



Санкт-Петербургский государственный университет
Кафедра системного программирования

Реализация процедурной генерации интерьера комнат в 3D-игре на Unity

Савельева Полина Андреевна, группа 22.Б07-мм

Научный руководитель: к.т.н. Ю. В. Литвинов, старший преподаватель кафедры системного программирования

Консультант: Р. А. Береснёв, студент 3-го курса СПбГУ

Санкт-Петербург
2023

- Ручное заполнение комнат в игре Time Reactor является трудоемким процессом
- Процедурная генерация, PCG
 - ▶ Создание разнообразного наполнения в больших масштабах
 - ▶ Нет необходимости хранить данные
 - ▶ Экономия времени
 - ▶ Повторное использование генераторов
- Процедурно генерируемое помещение
 - ▶ Паттерны
 - ▶ Временные ограничения
 - ▶ Неизменность при повторном посещении

Постановка задачи

Цель: реализация процедурной генерации интерьера комнат в игре Time Reactor на Unity

Задачи:

- Изучение существующих решений
- Реализация библиотеки на .NET для процедурной генерации интерьера игровых комнат и ее публикация на NuGet
- Реализация процедурной генерации интерьера комнат в игре Time Reactor на Unity с использованием разработанной библиотеки
- Проведение апробации среди заинтересованных игроков

Существующие решения

Критерий	Rule-based	Machine learning	Agents	Substitution
Сложные иерархические паттерны	✓	✓	✗	✓
Расстановка в реальном времени	✓	✓	✓	✓
Простота реализации	✓	✗	✓	✗
Место по кол-ву научных публикаций	1	2	3	4

Рис.: Сравнение существующих решений¹

¹Данные о количественном отношении взяты из обзорной статьи «Recent Advances in Procedural Generation of Buildings: From Diversity to Integration»

Основанный на правилах подход

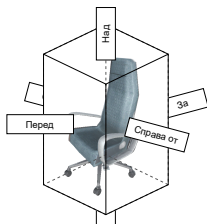


Рис.: Ограничительная рамка у объектов

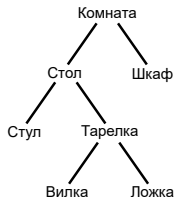
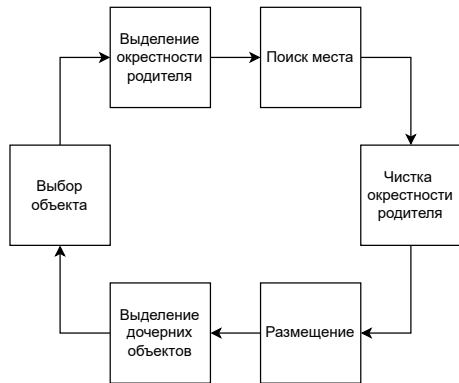


Рис.: Иерархическое представление комнаты

За основу решения взят основанный на правилах подход

- Ограничительная рамка и древовидное представление
- Входные табличные данные
- Рекурсивный алгоритм с возможностью удаления ранее расставленных объектов

Особенности реализации



- Использование F# библиотеки в сторонних проектах на .NET
- Генератор псевдослучайных чисел
- Отдельные входные данные
- Алгоритм генерации без возможности удаления ранее расставленных объектов

Рис.: Последовательность шагов в алгоритме генерации интерьера

Архитектура решения

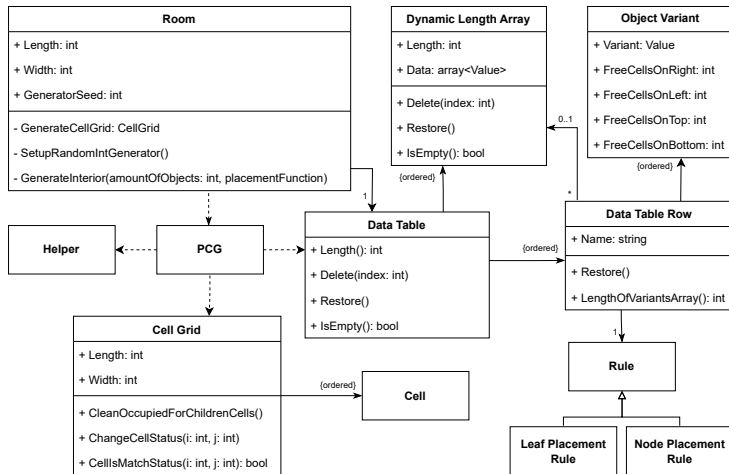


Рис.: UML диаграмма классов реализованной библиотеки

Интеграция в игру



- Логика в Unity Inspector
- Локальная координатная система
- Правила положения размещаемых объектов

Рис.: Процедурно заполненный компьютерный центр

Стороннее использование

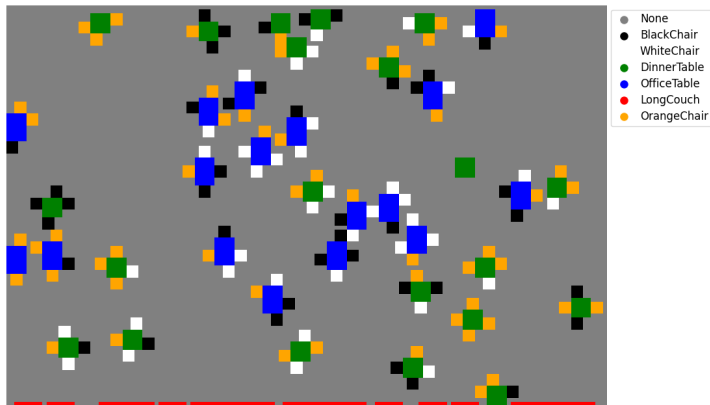


Рис.: Процедурно заполненная двумерная комната

Апробация

- Методика оценки качества продукта System Usability Scale
- В опросе приняли участие 13 человек
- Средний балл по системе SUS составляет 68,1 при среднем показателе в 68 для систем, использующих данную систему оценивания

Вопрос №	Формулировка
1	Интерьеры комнат разнообразны
2	Комнаты выглядят пустоватыми
3	Я могу свободно перемещаться по комнатам
4	Мне не нравится общее визуальное оформление комнат
5	Количество реалистичных комнат превышает количество нереалистичных
6	Время генерации интерьера не приемлемо
7	Иерархические паттерны выдержаны хорошо
8	Генератор не справляется с заполнением больших пространств
9	Мебель расставлена реалистично
10	Я уверен, что в игре не было заполненных вручную комнат

Рис.: Апробационные вопросы

Результаты апробации

Вопрос №	Полностью не согласен	Скорее не согласен	Затрудняюсь ответить	Скорее согласен	Полностью согласен
1	0,0	0,0	0,0	53,9	46,1
2	61,5	15,4	23,1	0,0	0,0
3	0,0	0,0	0,0	61,54	38,46
4	61,5	30,8	7,7	0,0	0,0
5	0,0	0,0	8,3	25,0	66,7
6	92,3	7,7	0,0	0,0	0,0
7	0,0	0,0	0,0	15,4	84,6
8	84,6	15,4	0,0	0,0	0,0
9	0,0	0,0	0,0	69,2	30,8
10	38,5	23,1	23,1	0,0	15,4

Рис.: Процент участников, выбравших предложенные варианты

Заключение

- Рассмотрены существующие решения в области процедурной генерации интерьера комнат
- Реализована библиотека на .NET для генерации интерьера комнат, доступная для скачивания на платформе NuGet² и GitHub³
- Библиотека интегрирована в игру Time Reactor⁴ на Unity Engine при помощи Unity Inspector и C# скриптов
- Проведена апробация среди заинтересованных игроков, которая подтвердила реалистичность генерируемых интерьеров и общую эффективность системы

²NuGet: <https://www.nuget.org/packages/RoomInteriorGenerator>

³GitHub: <https://github.com/PolinaSaveljeva/RoomInteriorGenerator>

⁴Time Reactor: <https://github.com/RuslanBeresnev/Time-Reactor-Game>