

Программный комплекс визуализации и моделирования на основе BPMN нотации

Н.Д. Баданина^{1,А,В}, В.А. Судаков^{2,В,С}, Н.А. Яшин^{3,С}

^А Московский авиационный институт

(национальный исследовательский университет)

^В Институт прикладной математики имени М. В. Келдыша РАН (ИПМ РАН)

^С Российский экономический университет имени Г.В. Плеханова

¹ ORCID: 0000-0002-5301-1526, natashabadanina99@gmail.com

² ORCID: 0000-0002-1658-1941, sudakov@ws-dss.com

³ ORCID: 0000-0002-9693-0063, niknlo007@gmail.com

Аннотация

Статья посвящена созданию программного обеспечения визуализации сложных процессов. Поскольку в научных организациях и бизнес-структурах, существует запрос на анализ и оптимизацию осуществляемых процессов, то задача их эффективного моделирования и визуализации с возможностью интерактивного взаимодействия актуальна. Имеющиеся на рынке программные решения ориентированы либо на статическое представление предметной области, без возможности проведения вычислительных экспериментов и оптимизации решений на их основе, либо на использование не стандартизированных представлений. Разработана программа, которая имеет возможность анимировать визуализацию загруженной BPMN схемы, а также содержит в себе функционал по моделированию процесса, описанного схемой, с возможностью гибкой настройки параметров. Разработанная программа визуализации отображает результат в браузере, причем поверх BPMN модели накладывается слой с анимацией прохождения задействованных ресурсов по схеме. Визуализация позволяет уменьшать вероятность ошибок при моделировании сложных систем. В работе также уделяется внимание аспектам оптимизации, сбору статистик для последующего анализа, моделированию процесса на заданное количество единиц времени. Этот функционал включен в разработанную программу, может быть использован как в консольном режиме, так и в веб-браузере.

Ключевые слова: визуализация алгоритмов, визуализация данных, BPMN нотация, BPMN модель, BPMN, прикладная программа, моделирование процессов, бизнес-процессы.

1. Введение

Нотация BPMN стала универсальным стандартом для представления в графическом виде множества разнонаправленных процессов, протекающих в коммерческих и научных организациях различных секторов. Например, с использованием этого подхода можно описать как систему бронирования авиабилетов, так и процесс получения докторской степени. Однако BPMN обеспечивает статичное представление о процессе, что может ухудшить понимание схемы пользователем, не дает представление о времени и ресурсо-затратах. В данной работе рассматривается оригинальный подход для визуализации бизнес-процессов с использованием нотации BPMN. Была разработана программа для построения имитационной модели на основе составленной BPMN схемы. Разработана критерии оценки результатов моделирования, которые включают подсчет среднего времени нахождения заявок в системе; среднего времени занятости ресурсов.

2. Обзор существующих нотаций

Рассмотрим примеры существующих нотаций, используемых для описания бизнес-процессов. IDEF0 – методология моделирования, применимая для построения диаграмм, ведения документации [1]. Результатом применения IDEF0 является модель, состоящая из иерархического ряда диаграмм с сопровождающим пояснительным текстом, иллюстрирующими графическими изображениями. Двумя основными компонентами моделирования диаграммы являются функции (представленные именованными блоками) и объекты данных, которые взаимосвязывают эти функции.

Как язык моделирования функций, IDEF0 обладает следующими преимуществами:

- последовательный и простой язык, позволяющий кратко описать бизнес-процесс;
- используется ВВС США и другими государственными учреждениями, а также частной промышленностью.

Недостатки IDEF0:

- нельзя увидеть алгоритм выполнения бизнес-процессов;
- требует определенной подготовки для разработки и чтения нотации;
- отсутствие формальной теоретической модели, а также отсутствия средств для описания артефактов и потоков информации;
- не отражает реакцию участников процесса на события внешней среды. В большинстве случаев из-за изменений процессов извне следует либо задержка, либо отмена процесса на некотором этапе, но не откат, как это происходит в нотации IDEF0. Именно поэтому нотация IDEF0 кажется более приемлемой для описания процессов технологических, а не бизнес;
- при добавлении в IDEF0 параллельность в выполнении, все параллельные функции будут при этом выполняться одновременно

IDEF0 применяется для верхнеуровневого моделирования бизнес-процессов.

Другой альтернативной нотацией является EPC [2]. Нотация используется на низших уровнях описания бизнес-модели, когда стоит задача описать подробный ход выполнения бизнес-процесса.

Преимущества – позволяет очень детально и точно описать выполнение бизнес-процесса, показать на диаграмме в графическом виде всех исполнителей, все используемые объекты. Плюсом EPC-диаграмм является тот факт, что, как и на диаграммах IDEF0, на них можно указать входные и выходные данные каждой функции, проследить логику передвижения входных и выходных данных от блока к блоку. К тому же, в отличие от IDEF0, есть возможность распараллеливать процесс, направляя его только по одной из альтернативных веток.

Недостатки – обладает очень широким набором графических элементов, что может быть сложным для понимания, по сравнению с другими нотациями. Для разработки процессов в этой нотации и их чтения требуется предварительная подготовка сотрудников. Диаграммы EPC не описывают расписание исполнения, опускают вопросы синхронизации ветвей одного процесса между собой и с другими, внешними процессами.

Для описания бизнес-процессов можно использовать различные модели. Одной из них являются UML-диаграммы [3]. Данный инструмент может быть использован для описания бизнес-процесса, однако для дальнейшей работы он малопригоден, так как имеется огромное количество различных форматов UML, и нету единой спецификации. Таким образом при создании приложения на основе UML будет необходимо учитывать возможность получения разных форматов UML, чего можно избежать, работая с другими форматами описания бизнес-процессов.

Диаграммы потоков данных (DFD) [4] были разработаны еще в семидесятых годах, и их целью является представление потоков данных, а не действий. Хотя аналитики бизнес-процессов одобряют диаграммы потоков данных, они в целом согласны с тем, что данный метод устарел и имеет одно большое ограничение: он фокусируется на ин-

формации, а не на действиях. Диаграммы потоков данных по определению ориентированы на данные и не обеспечивают четкого способа включения всех заинтересованных сторон в процесс.

Функциональные блок-схемы (FBD) [5], возможно, существуют уже несколько десятилетий, но они все еще используются в картографировании бизнес-процессов. Их внимание сосредоточено на порядке выполнения задач или функций в последовательности упорядоченных блоков. Каждый функциональный блок можно разбить на отдельную диаграмму, показывающую подзадачи в каждом функциональном блоке. Конечно, это приводит к большому количеству диаграмм, представляющих один процесс, но их легко сопоставить с диаграммой первого уровня. Некоторые предприятия предпочитают FFD, потому что, несмотря на необходимость нескольких диаграмм, им относительно легко следовать, даже если процесс довольно сложный.

BPMN (Business Process Modeling Notation) модели являются инструментом для понимания комплексных многоуровневых бизнес-процессов. При усложнении некоторого процесса возникает необходимость расширения документации, базы знаний, инструкции, в которых описывается связь этого процесса с прочими существующими составляющими. Решением проблемы возрастающей сложности для понимания лицом, принимающим решение (ЛПР) может стать построение BPMN модели, описывающей текущие процессы. Изучение графической визуализации процессов позволяет ЛПР обнаружить неточности, заикливание процессов, потенциально проблематичные участки, также собрать данные о задействованных ресурсах для последующего анализа и оптимизации.

Первая версия спецификации BPMN модели была представлена в 2006 году, затем был выпущен ряд обновлений, последнее из которых опубликовано в январе 2011 года. [6] BPEL (Business Process Execution Language) – стандарт проектирования и исполнения бизнес-процессов, схожий по своей структуре с BPMN моделями. Спецификация BPMN была введена с целью преодолеть разрыв между бизнес-анализом и техническим анализом, предоставив стандартизованную графическую нотацию для описанных в BPEL процессов [7]. В парадигме BPMN и BPEL используются различающиеся элементы для описания схожих бизнес-процессов, что ведет к дальнейшим несоответствиям. BPMN и BPEL применяются на различных стадиях жизненного цикла BPM. В работе [7] авторы обращают внимание на различия этих двух подходов и представляют ряд аргументов, которые показывают преимущество нотации BPMN при моделировании бизнес-процессов. Трудности при переносе моделей из BPMN в BPEL и в обратном порядке были обнаружены и преодолены в статье [8] с помощью создания системы BPMN2BPEL для автоматической конвертации. Таким образом, при необходимости, схемы в более продвинутой спецификации BPMN могут быть упрощены до других BPM форматов.

Спецификация BPMN может быть внедрена на предприятиях различных отраслей, масштаба, решающих задачи в отличных друг от друга областях. Так, в работе [9] была создана структура ситуационного центра для управления нештатными ситуациями в нефтяной отрасли. В работе [10] описана разработка сквозного процесса реализации государственных/муниципальных услуг в электронном виде с использованием нотации BPMN. Также опубликованы схемы реализации традиционной бизнес-модели займа на краудинвестинговой платформе, купли-продаж ценных бумаг, токенизации прав и их модификации. [11] Таким образом, делается вывод, что метод проектирования с использованием BPMN универсален и распространен, причем подход может быть применен и к описанию бизнес-процессов, и в качестве основы для программного обеспечения, предназначенного для моделирования.

По итогам сравнительного анализа была составлена таблица, в которой знаком «+» отмечено наличие свойства. По итогам сравнения была выбрана нотация BPMN, как наиболее универсальная и распространенная.

Табл. 1. Сравнительный анализ нотаций.

	BPMN	EPC	IDEFo	UML	DFD	FBD
Удобство понимания составленной схемы	+	+	-	+	+	+
Быстрое обучение элементам нотации	-	-	-	-	+	+
Универсальность	+	+	-	+	-	+
Возможность описания комплексных процессов	+	-	+	+	-	-
Поддержка другим ПО	+	-	-	+	-	-

3. Обзор существующих разработок

Существующие программные решения ориентированы либо на статическое представление предметной области, без возможности проведения вычислительных экспериментов и оптимизации решений на их основе, либо на использование не стандартизированных представлений, которые несмотря на свою эффективность не всегда понятны пользователям далеким от программирования и математики [12].

На текущий момент уже существуют площадки для анализа бизнес-процессов на основе BPMN моделей. Каждая из площадок может предложить различный функционал для пользователей.

Samunda – это платформа для автоматизации рабочих процессов и принятия решений с открытым исходным кодом [13]. Платформа Samunda поставляется с инструментами для создания моделей рабочих процессов и решений, эксплуатации развернутых моделей в производстве и предоставления пользователям возможности выполнять назначенные им задачи рабочих процессов. Она разработана на Java и выпущена как программное обеспечение с открытым исходным кодом в соответствии с условиями лицензии Apache [14].

Данная платформа предоставляет услуги по внедрению данной платформы на предприятие и дальнейшее сопровождение. Так же возможно бесплатное внедрение с помощью открытого исходного кода, однако в таком случае Samunda не предоставляет никакой помощи по внедрению и сопровождению продукта.

Также на данной платформе присутствуют инструменты по анализу эффективности бизнес-процесса, как, например, тепловая карта, показывающая загруженность определенных узлов (рисунок 1), однако данная платформа не предоставляет никаких возможностей по предоставлению визуализации.

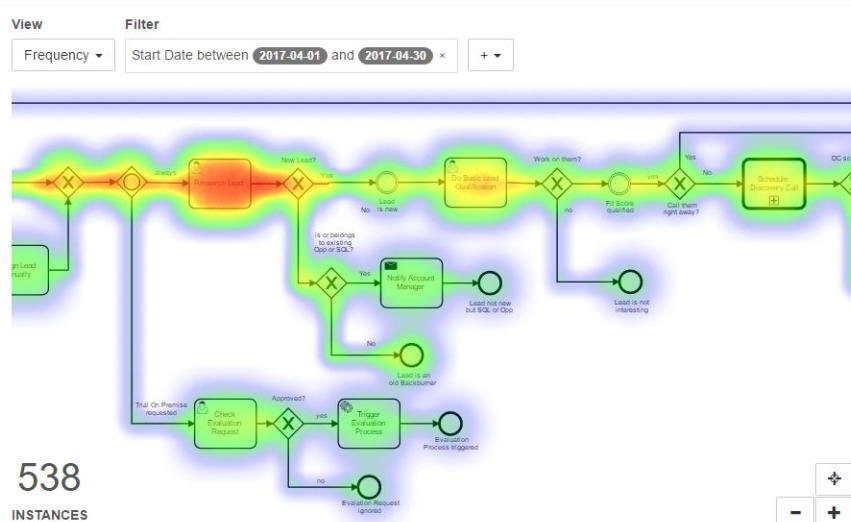


Рисунок 1 Тепловая карта загрузки узлов в продукте Camunda

Преимущества:

- Большое разнообразие предлагаемых инструментов
- Наличие открытого исходного кода
- Помощь по внедрению продукта

Недостатки:

- Не позволяет проводить оптимизацию
- Очень объемный продукт, требующий много ресурсов для внедрения
- Не является отечественным ПО.
- Отсутствует возможность анимирования BPMN схемы

ELMA365 – Low-code BPM-система, которая выстраивает взаимодействие сотрудников из разных департаментов в сеть взаимосвязанных бизнес-процессов [15]. Так, платформа позволяет качественно и легко получить быстрый результат без лишней бюрократии. Стратегические цели компании достигаются через моделирование, исполнение, контроль и непрерывное улучшение бизнес-процессов.

Данная платформа предоставляет услуги по внедрению данной платформы на предприятие и дальнейшее сопровождение. Так же платформа может предоставить бесплатный пробный период, однако, в отличие от предыдущего сервиса, не предоставляет бесплатного открытого кода для самостоятельного внедрения продукта.

На данной платформе присутствуют различные инструменты по контролю за бизнес-процессом, однако как таковые инструменты автоматической оптимизации отсутствуют. Предлагаемое улучшение процессов строится вручную, на основе конкретного бизнеса, что является преимуществом в плане точности возможных улучшений, но недостатком в плане скорости предложения и требуемых ресурсов для расчетов оптимизации.

Так же продукт предлагает готовые коннекторы для интеграции с различными сервисами, такими как 1С, базы данных MS SQL, Oracle, PostgreSQL. Удобные модули интеграции с веб-сервисами REST, SOAP. ELMA подгружает данные из внешних систем, операции с внутренними выполняет автоматически.

Преимущества:

- Большое разнообразие предлагаемых инструментов
- Удобный пользовательский интерфейс
- Использование Low-code подхода
- Продукт входит в реестр российского ПО
- Помощь по внедрению продукта

Недостатки:

- Не позволяет проводить оптимизацию
- Очень объемный продукт, требующий много ресурсов для внедрения
- Отсутствует возможность анимирования BPMN схемы

BPMN.Studio – Бесплатный онлайн-сервис для моделирования бизнес-процессов [16]. В онлайн-конструкторе бизнес-процессов можно быстро смоделировать процесс или группу процессов, используя популярную нотацию моделирования бизнес-процессов – BPMN 2.0.

Данный продукт позволяет создавать схемы BPMN моделей. Кроме этого, он позволяет импортировать их в различных форматах, для документов MS Office или PDF, а также для печати. Также, данный веб-сервис не предоставляет никаких возможностей.

Так, в данном сервисе нету анимированной визуализации, а также нету возможностей для расширения данного продукта в виде добавления нового функционала сторонними пользователями.

Преимущества:

- Удобный инструмент для создания BPMN схем
- Веб-приложение

Недостатки:

- Не позволяет проводить оптимизацию
- Отсутствие возможностей по расширению функционала
- Отсутствует возможность анимирования BPMN схемы

ПИТЕРСОФТ: BPM – это полный набор инструментов для автоматизации управления предприятием на платформе 1С: Предприятие 8 [17]. В зависимости от поставленных задач могут использоваться технологии процессного управления, адаптивный кейс-менеджмент или проектное управление.

Данная платформа позволяет создавать модели бизнес-процесса - фиксированная цепочка задач, последовательное выполнение которых приводит к достижению необходимого результата. Настраиваются модели в визуальном редакторе с помощью механизма настройки бизнес-процессов. Для любой точки процесса можно указать исполнителя и время, за которое он должен выполнить данную задачу. После настройки модель бизнес-процесса можно запускать неограниченное количество раз.

На данной платформе присутствуют различные инструменты по контролю за бизнес-процессами и их внедрению, однако инструменты оптимизации отсутствуют. Есть только инструменты анализа, однако улучшения не предлагаются ни в ручном, ни в автоматическом варианте.

Преимущества:

- Большое разнообразие предлагаемых инструментов
- Удобный пользовательский интерфейс
- Продукт входит в реестр российского ПО
- Помощь по внедрению продукта

Недостатки:

- Нет инструментов по оптимизации процессов
- Объемный продукт, требующий много ресурсов для внедрения
- Отсутствует возможность анимирования BPMN схемы

Оптимизации бизнес-процессов требуют либо создания собственного проприетарного решения, либо внедрения комплексной системы, требующей большого количества ресурсов.

Платформа Bizagi близка по функционалу создаваемой разработке, но не может быть детально исследована, так как недоступна на территории России. Судя по описанию, данная система ориентирована на автоматизацию текущих бизнес-процессов, а не на анализ критериев оценки результатов вычислительных экспериментов, с целью дальнейшей оптимизации.

Общая модель данных продуктов похожа друг на друга, но имеет свои достоинства и недостатки. Каждая из площадок позволяет работать с бизнес-процессами, однако данные площадки предоставляют комплексные решения по глубокому ручному анализу бизнес-процессов в компании. Это подразумевает долгую и объемную работу по аудиту и предоставлению возможных решений, что требует больших затрат ресурсов.

В большом количестве ПО используется цвет, как способ повышения качества визуализации. Действительно, цветовые акценты способны упростить понимание некоторых типов визуализации, например графиков. Однако, в нотации BPMN нет спецификации цветов, у блоков черные границы, белый фон. Различие по цветам может отвлекать пользователей при моделировании высоко составного процесса, как это происходит в нотации EPC. Однако, для целей кастомизации, в разработанном ПО присутствует возможность настройки изображения для анимации. К примеру, для моделирования завода может быть использовано изображение детали, а при моделировании процесса обработки заявки в кол-центре – изображение документа. Таким образом, ЛПР проще ориентироваться в схеме, понятен задействованный ресурс, визуализации становится более наглядной, чем при использовании классических элементов нотации.

4. Метод решения

При решении задачи оптимизации бизнес-процесса недостаточно опираться лишь на BPMN модель, необходимо тестирование и проверка гипотез на практике. Такая проверка в условиях реального бизнеса может быть сопряжена с существенными затратами времени, более того, в ходе тестирования ограничивается возможность экспериментов с различными настройками в узлах BPMN модели. В таком случае необходимо применить моделирование – построить виртуальную модели и провести исследование на ней.

Преимущества моделирования заключаются в скорости выполнения симуляции процессов; в возможности изменять параметры таким образом, что позволяет проверять и исследовать различные гипотезы, причем эти изменения не окажут влияния на реальный процесс. На основе множества вычислительных экспериментов необходимо выбрать наилучшую комбинацию значений параметров, основываясь на критериях оценки результатов моделирования, после чего, модель может быть применена для модификации бизнес-процессов.

Для решения описанной задачи было разработано программное обеспечение, позволяющее реализовать имитационную модель на основе BPMN модели, визуализировать поток элементов в данной модели с возможностью изменять параметры модели для ее оптимизации.

При анализе конкретного бизнес-процесса необходимо первоначально составить его описание с помощью нотации BPMN. Для корректного моделирования разработанной программой, составленная модель должна отвечать некоторым правилам, не входящим в нотацию. Так, у некоторых блоков должно присутствовать описание определенных параметров, например название функции и ее параметры для реализации генератора входного потока.

Составленная модель передается в разработанную программу, в которой может быть реализована её дальнейшая настройка. Указывается в каком виде должна проходить симуляция – с использованием веб-интерфейса или в режиме консоли. Веб-интерфейс нагляден, прост для понимания, в нем присутствует возможность настройки параметров симуляции. Консольный режим необходим для ускорения работы симуляции, что сокращает время работы при многократных запусках модели в случае подбора ее оптимальных параметров каким-либо способом.

После окончания процесса симуляции, из модели можно получить результирующие параметры. Выгружается статистика по наполняемости конкретных узлов, а также

по количеству сгенерированных и высвобожденных сущностей в каждом отдельном блоке.

Для узла-генератора сущностей выводится статистика по общему количеству сгенерированных сущностей. Для финального узла, высвобождающего сущности из модели, выводится количество высвобожденных сущностей, общее время нахождения всех сущностей внутри и среднее время одной сущности. Такая статистика выводится отдельно для каждого такого блока, что позволяет сравнить между собой разные «пути» сущностей. Например, при моделировании процесса приема заявки в некотором ведомстве, заявка может быть отклонена в самом начале, а может быть принята и обработана. Схожие значения по количеству отклоненных и обработанных заявок могут указывать на несовершенство процесса отбора, на проблемы в системе или в бизнес-процессе в целом.

Описанный в предыдущем пункте алгоритм реализован в программном комплексе, предназначенном для визуализации бизнес-процессов на основе BPMN модели. Программа была выполнена на языках программирования Python3 и JavaScript с использованием HTML в среде разработки PyCharm. В качестве вспомогательных библиотек была использована библиотека PixiJS (пакет для визуализации динамических интерфейсов в веб-страницах). Выбор языка программирования Python3 обусловлен его гибкостью при работе и с математическим аппаратом, и с веб-интерфейсом. JavaScript и язык разметки HTML были использованы для анимации процессов, описанных в схеме BPMN.

Функционально комплекс разделен на четыре части (рис. 2):

1. Блоки, отвечающие за функционал конкретного узла бизнес-модели. Например, блок-генератор или блок, отвечающий за эмуляцию «работы». Такие блоки имеют общий интерфейс и программу легко дополнить новыми блоками с другим функционалом.

2. Модуль ядра, который отвечает за создание конкретной модели из блоков, реализацию совместной работы этих блоков, а также за тип визуализации работы модели.

3. Модуль парсинга BPMN файлов, который получает данные из файла с описанием модели и переводит их в удобный для ядра формат.

4. Модуль веб-приложения, который используется при визуализации бизнес-процесса с помощью веб-страницы, а не с использованием консоли.

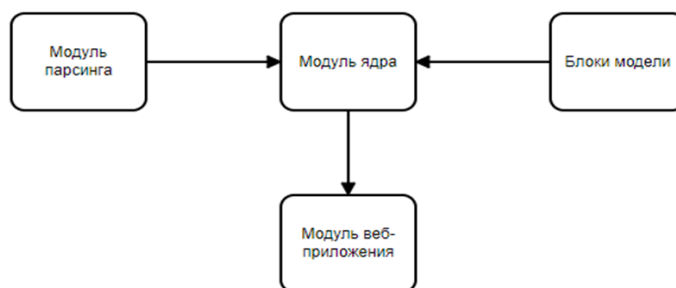


Рисунок 2. Структурная схема программного комплекса

На рисунке 3 приведена схема работы программного комплекса. Первый шагом файл направляется в парсеры, которые отфильтровывают избыточную информацию, содержащуюся в BPMN нотации, преобразовывают составляющие в конкретные блоки. После преобразования, эти блоки попадают в два отдельных независимых ядра – логическое и визуальное.

Логическое ядро отвечает за моделирование всех процессов, происходящих в симуляции. Реализованы следующие блоки логического ядра:

- Start – блок, отвечающий за генерацию сущностей. К нему подсоединен модуль start_functions, который содержит в себе различные функции способов генерации сущностей. Например, constant – генерация постоянного количества сущностей каждую

единицу времени или random – генерация случайного числа сущностей в заданном интервале каждую единицу времени.

- Delay – блок, отвечающий за симуляцию работы сущностей внутри блока. Он останавливает продвижение конкретной сущности на определенное время, чтобы симулировать какое-либо действие, производимое над сущностью в конкретном месте процесса. Также имеет отдельный модуль функций delay_functions, в котором прописаны различные способы задания времени работы в блоке. Для добавления новой функции в этом или предыдущем блоке, достаточно добавить саму функцию в соответствующий модуль и скорректировать BPMN файл.

- End – блок, отвечающий за высвобождение сущностей, которые достигли конечной точки в данном процессе. Собирает различную статистику по среднему времени сущностей внутри.

- Selection – блок, отвечающий за логическую развилку на пути движения сущности.

- Parallel_open и parallel_close – блоки, отвечающие за разделение одной сущности на несколько частей и последующее соединение этих сущностей в одну.

Визуальное ядро отвечает за вывод информации пользователю. Может быть консольным или веб-приложением. Имеет отдельные визуальные блоки, каждый из которых присоединен к соответствующему логическому блоку. Все визуальные блоки имеют общий интерфейс, поэтому логический блок не меняет своего поведения при смене типа визуального ядра.

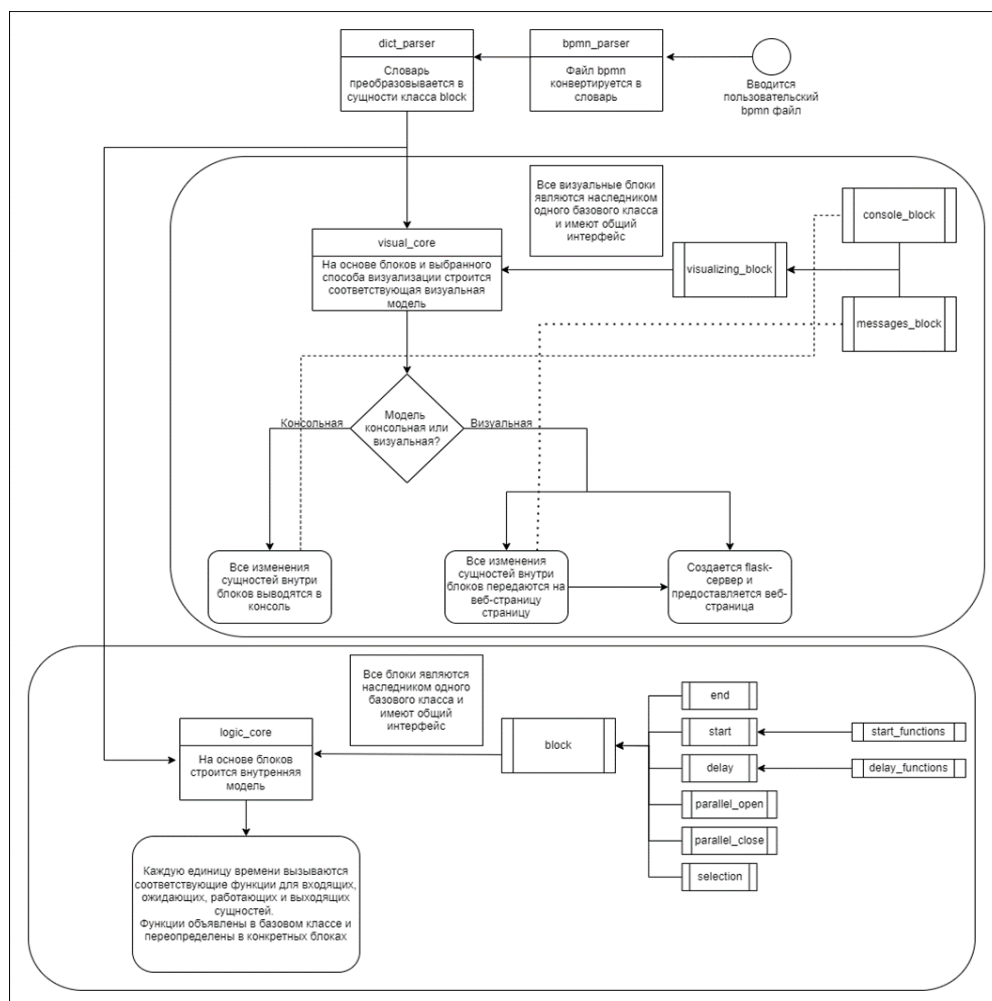


Рисунок 3. Схема работы программного комплекса.

В предоставленном программном комплексе уже реализованы некоторые функции для моделирования псевдослучайных задержек и входных потоков в блоках `delay` и `start` соответственно.

Оба блока имеют функцию `random`, которая при создании принимает два значения – минимальное и максимальное, и при каждом обращении возвращает случайное число в заданном интервале. Для блока `delay` это число означает, сколько времени задержится в конкретном блоке конкретная сущность, действия которой привели к вызову функции. В блоке `start` данная функция вызывается каждую единицу времени, а получаемое значение показывает, сколько сущностей будет сгенерировано в блоке в конкретный момент времени.

Также для блока `Delay` написана функция, которая позволяет внедрить в модель концепцию «производительности» и выделяемых ресурсов – линейная функция зависимости времени от ресурсов. Данная функция определяется пятью значениями – минимальное и максимальное значения ресурсов, которые могут быть в блоке (*Rlow* и *Rhigh*), значения времени, которое сущность будет проводить внутри блока при минимальном и максимальном значениях ресурса (*Tlow* и *Thigh*) и количество ресурсов *Rcurr*, которое выделено в текущем блоке в текущей модели.

Текущее количество ресурсов является величиной, которую можно изменять, меняя таким образом «производительность» блока и влияя на «производительность» модели в целом, что необходимо при проведении исследований и оптимизации. Для вычисления времени задержки в такой функции сначала рассчитывается коэффициент *k* прямой, которая проходит через точки минимальной и максимальной производительности (формула 1), а потом на основе этого коэффициента и формулы прямой по двум точкам, рассчитывается время (формула 2)

$$k = \frac{Tlow - Thigh}{Rlow - Rhigh} \quad (1)$$

$$Tcurr = k * Rcurr + Thigh - k * Rhigh; \quad k = \frac{Tlow - Thigh}{Rlow - Rhigh} \quad (2)$$

Также есть отдельная функция для блока `Start`, которая выдает количество сущностей, которые нужно сгенерировать в блоке в конкретный момент времени на основе изначально заданной последовательности *Seq*, имеющий длину *L*. Для определения количества сгенерированных сущностей *N* в момент времени *t* используется формула 3

$$N = Seq_j; j = \begin{cases} t; & t < L \\ t \bmod L; & t \geq L \end{cases} \quad (3)$$

С помощью такой функции можно моделировать неравномерное поступление сущностей в разное время. Например, для обозначения разного количества заявок в разные дни недели, можно принять одну единицу времени модели за день и предоставить значение *Seq* = [10,12,13,11,12,40,35]. Таким образом, будет показана более высокая нагрузка на систему в выходные дни. Также, с помощью *Seq* = [1, 0, 0] можно показать, что сущности генерируются в модели только раз в три единицы времени.

Однако предоставленных функций может быть недостаточно, и пользователю может быть необходимо использовать свою собственную функцию. Например, добавить функцию, использующую нормальное распределение. Это легко реализовать – для этого нужно лишь добавить свою собственную функцию в файл `start_functions.py` или `delay_functions.py` для блоков `Start` и `Delay` соответственно. Например, для добавления функции нормального распределения в блок `Delay`, нужно написать в соответствующем файле код на рисунке 4.

```
def normal(mu, sigma):
    def toRet(time):
        return int(np.random.normal(mu, sigma))
    return toRet
```

Рисунок 4. Код генератора функции для блока Delay

Данная функция генерирует новую функцию на основе передаваемых параметров распределения. Данная функция должна принимать значение текущего времени модели, даже если не использует его. В примере используется генератор нормального распределения из сторонней библиотеки numpy. Также сгенерированная функция должна возвращать целочисленные значения.

После добавления данной функции, для её применения нужно лишь обновить любой блок в модели, изменив название используемой им функции на normal и изменив параметры на соответствующие.

На рисунке 5 приведен пользовательский интерфейс разработанного программного комплекса.

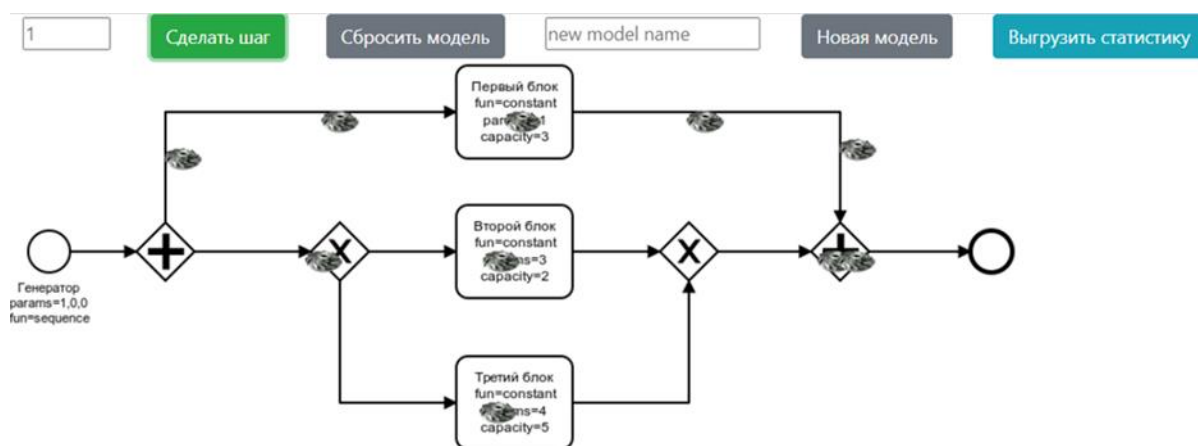


Рисунок 5. Пользовательский интерфейс программного комплекса.

Окно приложения представляет собой два блока – блок настройки модели и непосредственно сама модель. Модель представлена в виде изображения формата .png, поверх которого отображаются перемещающиеся по модели детали. В верхней части находятся кнопки, используемые для настройки, оптимизации и получения статистики.

- Кнопка «Сделать шаг» позволяет запустить модель на одну единицу времени. Так же данная кнопка связана с полем ввода до нее – в него можно ввести целое неотрицательное целое число, чтобы запустить модель сразу на введенное количество шагов.

- Кнопка «Сбросить модель» позволяет полностью очистить модель от всех сущностей внутри и сбросить внутреннее время модели до начального значения. Таким образом модель и вся собранная по ней статистика сбросятся до первоначального значения.

- Кнопка «Новая модель» позволяет заменить модель на новую, если такая существует в по заданному в программе пути. Путь до папки с моделями формата BPMN необходимо указать при запуске модели внутри кода.

- Кнопка «выгрузить статистику» позволяет загрузить файл со статистикой по блокам в формате json. Этот файл может быть использован для проведения анализа эффективности модели.

В качестве примера визуализации используется упрощенная схема завода по покраске деталей (рис 6).

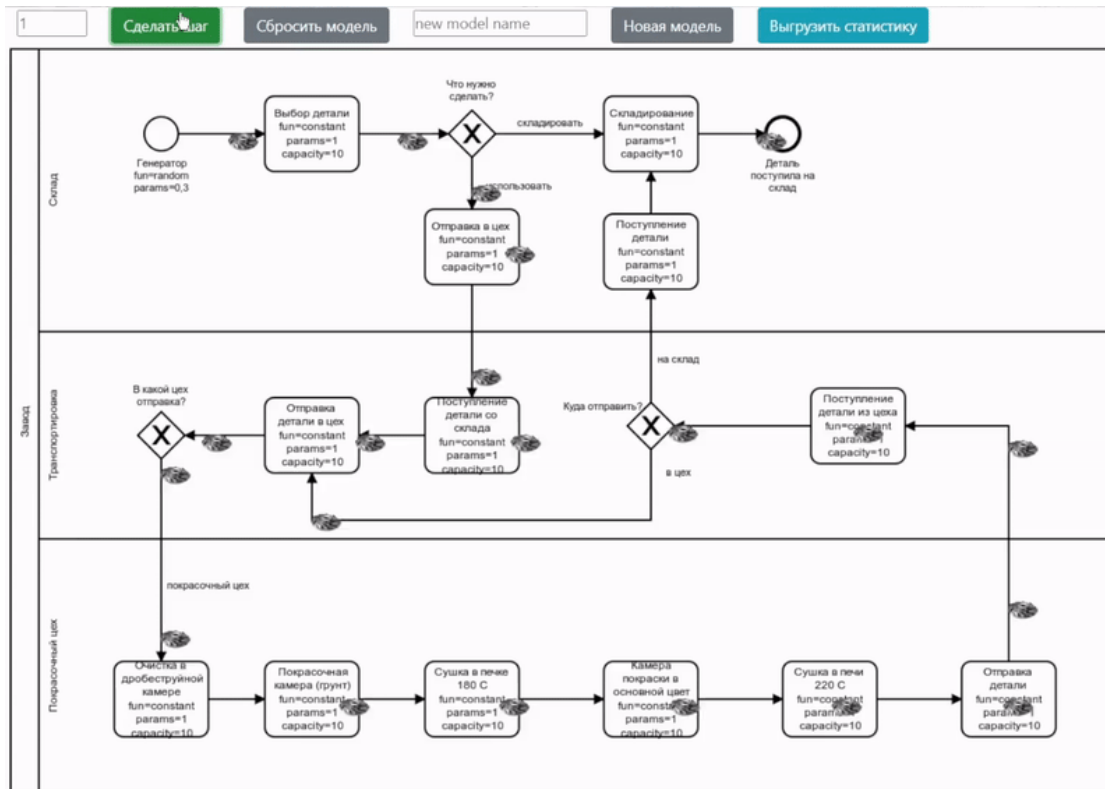


Рисунок 6. Визуализация процесса покраски деталей.

На представленном на рисунке 6 фрагменте анимации изображена смоделированная схема завода, где основным ресурсом является деталь, которая может быть складирована или отправлена на покраску. Визуализация позволяет уменьшать вероятность ошибок при моделировании сложных систем. Из приведенной в пример схемы пользователь может сделать вывод о том, что отсутствует обработка случая, когда деталь отправили в покрасочный цех по ошибке. Таким образом, для оптимизации производства, можно добавить в блок XOR с вопросом «В какой цех отправка?» еще одно ответвление – отправка повторно на склад.

Для исследования результатов работы модели с целью последующей оптимизации, анализа процессов, реализован функционал выгрузки статистики. На рисунке 7 можно увидеть статистику для трех разных блоков после запуска модели, изображенной на рисунке 5, на 200 единиц времени.

```

"Выбор детали": {
  "filledAmountDistribution": {
    "1": 200
  },
  "meanTimeInside": 1.0,
  "queueDistribution": {},
  "totalPassed": 199.0,
  "totalTime": 199.0,
  "type": "Delay"
},
"Генератор": {
  "totalGenerated": 200,
  "type": "Start"
},
"Деталь поступила на склад": {
  "countDeleted": 184.0,
  "meanEntityTime": 9.668478260869565,
  "totalEndTime": 1779.0,
  "type": "End"
}

```

Рисунок 7. Пример полученной статистики модели.

В статистике показаны результаты трех блоков – Start, Delay и End. В самой модели всего по одному блоку Start и End, но количество созданных и высвобожденных сущностей различается. Это говорит о том, что на момент снятия статистики 16 сущностей находилось внутри модели.

Сущности проходят по различным блокам и находятся там определенное время. Однако, разные сущности могут проходить по разным блокам, и в итоге суммарное время нахождения сущности во всех блоках, через которые она прошла, может быть разным для разных сущностей. Поэтому необходимо считать среднее время нахождения внутри модели T_{mean} для всех сущностей. Для этого необходимо считать количество всех сущностей, полностью прошедших через модель N_{total} и суммарное время, которые эти сущности провели внутри модели. Данное время можно вычислить по формуле 4 на основе времени нахождения каждой такой сущностей внутри модели ($T_{entity_j}, j = 1..N_{total}$)

$$T_{mean} = \left(\sum_{j=1}^{N_{total}} T_{entity_j} \right) / N_{total} \quad (4)$$

Данная статистика рассчитывается и для каждого блока отдельно, и сразу предоставляется в поле meanTimeInside, но для расчета глобальной статистики, предоставляются и отдельно значения totalPassed и totalTime.

Также выбранный блок Delay «Выбор детали» является блоком, следующим за генератором, поэтому через него проходят все сущности. Однако вышло из него всего 199 сущностей вместо 200 сгенерированных, что говорит о наличии одной сущности внутри блока. Пустое поле queueDistribution говорит об отсутствии очередей в блоке, а поле filledAmountDistribution показывает, что 200 единиц времени в блоке была в работе одна деталь.

Модель была запущена ещё на 200 шагов для анализа изменений. Результаты статистики после дополнительных 200 шагов показаны на рисунке 8:

```
"Выбор детали": {
  "filledAmountDistribution": {
    "1": 400
  },
  "meanTimeInside": 1.0,
  "queueDistribution": {},
  "totalPassed": 399.0,
  "totalTime": 399.0,
  "type": "Delay"
},
"Генератор": {
  "totalGenerated": 400,
  "type": "Start"
},
"Деталь поступила на склад": {
  "countDeleted": 387.0,
  "meanEntityTime": 11.426356589147288,
  "totalEndTime": 4422.0,
  "type": "End"
}
```

Рисунок 8. Изменения в статистике после дополнительного запуска моделирования

Видно, что генератор создал еще 200 сущностей. Однако, в соответствии с количеством удаленных блоком типа End сущностей, внутри модели еще находится 13 сущностей. Среднее время сущности внутри модели увеличилось, что может говорить о том, что внутри модели существует некая очередь, которая лишь увеличивается со временем, либо может являться допустимой погрешностью в связи с наличием разилок со случайным выбором в модели. Для более точного исследования данного феномена следует запускать модель еще несколько раз, обнуляя и не обнуляя её, собирая более обширную статистику для выводов.

Стоит отметить, что моменты времени свершения событий, которые указаны в статистике можно рассматривать, как непрерывные случайные величины в силу того, что в результате испытания (при запуске программы), эти переменные принимают одно из случайных возможных значений интервала числовой оси. Обозначим случайную величину за X . Тогда $X = f(\omega)$ – функция, заданная в пространстве элементарных событий. Наиболее исчерпывающим способом определения случайной величины является закон распределения. Непрерывная случайная величина распределена по нормальному закону, если её плотность вероятности имеет вид:

$$\varphi_N(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-a)^2}{2\sigma^2}} \quad (5)$$

Причем a, σ – параметры нормального закона распределения. Функция распределения случайной величины X , распределенной по нормальному закону, выражается через функцию Лапласа по следующей формуле:

$$F_N(x) = \frac{1}{2} + \frac{1}{2} \Phi\left(\frac{x-a}{\sigma}\right) \quad (6)$$

Для исследования изменений в модели и влияния этих изменений на результаты, необходимо менять модель в процессе использования. В программном комплексе присутствует такой функционал. Так, можно поменять параметры функции и вместимость конкретного блока, как показано на рисунке 9. Задержка внутри блока изменена с одной единицы времени до трех, а вместимость уменьшена с десяти до одной единицы.

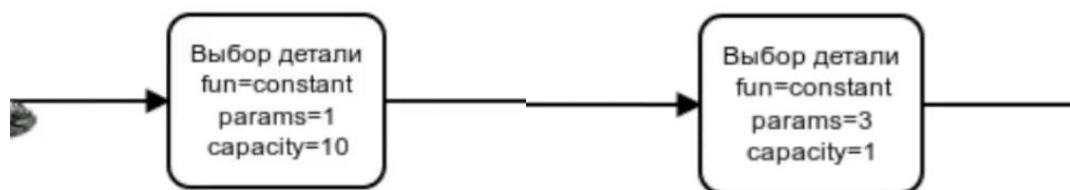


Рисунок 9. Пример изменения параметров моделирования.

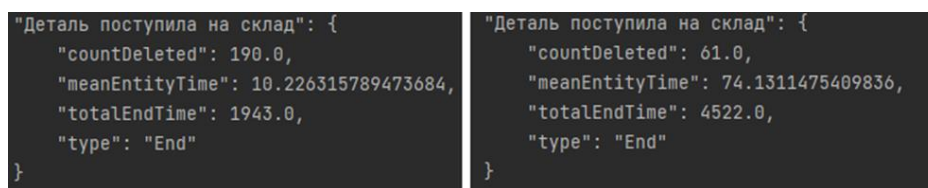
Очевидно, что такое изменение приведет к появлению очереди на входе в данный блок. Данное изменение отобразится в статистике. Для этого можно запустить модель на 10 шагов и посмотреть разницу в данном блоке (рисунок 10).

<pre> "Выбор детали": { "filledAmountDistribution": { "1": 10 }, "meanTimeInside": 1.0, "queueDistribution": {}, "totalPassed": 9.0, "totalTime": 9.0, "type": "Delay" }, </pre>	<pre> "Выбор детали": { "filledAmountDistribution": { "1": 10 }, "meanTimeInside": 3.0, "queueDistribution": { "1": 1, "2": 2, "3": 1, "4": 2, "5": 1, "6": 2 }, "totalPassed": 3.0, "totalTime": 9.0, "type": "Delay" }, </pre>
--	--

Рисунок 10. Статистика для модели с измененным параметром.

Видно, что в данном блоке появилась очередь, которая, вероятно, будет расти, так как пропускная способность данного блока после изменения меньше, чем выходной

поток генератора. Так же, с помощью статистики можно посмотреть, как изменится среднее время детали внутри модели при запуске на 200 единиц времени (рисунок 11).



```
"Деталь поступила на склад": {
  "countDeleted": 190.0,
  "meanEntityTime": 10.226315789473684,
  "totalEndTime": 1943.0,
  "type": "End"
}

"Деталь поступила на склад": {
  "countDeleted": 61.0,
  "meanEntityTime": 74.1311475409836,
  "totalEndTime": 4522.0,
  "type": "End"
}
```

Рисунок 11. Статистика для модели с измененным параметром.

Можно обратить внимание на то, что среднее время значительно увеличилось, а также значительно уменьшилось количество сущностей, которые полностью прошли через всю модель.

5. Заключение

В данной статье описан принцип работы разработанного программного комплекса по визуализации бизнес-процессов на основе BPMN модели. Программа была разработана с целью усовершенствования существующего способа анализа визуальных представлений бизнес-процессов.

Функциональные возможности данной программы позволяют загружать в неё различные модели, описанные в формате BPMN, запускать имитационную модель в веб-формате с анимацией или в консольном формате, останавливать и возобновлять симуляцию, а также сбрасывать результаты симуляции или использовать другую загруженную модель для новой симуляции.

На основе выводов программы, ЛПР может принимать решения по оптимизации моделируемых организационно-технических и социально-экономических систем. На примерах были показаны возможности визуализации, а также возможности по изменению модели для проведения анализа процессов. Описана методика моделирования некоторого заданного количества времени работы модели, разработан способ загрузки статистики, полученный после моделирования.

Новизна исследования заключается в создании уникального программного комплекса, готового к использованию в коммерческом секторе. Разработка нацелена на ЛПР, которые заинтересованы в оптимизации бизнес-процессов, в сборе статистики и принятии решения на основе полученной информации. Функционал визуализации на основе привычной BPMN нотации способствует быстрой интеграции программы в уже существующую систему построения схем в бизнесе, а дополнительная анимация обеспечивает улучшение понимания процессов.

Работа выполнена при финансовой поддержке ФГБОУ ВО "РЭУ им. Г.В. Плеханова".

Список литературы

1. Википедия: [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/IDEFo> (Дата обращения: 26.06.2022)
2. Википедия: [Электронный ресурс]. URL: https://en.wikipedia.org/wiki/Event-driven_process_chain (Дата обращения: 26.06.2022)
3. Макаров Н. С., Uml: поддержка проектирования и инструментальные среды // Прикладная информатика. 2007. №2.
4. Тасваева А. Н., Диаграммы потоков данных и вариантов использования как инструменты проектирования информационных систем // Модели, системы, сети в экономике, технике, природе и обществе. 2012. №2 (3).

5. Функциональные блок-схемы [Электронный ресурс] / Режим доступа: <https://rus-visio-blog-archive.github.io/2016/04/06/4605.htm> (дата обращения: 26.06.2022).
6. Chinosi, M., Trombetta, A. BPMN: An introduction to the standard. *Computer Standards & Interfaces*, 2012. 34(1), pp. 124–134. doi:10.1016/j.csi.2011.06.002.
7. Recker J., Mendling J. On the translation between BPMN and BPEL: conceptual mismatch between process modeling Languages, in: Thibaud Latour, Michael Petit (Eds.), *Proceedings 18th International Conference on Advanced Information Systems Engineering, Proceedings of Workshops and Doctoral Consortiums*, 2006, pp. 521–532.
8. Ouyang C., Aalst W.M., M. Dumas, Hofstede A.H. From business process models to process-oriented software systems: the BPMN to BPEL way // *Technical Report BPM-06-27*, BPM Center, 2006.
9. Куликов Г. Г., Шамсутдинов Р. Р., Алимбекова С. Р., Габбасов, Р. К. Применение методики построения моделей бизнес-процессов с использованием BPMN при регулярном, проектном и ситуационном управлении в нефтяной отрасли // *Вестник Уфимского государственного авиационного технического университета*, 2010. № 14 1(36), с. 136-147.
10. Куликов, Г. Г., Габбасов, Р. К., Навалихина, Н. Д., & Тымченко, А. А. Применение принципов системной инженерии и BPMN к процессу перевода государственных и муниципальных услуг в электронный вид с организацией межведомственного электронного взаимодействия // *Вестник Уфимского государственного авиационного технического университета*, 2013. 17 (5 (58)), с. 12-19.
11. Попов Е. В., Веретенникова А.Ю., Федореев С. А. Моделирование бизнес-процессов краудинвестинговых платформ на основе токенизации активов // *Мир новой экономики*, 2022. № 16 (1), с. 45-61.
12. Chetverushkin B.N., Sudakov V.A. Factor Model for the Study of Complex Processes // *Dokl. Math.* 2019. V. 100. – pp. 514–518.
13. The Universal Process Orchestrator [Электронный ресурс] / Режим доступа: <https://camunda.com/> (дата обращения: 26.06.2022).
14. APACHE LICENSE, VERSION 2.0 [Электронный ресурс] / Режим доступа: <https://www.apache.org/licenses/LICENSE-2.0> (дата обращения: 26.06.2022).
15. Low-code платформа для автоматизации и роботизации бизнес-процессов и CRM [Электронный ресурс] / Режим доступа: <https://elma365.com/ru/> (дата обращения: 26.06.2022).
16. Free online service for business process modeling [Электронный ресурс] / Режим доступа: <https://bpmn.studio/> (дата обращения: 26.06.2022).
17. Программный продукт "ПитерСофт: BPM" [Электронный ресурс] / Режим доступа: <https://piter-soft.ru/bpm/> (дата обращения: 26.06.2022).

Software Package for Visualization and Modeling Based on BPMN Notation

N.D. Badanina^{1,A,B}, V.A. Sudakov^{2,B,C}, N.A. Yashin^{3,C}

^A Moscow Aviation Institute (National Research University)

^B Keldysh Institute of Applied Mathematics

^C Plekhanov Russian University of Economics

¹ ORCID: 0000-0002-5301-1526, natashabadanina99@gmail.com

² ORCID: 0000-0002-1658-1941, sudakov@ws-dss.com

³ ORCID: 0000-0002-9693-0063, niknlooo7@gmail.com

Abstract

The article is devoted to the creation of software for visualization of complex processes. Since scientific organizations and business structures have demand for analysis and optimization of ongoing processes, the task of its effective modeling and visualization with the possibility of interaction is relevant. The software solutions available on the market are focused either on a static representation of the subject area, without the possibility of conducting computational experiments and optimizing solutions based on them, or on the use of non-standardized representations. A program has been developed that has the ability to animate the visualization of a loaded BPMN scheme, and also contains functionality for modeling the process described by the scheme, with the possibility of flexible parameter adjustment. The developed visualization program displays the result in the browser and a layer with animation of the passage of the involved resources according to the scheme is superimposed on top of the BPMN model. Visualization allows to reduce the probability of errors when modeling complex systems. The work also focuses on aspects of optimization, collecting statistics for subsequent analysis, modeling the process for a given number of time units. This functionality is included in the developed program and can be used both in console mode and in a web browser.

Keywords: visualization of algorithms, data visualization, BPMN notation, BPMN model, BPMN, applied program, process modeling, business processes.

References

1. Wikipedia: [web-resource]. URL: <https://ru.wikipedia.org/wiki/IDEFo> (date of application: 26.06.2022)
2. Wikipedia: [web-resource]. URL: https://en.wikipedia.org/wiki/Event-driven_process_chain (date of application: 26.06.2022)
3. Makarov N. S., Uml: podderzhka proektirovaniya i instrumentalnye sredi [Uml: design support and tooling] // Journal of Applied Informatics. 2007. №2 [in Russian].
4. Tasvaeva A. N., Diagrammi potokov dannih i variantov ispolzovaniya kak instrumenti proektirovaniya informazionnih sistem [Data flow and use case diagrams as information system design tools] // Models, systems, networks in economics, technology, nature and society. 2012. №2 (3) [in Russian].
5. Functional block diagrams [web-resource] / URL: <https://rus-visio-blog-archive.github.io/2016/04/06/4605.htm> (date of application: 26.06.2022).
6. Chinosi, M., Trombetta, A. BPMN: An introduction to the standard. Computer Standards & Interfaces, 2012. 34(1), pp. 124–134. doi:10.1016/j.csi.2011.06.002.
7. Recker J., Mendling J. On the translation between BPMN and BPEL: conceptual mismatch between process modeling Languages, in: Thibaud Latour, Michael Petit (Eds.),

Proceedings 18th International Conference on Advanced Information Systems Engineering, Proceedings of Workshops and Doctoral Consortiums, 2006, pp. 521–532.

8. Ouyang C., Aalst W.M., M. Dumas, Hofstede A.H. From business process models to process-oriented software systems: the BPMN to BPEL way // Technical Report BPM-06-27, BPM Center, 2006.

9. Kulikov G. G., Shamsutdinov R. R., Alimbekova S. R., Gabbasov, R. K. Application of Business Processes Models Generating Methodics Using BPM Notation while Regular, Project and Situation Management in Oil Sector // Bulletin of USATU, 2010. № 14 1(36), c. 136-147 [in Russian].

10. Kulikov G. G., Gabbasov, R. K., Navalihina N. D., Timchenko A. A. The adaptation of the principles of system engineering and BPMN to the process of transferring the governmental and municipal services into electronic form along with foundation of the interdepartmental electronic relations // Bulletin of USATU, 2013. 17 (5 (58)), c. 12-19 [in Russian].

11. Popov E. V., Veretennikova A. YU., Fedoreev S. A. Modelirovanie biznes-processov kraudinvestingovykh platform na osnove tokenizatsii aktivov [Modeling of business processes of crowdinvesting platforms based on asset tokenization] // The world of new economy, 2022 [in Russian].

12. Chetverushkin B.N., Sudakov V.A. Factor Model for the Study of Complex Processes // Dokl. Math. 2019. V. 100. – pp. 514–518.

13. The Universal Process Orchestrator [web-resource] / URL: <https://camunda.com/> (date of application: 26.06.2022).

14. APACHE LICENSE, VERSION 2.0 [web-resource] / URL: <https://www.apache.org/licenses/LICENSE-2.0> (date of application: 26.06.2022).

15. Low-code platform for automation and robotization of business processes and CRM [web-resource] / URL: <https://elma365.com/ru/> (date of application: 26.06.2022).

16. Free online service for business process modeling [web-resource] / URL: <https://bpmn.studio/> (date of application: 26.06.2022).

17. Software product "PiterSoft: BPM" [web-resource] / URL: <https://piter-soft.ru/bpm/> (date of application: 26.06.2022).