนาและมีชุมชนออนไลน์ที่เข้มแข็งที่พร้อมให้ความแนะนำช่วยเหลือ

ภานุทัตต์ และคณะ[6] ได้ใช้โปรแกรม FreeCAD ในการสร้างแบบจำลองสามมิติของบ้านสองชั้น และทำการตัดหน้าตัดของอาคารนำไปแสดงแบบในกระดาษเขียนแบบได้ ดังแสดงในรูปที่ 7 และส่งออกเป็นไฟล์ IFC ซึ่งจะเห็นได้ว่าสามารถใช้งานได้ดีสำหรับชิ้นส่วนรูปทรงมาตรฐาน

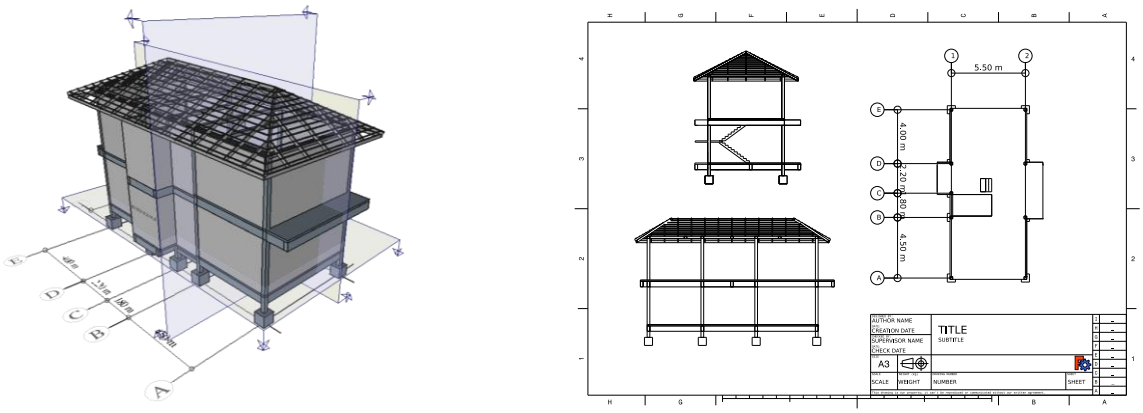
จักรี [7] ได้พัฒนาภาษา Python สำหรับวิเคราะห์โครงถักด้วยวิธีแมทริกซ์ ทำการป้อนข้อมูลในลักษณะตัวเลขและข้อความตัวแปร(Text Mode) แสดงผลลัพธ์เป็นแบบสามมิติในโปรแกรม FreeCAD ได้ ดังแสดงใน

รูปที่ **8** ดังนั้นจะเห็นได้ความตัวโปรแกรม FreeCAD มีความยืดหยุ่นสามารถเพิ่มเติมความสามารถโดยผู้ใช้งานได้



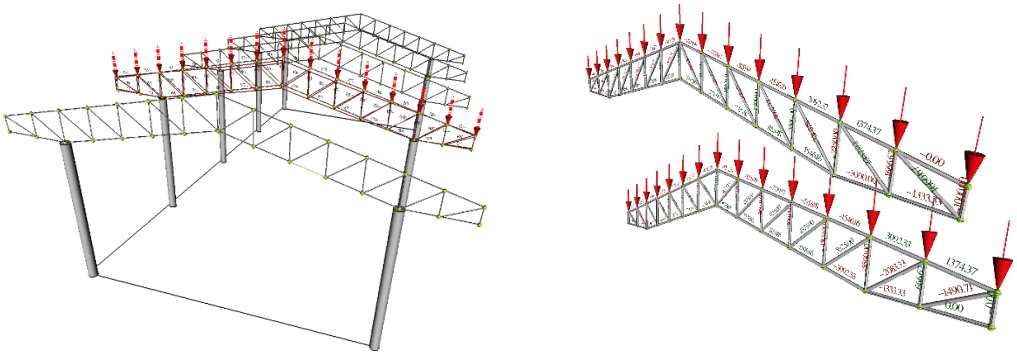
(ก) โมดูล ArchPrecast สำหรับชิ้นส่วนสำเร็จรูป (ข) โมดูล ArchRebar2 สำหรับเหล็กเสริมในคอนกรีต

รูปที่ 6: โปรแกรม FreeCAD [4,5]



(ก) แบบจำลองสามมิติใน FreeCAD (ข) รูปตัดและแบบแปลน(Drawing WB) ของแบบบ้าน

รูปที่ 7: การสร้างแบบจำลองสามมิติและแบบแปลนอาคารด้วยโปรแกรม FreeCAD [6]



(ก) แบบจำลองสามมิติวิเคราะห์โครงถักโรงงาน (ข) การวิเคราะห์โครงถักสองมิติ

รูปที่ 8: การวิเคราะห์โครงถักด้วยโมดูลภาษา Python ในโปรแกรม FreeCAD[7]

3.3. โมดูล IfcOpenShell-Python

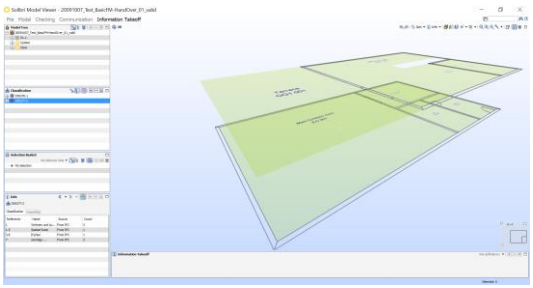
พัฒนาโดย Krijnen[8] เป็นภาษา C++ และได้พัฒนาส่วนขยายให้ใช้งานกับโปรแกรมต่างๆ ได้เช่น Blender และได้จัดทำส่วน Wrapper ให้ใช้งานได้ในโปรแกรม Python และโปรแกรม FreeCAD ตั้งแต่รุ่น 0.16 ได้ติดตั้ง IfcOpenShell-Python เรียบร้อยแล้วและส่วนการส่งออกไฟล์ IFC ของโปรแกรม FreeCAD ได้ใช้งาน IfcOpenShell เป็นหลักตั้งแต่รุ่น 0.16 ในการใช้งานสามารถ IfcOpenShell สามารถเรียกผ่าน Python Console หรือจะใช้งานในรูปแบบไฟล์มาโครภาษา Python ได้ ทำให้ผู้ใช้มีความยืดหยุ่นในการสร้างไฟล์ IFC เองได้ ในปัจจุบันยังอยู่ระหว่างการพัฒนาทำให้มีการนำไปใช้งานน้อย ดังนั้นจึงเห็นว่าควรทำการวิจัยและพัฒนาในส่วนนี้เพิ่มเติม

3.4. โปรแกรม IFC Viewer

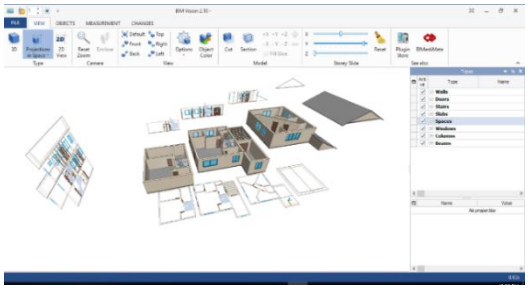
โปรแกรมที่ใช้สำหรับเปิดไฟล์ IFC เพื่อใช้ในการเข้าถึงข้อมูลของงานอาคาร โดยทั่วไปในปัจจุบันยังอยู่ในระดับ 3D BIM นั่นคือ สามารถดูแบบจำลองสามมิติได้เท่านั้น แต่ยังไม่สามารถดูข้อมูลในลักษณะ 4D หรือ 5D BIM ซึ่งจะสามารถกำหนดข้อมูลเวลา

และราคาได้ สำหรับความสามารถในระดับขั้นสูงในการใช้งานข้อมูลจะต้องใช้งานโปรแกรมเชิงพาณิชย์ โปรแกรม IFC Viewer ชนิดฟรีแวร์มีจำนวนมากที่ต้องทำการศึกษาในระดับเบื้องต้นมีโปรแกรมฟรีแวร์ที่สามารถเปิดไฟล์ IFC เพื่อใช้งานแบบจำลองที่ใช้ในการศึกษาของบทความนี้ มีรายชื่อดังต่อไปนี้

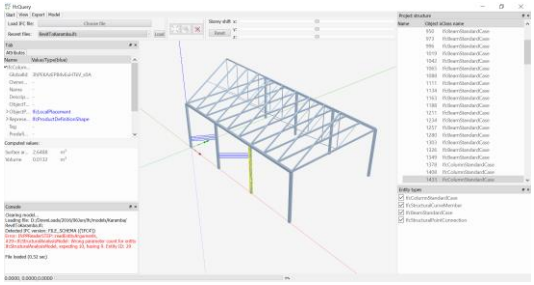
- โปรแกรม Solibri Model Viewer[9] สามารถแสดงข้อมูลแบบ Zone และแสดง FootPrint ได้
- โปรแกรม BIM Vision[10] สามารถเลื่อนชั้นเพื่อพิจารณาแต่องค์ประกอบในในแต่ละชั้นได้
- โปรแกรม IFC++[11] สามารถดูข้อมูลของ *IfcStructure* ได้
- โปรแกรม Constructivity[12] สามารถแสดงแผนภูมิแกนต์ของอิลีเมนต์ *IfcTask* ได้



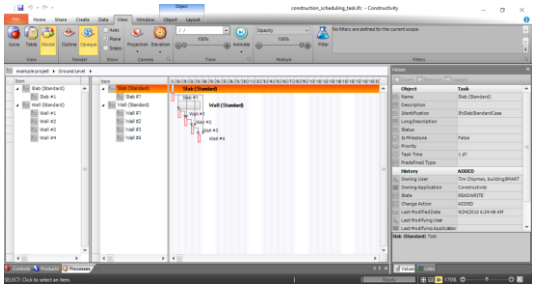
(ก) โปรแกรม Solibri Model Viewer[9]



(ข) โปรแกรม BIM Vision[10]

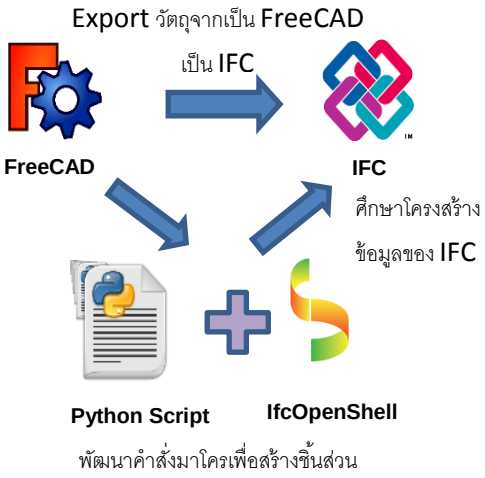


(ค) โปรแกรม IFC++[11]



(ง) โปรแกรม Constructivity[12]

รูปที่ 9: โปรแกรม IFC Viewer ต่างๆ สำหรับเปิดไฟล์ IFC



รูปที่ 10: ขั้นตอนการดำเนินการวิจัย

4. วิธีดำเนินการวิจัย

4.1. วิธีการดำเนินการวิจัย

1) ค้นคว้าหาข้อมูลเอกสารเกี่ยวกับ IFC เพื่อทำการศึกษาโครงสร้างข้อมูล IFC และหาไฟล์ IFC สำหรับศึกษาตัวอย่าง และตรวจสอบโปรแกรม IFC Viewer ต่างๆ ที่แสดงผลของไฟล์ IFC

2) ศึกษาการ นำเข้าและส่งออกไฟล์ IFC ด้วยโปรแกรมFreeCAD

3) ศึกษาการใช้โมดูล IfcOpenShell-Python สร้างไฟล์ IFC

4) ศึกษาการใช้งานไฟล์ IFC ด้วยโปรแกรมฟรีแวร์ต่างๆ เพื่อศึกษาขอบเขตที่จะสามารถนำมาใช้งานได้ในปัจจุบัน

4.2. ขอบเขตการวิจัย

1) ศึกษาโครงสร้างของ IFC รุ่น IFC2x3TC1 และพัฒนาคำสั่งภาษา Python ให้สามารถสร้าง IFC ได้โดยตรง

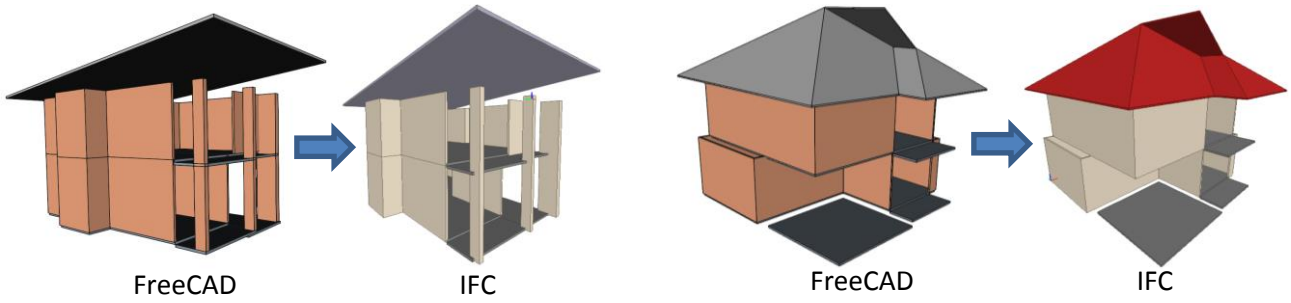
2) ใช้โปรแกรม FreeCAD รุ่น 0.16(Release) และ รุ่น 0.17(Development) และพัฒนาภาษา Python ภายใต้ Console และ Macro Editor เนื่องจากโมดูล IfcOpenShell-Python ได้ถูกตั้งรวมมากับ FreeCAD แล้ว

5. ผลและวิจารณ์

5.1. อภิปรายผล

5.1.1 การใช้โปรแกรม FreeCAD ในการสร้างไฟล์ IFC

โดยหลักการเบื้องต้นโปรแกรม FreeCAD จะสามารถสร้างชิ้นส่วนอาคารแล้วส่งออกเป็นไฟล์ IFC ได้ ดังแสดงใน ผิดพลาด! ไม่พบแหล่งการอ้างอิง และทีมงานวิจัยได้ทำการศึกษาการใช้ FreeCAD สร้างชิ้นส่วนอาคารด้วยชิ้นส่วนของ FreeCAD แต่ในกรณีส่วนเหล็กเสริมงานโครงสร้างต้องพัฒนาส่วนนี้เพิ่มเติมดังแสดงตัวอย่างรหัสคำสั่งที่ 1 แล้วทำการส่งออกเป็นไฟล์ IFC ทำการตรวจสอบผลลัพธ์และเปรียบเทียบขนาดของไฟล์ได้ดังแสดงในตารางที่ 1 จะเห็นได้ว่าในกรณีที่เป็นชิ้นส่วนทรงสี่เหลี่ยมจะมีขนาดไฟล์ IFC ที่ไม่ใหญ่มาก แต่ในกรณีของชิ้นส่วนอาคารทรงกลมจะมีขนาดไฟล์ที่ใหญ่มากขึ้นเนื่องเป็นส่งออกข้อมูลแบบพื้นผิว



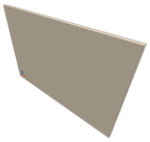
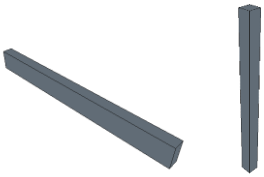
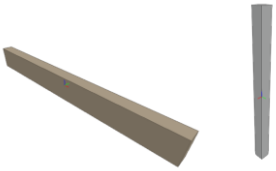
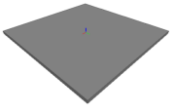
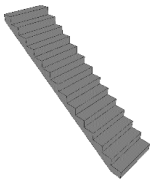
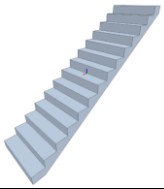
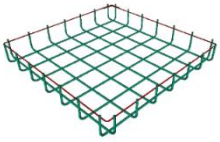
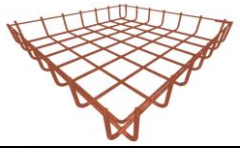
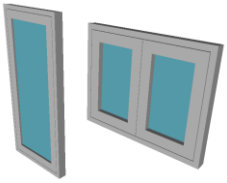
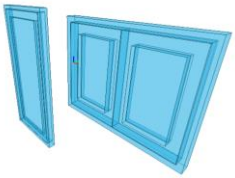
รูปที่ 11 : การส่งออกไฟล์ IFC จากโปรแกรม FreeCAD

รหัสคำสั่งที่ 1: Scripted Object ภาษา Python สร้างวัตถุชนิดใหม่ วัตถุเหล็กเสริมพื้นฐาน

```
"""
   ArchRebar2.py
"""

import ArchCommands,ArchComponent,FreeCAD
from FreeCAD import Vector , Placement
import Draft
import DraftGeomUtils , DraftVecUtils

from math import cos , radians

class _Rebar(ArchComponent.Component): # Base Class Object
    def __init__(self,obj):
        ArchComponent.Component.__init__(self,obj)

        obj.addProperty("App::PropertyDistance","diameter","Rebar","The diameter of this rebar")

        obj.Role = ['Rebar']
        self.Type = "Rebar"
        #obj.diameter = ['6 mm' , '9 mm' , '12 mm']

    def getProfile(self,obj,noplacement=True):
        return []
```

```
def getExtrusionVector(self,obj,noplacement=True):
    return FreeCAD.Vector()

def setShapeCode(self , ShapeCode):
    pass

def execute(self,obj):
    if self.clone(obj):
        return

    points = []
    lines = []

    for i in range(1, len(points)):
        lines.append(Part.Line(points[i],points[i-1]))

    spath = Part.Shape(lines)
    wpath  = Part.Wire(spath.Edges)
    wire = Part.Wire(wpath)
    wire = DraftGeomUtils.filletWire(wire,R)

    bpoint = points[0]
    bvec = points[1].sub(points[0]).normalize()
    circle = Part.makeCircle(dia/2,bpoint,bvec)
    circle = Part.Wire(circle)

    bar = wire.makePipeShell([circle],True,False,2)
    shapes = []
    shapes.append(bar)
    obj.Shape = bar
```

ตารางที่ 1 : การส่งออกข้อมูลจาก FreeCAD เป็น IFC

รายการ	แบบจำลองใน FreeCAD	ส่งออกไฟล์ IFC	ขนาดไฟล์
Wall			FreeCAD: 6 kB IFC: 3 kB
Beam Column			FreeCAD: 5 kB IFC: 3 kB
Slab			FreeCAD: 5 kB IFC: 3 kB
Stair			FreeCAD: 9 kB IFC: 17 kB
Rebar[1]			FreeCAD: 31 kB IFC: 4,616 kB
Door Window			FreeCAD: 20 kB IFC: 30 kB

5.1.2 การใช้ IfcOpenShell-Python ในการสร้างไฟล์ IFC

ทีมงานวิจัยได้ทำการศึกษาไฟล์ IFC และใช้งานคำสั่งมาโครภาษา Python โดยจะทำการทดสอบคำสั่งทีละส่วนแล้วเพิ่มอิลีเมนต์แต่ละชนิด ดังแสดงตัวอย่างในรหัสคำสั่งที่ 2 อธิบายขั้นตอนการทำงานได้ดังนี้ ทำการโหลดโมดูล IfcOpenShell แล้วสร้างตัวแปรจัดการไฟล์ หลังจากนั้นทำการสร้างชิ้นส่วน IfcProject ของโครงการทดสอบขั้นต้น และทำการบันทึกไฟล์ IFC ผลลัพธ์ของการรันคำสั่งมาโครจะได้ไฟล์ IFC โดยตรงแล้วทำการเปิดตรวจสอบความถูกต้องด้วยโปรแกรม IFC Viewer จะพบว่าสามารถสร้างไฟล์ IFC ได้ โดยจะเป็นสกีมารุ่น IFC2x3 และมีข้อมูลโครงการ #1 = IFCPROJECT() หนึ่งชิ้น พร้อมต้องกำหนดรหัสระบุจำเพาะ(Globally Unique Identifier) ดังแสดงตัวอย่างผลลัพธ์รหัสคำสั่ง IFC ดังแสดงในรหัสคำสั่งที่ 3

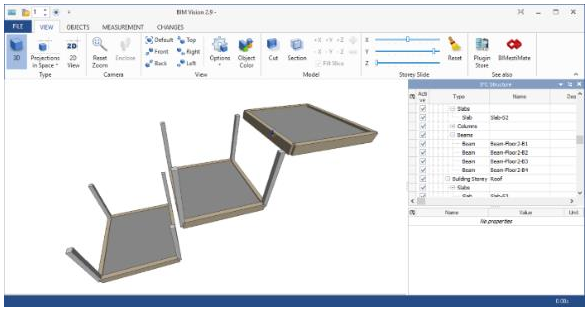
รหัสคำสั่งที่ 2: มาโครภาษา Python เรียกใช้โมดูล IfcOpenShell-Python ในการสร้างไฟล์ข้อมูล IFC

```
import sys, os
import ifcopenshell
ifc_file = ifcopenshell.file()
myPath = os.path.dirname(os.path.abspath(__file__)) + os.sep + 'OutPut'
myProject= ifc_file.createIfcProject(GlobalId=create_guid() ,
        Name='Test Project' , Description='Test Create IFC by ifcOpenShell')
myProject.Phase = 'Design'
ifc_file.write(myPath + os.sep+'Test01.ifc')
```

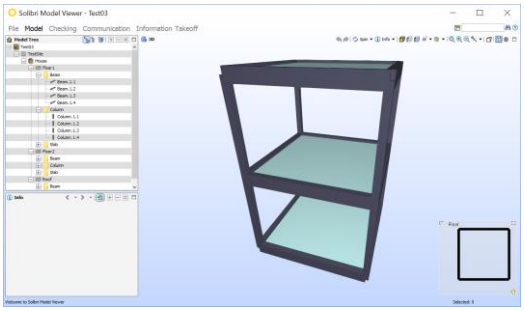
รหัสคำสั่งที่ 3: ไฟล์ข้อมูล IFC ที่ถูกสร้างขึ้นจาก IfcOpenShell-Python

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('ViewDefinition [CoordinationView]'),'2;1');
FILE_NAME(''Test01.ifc ','2016-06-19T20:11:24',(),(),'IfcOpenShell 0.5.0-dev',
        'IfcOpenShell 0.5.0-dev','');
FILE_SCHEMA(('IFC2X3'));
ENDSEC;
DATA;
#1=IFCPROJECT('1JDO00DXyHvhYmD1VhSTH7',#4,'Test Project',
    'Test Create IFC by ifcOpenShell',$,$,'Design',(#16,#15),#7);
ENDSEC;
END-ISO-10303-21;
```

ขั้นต่อมาได้ทำการสร้างข้อมูลส่วนต่างๆ เพิ่มเติมเช่น อาคาร(IfcBuilding), ชั้นของอาคาร (IfcBuildingStorey), ชิ้นส่วนเสา(IfcColumn), ชิ้นส่วนคาน(IfcBeam) และชิ้นส่วนแผ่นพื้น(IfcSlab) โดยกำหนดข้อมูลเป็นรูปทรงพื้นฐานเป็นทรงสี่เหลี่ยมในเบื้องต้น จากการค้นคว้าและพัฒนาทดสอบซ้ำจนสามารถเปิดไฟล์ IFC ที่ถูกต้องสามารถแสดงผลลัพธ์ได้ จะได้รหัสคำสั่งภาษา Python จำนวน 396 บรรทัดแต่ไม่สามารถนำเสนอบรรจุได้ในบทความนี้ เมื่อทำการสร้างไฟล์ IFC แล้วทำการเปิดตรวจทานข้อมูลภาพได้ผลดังแสดงในรูปที่ 12 จะเห็นได้ว่าสามารถสร้างแบบจำลองอาคารในระดับความละเอียดของการพัฒนาระดับหนึ่งได้ (LOD1) โดยไฟล์ที่ได้จะเป็นไฟล์ IFC ที่เป็นไฟล์กลางมาตรฐานใช้สำหรับการทำงานร่วมกัน(Interoperability) ซึ่งต้องทำการทดสอบต่อไปในการใช้งานร่วมกับ BIM Server

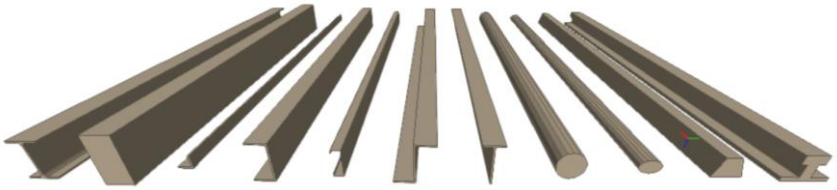


(ก) ไฟล์ IFC ที่เปิดตรวจสอบด้วย BIM Vision



(ข) ไฟล์ IFC ที่เปิดตรวจสอบด้วย Solibri Model Viewer

รูปที่ 12: ไฟล์ IFC ที่ถูกสร้างขึ้นด้วยคำสั่งที่พัฒนา และเปิดตรวจสอบด้วยโปรแกรม IFC Viewer



รูปที่ 13 : คำสั่งที่พัฒนาหน้าต่างรูปทรงต่างๆ ที่กำหนดไว้ใน IFC

ต่อมาได้พัฒนาคำสั่งทำการสร้างชิ้นคานเพื่อทำการเปลี่ยนแปลงหน้าตัด โดยใช้งานหน้าตัดกำหนดมาตรฐานของ IFC2x3 เช่น หน้าตัดตัวไอ(IfcIShapeProfileDef) หน้าตัดสี่เหลี่ยม(IfcRectangleProfileDef), หน้าตัดตัวยู(IfcUShapeProfileDef) เป็นต้น แล้วทำการสร้างชิ้นส่วนด้วย *IfcExtrudedAreaSolid* ดังแสดงตัวอย่างของคานรูปทรงหน้าตัดต่างๆ ใน
รูปที่ **13** ซึ่งจะทำให้จำนวนข้อมูลที่เกิดขึ้นน้อยกว่าการใช้งานแสดงผลแบบพื้นผิว

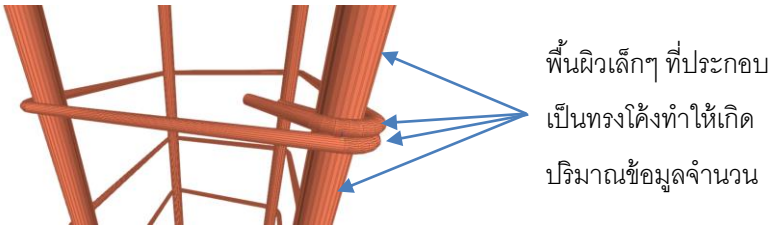
5.2. ปัญหาที่พบ

ก. การพัฒนาโปรแกรมในลักษณะ OpenSource ที่พัฒนาต่อยอด ยังขาดผู้ที่ให้ความสนใจในการพัฒนาและทำความเข้าใจ เนื่องจากการใช้ซอฟต์แวร์ละเมิดลิขสิทธิ์ยังเป็นแนวคิดที่ไม่สามารถเปลี่ยนกับนักวิจัยบางส่วน และส่งผลกระทบต่อนักศึกษาเมื่อจบการศึกษาแล้วยังคงใช้งานซอฟต์แวร์ละเมิดลิขสิทธิ์แล้วถูกตรวจพบก็จะเสียหายต่อตัวบัณฑิตและบริษัท จึงควรแนะนำการใช้ซอฟต์แวร์ให้เหมาะสมกับขนาดของโครงการ โครงการขนาดใหญ่มูลค่าสูงสามารถคิดเป็นต้นทุนดำเนินการ โครงการขนาดเล็กควรมีทางเลือกให้กับวิศวกร ซึ่งนักวิจัยควรพัฒนาโปรแกรม OpenSource ให้มีจำนวนมากขึ้นเพื่อเป็นทางเลือกในการใช้งานซอฟต์แวร์

ข. ในการพัฒนาคำสั่งภาษา Python และใช้ IfcOpenShell ในการสร้างไฟล์ IFC ปัญหาที่พบคือรูปแบบของโครงสร้างข้อมูล IFC ที่มีความซับซ้อน การสร้างข้อมูล IFC ที่ได้ไม่สามารถทดสอบความถูกต้องของข้อมูลได้ นอกจากการใช้วิธีเปิดไฟล์ตรวจสอบจากโปรแกรม IFC Viewer ซึ่งบางครั้งสามารถเปิดไฟล์ IFC ได้แต่แสดงผลแบบจำลองไม่เหมือนกันในรายละเอียดบางประการหรือตำแหน่ง ซึ่งจะต้องค้นคว้าหาส่วน IFC Model Checker แบบฟรีแวร์หรือแบบรหัสเปิดมาใช้ในการช่วยตรวจสอบต่อไป

ค. ในการส่งออกไฟล์ IFC ด้วยโปรแกรม FreeCAD จะทำการส่งออกด้วยข้อมูลแบบจำลองชนิด BREP(boundary representation) ทำให้ไฟล์ IFC มีขนาดใหญ่ โดยเฉพาะรูปทรงที่มีลักษณะผิวโค้งจะเกิดข้อมูลของผิว(Surface) จำนวนมากทำให้ไฟล์ที่ได้มีขนาดใหญ่ เช่น เหล็กเสริมในงานคอนกรีตดังแสดงใน

รูปที่ **14** ผู้วิจัยขอเสนอให้ทำการพัฒนาส่วนส่งออกข้อมูลขึ้นใหม่โดยใช้เทคนิค SweptAreaSolid จะทำให้ลดขนาดของข้อมูลได้



รูปที่ 14: ภาพแสดงพื้นผิวของทรงกระบอกในไฟล์ IFC ที่เกิดจากการส่งออกไฟล์แบบ BREP

6. สรุปผล

6.1. สรุปผล

จากการพัฒนาคำสั่งภาษา Python ที่ใช้งานโมดูล IfcOpenShell ส่วนเสริมของโปรแกรม FreeCAD สามารถสร้างชิ้นส่วนอาคาร IfcBeam, IfcSlab และ IfcColumn ในระดับขั้นต้นที่มีข้อมูลของแบบจำลองสามมิติ และสามารถนำไปเปิดใช้ด้วยโปรแกรม IFC Viewer ได้ ซึ่งสรุปได้สามารถ ข้อมูล IFC สามารถในการแลกเปลี่ยนข้อมูลอาคารสามารถทำงานร่วมกันได้ (Interoperablilty)

ณ ปัจจุบัน กลุ่มโปรแกรมสำหรับเปิดไฟล์ IFC ที่ไม่คิดมูลค่ายังอยู่ในระดับขั้นต้นคือในลักษณะ 3D BIM สามารถใช้เปิดดูแบบจำลองสามมิติ และข้อมูลขั้นต้น เช่น ปริมาตร พื้นที่ เป็นต้น ยังไม่เข้าสู่ลักษณะ 4D+5D BIM ที่จะต้องมีข้อมูลทางด้านเวลาและปริมาณวัสดุกับราคา

6.2. ข้อเสนอแนะ

ก. พัฒนาการสร้างข้อมูล IFC ของชิ้นส่วนของอาคารเพิ่มเติม เช่น IfcStair, IfcWall ที่มีข้อมูลวัสดุแบบแบ่งชั้น (Layered Material) , IfcWindow และ IfcDoor ที่เป็นข้อมูลของประเทศไทย สร้างชิ้นส่วนของเหล็กเสริมด้วย IfcReinforcingBar

ข. พัฒนาความสามารถของวัตถุใน FreeCAD ให้รองรับคุณสมบัติต่างๆของชิ้นส่วน IFC ให้มากขึ้น เพิ่มข้อมูลคุณสมบัติที่จะสามารถนำไปสู่การควบคุม 4D, 5D BIM เช่น ข้อมูลเวลาและการบริหารโครงการ และข้อมูลทางด้านราคาและข้อมูลผู้ผลิต

ค. สร้างกลุ่มผู้ใช้งาน และผู้พัฒนาเพื่อเพิ่มเครือข่ายในการพัฒนาโปรแกรม FreeCAD ตามแนวทางของกลุ่มโปรแกรมรหัสเปิด เพื่อใช้ได้มีโปรแกรมใช้งานทางด้าน BIM ที่ไม่ต้องใช้เงินจำนวนมากในการจัดซื้อ และสามารถนำมาพัฒนาต่อยอดเพิ่มเติมได้

7. กิตติกรรมประกาศ

ขอขอบคุณ มหาวิทยาลัยเอเชียอาคเนย์สำหรับสนับสนุนเวลา และสถานที่ ในการทำงานวิจัยนี้ ขอขอบคุณ นายภาณุทัตต์ แตงเจริญ, นายภูรินทร์ ธีระจรรยาภรณ์, นางสาวอังคณา สุขสมัย และนายอภิสิทธิ์ น้อยเจริญ นักศึกษาสาขาวิชาวิศวกรรมโยธา มหาวิทยาลัยเอเชียอาคเนย์ สำหรับการจัดทำข้อมูลอาคาร และคู่มือการใช้งาน FreeCAD

8. เอกสารอ้างอิง

[1] Wikipedia, the free encyclopedia, 2016, “Industry Foundation Classes”, https://en.wikipedia.org/wiki/Industry_Foundation_Classes/ [1 July 2016].

[2] Liebich, Thomas, 2007, “Industry Foundation Classes: IFC2x Edition 3 Technical Corrigendum 1”, www.buildingsmart-tech.org/ifc/IFC2x3/TC1/html/ [10 June 2016].

[3] Liebich, Thomas, 2009, “IFC 2x Edition 3 Model Implementation Guide”, buildingSMART International, www.buildingsmart-tech.org/downloads/accompanying-documents/guidelines/ [10 June 2016].

[4] Van Havre, Yorik, 2016, “FreeCAD and BIM FAQ”, http://yorik.uncreated.net/guestblog.php?tag=freecad/ [10 June 2016].

[5] FreeCAD, 2016, “FreeCAD: An Open Source parametric 3D CAD modeler," www.freecadweb.org/[10 June 2016].

[6] ภาณุทัตต์ แตงเจริญ และคณะ, 2558, “การศึกษาการใช้โปรแกรม FreeCAD สร้างไฟล์ IFC สำหรับการจำลองรายละเอียดข้อมูลอาคาร”, ปริญญานิพนธ์สาขาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเอเชียอาคเนย์.

[7] จักรี ติยะวงศ์สุวรรณ, 2559, “การพัฒนาโมดูลสำหรับวิเคราะห์โครงถักสองมิติในโปรแกรม FreeCAD”, การประชุมวิชาการระดับชาติสหวิทยาการเอเชียอาคเนย์ 2559 : งานวิจัยสีเขียว เพื่อการพัฒนาอย่างยั่งยืน, วันพฤหัสบดีที่ 23 มิถุนายน 2559 ณ โรงแรมริชมอนด์ สไตลิช คอนเวนชั่น.

[8] Krijnen, Thomas, 2016, “IfcOpenShell : the open source ifc toolkit and geometry engine”, http://ifcopenshell.org/python.html/[22 June 2016].

[9] Solibri inc., 2016, “Solibri Model Viewer (SMV)”, www.solibri.com/products/solibri-model-viewer/[4 July 2016].

[10] Datacomp., 2016, “BIM Vision - freeware IFC viewer”, www.bimvision.eu/ [4 July 2016].

[11] Gerold , Fabian, 2016, “IfcPlusPlus”, www.ifcplusplus.com/ [4 July 2016].

[12] Constructivity.com LLC., 2016, “Constructivity Model viewer”, www.constructivity.com/ [4 July 2016].