

UDC 004.65

**THE APPROACH TO THE DEVELOPMENT OF A COMMON
INFORMATION SPACE IN THE CLOUD-BASED REPOSITORY
FOR THE IMPLEMENTATION OF THE SYSTEM OF FINANCIAL
INTERMEDIATION IN THE SYSTEM OF GAS AND ELECTRICITY
DISTRIBUTION**

**ПОДХОД К РАЗРАБОТКЕ ЕДИНОГО ИНФОРМАЦИОННОГО
ПРОСТРАНСТВА НА ОСНОВЕ ОБЛАЧНОГО РЕПОЗИТАРИЯ
ДЛЯ РЕАЛИЗАЦИИ СИСТЕМЫ ФИНАНСОВОГО
ПОСРЕДНИЧЕСТВА В СИСТЕМЕ ГАЗО-ЭНЕРГОСБЫТА**

V.A. Kotelnikov, T.I. Grigorchuk, N.I. Yusupova

**FSBEI HPE “Ufa State Aviation Technical University”,
Ufa, the Russian Federation**

**FSBEI HPE “Ufa State Petroleum Technological University”,
Ufa, the Russian Federation**

Котельников В.А., Григорчук Т.И., Юсупова Н.И.

**ФГБОУ ВПО «Уфимский государственный авиационный
технический университет», г. Уфа, Российская Федерация**

**ФГБОУ ВПО «Уфимский государственный нефтяной технический
университет», г. Уфа, Российская Федерация**

e-mail: tgrigor@yandex.ru

Abstract. The article is devoted to the development and implementation of a common security space cloud for instant payments in favor of providers of services such as gas and electricity supply companies, utilities services, mobile services, etc. The analysis of the subject area of instant payments was carried out. The existing solutions for payment systems and the possibility of the use of net-

work services for the organization of payments reception were considered. We have considered the approach to describing network services based on the representation of the change service state of the world. To describe the formats and methods of data transfer between system elements instant payments are encouraged to use the language of WSDL, using the SOAP protocol and interface UDDI, which provides interaction of software systems, regardless of platform, ease of development and debugging of Web services. Schematic diagrams, models, algorithms for implementing a distributed payment system are presented. In solving this problem solved the problem of describing the common formats and methods of data transfer between system elements. A new scheme of interaction between service providers with regional and federal payment systems using network cloud services and distributed cloud repository for storing descriptions of services, thus allowing for the efficient operation of the system of instant payments in a decentralized environment. Developed a distributed system of instant payments as opposed to a centralized system provides fault tolerance at an inaccessible part of the system. Also, the system will reduce the time through the connection of additional services operators already connected to one hour. On the basis of the proposed system is designed software for instant payment.

Аннотация. Статья посвящена разработке и реализации обеспечения единого облачного пространства для осуществления моментальных платежей в пользу поставщиков услуг, таких как газо- и энергосбытовые компании, жилищно-коммунальные услуги и т.д. Проведен анализ предметной области осуществления моментальных платежей. Рассмотрены существующие решения по организации платежных систем и возможности использования сетевых сервисов для организации приема платежей.

Рассмотренный подход к описанию сетевых сервисов, базируется на представлении изменения сервисом состояния окружающего мира. Для описания форматов и методов передачи данных между элементами системы моментальных платежей предлагается использовать язык WSDL, с применением протокола SOAP и интерфейса UDDI, что обеспечивает

взаимодействие программных систем независимо от платформы, простоту разработки и отладки веб-служб.

Представлены схемы, модели, алгоритмы для реализации распределенной платежной системы. При решении поставленной задачи решен вопрос описания единых форматов и методов передачи данных между элементами системы. Предлагается новая схема взаимодействия провайдеров услуг с региональными и федеральными платежными системами с использованием сетевых облачных сервисов и распределенного облачного репозитория, для хранения описаний предоставляемых услуг, что позволяет обеспечить эффективную работу системы моментальных платежей в условиях децентрализации. Разработанная распределенная система моментальных платежей, в отличие от централизованной, обеспечивает отказоустойчивость работы системы при недоступности частей системы. Также система позволяет сократить время подключения дополнительных услуг через, уже подключенных операторов, до одного часа. На базе предложенной системы разработано программное обеспечение для реализации проведения моментальных платежей.

Key words: web application, database, gas and electricity distribution, Network Services, XML, Payments, WSDL.

Ключевые слова: веб-приложение, базы данных, газо- и энергосбытовые компании, сетевые сервисы, XML, платежи, WSDL.

Introduction

Today, delays in payments for the gas and energy supply companies services have become the biggest problem of modern system of energy supply both the population and the economy of Russia. Many gas and electricity supply companies are moving to a system of direct contracts with consumers. In this connection there is a problem of convenience of payments for such services.

This paper deals with the development of algorithms for implementation of cloud system of instant payments – a new high-tech network of points for the provision of financial intermediation services by receiving payments for gas and electricity supply companies services, public utilities and other payments. The mechanism works like a common model of payment acceptance, when the network enters into direct dealer contracts and creates its direct gateways to service operators. At each point the software is installed, by means of which the manager makes payments to customers.

Compared with the conventional model, in which each point independently concludes agreements with all operators and other resources, cloud instant payment system reduces connection time, dramatically simplify settlements, simplify the training of managers through a single interface, eliminate the need to have a supply of cards and considerably expand the range of services, such as payment by customers in other regions.

Service providers provide network services of financial transactions. There are payment systems that provide network services for their work with service providers and their number is constantly growing. In the context of decentralization for effective work we need to unite in a common information space based on cloud technologies. In solving this problem raises the question of subject area presentation in the form of ontology.

To ensure the expansion of the list of services without changing the software we need a unified description of services – cloud repository, and the ability of all participants in the system put data into it and receive information. The purpose of this study is to propose and implement a distributed system of instant payments.

1 The general scheme of the use of network services

By publishing a service meant that the owner of the service wants to make the services of its service known to other network users. That is, the owner of the service is to create a description of the functionality that it provides a

service, and publish it on the network. For those who will use this information, can act as a normal user (human) and computer performing some tasks in an automated process. Thus, in this case requires a formalized representation of services provided by the service provider. We call this representation – description of the Service Offer (SO).

There are several options on how the description of the SO (DSO) can get to the interested user (IU). The first option – when DSO stored in a central repository and IU is looking for a suitable service by himself. In this case, DSO should mainly contain formal information about how to call the service described [1].

The second option – a partial or complete automation of the process of finding a matching service. In this case, a repository for storing DSO may be distributed (data not stored in a central location and distributed to network nodes) or centralized, depending on the network infrastructure. Search is not running by the End User, but the client application. Thus, DSO along with information about how to invoke the service, must contain a functional formalized description of the functional of the service. This information is needed to perform machine-specific searches. When the last are made to find the desired user service, DSO comparison of different services to user requirements should be carried out. From this it follows that the IU must describe the service he needs and this description should include formalized presentation of the required service. A formal presentation of the required user service call user's request description (URD) and the requirements of the user to the service – user request.

The process of comparing various DSO and URD can occur on the client side and on the side of the repository. The first case is suitable for distributed storage options of DSO, and the second for centralized storage options of OPS. Figure 1 shows the general scheme of the network service [9].

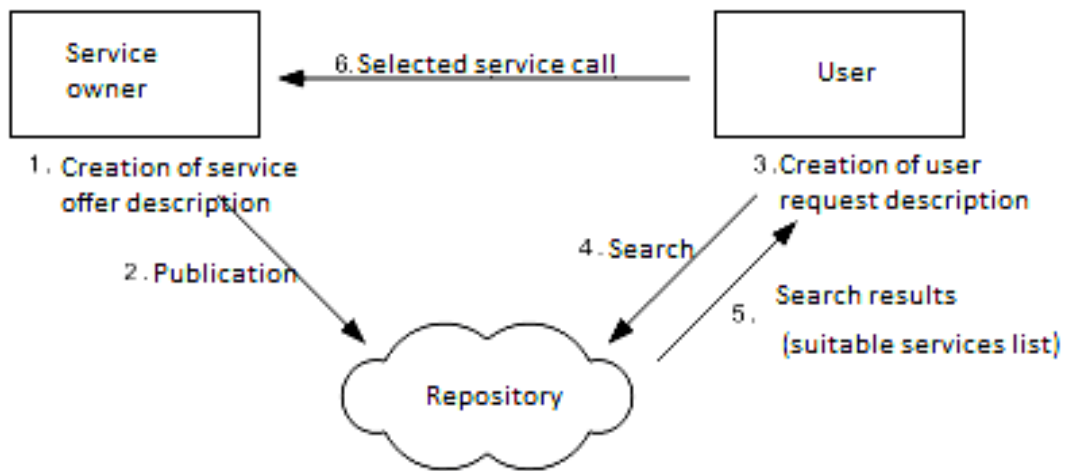


Figure 1. General scheme of the network service

From the analysis of the circuit shown in Figure 2, it follows that the search and the service call may be made by the user or be automated. To automate need to DSO and URD were machine-understandable formal form sufficient to execute machine-dependent operations. For example, automatic comparison of URD with specific DSO. It should also be noted that the expressive ability of formalized description of the service, directly determines the degree of possible automation of the process of using the service. Under the expressive ability (capacity) service description refers to the capacity of relevant display real functionality offered or requested service in a formalized URD or DSO. In other words, the ability to describe the expressive determines how accurately and completely, you can display the actual functionality of the service in the URD or DSO [10].

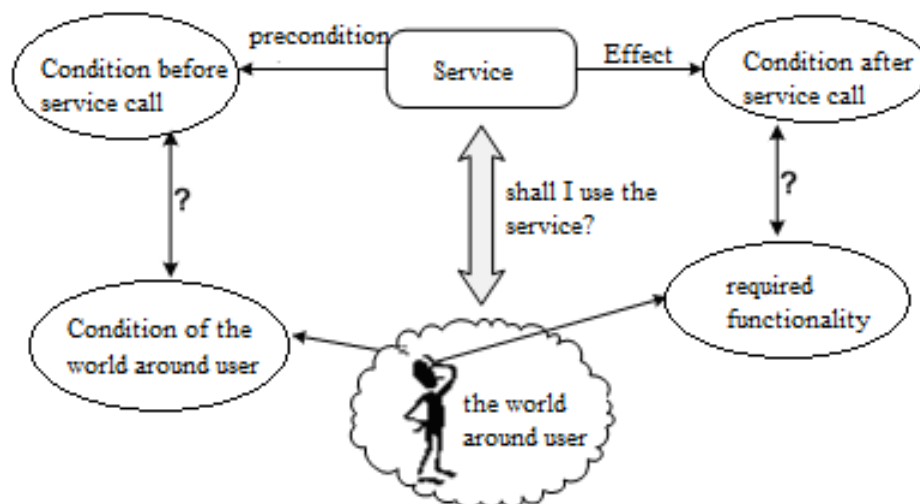


Figure 2. Description of the service functionality based on the states

Thus, the higher the expressive ability of description, the higher the maximum level of automation. In turn, if a service description has high expressive ability, and may transfer the service functionality in sufficient detail when full automation of find and execute operations of the services can be carried out [8].

2 Approaches describing network services

There are several basic approaches for describing network services. The most common approach is the representation of the functionality of the Service by describing the type of incoming and outgoing data service transmitted in the form of text messages. Most languages of service descriptions in this approach, based on syntactic technologies of information, such as XML. XML (eXtensible Markup Language) is a language for describing meta-information. Meta-information in this language is expressed by means of special labels or tags. XML-document using a markup tags may have a strictly defined structure defined XML schema. In turn XML scheme is also XML-document, which defines the set of tags allowed for use, as well as their structure and order of application.

A typical representative of the first approach to the description of the functionality of the Service is the language WSDL (Web Service Description Language – a language for describing web service). This technology is used to describe the most common network services – Web Services. Web service – a software system that can be identified using the URI (Unified Resource Identifier). This system has an open interface defined and described using XML. Other systems may find a web service using its XML-description. They can also call it by sharing with them through the Internet XML-messages, the format of which is given in the XML-description of a web service.

Communication between two Web services is carried out by means of messaging to ensure such an interaction protocol SOAP (Simple Object Access Protocol) is used. SOAP is a simple protocol for the exchange of text messages in XML format between the different network nodes. At the network layer to transmit SOAP-messages used protocols HTTP, HTTPS or SMTP.

Along with technology WSDL and SOAP third important component of Web services technology is a UDDI (Universal Description, Discovery and Integration). UDDI is based in a distributed network directory where users can register their own services or to search for existing ones [5].

Summarizing the above description of technologies WSDL, SOAP, and UDDI can be noted that the expressive power of Web Service Description Language determined expressive power of XML, i.e. representation of the data, which can handle by web services in the process of interaction, is syntactic in nature. Also contributes to the use of technology UDDI, which without additional extensions cannot provide full automation. That is not possible to implement a fully dynamic search and the use of Web services.

The second approach to the description of the existing functionality of the Service is based on the description of the changes he makes on the world during his work. To describe produced in the world changes the concept of state is used. State – a collection of various factors of the world, considered in the context of a particular situation. There are two groups of states: the state prior to the service and state after its use. When you call the service, given that the environment is in a certain precondition, service has an impact on the world and transforms the world into a state corresponding to the effect of the service. Therefore, if formalized description of these states, we obtain a formalized description of the service. This, in turn, is a formal representation of the functionality of the Service. Figure 2 shows the general scheme of service description, based on the states.

Currently, a similar approach to the description of services is applicable only in research projects. Technologies for implementing this approach still under development or are preparing to undergo standardization. In this regard, there are many open questions, a solution of which is required for practical application of this approach.

Obvious is the fact that for the formal presentation of the state of the world requires a description language with a great expressive power. In the current situation, these capabilities have only the so-called semantic technology. Mass devel-

opment of this area began in 2001, after the World Wide Web Consortium (W3C) proposed the concept of the Semantic Web. Semantic Web is a concept network in which each resource in human language would be equipped with a description, a computer understands. An example of such language can serve as a technology RDF (Resource Description Framework). This technology is standardized and has the status of the recommendations of the W3C. One of the fundamental concepts of semantic technologies in general, and RDF in particular, is the concept of ontology. Formal ontology consists of concepts of terms organized in a taxonomy, their descriptions and rules of inference. Taxonomy is hierarchically built system of objectives and results, organized in order from simple to complex system. Mathematically, a taxonomy is a tree structure classification of certain set of objects. Of all the possible varieties can distinguish two types of ontologies. First is an ontology that describes the structure of a certain abstract concepts, and the second type is ontology representing the concrete realization of this concept, or, in other words, its actual implementation in the world. In technology RDF ontology of the second type are represented by the triple: subject–predicate–object. All three elements are represented using the URI (Uniform Resource Identifier). The object can be set as well with the help of primitive types of XML. The predicate is a logical relation between subject and object, showing the relationship between them. The data described using RDF, can be represented in different notations: text in XML format and graphics [11].

The logical development of language RDF (S) is technology OWL (Web Ontology Language – The ontological language of the Internet). This technology is standardized as well and has the status of recommendations to the W3C. OWL compared with RDF has extended operations of inference, for example, integrity checking and automatic classification. Technically OWL – is an add-on RDF, which introduced a new integrated language constructs. Thus, OWL is a much more powerful tool for describing ontologies than RDF.

Two techniques presented above are intended solely to describe the data in the ontological and machine-understandable form. Consequently, the opportuni-

ties they provide is not enough for the realization of the provision and use of services, as they do not contain standard designs to describe the functional and technical features of the service. Technology RDF and OWL have great expressive power, but they do not have specific designs for the service description.

One of the developments using ontological description languages and the representation of various features of network services, technology is OWL-S (formerly called DAML-S). Creating OWL-S (Web Ontology Language) Began in 2000 as a research project organization called DARPA (Defense Advanced Research Projects Agency) [6].

Ontologies in OWL-S is divided into two large groups. The first group – ontologies describing the general knowledge of the world, and the second group – the ontological description of services. In general, the service description in OWL-S consists of three elements: the service profile (Service Profile), the service model (Service Model) and additional technical information (Service Grounding). The second and third elements, respectively, describe the communication and technical features of the service. The first element describes the functional aspect of the service, i.e., indicates "that makes the service".

The most interesting from the point of view of the second approach to the description of services is a service model. Service model provides information as to what steps should be taken of potential users for correct interaction with the service. This information defines when and how the data should be used in the service profile. And also there is actually a formalized representation of service functionality. An important concept in the service model is a process. Process is the conversion of data or states. A service may consist of one or a combination of several processes. Each process defines one transform information and state. Thus, for each process, the input and output data as well as acro- and the resulting state. These elements refer to the four attributes of the service profile. It was mentioned above that in the OWL-S ontologies divided into two types. Service model is part of the ontological description of the service, i.e., belongs to the second type of ontologies. Data streams, as well as states owned by process, covering the concepts of the

world, and are not specific only for the services. From this it follows that they belong to the first type of ontologies. With the help of these ontologies modeled information and situations preceding and following the call of service. Thus, the first and the second type ontologies, herein incorporated in OWL-S, form a semantic representation of the functionality of the Service.

We have considered the approach to describing network services based on the representation of the change service state of the world. A similar approach is used in technologies OWL-S, WSMO and DSD. The undeniable advantage of this approach is the ability to fully automate the process of using the service. Semantic description of the service has great expressive power sufficient to carry out the process of interaction with the service without human intervention. The implication of this is manifested in a strong complication of service description, which is the main drawback of this approach [7].

3 Model of cloud repository of network services

A new scheme of interaction between service providers with regional and federal payment systems using network cloud services and distributed cloud repository for storing descriptions of services, thus allowing for the efficient operation of the system of instant payments in a decentralized environment (Figure 3) [2].

The distributed architecture of the system will allow, in contrast to the traditional centralized, make payments through other elements having access to a gateway operators, as shown in Figure 4. Which will increase the fault tolerance time of the payments and reduce the financial costs of payment [3].

Consider the structure of the repository shown in Figure 5. It consists of two logical parts: local and distributed [4].

The local part contains information about payments made through the local part of the payment system and information about the local agents took payments to service operators.

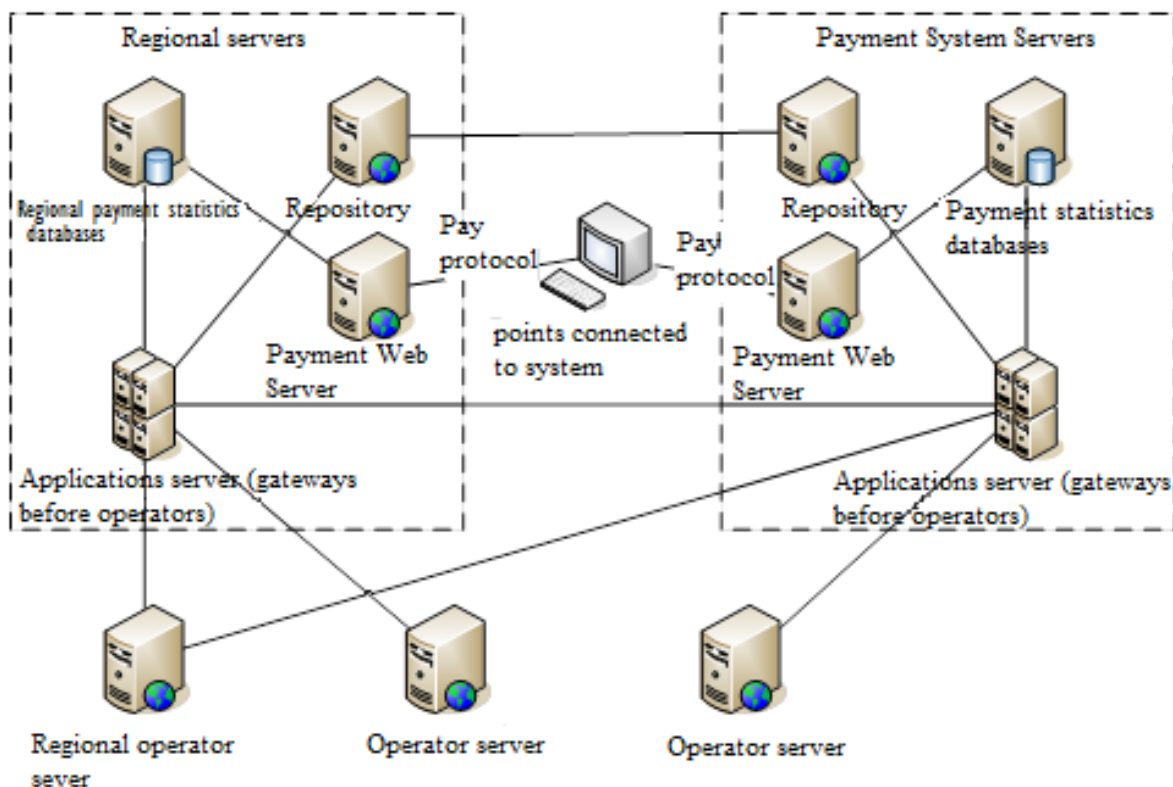


Figure 3. The general scheme of interaction between system elements

Distribution is often distributed repositories synchronized between the payment system and provides information about the services, suppliers, communication protocols and is available to all participants in a distributed system and is used to determine the optimal intermediaries between the service provider or the method of payment.

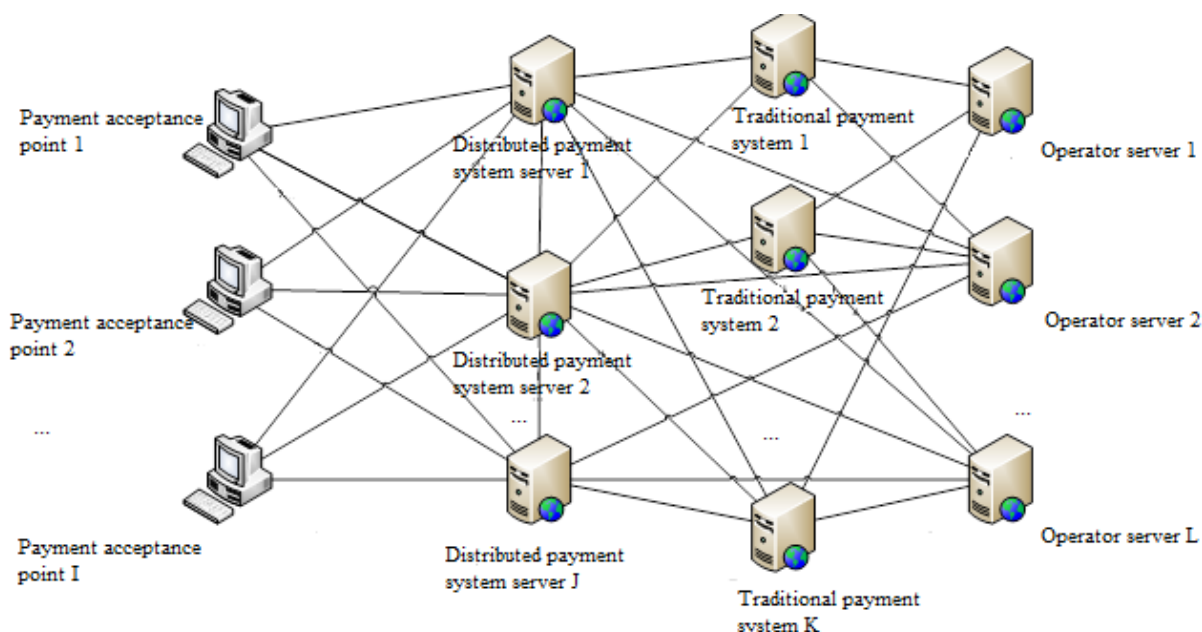


Figure 4. General topology of distributed payment system

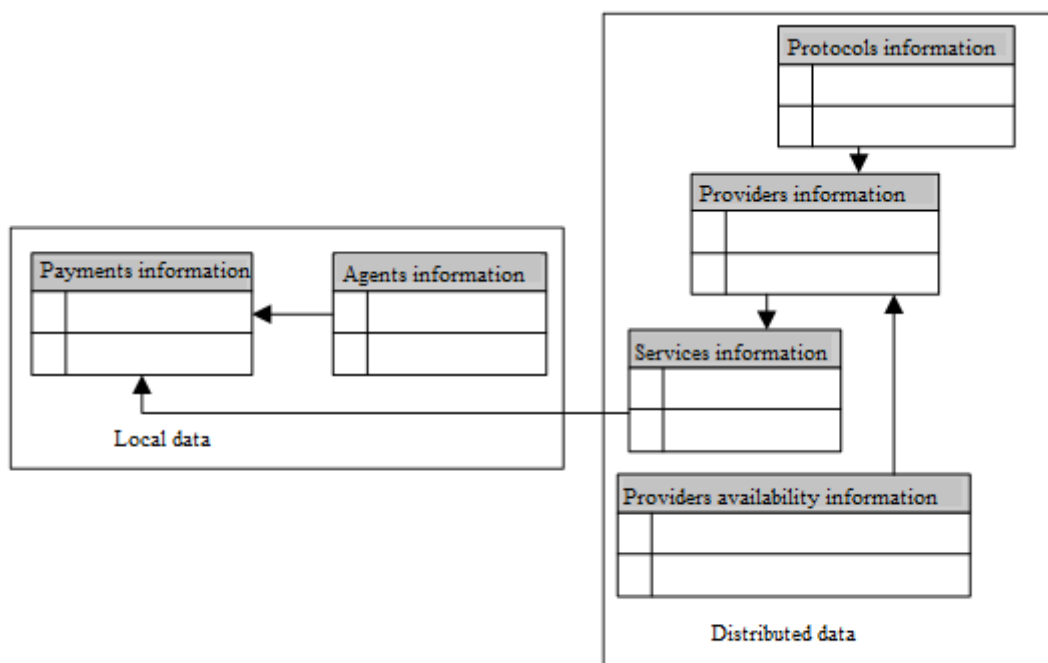


Figure 5. The structure of the repository payment system

Using a common semantic description of services allows you to integrate in a single payment system to provide customers with payment service providers as existing, and future, and the connection of a new provider is a simple trivial operation.

Conclusions

In the context of decentralization to work effectively we need to unite into a unified cloud information space, and it requires the same type description of services hosted in the cloud repository, and the ability of all participants in the system to put data into it and receive information. In solving this problem solved the problem of describing the common formats and methods of data transfer between system elements.

Novelty and practical significance of the results is as follows:

1. A new scheme of interaction between service providers with regional and federal payment systems using a distributed cloud repository for storing descriptions of services, thus allowing for the efficient operation of the system of instant payments in a decentralized environment.

2. To describe the formats and methods of data transfer between system elements instant payments are encouraged to use the language of WSDL, using the SOAP protocol and interface UDDI, which provides interaction of software systems, regardless of platform, ease of development and debugging of Web services.

3. The proposed distributed architecture of the system, as opposed to the traditional centralized, in which some elements of the system can make payments through other elements having access to a gateway operators, which improves fault tolerance time of the payments and reduce the financial costs of payment.

4. Using a common semantic description of services allows you to integrate in a single payment system to provide customers with fees as existing providers, and future, and the connection of a new provider is a simple trivial operation.

5. On the basis of the proposed system is designed software for instant payment to:

- providing fault tolerance of the system at an inaccessible part of the system;
- enabling service management at the level of system administrator without attracting developers;
- receive automatically the best possible rewards from making payment at the possibility of transactions in several payment systems, taking into account the financial and technical capabilities;
- ensuring the provision of information on payments and services.

Введение

Сегодня задержки в платежах за услуги газо- и энергосбытовых компаний стали бичем современной системы снабжения энергоресурсами как населения, так и всей экономики России. Многие газо- и энергосбытовые компании переходят на систему прямых договоров с потребителями. В связи с чем встает вопрос удобства оплаты таких услуг.

В данной работе рассматриваются вопросы разработки алгоритмов облачной реализации системы моментальных платежей – это новые высокотехнологичные сети точек по предоставлению услуг финансового посредничества путем приема оплаты за услуги газо- и энергосбытовых компаний, жилищно-коммунальные услуги и другие платежи. Механизм работы похож на распространенную модель приема платежей, когда сеть включает прямые дилерские договоры и создает свои прямые шлюзы с операторами услуг. В каждой точке устанавливается программное обеспечение, с помощью которого менеджер производит платежи клиентов.

По сравнению с обычной моделью - по которой каждая точка независимо заключает договоры со всеми операторами и другими ресурсами, облачная система моментальных платежей позволяет сократить время подключения, резко упростить взаиморасчеты, упростить подготовку менеджеров за счет единого интерфейса, избавиться от необходимости иметь запас карт и заметно расширить спектр услуг, например за счет оплаты клиентов других регионов.

Провайдеры услуг предоставляют сетевые сервисы финансовых операций. Существуют платежные системы, которые предоставляют свои сетевые сервисы для работы с провайдерами услуг и их число постоянно растет. В условиях децентрализации для эффективной работы необходимо объединение в единое информационное пространство на основе облачных технологий. При решении поставленной задачи встает вопрос представления предметной области в виде онтологии.

Для обеспечения расширения перечня услуг без изменения программного обеспечения необходимо единое описание услуг – облачный репозиторий и возможность всех участников системы помещать в него данные и получать информацию. Целью данного исследования является предложение и реализация распределенной системы моментальных платежей.

1 Общая схема использования облачных сервисов

Под публикацией некоторого сервиса подразумевается, что владелец данного сервиса хочет сделать услуги своего сервиса известными другим пользователям сети. То есть владелец сервиса, должен создать описание того функционала, который предоставляет его сервис и опубликовать его в сети. В качестве тех, кто будет использовать эту информацию, может выступать как обычный пользователь (человек), так и компьютер, выполняющий некоторые задачи в автоматизированном процессе. Таким образом, в данном случае требуется формализованное представление предоставляемых сервисом услуг. Назовем данное представление – описание Предложения Сервиса (ПС).

Существует несколько вариантов того, как описание ПС (ОПС) может попасть к интересующемуся пользователю (ИП). Первый вариант – это когда ОПС хранится в централизованном репозитории, и ИП сам ищет там подходящий ему сервис. В данном случае, ОПС должно в основном содержать формализованную информацию о том, как вызывается описываемый сервис [1].

Второй вариант – это частичная или полная автоматизация процесса поиска необходимого сервиса. В этом случае репозиторий для хранения ОПС может быть как распределенным (данные хранятся не в одном центральном месте, а распределены по сетевым узлам), так и централизованным, в зависимости от инфраструктуры сети. Поиск же выполняется не конечным пользователем, а клиентским приложением. Таким образом, в ОПС наряду с информацией о том, как вызывается сервис должно содер-

жать формализованное описание функционала сервиса. Эта информация нужна для выполнения машинно-зависимых операций поиска. При выполнении последних, чтобы найти необходимый пользователю сервис, должно быть проведено сравнение ОПС различных сервисов с требованиями пользователя. Из этого следует, что ИП должен описать необходимый ему сервис и данное описание должно содержать формализованное представление требуемого сервиса. Формализованное представление необходимого пользователю сервиса назовем Описанием Запроса Пользователя (ОЗП), а требования пользователя к сервису – Запрос Пользователя.

Процесс сравнения различных ОПС и ОЗП может происходить, как на стороне пользователя, так и на стороне репозитория. Первый случай подходит для распределенного варианта хранения ОПС, а второй для централизованного варианта хранения ОПС. На рисунке 1 представлена общая схема использования сетевого сервиса [9].

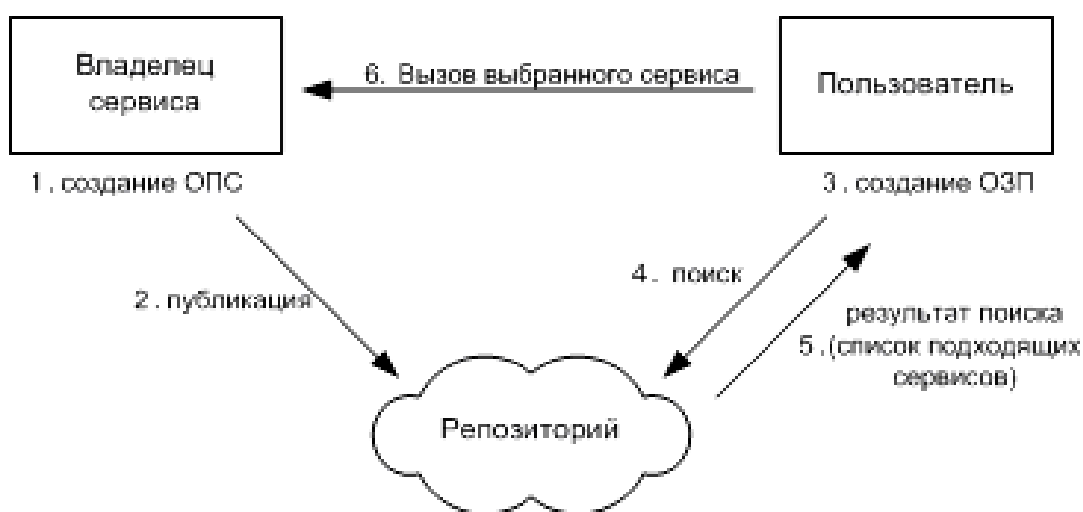


Рисунок 1. Общая схема использования сетевого сервиса

Из анализа схемы, приведенной на рисунке 2, следует, что процессы поиска и вызова сервиса могут осуществляться пользователем или быть автоматизированы. Для автоматизации необходимо, чтобы ОПС и ОЗП имели машинно-понятную формализованную форму, достаточную для выполнения машинно-зависимых операций. Например, автоматического сравнения ОЗП с конкретным ОПС.

Следует также отметить тот факт, что выразительная способность формализованных описаний сервиса напрямую определяет степень возможной автоматизации процесса использования сервиса. Под выразительной способностью (мощностью) описания сервиса понимается потенциал соответствия отображения реальной функциональности предлагаемого или запрашиваемого сервиса в формализованном ОЗП или ОПС. Другими словами, выразительная способность описания определяет, насколько точно и полно можно отобразить реальную функциональность сервиса в ОЗП или ОПС [10].

Таким образом, чем выше выразительная способность описания, тем выше максимальный уровень автоматизации. В свою очередь, если описание сервиса обладает высокой выразительной способностью и может достаточно детально передать функциональность услуг сервиса, то при таких условиях может быть осуществлена полная автоматизация операций поиска и выполнения сервиса [8].

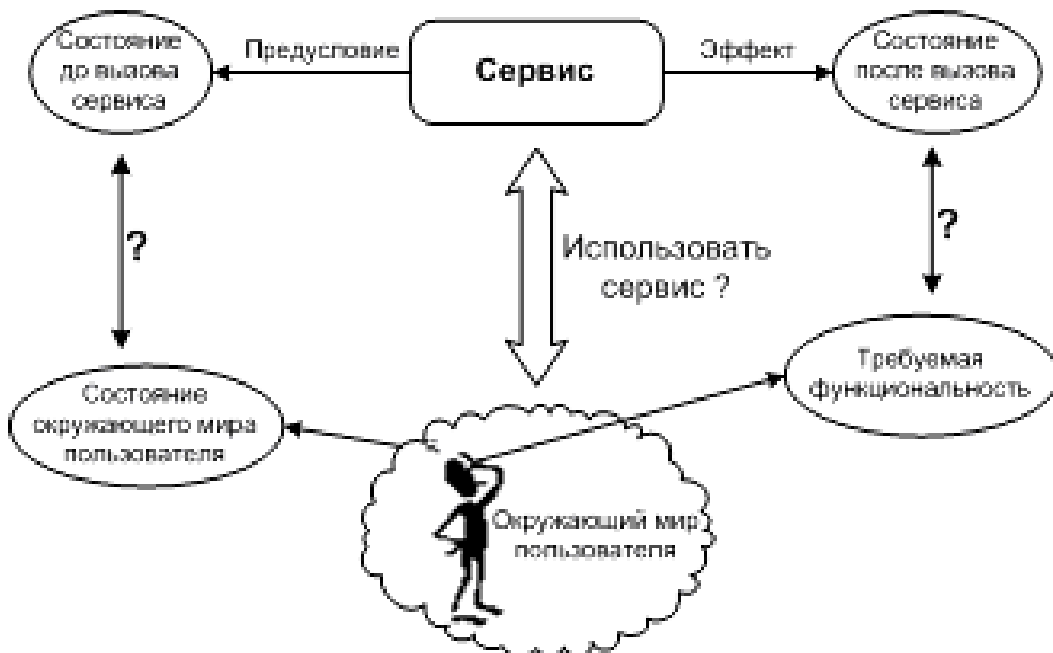


Рисунок 2. Описание функциональности сервиса, основанное на разных состояниях

2 Подходы описания облачных сервисов

Существует несколько основных подходов описания сетевых сервисов. Наиболее распространенный подход это представления функциональности услуг сервиса при помощи описания вида входящих и исходящих данных сервиса, передающихся в виде текстовых сообщений. Большинство языков описания сервисов в данном подходе, базируются на технологиях синтаксического представления информации, например, таких как XML. XML (eXtensible Markup Language (англ.) – расширяемый язык разметки) представляет собой язык для описания метаданных. Метаданные в этом языке выражаются при помощи специальных меток или тегов (англ. tag). XML-документ, размеченный при помощи тегов, может иметь жестко заданную структуру, заданную XML-схемой. XML-схема является, в свою очередь, тоже XML-документом, в котором определяется набор тегов, разрешенный для использования, а также их структура и порядок применения.

Типичным представителем первого подхода к описанию функциональности услуг сервиса является язык WSDL (Web Service Description Language (англ.) – язык описания веб-сервиса). Данная технология применяется для описания наиболее распространенных сетевых сервисов – веб-сервисов (Web-Service). Веб-сервис – это программная система, которая может быть идентифицирована при помощи URI (Unified Resource Identifier (англ.) – универсальный идентификатор ресурса). Данная система имеет открытый интерфейс, определенный и описанный при помощи XML. Другие системы могут находить веб-сервис при помощи его XML-описания. Также они могут вызывать его, обмениваясь с ним через Интернет XML-сообщениями, формат которых задан в XML-описании веб-сервиса.

Коммуникация между двумя веб-сервисами осуществляется при помощи обмена сообщениями, для обеспечения подобного взаимодействия используется протокол SOAP (Simple Object Access Protocol (англ.) – протокол

доступа к простым объектам). SOAP – это простой протокол для обмена текстовыми сообщениями в формате XML между различными сетевыми узлами. На сетевом уровне для передачи SOAP-сообщений применяются протоколы HTTP, HTTPS или SMTP.

Наряду с технологиями WSDL и SOAP третьей важной составляющей веб-сервисов является технология UDDI (Universal Description, Discovery and Integration (англ.) – универсальное описание, поиск и интеграция). UDDI представляет собой базирующийся в сети распределенный каталог, в котором пользователи могут регистрировать собственные сервисы или искать существующие [5].

Обобщая приведенное выше описание технологий WSDL, SOAP и UDDI можно отметить, что выразительная мощность описаний веб-сервисов определяются выразительно мощностью языка XML, т.е. представление данных, которыми могут оперировать веб-сервисы в процессе взаимодействия, носит синтаксический характер. Свой вклад вносит и применение технологии UDDI, которая без дополнительных расширений не может обеспечить полную автоматизацию. То есть невозможно реализовать полностью динамический поиск и использование веб-сервисов.

Второй существующий подход к описанию функциональности услуг сервиса базируется на описании изменений, которое он производит на окружающий мир во время своей работы. Для описания производимых на окружающий мир изменений используется понятие состояние. Состояние – это совокупность разнообразных факторов окружающего мира, рассматриваемых в контексте конкретной ситуации. Выделяют две группы состояния: состояния до начала работы сервиса и после его использования. При вызове сервиса, с учетом того, что окружающая обстановка находится в определенном предусловии состоянии, сервис оказывает влияние на окружающий мир и переводит окружающий мир в состояние, соответствующее эффекту от работы сервиса. Следовательно, если формализовать описание данных состояний, то мы получим формализованное описание рабо-

ты сервиса. Это, в свою очередь, и является формализованным представлением функциональности услуг сервиса. На рисунке 2 представлена общая схема описания сервиса, базирующегося на состояниях.

В настоящее время подобный подход к описанию сервисов применяется только в научно-исследовательских проектах. Технологии для реализации этого подхода еще только находятся на стадии разработки или готовятся к прохождению стандартизации. В связи с этим существует большое количество открытых вопросов, решение которых требуется для практического применения данного подхода.

Очевидным является тот факт, что для формального представления состояния окружающего мира, требуется язык описания, обладающий большой выразительной способностью. В сложившейся на данный момент ситуации подобными возможностями обладают только, так называемые, семантические технологии. Массовое развитие этой области началось в 2001 году, после того как Консорциум Всемирной паутины (W3C) предложил концепцию Семантическая паутина (Semantic web (англ.)). Семантическая паутина – это концепция сети, в которой каждый ресурс на человеческом языке был бы снабжён описанием, понятным компьютеру. В связи с этим был разработан ряд языков, при помощи которых понятия из человеческого языка, можно выражать в машинно-понятной форме с сохранением их смысла. Примером подобного языка может служить технология RDF (Resource Description Framework (англ.)). Данная технология стандартизована и имеет статус рекомендации Консорциума W3C. Одним из основополагающих понятий семантических технологий в целом, и RDF в частности, является понятие онтологии. Формально онтология состоит из понятий терминов, организованных в таксономию, их описаний и правил вывода. Таксономия – иерархически выстроенная система целей и результатов, организованная в порядке: от простой к сложной системе. Математически, таксономией является древообразная структура классификаций определенного набора объектов. Из всего возможного многообразия можно выделить

два типа онтологий. Первый – это онтологии, которые описывают структуру некоего абстрактного понятия, а второй тип – это онтологии, представляющие конкретную реализацию данного понятия, или, другими словами, его реальное воплощение в окружающем мире. В технологии RDF онтологии второго типа представляются тройкой: Субъект – Предикат – Объект. Все три элемента представляются при помощи URI (Uniform Resource Identifier (англ.) – унифицированный идентификатор ресурса). Объект может быть также задан при помощи примитивных типов XML. Предикат представляет собой логическое отношение между субъектом и объектом, показывающее взаимосвязь между ними. Данные, описываемые при помощи RDF, можно представить в различных 2-х нотациях: текстовой в формате XML и графической [11].

Логическим развитием языка RDF(S) выступает технология OWL (Web Ontology Language (англ.) – онтологический язык сети Интернет). Данная технология также стандартизована и имеет статус рекомендации Консорциума W3C. OWL по сравнению с RDF имеет расширенные операции логического вывода, например, проверка целостности или автоматическое классифицирование. Технически OWL – это надстройка над RDF, в которой введены новые встроенные языковые конструкции. Таким образом, OWL является гораздо более мощным инструментом для описания онтологий, чем RDF.

Две представленные выше технологии предназначаются исключительно для описания данных в онтологической и машинно-понятной форме. Следовательно, предоставляемых ими возможностей недостаточно для реализации системы предоставления и использования сервисов, так как они не содержат стандартных конструкций для описания функциональных и технических особенностей сервисов. Технологии RDF и OWL обладают большой выразительной мощностью, но у них не хватает специфичных конструкций для описания сервиса.

Одной из разработок использующих языки онтологического описания и возможность представления разнообразных особенностей сетевых сервисов, является технология OWL-S (ранее называлась DAML-S). Создание OWL-S (Web Ontology Language –Services (англ.)) началось в 2000 году в качестве исследовательского проекта организации под названием DARPA (Defense Advanced Research Projects Agency (англ.) – агентство по исследованию перспективных оборонных проектов) [6].

Онтологии в OWL-S разделяются на две большие группы. Первая группа – это онтологии, описывающие общие знания из окружающего мира, а вторая группа – онтологические описания сервисов. В общем случае описание сервиса в OWL-S состоит из 3 элементов: Профиля Сервиса (Service Profile), Модели Сервиса (Service Model) и Дополнительной Технической Информации (Service Grounding). Второй и третий элементы соответственно описывают коммуникационные и технические особенности сервиса. Первый элемент описывает функциональную сторону сервиса, т.е. указывает «что делает сервис».

Наиболее интересным, с точки зрения второго подхода к описанию сервисов, является Модель Сервиса. Модель Сервиса содержит информацию относительно того, какие шаги должен предпринять потенциальный пользователь, для осуществления корректного взаимодействия с сервисом. Эта информация определяет, когда и как должны использоваться данные из Профиля Сервиса. А также тут фактически находится формализованное представление функциональности сервиса. Важным понятием в Модели Сервиса является Процесс. Процесс представляет собой преобразование данных или состояний. Сервис может состоять из одного или комбинации нескольких процессов. Каждый процесс определяет одно преобразование информации и состояния. Таким образом, для каждого процесса задаются входные и выходные данные, а также инициальное и результирующее состояния. На эти элементы и ссылаются четыре атрибута из Профиля Сервиса. Выше упоминалось о том, что в OWL-S онтологии разделяются на

два типа. Модель Сервиса является частью онтологического описания сервиса, т.е. принадлежит к онтологиям второго типа. Потоки данных, а также состояния принадлежащие процессу, охватывают понятия из окружающего мира, и не являются специфичными только для сервисов. Из этого следует, что они принадлежат онтологиям первого типа. При помощи этих онтологий моделируется информация и ситуации предшествующие и последующие за вызовом сервиса. Таким образом, онтологии первого и второго типа, объединенные в описании OWL-S, образуют семантическое представление функциональности услуг сервиса.

Рассмотренный подход к описанию сетевых сервисов, базирующийся на представлении изменения сервисом состояния окружающего мира, используется в технологиях OWL-S, WSMO и DSD. Неоспоримым плюсом данного подхода является возможность полной автоматизации процесса использования сервиса. Семантическое описание сервиса обладает большой выразительной мощностью, достаточной для реализации процесса взаимодействия с сервисом без участия человека. Следствие этого проявляется в сильном усложнении описания сервиса, что и является главным недостатком данного подхода [7].

3 Модель распределенного облачного репозитория сетевых сервисов

Предлагается новая схема взаимодействия провайдеров услуг с региональными и федеральными платежными системами с использованием сетевых облачных сервисов и распределенного облачного репозитория, для хранения описаний предоставляемых услуг, что позволяет обеспечить эффективную работу системы моментальных платежей в условиях децентрализации (рисунок 3) [2].

Распределенная архитектура системы позволит, в отличие от традиционной, централизованной, осуществлять платежи через другие элементы, имеющие доступ к шлюзам операторов, как показано на рисунке 4. Что

позволит повысить отказоустойчивость, время прохождения платежей и снизить финансовые затраты осуществления платежа [3].

Рассмотрим структуру облачного репозитория изображенную на рисунке 5. Она состоит из двух логических частей: локальной и распределенной [4].

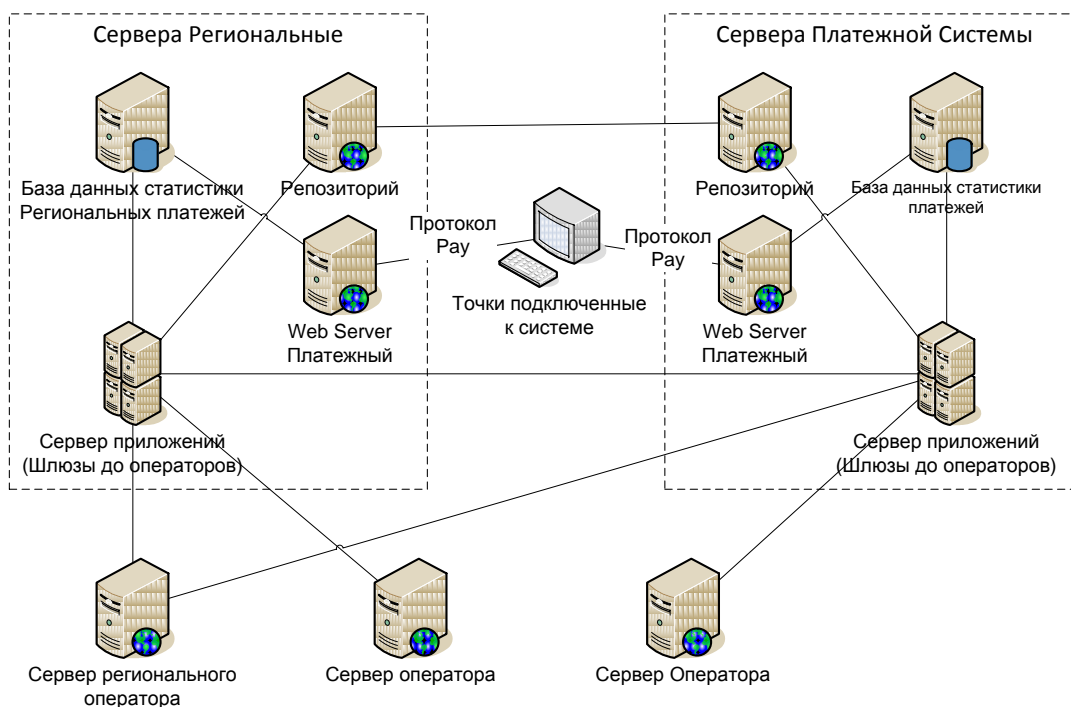


Рисунок 3. Общая схема взаимодействия элементов системы

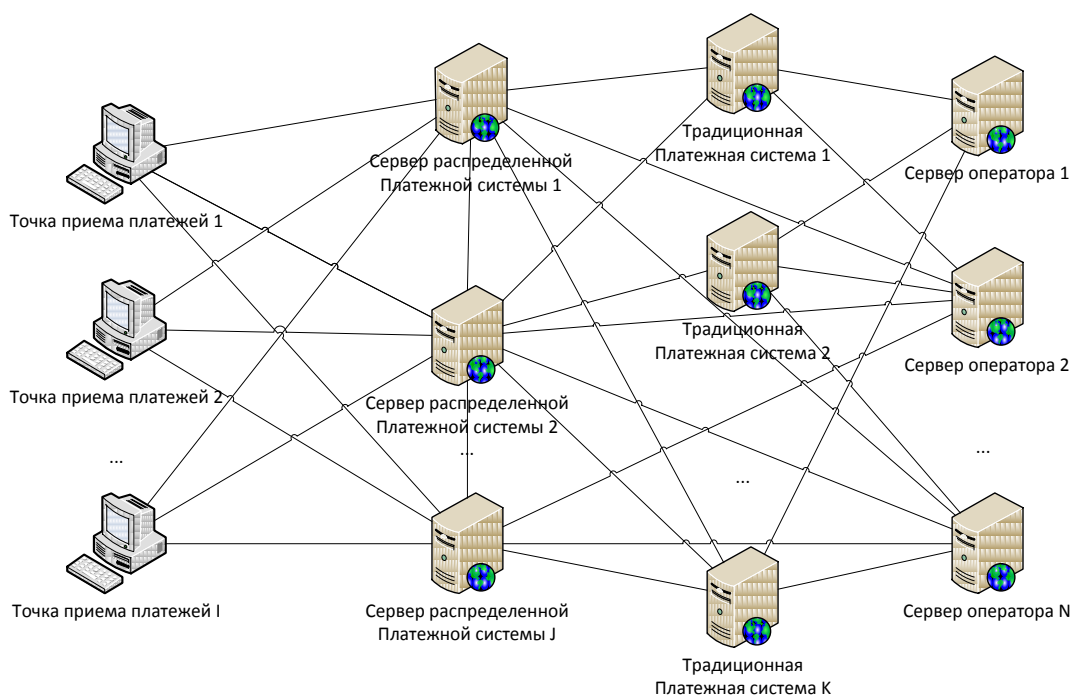


Рисунок 4. Общая топология распределённой платёжной системы

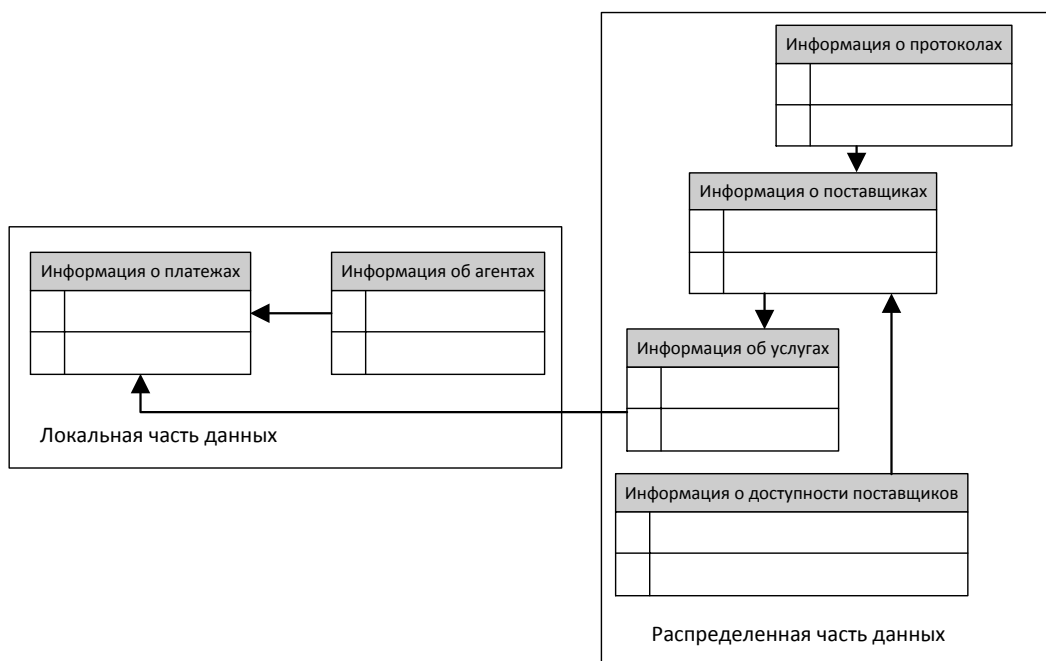


Рисунок 5. Структура облачного репозитория платежной системы

Локальная часть содержит информацию о платежах, проведенных через местную часть платежной системы и информацию о локальных агентах принявших платежи в пользу операторов услуг.

Распределенная часть синхронизируется между репозиториями распределенной платежной системы и содержит информацию об услугах, поставщиках, протоколах передачи информации и доступна всем участникам распределенной системы и используется для определения оптимального между посредниками или поставщиком услуги способа проведения платежа.

Использование единого семантического описания сервисов позволяет интегрировать в рамках одной платежной системы оказание клиентам оплаты услуг, как имеющихся провайдеров, так и будущих, при этом подключение нового провайдера является простой тривиальной операцией.

Выводы

В условиях децентрализации для эффективной работы необходимо объединение в единое облачное информационное пространство, а для этого необходимо однотипное описание услуг, размещенное в облачном репозитории и возможность всех участников системы помещать в него данные

и получать информацию. При решении поставленной задачи решен вопрос описания единых форматов и методов передачи данных между элементами системы.

Новизна и практическая значимость полученных результатов состоит в следующем:

1. Предложена новая схема взаимодействия провайдеров услуг с региональными и федеральными платежными системами с использованием распределенного облачного репозитория, для хранения описания предоставляемых услуг, что позволяет обеспечить эффективную работу системы моментальных платежей в условиях децентрализации.

2. Для описания форматов и методов передачи данных между элементами системы моментальных платежей предлагается использовать язык WSDL, с применением протокола SOAP и интерфейса UDDI, что обеспечивает взаимодействие программных систем независимо от платформы, простоту разработки и отладки веб-служб.

3. Предложенная распределенная архитектура системы, в отличие от традиционной централизованной, в которой одни элементы системы могут осуществлять платежи через другие элементы, имеющие доступ к шлюзам операторов, что позволяет повысить отказоустойчивость, время прохождения платежей и снизить финансовые затраты осуществления платежа.

4. Использование единого семантического описания сервисов позволяет интегрировать в рамках одной платежной системы оказание клиентам оплату услуг, как имеющихся провайдеров, так и будущих, при этом подключение нового провайдера является простой тривиальной операцией.

5. На базе предложенной системы разработано ПО для проведения моментальных платежей, направленное на:

- обеспечение отказоустойчивости работы системы при недоступности частей системы;
- предоставление возможности управления услугами на уровне администратора системы без привлечения разработчиков;

- получение в автоматическом режиме максимально возможного вознаграждения от проведения платежа, при возможности проведения транзакции через несколько платежных систем с учетом финансовой и технической возможности;
- обеспечение предоставления информации по проведенным платежам и услугам.

References

1 Kotelnikov V.A. Statement of the creating problem a distributed system of instant payments. // Intelligent Information Processing and Management. Collection of articles Regional Winter School-Seminar graduate students and young scientists, Februari 16-19, 2006. Ufa: Technology, 2006. T.1. pp. 25-28 [in Russian].

2 Kotelnikov V.A. Distributed system architecture instant payments // Intelligent Information Processing and Management. Collection of articles Regional Winter School-Seminar graduate students and young scientists, Februari 16-19, 2006. Ufa: Technology, 2006. T.2. pp. 118-123. [in Russian].

3 Kotelnikov V. A., Bogdanova D.R. Distributed system for financial transactions // Computer Science and Information Technology» (CSIT'2006). Proceedings of the 8th International Conference. Karlsruhe, Germany, 2006. T.2. pp. 56-64.

4 Kotelnikov V. A. Distributed system of instant payments on the basis of network services // Technologies Microsoft in the theory and practice of programming. Abstracts of the conference contest of works by students, graduate students and young scientists. Novosibirsk, Akademgorodok: NSU, 2007. pp. 223-225. [in Russian].

5 Kotelnikov V. A. About technologies to create a system of network services for financial transactions on the basis of ontological approach // Information Technologies and Mathematical Methods in Economics: Bashkir-Saxon forum, proceedings of the round table. Ufa: USATU, 2007. pp. 78-89.

6 Kotelnikov V. A. Ontological description of network services for distributed system of financial operations // Proceedings of the Kazan School of Computer and Cognitive Linguistics TEL-2006. Kazan: Otechestvo, 2007. pp. 127-138. [in Russian].

7 Kotelnikov V. A., Popov D.V. Development of network services based on ontologies for e-commerce: principles and technology // Review of Applied and Industrial Mathematics. 2008. T.15. B.1. pp. 145-148. [in Russian].

8 Kotelnikov V. A., Popov D.V., Bogdanova D.R. Description of networked services for a distributed system of financial operations on the basis of the ontological approach // Proceedings of the 11 International Workshop on Computer Science and Information Technologies. Crete, Greece, October 5–8, 2009. T.1. pp.147-152.

9 Kotelnikov V. A., Bogdanova D. R. Distributed system of financial operations based on network services // Proceedings of the 12 International Workshop on Computer Science and Information Technologies. Moscow – St-Petersburg, October 5–8, 2010. T.3–4. pp.157-162.

10 Bogdanova D. R., Popov D.V., Kotelnikov V. A., Formalization of the preferences of cloud services for the system of financial transactions based on ontologies // Information and Communication Technologies in science, business and education: The Fifth International Scientific Conference. Stavropol: North Caucasus Humanitarian and Technical Institute. 2012. Pp. 176-180. [in Russian].

11 Vainerman I.A. Factors and problems of the use of network services // Vestnik UGATU. 2006. T.8. № 4. pp. 43-47. [in Russian].

Список используемых источников

1 Котельников В.А. Постановка задачи создания распределенной системы моментальных платежей // Интеллектуальные системы обработки информации и управления: сб. ст. Региональной зимней школы–семинара аспирантов и молодых ученых, 16-19 февр. 2006. Уфа: Изд-во «Технология», 2006. Т.1. С. 25-28.

2 Котельников В.А. Архитектура распределенной системы моментальных платежей // Интеллектуальные системы обработки информации и управления: сб. ст. Региональной зимней школы–семинара аспирантов и молодых ученых, 16-19 февр. 2006. Уфа: Изд-во «Технология», 2006. Т.2. С. 118-123.

3 Kotelnikov V. A., Bogdanova D.R. Distributed system for financial transactions // Computer Science and Information Technology» (CSIT'2006). Proceedings of the 8th International Conference. Karlsruhe, Germany, 2006. Т.2. pp. 56-64.

4 Котельников В.А. Распределенная система моментальных платежей на основе сетевых сервисов // Технологии Microsoft в теории и практике программирования: тез. докл. конф.-конкурса работ студентов, аспирантов и молодых ученых. Новосибирск: Академгородок: НГУ, 2007. С. 223-225.

5 Kotelnikov V. A. About technologies to create a system of network services for financial transactions on the basis of ontological approach // Information Technologies and Mathematical Methods in Economics: Bashkir-Saxon forum, proceedings of the round table. Ufa: USATU, 2007. pp. 78-89.

6 Котельников В.А. Онтологическое описание сетевых сервисов для распределенной системы финансовых операций // Тр. Казанской школы по компьютерной и когнитивной лингвистике TEL-2006. Казань: Отечество, 2007. С. 127-138.

7 Котельников В.А., Попов Д.В. Разработка сетевых сервисов на основе онтологий для Интернет-коммерции: принципы и технологии // Обозрение прикладной и промышленной математики. 2008. Т.15. Вып.1. С. 145-148

8 Kotelnikov V. A., Popov D.V., Bogdanova D. R. Description of networked services for a distributed system of financial operations on the basis of the ontological approach // Proceedings of the 11 International Workshop on Computer Science and Information Technologies. Crete, Greece, October 5–8, 2009. Т.1. pp.147-152.

9 Kotelnikov V.A., Bogdanova D.R. Distributed system of financial operations based on network services // Proceedings of the 12 International Workshop on Computer Science and Information Technologies. Moscow – St-Petersburg, October 5–8, 2010. Т.3–4. pp.157-162.

10 Богданова Д.Р., Попов Д.В., Котельников В.А. Формализация предпочтений облачных сервисов для системы финансовых операций на основе онтологий // Инфокоммуникационные технологии в науке, производстве и образовании: пятая междунар. науч.-техн. конф. Ставрополь: Северо-Кавказский гуманитарно-технический институт. 2012. С. 176-180.

11 Вайнерман И.А. Факторы и проблемы предоставления и использования сетевых сервисов // Вестник УГАТУ. 2006. Т.8. № 4. С. 43–47.

About the authors

Сведения об авторах

V.A. Kotelnikov, Senior Teacher of FSBEI HPE “Ufa State Aviation Technical University”, Ufa, the Russian Federation

Котельников В.А., старший преподаватель ФГБОУ ВПО УГАТУ, г. Уфа, Российская Федерация

e-mail: vit_kot@mail.ru

T. I. Grigorchuk, Candidate of Engineering Sciences, Associate Professor of the Chair “Mathematics” FSBEI HPE USPTU, Ufa, the Russian Federation

Григорчук Т. И., канд. техн. наук, доцент кафедры «Математика», ФГБОУ ВПО УГНТУ, г. Уфа, Российская Федерация

e-mail: tgrigor@yandex.ru

N.I. Yusupova, Doctor of Engineering Sciences, Professor of FSBEI HPE “Ufa State Aviation Technical University”, Ufa, the Russian Federation

Юсупова Н.И., д-р техн. наук, профессор ФГБОУ ВПО УГАТУ, г. Уфа, Российская Федерация

e-mail: yussupova@ugatu.ac.ru