

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

РАЗРАБОТКА И РЕАЛИЗАЦИЯ МЕТОДОВ ЭФФЕКТИВНОГО
ВЗАИМОДЕЙСТВИЯ ПРОЦЕССОВ В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ

Автор: Губарев Владимир Юрьевич _____

Направление подготовки: 09.04.04 Программная
инженерия

Квалификация: Магистр

Руководитель: Косяков М.С., к.т.н. _____

К защите допустить

Руководитель ОП Парфенов В.Г., проф., д.т.н. _____

« ____ » _____ 20 ____ г.

Санкт-Петербург, 2020 г.

Студент Губарев В.Ю.

Группа Р42111 Факультет ИТиП

Направленность (профиль), специализация
Информационно-вычислительные системы

ВКР принята « _____ » _____ 20__ г.

Оригинальность ВКР _____ %

ВКР выполнена с оценкой _____

Дата защиты « _____ » _____ 20__ г.

Секретарь ГЭК Болдырева Е.А. _____

Листов хранения _____

Демонстрационных материалов/Чертежей хранения _____

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ»

УТВЕРЖДАЮ

Руководитель ОП
проф., д.т.н. Парфенов В.Г. _____
« ____ » _____ 20 ____ г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

Студент Губарев В.Ю.

Группа Р42111 Факультет ИТиП

Руководитель Косяков М.С., к.т.н., доцент ФПИиКТ

1 Наименование темы: Разработка и реализация методов эффективного взаимодействия процессов в распределенных системах

Направление подготовки (специальность): 09.04.04 Программная инженерия

Направленность (профиль): Информационно-вычислительные системы

Квалификация: Магистр

2 Срок сдачи студентом законченной работы: « ____ » _____ 20 ____ г.

3 Техническое задание и исходные данные к работе

Требуется разработать стилевой файл для системы \LaTeX , позволяющий оформлять бакалаврские работы и магистерские диссертации на кафедре компьютерных технологий Университета ИТМО. Стилиевой файл должен генерировать титульную страницу пояснительной записки, задание, аннотацию и содержательную часть пояснительной записки. Первые три документа должны максимально близко соответствовать шаблонам документов, принятым в настоящий момент на кафедре, в то время как содержательная часть должна максимально близко соответствовать ГОСТ 7.0.11-2011 на диссертацию.

4 Содержание выпускной работы (перечень подлежащих разработке вопросов)

Пояснительная записка должна демонстрировать использование наиболее типичных конструкций, возникающих при составлении пояснительной записки (перечисления, рисунки, таблицы, листинги, псевдокод), при этом должна быть составлена так, что демонстрируется корректность работы стилевого файла. В частности, записка должна содержать не менее двух приложений (для демонстрации нумерации рисунков и таблиц по приложениям согласно ГОСТ) и не менее десяти элементов нумерованного перечисления первого уровня вложенности (для демонстрации корректности используемого при нумерации набора русских букв).

5 Перечень графического материала (с указанием обязательного материала)

Графические материалы и чертежи работой не предусмотрены

6 Исходные материалы и пособия

- а) ГОСТ 7.0.11-2011 «Диссертация и автореферат диссертации»;
- б) С.М. Львовский. Набор и верстка в системе \LaTeX ;
- в) предыдущий комплект стилевых файлов, использовавшийся на кафедре компьютерных технологий.

7 Дата выдачи задания « ____ » _____ 20 ____ г.

Руководитель ВКР _____

Задание принял к исполнению _____

« ____ » _____ 20 ____ г.

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ»

АННОТАЦИЯ
ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

Студент: Губарев Владимир Юрьевич

Наименование темы ВКР: Разработка и реализация методов эффективного взаимодействия процессов в распределенных системах

Наименование организации, в которой выполнена ВКР: Университет ИТМО

ХАРАКТЕРИСТИКА ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

1 Цель исследования: Разработка удобного стилевого файла \LaTeX для бакалавров и магистров кафедры компьютерных технологий.

2 Задачи, решаемые в ВКР:

- а) обеспечение соответствия титульной страницы, задания и аннотации шаблонам, принятым в настоящее время на кафедре;
- б) обеспечение соответствия содержательной части пояснительной записки требованиям ГОСТ 7.0.11-2011 «Диссертация и автореферат диссертации»;
- в) обеспечение относительного удобства в использовании — указание данных об авторе и научном руководителе один раз и в одном месте, автоматический подсчет числа тех или иных источников.

3 Число источников, использованных при составлении обзора: 0

4 Полное число источников, использованных в работе: 0

5 В том числе источников по годам:

Отечественных			Иностранных		
Последние 5 лет	От 5 до 10 лет	Более 10 лет	Последние 5 лет	От 5 до 10 лет	Более 10 лет
0	0	0	0	0	0

6 Использование информационных ресурсов Internet: нет

7 Использование современных пакетов компьютерных программ и технологий:

Пакеты компьютерных программ и технологий	Раздел работы
Пакет <code>tabularx</code> для чуть более продвинутых таблиц	??, Приложения А, Б
Пакет <code>biblatex</code> и программное средство <code>biber</code>	Список использованных источников

8 Краткая характеристика полученных результатов

Получился, надо сказать, практически неплохой стилевик. В 2015–2018 годах его уже использовали некоторые бакалавры и магистры. Надеюсь на продолжение.

9 Гранты, полученные при выполнении работы

Автор разрабатывал этот стилевик исключительно за свой счет и на добровольных началах. Однако значительная его часть была бы невозможна, если бы автор не написал в свое время кандидатскую диссертацию в L^AT_EX, а также не отвечал за формирование кучи научно-технических отчетов по гранту, известному как «5-в-100», что происходило при государственной финансовой поддержке ведущих университетов Российской Федерации (субсидия 074-U01).

10 Наличие публикаций и выступлений на конференциях по теме работы

По теме этой работы я (к счастью!) ничего не публиковал. Однако покажу, как можно ссылаться на свои публикации из списка литературы:

Студент Губарев В.Ю. _____

Руководитель Косяков М.С. _____

« _____ » _____ 20__ г.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	6
ГЛАВА 1. Обзор методов межпроцессного взаимодействия	9
Выводы по главе 1	9
ГЛАВА 2. Анализ методов межпроцессного взаимодействия	10
Выводы по главе 2	10
ГЛАВА 3. Разработка и реализация методов эффективного взаимодействия процессов	11
Выводы по главе 3	11
ГЛАВА 4. Проведение эксперимента и обработка результатов	12
4.1. Постановка эксперимента	12
4.1.1. Конфигурация экспериментального стенда	12
4.1.2. Конфигурация экспериментальной системы	12
4.1.3. Характер экспериментальной нагрузки	13
4.1.4. Время обслуживания заявок в процессах	14
4.2. Результаты экспериментов	14
4.2.1. Использование ТСП для передачи данных	14
4.2.2. Использование разделяемой памяти для передачи данных	16
Выводы по главе 4	20
ЗАКЛЮЧЕНИЕ	22
ГЛАВА А. Пример приложения	23
ГЛАВА Б. Еще один пример приложения с невероятно длинным названием для тестирования переносов	24
ГЛАВА В. Пример огромного листинга	25

ВВЕДЕНИЕ

Объектом исследования являются методы межпроцессного взаимодействия.

Предметом исследования является временная задержка на передачу данных между процессами распределенной системы в пределах одного физического узла.

Цель работы – уменьшение временной задержки на передачу данных между процессами в пределах одного физического узла путем разработки и применения методов эффективного межпроцессного взаимодействия.

В рамках настоящей работы поставлены следующие **задачи**:

- рассмотреть существующие методы межпроцессного взаимодействия, доступные при взаимодействии процессов, находящихся на одном физическом узле;
- произвести анализ и отбор методов межпроцессного взаимодействия для реализации новых методов межпроцессного взаимодействия;
- разработать и реализовать эффективные методы межпроцессного взаимодействия;
- экспериментально оценить полученные методы межпроцессного взаимодействия.

Актуальность исследования.

Для некоторых систем эффективное межпроцессное взаимодействие является критически важной частью их работы. Требование по минимизации времени обслуживания заявок может напрямую следовать из области применения системы, как в случае с системами для алгоритмической торговли на финансовых рынках. Обслуживание заявок множеством логически связанных процессов может быть существенно ускорено при размещении таких процессов на одном физическом узле. Современные процессоры с количеством с десятками вычислительных ядер могут обеспечить такую конфигурацию нужными ресурсами. Это позволяет использовать более эффективные методы межпроцессного взаимодействия, а именно методы на основе разделяемой памяти [Smith2012DraftH]. Эффективные методы межпроцессного взаимодействия могут использоваться для связи виртуальных машин или контейнеров в пределах машины-хозяина [IPCInterVirtualMachineShmem,

IPCInterVirtualMachineShmemOptimizations]. Для связи программных модулей, исполняющихся в разных процессах для обеспечения отказоустойчивости за счет изоляции процессов на уровне ОС. Для высокопроизводительных вычислений, таких как анализ научных данных или прогнозирование погоды.

При разработке сложной многокомпонентной распределенной системы программисту необходимо сосредоточиться на логике и корректности работы самой системы. В то время как методы межпроцессного взаимодействия должны быть для него прозрачны. Этого можно достичь, используя единый унифицированный интерфейс для межпроцессного взаимодействия. Это упрощает разработку, снимает необходимость сложного управления ресурсами для межпроцессного взаимодействия. А также позволяет автоматически использовать наиболее подходящие методы межпроцессного взаимодействия для данных *пространственных конфигураций* (**TBD: может, убрать?**) процессов, что может повысить эффективность выполнения некоторых задач этой системой.

Таким образом, разработка и реализация эффективных методов межпроцессного взаимодействия и интерфейса для автоматического доступа к наиболее подходящим из них необходима и обоснована. Посредством этого интерфейса программист прозрачно для себя использует методы межпроцессного взаимодействия на основе разделяемой памяти при взаимодействии с локальными процессами без необходимости перекомпиляции программы. Но поскольку зачастую нельзя разместить всю систему на одном, даже очень производительном, сервере используется ТСП при взаимодействии с процессами на других физических узлах.

Методы исследования включают в себя анализ существующих методов межпроцессного взаимодействия, экспериментальную оценку разработанных методов эффективного межпроцессного взаимодействия и методы математической статистики для обработки экспериментальных данных.

Средства исследования:

- язык программирования C++, компилятор *Clang 6.0.1*, стандартная библиотека C++ *libstdc++*;
- Библиотека Boost.Interprocess [**BoostInterprocess**] для управления разделяемой памятью;
- система трассировки событий [**LTNgThesis**] на основе инструмента LTNg [**LTNgSite**].

Научная новизна заключается в предложенных новых методах эффективного межпроцессного взаимодействия в пределах одного физического узла, которые не описаны в существующих исследованиях.

Положения, выносимые на защиту

TBD.

- а) shmem + mux + futex + lf;
- б) shmem + mux + futex + hsha;
- в) shmem + mux + busy wait.

Апробация результатов.

Основные результаты работы были представлены на IX Конгрессе Молодых Ученых.

Предложенные методы межпроцессного взаимодействия на основе разделяемой памяти подготовлены к использованию в платформе для торговли на финансовых рынках Tbricks от компании Itiviti.

ГЛАВА 1. ОБЗОР МЕТОДОВ МЕЖПРОЦЕССНОГО ВЗАИМОДЕЙСТВИЯ

Пример ссылок в рамках обзора: [**example-english**, **example-russian**, **unrestricted-jump-evco**, **nsga-ii-steady-state**]. Вне обзора: [**bellman**].

Выводы по главе 1

В конце каждой главы желательно делать выводы. Вывод по данной главе — нумерация работает корректно, ура!

ГЛАВА 2. АНАЛИЗ МЕТОДОВ МЕЖПРОЦЕССНОГО ВЗАИМОДЕЙСТВИЯ

Выводы по главе 2

В конце каждой главы желательно делать выводы. Вывод по данной главе — нумерация работает корректно, ура!

ГЛАВА 3. РАЗРАБОТКА И РЕАЛИЗАЦИЯ МЕТОДОВ ЭФФЕКТИВНОГО ВЗАИМОДЕЙСТВИЯ ПРОЦЕССОВ

Листинг 1 должен иметь номер 4.

Листинг 1 – Исходный код и флюид `algorithm`

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Рисунок 1 должен иметь номер 2.

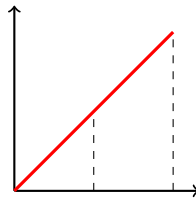


Рисунок 1 – Пример рисунка

Таблица 1 должна иметь номер 3.

Таблица 1 – Таблица умножения с помощью `tabularx` (фрагмент)

–	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34
3	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48	51
4	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68

Выводы по главе 3

В конце каждой главы желательно делать выводы. Вывод по данной главе — нумерация работает корректно, ура!

ГЛАВА 4. ПРОВЕДЕНИЕ ЭКСПЕРИМЕНТА И ОБРАБОТКА РЕЗУЛЬТАТОВ

4.1. Постановка эксперимента

4.1.1. Конфигурация экспериментального стенда

4.1.1.1. Аппаратное обеспечение

Процессор Intel(R) Xeon(R) Platinum 8168 CPU 2.7 GHz. Технология Hyper-Threading отключена для уменьшения влияния на время обработки заявок в системе [LowLatencyHT].

Оперативная память: DDR4-2666 128 GiB.

4.1.1.2. Программное обеспечение

Операционная система Red Hat Enterprise Linux Server release 7.8 (Maipo).
Ядро Linux 3.10.0-1127.el7.x86_64

Компилятор C++ Clang 6.0.1.

Стандартная библиотека C++: libstdc++ 8.1.0.

Библиотека Boost.Interprocess 1.68.0.

4.1.2. Конфигурация экспериментальной системы

Система для проведения эксперимента состоит из двух процессов:

- Процесс-шлюз отвечает за преобразование заявок из формата внешнего мира во внутренний формат системы и обратно.
- Процесс-обработчик совершает некоторые преобразования над заявкой и отправляет результат за пределы системы через процесс-шлюз.

Процессы выполняются на двух процессорах, расположенных в разных разъемах на материнской плате физического узла.

Снаружи системы находится симулятор внешнего мира. Он генерирует поток заявок в систему и получает результат обработки заявки в системе. Схема взаимодействия процессов в эксперименте представлена на Рисунке 2.

В настоящей работе измеряется временная задержка на передачу данных между процессами внутри системы, а именно из процесса-шлюза в процесс-обработчик и обратно (сообщения типа №1 и №2 в запросе и ответе между процессом-шлюзом и процессом обработчиком на Рисунке 2).

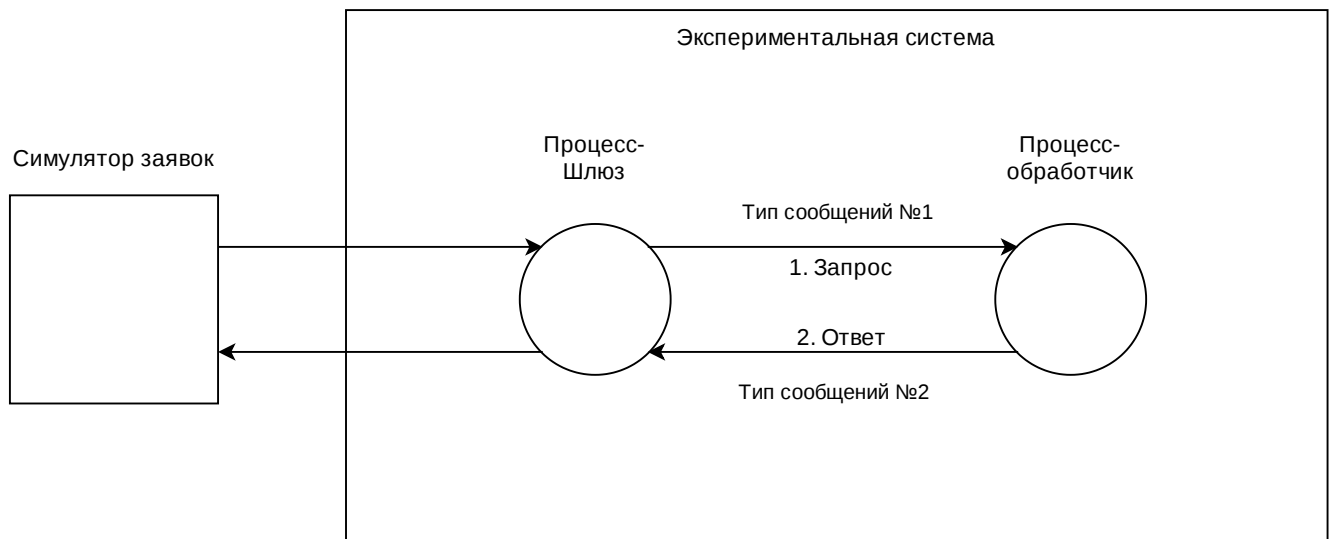


Рисунок 2 – Схема взаимодействия процессов в эксперименте

Процессы системы взаимодействуют используют одно соединение, в рамках которого заявки обрабатываются строго последовательно. Обработка это: получение заявки, обработка заявки и, если необходимо, отправка ответа. Временной задержкой на передачу данных в настоящей работе принимается временной промежуток от начала отправки заявки до **начала обработки заявки**. Таким образом, возможен случай, когда во время обработки очередной заявки процессом в очереди уже находится следующая заявка, временная задержка на передачу которой, таким образом, увеличится на время обработки текущей заявки.

Данный сценарий актуален для процесса-обработчика, в котором обслуживание заявки осуществляется непосредственно в транспортном потоке **TBD: Дать определение транспортному потоку**. В случае с процессом-шлюзом транспортный поток только читает и диспетчеризует асинхронную обработку заявки, т.е. не выполняет обработку самой заявки.

4.1.3. Характер экспериментальной нагрузки

Симулятор отправляет в систему заявки сериями с интервалом в 10мс по 4 заявки в серии с интервалом в несколько десятков микросекунд между ними. Так как для каждого эксперимента производился отдельный запуск симулятора, средние значения обозначенных величин с коэффициентом доверия 95% будут приведены для каждого эксперимента отдельно.

4.1.4. Время обслуживания заявок в процессах

TBD: разобраться с обработкой соединений и обслуживанием заявок. Должно быть одно слово для обозначения одной сущности.

На Рисунке 3 представлена гистограмма времени обслуживания заявок в процессе-обработчике. Время обслуживания заявок с 95% доверительной вероятностью укладывается в диапазон 13 ± 7 мкс.

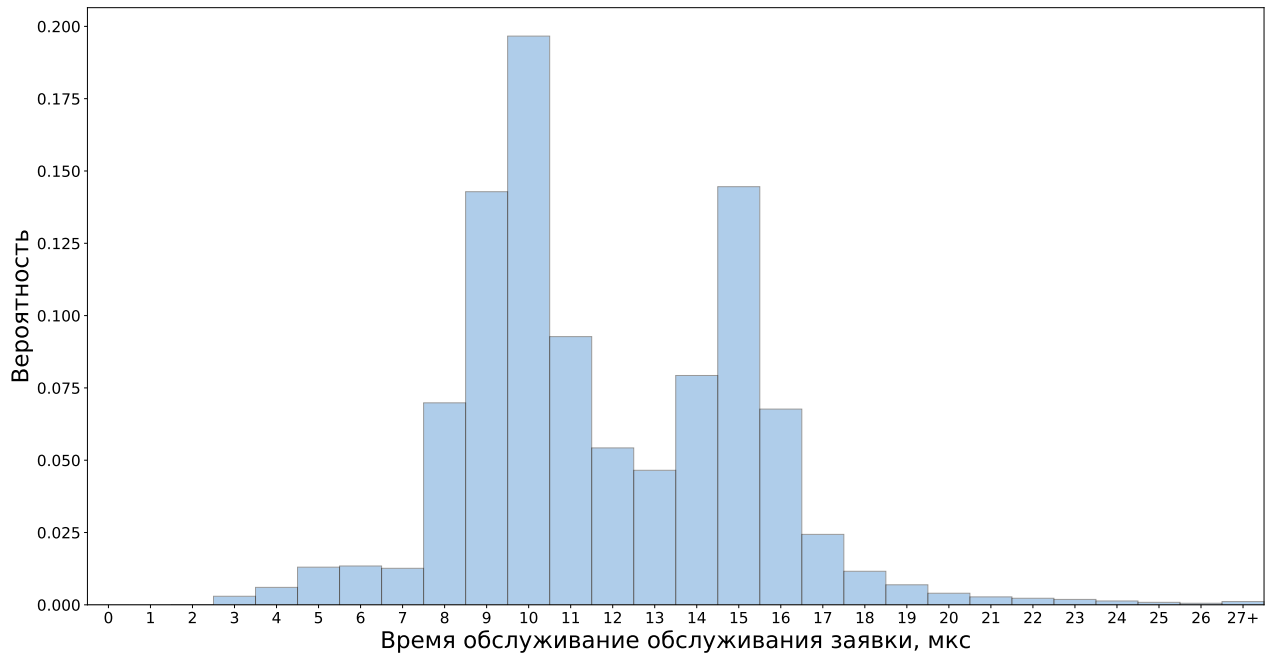


Рисунок 3 – Гистограмма времени обслуживания заявки в процессе-обработчике

На Рисунке 4 представлена гистограмма времени обслуживания заявок в процессе-шлюзе. Время обслуживания заявок с 95% доверительной вероятностью укладывается в диапазон 19 ± 6 мкс.

4.2. Результаты экспериментов

4.2.1. Использование TSP для передачи данных

В качестве точки отсчета в настоящей работе выступает метод межпроцессного взаимодействия на основе TSP, используемый посредством сокетов **TBD: ссылка на определение и на раздел диссертации ??**. Гистограмма временной задержки на передачу данных для данного метода приведена на Рисунке 5.

В данном эксперименте симулятор отправляет серию заявок каждые 10 ± 4 мс с интервалом 50 ± 26 мкс с коэффициентом доверия 95%.

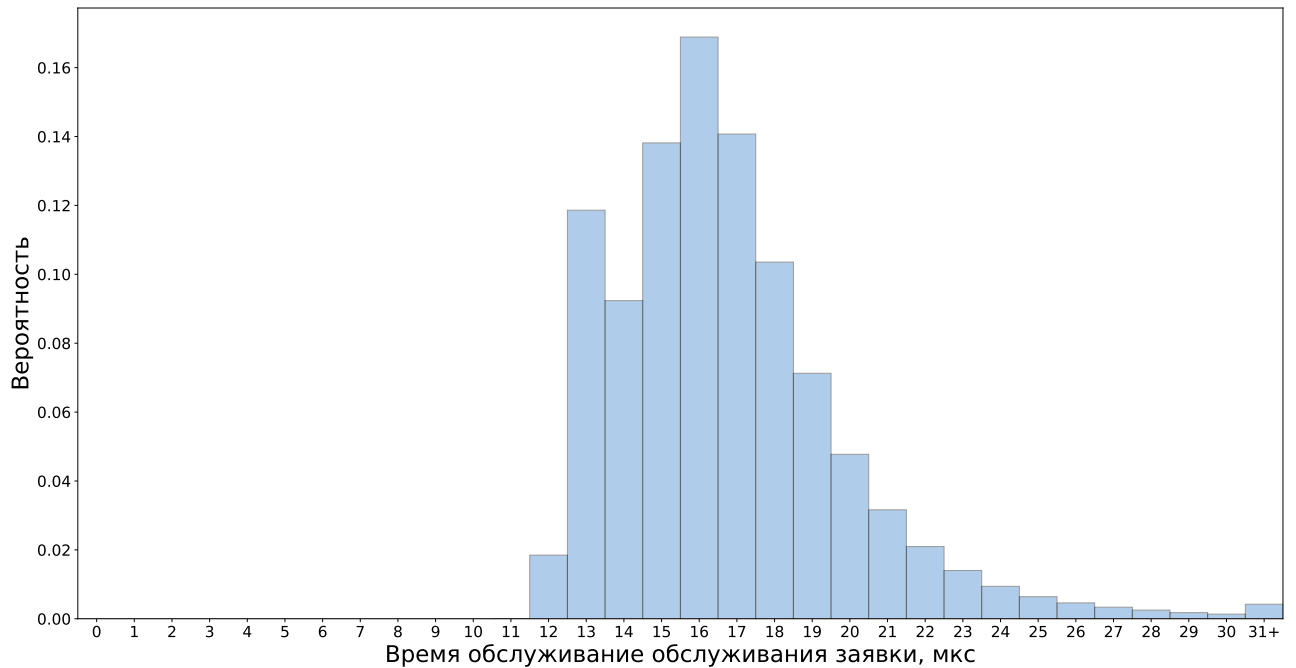


Рисунок 4 – Гистограмма времени обслуживания заявки в процессе-шлюзе

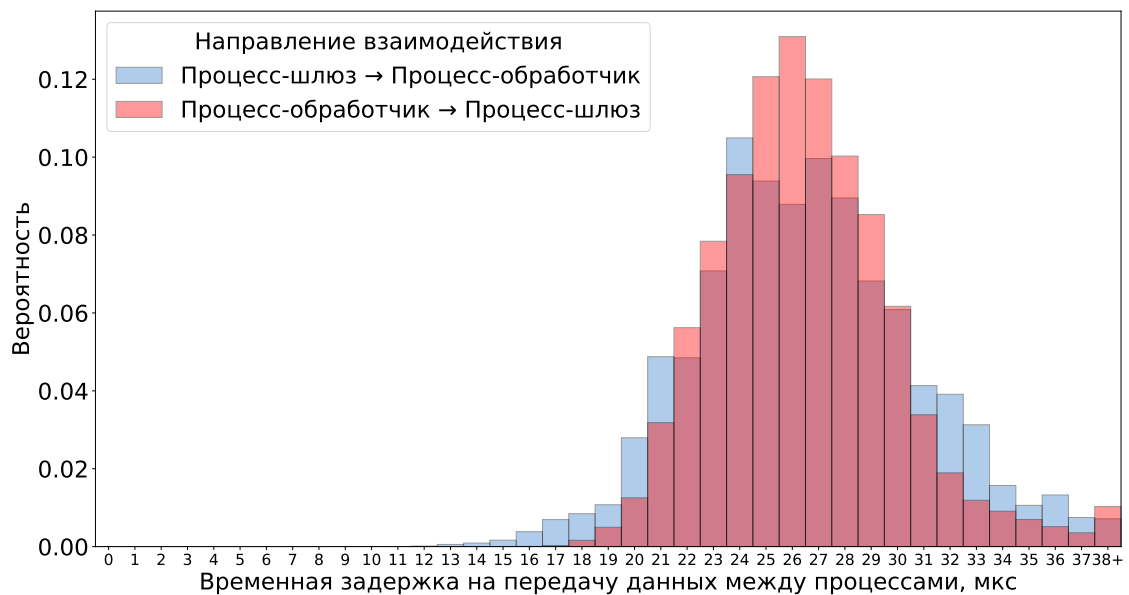


Рисунок 5 – Гистограмма временной задержки на передачу данных между процессами при использовании TCP

В Таблице 2 приведены основные временные характеристики данного метода. Временная задержка на передачу данных в обоих направлениях имеет схожие значения.

Таблица 2 – Основные показатели временной задержки на передачу данных для метода на основе ТСР

Направление взаимодействия/ Показатель	Процесс-шлюз → Процесс-обработчик	Процесс-обработчик → Процесс-шлюз
$\min(t)$, мкс	9	13
$M(t) \pm 95\%$, мкс	27.5 ± 8.5	28 ± 7
$\max(t)$, мс	2.1	9.2
δ между сериями, мс	10 ± 4	10 ± 4
δ между заявками в серии, мкс	50 ± 26	87 ± 32

4.2.2. Использование разделяемой памяти для передачи данных

4.2.2.1. Использование ТСР для оповещения о появлении данных

В данном эксперименте симулятор отправляет серию заявок каждые 10 ± 4 мс с интервалом 50 ± 27 мкс с коэффициентом доверия 95%.

В данном подразделе приведены данные об экспериментах с методом меж-процессного взаимодействия, описанным в Разделе ??.

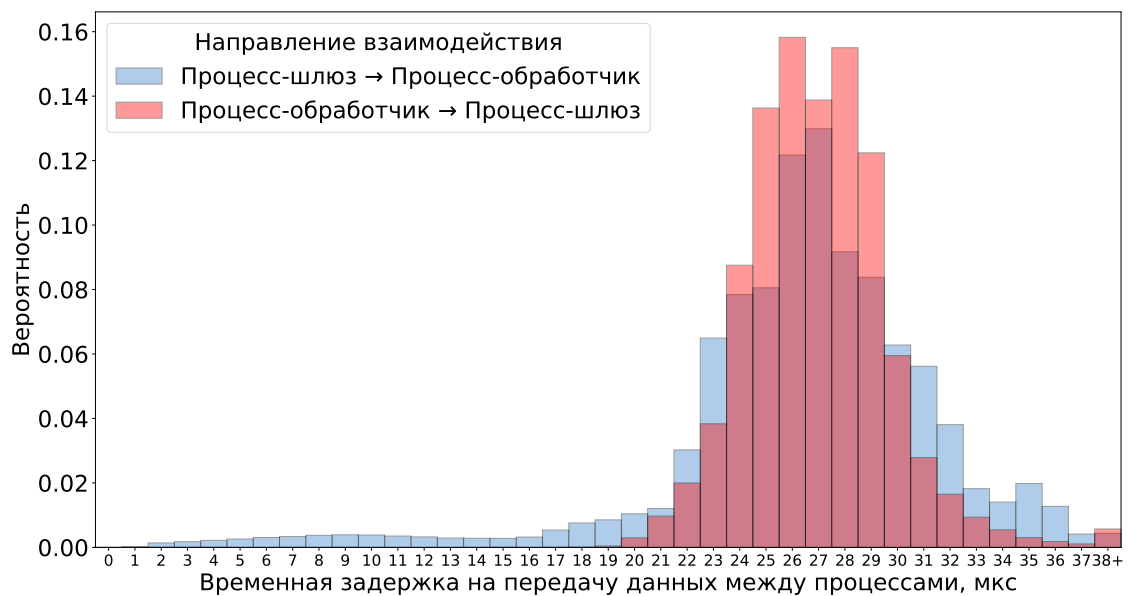


Рисунок 6 – Гистограмма временной задержки на передачу данных между процессами при использовании разделяемой памяти для передачи данных и ТСР для оповещения о появлении данных в ней

В Таблице 3 приведены основные временные характеристики данного метода. Межпроцессное взаимодействие в сторону процесса-обработчика работает быстрее, т.к. в среднем время обслуживания заявок в процессе-обработчике заметно меньше (см. Рисунок 3), чем скорость поступления новых заявок, что позволяет использовать оптимизацию, описанную в Разделе ?? TBD: сослаться на оптимизацию про отсутствие необходимости отправки новых сигналов, когда старые еще не были обработаны..

Таблица 3 – Основные показатели временной задержки на передачу данных для метода, использующего разделяемую память для передачи данных и ТСР для оповещения о появлении данных в ней

Направление взаимодействия/ Показатель	Процесс-шлюз → Процесс-обработчик	Процесс-обработчик → Процесс-шлюз
$\min(t)$, мкс	1	3
$M(t) \pm 95\%$, мкс	22.5 ± 12.5	27.5 ± 5.5
$\max(t)$, мс	3	9ю8
δ между сериями, мс	10 ± 4	10 ± 4
δ между заявками в серии, мкс	50 ± 27	87 ± 32

4.2.2.2. Использование мультиплексора в разделяемой памяти для оповещения о появлении данных

В данном подразделе приведены данные об экспериментах с методом межпроцессного взаимодействия, описанным в Разделе ??.

Блокирующие методы В блокирующих методах поток мультиплексора событий использует примитив *futex* **Может, сослаться на определение futex?** для пассивного ожидания новых сигналов (см. Раздел ??).

Диспетчеризация и обработка соединений по модели "Полусинхронный/Полуреактивный" В данном эксперименте симулятор отправляет серию заявок каждые 10 ± 4 мс с интервалом 51 ± 28 мкс с коэффициентом доверия 95%.

В данном подразделе приведены данные об экспериментах с методом межпроцессного взаимодействия, описанным в Разделе ??.

В Таблице 4 приведены основные временные характеристики данного метода. **TBD**

На Рисунке 7 приведена плотность вероятности временной задержки на передачу данных для данного метода.

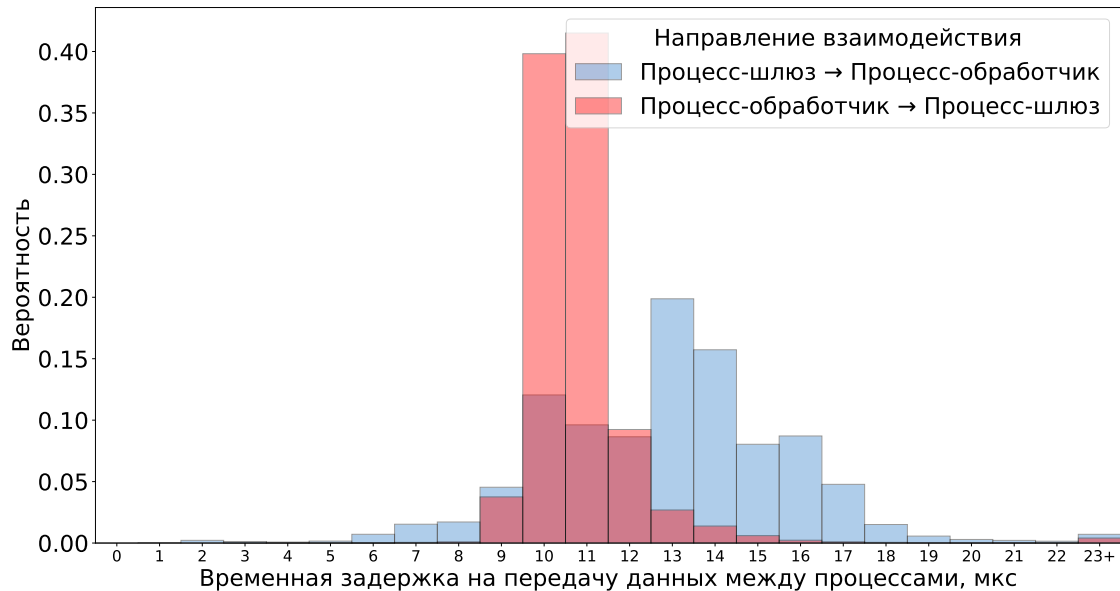


Рисунок 7 – Гистограмма временной задержки на передачу данных между процессами для метода, использующего разделяемую память для передачи данных, блокирующий мультиплексор в разделяемой памяти и модель ”Полусинхронный/Полуреактивный“ при обслуживании заявок

Таблица 4 – Основные показатели временной задержки на передачу данных между процессами для метода, использующего разделяемую память для передачи данных, блокирующий мультиплексор в разделяемой памяти и модель ”Полусинхронный/Полуреактивный“ при обслуживании заявок

Направление взаимодействия/ Показатель	Процесс-шлюз → Процесс-обработчик	Процесс-обработчик → Процесс-шлюз
$\min(t)$, мкс	1	3
$M(t) \pm 95\%$, мкс	12.5 ± 5.5	11.5 ± 2.5
$\max(t)$, мс	6.9	11.6
δ между сериями, мс	10 ± 4	10 ± 4
δ между заявками в серии, мкс	51 ± 28	91 ± 36

Диспетчеризация и обработка соединений по модели "Лидер/Последователи" В данном эксперименте симулятор отправляет серию заявок каждые 8.4 ± 5.3 мс с интервалом 52 ± 28 мкс с коэффициентом доверия 95%.

В данном подразделе приведены данные об экспериментах с методом меж-процессного взаимодействия, описанным в Разделе ??.

В Таблице 5 приведены основные временные характеристики данного метода. **TBD**

На Рисунке 8 приведена плотность вероятности временной задержки на передачу данных для данного метода.

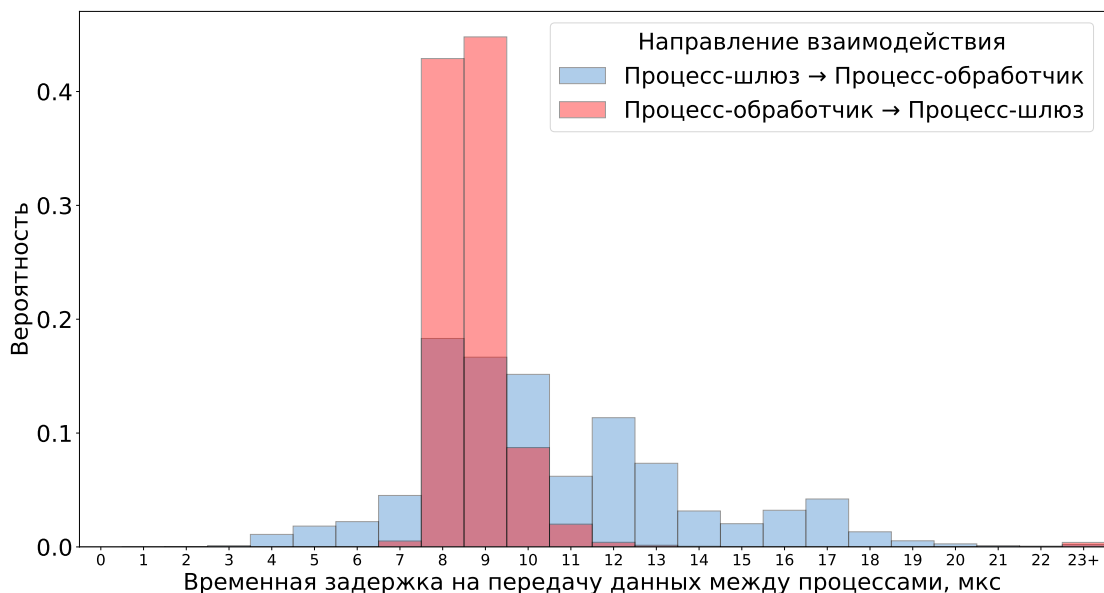


Рисунок 8 – Гистограмма временной задержки на передачу данных между процессами для метода, использующего разделяемую память для передачи данных, блокирующий мультиплексор в разделяемой памяти и модель "Лидер/Последователи" при обслуживании заявок

Неблокирующий метод В неблокирующем методе поток мультиплексора событий метод активного ожидания новых сигналов (см. Раздел ??). В данном параграфе рассматривается исключительно модель обслуживания заявок "Лидер/Последователи" т.к. в параграфе выше она показала лучшие результаты по сравнению с моделью "Полусинхронный/Полуреактивный".

Таблица 5 – Основные показатели временной задержки на передачу данных между процессами для метода, использующего разделяемую память для передачи данных, блокирующий мультиплексор в разделяемой памяти и модель ”Лидер/Последователи“ при обслуживании заявок

Направление взаимодействия/ Показатель	Процесс-шлюз → Процесс-обработчик	Процесс-обработчик → Процесс-шлюз
$\min(t)$, мкс	1	2
$M(t) \pm 95\%$, мкс	11.5 ± 6.5	9.5 ± 1.5
$\max(t)$, мс	2.4	9.5
δ между сериями, мс	8.4 ± 5.3	8.4 ± 5.3
δ между заявками в серии, мкс	52 ± 29	90 ± 35

Диспетчеризация и обработка соединений по модели ”Лидер/Последователи” В данном эксперименте симулятор отправляет серию заявок каждые 7.4 ± 5.9 мс с интервалом 55 ± 32 мкс с коэффициентом доверия 95%.

В данном подразделе приведены данные об экспериментах с методом меж-процессного взаимодействия, описанным в Разделе ??.

В Таблице 6 приведены основные временные характеристики данного метода. **TBD**

На Рисунке 9 приведена плотность вероятности временной задержки на передачу данных для данного метода.

Выводы по главе 4

Вывод по очередной главе

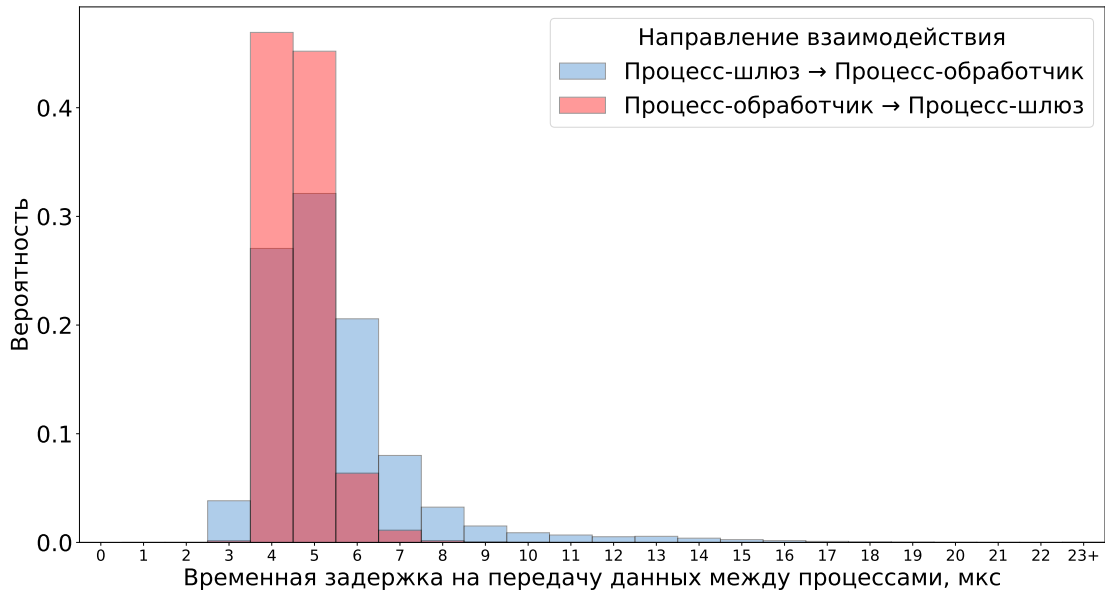


Рисунок 9 – Гистограмма временной задержки на передачу данных между процессами для метода, использующего разделяемую память для передачи данных, блокирующий мультиплексор в разделяемой памяти и модель ”Лидер/Последователи“ при обслуживании заявок

Таблица 6 – Основные показатели временной задержки на передачу данных между процессами для метода, использующего разделяемую память для передачи данных, блокирующий мультиплексор в разделяемой памяти и модель ”Лидер/Последователи“ при обслуживании заявок

Направление взаимодействия/ Показатель	Процесс-шлюз → Процесс-обработчик	Процесс-обработчик → Процесс-шлюз
$\min(t)$, мкс	1	3
$M(t) \pm 95\%$, мкс	7 ± 4	5 ± 1
$\max(t)$, мс	8.5	0.003
δ между сериями, мс	7.4 ± 5.9	7.4 ± 5.9
δ между заявками в серии, мкс	58 ± 35	167 ± 113

ЗАКЛЮЧЕНИЕ

В данном разделе размещается заключение.

ПРИЛОЖЕНИЕ А. ПРИМЕР ПРИЛОЖЕНИЯ

**ПРИЛОЖЕНИЕ Б. ЕЩЕ ОДИН ПРИМЕР ПРИЛОЖЕНИЯ С
НЕИМОВЕРНО ДЛИННЮЩИМ НАЗВАНИЕМ ДЛЯ ТЕСТИРОВАНИЯ
ПЕРЕНОСОВ**

ПРИЛОЖЕНИЕ В. ПРИМЕР ОГРОМНОГО ЛИСТИНГА