

Вестник Евразийской науки / The Eurasian Scientific Journal <https://esj.today>

2018, №6, Том 10 / 2018, No 6, Vol 10 <https://esj.today/issue-6-2018.html>

URL статьи: <https://esj.today/PDF/94SAVN618.pdf>

Статья поступила в редакцию 28.11.2018; опубликована 25.01.2019

Ссылка для цитирования этой статьи:

Лосев К.Ю. Подход к информационной поддержке среды общих проектных данных в жизненном цикле объекта капитального строительства // Вестник Евразийской науки, 2018 №6, <https://esj.today/PDF/94SAVN618.pdf> (доступ свободный). Загл. с экрана. Яз. рус., англ.

For citation:

Lossev K.Yu. (2018). The approach to CDE design information support within a construction object life cycle. *The Eurasian Scientific Journal*, [online] 6(10). Available at: <https://esj.today/PDF/94SAVN618.pdf> (in Russian)

УДК 72

ББК 38.2

ГРНТИ 67.23.15

Лосев Константин Юрьевич

ФГБОУ ВО «Национальный исследовательский Московский государственный строительный университет»
Москва, Россия

Доцент кафедры «Информационные системы и технологии автоматизации строительства»

Кандидат технических наук

E-mail: LossevKY@mgsu.ru; c.lossev@gmail.com

Подход к информационной поддержке среды общих проектных данных в жизненном цикле объекта капитального строительства

Аннотация. Современные отечественные технологии организации строительства, рассматривая объекты строительства в контексте их жизненного цикла, должны обеспечивать наличие среды общих данных для управления информацией об объекте строительства в составе информационной модели. В данной статье рассматриваются возможность организации такой среды на примере "безшовной" интеграции информационных моделей различных систем автоматизированного проектирования, обобщения их в одно целое, не используя посреднические нейтральные форматы.

Рассмотрены основные подходы к созданию и поддержанию среды общих данных в процессе архитектурно-строительного проектирования. В статье дан обзор объективных неудобств при использовании форматов-посредников в процесс проектирования и интеграции информационных моделей объектов капитального строительства.

Предметом рассмотрения являются два варианта метода "безшовной" интеграции информационных моделей, названные "интеграция in situ" и "web-интеграция". Оба варианта интеграции основаны на активном использовании служебных приложений в лицензионных пакетах систем автоматизированного проектирования. Автором рассмотрена схема реализации вариантов метода, его преимущества и недостатки в каждом из вариантов. Основными преимуществами являются качество и скорость конвертации моделей в сравнении с методами, использующими нейтральные форматы-посредники. Также важным преимуществом является двунаправленность конвертации и возможность реализовывать варианты интеграций между различными системами автоматизированного проектирования на одном Интернет-сайте,

причем каждая следующая разработка будет дешевле предыдущих за счет наработок последних.

Автором предлагается гипотеза о том, что метод прямой интеграции на основе подхода "web-интеграция" будет востребован в производственных кластерах, где он даст организациям, входящим в кластер, ряд конкурентных преимуществ.

Ключевые слова: среда общих данных; СОД; системы автоматизированного проектирования; САПР; информационная модель; объект капитального строительства; API; интеграция информационных систем; производственные кластеры; конвертирование моделей

Введение

Современные отечественные технологии организации строительства обязательно должны рассматривать объекты строительства в контексте своего жизненного цикла (ЖЦ), о чем свидетельствует ст.3, п.2 Технического регламента о безопасности зданий и сооружений ФЗ-№384. Одним из условий информационной поддержки ЖЦ объектов строительства и, в частности, объектов капитального строительства (ОКС) является организация среды общих данных (СДО) или, как ее именуют за рубежом, Common Data Environment (CDE). ГОСТ Р 57311-2016 определяет, что управление информацией об ОКС в составе информационной модели должно осуществляться с использованием СОД, которая является единым источником информации об ОКС и может включать в себя серверное оборудование, каналы связи, файловые системы поиска и другие программно-технические средства [1].

В контексте предметной области ОКС одной из основ СОД являются конструктивные элементы здания и дополнительная информация о них. Раньше, до внедрения безбумажных технологий проектирования конструктивные элементы содержались в бумажной проектной документации. Сегодняшнее состояние дел предполагает одновременное существование и бумажной и цифровой (информационной, виртуальной, компьютерной) документации ОКС, причем бумажная документация является производной от цифровой, а та, в свою очередь, является производной от информационной модели ОКС, создаваемой средствами систем информационного моделирования, традиционно обобщенно называемыми САПР или в международной терминологии CAD, CAE, CAM (Computer Aided Design / Engineering / Manufacturing) [2].

Данные средства разработки информационных моделей ОКС имеют различных производителей, собственные внутренние методы хранения данных, различные графические ядра и различные коммерческие форматы данных для представления файлов информационных моделей ОКС [3].

По существу, инженерные данные, создаваемые САПР, в большинстве случаев имеют проприетарные, то есть закрытые коммерческие форматы, специфичные для их собственных информационных систем, и не являются общими в прямом смысле этого слова. Поэтому для создания и поддержания СОД в процессе архитектурно-строительного проектирования существует три подхода с использованием:

1. единого всеобщего для предметной области OKS формата представления данных;
2. "посредника" – нейтрального открытого формата данных;
3. API (application programming interface) – набора способов прямой передачи данных между различными САПР.

Первый подход в обозримой перспективе пока затруднителен в ситуации мировой рыночной экономики.

Второй подход в настоящее время активно развивается в международной строительной сфере на примере текстовых форматов данных XML, IFC, IDM, BCF и т. д. Примером данного подхода является международный обмен данными на основе пяти открытых форматов представления архитектурно-строительных данных OPEN BIM [4].

Существуют объективные неудобства при использовании форматов-посредников типа ifc. Во-первых, экспорт в формат ifc больших информационных моделей почти всегда не обеспечивает 100 % идентичности такой модели при открытии в другой САПР, поскольку в каждом САПР имеются структурные элементы модели ("семейства" в Revit, "стили" в Renga, "gdl-объекты" в Archicad и т. д.), которые для своего экспорта в ifc нуждаются в ручной настройке [5].

Во-вторых, модуль конвертации из проприетарного формата в ifc и обратно пишется разработчиком программного обеспечения конкретного САПР для своей системы, и степень проработанности и использования особенностей формата ifc в таких модулях различна. Это влияет на СОД даже в случае малых информационных моделей. Опыт экспериментов автора статьи показывает, что, например, в САПР Tekla 2017 (Service Pack 3 build 8792) создается простейшая информационная модель здания из пяти стен, двускатной крыши, двух оконных и двух дверных проемов, перекрытия над фундаментом и ленточного фундамента. Модель экспортируется в ifc-файл инструментами Tekla. Модель корректно импортируется в САПР Revit (ver.18.0.0.420). Но попытке импорта модели в САПР Renga (ver.2.11.14361.53646x64), теряется внутренняя стена с дверным проемом верхней подрезкой по форме двускатной крыши. В свою очередь, если эту модель Tekla, импортированную программой Revit, сохранить в проприетарном формате Revit (.rvt) и экспортировать в формат ifc из Revit, то импортирование такой ifc-модели в Renga происходит без потери указанной стены. Это происходит во многом по тому, что язык формата IFC достаточно обширен структурно, например, версия 2×4 RC4 2012d состоит из: 126 определяемых типов, 206 перечисляемых типов, 59 типов по выбору, 764 определений сущностей, 43 функций, 408 групп свойств, 91 группы величин, 1691 индивидуального свойства. В дополнение к этому сложность работы с форматом увеличивается из-за возможности смоделировать один объект несколькими формами моделирования, например, один структурный блок может быть смоделирован и с помощью представления, ограниченного четырьмя планами(проекциями), и через сжатие поверхности и вектора. Эти две альтернативные формы IFC-представления одного и того же объекта имеют различные семантические значения в итоговой информационной модели, и, хотя они могут иметь одинаковый вид в объемном изображении на экране монитора, в структурном методе анализа модели и при конвертировании они будут рассматриваться по-разному.

В-третьих, модели IFC, как правило, имеют большие размеры файлов. Например, полная информационная модель здания с 19-ю панелями составляет около 360 Мб (Общий знаменатель. IFC – это намного больше, чем простой формат файла. URL: <https://bimlib.ru/articles/obshchiy-znamenatel-ifc---eto-namnogo-bolshe-chem-prostoy-format-fayla-17/>).

К дополнительным трудностям можно отнести и то, что модель в IFC формате относительно долго загружается любой САПР, например, информационная модель с детализацией LOD 350 (level of detalization) стандартного 5-6 этажного дома будет загружаться около 6 минут на типовом университетском компьютере.

В итоге проблема формата IFC, на наш взгляд состоит в том, что он, прекрасно поддерживая типовые структурные архитектурно-конструкторские объекты в декартовой системе координат, начинает давать некорректные результаты при переносе более или менее индивидуальных проектов с нестандартной архитектурой [7]. Исчезают части конструкции крыши, не производится подрезка элементов в нужных местах или не создаются окна нужного формата и пр. Количество ошибок, подобных указанным существенно вырастает по мере усложнения архитектурно-строительного проекта.

Третий подход также активно развивается в связи с тенденцией предложений разработчиками программного обеспечения дополнительных инструментов расширения функционала своих продуктов. Такие инструменты разработчик сосредотачивает преимущественно в SDK (Software Development Kit) – наборах утилит, исходных кодов, различных API (Application Programming Interface), которые размещаются в свободном доступе на информационных ресурсах разработчиков. Использование API – комплектов методов, процедур, классов и т. п., принадлежащих конкретным САПР – позволяет реализовать "безшовную" интеграцию между различными САПР, то есть избегая "швов", в виде представления информационной модели в формате-посреднике.

При организации среды общих данных между проектными организациями в каждой из них, чаще всего установлена одна или две лицензионных САПР, а не целый набор, что обуславливается рациональностью и экономическим расчетом, поэтому данный подход рассматривается нами как наиболее перспективный к информационной поддержке среды общих проектных данных в жизненном цикле объекта капитального строительства.

Метод

Для обеспечения информационной поддержки среды общих проектных данных в жизненном цикле объекта капитального строительства рассматривается метод использования API при переводе конкретной информационной модели ОКС из одной САПР в другую.

В качестве объектов исследования были взяты две САПР, ориентированные на информационное моделирование зданий: Revit (компания Autodesk, США) и Renga (компания АСКОН, РФ) [8]. Первая САПР была выбрана по причине ее полнофункциональности (архитектура, конструкции, инженерные сети) и фактического известности среди отечественных проектировщиков; вторая САПР была выбрана, поскольку является одной из первых полнофункциональных динамично развивающихся отечественных САПР с полноценным 3D-графическим ядром [9].

Файл, представляющий собой информационную модель здания или, другими словами, архитектурно-строительный проект в Revit имеет расширение .rvt, в Renga – .rnp. Необходимо отметить, что оба файла являются по сути своей zip-архивами, то есть наборами данных (папок и файлов), сжатыми в популярном формате архивации файлов и сжатия данных ZIP.

Итого, первичная структура данных обеих САПР является архивом и выглядит после разархивирования следующим образом:

Имя	Дата изменения	Тип
Formats	24.09.2018 5:51	Папка с файлами
Global	24.09.2018 5:51	Папка с файлами
Partitions	24.09.2018 5:51	Папка с файлами
BasicFileInfo	24.09.2018 5:51	Файл
Contents	24.09.2018 5:51	Файл
ProjectInformation	24.09.2018 5:51	Файл
RevitPreview4.0	24.09.2018 5:51	Файл "0"
TransmissionData	24.09.2018 5:51	Файл

Рисунок 1. Первичная структура данных информационной модели здания формата .RVT, САПР Revit (сделано автором, Лосев К.Ю., 2018)

Name	Size	Packed	Type
..			Папка с файлами
assemblies			Папка с файлами
c3d			Папка с файлами
sheets			Папка с файлами
specs			Папка с файлами
tables			Папка с файлами
preview.png	95 253	90 722	PNG Image
assoc_dim_data.xml	317	192	Документ XML
filters.xml	4 121	889	Документ XML
levelViewSettings.xml	317	193	Документ XML
project.xml	2 899 716	138 328	Документ XML
styles.xml	4 108 372	132 003	Документ XML
user_attrs.xml	19 719	1 379	Документ XML
checksum	32	34	Файл
metadata	59	53	Файл
relationGraph.graph	304	190	Файл "GRAPH"

Рисунок 2. Первичная структура данных информационной модели здания формата .RNP, САПР Renga (сделано автором, Лосев К.Ю., 2018)

Анализ показывает, что Revit хранит содержание информационной модели здания – в зашифрованном бинарном виде, Renga – в незашифрованном открытом xml формате.

В связи с этим, рассматриваемый метод распадается на два предлагаемые ниже варианта, которые имеют свои положительные и отрицательные стороны.

Первый вариант, названный "интеграция insitu", заключается в том, что создается приложение "plug-in" (независимо компилируемый программный модуль, динамически подключаемый к Revit) с помощью Revit API.

Из всех 4 возможных вариантов использования API при "безшовной" интеграции информационных моделей между САПР Revit и Renga, возможен только вариант перенесения модели из Renga в Revit с использованием Revit API [10].

Задачей данного приложения, которое для пользователя выглядит как кнопка в графическом интерфейсе Revit (в закладках "Управление" или "Настройки"), является конвертация и открытие в Revit файлов в .gnr форматах из САПР Renga.

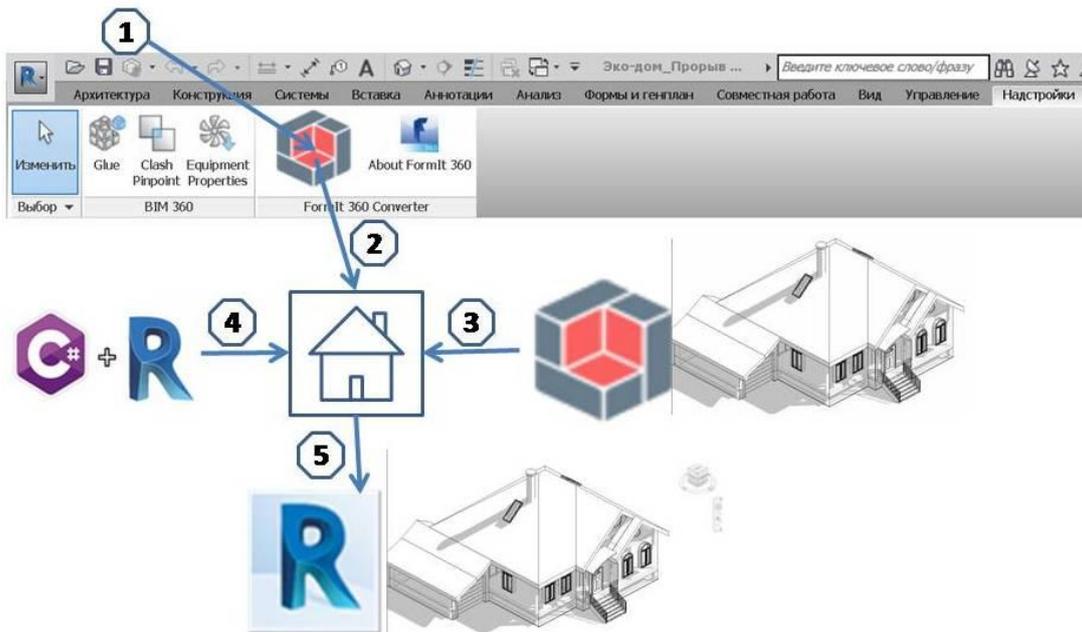


Рисунок 3. Схема интеграции информационной модели между САПР Renga и Revit через создание plug-in для Revit (сделано автором, Лосев К.Ю., 2018)

Схема интеграции информационной модели в первом варианте представлена на рис. 3 и состоит из следующих шагов:

1. Пользователь активирует кнопку в графическом интерфейсе Revit.
2. Кнопка вызывает приложение plug-in, динамически подключенное к Revit.
3. Plug-in считывает открывает zip-архив информационной модели Renga и по алгоритму считывает данные модели в текстовом формате xml.
4. Plug-in обращается к API Revit и конвертирует данные информационной модели Revit.
5. Plug-in направляет данные полученной модели на визуализацию пользователю в графический интерфейс Revit.

Напомним, что .gnr-файлы являются текстовыми данными в формате xml, заархивированными zip-архивом. Данное приложение plug-in реализуется кодом на языке C#:

Преимуществом данного варианта является:

- скорость конвертации информационной модели минимум в 3 раза быстрее времени конвертации через формат IFC, к примеру время конвертации 5-ти этажного одноподъездного дома: 1,5 мин. против 5 мин.;
- низкая себестоимость разработки для проектной организации;
- отсутствие требований к специфическим знаниям для пользователя, удобство использования (приложение plug-in, в отличие от IFC, подсветит все неверно распознанные и конвертированные объекты);
- возможность доработки под нужды организации и доработки в процессе обновления информационных продуктов.

Недостатками данного варианта является:

- односторонняя интеграция моделей, обратная конвертация информационной модели (из Revit в Renga) силами данного приложения невозможна, так как в файле формата .rnr нужно решить проблему с checksum ("контрольной суммой", защищающей исходный файл от нештатных изменений);
- при обновлении версий каждого из двух САПР данное приложение необходимо будет изменять и дорабатывать в соответствии с изменениями версий, что будет являться небольшими, но регулярными расходами для организации.

Анализ данного метода с целью избежать вышеописанных недостатков привел к разработке второго варианта интеграции двух САПР, названного "web-интеграция".

Пользователь через сайт загружает на сервер файл информационной модели здания в формате .rnr (Renga). Файл проходит первичную проверку и попадает в файловую систему сервера. Модуль управления проверяет папку хранения данных, и, если там есть в наличии файл, направляет его в программу Renga, открывая файл модели, запуская приложение "plug-in Renga" и ожидая получения от него ответа с данными. Запущенное "plug-in Renga" выбирает все объекты в открытом файле, считывает их геометрию и семантические характеристики и отправляет их модулю управления. Загруженный пользователем файл модели Renga удаляется и/или архивируется.

Модуль управления перебрасывает данные, полученные приложением "plug-in Renga", приложению "plug-in Revit" и запускает конвертирование информационной модели. Приложение "plug-in Revit", работая в связке с программой Revit, сохраняет новый файл модели и дает знать о завершении работы модулю управления. Новый файл информационной модели в формате .rvt (Revit) кэшируется и/или архивируется. Модуль управления выдает на сайт сообщение пользователю, которое содержит ссылку на скачивание информационной модели в формате .rvt. Далее модуль ожидает загрузки нового файла информационной модели для конвертации.

Схема интеграции информационной модели во втором варианте представлена на рис. 4 и состоит из следующих шагов:

1. Пользователь заходит на сайт проектной организации.
2. Пользователь выбирает файл для конвертации и делает запрос (нажимает на кнопку) для конвертации информационной модели из САПР Renga в САПР Revit. Происходит обращение к управляющему модулю на сервере.
3. Управляющий модуль открывает указанный пользователем файл информационной модели в Renga.
4. Renga предоставляет доступ к своим API функциям.
5. Приложение "plug-in Renga" с помощью API Renga выбирает все элементы информационной модели в открытом файле Renga.
6. Управляющий модуль на сервере с помощью API Renga получает данные модели от "plug-in Renga".
7. Управляющий модуль на сервере перенаправляет полученные от Renga данные в приложение "plug-in Revit".
8. Revit предоставляет доступ к API функциям Revit.
9. "plug-in Revit" с помощью API Revit открывает данные информационной модели в Revit.

10. "plug-in Revit" с помощью API Revit сохраняет модель в виде файла Revit на сервере.
11. Управляющий модуль на сервере перенаправляет ссылку на сайт.
12. Пользователь получает ссылку на скачивание файла информационной модели для САПР Revit.

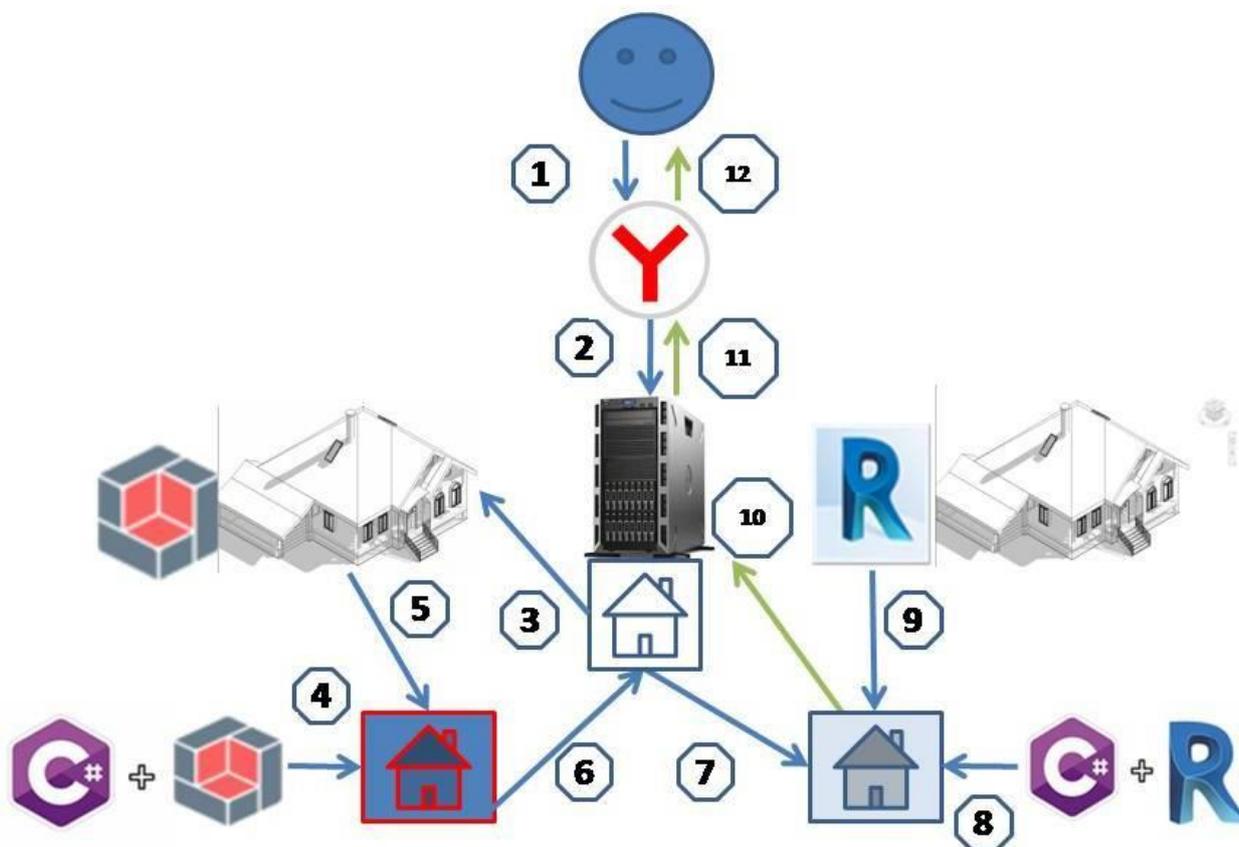


Рисунок 4. Схема интеграции информационной модели между САПР Renga и Revit через Web-клиента и создание 2 приложений "plug-in" для Revit и Renga (сделано автором, Лосев К.Ю., 2018)

Схема интеграции информационной модели между из САПР Revit в САПР Renga зеркальна схеме на рис. 4.

Преимуществом данного варианта является:

- двусторонняя интеграция моделей, управляющий модуль определяет формат модели и выбирает соответствующую ветку алгоритма;
- независимость от обновлений версий каждого из двух САПР – нововведения быстро реализуются через API;
- удобство интерфейса для пользователя и постоянная доступность в сети Интернет с любого устройства;
- возможность доработки под нужды организации. Возможность реализовывать варианты интеграций между различными САПР на одном сайте, причем каждая следующая разработка будет дешевле предыдущих за счет наработок последних.

Недостатками данного варианта является:

- постоянная поддержка программного обеспечения на сервере, самого сервера, регулярные доработки;
- скорость интеграции информационных моделей временами может быть низкой из-за колебаний пропускной способности канала связи и очередей из файлов при большом количестве запросов;
- более высокая стоимость реализации варианта "web-интеграция" в сравнении с вариантом "интеграция in situ".

На основе вышеизложенного предлагается гипотеза о том, что прямая "безшовная" интеграция САПР на основе подхода "web-интеграция" будет востребована в производственных кластерах, что даст:

- независимость проектировщиков-создателей информационных моделей ОКС от методологических ограничений нейтральных форматов данных типа IFC;
- увеличение скорости и качества интеграции информационных моделей;
- появление строительных элементов ОКС, включаемых проектировщиками в информационные модели, которые неотъемлемо и предметно связаны с условиями ЖЦ ОКС – строительства, эксплуатации, демонтажа;
- стимулирование развития проектирования объектно-ориентированных серий ОКС, то есть строительство на основе информационной модели, являющейся проектной серией.

Выводы

1. Организация СОД для управление информацией об ОКС в составе информационной модели без использования открытых нейтральных форматов, направленная на прямую "безшовную" интеграцию моделей, требует от проектной организации финансовых затрат и регулярной поддержки программного обеспечения, которое обеспечивает данную интеграцию.
2. Из двух, рассмотренных вариантов метода "безшовной" интеграцией информационных моделей предпочтительным является вариант "web-интеграции", расширяющий возможности двунаправленной конвертации моделей из различных САПР и повышающий качество интеграции.
3. Подходы, связанные с "безшовной" интеграцией информационных моделей при рациональной организации работ, укрепят позиции отечественного производителя САПР в строительстве на российском рынке за счет создания производственных кластеров, заинтересованных именно в прямой "безшовной" интеграции в своей внутрикластерной СОД.
4. Выдвинута гипотеза, нуждающаяся в дальнейших исследованиях, что метод прямой "безшовной" интеграции САПР на основе подхода "web-интеграция" будет востребован в производственных кластерах, где он даст организациям, входящим в кластер, ряд конкурентных преимуществ.

ЛИТЕРАТУРА

1. Лосев К.Ю. Создание и внедрение технологии управления жизненным циклом объектов строительства // Промышленное и гражданское строительство. 2014. №11, С. 67-70.
2. Абрамов А. Как эффективно управлять проектной организацией с помощью Lement Pro [Электронный ресурс] <http://www.ecm-portal.ru/practice/expertise/kak-effektivno-upravlyat-proektnoy-organizatsiey-s/> (дата обращения 20.04.2018).
3. Гришко О. Управление проектной организацией – пять задач для Pilot-ICE // САПР и Графика. 2017. №2, С. 2-4.
4. Роеф Р. OPEN BIM как способ взаимодействия [Электронный ресурс] URL: <https://archi.ru/world/81780/rob-roef-open-bim-edinstvennyi-put/> (дата обращения 01.12.2018).
5. Eastman Ch. The Future of IFC: Rationale and Design of a SEM IFC Layer, Materials from IDDS Workshop, April 2012.
6. Saygi G., Agugiaro G., Намамсцюğlu-Turan M., Remondino F. Evaluation of GIS and BIM roles for the information management of historical buildings, Materials from XXIV International CIPA Symposium, 02-06.09., Strasbourg, France, 2013, Volume II-5/W1, 283-288.
7. Режепов С. Renga vs Revit. Выбираем BIM-инструмент на примере возможностей армирования // САПР и Графика. 2018. № 8, С. 16-20.
8. Дьячева И. Системы проектирования от компании Renga Software // САПР и Графика. 2017. № 10, С. 34-37.
9. Чекалин В. Доступ к Revit из внешнего приложения. [Электронный ресурс] URL: <http://adn-cis.org/dostup-k-revit-iz-vneshnego-prilozheniya.html> (дата обращения 01.12.2018).
10. Chun D.-M., Kim H.-J., Lee J-Ch., Ahn S.-H. Web-Based Material Database for Material Selection and its Application Programming Interface (API) for CAD // Key Engineering Materials Vols. 345-346, 2007, pp. 1593-1596.
11. Paviot T. Web ready. [Электронный ресурс] URL: http://www.pythonocc.org/features_overview/experimental-webgl-renderer-towards-cad-in-a-browser/ (дата обращения 01.12.2018).

Losev Konstantin Yur'evich

Moscow state university of civil engineering national research university, Moscow, Russia
E-mail: LossevKY@mgsu.ru; c.lossev@gmail.com

The approach to CDE design information support within a construction object life cycle

Abstract. When considering construction objects in the context of its life cycle, the modern domestic technologies of construction management should provide common data environment. It is needed for the information management about a construction object as a part of the information model. This article discusses the possibility of organizing such an environment on the example of "seamless" integration of information model of various computer-aided design systems, generalizing them into the whole, without using intermediary neutral formats. The main approaches to the creation and maintenance of the common data environment in the process of architectural and construction design are considered. The article provides an overview of the objective inconveniences in the use of intermediary formats within the process of design and integration of construction object information models. The subject of consideration are two variants of the method of such integration, called "in situ integration" and "web-integration". Both integration variants are based on the active use of vendor's licensed packages API of computer-aided design systems. The author considers the scheme of method variants implementation, its advantages and disadvantages in each of the variant. The main advantages are the quality and speed of model conversion compared to methods using neutral intermediary formats. Another important advantage is the direct bidirectional conversion and the ability to implement integration options between different computer-aided design systems on the same website, and each subsequent development will be cheaper than the previous ones due to developments of the latter. The realization scheme of the method variants, its advantages and disadvantages in each of the variants is considered. The author proposes a hypothesis that the method of direct integration based on the "web-integration" approach will be in demand by production clusters, where it will give organizations within the cluster a number of competitive advantages.

Keywords: common data environment; CDE; computer aided design; CAD; information model; object of construction after completion; API; integration of information systems; industry clusters; model conversion