

УДК 004.75, 004.772, 004.62

ПРОБЛЕМЫ ЭФФЕКТИВНОГО ИСПОЛЬЗОВАНИЯ СЕТЕВЫХ СЕРВИСОВ

А.М. Дергачев

Рассматриваются основные проблемы, которые возникают при использовании сетевых сервисов на этапах поиска, выбора и интеграции. Рассмотрены качественные характеристики, по которым можно производить сравнение и выбор сервисов, и абстрактная модель, наглядно демонстрирующая взаимодействие сетевых сервисов.

Ключевые слова: сетевой сервис, проблемы, модель, взаимодействие, качество сервиса.

Введение

С увеличением числа организаций, вовлеченных в использование технологии сетевых сервисов, растет количество сетевых сервисов, предоставляющих родственные услуги на получение однотипной информации. Прозрачность выбора между родственными по своим функциям сервисами, а также эффективность поиска среди сервисов запрашиваемой информации или услуги может быть обеспечена методами оптимизации поиска и построением плана выполнения запроса. Автоматизация проектирования таких систем может быть обеспечена набором типовых решений, поддерживаемых интегрированной средой разработки. В свою очередь, необходима среда выполнения, реализующая соответствующие алгоритмы планирования и выполнения запроса поиска по сетевым сервисам в соответствии с выбранными на стадии проектирования типовыми решениями.

Постановка задачи

Основной целью данной работы является описание аналитической модели взаимодействия сетевых сервисов и выбор математической формы ее представления, позволяющей исследовать эффективность использования сетевых сервисов программными приложениями корпоративных информационных систем и влияние на нее различных методов оптимизации поиска и алгоритмов построения плана выполнения запроса по сетевым сервисам. Результаты исследования планируется использовать для разработки типовых решений проектирования программных приложений корпоративных информационных систем, основанных на указанных выше методах, и программной среды их выполнения.

Описание проблем использования сетевых сервисов

Первая проблема – это проблема поиска сетевых сервисов. На сегодняшний день существует два различных подхода: использование реестров и использование поисковых систем. Каждый из подходов имеет свои преимущества и недостатки. Реестры, как правило, обладают структурой и реализуют стандарт UDDI, но ограничены списком сервисов, входящих в базу данных. Поисковые системы производят поиск по всему Интернету, но результаты поиска не содержат какого-либо описания сервиса. Последние исследования [1] показывают, что поисковые системы находят большее количество Интернет-сервисов и процент рабочих сервисов от всех найденных у поисковых систем выше. Но ввиду сложности автоматизации использования результатов поисковых систем отдавать им предпочтение преждевременно.

В большинстве случаев результатом поиска будет не один сервис, а список сервисов, который может быть достаточно большим (в Интернете существуют сотни, а может быть, и тысячи сетевых сервисов, предоставляющих прогноз погоды). Появляется проблема выбора. Основная сложность с выбором единственного сервиса из множества аналогичных заключается в определении набора параметров, по которым будет производиться выбор. Целью выбора является предоставление пользователю максимального качества сервиса. Для описания качества сервиса (QoWS) [2] предлагается использовать различные

характеристики сервисов, такие как доступность, надежность, быстродействие, безопасность, соответствие стандартам и др. Сбор и хранение такой информации должны быть организованы отдельно от самих сервисов, в какой-то централизованной системе, так как поставщик сервиса может предоставить заведомо ложную информацию и тем самым обмануть пользователя. Также стоит предусмотреть возможность выбора сервиса по нескольким параметрам QoS, а значит, необходимо решить задачу многокритериальной оптимизации. Но выбор сервиса – не последняя проблема, с которой придется столкнуться при использовании сервисов.

После того как сервисы найдены и отобраны, необходимо организовать их взаимодействие. На этом этапе возникает сразу несколько проблем, такие как синтаксическая совместимость, семантическая совместимость, проблемы, связанные с глобализацией и локализацией.

Проблема синтаксической совместимости связана с тем, что сервисы могут быть реализованы на разных программных и аппаратных платформах, и необходимо организовать обмен данными между ними. Для организации взаимодействия необходимо знать список доступных операций и параметров, типы данных параметров и возвращаемых результатов. Для решения этой проблемы создан язык описания сетевых сервисов (WSDL), который позволяет автоматизировать процесс их разработки и вызова.

Проблема семантической совместимости намного сложнее, и готового решения на сегодняшний день здесь нет. Построение семантического графа [3] и использование теоретико-множественного подхода к получению логических выводов на основе баз знаний [4] позволяет наделять операции и параметры поиска физическим смыслом и использовать для их формализации языки описания онтологии, например, OWL.

Вечной проблемой интеграции различных приложений являются глобализация и локализация. При организации взаимодействия различных сетевых сервисов нужно учесть различия в представлении данных в разных странах: форматы дат и денежных сумм, различные единицы измерения массы, длины, температуры и т.д. Несмотря на то, что сетевые сервисы являются машинно-ориентированными приложениями, они могут оперировать текстовыми данными, а значит, нужно предусмотреть и возможность перевода текстов с одного языка на другой.

Очевидно, что проблем использования сетевых сервисов слишком много, и разработанные языки, протоколы и стандарты (WSDL, SOAP, OWL...) не решают их в полной мере. Возможным решением могла бы стать единая, централизованная система, способная автоматизировать поиск, выбор и взаимодействие сервисов, обеспечив тем самым наилучшее качество сервиса. Для создания такой системы необходимо исследовать возможные варианты взаимодействия сетевых сервисов. Исследование можно проводить на абстрактной модели, которая отражала бы сущность взаимодействия.

Схематичное изображение модели сервиса в виде графа

Сервис можно изобразить в виде графа (рис. 1, а), где вершинам графа соответствуют операции, входящие в сервис (А, В и С), а ребра графа отображают зависимости между операциями: операция С является зависимой от операции В. Аналогичным образом могут быть показаны зависимости между операциями разных сетевых сервисов (рис. 1, б).

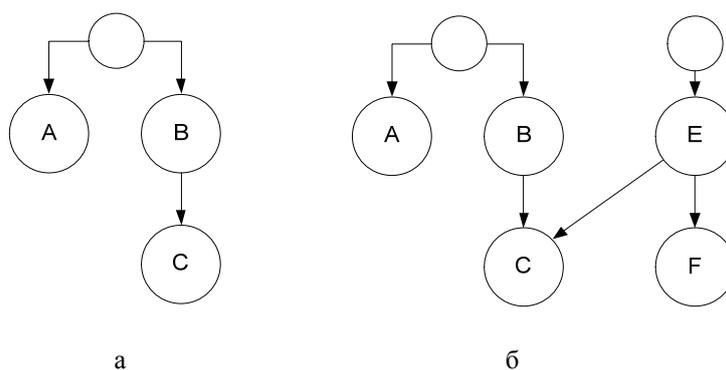


Рис. 1. Зависимости между операциями: одного сервиса (а), разных сервисов (б)

Использование графов для отображения схем сервисов позволяет не только наглядно отображать взаимозависимости сервисов, но и решать практические задачи. Рассмотрим пример с картографическим сетевым сервисом, который имеет две операции: операцию преобразования адреса в координаты (высоту и долготу) и операцию получения изображения местности, соответствующей заданным координатам (рис. 2, а). Предположим, что аналогичные сервисы предоставляют сразу три конкурирующие компании. Операции от разных провайдеров сервиса обладают различными параметрами QoS. Тогда, выполнив операцию объединения графов, отображающих схемы сервисов, и указав для каждой операции параметр качества, например, время выполнения операции, получаем граф, изображенный на рис. 2, б.

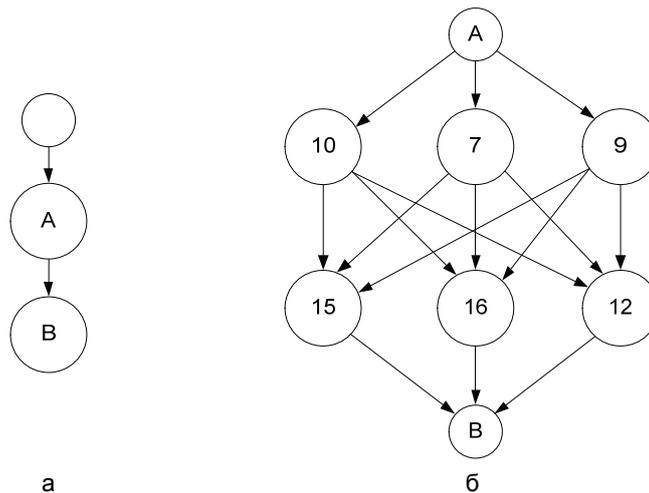


Рис. 2. Граф операций: одиночного сервиса (а), объединенного сервиса (б)

Решив задачу поиска наикратчайшего пути из точки А в точку В, можно ответить на вопрос, операциями каких сервисов нужно воспользоваться, чтобы выполнить необходимую задачу за минимальное время. Стоит заметить, что в качестве значений мог быть выбран абсолютно любой параметр – надежность, репутация или цена использования сервиса. По сути, получаем готовое решение для задачи одно-критериальной оптимизации. Более того, такой подход позволяет комбинировать отдельные операции сервисов и создавать из них новые сервисы с лучшими показателями качества обслуживания.

Математическое описание модели сервиса

Математическое описание модели сервиса необходимо для применения методов оптимизации поиска и нахождения оптимального плана вызова сервисов. Для описания сервиса использованы такие понятия, как схема сервиса, связи сервиса, запрос сервиса, сервисный путь, граф операций и граф набора операций [5]. Схема сервиса отображает ключевые возможности всех сетевых сервисов предметной области. Она предоставляет словарь и позволяет создать язык сервисных запросов. Схема сервиса определяется записью $S = (SG_1, SG_2, \dots, SG_n, D)$, где $SG_i = (V_i, E_i, \varepsilon_i), i = 1, \dots, n$, является ориентированным ациклическим графом (сервисным графом), где:

- $V_i = \{op_{ij} \mid 1 \leq j \leq m\}$ – набор операций сетевого сервиса;
- ε_i – корневой элемент графа (точка входа, через которую можно получить доступ ко всем операциям сервиса);
- $E_i = \{e_{ij} \mid 1 \leq j \leq l\}$ отражает зависимости между двумя операциями из одного графа;
- $D = \{D_{i,j} \mid 1 \leq i \leq n \wedge 1 \leq j \leq n \wedge i \neq j\}$ отражает зависимости между двумя некорневыми операциями из различных графов;
- $SG' = SG_i \circ SG_j$ является конкатенацией двух сервисных графов, полученной объединением корневых элементов SG_i и SG_j .

Зависимости между двумя внутренними операциями относятся к внутрисервисным зависимостям, и могут быть удовлетворены вызовом двух операций в определенном порядке. Зависимости между сервисами могут быть удовлетворены, когда осуществляется доступ к множеству сервисов. Предполагается отсутствие зависимости между корневыми элементами графов. Также предполагается, что все множественные зависимости к одной операции должны быть удовлетворены до выполнения этой операции. Например, есть две зависимости к операции op_k : op_i и op_j . В этом случае операции op_i и op_j должны быть вызваны до операции op_k . Также следует отметить, что в случае присутствия в схеме сервиса единственного графа, т.е. $n = 1$, S становится сервисной схемой с одиночным графом.

Сервисный путь для сервисного графа $SG = (V, E, \varepsilon)$ определяется как $P_i = (\{op_{i1}, \dots, op_{ij}, \dots, op_{ik}\}, E', \varepsilon)$, где ε – это корень SG , $E' \subseteq E$, и $k \geq 1$; $op_{ij} \in V$ для $1 \leq j \leq k$, и для каждого op_{ij} , $0 \leq j \leq (k-1)$, $\exists e_j \in E' : e_j = (op_{ij}, op_{i(j+1)})$.

Граф операции $G(op)$ для сервисного графа $SG_i = (V_i, E_i, \varepsilon_i)$ – это объединение всех сервисных путей в SG , которые ведут к операции op , $G(op) = \bigcup P_i$, где $P_i = (\{op_{i1}, \dots, op_{ij}, \dots, op_{ik}\}, E', \varepsilon)$. Граф опера-

ций $G(op)$ является направленным подграфом сервисного графа SG . На рис. 3, б, показан граф операции $G(d)$, который получен из SG (рис. 3, а) путем объединения двух сервисных путей P_1 и P_2 , которые ведут к сервисной операции d .

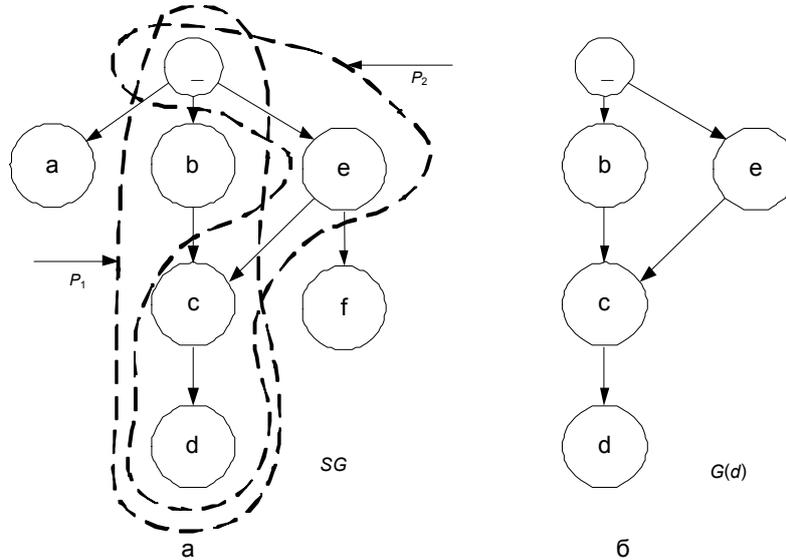


Рис. 3. Граф: сервиса SG (а), операции d (б)

Граф набора операций для сервисного графа $SG_i = (V_i, E_i, \varepsilon_i)$ определяется как $G(op) = \bigcup_{i=1}^k G(op_i)$, где $op = \{op_{ij} \mid 1 \leq i \leq k\}$. $G(op)$ является направленным подграфом сервисного графа SG . Например, на рис. 3 граф для набора операций $\{a, d, f\}$ – это сам граф SG , т.е. $G(\{a, d, f\}) = SG$.

Модель QoWS формально определяет набор качественных параметров для сетевых сервисов, которые указаны в таблице – время реакции, надежность, доступность, цена и репутация. Время реакции – время между моментом вызова op и моментом получения результата. Надежность – это возможность вызова op с гарантированным временем ответа. Доступность – это вероятность того, что операция доступна. Цена определяет стоимость вызова op . Репутация – это мера достоверности операции. Она определяется соотношением того, что операция предоставила и тем, что было обещано. Значения параметров в QoWS модели могут быть как целочисленными, так и дробными.

Параметр	Определение	Тип	Индекс
Время реакции	Время обработки операции + время возвращения результатов	число	1
Надежность	$N_{удачных}(op)/N_{всего}(op)$ – отношение удачных вызовов к общему количеству вызовов op	число	2
Доступность	$UpTime(op)/TotalTime(op)$, где $UpTime$ – время, когда op была доступна; $TotalTime$ – общее время	число	3
Цена	Денежная сумма, необходимая для вызова операции	число	4
Репутация	$\sum_{u=1}^k \text{Оценка}(op)/n$ $1 \leq \text{репутация} \leq 10$, где Оценка – оценка, поставленная пользователем; n – количество оценок	число	5

Таблица. Параметры QoWS

Сервисная связь определяется как набор экземпляров сервисов $I = (\{sid, op_1, \dots, op_n\})$, где sid – уникальный идентификатор сервиса; op – операция сервиса, определенная как пара $op = (opid, \lambda(op))$, где $opid$ является идентификатором операции, λ – меточной функцией, которая присваивает каждой сервисной операции набор значений параметров QoWS, обозначенный $Q = \sum_{i=1}^k Q_i$. Функция определяет i -й параметр op , где i – индекс параметра согласно таблице. Каждый экземпляр сервиса I в сервисной связи согласован с сервисным графом SG , т.е. операции в I определены в SG и следуют зависимостям, определенным в SG .

Сервисный запрос – это выражение сервисных вычислений в форме $\{s | \psi(s)\}$, где s – это переменная, представляющая запись из сервисного отношения SR , а ψ – это формула сервисных вычислений. Формулы составляются рекурсивно и имеют один из следующих форматов:

- $x\theta_q y$, где θ_q – оператор сравнения параметров QoS (один из $\{=, >, \geq, <, \leq, \neq\}$); x и y могут быть параметрами QoS или константами;
- $s.op == t.op$, где $s.op$ означает операцию из записи s , а $t.op$ – операцию из записи t ; $s.op == t.op$ принимает истинное значение, если t и s имеют одинаковый набор сервисных операций, и ложное в противном случае;
- $SR(s)$, где SR – сервисное отношение; s – переменная, содержащая сервисную запись; отношение $SR(s)$ принимает истинное значение, если s принадлежит SR , и ложное в противном случае.

Заключение

Практическое применение представленной выше модели и ее математического описания можно продемонстрировать на примере использования сервиса-реестра недвижимости. Сервисный запрос на получение истории покупаемой недвижимости при стоимости выполнения запроса не более 200 рублей может быть составлен следующим образом:

$$Q_1 = \{s.sid, s[G(history_report)] | RE(s) \wedge fee(s.history_report) \leq 200\},$$

где $G(history_report)$ – граф операций получения $history_report$; $fee(s.history_report)$ указывает на стоимость вызова операции; $RE (RealEstate)$ – сервис-реестр недвижимости.

Такая модель охватывает функциональность, поведение и качественные параметры сетевых сервисов, что позволяет осуществлять запросы с указанием критериев необходимого качества обслуживания. В дальнейшем предполагается расширение модели за счет использования математических методов оптимизации и алгоритмов создания плана вызова сервисов с учетом требований пользователя.

Литература

1. Eyhab Al-Masri, Qusay H. Mahmoud. Investigating Web Services on the World Wide Web // WWW 2008. – P. 795–804.
2. Anbazhagan Mani, Arun Nagarajan. Understanding quality of service for Web services // Материалы компании IBM [Электронный ресурс]. – Режим доступа: <http://www.ibm.com/developerworks/library/ws-quality.html>, свободный.
3. Клименков С.В., Максимов А.Н., Харитонов А.Е. Семантический анализ проектной документации // Научно-технический вестник СПбГУ ИТМО. – 2008. – № 46. – С. 198–202.
4. Бессмертный И.А. Теоретико-множественный подход к логическому выводу в базах знаний // Научно-технический вестник СПбГУ ИТМО. – 2010. – № 66. – С. 43–48.
5. Qi Yu, Athman Bouguettaya. Foundations for Efficient Web Service Selection // Springer – 2009. – 160 с.

Дергачев Андрей Михайлович – Санкт-Петербургский государственный университет информационных технологий, механики и оптики, старший преподаватель, dsm600@gmail.com