

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.

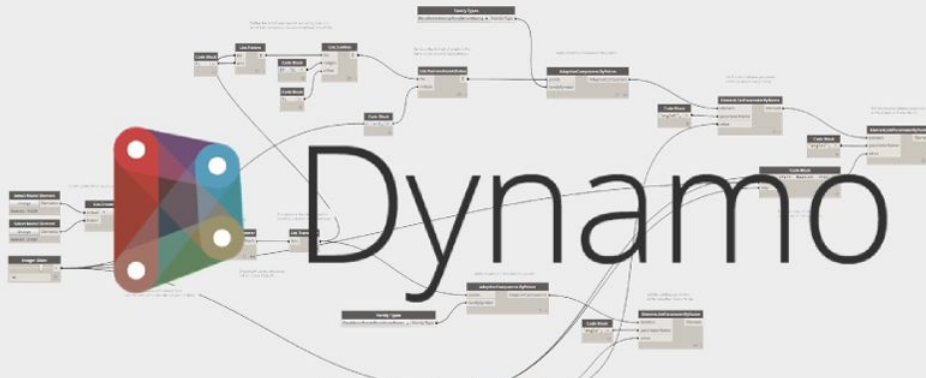
ВМ-координатор, ВМ-менеджер

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

ВМ-менеджер

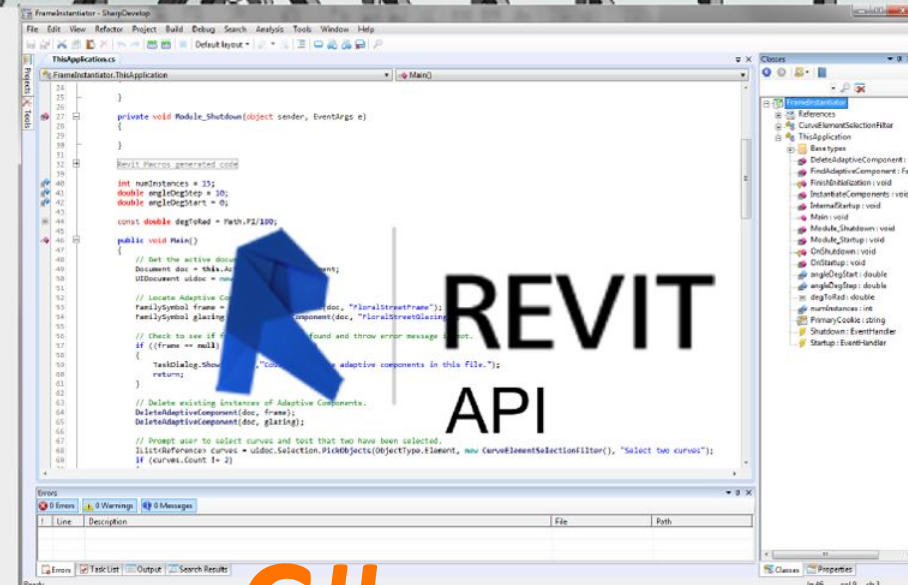


Чем мы можем дорабатывать Revit?



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';

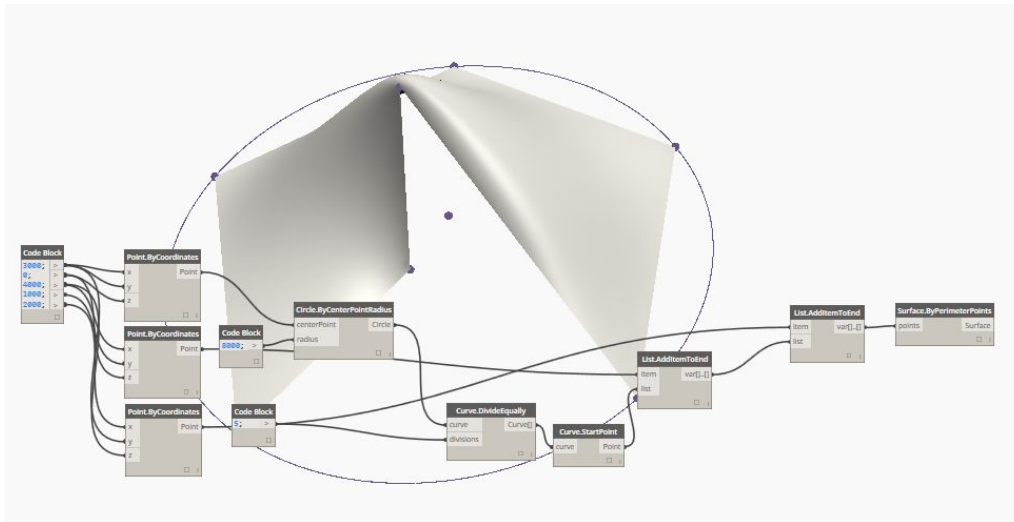
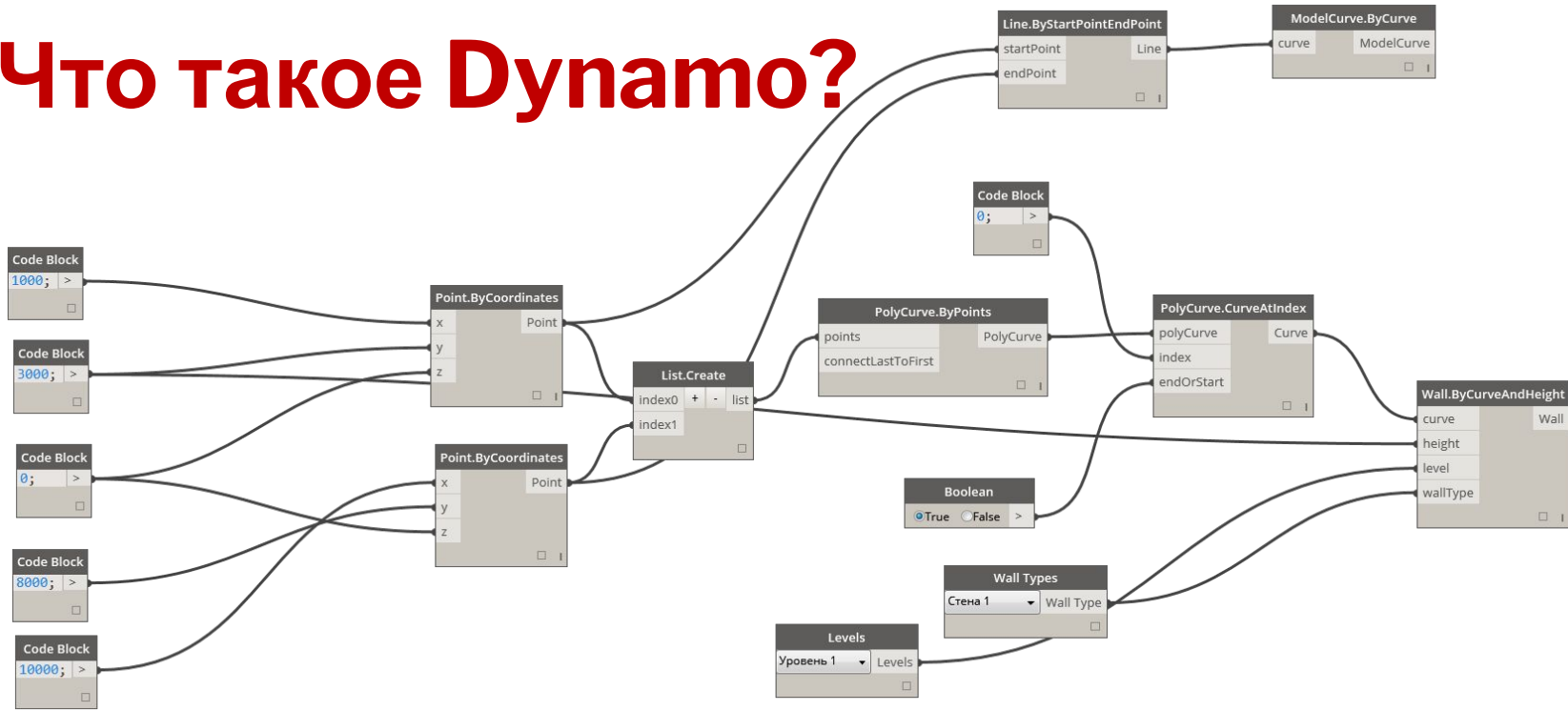
if (!defined('PSI_CONFIG_FILE') || !defined('PSI_DEBUG')) {
    $tpl = new Template("/templates/html/error_config.html");
    echo $tpl->fetch();
    die();
}
```



```
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?

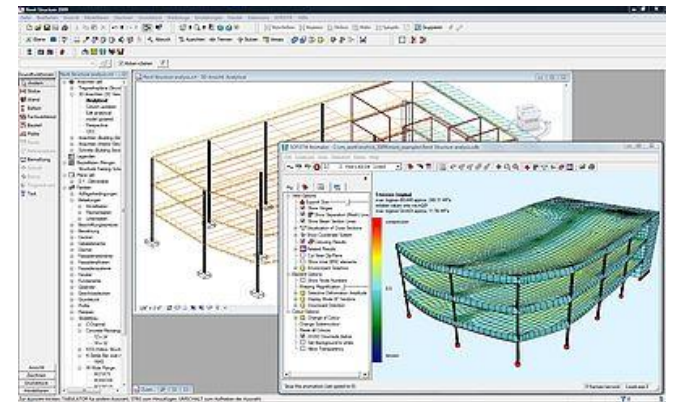


```
Редактирование сценария Python...
14 from RevitServices.Transactions import TransactionManager
15 from System.Collections.Generic import *
16 # Import RevitAPI
17 clr.AddReference("RevitAPI")
18 import Autodesk
19 from Autodesk.Revit.DB import *
20
21 doc = DocumentManager.Instance.CurrentDBDocument
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
```

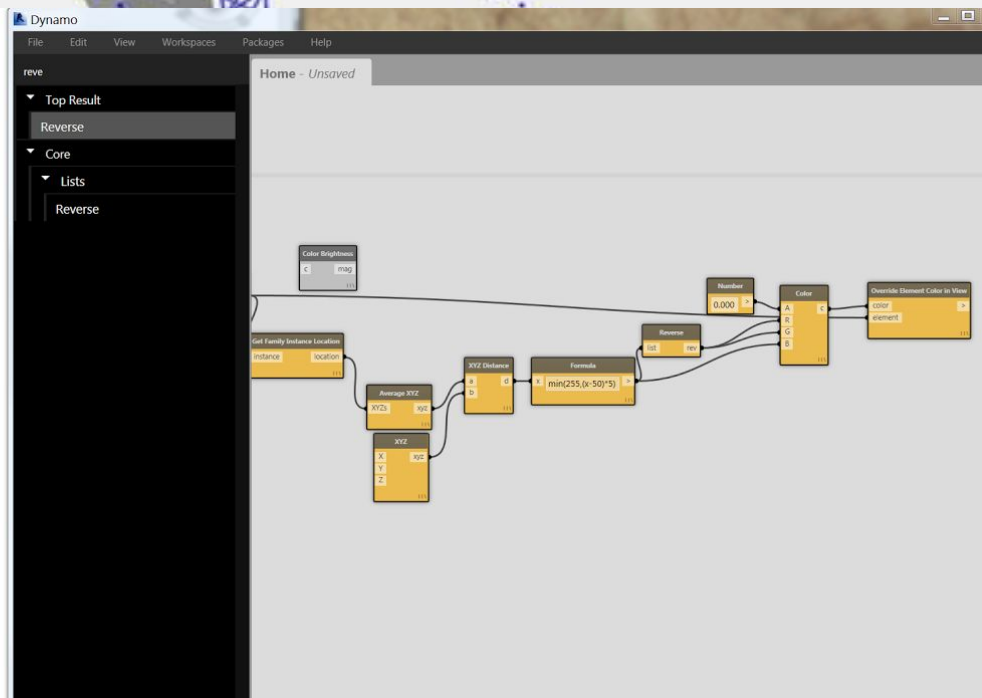
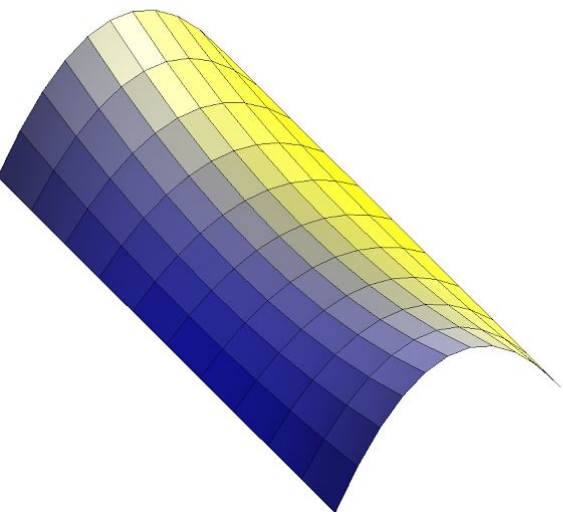
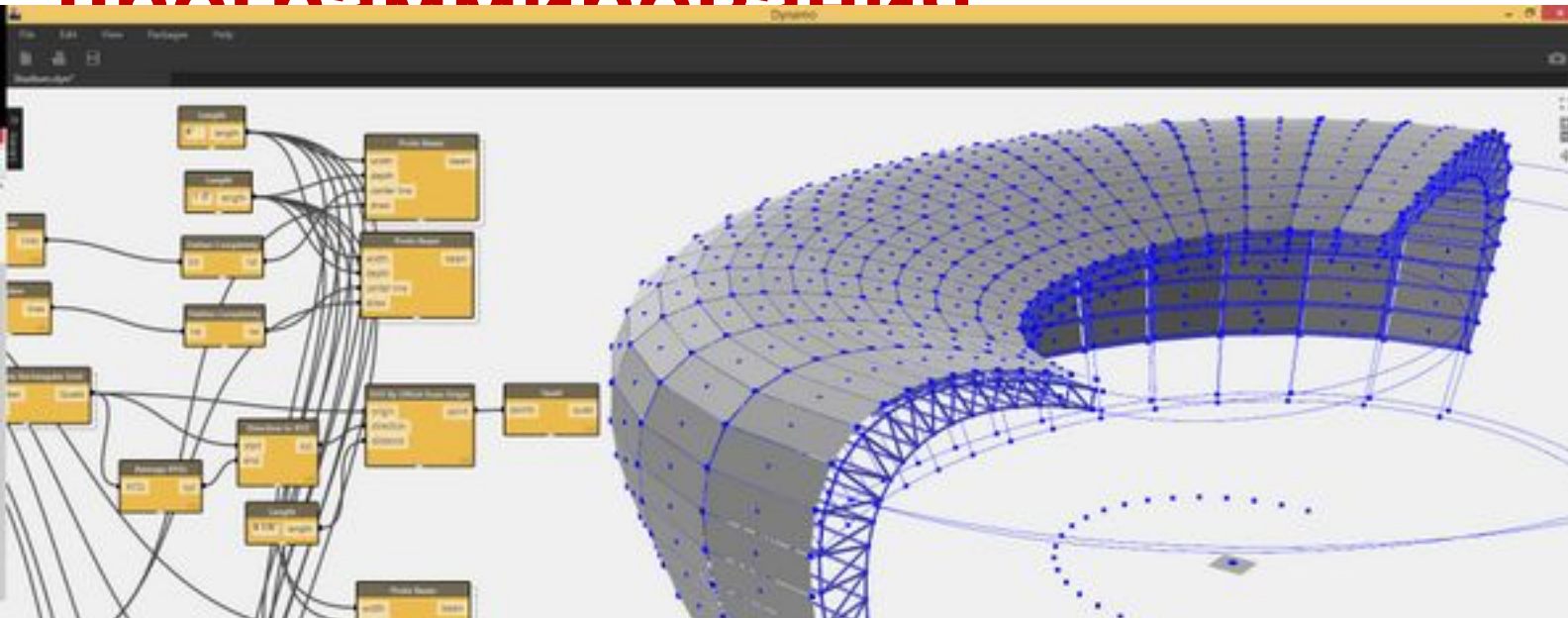
принять изменения Отмена

Задачи для визуального программирования

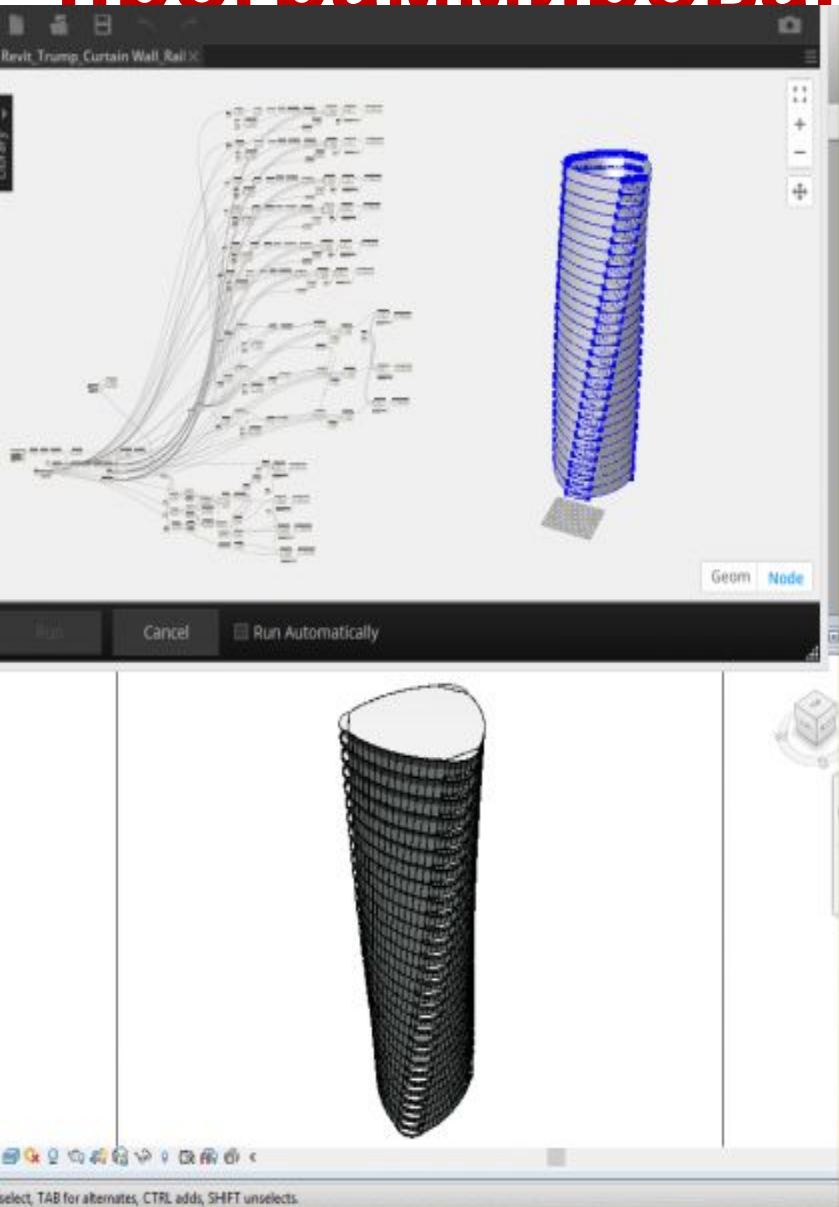
- Формирование сложной архитектуры
- Автоматизация инженерных расчетов
- Перемещение БД между инженерами
- Операции с БД
- Автоматическое формирование чертежей, спецификаций, смет



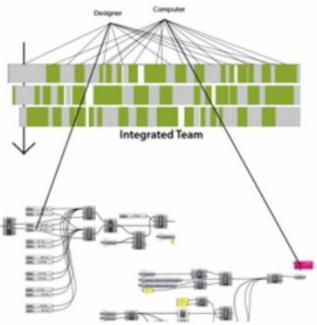
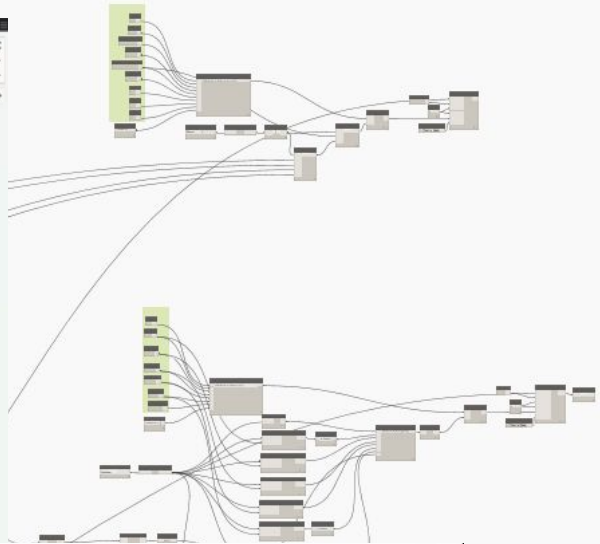
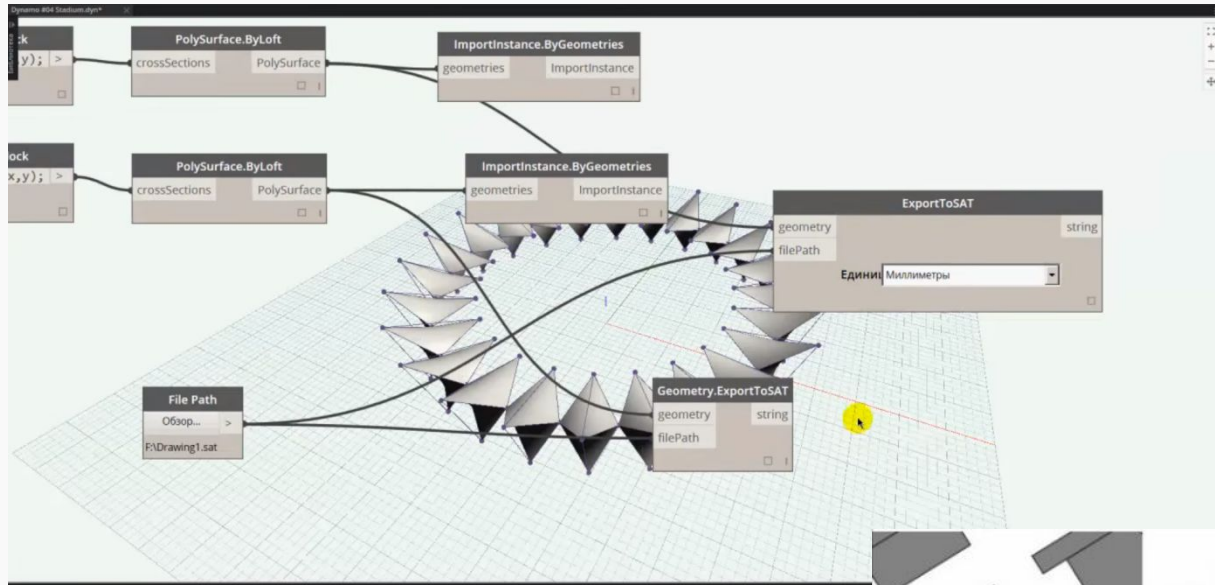
Примеры визуального программирования



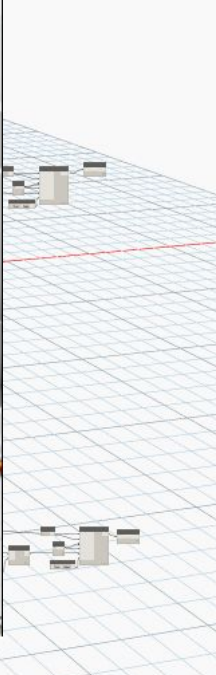
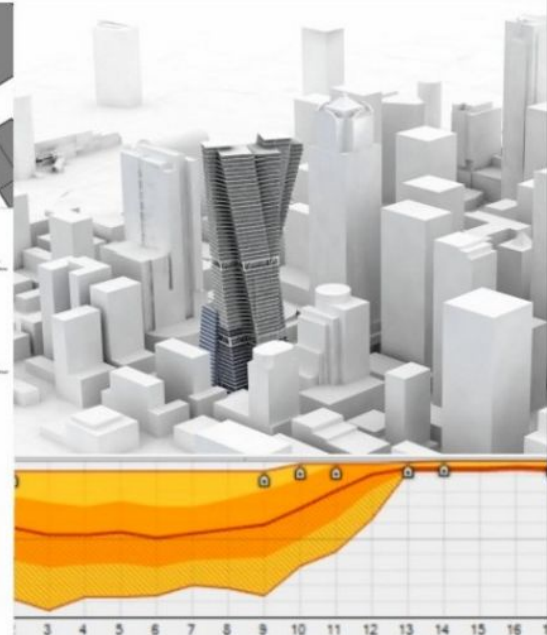
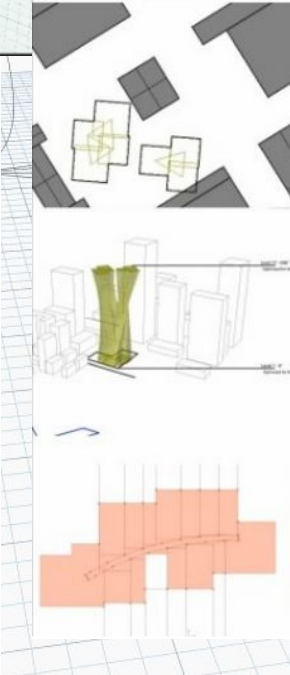
Примеры визуального программирования



Расчеты с визуальным программированием

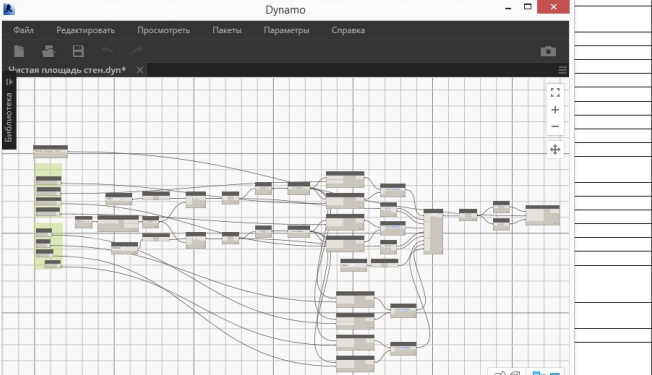


A Field of possible solutions Generated, Analyzed and Sorted by the integrated team

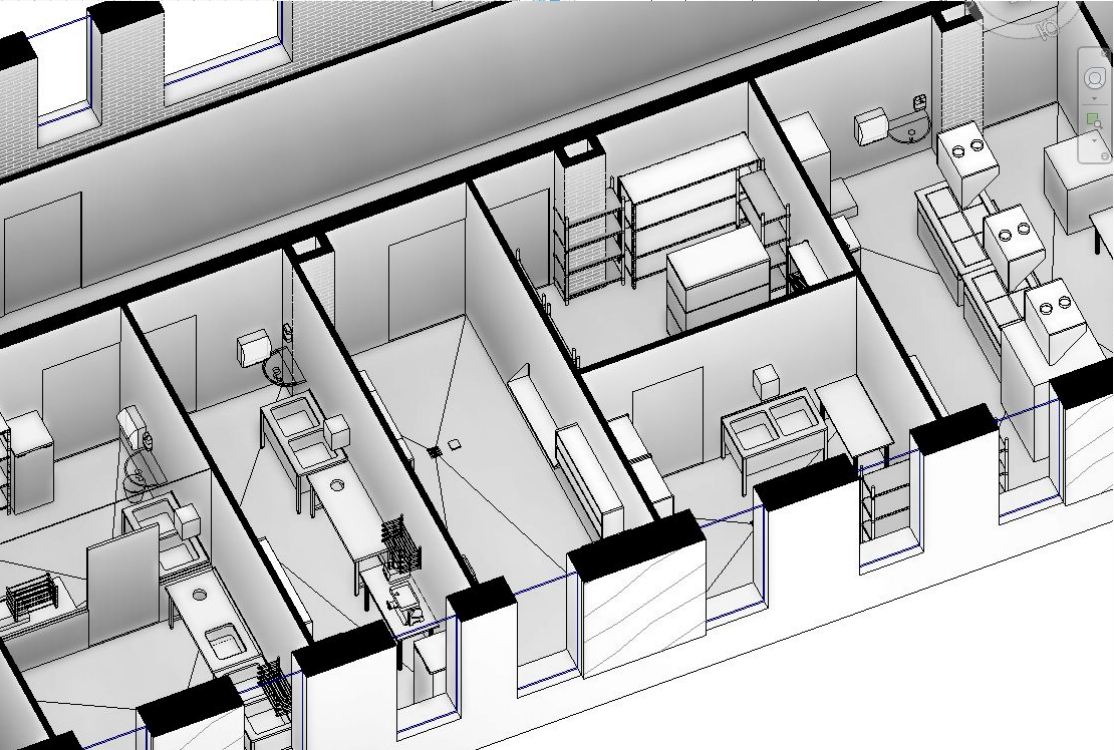
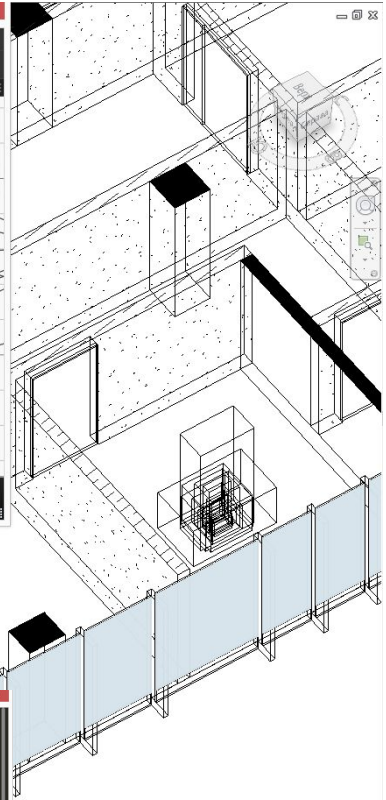
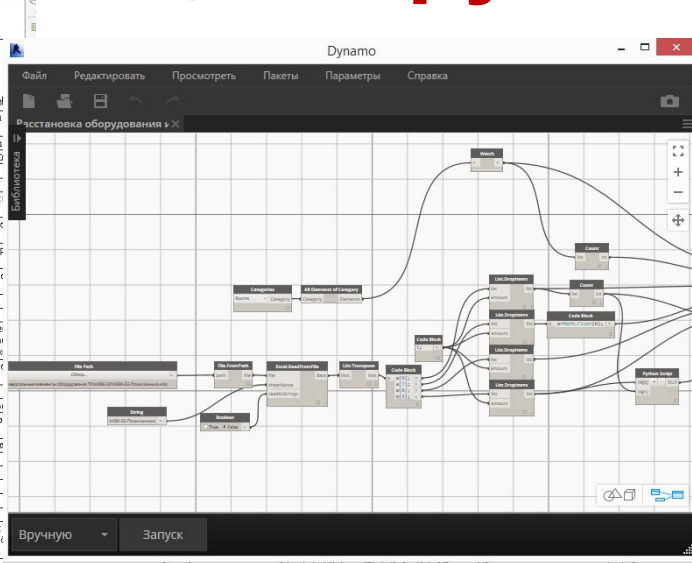


Расстановка мебели, оборудования

Ведомость отделки помещений					
	Вид отделки элементов интерьера				
Наименование	Полылок	Площадь, м ²	Стены	Чистая площадь стен	Примечание
Отделочные работы: оклеивание обоев, окраска	Акриловая краска	6.7	Керамическая плитка	44.43	
Отделочные работы: оклеивание обоев, окраска	Воздушно-силикатная краска	9.2	Керамическая плитка	44.43	
Отделочные работы: оклеивание обоев, окраска	Воздушно-силикатная краска	2.6	Воздушно-силикатная краска	18.45	
Отделочные работы: оклеивание обоев, окраска	Акриловая краска	18.1	Керамическая плитка	67.63	



№ пом.	Наименование
1149	Кабинет врача
1150	Помещение хр.
1151	Помещение мед.
1152	Шлюз
1153	Служ.
1154	Половое инвент. веранды
1155	Секундарный пс.
1156	Кладовая резе. оборудования
1157	Кладовая расх. материала
1158	Танбур
1159	Сануз.
1160	Помещение бр. хранения аптеч. материалов
1161	Кладовая убор. инвентаря
1162	Коридор
1163	Элеваторовый
1164	Комната на сла. насабия
1165	Комната для на.
1166	Коридор
1167	Коридор
1169	Коридор
1170	Танбур
1174	Резид. квартира
1175	Танбур-влез с. насабия пох. вездуха
1176	Лесничная к.
1177	Лесничная к.



```
Redaктирование сценария Python...

import clr
clr.AddReference('ProtoDecomposition')
from Autodesk.DesignScript.Geometry import *

SdDoor = IN[0]
room1 = IN[1]
room2 = IN[2]
SdDoor = IN[3]
room1 = IN[4]
rooms = IN[5]
SdWindow = IN[6]
SdDoor = IN[7]
list = IN[8]

for room in rooms:
    if UserElement(room).Areas:
        SubUserElement(room).Parameter("Department").AsDouble["UserElement"]
        (room).Parameter("Human Inertia").AsDouble["/J.2880/3.2880"]
        list.append(room)

count1=room1.Count
while i1<count1:
    if room[1] is not None:
        if room[1].UserElement(room[1].Id):
            if SdWindow[1] is not None:
                S=SdWindow[1]
            else:
                S=SdWindow[11]
            i1=i1+1
count2=room2.Count
while i2<count2:
    if room[2] is not None:
        if room[2].UserElement(room[2].Id):
            if SdWindow[2] is not None:
                S=SdWindow[2]
            else:
                S=SdWindow[12]
            i2=i2+1
count3=room3.Count
while i3<count3:
    if room[3] is not None:
        if room[3].UserElement(room[3].Id):
            if SdDoor[3] is not None:
                S=SdDoor[3]
            else:
                S=SdDoor[13]
            i3=i3+1
count4=room4.Count
while i4<count4:
    if room[4] is not None:
        if room[4].UserElement(room[4].Id):
            if SdDoor[4] is not None:
                S=SdDoor[4]
            else:
                S=SdDoor[14]
            i4=i4+1
list.append(room, S)

return list
```

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!