# Спецификация требований к алгоритмам и логике работы BPMN-конвертера

## Информация о документе

|  |  |
| --- | --- |
| **Название документа** | Спецификация требований к алгоритмам и логике работы BPMN-конвертера |
| **Версия** | 1.0 |
| **Дата создания** | 20.08.2025 |
| **Статус** | В разработке |
| **Тип документа** | Техническая спецификация |
| **Назначение** | Определение требований к существующему функционалу BPMN-конвертера StormBPMN → Camunda |

## 1. Общие сведения

### 1.1 Цель документа

Данный документ определяет требования к алгоритмам и логике работы BPMN-конвертера, предназначенного для автоматического преобразования BPMN диаграмм из формата **StormBPMN** в формат **Camunda Platform 7**.

### 1.2 Область применения

Требования распространяются на:

1. Основной модуль конвертации BPMN диаграмм (bpmn\_converter.py)
2. Скрипты интеграции с StormBPMN API (get\_diagram\_xml.py, get\_diagram\_assignees.py, get\_diagrams\_list.py)
3. Скрипт конвертации (convert.py)
4. Скрипт развертывания в Camunda (deploy.py)

### 1.3 Целевая платформа

1. **Источник**: StormBPMN (облачная платформа для моделирования бизнес-процессов)
2. **Назначение**: Camunda Platform 7 (движок выполнения бизнес-процессов)
3. **Среда выполнения**: Python 3.6+

## 2. Требования к преобразованию XML структуры

### 2.1 Добавление Camunda namespaces

**REQ-001: Корневые пространства имен**

1. Система ДОЛЖНА добавлять в корневой элемент BPMN диаграммы следующие атрибуты:
   * xmlns:camunda="http://camunda.org/schema/1.0/bpmn"
   * xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

### 2.2 Обновление атрибутов процесса

**REQ-002: Обязательные атрибуты процесса**

1. Система ДОЛЖНА добавить атрибут isExecutable="true" ко всем элементам <bpmn:process>
2. Система ДОЛЖНА добавить атрибут camunda:historyTimeToLive="1" ко всем элементам <bpmn:process>

**REQ-003: Сохранение исходных атрибутов**

1. Система НЕ ДОЛЖНА изменять следующие атрибуты:
   * ID процессов (остаются оригинальными)
   * exporterVersion (остается из исходной схемы)
   * Названия процессов (берутся из исходной схемы)

## 3. Требования к удалению элементов

### 3.1 Удаление collaboration секций

**REQ-004: Полное удаление collaboration**

1. Система ДОЛЖНА удалить элемент <bpmn:collaboration> со всем содержимым
2. Система ДОЛЖНА удалить все связанные участники, аннотации и группы

### 3.2 Удаление промежуточных событий

**REQ-005: Удаление всех промежуточных событий**

1. Система ДОЛЖНА удалить все элементы типа:
   * <bpmn:intermediateCatchEvent> (все типы)
   * <bpmn:intermediateThrowEvent> (все типы)
   * <bpmn:messageEventDefinition>
   * <bpmn:timerEventDefinition>

**REQ-006: Сохранение конечных событий**

1. Система НЕ ДОЛЖНА удалять элементы <bpmn:endEvent>

### 3.3 Удаление диаграммных элементов

**REQ-007: Очистка диаграммных элементов**

1. Система ДОЛЖНА удалить диаграммные элементы для удаленных объектов:
   * <bpmndi:BPMNShape> для удаленных элементов
   * <bpmndi:BPMNEdge> для удаленных потоков

## 4. Требования к перенаправлению потоков

### 4.1 Логика перенаправления

**REQ-008: Алгоритм перенаправления потоков**

При удалении промежуточного события система ДОЛЖНА:

1. Найти входящие потоки (targetRef = ID события)
2. Найти исходящие потоки (sourceRef = ID события)
3. Перенаправить входящие потоки на цели исходящих потоков
4. Удалить все связанные с событием потоки

**Пример трансформации:**

БЫЛО: Task\_A → IntermediateEvent → Task\_B

СТАЛО: Task\_A → Task\_B

**REQ-009: Обработка множественных потоков**

1. Если у промежуточного события несколько входящих или исходящих потоков, система ДОЛЖНА корректно перенаправить все комбинации
2. Система ДОЛЖНА избегать создания дублирующих потоков

## 5. Требования к преобразованию задач

### 5.1 Конвертация в serviceTask

**REQ-010: Преобразование типов задач**

Система ДОЛЖНА преобразовать следующие типы задач в <bpmn:serviceTask>:

* <bpmn:userTask>
* <bpmn:manualTask>
* <bpmn:callActivity>
* <bpmn:businessRuleTask>
* <bpmn:scriptTask>
* <bpmn:sendTask>
* <bpmn:receiveTask>
* <bpmn:task>

### 5.2 Добавление Camunda атрибутов

**REQ-011: Обязательные атрибуты serviceTask**

1. Система ДОЛЖНА добавить ко всем serviceTask элементам атрибуты:
   * camunda:type="external"
   * camunda:topic="bitrix\_create\_task"

**REQ-012: Сохранение исходных атрибутов задач**

1. Система ДОЛЖНА сохранить:
   * ID элемента
   * Название задачи (name)
   * Входящие и исходящие потоки

## 6. Требования к условным выражениям

### 6.1 Автоматическое добавление условий

**REQ-013: Условия для потоков "да"/"нет"**

Система ДОЛЖНА добавлять условные выражения к потокам:

Для потоков с name="да":

* <bpmn:conditionExpression xsi:type="bpmn:tFormalExpression">${result == "ok"}</bpmn:conditionExpression>

Для потоков с name="нет":

* <bpmn:conditionExpression xsi:type="bpmn:tFormalExpression">${result != "ok"}</bpmn:conditionExpression>

**REQ-014: Условия нечувствительны к регистру**

1. Система ДОЛЖНА распознавать варианты написания: "да", "Да", "ДА", "нет", "Нет", "НЕТ"

## 7. Требования к встраиванию ответственных

### 7.1 Обработка JSON данных ответственных

**REQ-015: Источник данных ответственных**

1. Система ДОЛЖНА искать JSON файл с ответственными по шаблону: {base\_name}\_assignees.json
2. Если JSON файл не найден, система ДОЛЖНА продолжить конвертацию без ответственных

**REQ-016: Структура данных ответственных**

Система ДОЛЖНА обрабатывать JSON данные в формате:

{

"assigneeEdgeId": 15700296,

"assigneeName": "Рук.отд. арх.сопр. и анализа проектов",

"assigneeId": 15298311,

"elementId": "Activity\_1597r5e",

"elementName": "Поставить задачу",

"assigneeType": "HUMAN",

"duration": 900

}

### 7.2 Встраивание в serviceTask

**REQ-017: Алгоритм встраивания**

Система ДОЛЖНА:

1. Проанализировать JSON файл с ответственными
2. Для каждого ответственного найти соответствующий serviceTask по elementId
3. Создать (или дополнить) секцию <bpmn:extensionElements>
4. Добавить <camunda:properties> с двумя свойствами:
   * assigneeName - имя ответственного
   * assigneeId - ID ответственного

**REQ-018: Обработка множественных ответственных**

1. Если для элемента назначено несколько ответственных, система ДОЛЖНА встроить только первого
2. Система ДОЛЖНА логировать информацию о пропущенных ответственных

**Пример результата:**

<bpmn:serviceTask id="Activity\_14qyrmj" name="Отправить сообщение" camunda:type="external" camunda:topic="bitrix\_create\_task">

<bpmn:extensionElements>

<camunda:properties>

<camunda:property name="assigneeName" value="Рук.отд. инж.коммуникаций" />

<camunda:property name="assigneeId" value="15299256" />

</camunda:properties>

</bpmn:extensionElements>

<bpmn:incoming>Flow\_1</bpmn:incoming>

<bpmn:outgoing>Flow\_2</bpmn:outgoing>

</bpmn:serviceTask>

## 8. Требования к исправлению конфликтов

### 8.1 Конфликты default flow

**REQ-019: Проблема default flow**

Система ДОЛЖНА решать конфликт, возникающий когда:

1. Эксклюзивный шлюз имеет атрибут default
2. Соответствующий поток имеет условное выражение

Это вызывает ошибку: ENGINE-09005 Exclusive Gateway 'Gateway\_1' has outgoing sequence flow 'Flow\_1' which is the default flow but has a condition too.

**REQ-020: Алгоритм исправления**

Система ДОЛЖНА:

1. Проверить все <bpmn:exclusiveGateway> элементы с атрибутом default
2. Найти соответствующий поток по ID
3. Если поток содержит <bpmn:conditionExpression>, удалить атрибут default у шлюза

### 8.2 Сохранение корректных default flow

**REQ-021: Сохранение корректных default потоков**

1. Система НЕ ДОЛЖНА удалять атрибут default, если соответствующий поток не имеет условия
2. Система ДОЛЖНА сохранять функциональность default потоков где это возможно

## 9. Требования к порядку элементов

### 9.1 Соответствие BPMN спецификации

**REQ-022: Корректный порядок элементов**

Система ДОЛЖНА упорядочить элементы внутри BPMN узлов согласно спецификации:

**Для обычных элементов процесса:**

* <extensionElements> (если есть)
* <ioSpecification> (если есть)
* <property> (если есть)
* <dataInputAssociation> / <dataOutputAssociation> (если есть)
* <resourceRole> (если есть)
* <loopCharacteristics> (если есть)
* Все <incoming> элементы
* Все <outgoing> элементы
* Остальные элементы

**Для sequenceFlow:**

* <extensionElements> (если есть)
* <conditionExpression> (если есть)
* Остальные элементы

**REQ-023: Группировка однотипных элементов**

1. Все <incoming> элементы ДОЛЖНЫ быть сгруппированы вместе
2. Все <outgoing> элементы ДОЛЖНЫ быть сгруппированы вместе

## 10. Требования к обработке файлов

### 10.1 Входные файлы

**REQ-024: Формат входных файлов**

1. Система ДОЛЖНА принимать файлы с расширением .bpmn
2. Система ДОЛЖНА корректно обрабатывать XML в кодировке UTF-8
3. Система ДОЛЖНА выдавать предупреждение при обработке файлов без расширения .bpmn

**REQ-025: Валидация входных файлов**

1. Система ДОЛЖНА проверить существование входного файла
2. Система ДОЛЖНА валидировать корректность XML структуры
3. При ошибках валидации система ДОЛЖНА выдать информативное сообщение

### 10.2 Выходные файлы

**REQ-026: Формат выходных файлов**

1. Система ДОЛЖНА сохранять результат в той же папке, что и исходный файл
2. Имя выходного файла ДОЛЖНО иметь префикс camunda\_
3. Файл ДОЛЖЕН сохраняться в кодировке UTF-8
4. Система ДОЛЖНА сохранять читаемое форматирование XML

**REQ-027: Обработка конфликтов имен**

1. Если выходной файл уже существует, система ДОЛЖНА его перезаписать
2. Система ДОЛЖНА логировать информацию о перезаписи файлов

## 11. Требования к интеграции с внешними системами

### 11.1 Интеграция с StormBPMN API

**REQ-028: Получение диаграмм**

1. Система ДОЛЖНА поддерживать получение BPMN диаграмм по ID через StormBPMN API
2. Система ДОЛЖНА поддерживать получение списка диаграмм с пагинацией
3. Система ДОЛЖНА поддерживать получение списка ответственных по диаграмме

**REQ-029: Аутентификация StormBPMN**

1. Система ДОЛЖНА использовать Bearer Token аутентификацию
2. Токен ДОЛЖЕН читаться из переменной окружения STORMBPMN\_BEARER\_TOKEN
3. При ошибках аутентификации система ДОЛЖНА выдать информативное сообщение

**REQ-030: Обработка ошибок StormBPMN**

1. Система ДОЛЖНА корректно обрабатывать ошибки "Диаграмма не найдена"
2. Система ДОЛЖНА корректно обрабатывать ошибки сети и тайм-ауты
3. Все ошибки ДОЛЖНЫ логироваться с соответствующим уровнем важности

### 11.2 Интеграция с Camunda API

**REQ-031: Развертывание в Camunda**

1. Система ДОЛЖНА поддерживать развертывание BPMN диаграмм через Camunda REST API
2. Система ДОЛЖНА создавать новый деплой с определениями процессов
3. Система ДОЛЖНА отображать информацию о развернутых процессах

**REQ-032: Аутентификация Camunda**

1. Система ДОЛЖНА поддерживать Basic аутентификацию
2. Учетные данные ДОЛЖНЫ читаться из переменных окружения:
   1. CAMUNDA\_BASE\_URL
   2. CAMUNDA\_AUTH\_USERNAME
   3. CAMUNDA\_AUTH\_PASSWORD

**REQ-033: Валидация перед развертыванием**

1. Система ДОЛЖНА проверить соединение с Camunda перед развертыванием
2. Система ДОЛЖНА валидировать BPMN схему перед отправкой
3. При ошибках валидации система ДОЛЖНА выдать рекомендации по исправлению

## 12. Требования к логированию и отчетности

### 12.1 Уровни логирования

**REQ-034: Обязательные логи**

Система ДОЛЖНА логировать:

1. **INFO**: Начало и завершение основных операций
2. **INFO**: Статистику преобразований (количество удаленных/преобразованных элементов)
3. **WARNING**: Неожиданные ситуации, не прерывающие работу
4. **ERROR**: Критические ошибки, прерывающие выполнение

**REQ-035: Детализация операций**

Система ДОЛЖНА логировать для каждой операции:

1. Добавление Camunda namespaces
2. Количество найденных и удаленных промежуточных событий
3. Количество перенаправленных потоков
4. Количество преобразованных задач
5. Количество добавленных условных выражений
6. Количество встроенных ответственных
7. Количество исправленных конфликтов default flow

### 12.2 Формат вывода

**REQ-036: Пользовательский интерфейс**

1. Система ДОЛЖНА использовать эмодзи и цветовое кодирование для улучшения читаемости
2. Система ДОЛЖНА отображать прогресс выполнения операций
3. Система ДОЛЖНА показывать итоговую статистику конвертации

**REQ-037: Структурированный вывод**

1. Логи ДОЛЖНЫ иметь единообразный формат с временными метками
2. Ошибки ДОЛЖНЫ содержать информацию о типе и местоположении проблемы
3. Система ДОЛЖНА предоставлять рекомендации по устранению ошибок

## 13. Требования к обработке ошибок

### 13.1 Классификация ошибок

**REQ-038: Типы обрабатываемых ошибок**

Система ДОЛЖНА корректно обрабатывать:

1. **FileNotFoundError**: Файл не найден
2. **XML Parse Error**: Некорректный XML/BPMN файл
3. **StormBPMNAuthError**: Ошибка аутентификации StormBPMN
4. **StormBPMNNotFoundError**: Диаграмма не найдена в StormBPMN
5. **CamundaAuthError**: Ошибка аутентификации Camunda
6. **CamundaValidationError**: Ошибка валидации BPMN в Camunda
7. **CamundaDeployError**: Ошибка развертывания в Camunda

### 13.2 Стратегии восстановления

**REQ-039: Graceful degradation**

1. При ошибках загрузки ответственных система ДОЛЖНА продолжить конвертацию без них
2. При ошибках в JSON файле ответственных система ДОЛЖНА логировать ошибку и продолжить
3. При частичных ошибках преобразования система ДОЛЖНА завершить обработку остальных элементов

**REQ-040: Информативность ошибок**

1. Каждая ошибка ДОЛЖНА содержать рекомендации по устранению
2. Система ДОЛЖНА предоставлять диагностическую информацию для отладки
3. При критических ошибках система ДОЛЖНА предложить альтернативные действия

## 14. Требования к производительности

### 14.1 Ограничения по размеру файлов

**REQ-041: Обработка больших файлов**

1. Система ДОЛЖНА корректно обрабатывать BPMN файлы размером до 10 МБ
2. Система ДОЛЖНА показывать информацию о размерах входного и выходного файлов
3. При обработке больших файлов система ДОЛЖНА отображать прогресс

### 14.2 Оптимизация памяти

**REQ-042: Управление памятью**

1. Система ДОЛЖНА эффективно использовать память при обработке XML
2. Система НЕ ДОЛЖНА загружать в память избыточные данные
3. При возможности система ДОЛЖНА использовать потоковую обработку XML

## 15. Требования к обратной совместимости

### 15.1 Поддержка версий BPMN

**REQ-043: Совместимость с BPMN 2.0**

1. Система ДОЛЖНА корректно обрабатывать диаграммы BPMN 2.0
2. Система ДОЛЖНА сохранять совместимость с различными экспортерами BPMN
3. Система НЕ ДОЛЖНА изменять version информацию в исходных файлах

### 15.2 Совместимость с Camunda Platform

**REQ-044: Поддержка Camunda Platform 7**

1. Выходные файлы ДОЛЖНЫ быть совместимы с Camunda Platform 7.x
2. Система ДОЛЖНА использовать корректные Camunda namespaces и атрибуты
3. Результат ДОЛЖЕН успешно импортироваться в Camunda Modeler

## 16. Требования к тестированию

### 16.1 Модульное тестирование

**REQ-045: Покрытие основных функций**

1. Система ДОЛЖНА иметь тесты для всех основных алгоритмов преобразования
2. Тесты ДОЛЖНЫ покрывать позитивные и негативные сценарии
3. Тесты ДОЛЖНЫ проверять корректность XML структуры результата

### 16.2 Интеграционное тестирование

**REQ-046: Тестирование с реальными данными**

1. Система ДОЛЖНА тестироваться на реальных BPMN диаграммах из StormBPMN
2. Результаты ДОЛЖНЫ проверяться на возможность развертывания в Camunda
3. Система ДОЛЖНА иметь тесты интеграции с внешними API

## 17. Требования к безопасности

### 17.1 Обработка конфиденциальных данных

**REQ-047: Защита учетных данных**

1. Система НЕ ДОЛЖНА логировать пароли и токены доступа
2. Учетные данные ДОЛЖНЫ читаться только из переменных окружения
3. Система ДОЛЖНА предупреждать о небезопасном хранении учетных данных

### 17.2 Валидация входных данных

**REQ-048: Защита от вредоносных данных**

1. Система ДОЛЖНА валидировать все входные XML файлы
2. Система ДОЛЖНА защищаться от XML External Entity (XXE) атак
3. Система ДОЛЖНА ограничивать размер обрабатываемых файлов

## 18. Приложения

### 18.1 Примеры преобразований

**Приложение A: Примеры XML трансформаций**

1. Пример удаления промежуточного события
2. Пример преобразования userTask в serviceTask
3. Пример встраивания ответственных
4. Пример исправления default flow конфликта

### 18.2 Конфигурационные файлы

**Приложение B: Шаблон .env файла**

# StormBPMN Configuration

STORMBPMN\_BEARER\_TOKEN=your\_storm\_bpmn\_token\_here

# Camunda Configuration

CAMUNDA\_BASE\_URL=https://camunda.eg-holding.ru/engine-rest

CAMUNDA\_AUTH\_USERNAME=your\_username

CAMUNDA\_AUTH\_PASSWORD=your\_password

### 18.3 Типовые сценарии использования

**Приложение C: Сценарии использования**

1. Получение диаграммы из StormBPMN и конвертация
2. Конвертация локального BPMN файла
3. Конвертация с встраиванием ответственных
4. Развертывание сконвертированной диаграммы в Camunda