








Transformation of Internal CAD's 3D Coordinate System and IFC Files Coordination

Grebenyuk, Egor Alexandrovich^{1*} 
Vysotsky, Alexander Evgenievich² 
Novik, Anatoly Nikolaevich³ 
Ovchinnikov, Vladislav Vyacheslavovich⁴ 
Sidelnikov, Egor Romanovich⁵ 

^{1,2,3,4,5} Peter the Great St.Petersburg Polytechnic University, St.Petersburg, Russian Federation
Correspondence:* email inj5@yandex.ru contact phone [+79819856494](tel:+79819856494)

Keywords:

IFC; coordination; coordination model; BIM; CAD; coordinate transformation; Navisworks

Abstract:

Combining files in shared coordinates from software's internal coordinates is a widespread problem in BIM working processes because only some programs support both coordinate types. The mathematical algorithm was proposed to prevent manual operations and increase accuracy. The algorithm's practical implementation finds transformation parameters between CAD's internal 3D-coordinate system and shares coordinates using at least 3 points in both systems use a method of square leasts. The proposed algorithm is a particular case of general geodetic Helmert transformation used in a local area without some parameters (only dX, dY, dZ and ω_z are using). To check the accuracy of calculated parameters was entered a linear error that equals a returning value of square leasts's method for most optimum decision and comparing that linear error with acceptable value in CAD systems. By way of file converting example, a popular exchange data format in BIM projects as IFC files were considered. The program's implementation was executed in C# program language as a desktop application and class library, that will let change any IFC-files to adding the needing shared coordinates. All source code was published at GitHub https://github.com/GeorgGrebenyuk/IFC_GeoSupport.

1 Introduction

Современная практика автоматизированного проектирования зданий и сооружений подразумевает комплексный подход к вопросам моделирования объектов; при этом работа над моделями может выполняться в разных программных комплексах. Речь идет не только о непосредственном создании геометрии объектов и наполнении их параметрическими свойствами, но также о проверках всех моделей на коллизии и формировании общей 3D сцены с данными.

Существует ряд крупных программных решений в области автоматизированного проектирования отдельных объектов – это отраслевые САПР для гражданской и промышленной архитектуры, решения для инженерных систем, для работы с участками и генпланом, представленные как мультивендорами так и отдельными разработчиками зачастую в своих внутренних программных форматах хранения данных. Случай с необходимостью расчета параметров трансформации файлов с геометрией и свойствами моделей связан как раз с открытыми форматами представления данных, в которые выгружаются результат работы в данном ПО. Открытых общепринятых форматов не так много; один из них – это IFC (Industry Foundation Classes) [1,2]. В большей части, он применяется для описания геометрии зданий и их содержимого (трубопроводы, конструктивные элементы), но также используется как промежуточная ступень в преобразовании файлов в иные форматы данных [3-7]. Часть инфраструктуры пока не представлена классами IFC; для нее в качестве обходного пути действует механика выдавливания 3D тел по линейным объектам с наполнением их свойствами



от родительских объектов; тем не менее в рамках проводимых инфраструктурных BIM-проектов собирается опыт использования различных категорий объектов для будущей официальной спецификации IFC [8-9]. Данные генплана в виде проектных поверхностей как правило, выгружаются в формат LandXML.

Особняком стоит формат данных и, одновременно, спецификация для моделей в масштабе города – City GML. Вопросу потенциального взаимодействия элементов данной модели и IFC также посвящена статья [10]. Отдельно выделяют задачи по анализу внутренней структуры зданий в разрезе городской застройки, где в качестве данных используются файлы IFC [11-12].

Наряду с классической проектной документацией для отдельных категорий объектов теперь обязательно сдавать в экспертизу и трехмерные модели в формате IFC. Кроме того, данный формат является по большей части единственным способом импорта в отдельные отраслевые САПР для подгрузки результата работ из сторонних программ. Перейдем к проблеме глобальной координации файлов в проектных координатах.

Любая программа использует внутреннюю систему координат для расчета геометрии элементов и сохранения данной геометрии в обменный файл данных (здесь – IFC). К примеру, Autodesk Revit поддерживает возможность задания координат на уровне пользователя (формирование файла IFC в общих координатах); прочие САПР, если и имеют опцию определения геоположения, то ее параметры не экспортируются. Между тем есть возможность определить текущие координаты точек модели как из родительской САПР (где он формируется), так и в среде просмотра файла IFC (бесплатные решения, такие как BIM Vision, Solibri Model Viewer и др.). Конечные координаты данных точек могут быть определены из целевого генплана или непосредственно из САПР, куда данный IFC будет выгружен.

Несмотря на то, что в общем случае любой файл может быть вставлен путем ручного выравнивания в конечной модели (задания сдвижки, углов поворота и масштаба), такой подход не уместен для работы в единой BIM-экосистеме. Требуется оптимальный путь преобразования файла на программном уровне для использования результатов моделирования между различными САПР без их ручной обработки. Речь идет о необходимости менять параметры трансформации систем координат, относительно которых формируется геометрия всех элементов модели файла (здесь, IFC) по набору точек в исходной и целевой системах координат.

Тема преобразований координат не единожды освещалась в публикациях зарубежных ученых. Так, авторы в [13-16] рассматривают понятия внутренней системы координат модели и внешней общемировой, однако, если и касаются преобразований – то только лишь в плоской постановке задачи [14] для внутренних элементов модели. Основная цель рассмотренных статей – это настроить переход от данных IFC к ГИС данным, интегрируя результат моделирования с картографическими данными в Web-среде [16-19]. При этом в [19] прописывается общий вид матрицы трансформации координат, но далее тема остается нераскрытой. Публикации [20-21] и [22] освящают методику трансформации координат при помощи параметра `IfcCoordinateReferenceSystem`, появившегося в спецификации IFC 4 путем задания определения геодезических систем координат. В [23], автор акцентирует внимание на необходимости использования локальной (внутренней) системы координат для представления данных облака точек в формате IFC без пояснений как это рассчитать.

Таким образом, обозначим цель настоящей статьи как вывод алгоритма расчета параметров трансформации между двумя системами координат по набору общих точек. При этом будем пользоваться использовать индуктивный подход от общей геодезической трактовки данного вопроса к частному случаю постановки задачи. Дополнительно к алгоритму также приведем конкретные действия по изменению структуры IFC-файла.

2 Materials and Methods

2.1. Расчетный алгоритм

Задачи трансформации систем координат в пространстве являются в общем случае предметом изучения геодезии (вводится зависимость от формы земли, связь с проекциями, типом системы координат), но для текущего случая речь идет об изометрическом преобразовании (сочетание поворота и параллельного переноса). В силу междисциплинарности поднимаемого вопроса (трансформация систем координат для локальной территории (пространства модели)), методика решения будет опираться на общую формулу перехода от одной геодезической системы координат к другой, а алгоритм нахождения нужных параметров трансформации –



основываться на линейной процедуре с применением метода наименьших квадратов (далее - МНК).

Упомянутая выше общая формула выглядит следующим образом:

$$\begin{bmatrix} XN \\ YN \\ ZN \end{bmatrix} = M \cdot \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} XC \\ YC \\ ZC \end{bmatrix} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} \quad (1)$$

Где XN, YN, ZN – координаты в целевой СК (здесь и далее, СК – система координат); XC, YC, ZC – координаты точек в исходной СК; $\Delta X, \Delta Y, \Delta Z$ – параметры сдвига (параллельный перенос), a_{ij} ($i=1,2,3; j=1,2,3$) – элементы ортогональной матрицы поворота, которые являются нелинейными функциями вращения углов $\omega_x, \omega_y, \omega_z$ вращения вокруг соответствующих осей координат; M – увеличение масштаба.

В рассматриваемой трехмерной задаче для ограниченной зоны (области модели), параметр M равен единице. Принимая, что в качестве основной, в САПР используется правосторонняя декартова система прямоугольных координат с осью Z , направленной вверх, параметры поворота ω_x и ω_y принимаются равными нулю. Тогда целевая функция $f(N)$ для вычисления значений в целевой СК будет иметь общий вид:

$$f(N) = f(\Delta X, \Delta Y, \Delta Z, \omega_z) \quad (2)$$

Методика использования линейной процедуры для решения задачи встречается в общем случае геодезических координат [24]. Упрощая до нашей задачи, сведем процесс по вычислению параметров трансформации (членов формулы (2)) к следующим итеративным процессам:

- вычисление элементов матрицы поворота (для поиска значения ω_z);
- вычисление параметров параллельного переноса/сдвига ($\Delta X, \Delta Y, \Delta Z$).

Ниже приведена формула (22), используемая для МНК:

$$\begin{cases} Ma_{11} \cdot \Delta XC_{i,i+1} + Ma_{12} \cdot \Delta YC_{i,i+1} + Ma_{13} \cdot \Delta ZC_{i,i+1} - \Delta XN_{i,i+1} = v_{x_i}, \\ Ma_{21} \cdot \Delta XC_{i,i+1} + Ma_{22} \cdot \Delta YC_{i,i+1} + Ma_{23} \cdot \Delta ZC_{i,i+1} - \Delta YN_{i,i+1} = v_{y_i}, \\ Ma_{31} \cdot \Delta XC_{i,i+1} + Ma_{32} \cdot \Delta YC_{i,i+1} + Ma_{33} \cdot \Delta ZC_{i,i+1} - \Delta ZN_{i,i+1} = v_{z_i} \end{cases} \quad (3)$$

Используя в качестве исходных данных не координаты, а их разности, можно составить систему уравнений для определения только лишь элементов матрицы поворота – в нашем случае, для одной переменной ω_z по методу наименьших квадратов. Преобразуем формулу (3) в систему уравнений для нашего варианта задачи с учетом равенства нулю углов поворота ω_x и ω_y ($a_{13} = 0, a_{23} = 0, a_{31} = 0, a_{32} = 0, a_{33} = 1$), при этом в качестве разностей примем парные разности соответствующих координат для трех точек (1 и 2, 2 и 3, 1 и 3), не лежащих на одной прямой и лежащих в одной вертикальной плоскости (по высотным отметкам):

$$\begin{cases} a_{11} \cdot \Delta XC_{i,i+1} + a_{12} \cdot \Delta YC_{i,i+1} - \Delta XN_{i,i+1} = v_{x_i}, \\ a_{21} \cdot \Delta XC_{i,i+1} + a_{22} \cdot \Delta YC_{i,i+1} - \Delta YN_{i,i+1} = v_{y_i}, \\ 1 \cdot \Delta ZC_{i,i+1} - \Delta ZN_{i,i+1} = v_{z_i} \end{cases} \quad (4)$$

Где $i = 1,2,3$ – опытные точки. Третье уравнение системы (4) не рассматривается, потому что v_{z_i} обращается в ноль, так как точки лежат в одной вертикальной плоскости. Преобразуем данную систему уравнений к виду (5), заменив значения коэффициентов a :

$$\begin{cases} \cos(\omega_z) \cdot \Delta XC_{i,i+1} + \sin(\omega_z) \cdot \Delta YC_{i,i+1} - \Delta XN_{i,i+1} = v_{x_i}, \\ -\sin(\omega_z) \cdot \Delta XC_{i,i+1} + \cos(\omega_z) \cdot \Delta YC_{i,i+1} - \Delta YN_{i,i+1} = v_{y_i}, \end{cases} \quad (5)$$

Используемый в решении метод наименьших квадратов заключается в условии $\sum v^2 \rightarrow \min$, где

$$\sum v^2 = \sum_{i=0}^3 (v_{x_i}^2 + v_{y_i}^2)$$

Несмотря на наличие 4 переменных, использование 2 точек (одной разности) даст менее точный результат в отличие от трех проверяемых позиций. Рекомендация использования точек, лежащих в одной вертикальной плоскости, обоснована тем, что целевой генплан как правило представлен 2D-моделью с единственной абсолютной уровенной отметкой.

По окончанию перебора всего диапазона градусной меры угла (360°), будет найдено такое значение ω_z , при котором использование МНК приводит к минимальному значению. В реализации программного алгоритма можно выставить возвращаемое значение параметра $\sum v^2$ для оценки точности найденных значений в линейной мере. В практике САПР допустимым может считаться



значение менее 1 мм. Для случая 4 переменных в программном алгоритме на языке программирования С# был взят шаг прибавления для параметра ω_z равный 0.000001 радиан, что составляет 3141593 итерации для программного сценария. Использование такого шага давало значение параметра $\sum v^2$ порядка 10^{-7} м при продолжительности выполнения итеративного процесса порядка 5с.

По нахождения параметра ω_z формируется матрица поворота Π . В общем случае (для геодезических координат) перед матрицей есть параметр M , которым в локальной постановке задачи мы пренебрегаем.:

$$\Pi = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} \cos(\omega_z) & \sin(\omega_z) & 0 \\ -\sin(\omega_z) & \cos(\omega_z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Обозначенная матрица (6) используется для перехода к вспомогательным параметрам x_i , y_i , z_i ($i=1,2,3$) для вычисления параметров сдвижки - ΔX , ΔY , ΔZ :

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \Pi \cdot \begin{bmatrix} XC_i \\ YC_i \\ ZC_i \end{bmatrix} \quad (7)$$

Найденные вспомогательные параметры по формуле (7) нужны для вычисления матрицы L_Δ , которая в свою очередь наряду с матрицей H применяются для непосредственного вычисления ΔX , ΔY , ΔZ .

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; L_\Delta = \begin{bmatrix} XN_1 - x_1 \\ YN_1 - y_1 \\ ZN_1 - z_1 \\ XN_2 - x_2 \\ YN_2 - y_2 \\ ZN_2 - z_2 \\ XN_3 - x_3 \\ YN_3 - y_3 \\ ZN_3 - z_3 \end{bmatrix} \quad (8)$$

Теперь используя зависимости (8) можно перейти к расчету параметров для параллельного переноса (ΔX , ΔY , ΔZ) по формуле 9:

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} = (H^T \cdot H)^{-1} \cdot H^T \cdot L_\Delta \quad (9)$$

Программная реализация итеративного процесса по нахождению значения ω_z представлена в Приложении 1.

2.2. Действия с IFC

Теперь перейдем к обзору структуры файла IFC и параметров, задающих трансформацию системы координат. Спецификация IFC является открытыми сведениями, распространяемыми buildingSMART. Последней актуальной официальной версия формата IFC является версия 4.1.0.0. В настоящей статье была использована именно она (к слову, параметры геоположения и трансформации систем координат справедливы и для всех старших версий вплоть до 2x3).

В основном, через формат IFC передают 3D вид объектов, поэтому и рассмотрим далее такую структуру (к слову, методика действий при другом варианте будет аналогичная). Корневой элемент – `IfcProject`; он хранит в одном из атрибутов параметр `IfcGeometricRepresentationContext`, содержащий интересующую нас систему координат `WorldCoordinateSystem` и ориентацию на истинный север – `TrueNorth`. Каждый элемент геометрии отсчитывает своё положение от нуля внутренней системы координат, который представлен тегом `IfcLocalPlacement`. Корневой `IfcLocalPlacement` как правило содержит следующую структуру:

```
#19= IFCCARTESIANPOINT((0.0000000000000000,0.0000000000000000,0.0000000000000000));
#20= IFCDIRECTION((1.0000000000000000,0.0000000000000000,0.0000000000000000));
#21= IFCDIRECTION((0.0000000000000000,1.0000000000000000,0.0000000000000000));
#22= IFCAXIS2PLACEMENT3D(#19,#20,#21);
#23= IFCLocalPLACEMENT($,#22);
```



Здесь он принимает на вход два аргумента, первый из которых содержит значение null, а второй – параметр `IfcAxis2Placement3D`, задающий ориентацию и местоположение объекта в трехмерном пространстве. Подобных `IfcLocalPlacement` может быть 1-3 штуки (для каждой из декартовых осей вращения), и таким образом последний параметр `IfcLocalPlacement` определяет «площадку», ориентированную в пространстве, от которой разбивается геометрия всей модели. Правильнее всего менять именно корневой элемент, а остальные обнулять/удалять. Методика настоящей статьи – изменение данных параметров, а именно – базовой `IfcCartesianPoint`, хранящей параметры ΔX , ΔY , ΔZ и `IfcDirection`, задающих вектор поворота (функция угла ω_z). Также этот же `IfcDirection` будет задан как `TrueNorth` в `IfcGeometricRepresentationContext` (с версий IFC 4).

Обратим внимание, что менять необходимо именно второй параметр `IfcDirection`, так как первый следующий за `IfcCartesianPoint` определяет лишь направление текущей оси. Для оси Z – этот «первый» параметр будет выглядеть как (0,0,1) всегда, тогда как второй параметр, содержащий рассчитанное значение ω_z , будет записан как $(\cos(\omega_z), -\sin(\omega_z), 0)$.

Условившись, что в статье рассматривается частный случай направлений осей координат (правосторонняя декартова система и ориентация оси Z вверх), прочие параметры `IfcLocalPlacement` для осей O_x, O_y нам менять не потребуется, но при необходимости (задав целью выгружать геометрию, к примеру, в Unity 3D где используется левосторонняя система координат) есть возможность поменять соответствующие параметры.

Подытожим потенциальные изменения в структуре IFC-файла, которые необходимо будет сделать для внесения информации о трансформации системы координат и представим данное описание в виде последовательного измененного кода (см. Приложение 2, 3)

Параметр `IFCSITE` не является геометрическим элементом структуры файла; он лишь позволяет задать «участок», расположенный в определенных координатах. Теперь каждый элемент геометрии модели будет ссылаться на площадку `IfcLocalPlacement (#33)`, которой задано верное пространственное положение относительно целевого файла.

3 Results and Discussion

Изложенный в настоящей статье алгоритм определения параметров трансформации между двумя декартовыми системами координат при математической постановке задачи является универсальным и может применяться не только в случае изменения структуры IFC-файла (и в целом, не только для IFC файлов, в принципе).

Ввиду существования разных алгоритмов формирования структуры IFC файла различными САПР, представленный алгоритм не претендует на универсальность применения в части замены параметров, но позволяет точно рассчитать параметры трансформации. Его работоспособность проверена на сценарии загрузки IFC из Autodesk Revit, Renga, Aveva Everything 3D и Tekla в среду Autodesk Navisworks.

Необходимо также сделать ремарку, что в последних спецификациях формата появился параметр `IfcCoordinateReferenceSystem`, позволяющий задавать описание геодезической системы координат. Тем не менее важно определить грань между точной математической трансформацией данных в пространстве и геодезическими преобразованиями, зависящими от формы земли и учитывающие различные системы координат, как и численные подходы для пересчета из одной системы в другую.

В результате применения рекомендаций данной статьи, можно менять параметры в IFC ранних спецификаций, которые в отличие от IFC 4 не поддерживают функции прямого задания системы координат и реализовывать процесс смены также программным путем.

Представленный алгоритм был реализован как самостоятельное десктопное приложение с открытым исходным кодом, доступное по ссылке (см. Приложение 4).

В общем случае подобные кейсы с программной координацией должны затрагиваться при обучении специалистов по информационному моделированию, подобный опыт описывается в [25], где на базе университета в городе Maribor была развернута программа обучения для студентов гражданского строительства с применением специализированной ГИС-платформы.



Аналогичная программа подготовки развернута на базе Санкт-Петербургского политехнического университета, для неё характерно большое число пересекающихся учебных дисциплин, что подробно описано в [26].

4 Conclusions

Был реализован итеративный алгоритм вычисления параметров трансформации между внутренней системой координат файла IFC и общими координатами, используемыми в проекте для возможности автоматической сборки данных в единой среде без дополнительных преобразований. Уделено внимание ранним спецификациям формата IFC – то есть универсальность изменений структуры файла вне зависимости от его спецификации (версии). Дополнительно к математическому описанию приведена программная реализация метода наименьших квадратов как основного оптимизационного сценария для поиска угла поворота в плоскости XOY для расчета параметров трансформации. Исходный код приложения выложен в открытый доступ на платформе GitHub, как и сведения об изменении файлов IFC от прочих вендоров, не представленных в Приложении к настоящей статье. [4]

References

1. Li, K.J., Zlatanova, S., Torres-Sospedra, J., Perez-Navarro, A., Laoudias, C., Moreira, A. Survey on indoor map standards and formats. 2019 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2019. 2019. DOI:10.1109/IPIN.2019.8911796.
2. Lu, Q., Chen, L., Li, S., Pitt, M. Semi-automatic geometric digital twinning for existing buildings based on images and CAD drawings. *Automation in Construction*. 2020. 115. DOI:10.1016/j.autcon.2020.103183.
3. González, E., Piñeiro, J.D., Toledo, J., Amay, R., Acosta, L. An approach based on the ifcOWL ontology to support indoor navigation. *Egyptian Informatics Journal*. 2021. 22(1). Pp. 1–13. DOI:10.1016/j.eij.2020.02.008.
4. Yu, X., Ergan, S. BIM coverage in demand response management: A pilot study in campus buildings. *Construction Research Congress 2018: Construction Information Technology - Selected Papers from the Construction Research Congress 2018*. 2018. 2018-April. Pp. 316–325. DOI:10.1061/9780784481264.031.
5. Valinejadshoubi, M., Bagchi, A., Moselhi, O., Shakibabarough, A. Investigation on the potential of building information modeling in structural health monitoring of buildings. *CSCE General Conference 2018, Held as Part of the Canadian Society for Civil Engineering Annual Conference 2018*. 2018. Pp. 407–416.
6. Ma, G., Jiang, J., Shang, S. Visualization of Component Status Information of Prefabricated Concrete Building Based on Building Information Modeling and Radio Frequency Identification: A Case Study in China. *Advances in Civil Engineering*. 2019. 2019. DOI:10.1155/2019/6870507.
7. Chen, W., Chen, K., Cheng, J.C.P., Wang, Q., Gan, V.J.L. BIM-based framework for automatic scheduling of facility maintenance work orders. *Automation in Construction*. 2018. 91. Pp. 15–30. DOI:10.1016/j.autcon.2018.03.007.
8. Exinger, C., Mülitzer, G., Felsner, R., Lemmerer, J., Matt, R., Griesser, E. BIM pilot project Granitztal tunnel chain – Development of data structures for tunnel structure and track superstructure. *Geomechanik und Tunnelbau*. 2018. 11(4). Pp. 348–356. DOI:10.1002/geot.201800022.
9. Providakis, S., Rogers, C.D.F., Chapman, D.N. Predictions of settlement risk induced by tunnelling using BIM and 3D visualization tools. *Tunnelling and Underground Space Technology*. 2019. 92. DOI:10.1016/j.tust.2019.103049.
10. Floros, G.S., Ellul, C., Dimopoulou, E. Investigating interoperability capabilities between IFC and CityGML LoD 4 - Retaining semantic information. *International Archives of the Photogrammetry*,



Remote Sensing and Spatial Information Sciences - ISPRS Archives. 2018. 42(4/W10). Pp. 33–40. DOI:10.5194/isprs-archives-XLII-4-W10-33-2018.

11. Lin, W.Y. Automatic generation of high-accuracy stair paths for straight, spiral, and winder stairs using IFC-based models. ISPRS International Journal of Geo-Information. 2020. 9(4). DOI:10.3390/ijgi9040215.

12. Lin, W.Y., Lin, P.H. Intelligent generation of indoor topology (i-GIT) for human indoor pathfinding based on IFC models and 3D GIS technology. Automation in Construction. 2018. 94. Pp. 340–359. DOI:10.1016/j.autcon.2018.07.016.

13. Funtík, T., Gašparík, J. Site plan development: Tower crane placement based on data obtained from IFC file. ISARC 2016 - 33rd International Symposium on Automation and Robotics in Construction. 2016. Pp. 36–41.

14. Shi, J., Dao, J., Jiang, L., Pan, Z. Research on IFC- And FDS-Based Information Sharing for Building Fire Safety Analysis. Advances in Civil Engineering. 2019. 2019. DOI:10.1155/2019/3604369.

15. Kenley, R., Yeo, R., Harfield, T. Three location referencing method families: A new theoretical framework. IOP Conference Series: Materials Science and Engineering. 2019. 512(1). DOI:10.1088/1757-899X/512/1/012059.

16. Zhu, J., Tan, Y., Wang, X., Wu, P. BIM/GIS integration for web GIS-based bridge management. Annals of GIS. 2020. DOI:10.1080/19475683.2020.1743355.

17. O'Donovan, J., O'Sullivan, D., McGlinn, K. A method for converting IFC geometric data into GeoSPARQL. CEUR Workshop Proceedings. 2019. 2389. Pp. 7–20.

18. Xu, Z., Zhang, L., Suo, H., Chi, Y.Z. IFC-based data visualization of 3D Tiles for buildings. Zhejiang Daxue Xuebao (Gongxue Ban)/Journal of Zhejiang University (Engineering Science). 2019. 53(6). Pp. 1047–1056. DOI:10.3785/j.issn.1008-973X.2019.06.004.

19. Sani, M.J., Rahman, A.A. GIS and BIM integration at data level: A review. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives. 2018. 42(4/W9). Pp. 299–306. DOI:10.5194/isprs-archives-XLII-4-W9-299-2018.

20. Ugglá, G., Horemuz, M. Georeferencing Methods for IFC. Proceedings - 2018 Baltic Geodetic Congress, BGC-Geomatics 2018. 2018. Pp. 207–211. DOI:10.1109/BGC-Geomatics.2018.00045.

21. Ugglá, G., Horemuz, M. Geographic capabilities and limitations of Industry Foundation Classes. Automation in Construction. 2018. 96. Pp. 554–566. DOI:10.1016/j.autcon.2018.10.014.

22. Jaud, Š., Donaubauer, A., Bormann, A. Georeferencing within IFC: A Novel Approach for Infrastructure Objects. Computing in Civil Engineering 2019: Visualization, Information Modeling, and Simulation - Selected Papers from the ASCE International Conference on Computing in Civil Engineering 2019. 2019. Pp. 377–384. DOI:10.1061/9780784482421.048.

23. Krijnen, T., Beetz, J. An IFC schema extension and binary serialization format to efficiently integrate point cloud data into building models. Advanced Engineering Informatics. 2017. 33. Pp. 473–490. DOI:10.1016/j.aei.2017.03.008.

24. Ioannidou, S., Pantazis, G. Helmert transformation problem. From euler angles method to quaternion algebra. ISPRS International Journal of Geo-Information. 2020. 9(9). DOI:10.3390/ijgi9090494.

25. Kovačič, B., Kamnik, R., Pustovgar, A., Vatin, N. Landscape Interventions and Appreciations with Program Software RO. Procedia Engineering. 2016. 165. Pp. 918–925. DOI:10.1016/j.proeng.2016.11.801.

26. Vatin, N.I., Usanova, K.Y. BIM end-to-end training: From school to graduate school. Advances and Trends in Engineering Sciences and Technologies III- Proceedings of the 3rd International Conference on Engineering Sciences and Technologies, ESaT 2018. 2019. Pp. 651–656. DOI:10.1201/9780429021596-102.

Grebenyuk, E.; Vysotsky A.; Novik A.; Ovchinnikov, V.; Sidelnikov, E. Transformation of internal CAD's 3D coordinate system and IFC files coordination; 2021; *AlfaBuild*; 17 Article No 1703. doi: 10.34910/ALF.17.3

