

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**СОГЛАСОВАНО**

Генеральный директор  
ЗАО «АйТи»



Бакиев О.Р.  
2011 г.

**УТВЕРЖДАЮ**

Ректор НИУ ИТМО



Васильев В.Н.  
2011 г.

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

**ПРОГРАММНЫЙ КОМПОНЕНТ УЧЕТА ИСПОЛЬЗОВАНИЯ  
РЕСУРСОВ CLAVIRE/BILLING**

**ОПИСАНИЕ ПРОГРАММЫ**

**ЛИСТ УТВЕРЖДЕНИЯ**

RU.СНАБ.80066-06 13 34-ЛУ

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

Представители  
Организации-разработчика

Руководитель разработки,  
профессор НИУ ИТМО

Бухановский А.В.

“20” декабря 2011 г.

Ответственный исполнитель,  
с.н.с. НИУ ИТМО

Луценко А.Е.

“20” декабря 2011 г.

Нормоконтролер  
ведущий инженер НИУ ИТМО

Позднякова Л.Г.

“20” декабря 2011 г.

2011

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

---

**УТВЕРЖДЕН**  
**RU.СНАБ.80066-06 13 34-ЛУ**

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

**ПРОГРАММНЫЙ КОМПОНЕНТ УЧЕТА ИСПОЛЬЗОВАНИЯ  
РЕСУРСОВ CLAVIRE/BILLING**

**ОПИСАНИЕ ПРОГРАММЫ**

**RU.СНАБ.80066-06 13 ОШИБКА! ИСТОЧНИК ССЫЛКИ НЕ НАЙДЕН.4**

<b>Ине.№ подл.</b>	
<b>Подп. и дата</b>	
<b>Взам. ине. №</b>	
<b>Ине.№ дубл.</b>	
<b>Подп. и дата</b>	

**ЛИСТОВ 14**

2011

## **АННОТАЦИЯ**

Компонент учета использованных ресурсов CLAVIRE/Billing RU.СНАБ.80066-06 01 34 предназначен для агрегирования данных об использовании ресурсов пользователями комплекса, задания тарифов и квот на ресурсы и выставления счетов (биллинга) по итогам вычислений.

Программный компонент учета использования ресурсов 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 .....	4
2.	ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ .....	4
3.	ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ .....	5
3.1.	Внутренняя структура компонента.....	5
3.2.	Взаимодействие с компонентами комплекса.....	6
3.3.	Классы компонента .....	8
4.	ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА .....	10
5.	ВЫЗОВ И ЗАГРУЗКА.....	10
6.	ВХОДНЫЕ ДАННЫЕ .....	11
7.	ВЫХОДНЫЕ ДАННЫЕ .....	11
	ПЕРЕЧЕНЬ СОКРАЩЕНИЙ .....	13

## 1. ОБЩИЕ СВЕДЕНИЯ

Компонент учета использованных ресурсов CLAVIRE/Billing RU.СНАБ.80066-06 01 34 предназначен для агрегирования данных об использовании ресурсов пользователями комплекса, задания тарифов и квот на ресурсы и выставления счетов (биллинга) по итогам работы.

Компонент состоит одного модуля – WCF-сервиса, реализованного на языке программирования C# версии 4.0. Для его работы необходимо следующее системное программное обеспечение: ОС семейства Windows NT (версии старше Windows 2000), .NET 4.0, Microsoft Internet Information Services (с версией старше 6.0).

Компонент может функционировать на аппаратных системах с архитектурой процессора x86, x86\_64 и IA64. Для хранения данных компонент требует наличия в платформе установленной документо-ориентированной системы управления базами данных MongoDB.

## 2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

Компонент учета использованных ресурсов предназначен для решения следующих задач.

- 1) Управление квотами пользователей на использование ресурсов платформы.
- 2) Управление тарифами на ресурсы, применение различных схем тарификации.
- 3) Вычисление стоимости использованных ресурсов на основе текущих тарифов.
- 4) Пересчет квот при успешном завершении вычисленных задач пользователей.
- 5) Ведение аудита операций со счетами пользователя и тарифами

Компонент учета использованных ресурсов обладает функциональностью для расчета затрат при использовании ресурсов, причем как для предварительных расчетов так и для окончательных с функцией изменения квот пользователей. Все операции, изменяющие квоты пользователя и тарифы, заносятся в журнал истории операций, который может быть представлен пользователю методами компонента.

### 3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

#### 3.1. Внутренняя структура компонента

Программный компонент реализован как один сервис Windows Communication Foundation, но для облегчения администрирования логически разделен на три интерфейса (контракта). Помимо удобства при разработке и в использовании такой подход делает возможным вариант установки сервиса, когда каждый контракт выполняется на разных экземплярах сервиса.

Первый контракт предоставляет интерфейс пользовательского профиля и такие функции, как уточнение текущего баланса, добавление и снятие денег со счета, проверка истории операций по счету пользователя. Второй контракт тарификации предоставляет методы для задания тарифов на использование ресурсов и их отмену, а также просмотр архива тарифов. Третий – интерфейс биллинга – для вычисления стоимости использованных ресурсов и изменения квот, оценки стоимости задания перед выполнением.

Компонент хранит информацию о пользователях, тарифах и операциях в системе управления базами данных MongoDB. Таким образом, происходит отделение логики программного компонента от данных, которыми он оперирует, формируется классическая двухуровневая структура web-сервиса.

База данных содержит три таблицы данных: пользователей, тарифов и выполненных операций. Таблица пользователей интегрирована с таблицей учетных данных компонента контроля доступа CLAVIRE/GateKeeper для удобства синхронизации данных. Порядок работы интерфейсов компонента с базой данных представлен на рис. 3.1.

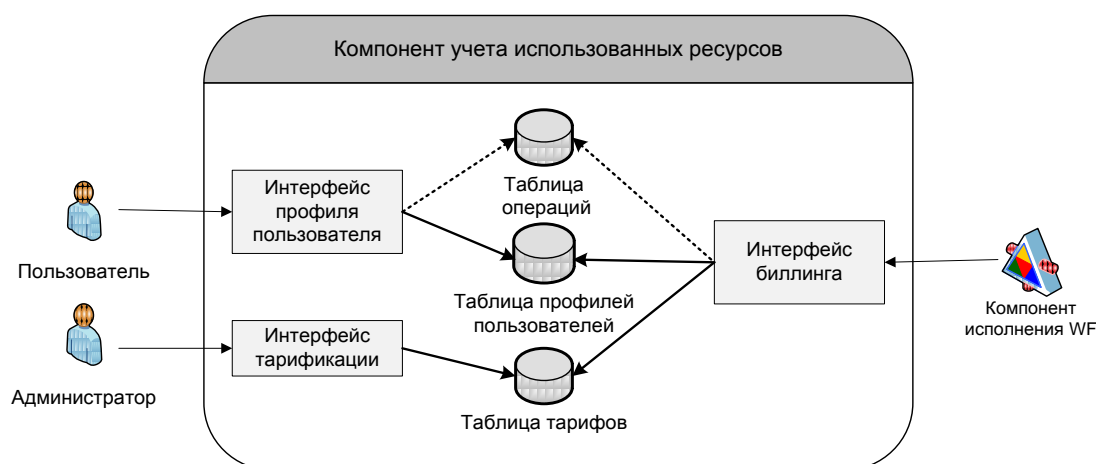


Рисунок 3.1 – Архитектура компонента учета использованных ресурсов

Интерфейс профиля пользователя работает непосредственно с таблицей профилей, при этом сохраняет подробности операции в таблице истории операций. Интерфейс тарификации работает только со своей таблицей тарифов, но при этом не удаляет из нее никаких записей – только помечает тарифы как неактивные – это необходимо для сохранения возможности пересчета завершенных операций.

Интерфейс биллинга по своей специфике работает со всеми таблицами: берет данные из базы тарифов, изменяет таблицу профилей пользователей и добавляет записи в таблицу истории операций. Примерами вызовов функций интерфейса биллинга могут служить завершения оплачиваемых WF, а также вызовы некоторых компонентов для оценки стоимости предполагаемого выполнения WF. Все действия, приводящие к изменению квот пользователей, выполняются опосредованно, через интерфейс профиля пользователя и записываются в таблицу операций.

Кроме трех своих контрактов компонент учета использования ресурсов реализует интерфейс INotifiable взаимодействия с компонентом событийного взаимодействия. Это необходимо для получения сообщений о завершении выполнения WF и его шагов, для дальнейшего проведения расчетов за потребленный ресурс.

### **3.2. Взаимодействие с компонентами комплекса**

Как видно из рис. 3.1, каждый интерфейс используется различными компонентами комплекса по-своему.

Пользователи работают с интерфейсом профиля пользователя – в основном в информационном аспекте: уточнение своего текущего баланса, получение истории операций со счетом. Эти взаимодействия происходят через компоненты CLAVIRE/Ginger и CLAVIRE/AdminTool.

Компонент учета использованных ресурсов предоставляет пользователю инструменты работы с его личным профилем, такие как просмотр текущих квот на ресурсы, просмотр истории операций с квотами пользователя, пополнение квот. На данный момент используется агрегационная модель пользовательских квот, это означает, что квота пользователя едина для всех ресурсов и измеряется в некоторых условных единицах. Тарифы на ресурсы и все транзакции также задаются в этой условной валюте, при проведении каких-либо вычислений стоимость всех использованных ресурсов суммируется и вычитается из квоты пользователя.

RU.СНАБ.80066-06 13 34 **Ошибка! Источник ссылки не найден.**

Администратор пользуется интерфейсом тарификации для изменения тарифов на ресурсы с помощью компонента администрирования. Минимальной единицей измерения потребления ресурсов является одна миллисекунда одного ядра процессора, при этом учитывается все время, проведенное задачей в системе от запуска до завершения. Для большей гибкости тарифы могут быть заданы как на ресурс, так и на конкретный исполняемый прикладной пакет. Тариф также может зависеть от времени запуска задачи на вычисление, например, для МИТП-К может быть важно снижение нагрузки на инфраструктуру в течение дня, этого можно достичь снижением стоимости ресурсов в ночное время.

Самый масштабный – интерфейс биллинга, который используется исключительно внутренними компонентами платформы и зачастую, опосредованно, – с помощью компонента событийного взаимодействия. Ему передается описание выполненного (или еще только предполагаемого) workflow, на выходе он предоставляет стоимость (или оценку стоимости) данного потока.

На рис. 3.2 показано взаимодействие компонента Billing при постановке пользователем задачи на расчет.

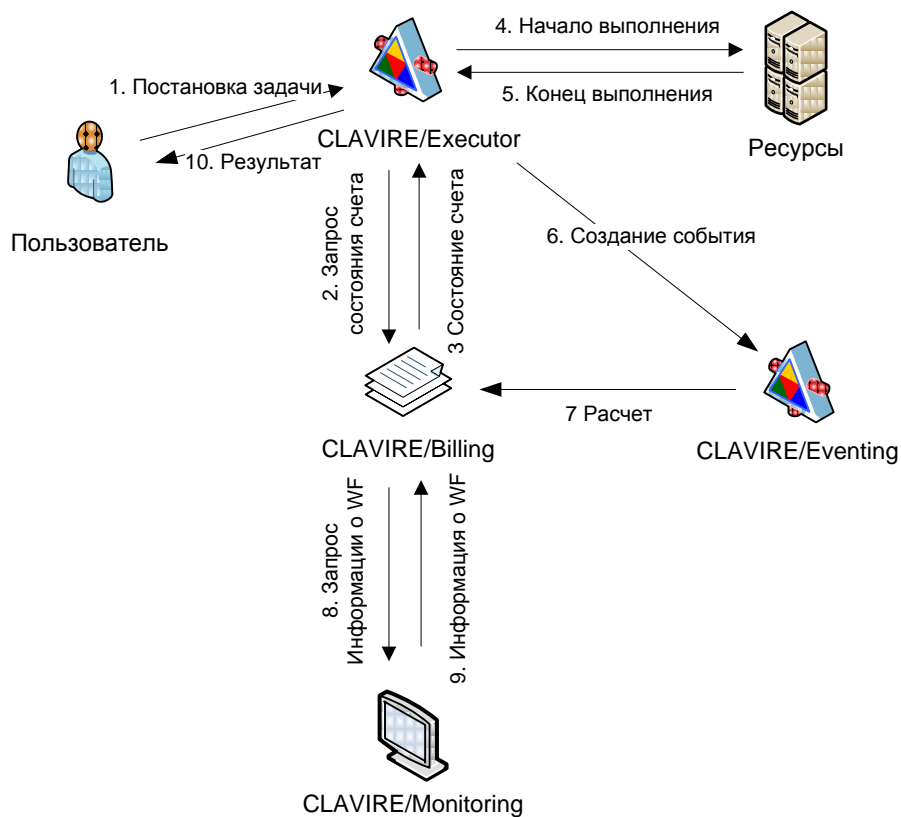


Рисунок 3.2 – Взаимодействие компонента с платформой



RU.СНАБ.80066-06 13 34 **Ошибка! Источник ссылки не найден.**

Шаг 1. Пользователь отправляет описание WF на выполнение через CLAVIRE/Ginger.

Шаг 2–3. Компонент исполнения WF уточняет у компонента Billing состояние счета пользователя. Если у пользователя недостаточно средств на счету – задача останавливается.

Шаг 4–5. Выполнение расчета пользователя.

Шаг 6. Компонент исполнения WF создает событие окончания расчета и направляет его компоненту событийного взаимодействия вместе с данными о завершённом потоке заданий.

Шаг 7. Компонент событийного взаимодействия сообщает компоненту учета использования ресурсов о событии и передает идентификатор WF.

Шаг 8–9. Пользуясь идентификатором, компонент получает всю информацию о WF от компонента мониторинга и компонента исполнения WF.

Шаг 10. Пользователь получает результат вычислений, программный компонент Billing вычисляет стоимость расчета по полученным данным.

### **3.3. Классы компонента**

На рис. 3.3 представлена диаграмма классов компонента.

- 1) BillingService – это главный класс, реализующий функциональность компонента учета использованных ресурсов. Как видно из диаграммы, он наследуется от класса MongoBaseForBilling и реализует интерфейсы IAccountingService, IBillingService, ITariffingService.
- 2) MongoBaseForBilling – специальный класс для работы с базой данных MongoDB, разработанный для нужд компонента учета использования ресурсов.
- 3) IAccountingService – интерфейс пользовательских профилей, содержит методы для работы со счетом пользователя и историей операций.
- 4) ITariffingService – интерфейс тарификации.
- 5) IBillingService – интерфейс расчета использования ресурсов (биллинга).

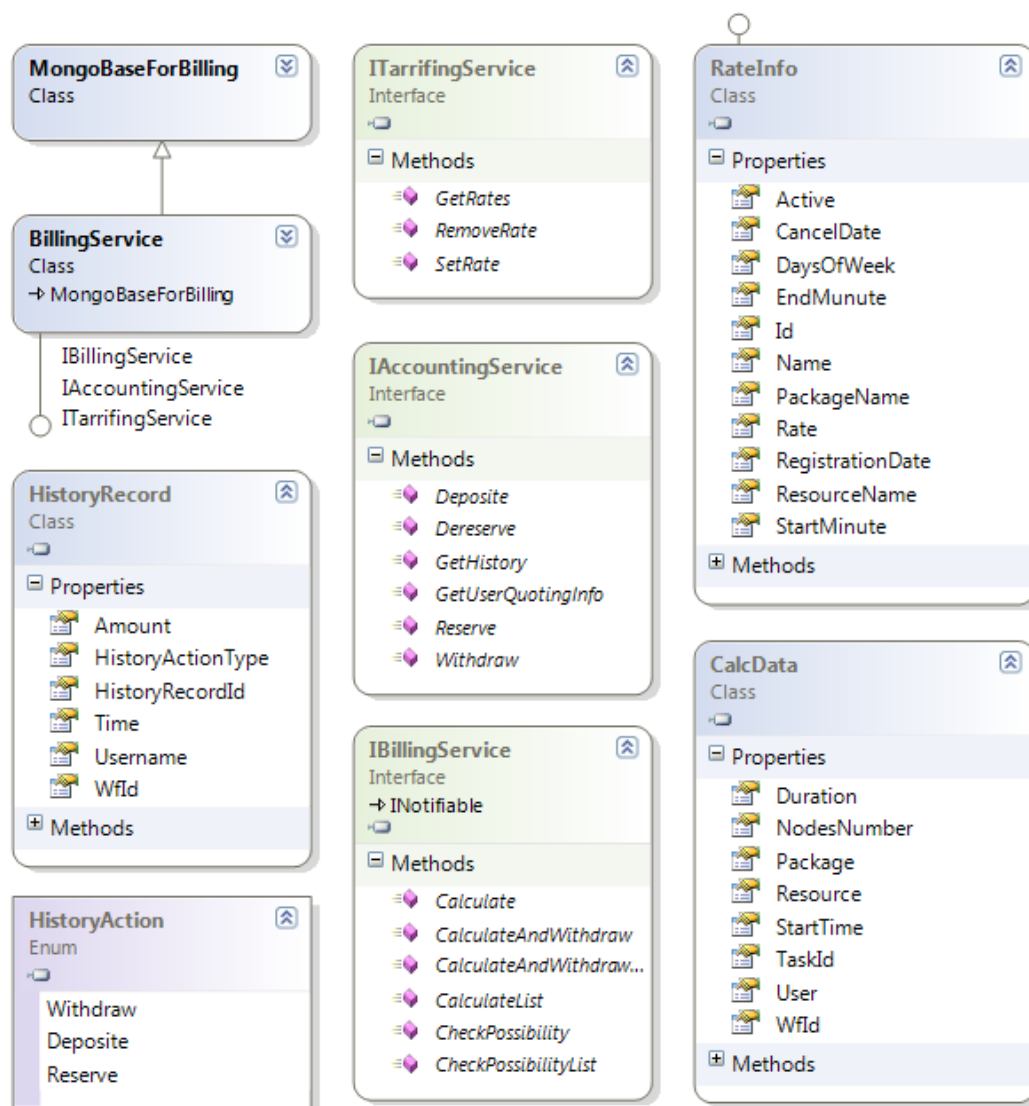


Рисунок 3.3 – Диаграмма классов компонента

Следующие классы служат цели представления данных.

Перечисление HistoryAction определяет виды операций, производящихся над счетами пользователей: Deposit – добавление денег на счет, Withdraw – снятие денег со счета, Reserve – операция для резервирования денег перед выполнением вычислений.

HistoryRecord – содержит информацию о проведенных операциях со счетом пользователей. Состоит из полей типа операции, суммы на которую изменился счет, времени выполнения, операции имя пользователя и идентификатора WF.

RateInfo – класс для представления данных тарифа, включает такие поля, как идентификатор тарифа, имя и значение тарифа, имя ресурса и пакета, дни действия тарифа в течение недели и время действия в течение дня, а также даты введения и отмены тарифа.

RU.СНАБ.80066-06 13 34 **Ошибка! Источник ссылки не найден.**

CalcData – информация, необходимая для вычисления стоимости использованных ресурсов одного шага WF: длительность расчетов и количество использованных узлов, вычислительный ресурс и прикладной пакет, идентификаторы пакета и WF, задавший задачу пользователь и время старта вычислений.

#### **4. ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА**

Программный компонент предназначен для функционирования на аппаратных системах с характеристиками:

- архитектура процессора – x86, x86\_64, IA64;
- минимальный объем оперативной памяти – 1 ГБ;
- минимальный объем свободного пространства на жестком диске – 1 ГБ;
- минимальная тактовая частота процессора – 1 ГГц.

Компонент требует для своей работы наличия следующего системного ПО: ОС семейства Windows NT (версии старше Windows 2000), .NET 4.0, IIS (с версией старше 6.0).

#### **5. ВЫЗОВ И ЗАГРУЗКА**

Компонент реализован в виде SOAP WCF-сервиса платформы .NET. Для запуска сервиса используется стандартный механизм используемого web-сервера. Загрузка сервиса в этом случае выполняется web-сервером автоматически по мере поступления запросов от клиентов сервиса.

После запуска компонента доступ к нему можно получить по адресу, заданному при развертывании, например, <http://SampleServer.ru:8008/SampleService>. Компонент не имеет жестких зависимостей от других частей комплекса, при установке важно только наличие уже установленного компонента обеспечения доступа к инфраструктуре.

Вызов сервиса производится стандартным для технологии WCF способом: по опубликованному описанию сервиса (WSDL) необходимо создать прокси-класс, через который осуществляется взаимодействие с сервисом путем вызова необходимых операций (методов). Процедура создания прокси-класса зависит от того, на базе каких технологий строится клиентское приложение. Если выбран язык программирования C# и платформа .NET, построение клиента производится за счет вызова служебного программного

RU.СНАБ.80066-06 13 34 **Ошибка! Источник ссылки не найден.**

средства svcutil.exe, распространяемого в составе платформы .NET, либо за счет создания ссылки на сервис в среде Microsoft VisualStudio.

## 6. ВХОДНЫЕ ДАННЫЕ

Компонент основан на протоколе SOAP и реализован web-сервис с применением технологии Windows Communication Foundation. Разработка такого сервиса состоит из двух частей: написание ядра сервиса (реализация контрактов обеспечивающее выполнение необходимой логики) и расширение поведения сервиса для предоставления дополнительной функциональности. Входными данными являются параметры различных типов, передаваемые вызывающей стороной и инкапсулированные в сообщения SOAP. Так как формат сообщений SOAP опубликован и доступен для изучения, его описание не приводится.

## 7. ВЫХОДНЫЕ ДАННЫЕ

Выходными данными являются SOAP-сообщения, генерируемые при ответе на запросы, также ими можно считать данные, сохраняемые в таблицах базы данных. MongoDB использует стандарт JSON для хранения данных, все специализированные форматы являются его расширениями. На рис. 7.1 приведен пример данных, используемых для задания тарификации.

```
{
  "_id": "f2c21cc2-e3a4-4b0b-af4e-7671fc319ea8",
  "Name": "Sample",
  "ResourceName": null,
  "PackageName": null,
  "DaysOfWeek": [0, 1, 2, 3, 4, 5, 6],
  "StartMinute": 0,
  "EndMinute": 600,
  "Rate": "0.00001",
  "Active": true,
  "RegistrationDate": "Fri, 23 Dec 2011 18:09:36 GMT +03:00",
  "CancelDate": "Mon, 01 Jan 0001 03:00:00 GMT +03:00"
}
```

Рисунок 7.1 – Пример записи данных о тарифе

В табл. 7.1 приведены значения полей формата тарификации.

Таблица 7.1

## Поля формата данных тарифов

Название поля	Описание поля
_id	Внутренний идентификатор объекта в MongoDB
Name	Удобное для чтения имя тарифа
ResourceName	Название тарифицируемого ресурса
PackageName	Название тарифицируемого прикладного пакета
DaysOfWeek	Дни действия сервиса
StartMinute	Время начала действия тарифа
EndMinute	Время окончания действия тарифа
Rate	Стоимость 1 мс работы на одном ядре ресурса
Active	Актуальность тарифа в текущий момент времени
RegistrationDate	Дата и время задания сервиса
CancelDate	Дата и время отмены сервиса (если он отменен)

На рис. 7.2 и в табл. 7.2 представлена информация о хранении данных об операциях.

```

{
  "_id": {
    "$oid": "4df9e8d565f5110bb090ff9c"
  },
  "HistoryActionType": withdraw,
  "Username": "exon",
  "Time": "Thu, 16 Jun 2011 15:28:21 GMT +04:00",
  "wfId": "2587cd1f-6dc7-4bae-9c5a-de8f316f0a8b",
  "Amount": 0.016781249999999998
}

```

Рис. 7.2. Пример записи данных об операциях

Таблица 7.2

## Поля формата данных об операциях

Название поля	Описание поля
_id	Внутренний идентификатор объекта в MongoDB
HistoryActionType	Тип операции с квотой: Withdraw – снятие наличных, Deposit – взнос
Username	Имя пользователя
Time	Время проведения операции
WfId	Идентификатор завершившего WF (при типе операции Withdraw)
Amount	Сумма изменения квоты

## **ПЕРЕЧЕНЬ СОКРАЩЕНИЙ**

SOAP	Simple Object Access Protocol (простой протокол доступа к объектам)
БД	База данных
МИТП	Многопрофильная инструментально-технологическая платформа
ОС	Операционная система
ПО	Программное обеспечение

