

DOI: <https://doi.org/10.37909/2542-1352-2024-1-1008>

Программирование и параметрическое моделирование в ArchiCAD как творческий инструмент архитектора

Евгений Мусиенко

Доцент

Новосибирский государственный университет архитектуры, дизайна и искусств имени А.Д. Крячкова
mus@azd.ru, [ORCID](#)

Аннотация

В статье затронуты общие вопросы создания параметрических объектов в различных профессиональных программах для архитектурного проектирования. Предложен творческий метод, базирующийся на создании параметрических объектов с помощью программирования и на основе художественных экспериментов, заключающихся в получении массивов эстетически значимых графических данных. Эти эксперименты проводились в программе Архикад с использованием скриптов GDL, модуля PARAM-O в сочетании с традиционными графическими возможностями. Метод проиллюстрирован на конкретных примерах, сделан вывод, что архитектор, умеющий программировать либо работающий вместе с программистом, способен обогатить свою творческую палитру, а использование синтетического метода проектирования через эвристическую архитектурную задачу и ее решение программными средствами, дает нестандартные интересные с художественной точки зрения результаты.

Ключевые слова: параметризм, параметрическая модель, Архикад, архитектурное творчество, творческий метод, скрипты GDL

Для цитирования: Мусиенко Е.И. Программирование и параметрическое моделирование в ArchiCAD как творческий инструмент архитектора // Творчество и современность. 2024. № 1. С. 52–59.

DOI: <https://doi.org/10.37909/2542-1352-2024-1-1008>

Programming and Parametric Modelling In ArchiCAD as a Creative Tool for the Architect

Evgenij Musienko

Associate Professor

Kryachkov Novosibirsk State University of Architecture, Design and Arts

mus@a3d.ru, [ORCID](#)

Abstract

The article touches upon the general issues of creating parametric objects in various professional programs for architectural design. A creative method based on creation of parametric objects with the help of programming and on the basis of artistic experiments consisting in obtaining arrays of aesthetically significant graphical data is proposed. These experiments were carried out in Archicad program using GDL scripts, PARAM-O module in combination with traditional graphic capabilities. The method is illustrated on concrete examples, the conclusion is made that the architect, who knows how to program or works together with a programmer, is able to enrich his creative palette, and the use of synthetic method of design through a heuristic architectural problem and its solution by software means, gives non-standard interesting results from the artistic point of view.

Keywords: parametricism, parametric model, ArchiCAD, architectural creativity, creative method, GDL scripts

For citation: Musienko E. (2024) Programming and Parametric Modelling In ArchiCAD as a Creative Tool for the Architect. *Creativity and modernity*. 23 (1). 52–59.

Введение

В современных системах архитектурного проектирования (САПР) среднего и тяжёлого классов наличие параметрической модели заложено в идеологию самих систем. Поэтому естественно, что многие из них имеют встроенные или подключаемые языки программирования. Таким образом, архитектор или любой пользователь может не только чертить, рисовать, моделировать с помощью собственно графических возможностей программы, но и применять для тех же целей программирование. Так как в дальнейшем наши рассуждения и примеры будут базироваться именно на возможности использования программирования для создания объектов, помещаемых внутрь САПР, рассмотрим это для различных систем.

- Ревит (Revit) имеет возможность создания новых параметрических объектов с помощью комплекта разработчика (SDK), в котором можно использовать языки программирования VB.NET, Python и C# [Divin 2020].
- Архикад (ArchiCAD) позволяет создавать параметрические объекты с помощью написания скриптов GDL (Языка геометрических определений, встроенного в программу.) [Кравченко].
- Российская САПР Nanosad имеет программный интерфейс (API), позволяющий разрабатывать собственные приложения и дополнительные модули автоматизации с использованием .NET, C++, JS, VBS и LISP [NanoCAD для разработчиков].
- Российская САПР Renga также имеет API, позволяющий создавать расширения с помощью Visual C#, Visual C++, Python, но в настоящее время в Renga нет возможности создавать новые объекты, а только менять свойства существующих [Документация по API Ренга].

В помощь проектировщикам, ориентированным на работу с визуальной информацией при создании нового параметрического объекта, имеются специальные инструменты. Вместо написания текста программы со сложными и строгими правилами проектировщик использует инструменты для визуального программирования, которые автоматически формируют текст программы. Так в Revit имеется дополнение Dupato, которое позволяет работать в рамках визуального программирования, автоматически получая программу для создания объекта [Что такое Dupato]. А в Архикаде такую же функцию осуществляет PARAM-O — инструмент графического создания объектов — одного из типов библиотечных элементов программы.

Российская система параметрического 3D моделирования Сапфир-3D совместно с Генератором позволяют визуально программировать модель с последующим ее экспортом в ЛИРА-САПР [Система параметрического 3D].

По мнению автора, архитектору легче всего освоить использование GDL скриптов и модуля PARAM-O. Поэтому на примере их использования мы будем рассматривать создание нужных нам параметрических объектов.

Поскольку все элементы, создаваемые в САПР системах, а, следовательно, и архитектурные объекты в них, имеют различные параметры, то возможно с помощью программирования и изменения параметров управлять свойствами этих объектов, создавая массивы вариантов. Можно проводить программируемые архитектурные эстетические эксперименты, где результатами должно стать получение максимально гармоничных и адекватных архитектурной задаче творений архитектора-программиста. Неоспоримым преимуществом метода становится множество получаемых результатов, изображений модели, отличающихся измененными параметрами. Из этого массива можно выбрать необходимое, эстетически ценное и с выбранными вариантами продолжать эксперименты. Графическими средствами создание огромной массы вариантов очень трудно, либо, вообще невозможно.

Эксперимент по «проигрыванию» параметрического объекта может быть так организован, что предвосхитить его результаты частично или полностью невозможно. Поэтому, проводя такие эксперименты, можно:

- подтвердить или опровергнуть интуитивные ожидания;
- организовать доказательную базу своей идеи, так как все смогут увидеть массив вариантов и убедиться, что выбран лучший;
- получить новый, заранее не представимый эстетический результат.

Справедливости ради стоит оговориться, что различного рода архитектурные эксперименты с параметрическими моделями проводятся уже давно, что стало одним из основных формирующих факторов появления такого стиля, как «параметризм».

О параметризме как актуальном стиле в архитектуре и дизайне

«В 2008 г. Патрик Шумахер (ZNA, Zaha Hadid Architects) заявил, что в течение прошедших 15 лет созрел новый стиль внутри авангардистской архитектуры.

Поэтому параметризм был представлен на 11-й архитектурной биеннале в Венеции, а статья Шумахера [Шумахер] была названа «манифестом параметризма» [Надыршин 2013].

Этот стиль определен Шумахером как сложное системное явление, начиная от философского, социального его определения, заканчивая технологическим. Именно для параметризма характерен как творческий, так и инженерный метод, использующий создание цифровых параметрических моделей. Параметризм характеризуется специфической эстетикой, которая так же определена в статьях Патрика Шумахера. Это нелинейная эстетика текучих, криволинейных форм и поверхностей. Во многом она копирует объекты природы, состоящие из подобных, похожих, но не одинаковых элементов. Соответственно, параметризм требует изменения традиционных строительных технологий как проектирования, так и собственно возведения зданий и сооружений.

«В архитектурном дизайне — это в первую очередь информационное моделирование зданий (BIM). Суть этих изменений в так называемом гибком (Flexible Manufacturing System — FMS) проектировании на базе CAD, CAE, CAM технологий, когда дизайнер с дисплея на рабочем столе может контролировать весь процесс автоматизированного проектирования, изготовления и сборки деталей» [8].

Таким образом, архитектурное творчество в рамках параметризма требует и специальной строительной базы, и соответствующих программных продуктов, например, Rhino и Grasshopper.

Однако, многие российские архитекторы до сих пор работают в рамках пусть и постоянно совершенствуемых, но все-таки традиционных технологий архитектурно-строительного комплекса. Программные продукты, на которых ведется проектирование, поддерживают BIM и при этом носят универсальный характер, позволяющий не замыкаться в рамках конкретного стиля.

Соответственно, в данной статье рассматривается метод использования программирования в творческой работе архитектора в самом широком смысле: в рамках любого стиля и строительной парадигмы.

Параметрические модели в Архикад

В Архикаде массив элементов создается командой «Тиражирование» и представляет собой несколько копий исходного элемента или исходной группы элементов. Эти копии после завершения команды

друг с другом никак не связаны, и если понадобится изменить, например, расстояние между ними, то нужно их удалить и повторить команду «Тиражировать» с другими параметрами, задающими расстояние. В различных САПР программах создание массива реализовано по-разному, например, в Naposad массив представляет собой параметрический объект и его параметры можно редактировать уже после построения.

В Архикаде можно добиться того же, но придется добавить элементы программирования, нужно задать параметрический массив с помощью создания предназначенного для этой цели объекта. Напомним, что в Архикаде объекты являются одним из типов библиотечных элементов и создаются пользователями на основе скриптов GDL.

Создадим три таких объекта, которые назовем «Параметрический массив 1», «Параметрический массив 2», «Параметрический массив 3», соответственно различным траекториям расположения тиражируемых объектов: квадратной, круглой, в виде волны из двух полуокружностей. Эти траектории изображены в верхней левой части рисунка 1. В каждом из этих объектов «Параметрический массив ...» мы можем менять тиражируемый объект (элемент), который является одним из его параметров. Кроме этого, мы можем изменять

- расстояние между тиражируемыми объектами (элементами);
- количество тиражируемых объектов(элементов);
- смещение точки вставки тиражируемого объекта (элемента) относительно его точки вставки на траекторию.

На рисунке 1 в левой нижней части изображены три объекта (элемента), которые тиражировались для демонстрации возможностей использования созданного автором статьи параметризованного библиотечного элемента «Параметрический массив ...», а результат этого использования изображен в правой части рисунка 1.



Рисунок 1. Образцы параметрических массивов
Figure 1. Samples of parametric arrays

Систематизируя результаты изменения параметров библиотечного элемента, становится понятно, что был создан не просто библиотечный элемент, а достаточно мощный инструмент для проектирования, например, пергол, детских площадок, мест отдыха и т. д., так как на различных массивах данных возможно получить различные эстетически значимые результаты.

Следуя такой же методике, самые разнообразные параметрические «объекты-инструменты» можно создавать не только как параметрические массивы, но и как любые другие логически связанные параметризованные структуры для конкретных архитектурных задач.

Перейдем к рассмотрению различных примеров.

Параметрическая модель колонны

Условимся, что существует задача по проектированию опорной части калитки (рисунок 2, слева). И это, не просто столбик, а, в рамках постмодернистской эстетики, архитектору нужна неоклассическая стилизация колонны. Для решения задачи автором была создана схематичная модель колонны в виде параметрического трехмерного объекта, изображенного на рисунке 2 справа.



Рисунок 2. Задача по проектированию опорной части калитки

Figure 2. The task of designing the supporting part of the gate

Параметрами колонны являются: ширина, глубина и высота основания; радиус и высота цилиндра над основанием, радиус и высота тела колонны; ширина, глубина и высота полки, и радиус шара.

Колонна была построена с помощью инструмента PARAM-O. Построение осуществлялось из блоков, цилиндров и сферы с использованием «узлов», задающих параметры перемещения. Все тела согласованы по высоте: высота низа следующего элемента зависит от высоты предыдущих элементов, изменение высота одного из них сказывается на высоте всей колонны. При завершении работы с этим инструментом объект был сохранен в библиотеку Архикад. Параметры созданного PARAM-O объекта модели колонны можно задавать в диалоговом окне инстру-

мента «Объект» вручную, но для того, чтобы создать массив с автоматически изменяющимися параметрами из уже построенной модели колонны, возможностей инструмента PARAM-O не хватило. Поэтому сохраненный объект модели колонны был открыт из библиотеки Архикад и изменен его 3D скрипт. Из автоматически созданного PARAM-O в момент сохранения объекта 3D скрипта, мною была создана подпрограмма, с помощью обращения к которой с различными значениями параметров, строится массив колонн. При этом пришлось редактировать 3D скрипт, делая параметрами имена групп, на которые PARAM-O разбивал элементы скрипта. Это было необходимо для того, чтобы при каждом обращении к подпрограмме с разными значениями параметров тел, формирующих колонну, результат получался разный. Отредактированный объект, в который были добавлены вложенные циклы с вызовами подпрограмм с разными значениями параметров, был сохранен под новым именем.

На рисунке 3 изображен объект, представляющий из себя массив колонн, которые имеют в различных строках разный объем шара, а в разных столбцах разную высоту тела колонны.

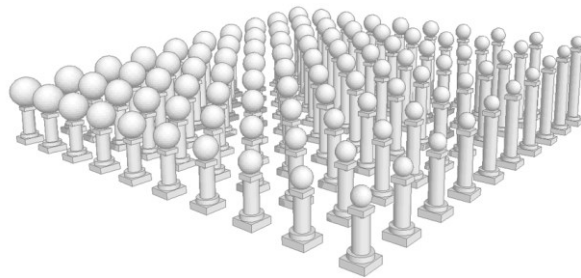


Рисунок 3. Массив колонн с различными параметрами высоты и радиуса шара

Figure 3. An array of columns with different height and radius parameters of the ball

Массив колонн можно создать по-разному, изменяя как сами параметры, так и наборы изменяемых параметров, под конкретную задачу. Отсмотрев все колонны в полученном массиве, выбрать максимально элегантную (эстетически значимую для автора), вставить ее в контекст конкретной калитки и продолжить эксперименты с параметрами для доведения пропорций колонны до адекватного задумке автора эстетического совершенства.

Причем, спрограммированный параметрический инструмент «колонна» можно сохранить и использовать в дальнейшем, совершенствовать, изменять сообразно задаче, добавляя детали и параметры при проектировании таких архитектурных объектов как, например, ротонда, портик, колоннада и т. д.

Сам характер архитектурных задач, как и характер параметрически программируемых объектов, может быть очень разным. Можно работать не только с объемами, но и с готовыми изображениями и их частями, с элементами плоской графики. Задачей может стать получение массы графических изображений с перебором всех возможностей. Рассмотрим такой пример.

Варианты расстановки окон на фасаде

Предположим, что на 3 места на фасаде претендуют 3 различных по форме окна. Задача в том, чтобы увидеть, все варианты того, как выглядит фасад с разным размещением окон. Таких наборов можно сформировать 6. Для решения задачи создания всех вариантов размещения окон был создан объект, в котором в качестве фона размещено изображение фасада

здания. Это изображение представляет собой растровую картинку. Окна также представлены в виде растровых изображений. Имена файлов, в которых сохранены эти изображения, являются параметрами объекта. Еще одним параметром этого объекта является номер этапа формирования вариантов расположения окон.

Если значение номера этапа задать равным 1, то этот объект будет представлять изображение фасада здания повторяющегося столько раз, сколько разных окон участвует в этой операции. Пользователь должен расставить каждое окно на возможное для него место. Это необходимо, поскольку окна могут иметь разные габариты и в одном и том же месте фасада будут размещены по-разному. Если окон 3 и точек их размещения 3, то потребуется 9 операций размещения окон (рисунок 4).



Рисунок 4. Размещение окон на возможные для них места

Figure 4. Placing windows on possible locations for them

После расстановки окон на возможные места, пользователь открывает диалоговое окно параметров объекта и изменяет номер этапа с 1 на 2. На втором этапе объект изменяет изображение, превращаясь в массив

изображений всех вариантов размещения окон. На рисунке 5 изображены все варианты размещения 3 окон на 3 возможных местах.



Рисунок 5. Все варианты размещения 3 окон на 3 возможных местах

Figure 5. All options for placing 3 windows in 3 possible locations

Безусловно, мест размещения окон, да и самих окон может быть больше. Создание вариантов не требует от пользователя большого труда, но стоит учитывать резкое возрастание массива данных, поэтому задавать большое количество окон и мест размещения нужно разумно. Если количество окон равно n , а число точек размещения — m , то программа создаст $n!/(n-m)!$ вариантов, где $n!$ — произведение всех целых чисел от 1 до n . Например, если окон 5, а мест размещения 4, то программой будет создано 120 вариантов, а если окон 9, а мест размещения 4, то вариантов будет 3024. Кроме этого, стоит учитывать, что программа создает длинную цепочку вариантов, которую приходится отсматривать по частям, если вывести на экран все варианты сразу, то изображение каждого из вариантов будет очень маленьким.

Однако, при продуманном использовании, на основе выше продемонстрированной идеи можно создавать разнообразные и интересные творческие инструменты.

Объемный параметрический объект на основе существующей постройки

К проведению следующего эксперимента автора подтолкнула церковь St. Louis Priory Chapel (Монастырская часовня) в Сент-Луис, штат Миссури, США, изображенная слева на рисунке 6. Она имеет четкую, логически понятную структуру. Очевидно, что созданный на ее основе параметрический объект несет в себе большой эстетический потенциал. Хотелось подтвердить или опровергнуть эти интуитивные догадки.

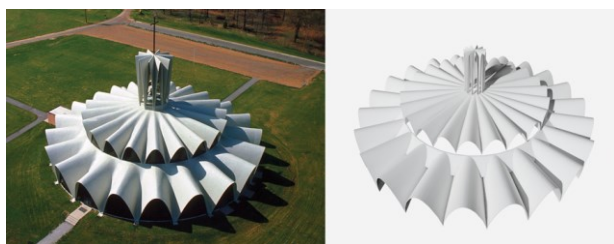


Рисунок 6. Церковь St. Louis Priory Chapel и ее условная модель

Figure 6. St. Louis Priory Chapel and its conditional model

Сначала с помощью инструмента PARAM-о была создана условная параметрическая модель часовни (изображена на рисунке 6 справа), затем сохраненный в библиотеку Архикада объект был открыт, и был отредактирован его 3D скрипт для того, чтобы была возможность автоматически создать массив моделей с различными значениями параметров (то есть, объект

был параметризован). Получившийся новый объект был сохранен под другим именем уже как собственно творческий параметрический инструмент.

Дальнейшим шагом стало освоение возможностей этого инструмента.

Задумывая объект, создавая его, параметризуя, нужно учитывать не только его собственную внутреннюю логику (которая отражена в структуре объекта в базовом варианте), но и предусматривать потенциальное «поле возможностей», «облако изменений», которое можно будет изучать, рассматривать, получив в визуальном выражении через каждый из доведенных до вывода на экран монитора вариантов при задании конкретного сочетания параметров.

Поэтому число параметров, логика их взаимодействия также очень важны, как и собственно область значения этих параметров. Если параметров не много, и они дискретны, мы можем получить все облако возможностей методом «полного перебора вариантов». Сложнее дела обстоят в ситуации, когда параметров много, либо вообще бесконечное множество, тогда при изучении визуальных возможностей модели просто нужно использовать разные стратегии.

Самая простая стратегия — дискретно и контрастно менять один из параметров.

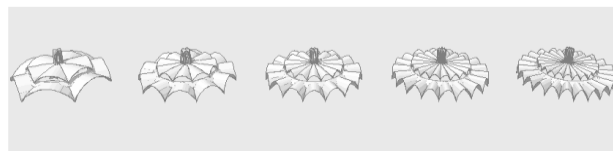


Рисунок 7. Изменение числа сегментов модели
Figure 7. Changing the number of model segments

Так, на рисунке 7 показано изменение числа сегментов модели в меньшую и большую сторону от «базовой модели», которая расположена в центре. Все остальные параметры неизменны.

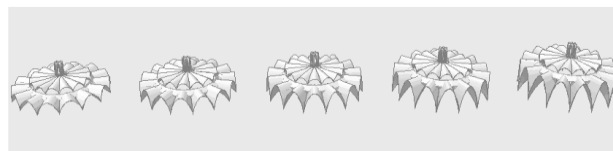


Рисунок 8. Изменение параметра высоты нижних арок модели
Figure 8. Changing the height parameter of the lower arches of the model

На рисунке 8 «базовая модель» расположена слева и «облако изображений» отражает изменение параметра высоты нижних арок при неизменности значений остальных параметров.

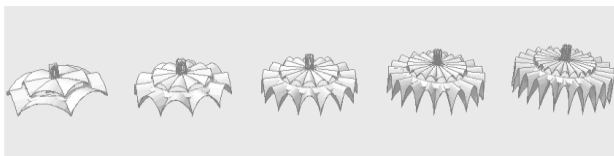


Рисунок 9. Изменение нескольких параметров модели
Figure 9. Changing several model parameters

Несколько более сложная логика — изменение нескольких параметров одновременно. Рисунок 9 становится иллюстрацией того, что получается при совместном изменении параметров, соответствующих рядам на рисунках 7 и 8.

Зоркий глаз архитектора, вероятнее всего, уже в этих экспериментах может заметить интересную идею, понять, в правильном ли направлении движутся его эстетические поиски, соответственно, наметить новые идеи по сочетанию параметров, либо, изменению самого параметрического объекта (базовой модели).

Хотя, очевидно, что на рисунках 7, 8, 9 — представлена лишь малая часть изначально заложенного в модели потенциала. Что можно доказать даже без иллюстраций, а чисто теоретически.

При создании данного параметрического объекта автор статьи параметризовал ее по следующим значениям:

- количество сегментов;
- высота нижней арки;
- радиус нижней арки;
- высота средней арки;
- радиус средней арки;
- высота верхней арки.

Если каждый из параметров может принимать, например, 5 дискретных значений, то метод «полного перебора» даст $5^6=15625$ вариантов. Поэтому вместо «полного перебора» было решено положиться на волю случая и был запрограммирован библиотечный элемент, в котором параметры выбираются из заданного интервала случайным образом. На рисунке 11 изображен эксперимент с различными вставками этого библиотечного элемента в проект.

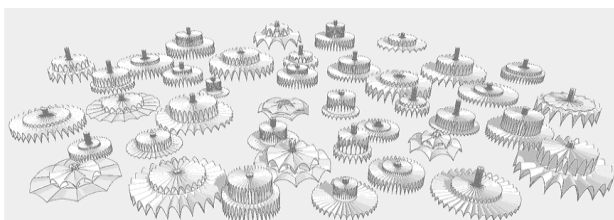


Рисунок 10. Объекты с различными значениями параметров
Figure 10. Objects with different parameter values

Список литературы

1. Divin N.V. BIM by using Revit API and Dynamo. A review // Сетевое издание «AlfaBuild». Санкт-Петербургский политехнический университет Петра Великого. Т. 14., 2020 г. URL: <https://alfabuild.spbstu.ru/en/article/2020.14.4>.
2. Надыршин Н.М. Параметризм как стиль в архитектурном дизайне. // Вестник Оренбургского государственного университета. № 1 (150). 2013.

Список источников

1. Кравченко С. ArchiCAD: открывая заново. Язык GDL: программирование для уникальных задач архитектора [Электронный ресурс]. URL: <https://archi.ru/tech/75350/archicad-otkryvaya-zanovo-yazyk-gdl-programmirovaniye-dlya-unikalnykh-zadach-arkhitekatora>.
2. Документация по API Ренга [Электронный ресурс]. URL: <https://help.rengabim.com/api>.
3. NanoCAD для разработчиков [Электронный ресурс]. URL: <https://nanocad.com/community/developers>.
4. Что такое Dynamo [Электронный ресурс]. URL: https://primer.dynamobim.org/ru/01_Introduction/1-2_what_is_dynamo.html.
5. Система параметрического 3D моделирования Сапфир-3D. URL: <https://lira.land/sapfir>.
6. Шумахер П. Параметризм [Электронный ресурс]. URL: http://www.patrikschumacher.com/Texts/Parametricism_Russian%20text.html.

References

1. Divin, N.V. BIM by using Revit API and Dynamo. A review; 2020; AlfaBuild; Volume 14 Article No 1404.
2. Nadyrshin N.M. Parametrism as a style in architectural design. // Bulletin of the Orenburg State University. № 1 (150). 2013.

Материал передан в редакцию 3 февраля 2024 г.