

Докладчик

Образование:

Петрозаводский Государственный университет

2014 – Системы автоматизированного проектирования;

Карьера:

2012 – Промпроект, ПГС. *Конструктор*

2014 – Градпроект (Петрозаводск),
Жилищное строительство.

Конструктор

2015 – **ВЕРФАУ**, Проектирование и строительство медицинских учреждений.

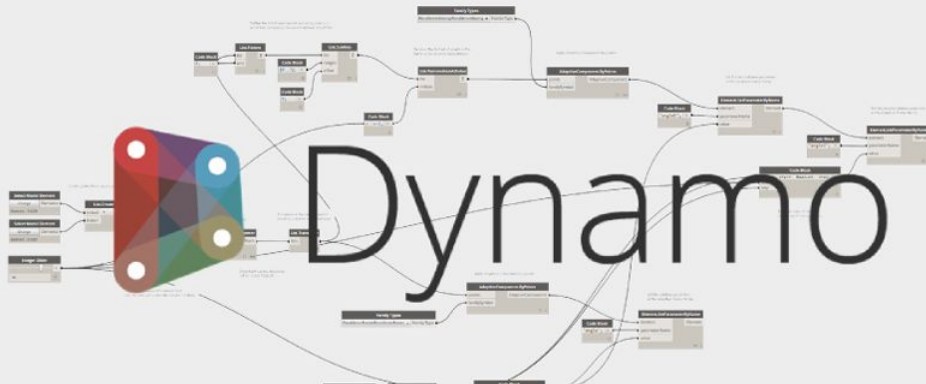
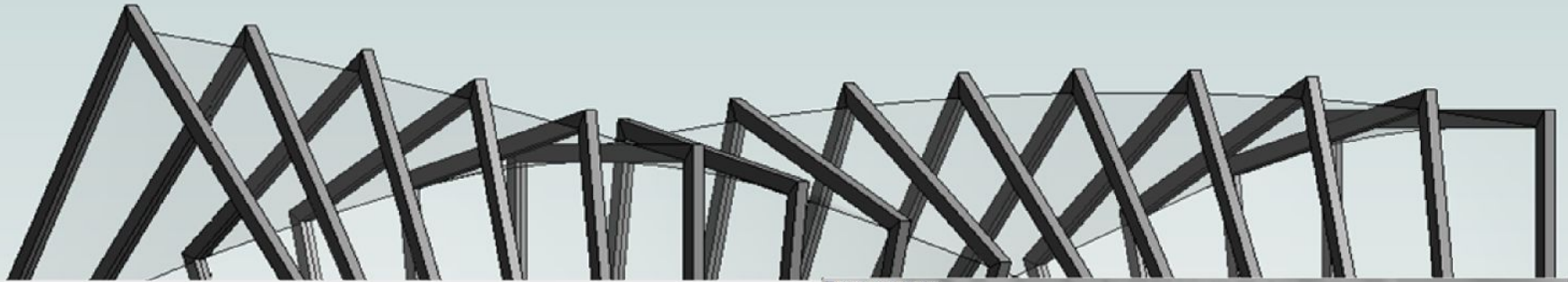
ВМ-координатор, ВМ-менеджер

2016 – **АЕСОМ**, Проектирование, управление строительством, логистика, инжиниринговые услуги.

ВМ-менеджер

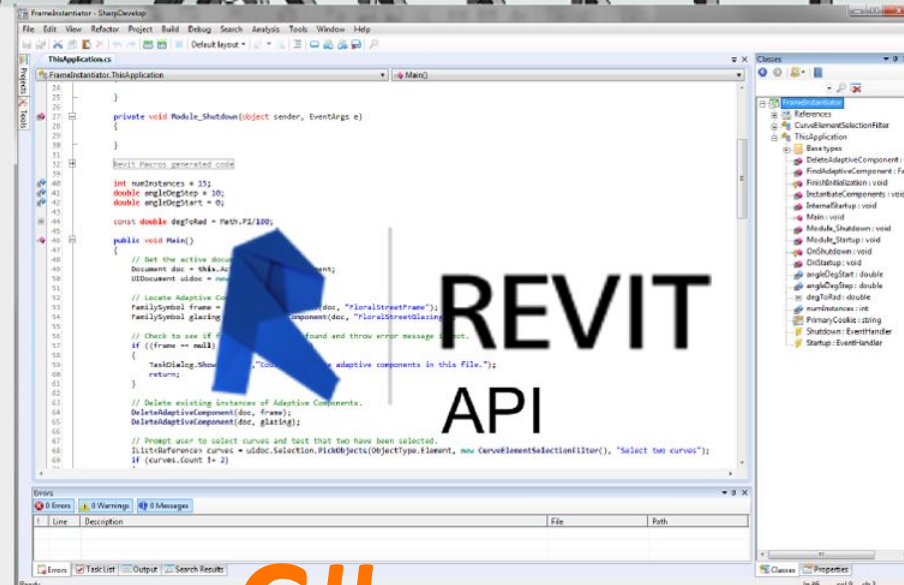


Чем мы можем дорабатывать Revit?



DesignScript

Python C#



REVIT
API

Что такое Дунато?

Как вы представляете себе

программирование?

```
@var boolean
define('PSI_INTERNAL_XML', false);
if (version_compare("5.2", PHP_VERSION, ">")) {
    die("PHP 5.2 or greater is required!!!");
}
if (extension_loaded("pcre")) {
    die("phpSysInfo requires the pcre extension to php in order to work properly.");
}

require_once APP_ROOT.'/includes/autoloader.inc.php';

// Load configuration
require_once APP_ROOT.'/config.php';

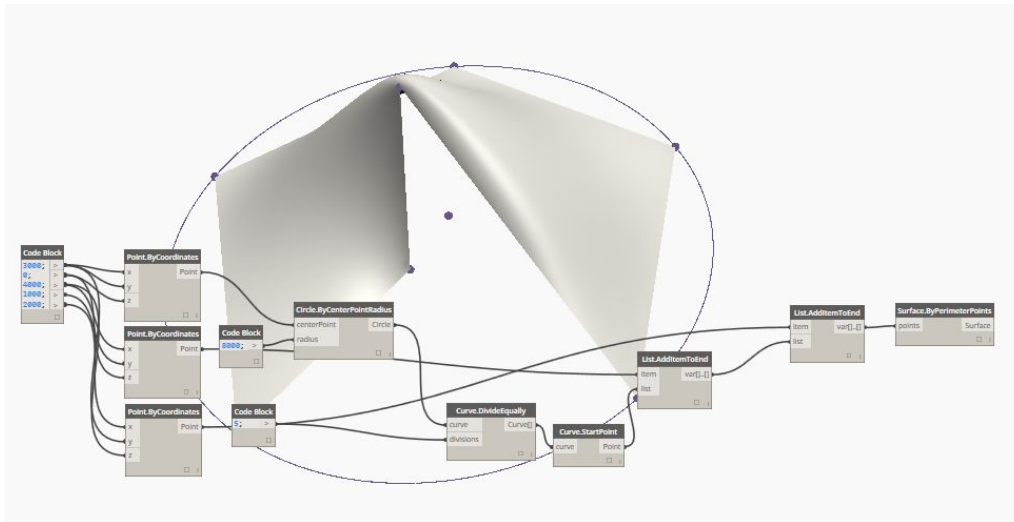
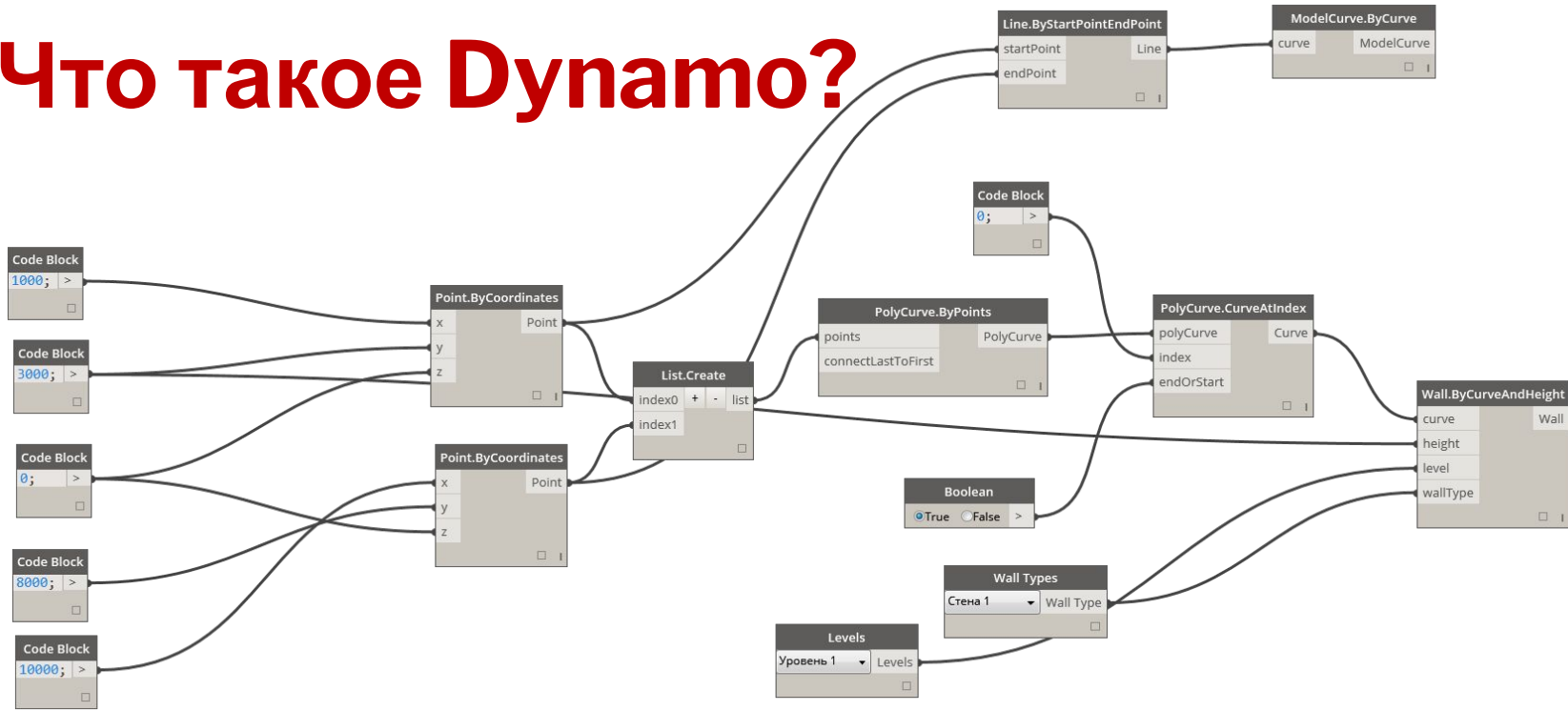
if (!defined('PSI_CONFIG_FILE') || !defined('PSI_DEBUG')) {
    $tpl = new Template("/templates/html/error_config.html");
    echo $tpl->fetch();
    die();
}
```



```
Редактирование сценария Python...
14 from RevitServices.Transactions import TransactionManager
15 from System.Collections.Generic import *
16 # Import RevitAPI
17 clr.AddReference("RevitAPI")
18 import Autodesk
19 from Autodesk.Revit.DB import *
20
21 doc = DocumentManager.Instance.CurrentDBDocument
22
23 rooms = IN[0]
24 levels = IN[1]
25
26 i=0
27 count = levels.Count
28 roomlist=[]
29 # Start Transaction
30 TransactionManager.Instance.EnsureInTransaction(doc)
31 while i<count:
32     point = UnwrapElement(rooms[i]).Location.Point
33     level=UnwrapElement(levels[i])
34     uv=UV(point.X,point.Y)
35     a=doc.Create.NewSpace(level,uv)
36     roomlist.append(a)
37     i=i+1
38 # End Transaction
39 TransactionManager.Instance.TransactionTaskDone()
40 OUT = roomlist
```

принять изменения Отмена

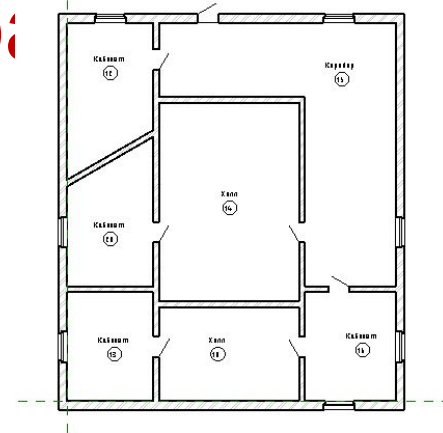
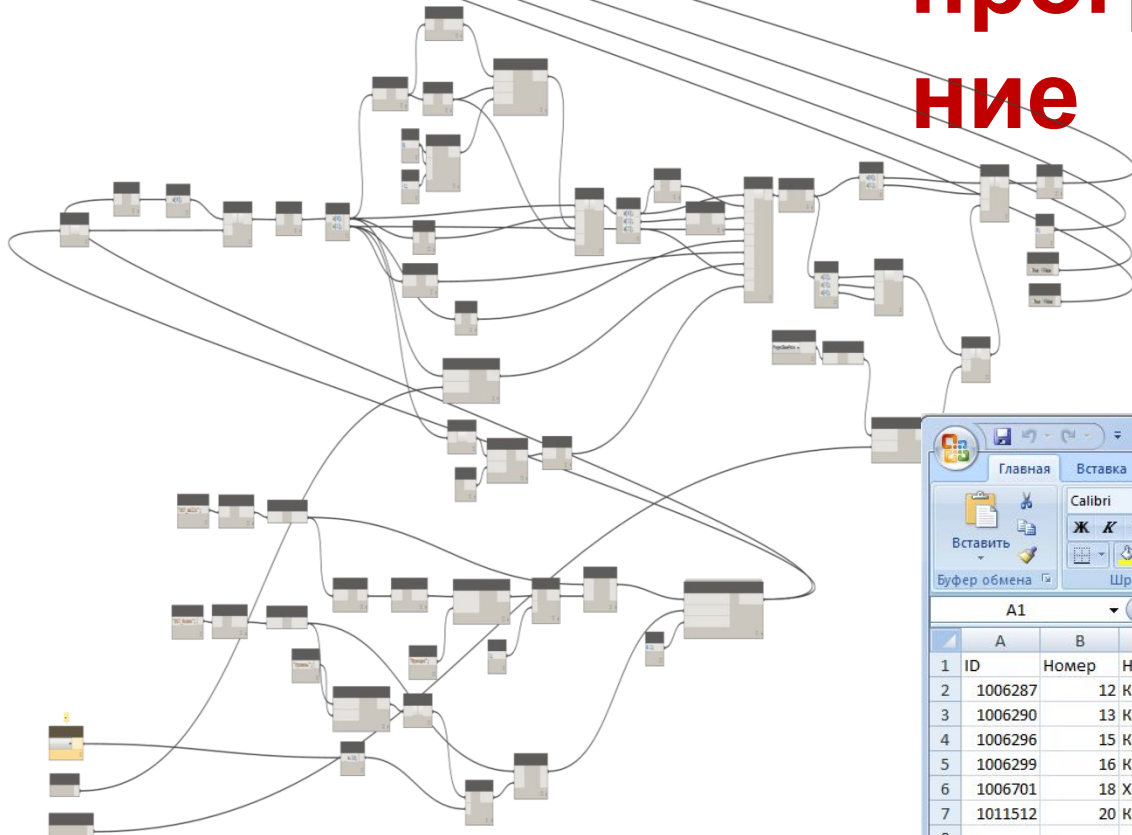
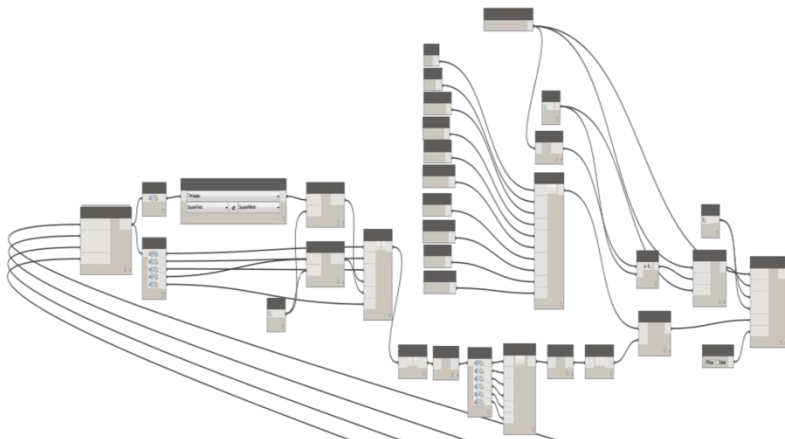
Что такое Дунато?



```
Редактирование сценария Python...
14 from RevitServices.Transactions import TransactionManager
15 from System.Collections.Generic import *
16 # Import RevitAPI
17 clr.AddReference("RevitAPI")
18 import Autodesk
19 from Autodesk.Revit.DB import *
20
21 doc = DocumentManager.Instance.CurrentDBDocument
22
23 rooms = IN[0]
24 levels = IN[1]
25
26 i=0
27 count = levels.Count
28 roomlist=[]
29 # Start Transaction
30 TransactionManager.Instance.EnsureInTransaction(doc)
31 while i<count:
32     point = UnwrapElement(rooms[i]).Location.Point
33     level=UnwrapElement(levels[i])
34     uv=UV(point.X,point.Y)
35     a=doc.Create.NewSpace(level,uv)
36     roomlist.append(a)
37     i=i+1
38 # End Transaction
39 TransactionManager.Instance.TransactionTaskDone()
40 OUT = roomlist
```

принять изменения Отмена

Что такое Дунато? Визуальное программирование

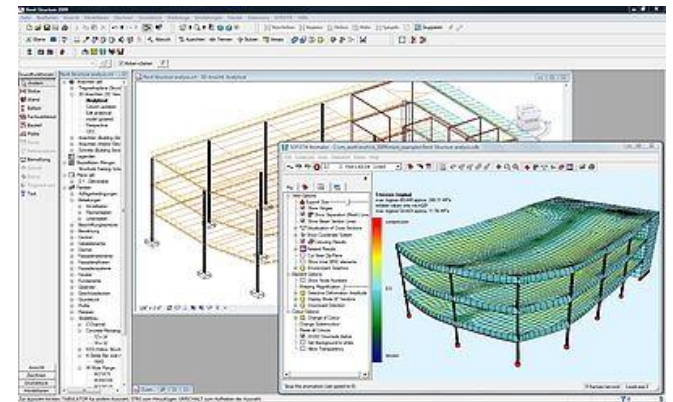


Площади наружных стен.xlsx - Microsoft Excel

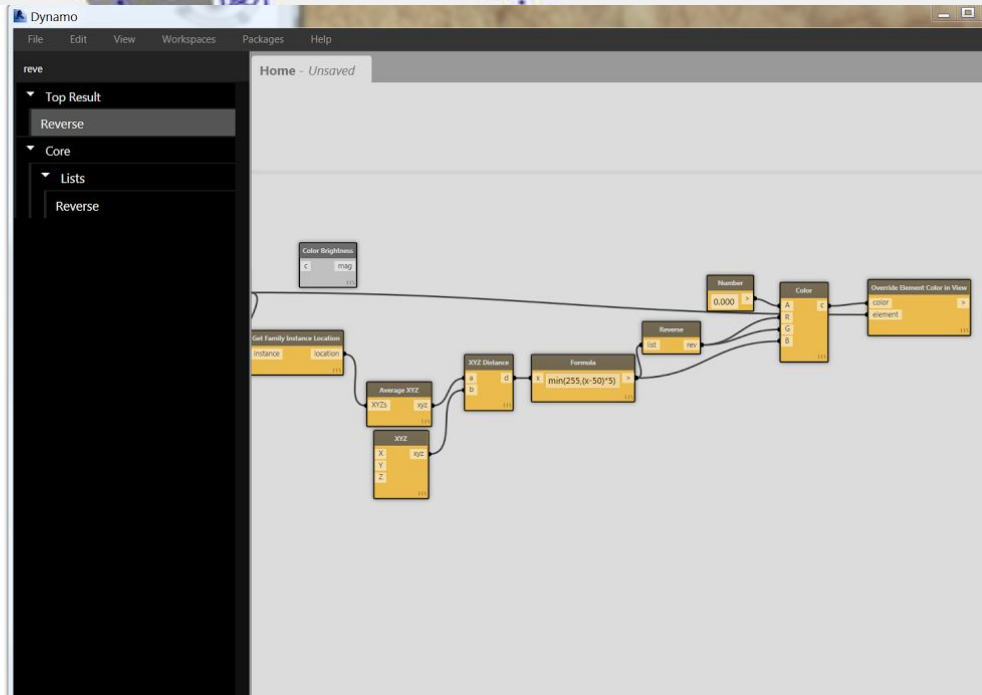
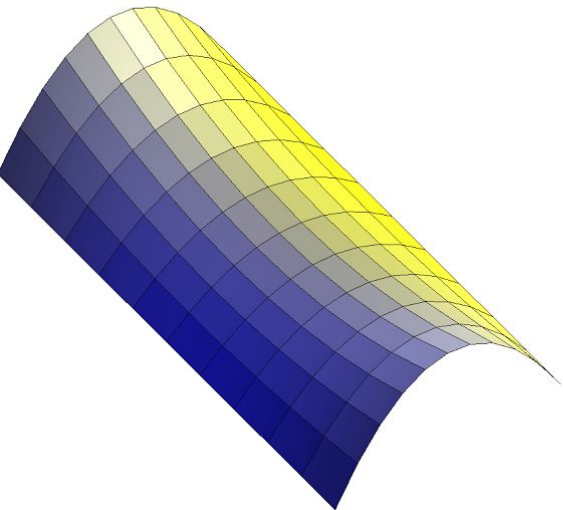
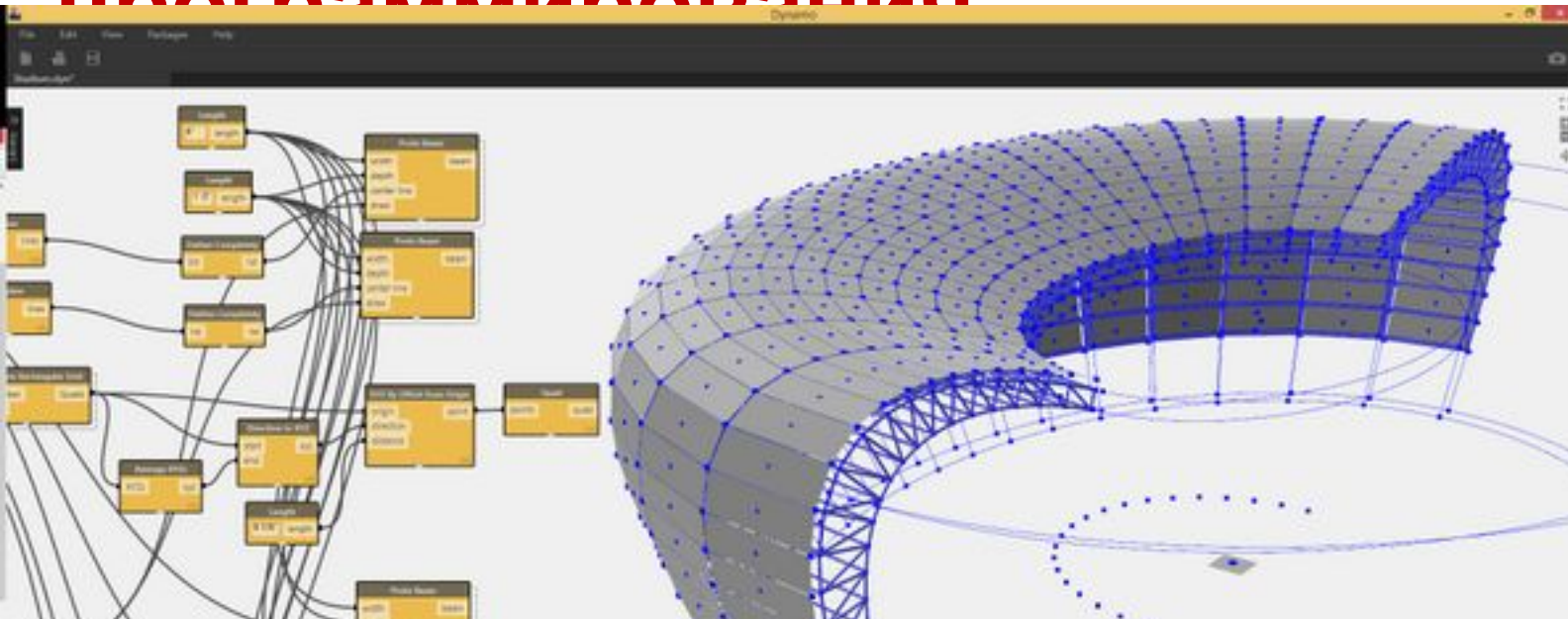
ID	Номер	Наименование	Площадь	Площадь	Ориентация	Площадь	Ориентация	Площадь	Ориентация	НСЗ
1006287	12	Кабинет	21,94	20,68	З	11,33	С			
1006290	13	Кабинет	18,08	11,32	Ю	14,36	З			
1006296	15	Коридор	67,19	31,18	С	34,75	В			
1006299	16	Кабинет	19,16	12	Ю	14,35	В			
1006701	18	Холл	25,44	18,44	Ю					
1011512	20	Кабинет	20,75	13,2	З					

Задачи для визуального программирования

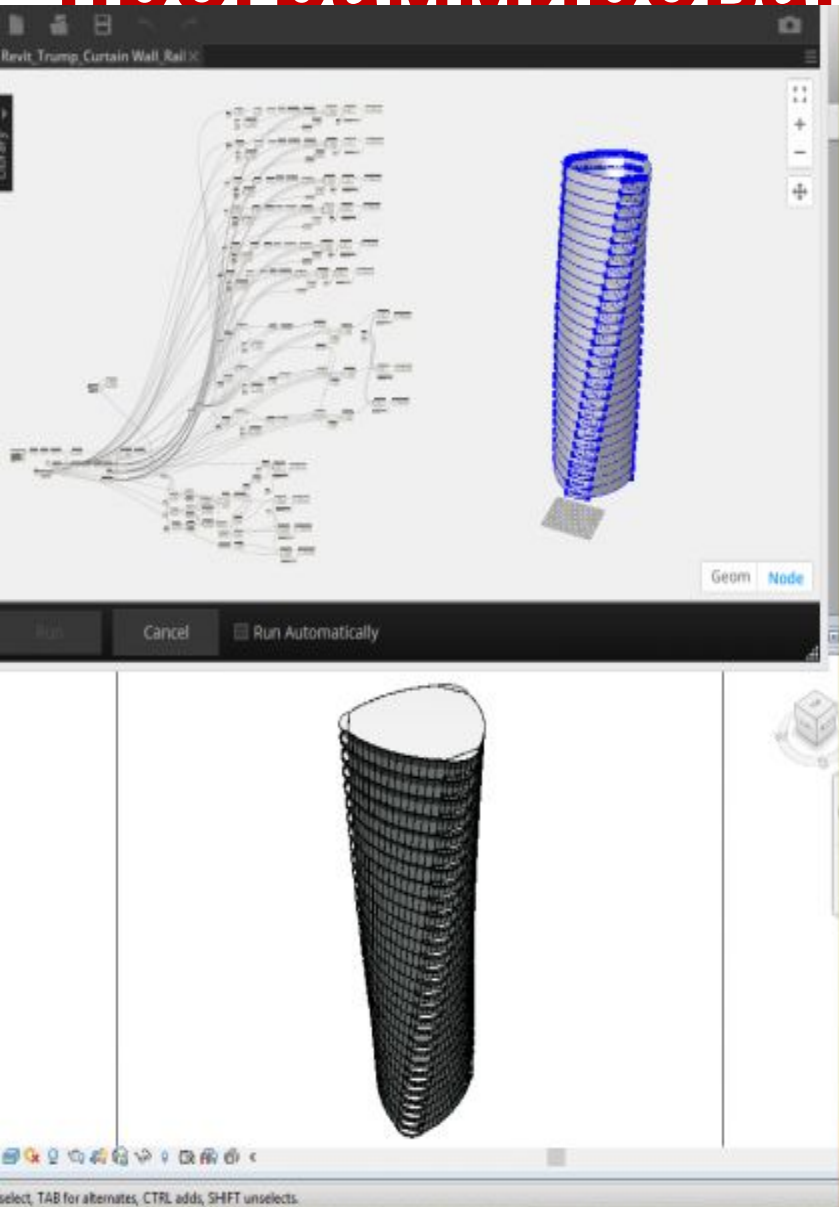
- Формирование сложной архитектуры
- Автоматизация инженерных расчетов
- Перемещение БД между инженерами
- Операции с БД
- Автоматическое формирование чертежей, спецификаций, смет



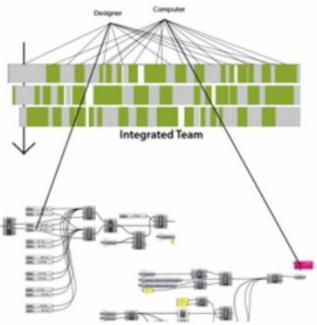
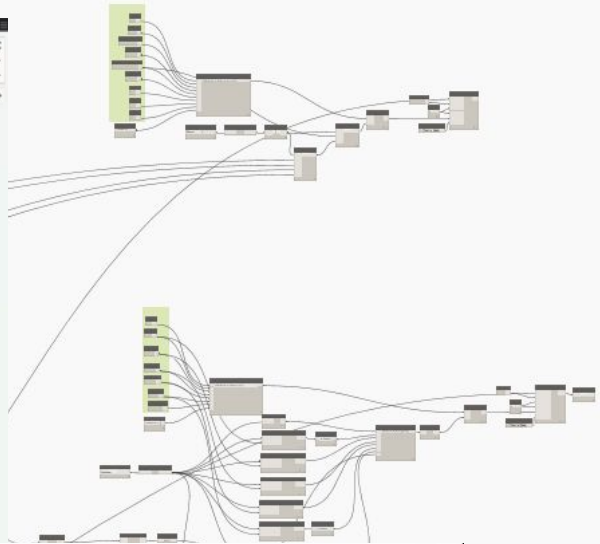
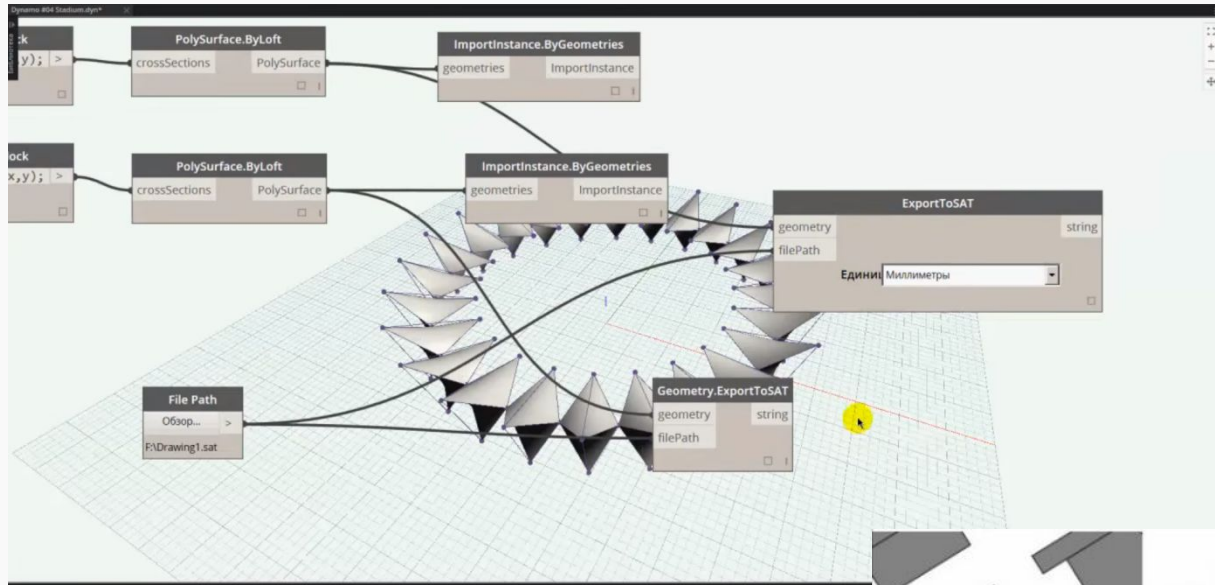
Примеры визуального программирования



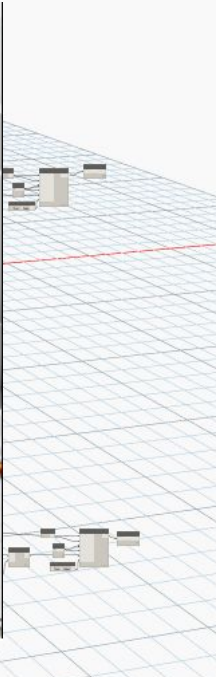
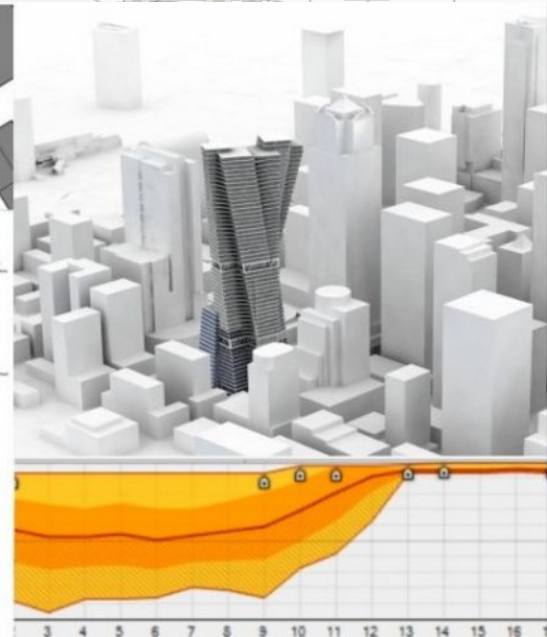
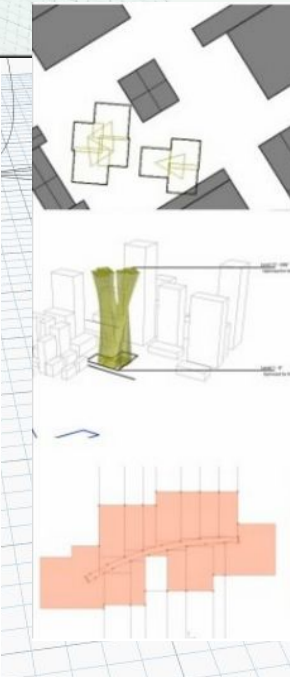
Примеры визуального программирования



Расчеты с визуальным программированием

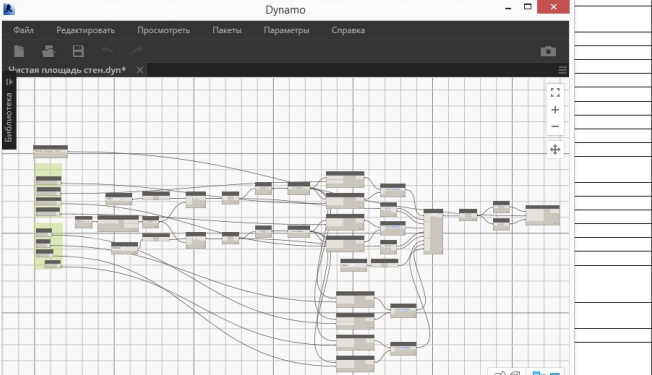


A Field of possible solutions Generated, Analyzed and Sorted by the integrated team

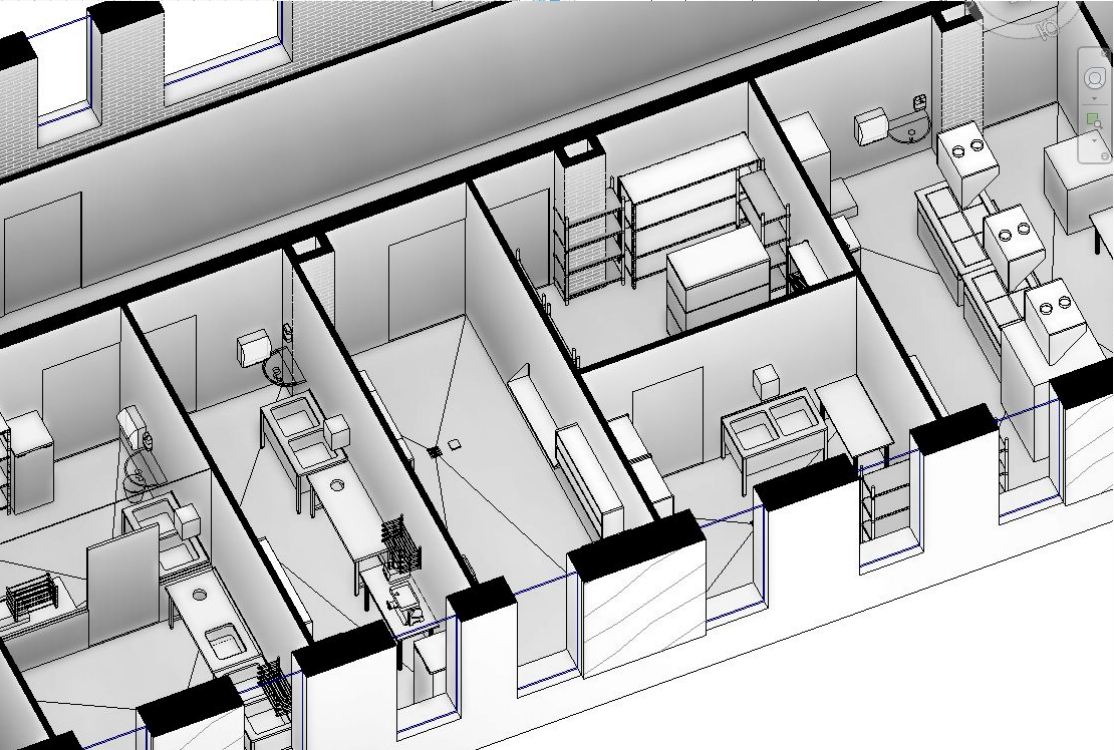
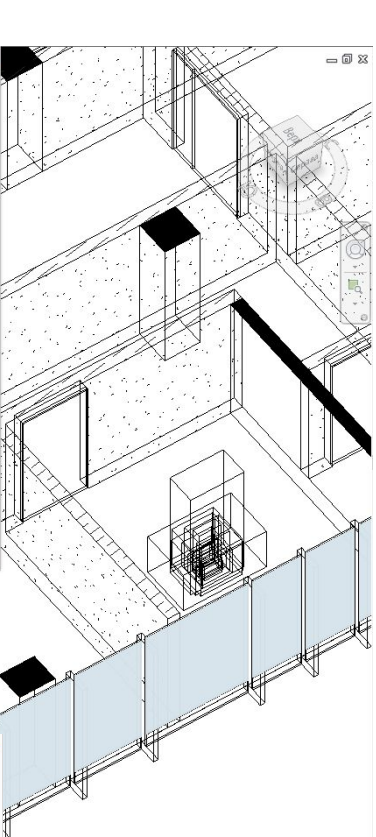
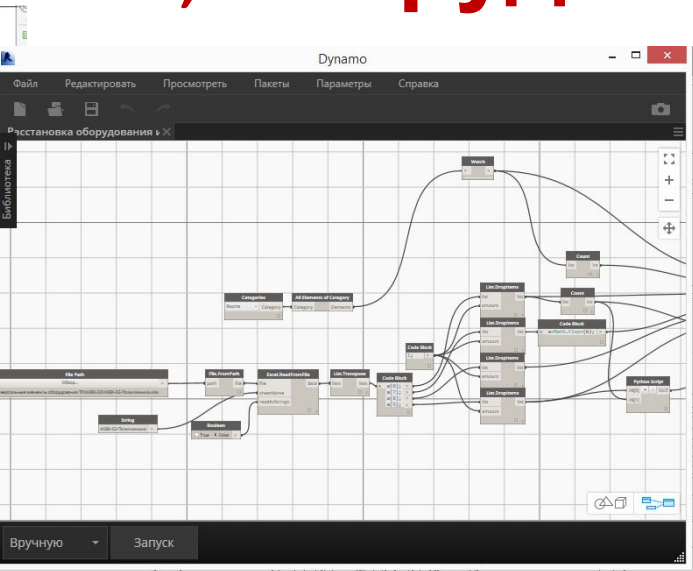


Расстановка мебели, оборудования

Ведомость отделки помещений					
Наименование	Вид отделки элементов интерьера		Чистая площадь стен	Примечание	
	Пол	Стены			
Отделка полов	Акриловая краска	Керамическая плитка	44.43		
Отделка стен и потолка	Воздушная краска	Керамическая плитка	44.43		
Отделка потолка	Воздушная краска	Воздушная краска	18.45		
Отделка стен	Акриловая краска	Керамическая плитка	67.63		



№ пом.	Наименование
1149	Кабинет врача
1150	Помещение хр
1151	Помещение мед
1152	Шлюз
1153	Служ
1154	Полоса инвент
1155	Секционный лс
1156	Кладовая резе
1157	Кладовая расх
1158	Танбур
1159	Санузел
1160	Помещение бр
1161	Кладовая убор
1162	Коридор
1163	Элеваторовод
1164	Каннава на сла
1165	Каннава для ис
1166	Коридор
1167	Коридор
1169	Коридор
1170	Танбур
1174	Резидария
1175	Танбур-влез с
1176	Лесничная к
1177	Лесничная к



```
Redaктирование сценария Python...  
import clr  
clr.AddReference('ProtoDestructory')  
from Autodesk.DesignScript.Geometry import *  
SdWindow = IN[0]  
room1 = IN[1]  
room2 = IN[2]  
SdDoor = IN[3]  
room3 = IN[4]  
rooms = IN[5]  
SdWindow = IN[7]  
SdDoor = IN[8]  
list = []  
for room in rooms:  
    if UserElement(room).Areas:  
        SubUserElement(room).Parameter("Department").AsDouble()["UserAgentElement  
(room).Parameter("Human Inertia").AsDouble() / 2880 / 3.2880  
        list.append(room)  
        count1=rooms.Count  
        while i1<count1:  
            if room[1] is not None:  
                if room[1].UserAgentElement(room[1].Id):  
                    if SdWindow[1] is not None:  
                        S=SdWindow[1]  
                        i1=i1+1  
                        S=S-SdWindow[1]  
                    else:  
                        S=S-SdWindow[1]  
                else:  
                    i1=i1+1  
                    S=S-SdWindow[1]  
            else:  
                count2=rooms.Count  
                while i2<count2:  
                    if room[2] is not None:  
                        if room[2].UserAgentElement(room[2].Id):  
                            if SdWindow[2] is not None:  
                                S=S-SdWindow[2]  
                                i2=i2+1  
                                S=S-SdWindow[2]  
                            else:  
                                i2=i2+1  
                                S=S-SdWindow[2]  
                        else:  
                            count3=rooms.Count  
                            while i3<count3:  
                                if room[3] is not None:  
                                    if room[3].UserAgentElement(room[3].Id):  
                                        if SdDoor[3] is not None:  
                                            S=S-SdDoor[3]  
                                            i3=i3+1  
                                            S=S-SdDoor[3]  
                                        else:  
                                            i3=i3+1  
                                            S=S-SdDoor[3]  
                            else:  
                                count4=rooms.Count  
                                while i4<count4:  
                                    if room[4] is not None:  
                                        if room[4].UserAgentElement(room[4].Id):  
                                            if SdDoor[4] is not None:  
                                                S=S-SdDoor[4]  
                                                i4=i4+1  
                                                S=S-SdDoor[4]  
                                            else:  
                                                i4=i4+1  
                                                S=S-SdDoor[4]  
                                else:  
                                    list.append(room, S)  
                                i4=i4+1  
                                S=S-SdDoor[4]
```

Преимущества Dypato

- Легко в установке и легко начать создавать программы
- Можно начать создавать приложения без программирования
- Можно копировать части чужих скриптов
- Скрипты открываются на любых версиях ревита
- Есть стандартные ноды для взятия/внесения параметров, которые работают стабильнее программных
- Есть стандартные ноды для открытия и записи Excel файлов
- Можно разделить программу на части, контролируя выполнение каждой отдельно
- Более быстрая отладка программ, не требуется перезапускать программу
- Не требуется создавать интерфейсы для ввода значений
- Можно работать без Revit с другими программами

Недостатки Dynato

- На Python меньше примеров программной реализации задач, чем на C#
- Нет защиты кода
- Сложнее делать интерфейс, чем на Visual Studio
- Нет удобного сворачивания кода и навигации по коду, которые нужны при крупных программах
- Нет стандартной связи с БД, такой как через Visual Studio
- Нет инсталлятора, для установки кнопок запуска в ленту
- Скорость работы немного ниже, чем при прямом кодировании.

Итог

- На Dynamo быстрее обрабатывать решения
- На Dynamo эффективнее решать небольшие прикладные задачи
- Dynamo удобнее для того, чтобы учиться программировать
- Dynamo удобнее для не программистов в решении задач
- Если потребуется коммерческая разработка, со своим интерфейсом и связью с БД, то лучше писать в Visual Studio на C#
- Dynamo удобно для решения небольших задач внутри своей организации
- Dynamo = BIM Excel

Уровни знания Дунато

1. А что это?
2. Не в жизнь не запущу этого зверя!
3. А он точно ничего не попортит?
4. Могу запускать Дунато скрипты
5. Могу составлять скрипт из нодов
6. Знаю как скачать пакет и встроить в свой скрипт
7. Могу подправить чужой скрипт
8. Могу подправить python код
9. Когда не хватает нодов подключаю python
10. Програмирую всё в python'е
11. Мне не нужен Дунато, я пишу всё на C#

Попов Александр

VIM-менеджер

E-mail: alexandr_popov_89@mail.ru

Тел: +7 (904) 611-86-44

Skype: alexandr_popov_89

Спасибо за внимание!