

black2

Глоссарий

Распределённая система обработки информации — система взаимодействующих независимых автоматизированных информационных систем, каждая из которых принадлежит и администрируется различными организациями (лицами), которые преследуют свои собственные цели.

Участник РСОИ — независимая автоматизированная информационная система, входящая в состав распределенной системы обработки информации.

Сообщение — минимально передаваемая единица полезной информации.

Заявка — единица обслуживания.

Протокол обмена — это пятёрка:

- а) назначение протокола (предоставляемые им возможности);
- б) используемый нижестоящий протокол или протоколы;
- в) алфавит;
- г) словарь сообщений и синтаксис сообщений;
- д) возможная последовательность сообщений и их семантика.

Жизненный цикл заявки — конечное множество возможных состояний заявки.

Релевантность — способность информации соответствовать потребностям.

Перечень используемых сокращений и аббревиатур

РСОИ — Распределённая система обработки информации

АИС — Автоматизированная информационная система

БД — База данных

ЛПО — Логика предметной области

ИП — Интерфейс пользователя

ПОЗ — Подсистема обработки заявок

ПФС — Подсистема фильтрации сообщений

ПОС — Подсистема обмена сообщениями

ПС — Подсистема статистики

ПФЗ — Подсистема фильтрации заявок

ПО — Программное обеспечение

1 Введение

Данное техническое задание составлено для проектирования ПО «Подсистема фильтрации нежелательных заявок участника РСОИ». Техническое задание выполнено на основе ГОСТ 19.201-78. «ЕСПД. Техническое задание. Требования к содержанию и оформлению».

1.1 Краткое описание предметной области

Типичная структура РСОИ представлена на рис. 1.1. АИС, входящие в состав РСОИ являются независимыми и не могут ни получить полностью достоверную информацию о состоянии других систем на заданный момент времени, ни полностью контролировать её поведение. Каждая из АИС имеет свою логику работы (с точки зрения других участников эта логика полностью не просматривается) и свою БД или несколько БД, в которых хранит состояние своей модели предметной области. Прямой доступ на уровне SQL к этим БД другим участникам не даётся, даже на чтения с ограниченными правами. В силу этого системы могут выдавать упрощённую или искажённую информацию другим участникам РСОИ. В общем случае для создания РСОИ всем организациям, владеющим непосредственно взаимодействующими участниками РСОИ необходимо заключить друг с другом договор.

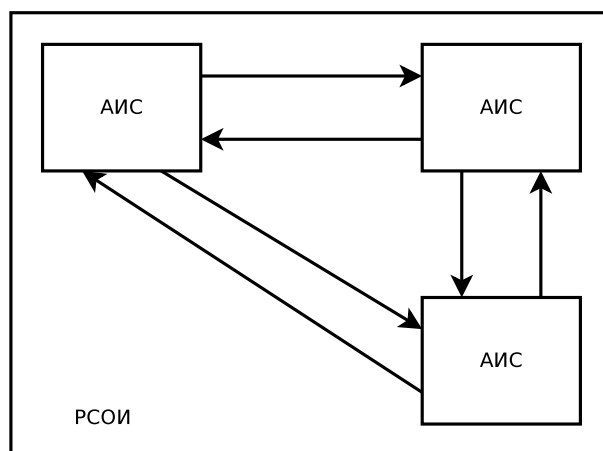


Рисунок 1.1 — Структура РСОИ, состоящей из 3 независимых АИС

Структура участника РСОИ в общем случае показана на рис. 1.2. ПОС занимается вопросами преобразования полученных от ПОЗ сообщений в форму, понятную другим системам, и принимаемых от других систем сообщений в форму, понятную непосредственно самой системе. ПФС фильтрует сообщения во внутреннем представлении, чтобы не загружать систему работой, которая выглядит неправдоподобной. Входящая в её состав ПФЗ выносит «вердикт» по заявкам, полученным от других систем, исходя из которого протокол обмена предпринимает те или иные дальнейшие шаги. Решение принимается на основе:

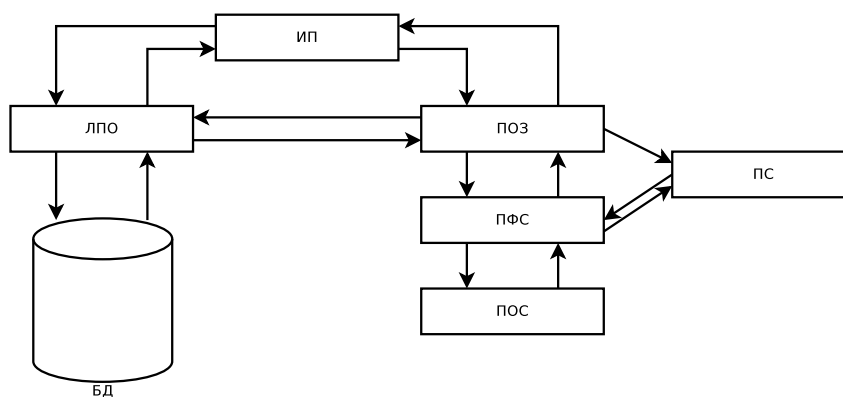


Рисунок 1.2 — Структура участника РСОИ

- а) текущего состояния модели предметной области;
- б) истории взаимоотношений с партнёрами;
- в) параметров заявки (например, количестве запрашиваемых ресурсов);
- г) правил принятия решений.

ПОЗ является ключевой с точки зрения участия в распределенной системе. Она реализует часть высокоуровневого протокола участника РСОИ, связанную с жизненным циклом выполнения заявки.

1.2 Существующие аналоги

В настоящее время нет аналогичных ПП, позволяющих выполнить анализ релевантности и соответственно фильтрацию заявок на основе недельно-сезонных колебаний без привязки к конкретной предметной области.

1.3 Описание подсистемы фильтрации заявок

Главное назначение ПО «Подсистема фильтрации нежелательных заявок участника РСОИ» - определение релевантности поступающих в систему заявок. Алгоритм работы подсистемы показан на рис. 1.3. Для определения релевантности заявки на основе статистики отношений с клиентом подсистеме необходимо пройти предварительное обучение с использованием данных, предоставляемых ей ПС. Правила принятия решений задают дополнительные ограничения, накладываемые на поступающие заявки, исходя из здравого смысла и бизнес-логики работы системы. Как видно из рисунка правила бывают двух типов. Выполнение правил 1-го типа проверяется до проверки релевантности системы на основе статистики отношений с клиентом (например, для данного клиента пропускать все поступающие заявки без анализа статистики отношений с ним и т.п.). А выполнение правил 2-го типа проводится после него в случае, если заявка была признана релевантной, исходя из статистики отношений с клиентом (например, заявка на количество ресурса, большее половины имеющихся на данный момент в системе, признается нерелевантной).

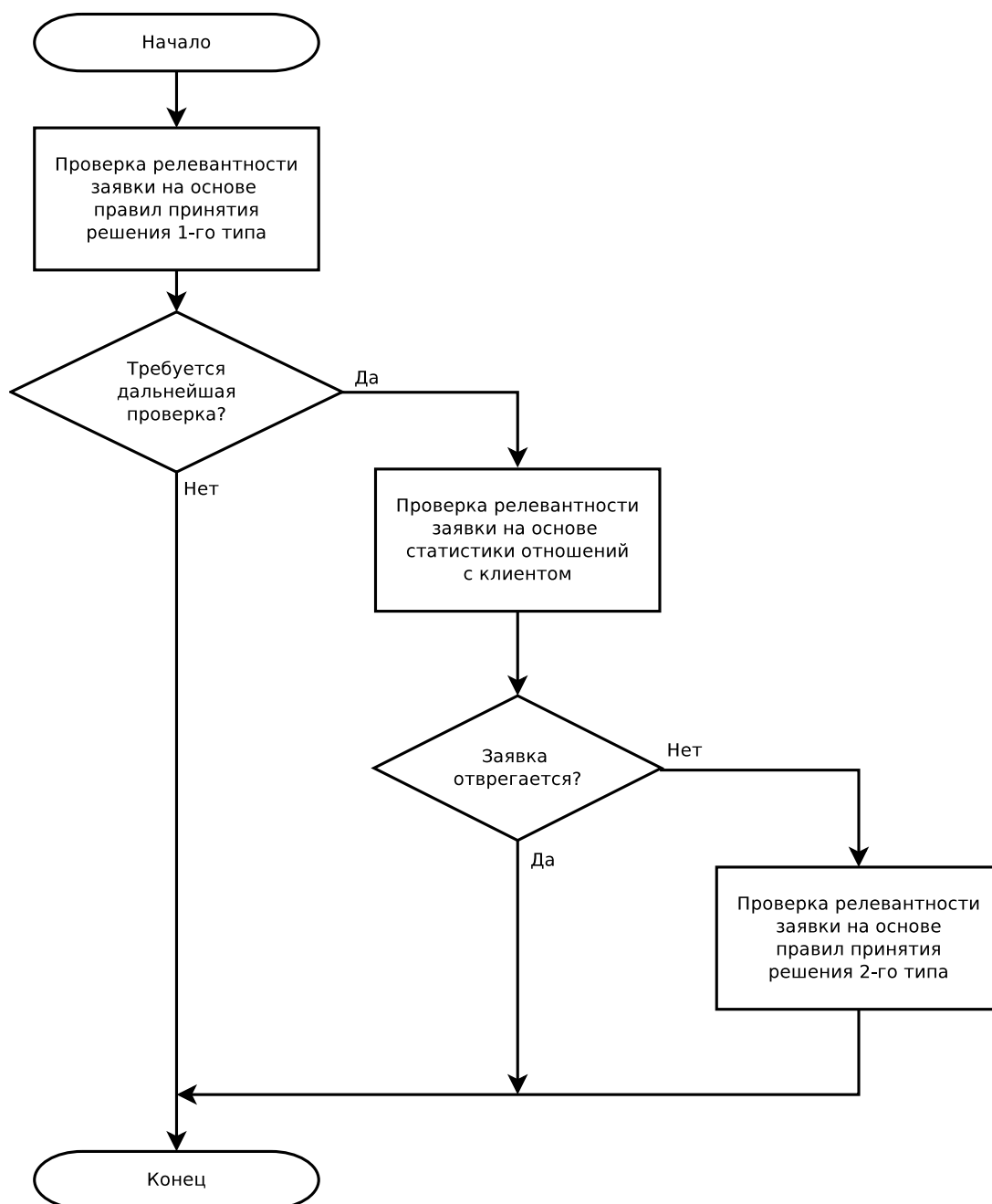


Рисунок 1.3 — Алгоритм определения релевантности заявки

2 Основания для разработки

Разработка ведётся в рамках выполнения лабораторных работ по курсу «Методология программной инженерии» на основе утверждённого учебного плана и в рамках курсового проектирования по курсу «Распределённые системы обработки информации».

3 Назначение разработки

ПО «Подсистема фильтрации нежелательных заявок участника РСОИ» предназначено для определения релевантности заявок, поступающих в систему от других участников РСОИ, чтобы не загружать систему бесполезной работой.

4 Требования к программному изделию

4.1 Требования к подсистеме

а) Разрабатываемое ПО должно обеспечить функционирование подсистемы в режиме 24/7/365.

б) Время восстановления после сбоя должно составлять не более 1 часа.

4.2 Требования к функциональным характеристикам

а) Время переобучения подсистемы должна составлять не более 3-х часов.

б) Подсистема должна обеспечить обработку не менее 100 заявок в минуту.

в) Количество неправильно классифицированных заявок за сутки должно составлять не более 5% от общей массы заявок, поступающих в систему за этот период.

4.3 Требования к реализации

а) Клиенты должны обладать уникальными идентификаторами.

б) Каждый ресурс системы должен иметь уникальный идентификатор и количественную меру его измерения.

в) Обращение к подсистеме фильтрации заявок для определения релевантности заявок должно вестись с использованием протокола XML-RPC.

г) Для проведения переобучения вызывающая сторона (ПФС) должна обеспечить реализацию абстрактного метода доступа к БД статистики отношений с клиентом со следующей сигнатурой: `get_statistic_records(date, client_id)`. Входные и выходные параметры метода описаны в таблице ??.

д) Правила принятия решения должны описываться на языке SWI-Prolog. Правила 1 и 2 типа должны быть описаны в разных файлах, пути к которым должны задаваться в конфигурационном файле подсистемы. Применение новых правил должно быть возможно только после перезапуска подсистемы.

е) Подсистеме необходимо указывать диапазон дат, статистику за которые необходимо использовать для переобучения, в конфигурационном файле.

4.4 Функциональные требования к подсистеме

а) Подсистема должна обеспечить определение релевантности заявок на основе статистики отношений с клиентом и правил принятия решения (1 и 2 типа).

б) Подсистема должна предоставить возможность переобучения на основе статистики отношений с клиентом.

в) Подсистема должна предоставить возможность изменения правил принятия решения (1 и 2 типа).

г) На время переобучения подсистема должна останавливать свою работу.

Таблица 4.1 — Параметры метода get_statistic_records

Название параметра	Назначение	Тип	Допустимые значения
client_id	Идентификатор клиента	integer	(0; 2 ³²)
date	Дата, статистика за которую интересует систему	timestamp	(0; 2 ³²)
output	Данные возвращаемые методом	Массив кортежей вида (ResourceID, ResourceCount, ResourceLostCount, OrderState)	
ResourceID	Идентификатор ресурса	integer	(0; 2 ³²)
ResourceCount	Запрашиваемое количества ресурса	integer	(0; 2 ³²)
ResourceLostCount	Оставшееся количество ресурса	integer	(0; 2 ³²)
OrderState	Заключительное состояние заказа	integer	1 - заказ завершён, 0 - заказ отменён

4.4.1 Входные параметры подсистемы

а) Для определения релевантности вход подсистемы должны подаваться данные, представленные в таблице 4.2.

б) Для описания общих правил принятия решений могут использоваться переменные, описанные в таблице 4.3.

4.4.2 Выходные параметры подсистемы

После определения релевантности поступившей в систему заявки ПФЗ должна выдать решение о целесообразности обработки заявки статистики отношений с клиентом и правил принятия решений. Значения выходного параметра имеют логический тип и могут принимать следующие значения:

- а) True - заявку целесообразно обработать;
- б) False - иначе.

Таблица 4.2 — Входные параметры для определения релевантности заявки

Название параметра	Тип	Допустимые значения
Идентификаторы запрашиваемых ресурсов	Массив, содержащий элементы типа integer	(0; 2^{32})
Запрашиваемые количества ресурсов	Массив, содержащий элементы типа integer	(0; 2^{32})
Оставшиеся в системе количества запрашиваемых	Массив, содержащий элементы типа integer	(0; 2^{32})
Идентификатор клиента	integer	(0; 2^{32})

Таблица 4.3 — Переменные для описания правил принятия решений

Название переменной	Назначение	Тип	Допустимые значения
ParticipantID	Идентификатор клиента	integer	(0; 2^{32})
ResourceIDs	Идентификаторы ресурсов	Список, содержащий элементы типа integer	(0; 2^{32})
ResourceCounts	Количество запрашиваемых ресурсов	Список, содержащий элементы типа integer	(0; 2^{32})
ResourceLostCounts	Оставшееся количество ресурсов	Список, содержащий элементы типа integer	(0; 2^{32})
ResourceCount	Размер списка запрашиваемых ресурсов	Список, содержащий элементы типа integer	(0; 2^{32})
Resource	Идентификатор ресурса	integer	(0; 2^{32})
Count	Количество ресурса	integer	(0; 2^{32})

4.5 Требования к надёжности

Для повышения надёжности работы подсистемы необходимо:

- а) производить переобучение подсистемы не реже 1 раза в неделю;

б) производится журналирование поступающих заявок и принимаемых подсистемой решений;

в) срок хранения каждого принятого подсистемой решения должен составлять не менее 3 недель.

4.6 Требования к составу и параметрам технических средств

Минимальные технические требования:

- а) 2-х ядерный процессор с тактовой частотой 2ГГц;
- б) ОЗУ 4 ГБ;
- в) сетевая карта Ethernet стандарта 1000BASE-T.

Требования к операционному окружению:

- а) Операционные системы: Ubuntu 12.04.

4.7 Требования к информационной и программной совместимости

Разработка должна вестись с использованием открытого платформенно-независимого ПО.

4.8 Требования к документации

Документация должна включать:

- а) руководство по развертыванию подсистемы;
- б) руководство по настройке подсистемы.

Список стандартов и литературы

1. ОСТ 19.201-78. ЕСПД. Техническое задание. Требования к содержанию и оформлению.

2. Ишневская Т.И., Романова Т.Н. Технология программирования: Метод. указания к лабораторному практикуму. - Ч. 2. – М: Изд-во МГТУ им. Н.Э. Баумана, 2009