МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

СОГЛАСОВАНО
Генеральный директор
ЗАО «АйТи»

_____ Бакиев О.Р.
"28" декабря 2011 г.

УТВЕРЖДАЮ
Ректор НИУ ИТМО

_____ Васильев В.Н.
"29" декабря 2011 г.

МНОГОПРОФИЛЬНАЯ ИНСТРУМЕНТАЛЬНО-ТЕХНОЛОГИЧЕСКАЯ ПЛАТФОРМА СОЗДАНИЯ И УПРАВЛЕНИЯ РАСПРЕДЕЛЕННОЙ СРЕДОЙ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ CLAVIRE

ПРИКЛАДНОЙ СЕРВИС АНАЛИЗА ПРОЦЕССОВ РАСПРОСТРАНЕНИЯ ИНФОРМАЦИИ В ВИРТУАЛЬНЫХ СООБЩЕСТВАХ В ИНТЕРНЕТ, СВЯЗАННЫХ С ПОТРЕБЛЕНИЕМ НАРКОТИКОВ

ОПИСАНИЕ ПРОГРАММЫ

ЛИСТ УТВЕРЖДЕНИЯ

RU.СНАБ.80066-06 13 56-ЛУ

Представители
Организации-разработчика

Руководитель разработки,
профессор НИУ ИТМО

_____ Бухановский А.В.
"28" декабря 2011 г.

Ответственный исполнитель,
с.н.с. НИУ ИТМО

_____ Луценко А.Е.
"28" декабря 2011 г.

Нормоконтролер
ведущий инженер НИУ ИТМО

_____ Позднякова Л.Г.
"28" декабря 2011 г.

2011

Инв.№ подл.	Подп. и дата	Взам.инв.№	Инв.№ дубл.	Подп. и дата

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

УТВЕРЖДЕН

RU.СНАБ.80066-06 13 56-ЛУ

МНОГОПРОФИЛЬНАЯ ИНСТРУМЕНТАЛЬНО-ТЕХНОЛОГИЧЕСКАЯ ПЛАТФОРМА СОЗДАНИЯ И УПРАВЛЕНИЯ РАСПРЕДЕЛЕННОЙ СРЕДОЙ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ CLAVIRE

ПРИКЛАДНОЙ СЕРВИС АНАЛИЗА ПРОЦЕССОВ РАСПРОСТРАНЕНИЯ ИНФОРМАЦИИ В ВИРТУАЛЬНЫХ СООБЩЕСТВАХ В ИНТЕРНЕТ, СВЯЗАННЫХ С ПОТРЕБЛЕНИЕМ НАРКОТИКОВ

ОПИСАНИЕ ПРОГРАММЫ

RU.СНАБ.80066-06 13 56

ЛИСТОВ 25

2011

Инв.№ подл.	Подп. и дата	Взам.инв.№	Инв.№ дубл.	Подп. и дата

АННОТАЦИЯ

Документ содержит описание прикладного сервиса позволяющего искать в сети Livejournal неявные сообщества людей, объединенных единой темой интересов, и определять скорости распространения информации в их сети дружбы. В работе описывается поиск людей, заинтересованных в теме наркотиков. Прикладной сервис реализует композитное приложение для многоцелевой инструментально-технологической платформы (МИТП) CLAVIRE, которое позволяет определять скорость распространения слухов в сети людей заинтересованных в теме наркотиков. Прикладной сервис разработан в ходе выполнения проекта «Создание распределенной вычислительной среды на базе облачной архитектуры для построения и эксплуатации высокопроизводительных композитных приложений» (Договор № 21057 от 15 июля 2010 г., шифр 2010-218-01-209) в рамках реализации постановления Правительства РФ № 218 «О мерах государственной поддержки развития кооперации российских высших учебных заведений и организаций, реализующих комплексные проекты по созданию высокотехнологичного производства».

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ

Прикладной сервис (ПС) анализа процессов распространения информации в виртуальных сообществах в Интернет, связанных с потреблением наркотиков RU.СНАБ.80066-06 01 56 реализует композитное приложение, которое позволяет, основываясь на заданных семантических описаниях темы наркотиков, осуществлять поиск пользователей сети Livejournal, соответствующих этим описаниям, и рассчитывать скорость распространения слухов в найденной сети. ПС разработан в ходе выполнения проекта «Создание распределенной вычислительной среды на базе облачной архитектуры для построения и эксплуатации высокопроизводительных композитных приложений» (Договор № 21057 от 15 июля 2010 г., шифр 2010-218-01-209) в рамках реализации постановления Правительства РФ № 218 «О мерах государственной поддержки развития кооперации российских высших учебных заведений и организаций, реализующих комплексные проекты по созданию высокотехнологичного производства».

ПС функционирует в рамках распределенной среды облачных вычислений под управлением многофункциональной инструментально-технологической платформы (МИТП) CLAVIRE RU.СНАБ.80066-06. Он разработан на предметно-ориентированном языке EasyFlow описания композитных приложений.

ПС использует следующие пакеты прикладных программ, доступные в распределенной среде под управлением МИТП (см. также раздел 3.2):

Hadrawler - прикладной пакет для осуществления сбора данных о пользователях сети Livejournal на основе семантического описания текстов;

ISM - прикладной пакет распространения информации по сети с заданной топологией;

SciLab - многоцелевой пакет компьютерной математики и визуализации (заимствуется).

Перечисленные пакеты описываются на языке EasyFlow и регистрируются в базе пакетов МИТП.

2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

2.1. Область применения

ПС предназначен для сбора информации о структуре реальной социальной сети Livejournal [1] [2] и изучения процессов распространения слухов на основе аппарата

комплексных сетей [3] [4]. Отличительной особенностью ПС является поиск информации о неявных сообществах пользователей, то есть множестве пользователей объединенных общим признаком, на основе семантического описания текстов, которое задается посредством словаря ключевых слов [5]. Например, для поиска пользователей, заинтересованных в теме наркотиков, задается словарь терминов, использующихся в среде людей употребляющих наркотики и людей им противодействующих, на его основе строится алгоритм позволяющий классифицировать тексты на относящиеся к этой области и не относящиеся. При наличии хотя бы одного текста выбранной тематики пользователь так же помечается, как заинтересованный в ней и все друзья пользователя заносятся в список для посещения краулером – программой осуществляющий обход социальной сети и сбор данных о ее структуре и данных хранимых в узлах. При завершении работы краулера происходит анализ найденной им сети на предмет скорости распространения слухов. Распространение слухов сети характеризуется статистическими свойствами сети, и исследование этих процессов может обеспечить поддержку принятия решений при планировании противодействия распространения противозаконной информации в этой группе людей.

2.2. Функциональное назначение

ПС предназначен для решения следующих задач:

- Сбор данных о структуре сети – граф дружбы пользователей сети;
- Сбор индивидуальных данных пользователей сети – интересы пользователей и их посты;
- Интеллектуальный обход сети, при котором обходятся неявные сообщества пользователей – пользователей заинтересованных в одной теме и «дружащих» друг с другом;
- Моделирование процесса распространения информации по социальной сети, исходя из заданной вероятности передачи слуха и вероятности перехода в пассивное состояние нераспространения оного (потеря интереса, забытый слух);
- Расчет интервальных характеристик (минимальные и максимальные порядковые статистики, медиана, размах выборки) при заданных параметрах процесса моделирующего распространения слухов.

2.3. Ограничения на применение

Используемые в ПС методы определения тематики документов подходят для тем, описывающихся узкоспециализированными словами, соответствующих одной семантической теме. ПС применим для моделирования распространения слухов, характеризуемых однородным течением, при котором невозможно повторное распространение слухов индивида при переходе в пассивное состояние.

3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

3.1. Структура композитного приложения

Композитное приложение, реализуемое ПС, предназначено, на основе заданных семантических описаниях темы наркотиков, осуществлять поиск пользователей сети Livejournal, соответствующих этим описаниям, и рассчитывать скорость распространения слухов в найденной сети. Для этого необходимо осуществить краулинг сети, при котором происходит классификация текстов пользователей на тексты посвященные наркотикам и не посвященные, для чего используется составленный специалистом словарь терминов. При наличии у пользователя хотя бы одного текста, посвященного наркотика, друзья данного пользователя заносятся в специальный список, который используется краулером при выборе следующих узлов для посещения. Таким образом, реализуется тематический краулинг сети. Затем найденная сеть исследуется на предмет скорости распространения слухов. Структура композитного приложения показана на рис. 3.1.

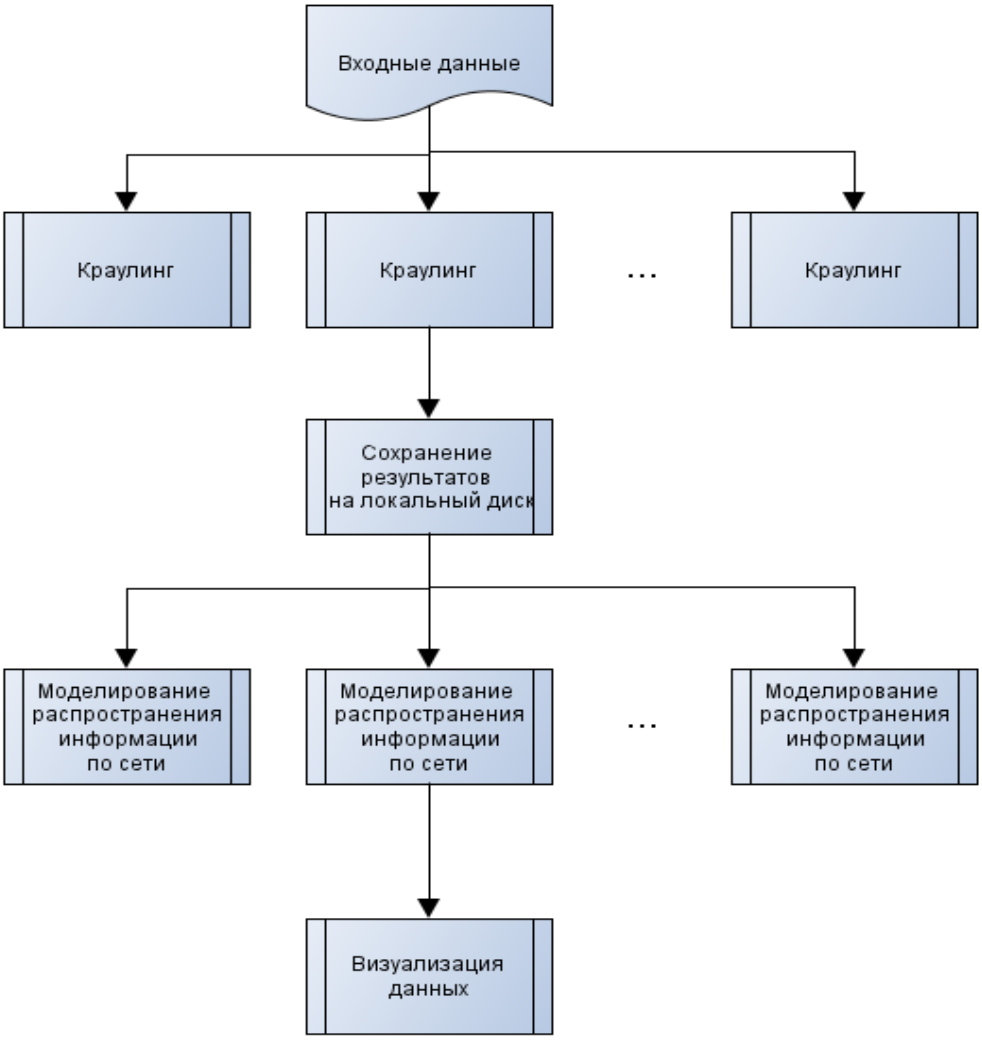

Рисунок 3.1 – Структура композитного приложения

ПС использует следующие пакеты прикладных программ, доступные в распределенной среде под управлением МИТП (см. также раздел 3.2):

- Hadrawler - прикладной пакет для осуществления сбора данных о пользователях сети Livejournal на основе семантического описания текстов;
- ISM - прикладной пакет распространения информации по сети с заданной топологией;
- SciLab - многоцелевой пакет компьютерной математики и визуализации (заимствуется).

В листинге 3.1 приведен скрипт композитного приложения на языке EasyFlow, а на рис. 3.2 - результат его интерпретации в МИТП, который создает очередь из четырех последовательных процессов запуска пакетов в виде AWF.

Листинг 3.1 Описание композитного приложения на языке EasyFlow

```
require inKeywords, inSeeds;
require spreadConfFile;

//Start network crawling
step HadrawlerCrawl runs hadrawler (
        crawl = true,
        inKeyWords = inKeywords,
        inUsers = inSeeds,
        depth = 30
);

//Dump crawled network
step HadrawlerDump runs hadrawler after HadrawlerCrawl (
        dumpGraph = true
);

//Information spread Modelling
step InfSpreadModelling runs ism after HadrawlerDump (
        inDataFile = sweep [HadrawlerDump.Result.outs["output.dat"],
                            HadrawlerDump.Result.outs["output.dat"],
                            HadrawlerDump.Result.outs["output.dat"],
                            HadrawlerDump.Result.outs["output.dat"]],
        inConfigFile = spreadConfFile
);

//Result visualization
step Visualization runs scilab after InfSpreadModelling(
        script_name = "ISM_chart.sce",
        input_folder = InfSpreadModelling.Result.sweep_outs["output.dat"],
        output_file = "chart",
        output_ext = "png"
);
```

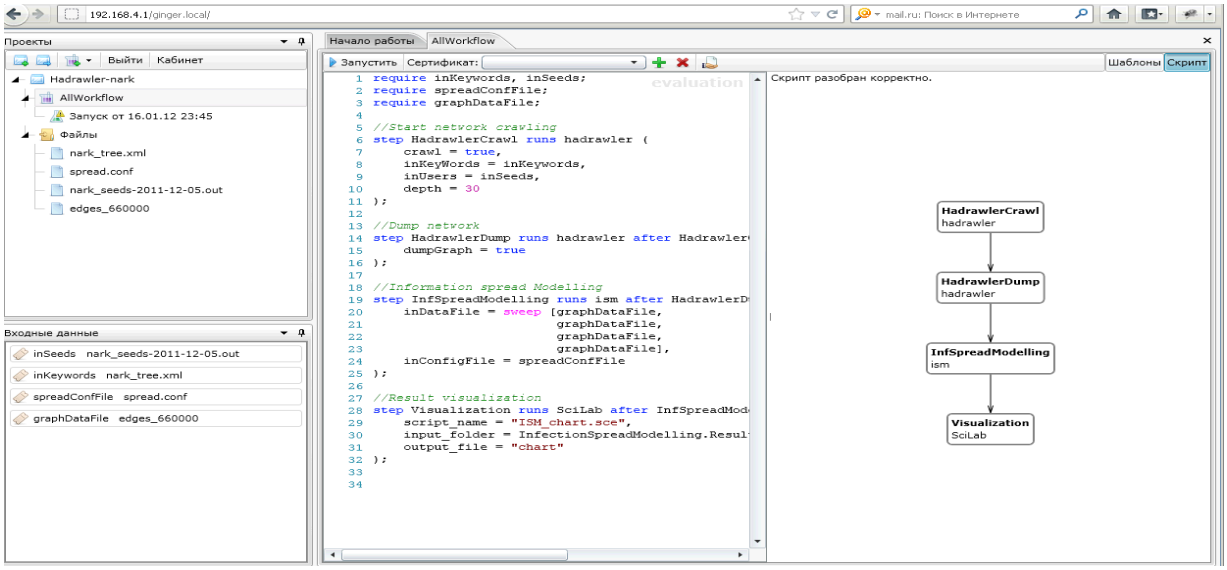

Рисунок 3.2 - Подготовка композитного приложения к запуску на выполнение в МИТП

3.2. Характеристики модулей композитного приложения

3.2.1. Прикладной пакет Hadrawler для обхода социальной сети

Прикладной пакет Hadrawler – программный компонент для осуществления сбора информации (краулинга) социальной сети Livejournal. В основе этого компонента лежит программа краулер, реализованная на базе библиотеки с открытым исходным кодом Apache Hadoop. При реализации краулера на основе библиотеки Hadoop мы возлагаем на нее все обязанности по запуску, мониторингу и контролю распределенных операций, тем самым концентрируясь на непосредственной реализации задач решаемых краулером. При такой архитектуре каждый из логических модулей архитектуры краулера (рисунок 3.3) представляет собой отдельное задание в терминах библиотеки Hadoop. В этой схеме каждый из модулей читает некоторую информацию из распределенной базы данных, обрабатывает ее и записывает результат обратно, после чего он передает управление следующему модулю. Модули выполняются последовательно друг за другом, однако каждый из них будет выполняться параллельно на нескольких машинах кластера. Таким образом, можно говорить, что процесс краулинга является итерационным процессом.

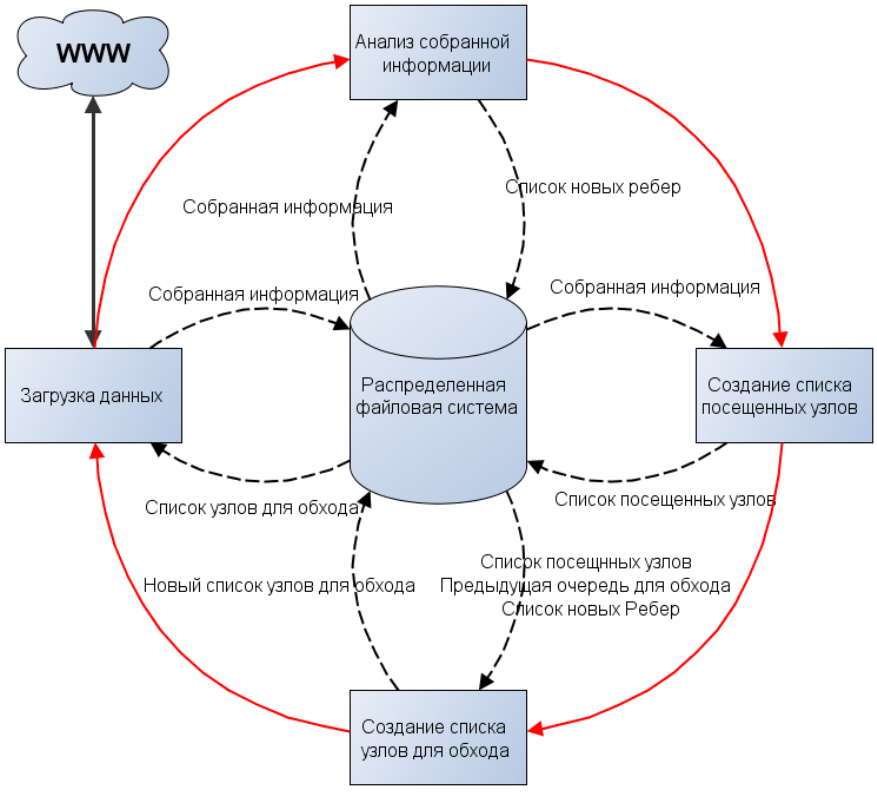

Рисунок 3.3 - Архитектура краулера, построенного на основе библиотеки ApacheHadoop

Краулер работает итерационно: на каждой итерации он посещает фиксированное число пользователей, после чего происходит анализ собранной информации. На рисунке 3.4 приведена схема работы краулера на одной итерации (эллипсы – процессы, проводимые над данными, а прямоугольники – данные). Процессы выполняются последовательно друг за другом, и их взаимное расположение по оси абсцисс указывает на порядок выполнения.

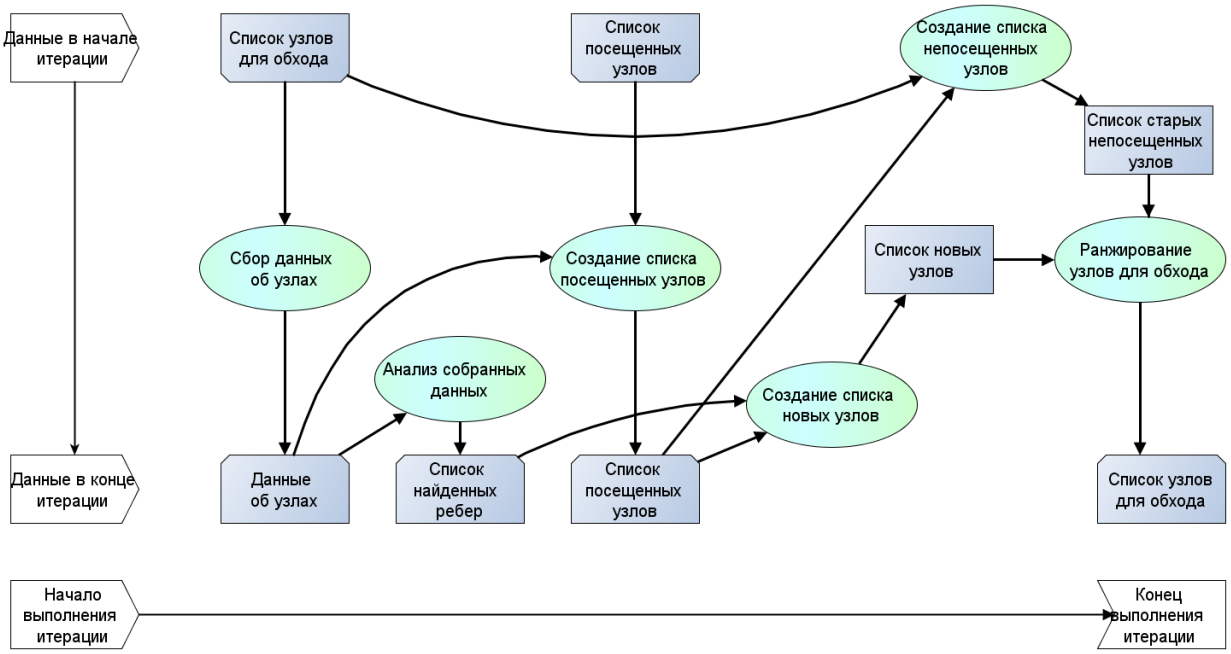

Рисунок 3.4 – Схема работы краулера на одной итерации

На первом этапе работы краулера создается список новых узлов, найденных на каждой итерации. Для этого анализируются все ребра, найденные на итерации, и определяется, что они ведут в еще не посещенные узлы, для чего в память загружается список посещенных узлов. На втором этапе из списка узлов для обхода, построенного на предыдущей итерации, отфильтровываются узлы, которые еще не были посещенные. Если обнаружилось, что какой-либо узел не посещен вследствие ошибки, счетчик числа попыток посещения увеличивается. Затем оба списка узлов объединяются, при этом происходит обновление атрибутов узла (числа попыток, затраченных на его посещение и приоритет узла), происходит удаление из очереди узлов, для которых счетчик числа попыток превысил заданное значение. Далее полученный список узлов ранжируется согласно значению приоритета и становится списком узлов для обхода на новой итерации.

Приоритет узла считается равным числу найденных ссылок, указывающих на него. Чем больше ссылок, тем выше приоритет и тем вероятнее, что узел содержит информацию

по заданной тематике. Число ссылок легко поддержать в актуальном состоянии в узле от итерации к итерации, оно равно сумме ссылок, найденных на предыдущих итерациях, и числу ребер, найденных на текущей итерации, ведущих в этот узел.

Для реализации тематической политики обхода краулера, используется подход, основанный на анализе контекста пользователя социальной сети: его интересов и его документов, в предположении, что люди, интересующиеся схожими темами, образуют неявные сообщества в социальной сети, и между ними существуют связи. Таким образом, обход сети выглядит следующим образом.

1) Анализ интересов и/или документов пользователя и проверка их соответствия заданным темам. Интересы и документы пользователя образуют контекст.
2) Если контекст пользователя отнесен к определенной теме, то добавить его "друзей" в очередь узлов для обхода.
3) Если контекст не отнесен к определенной теме, то продолжить работу.

Для содержательного анализа текстов и интересов пользователя используются методы, основанные на применении списка ключевых слов, которые достаточно полно описывают некоторую тему, например, тему наркотизации населения. Для этого эксперт предметной области составляет список ключевых слов – наиболее известные и однозначные термины, которые четко определяют тематику текста, а также различные многозначные жаргонизмы, что вносит определенную погрешность. Отметим, что каждое ключевое слово семантически отнесено к некоторой более конкретной теме. Так для темы наркотиков, экспертом были выделены следующие ключевые слова и словосочетания: шприц, иглы, первитин, эфедрон, героин, марихуана, кокаин, опий-сырец, таблетки, инъекции, сленг приготовления, общие слова (12 слов).

Учитывая морфемное многообразие, эксперт предметной области указывает только паттерн ключевых слов. Выделены следующие паттерны слов:

1) Точная (конкретная) форма слова – например, паттерн «доза».
2) Префикс слова – фиксируется только начало слова, суффиксы и окончания могут быть любыми. Например, паттерн: “наркот*”, под который попадают «наркотики», «наркота».
3) Суффикс слова – фиксируется окончание слова, приставки могут быть любыми. Например, паттерн “*курить”, под который попадают: «покурить», «вкурить».
4) Задана середина слова – и приставки, и суффиксы могут варьировать. Например, паттерн «*колбасит*».

5) Стемминг слова [6] – посредством различных эвристик, специфичных для каждого языка, от слова отбрасываются окончания и суффиксы, и получается словоформа, близкая по своей структуре к объединению приставки и корня слова. Данные эвристики не всегда имеют высокую точность, но обладают быстродействием, что в рассматриваемом случае является критичным параметром. Существуют различные наборы эвристик для каждого языка, нами было реализовано семейство эвристик, называемых стеммер Портера.

В результате в документе проверяется наличие ключевых слов и фраз – неупорядоченных наборов ключевых слов, которые свидетельствуют о тематической принадлежности текста. При наличии хотя бы одного ключевого слова или фразы считается, что документ посвящен теме наркотиков, что инициирует обход «друзей» текущего пользователя. Данный подход модернизируется за счет введения весов: для окончательного определения принадлежности документа определенной тематике, высчитывается его вес, равный сумме весов всех входящих в него ключевых слов и ключевых фраз. Затем полученный вес сравнивается с пороговой величиной: полагается, что все документы с весом больше этого порога имеют заданную тематику. Пользователь с большим (относительно порога) весом документа добавляется в очередь для обхода.

Для запуска различных функций пакета реализован скрипт, запускающий необходимые операции на кластере машин работающих под управлением библиотеки Hadoop.

Приложение состоит из множества пакетов и классов, каждый из которых выполняет специфические функции. Ниже приведен список основных классов, реализующих функции пакета.

- `ru.ifmo.hadrawler.`Crawler – класс, выполняющий итерации краулера.
- `ru.ifmo.hadrawler.`fetcher.LJFetcher – класс, реализующий скачивание информации о пользователе из сети Livejournal.
- `ru.ifmo.hadrawler.`links.LinkExctractor – класс, выполняющий фильтрацию пользователей на основе тематики их записей и создающий список новых ребер.
- `ru.ifmo.hadrawler.`frontier.`Frotier` – класс, создающий список посещенных узлов и новую очередь узлов для обхода
- `ru.ifmo.hadrawler.` frontier.VisitInfoCreator – класс, создающий список посещенных узлов.

- `ru.ifmo.hadrawler.` frontier.QueueCreator – класс, создающий новую очередь для обхода краулера

Для работы краулера требуются следующие библиотеки с открытым исходным кодом: java colt, fastutil, apache httpclient, jetbrains annotations, jakarta commons, joda-time, jsoup, log4j, piccolo, yandex bolts.

Для запуска пакета Hadrawler требуется в консольном приложении выполнить команду:

`Crawler.bat` <параметры вызова>

В качестве выходных данных Hadrawler создает в распределенной файловой системе Apache Hadoop файлы, содержащие информацию о посещенных им узлах, их атрибутах и связях между ними. Помимо этого пакет позволяет выгрузить эти данные на локальный диск в удобном для анализа формате.

3.2.2. Прикладной пакет ISM моделирования распространения информации

Прикладной пакет ISM – программный компонент моделирования процесса распространения информации по сети. Для моделирования процесса распространения слухов используется модель Далея-Кендалла [4] [7].

Переменные, необходимые для описания алгоритма ДК:

I – множество вершин, которым не известна информация.

S – множество вершин, которые владеют информацией.

R – множество вершин, которые владеют информацией но не передают ее дальше.

l, a – конфигурационные параметры.

k – минимальная степень вершины, с которой начинается распространение информации.

Алгоритм моделирования процесса распространения информации ДК состоит пошагового расчета вероятности передачи информации далее. Формально эту процедуру для каждого шага можно описать следующим образом:

- $I(i) + S(j) \xrightarrow{\lambda} S(i) + S(j)$
- $S(i) + S(j) \xrightarrow{\alpha} R(i) + S(j)$
- $S(i) + R(j) \xrightarrow{\alpha} R(i) + R(j)$,

где i и j – соседние вершины.

Пакет ISM разработан в виде консольного приложения с использованием языка программирования Java версии 1.6. ISM может функционировать на всех операционных системах, на которых может быть установлена платформа Java JRE 1.6.

Приложение состоит из нескольких пакетов и классов, каждый из которых выполняет специфические функции. Ниже приведены наименования классов, и описание их функций:

- `ru.ifmo.hpc.structure.Node` – класс, описывающий узлы сети.
- `ru.ifmo.hpc.structure.Graph` – класс, представляющий структуру сети, хранит информацию об узлах и ребрах между ними в виде списков смежности.
- `ru.ifmo.hpc.util.DataIO` – класс, реализующий статические методы для импорта сети и параметров работы алгоритма, а так же для экспорта результатов моделирования.
- `ru.ifmo.hpc.main.algorithm.SpreadProcess` – класс, реализующий алгоритм ДК.
- `ru.ifmo.hpc.main.SpreadModel` – класс, являющийся точкой входа программы. Производит считывание данных и вызывает методы описанных выше классов, реализующих соответствующие шаги алгоритма.

Для обработки параметров запуска применяется свободная библиотека Commons CLI.

Для запуска пакета ISM требуется в консольном приложении выполнить команду:

```
java –jar ISM.jar <параметры вызова>
```

В качестве выходных данных ISM предоставляет описание начальных условий работы алгоритма, а так же описание каждого шага распространения вируса по сети, включая детализацию по классам вершин.

3.3. Описание модулей композитного приложения в МИТП

3.3.1. Прикладной пакет Hadrawler для обхода социальной сети

Для интеграции пакета Hadrawler в МИТП необходимо описать его взаимодействия с платформой, опираясь на формат представления входных и выходных данных, на языке EasyPackage. Платформенный скрипт описания пакета Hadrawler позволяет определить уровень абстракции и интерпретации в работе с входными и выходными параметрами, что обеспечивает гибкость и упрощение процедуры взаимодействия пользователя с платформой на этапе запуска задания. Платформенный скрипт описания пакета Hadrawler определяется в соответствии с листингом 3.1.

Листинг 3.1 Скрипт описания пакета Hadrawler

```
name "hadrawler"
#version nil
#display "hadrawler"
#vendor "Itmo"
#url "http://escience.ifmo.ru/"
#license "GPLv3"
#description "Crawler is used to gathering user profiles in the social
networks."
#logo ""

inputs {
      public param {
        name "crawl"
        display "Start crawling for data mining"
        type bool
        default false
    }

      public param {
        name "depth"
        display "Depth of crawling"
        type int
        default 3
    }

      public param {
        name "dumpGraph"
        display "Start crawler for dumping result graph"
        type bool
            default false
    }

      public file {
             name "inUsers"
             filename "users.txt"
             package
             path "/"
      }

      public file {
             name "inKeyWords"
             package
             filename "keywords.xml"
             path "/"
      }

    cmdline { |ctx|
            if (!ctx.crawl)
                  "{0} dump graph --output output.dat"
            else
                  "{0} crawl depth = " + ctx.depth.to_s + " --keywords
keywords.xml --seeds users.txt"
            end
      }
}

outputs {
      public file_group {
```

```
            name "out_files"
            filters ["\.dat$"]
      }
}

prepare_package
```

Как видно из листинга 3.1, входные параметры описаны со следующими дополнительными атрибутами:

- «значение по умолчанию» (default) – определяет начальное значение, в случае depth – это 3;
- «тип» (type) – необходим для отсеивания низкоуровневых ошибок связанных с типизацией данных. В Hadrawler параметрами с данным атрибутом являются depth, crawl, dump.
- «пакет» (package) – используется только для файлов и показывает, что данный файл является входным для пакета. В Hadrawler параметром с таким атрибутом является inUsers, inKeyWords.

Выходные параметры пакета описаны со следующими атрибутами: «ожидаемый» (expected) – используется только для файлов и показывает, что данный файл является выходным для пакета. В Hadrawler параметром с таким атрибутом является out_files.

Из описания так же видно, что данный пакет может выполнять две функции: краулинг данный и сохранение на локальный диск полученного в результате краулинга графа сети.

3.3.2. Прикладной пакет ISM моделирования распространения информации

Для интеграции пакета ISM в МИТП необходимо описать его взаимодействия с платформой, опираясь на формат представления входных и выходных данных, на языке EasyPackage. Платформенный скрипт описания пакета ISM позволяет определить уровень абстракции и интерпретации в работе с входными и выходными параметрами, что обеспечивает гибкость и упрощение процедуры взаимодействия пользователя с платформой на этапе запуска задания. Платформенный скрипт описания пакета ISM определяется в соответствии с листингом 3.2.

Листинг 3.2 – Скрипт описания пакета ISM

```
name "ISM"
#version nil
display_as "Information spreading modelling"
```

```
vendor "Itmo"
url "http://escience.ifmo.ru/"
license "GPLv3"
description "ISM help to understand and analise how i.e. rumors are being
spread."
#logo ""

inputs {
      public file {
             name "inDataFile"
             required
             filename "input.dat"
             path "/"
             package
      }

      public file {
             name "inConfigFile"
             required
             filename "config.dat"
             path "/"
             package
      }

    cmdline { |ctx| "java -jar {0} --input input.dat --config config.dat --
output output.dat"}
}

outputs {
      public file {
            name "outputFile"
            path "/"
            filename "output.dat"
            expected
      }
}

prepare_package
```

Как видно из листинга 3.2, входные параметры описаны со следующими дополнительными атрибутами:

«необходимый» (required) – данный атрибут отражает условие присутствие этого параметра при запуске, в описания пакета ISM это – inDataFile и inConfigFile;

«пакет» (package) – используется только для файлов и показывает, что данный файл является входным для пакета. В ISM параметрами с таким атрибутом являются inDataFile и inConfigFile.

Выходные параметры пакета описаны со следующими атрибутами:

«ожидаемый» (expected) – используется только для файлов и показывает, что данный файл является выходным для пакета. В ISM параметром с таким атрибутом является outputFile.

4. ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА

ПС функционирует в рамках распределенной среды облачных вычислений под управлением многофункциональной инструментально-технологической платформы (МИТП) CLAVIRE RU.СНАБ.80066-06. Для использования ПС необходима рабочая станция с подключением к Интернет, со следующими минимальными характеристиками:

- архитектура процессора – x86, x86_64, IA64;
- объем оперативной памяти – 1 ГБ;
- объем свободного пространства на жестком диске – 1 ГБ;
- тактовая частота процессора – 1 ГГц.

Для работы с композитным приложением необходимо использовать браузеры Mozilla FireFox (версия 3.0 и выше), Google Chrome (версия 13 и выше), Opera (версия 9.0 и выше) и Internet Explorer (версия 7.0 и выше)

Для работы пакета Hadrawler необходим кластер машин с установленной библиотекой Apache Hadoop. К машинам этого кластера предъявляются следующие требования:

- архитектура процессора – x86, x86_64, IA64;
- объем оперативной памяти – 1 ГБ;
- объем свободного пространства на жестком диске – 100 ГБ;
- тактовая частота процессора – 1 ГГц.
- Операционная система – Linux
- Доступ к сети Интернет

5. ВЫЗОВ И ЗАГРУЗКА

Исполнение ПС выполняется средствами МИТП CLAVIRE. При запуске композитное приложение описывает следующий процесс, реализуемый в среде облачных вычислений средствами МИТП:

1. По заданному перечню параметров запуска (в тексте программы на языке EasyFlow) определяются входные данные пакетов.
2. На основе мониторинга доступных вычислительных ресурсов МИТП CLAVIRE строит оптимальное (с точки зрения минимизации общего времени выполнения) расписание исполнения цепочки запусков.
3. Экземпляр пакета генерации комплексной сети Hadrawler запускается на исполнение на выбранном вычислительном ресурсе.

4. По окончании расчета экземпляра пакета Hadrawler выходные данные (в форме файлов) передаются на вход пакету моделирования распространения информации по сети ISM, несколько экземпляров которого параллельно запускаются на исполнение на подготовленных вычислительных ресурсах.
5. После завершения работы пакета ISM выходные файлы передаются пакету scilab, который на основании полученных данных проводит визуализацию интегральных характеристик посредством построения графиков.
6. После окончания работы приложения выходные данные передаются в хранилище данных, и пользователь уведомляется средствами МИТП об успешном выполнении задания.

На рисунке 5.1 показан CWF в процессе исполнения композитного приложения под управлением МИТП.

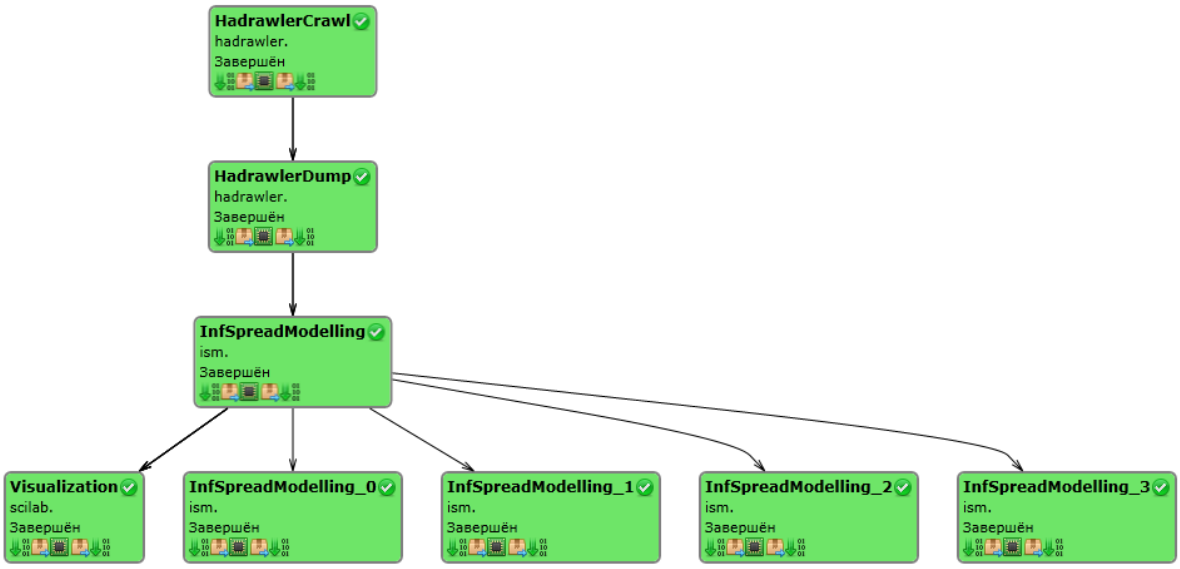

Рисунок 5.1 - Образ WF исполнения композитного приложения

6. ВХОДНЫЕ ДАННЫЕ

Входными данными для композитного приложения являются: словарь терминов для построения классификатора текстов в пакете Hadrawler, набор начальных узлов, с которых производится краулинг в пакете Hadrawler и config-файл для пакета ISM.

Словарь терминов для построения классификатора задается в виде xml файла специального формата, отдельные элементы которого приведены в листинге 6.1.

Листинг 6.1. Пример (отрывок) словаря терминов для пакета Hadrawler

```
<?xml version="1.0" encoding="UTF-8"?>
<words>
    <!-- Weights map -->
    <weight id="w_A" value="10"/>
    <weight id="w_B" value="6"/>
    <weight id="w_C" value="3"/>
    <weight id="w_C2" value="6"/>
    <weight id="w_B1" value="10"/>
    <weight id="w_B2" value="5"/>
    <weight id="w_A2" value="13"/>
<!-- Keywords map for group A -->
    <word_a name="абстяг*" weight="w_A" deff="общие_слова">
        <word_b name="ломк*" weight="w_A+w_B+w_B2" deff="ломка"/>
        <word_b name="отлом*" weight="w_A+w_B+w_B2" deff="ломка"/>
        <word_b name="бахнут*" weight="w_A+w_B+w_B2" deff="инъекция">
            <word_c name="вливат*" weight="w_A+w_B+w_B2+w_C2"
deff="инъекция"/>
            <word_c name="лекарств*" weight="w_A+w_B+w_B2+w_C2"
deff="наркотик"/>
        </word_b>
    </word_a>
<!-- Alias for group A -->
    <ampl_a word_1="анаш*" word_2=" башатумн*" weight="w_A+w_A2"/>
    <ampl_a word_1="анаш*" word_2=" бошк*" weight="w_A+w_A2"/>
<!-- Keywords map for group B -->
    <word_b name="агрегат*" weight="w_B" deff="шприц"/>
    <word_b name="астрал*" weight="w_B" deff="общие_слова"/>
    <word_b name="барыга*" weight="w_B" deff="общие_слова"/>
    <word_b name="бахнут*" weight="w_B" deff="инъекции"/>
    <!-- Alias for group B -->
    <ampl_b word_1="гарик*" word_2="[дурь]" weight="w_B+w_B1"/>
    <ampl_b word_1="гарик*" word_2="[дым]" weight="w_B+w_B1"/>
    <ampl_b word_1="гарик*" word_2="жарех*" weight="w_B+w_B1"/>
    <!-- Keywords map for group C -->
    <word_c name="боинг*" weight="w_C" deff="шприц"/>
    <word_c name="емкост*" weight="w_C" deff="шприц"/>
    <word_c name="каранд*" weight="w_C" deff="шприц"/>
    <word_c name="[конь]" weight="w_C" deff="шприц"/>
    <word_c name="[машина]" weight="w_C" deff="шприц"/>
</words>
```

Файл, содержащий список узлов, с которых краулер начинает обход сети, имеет вид текстового файла, на каждой строке которого записано имя пользователя. Пример приведен в листинге 6.2:

Листинг 6.2. Пример (отрывок) файла со списком начальных узлов для пакета Hadrawler

```
Likesky
umka379
barhano
```

Config-файл для пакета ISM имеет следующий формат: строка чисел, разделенных пробелами. Первое число: параметр λ - вероятность, с которой вирус передается

соседнему узлу. Второе число: параметр α - вероятность, с которой узел становится невосприимчивым к инфекции. Третье число: минимальная степень вершины, с которой начинается распространение вирусной инфекции.

7. ВЫХОДНЫЕ ДАННЫЕ

7.1. Прикладной пакет Hadrawler для обхода социальной сети

В качестве выходных данных Hadrawler предоставляет граф, который является результатом обхода сети Livejournal. Формат вывода:

- Узлы идентифицируются числами.
- Если между узлами есть ребро, то в выходной файл записывается информация вида: `номер_первой вершины--номер_второй_вершины`.

Чтобы этот формат файла был корректно разобран пакетом ISM, в начале файла пишется `cluster_1`, а в конце файла пишется `cluster_cross`

Пример:

```
cluster_0
1--2
2--3
cluster_cross
```

7.2. Прикладной пакет ISM

В качестве выходных данных ISM предоставляет описание начальных условий работы алгоритма, а так же описание каждого шага распространения информации по сети, включая детализацию по классам вершин.

Формат вывода:

- Первая строка, три натуральных числа, разделенных пробелами:
  - число классов вершин исходном графе n;
  - номер класса вершины с которой началось распространение информации;
  - степень этой вершины.
- Далее следуют строки, описывающие информацию о каждом шаге алгоритма. В каждой строке указываются следующие данные:
  - Первое число – номер шага алгоритма;
  - (n + 1) группа чисел, описывающая число зараженных (R), незараженных (I) и невосприимчивых (R) вершин для всех классов (первая группа) и для каждого класса в отдельности (еще n групп).

Пример (отрывок):

```
2 0 27

1 9998 0 1 3223 0 0 6775 0

13 9986 0 1 3223 0 12 6763 0

35 9957 7 21 3202 1 14 6755 6

23 9934 42 0 3202 22 23 6732 20

46 9897 56 37 3165 22 9 6732 34

41 9856 102 2 3163 59 39 6693 43

10 9846 143 10 3153 61 0 6693 82
```

7.3. Пакет визуализации scilab

На рис. 7.1 приведен пример выходного файла после работы пакета scilab в составе данного композитного приложения.

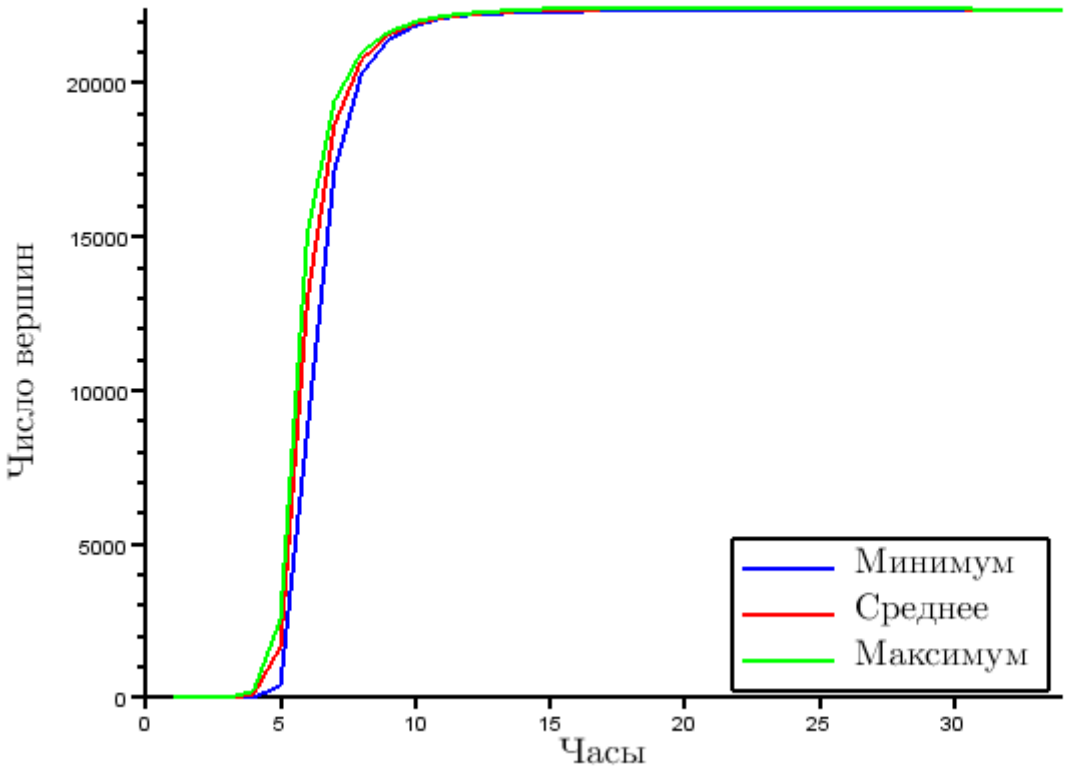

Рисунок 7.1 - Пример выходного файла пакета SciLab

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

БД	База данных
КСВ	Компонент событийного взаимодействия
МИТП	Многофункциональная инструментально-технологическая платформа

ПЕРЕЧЕНЬ ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Olston, C. Web Crawling. // *Foundations and Trends in Information Retrieval.* 2010

[2] *Craswell N., Hawking D., Robertson S.* Effective Site Finding using Link Anchor Information. // 2001.

[3] Newman M.E.J. The structure and function of complex networks // SIAM Review, Vol. 45, N 2, pp. 167–256, 2003.

[4] Wen-Jie Bai, Tao Zhou, Bing-Hong Wang. Interplay between HIV/AIDS Epidemics and Demographic Structures Based on Sexual Contact Networks // arXiv:physics/0602173v1, 2006.

[5] *Gupta V., Lehal G. S.* A Survey of Text Mining Techniques and Applications. // 2009.

[6] *Manning C. D., Raghaven P., Schuetze H.* Stemming and Lemmatization. // 2009.

[7] Daley D., Kendall D. Epidemics and rumours // Nature. 1964.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

Лист регистрации изменений									
Изм.	Номера листов (страниц)				Всего листов (страниц) в докум.	№ документа	Входящий № сопроводительного докум. и дата	Подп.	Дата
	изменённых	заменённых	новых	Аннулированных					