



Министерство науки и высшего образования Российской Федерации  
федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
«Московский государственный технический университет имени  
Н.Э. Баумана (национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ «Робототехники и комплексной автоматизации»  
КАФЕДРА «Системы автоматизированного проектирования (РК-6)»

**НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЕ ЗАМЕТКИ**  
по направлению «Разработка систем инженерного анализа и  
ресурсоемкого ПО (rndhpc)»

Авторы (исследователи):	Крехтунова Д., Ершов В., Муха В., Тришин И.
Научный(е) руководитель(и):	Соколов А.П., Першин А.Ю.
Консультанты:	@Фамилия И.О.@

Москва, 2021–2021

Работа (документирование) над научным направлением начата 20 сентября 2021 г.

**Руководители по направлению:**

СОКОЛОВ,	–	канд. физ.-мат. наук, доцент кафедры САПР,
Александр Павлович		МГТУ им. Н.Э. Баумана
ПЕРШИН,	–	PhD, ассистент кафедры САПР,
Антон Юрьевич		МГТУ им. Н.Э. Баумана

**Исследователи (студенты кафедры САПР, МГТУ им. Н.Э. Баумана):**

Крехтунова Д., Ершов В., Муха В., Тришин И.

C59      **Крехтунова Д., Ершов В., Муха В., Тришин И.. Разработка систем инженерного анализа и ресурсоемкого ПО (rndhpc):** Научно-исследовательские заметки. / Под редакцией Соколова А.П. [Электронный ресурс] — Москва: 2021. — 8 с. URL: <https://arch.rk6.bmstu.ru> (облачный сервис кафедры РК6)

Документ содержит краткие материалы, формируемые обучающимися и исследователями в процессе их работ по одному научному направлению.

Документ разработан для оценки результативности проведения научных исследований по направлению «Разработка систем инженерного анализа и ресурсоемкого ПО» в рамках реализации курсовых работ, курсовых проектов, выпускных квалификационных работ бакалавров и магистров, а также диссертационных исследований аспирантов кафедры «Системы автоматизированного проектирования» (РК6) МГТУ им. Н.Э. Баумана.

RNDHPC



Крехтунова Д., Ершов В., Муха В., Тришин И.,  
Соколов А.П., Першин А.Ю., 2021

## Содержание

<b>1</b>	<b>Разработка графоориентированного дебаггера</b>	<b>4</b>
	2021.09.19: Содержание научно-исследовательской заметки . . . . .	4
<b>2</b>	<b>Разработка web-ориентированного редактора графовых моделей</b>	<b>4</b>
	2021.10.05: Обзор языка описания графов DOT . . . . .	4
<b>3</b>	<b>Графоориентированная методология разработки средств взаимодействия пользователя в системах автоматизированного проектирования и инженерного анализа</b>	<b>6</b>
	2021.11.06: Особенности применения графового описания процессов обработки данных в pSeven (DATADVANCE) . . . . .	6
	Принципы функционирования . . . . .	6
	Поддержка циклов и ветвлений . . . . .	7
	Особенности выполнения расчётных схем . . . . .	7

# 1 Разработка графоориентированного дебаггера

## 2021.09.19: Содержание научно-исследовательской заметки

Заметка размещается в рамках L<sup>A</sup>T<sub>E</sub>X-подраздела (`\subsection`).

В состав заметки следует включать:

- заметку следует создавать с помощью вспомогательной команды `\notestatement{@prjsid@}{@NoteTitle@}`;
- атрибуты заметки (дата, автор, идентификатор исследовательского проекта, тема заметки) следует заполнять явно, без введения дополнительных макроподстановок;
- **рекомендуется** в состав заметки включать: рисунки; схемы; графические результаты расчетов; формулы; математические постановки задач, представляемые исключительно в математически строгом виде;
- при включении в состав заметки утверждения следует добавлять сноску с выходными данными источника (при этом следует добавлять соответствующий источник в файл библиографии `bibliography.bib`);
- все сопроводительные документы по текущей заметке следует размещать в каталоге, имеющем такое же имя, как имя файла заметки (рис. 1);
- объём одной заметки: не более 2-3 страницы.

Подготовлено: *Соколов А.П. (РК6), 2021.09.19*

## 2 Разработка web-ориентированного редактора графовых моделей

### 2021.10.05: Обзор языка описания графов DOT

Язык описания графов DOT предоставляется пакетом утилит Graphviz (Graph Visualization Software). Пакет состоит из набора утилит командной строки и программ с графическим интерфейсом, способных обрабатывать файлы на языке DOT, а также из виджетов и библиотек, облегчающих создание графов и программ для построения графов. Более подробно будет рассмотрена утилита `dot`.

`dot` - инструмент для создания многоуровневого графа с возможностью вывода изображения полученного графа в различных форматах (PNG, PDF, PostScript, SVG и др.).

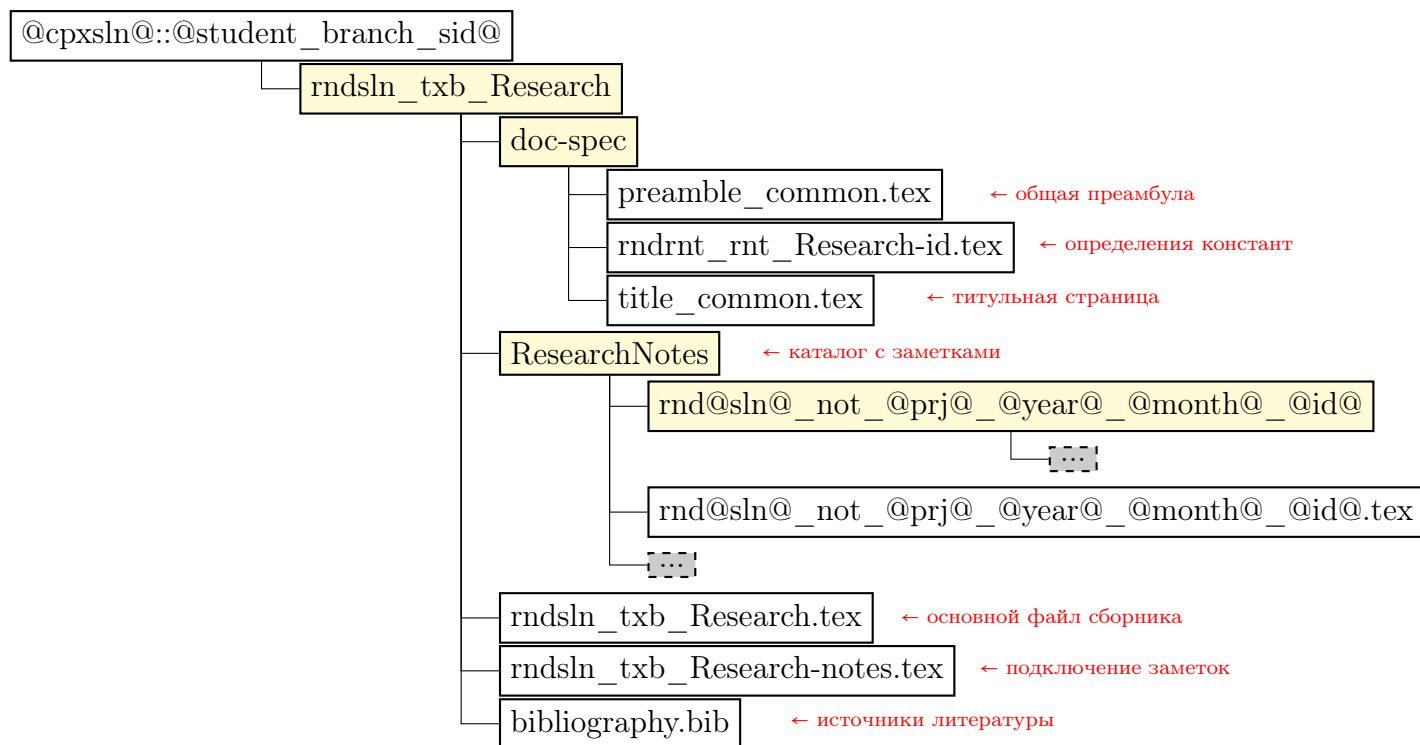


Рис. 1. Структура файловой системы исходников сборника исследовательских заметок

## Установка graphviz

Linux: `sudo apt install graphviz`

MacOS: `brew install graphviz`

Вызов всех программ Graphviz осуществляется через командную строку, в процессе ознакомления с языком использовалась следующая команда

`dot -Tpng <pathToDotFile> -o <imageName>`

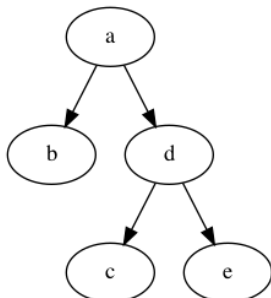
В результате выполнения этой команды будет создано изображение графа в формате png

## Пример описания простого графа

```

digraph G {
  a -> b;
  a -> d -> c;
  d -> e;
}

```



Более подробная информация с примерами представлена в обзоре литературы, который находится по следующему пути:

01 - Курсовые проекты/2021-2022 - Разработка web-ориентированного редактора графовых моделей /0 - Обзор литературы/

Подготовлено: *Ершов В. (РК6-72Б), 2021.10.05*

### **3 Графоориентированная методология разработки средств взаимодействия пользователя в системах автоматизированного проектирования и инженерного анализа**

#### **2021.11.06: Особенности применения графового описания процессов обработки данных в pSeven (DATADVANCE)**

pSeven – это платформа для анализа данных, оптимизации и создания аппроксимационных моделей, дополняющая средства проектирования и инженерного анализа. pSeven позволяет интегрировать в единой программной среде различные инженерные приложения, алгоритмы многодисциплинарной оптимизации и инструменты анализа данных для упрощения принятия конструкторских решений[1].

#### **Принципы функционирования**

На концептуальном уровне в pSeven вводятся следующие понятия:

- Проект – набор файлов, используемых в pSeven для описания решений одной или нескольких задач и хранения результатов их решения.
- Расчётная схема (workflow) – формальное описание процесса решения некоторой задачи в виде ориентированного графа, узлами которого являются блоки, а рёбрами - связи. Такое описание хранится в бинарном файле с расширением .p7wf, использующем некоторый специализированный формат хранения подобного рода описаний.
- Блок – функциональный элемент расчётной схемы, отвечающий за обработку входных данных и формирование выходных данных.[2]
- Порт – переменная определённого типа, описанная в блоке и имеющая в нём уникальное имя, значение которой может быть передано в другие блоки или получено от них через связи.

- Связь (link) – одностороннее соединение между двумя портами, обеспечивающее передачу данных от одного к другому.

Проекты в **pSeven** имеют единую базу данных, куда записываются все результаты запусков расчётных схем и откуда берутся данные для их последующей презентации пользователю и их анализа. Для определения переменных, значения которых должны быть записаны в неё записаны, предусмотрены специализированные порты для самих расчётных схем, с которыми связываются те блоки, результаты выполнения которых интересуют пользователя.

Связи служат для маршрутизации данных. С их помощью осуществляется и взаимодействие между блоками и, кроме того, определяется очерёдность их запуска. В момент добавления связи в пакете **pSeven** выполняется проверка портов на совместимость. Они считаются совместимыми, если тип данных источника можно преобразовать к типу данных адресата.[\[2\]](#)

## Поддержка циклов и ветвлений

Расчетная схема может включать расчетные циклы. Для их создания применяются специализированные блоки, имеющие функциональную возможность управления запуском других блоков в теле цикла и принятия решения о его прекращении. Такие блоки называются управляющими блоками циклов. Одним из характерных примеров является оптимизационный цикл, управление которым осуществляется из блока **Optimizer** [\[2\]](#), позволяющего, например, настроить максимальное число итераций или требуемую точность, как условия окончания.

Если речь идёт о задачах анализа данных, существует отдельный блок, обозначающий входную точку цикла (**Loop**). Такой цикл принимает на вход список наборов аргументов, каждый из которых будет обработан на соответствующем шаге цикла, и выдаёт список наборов выходных значений, которые сохраняются в базе данных проекта.

Кроме того, существует возможность включения в расчётную схему условного и безусловного ветвления. Первое достигается за счёт создания связей для подключения одного и того же порта вывода к различным портам ввода. В этом случае по каждой связи передается копия выходных данных, полученных у источника [\[2\]](#). Условные ветвления создаются с помощью специального блока **Condition**, который по определённому условию передаёт входные данные одному из подключенных блоков. Их целесообразно использовать для устранения ошибок в работе блока, отбраковки некорректных входных данных и других аналогичных целей[\[2\]](#).

## Особенности выполнения расчётных схем

При выполнении расчётных схем каждый блок запускается в отдельном процессе на уровне операционной системы. Как сказано ранее, начало выполнения блока определяется его связями с другими. Любой блок будет ожидать завершения работы другого блока только в том случае, если ему необходимо получить от него входные данные. Это

означает, что два блока, не имеющих связей друг с другом Блоки, входящие в состав различных ветвлений расчетной схемы, могут запускаться параллельно, поскольку они не зависят друг от друга по используемым данным[2].

Кроме того, при обработке больших выборок данных может потребоваться обрабатывать по несколько наборов данных одновременно в независимых потоках исполнения. Помимо прочего этой цели служит блок **Composite**, который является контейнером для нескольких блоков. В его настройках можно включить опцию параллельного исполнения, указать, с какого порта данные будут обрабатываться параллельно, и указать максимальное число потоков. При запуске расчетной схемы пакет **pSeven** создает несколько виртуальных экземпляров блока **Composite** и автоматически распределяет входные наборы данных между ними. Как только в одном из таких виртуальных блоков завершается расчет, он получает из выборки следующий набор для обработки.[3]

Подготовлено: *Тришин И.В. (РК6), 2021.11.06*

## Список литературы

- [1] ДАТАДВАНС | Программное обеспечение для анализа данных и оптимизации [Электронный ресурс] [Офиц. сайт]. 2021. (дата обращения 06.11.2021).
- [2] Расчётные схемы - Руководство пользователя pSeven 6.27 [Электронный ресурс] [Офиц. сайт]. 2021. Дата обращения: 15.11.2021.
- [3] Параллельные вычисления - Руководство пользователя pSeven 6.27 [Электронный ресурс] [Офиц. сайт]. 2021. (дата обращения 15.11.2021).