

# ПОДХОД К ОРГАНИЗАЦИИ ВЗАИМОДЕЙСТВИЯ ВЕБ-СЕРВИСОВ НА ОСНОВЕ МОДЕЛИ ПОТОКА РАБОТ

А. А. БАБОШИН, А. М. КАШЕВНИК

Санкт-Петербургский институт информатики и автоматизации РАН

СПИИРАН, 14-я линия ВО, д. 39, Санкт-Петербург, 199178

<alexey@iias.spb.su>, <andrey.baboshin@gmail.com>

---

УДК 004.8

Бабошин А. А., Кашевник А. М. **Подход к организации взаимодействия веб-сервисов на основе модели потока работ** // Труды СПИИРАН. Вып. 5. — СПб.: Наука, 2007.

**Аннотация.** В данной статье описывается подход к организации параллельного взаимодействия веб-сервисов. Предложен алгоритм, позволяющий осуществить параллельную обработку сценариев взаимодействующих веб-сервисов. Алгоритм предусматривает работу с альтернативными веб-сервисами, которые могут отличаться друг от друга стоимостью, временем обработки задания и т.п. Для выбора альтернативного веб-сервиса используется весовая функция. Подход предусматривает использование графового формализма и модели потока работ. — Библ. 6 назв.

UDC 004.8

Baboshin A. A., Kashevnik A. M. **Workflow-based approach of organization of collaboration of web-services** // SPIIRAS Proceedings. Issue 5. — SPb.: Nauka, 2007.

**Abstract.** In this article an approach to organization of parallel collaboration of web-services is described. An algorithm for processing parallel scenarios of interacting web-services is proposed. The algorithm deals with alternative web-services which may have different cost, task processing time, etc. A weight-function is used for choosing an alternative web-service. The approach assumes usage of the graph-formalism and workflow-model. — Bibl. 6 items.

---

## 1. Введение

В последнее время наблюдается значительное увеличение числа чрезвычайных ситуаций, включая как природные, так и техногенные. Одной из наиболее сложных задач при организации спасательных операций является доставка необходимых ресурсов в места, где они требуются, в соответствующее время, в нужном количестве. Таким образом, сформировалось новое направление исследований «гуманитарная логистика», занимающаяся изучением процессов и систем, вовлекаемых в мобилизацию материальных ресурсов, а также людей с их навыками, умениями и знаниями с целью оказания помощи пострадавшим при природных катастрофах и техногенных авариях [1].

В лаборатории интегрированных систем автоматизации СПИИРАН была разработана распределённая система интеллектуальной поддержки принятия решений в спасательных операциях, основанная на веб-сервисах [2].

Для эффективного управления спасательными операциями необходима быстрая реакция системы на чрезвычайную ситуацию. По этой причине, для ускорения принятия решения предлагается применять распараллеливание взаимодействия веб-сервисов, так как общее время реакции системы будет определяться временем выполнения самой длительной связанной цепочки веб-сервисов.

Предложенный в этой статье подход позволяет осуществить параллельную обработку сценариев взаимодействующих веб-сервисов. Для повышения отказо-

устойчивости системы в целом вводятся: (а) альтернативные веб-сервисы, (б) ограничение по времени выполнения веб-сервиса и (в) значение по умолчанию для каждой группы альтернативных веб-сервисов, используемое вместо выходного значения при невозможности выполнения ни одного из этих веб-сервисов.

## 2. Модель «поток работ» («workflow») для организации взаимодействия параллельно выполняющихся процессов

Модель «поток работ» («*workflow*») описывает типовые шаблоны поведения относительно взаимодействия процессов: передача управления (*control-flow*), обмен информацией (*data-patterns*), общий доступ к ресурсам (*resources patterns*), обработка исключений (*exception handling*).

Данный подход является совместной разработкой Технологического Университета г. Эйдховена и Технологического университета Квинсленда. Исследования начаты в 1999 году. Цель исследований — предоставление концептуальной базы для управления процессами.

Группа шаблонов *Control-flow* содержит шаблоны для передачи управляющих сигналов между работами. Шаблоны делятся на (а) базовые, (б) расширенные шаблоны разветвления и синхронизации, (в) обеспечения множественного параллельного выполнения задачи, (г) отслеживания состояния, (д) отмены и принудительного выполнения действий, (е) итерационные, (ж) завершение работы, (з) шаблоны вызова (*triggered patterns*). Среди них стоит выделить в первую очередь: «последовательность» (*sequence*), «параллельное расщепление» (*parallel split*), «синхронизация» (*synchronization/rendezvous/AND-Join*), «эксклюзивный выбор» (*XOR-split*), «простое соединение» (*simple merge*). Также присутствуют расширенные версии шаблонов для синхронизации и разветвления (*advanced branching and synchronisation*), множественного воплощения (когда несколько потоков выполняют одинаковые действия) (*multiple instance*), шаблоны, базирующиеся на состоянии (*state-based*), отмена и принудительное завершение (*cancellation and force completion*), итерационные (*iteration*), завершение работы (*termination*), сигнализация (*trigger*). [3].

Рассмотрим некоторые шаблоны более подробно.

«*Последовательность*». Два узла соединены переходом. После того, как исполнитель выполнил действие первого узла, управление переходит ко второму. Иллюстрация работы шаблона приведена на рис. 1.



Рис. 1. Шаблон «Последовательность».

«*Параллельное расщепление*». После выполнения узла происходит одновременный переход к двум и более последующим узлам. Причем после того, как управление передано узлам, поток управления распадается на несколько параллельных потоков. Для каждого перехода создается свой поток управления.

Данный шаблон часто используется в связке с шаблоном «синхронизация». Иллюстрация его работы приведена на рис. 2.

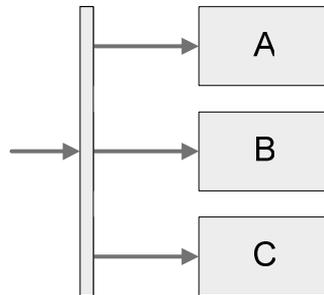


Рис. 2. Шаблон «параллельное расщепление».

«Синхронизация». В узле происходит ожидание выполнения предыдущих узлов и, после завершения ожидания, будет осуществлена передача управления далее. На выходе будет задействован только один поток управления. Иллюстрация его работы приведена на рис. 3.

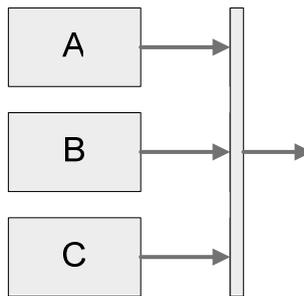


Рис. 3. Шаблон «Синхронизация».

«Эксклюзивный выбор». Узел в графе WF-процесса, в который приходит только один и из которого исходит два или более переходов. Причем, после того, как управление перешло к данному узлу, в нем делается выбор, по какому из исходящих переходов оно будет передано далее.

Шаблон часто используется совместно с шаблоном «простое соединение». Его работа проиллюстрирована на рис. 4.

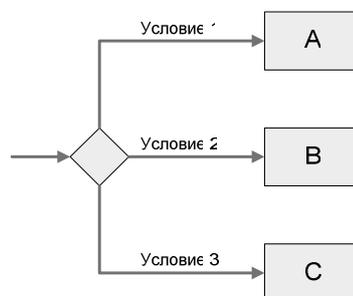


Рис. 4. Шаблон «Эксклюзивный выбор».

«Простое слияние». Узел, в котором соединяются два или более перехода, а выходит только один. После того, как в узел пришло управление от любого из входящих потоков, поток управления передается на единственный исходящий переход.

Предполагается, что управление может прийти в узел только по одному из входящих переходов. Иллюстрация работы шаблона приведена на рис. 5.

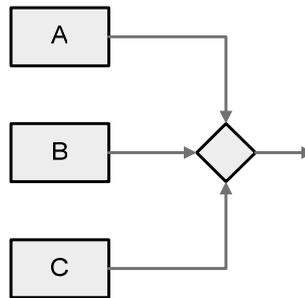


Рис. 5. Шаблон «Простое слияние».

Группа шаблонов *Data-patterns* содержит шаблоны для совместного доступа к данным следующего вида: (а) шаблоны области видимости данных, (б) взаимодействия данных, (в) передачи данных, (г) управления по данным. Среди них стоит выделить в первую очередь следующие: создания (*creation*) ресурсов, проталкивание (*push*), вытягивание (*pull*), откат использования (*detour*), автостарт, вызов по требованию (*auto-start*), разграничение доступа (*visibility*), множественный доступ (*multiple resource*) [4].

Группа шаблонов *Resource-patterns* содержит шаблоны для совместного доступа к ресурсам (а) создание ресурсов, (б) проталкивание ресурсов, (в) вытягивание ресурсов, (г) откат состояния, вызов по требованию, разграничение доступа, множественный доступ [5].

Группа шаблонов *Exception-handling* содержит шаблоны для обработки разного рода внештатных ситуаций и способы их обработки (а) ошибка выполнения работы (*work item failure*), (б) превышение максимального времени выполнения (*deadline*), (в) недоступность ресурса (*resource unavailability*), (г) внешнее прерывание (*external trigger*), (д) нарушение ограничений (*constraint violation*). Рассматриваются различные стратегии реагирования на ошибки (а) откат, (б) отсутствие действия, (в) компенсации [6].

### 3. Алгоритм организации взаимодействия веб-сервисов

Для описания сценария взаимодействия веб-сервисов в системе используются элементы теории графов. Вершины графа ставятся в соответствии веб-сервисам, которые необходимо выполнить, а дуги — входным и выходным атрибутам. Вершина графа может быть выполнена в случае, если выполнены все те вершины, из которых приходят входящие дуги.

Также, вводится фиктивная вершина «Конец» — для ситуации, когда для нескольких вершин не существует других вершин, в которые бы входили выходные дуги из этих вершин.

В качестве текущей вершины будем понимать вершину графа, которая обрабатывается в данный момент времени в данной параллельной ветви графа.

Предлагается использовать следующий алгоритм для организации взаимодействия параллельно выполняющихся процессов (веб-сервисов), берущий за основу модель потока работ (*workflow*):

1. в качестве текущей выбирается вершина «Конец», выполнение которой фактически является решением задачи;
2. одновременно запрашивается результат от всех вершин графа, дуги из которых являются входными для текущей вершины, а текущий веб-сервис переходит в состояние ожидания входных значений (шаблон *synchronization*). После получения всех входных значений происходит переход на шаг 4. Если же входных дуг нет, то переход на шаг 4 происходит сразу;
3. каждый из одновременно запрашиваемых веб-сервисов становится текущим и выполняется переход на шаг 2;
4. выполняется текущий веб-сервис.

Условием окончанием работы алгоритма является выполнение вершины «Конец».

Для обеспечения устойчивости к возможным отказам в работе веб-сервисов делается следующее:

1. Вводится ограничение на время выполнения веб-сервиса (*deadline*). Это делается по той причине, что может быть ограничено максимально допустимое время выполнения конкретного веб-сервиса, при превышении которого дальнейшее ожидание завершения его работы не представляется возможным.
2. На случай недоступности некоторого веб-сервиса вводятся альтернативные веб-сервисы, которые будут выполняться в случае невозможности его выполнения.
3. Вводятся значения по умолчанию для веб-сервисов. В том случае, когда нет возможности ждать завершения веб-сервиса, либо ни один из альтернативных веб-сервисов недоступен, но продолжить выполнение цепочки веб-сервисов необходимо, используется значение по умолчанию.

Для осуществления выбора альтернативных веб-сервисов вводится весовая функция  $F$ , определяющая приоритет выбора:

$$F = C_1 O_t + C_2 T_{cp} + C_3 P$$

Где  $O_t$  — количество отказов, отнесенное к количеству вызовов,  $T_{cp}$  — среднее время доступа, отнесённое к максимальному из всех альтернативных веб-сервисов, вычисляется динамически,  $P$  — стоимость доступа, заданная экспертом, отнесённая к максимальной стоимости доступа из всех альтернативных веб-сервисов, а  $C_i$  — весовые коэффициенты, с помощью которых может задаваться важность каждого параметра для данного веб-сервиса в данный момент времени (с учётом контекста); причём  $\sum_{i=1}^3 C_i = 1$ .

На рис. 6 проиллюстрирован механизм выбора альтернативного веб-сервиса и возврат значения по-умолчанию, в случае невозможности выполнения веб-сервиса.

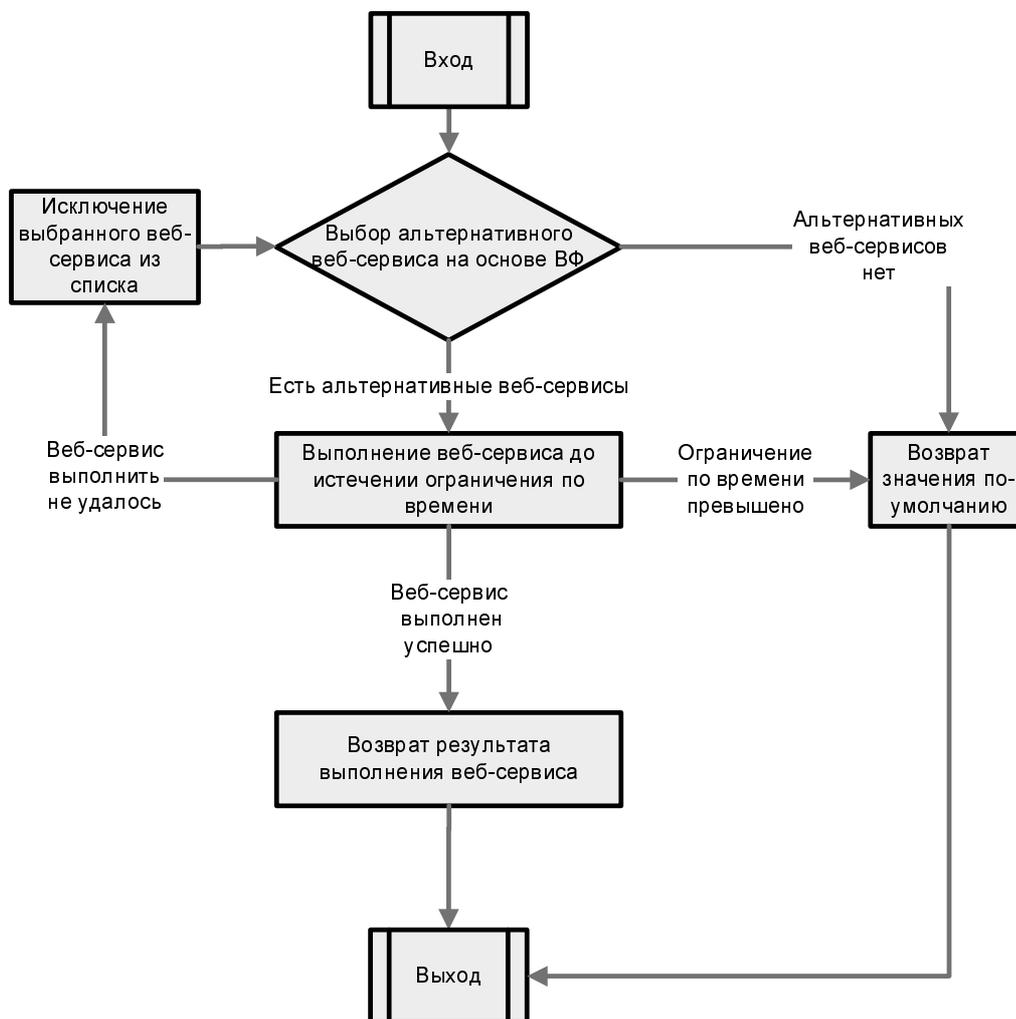


Рис. 6. Выбор альтернативного веб-сервиса.

На рис. 7 показано взаимодействие веб-сервисов в системе интеллектуальной поддержки принятия решений в спасательных операциях, в которых задействованы вертолёты и наземные средства доставки участников операции. Данный граф формируется на основе контекста текущей задачи, который генерируется в системе поддержки принятия решений при поступлении запроса пользователя.

В качестве текущей выбираем вершину «Конец». Осуществляем одновременный переход к выполнению вершин «Выбор пожарных бригад» и «Выбор медицинских бригад». Для вершины «Выбор пожарных бригад» одновременно запрашивается результат выполнения вершин «Расчет кратчайших путей» и «Расчет числа требуемых пожарных бригад». Вершина «Расчет числа требуемых пожарных бригад» может выполняться, так как нет вершин, от результата выполнения которых она зависит. Результат её выполнения будет получен вершиной «Выбор пожарных бригад». Аналогично для вершины «Выбор пожарных бригад» получают результаты от остальных вершин. После чего она выполняется, и результат ее выполне-

ния будет доступен вершине «Конец». Аналогично для вершины «Выбор медицинских бригад».

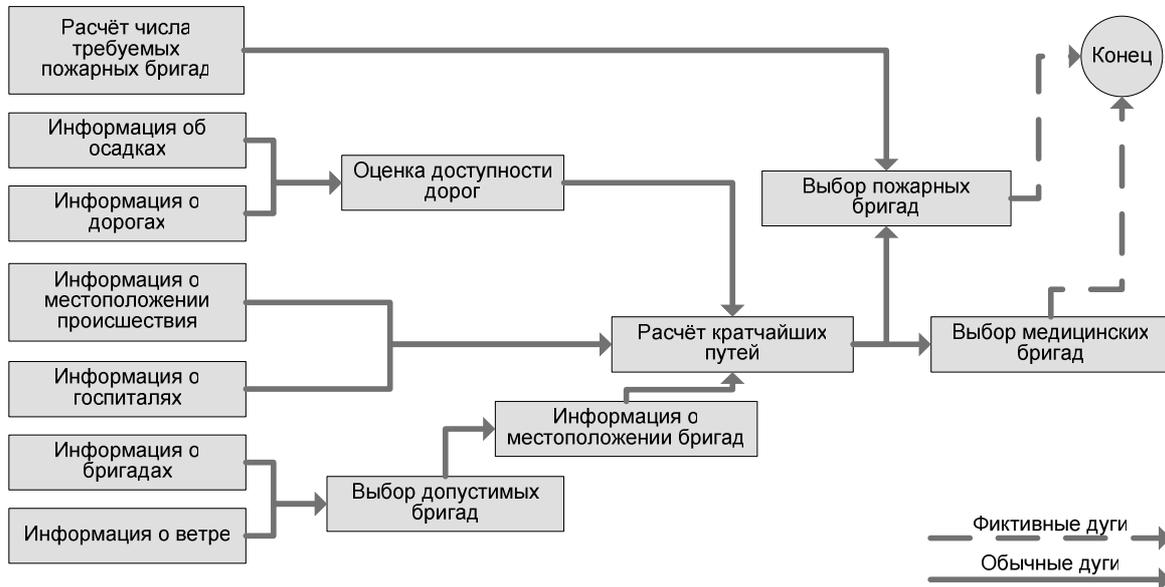


Рис. 7. Сценарий взаимодействия веб-сервисов в системе интеллектуальной поддержки принятия решений.

## 4. Заключение

Предложенный в статье алгоритм, разработанный на основе модели потока работ, позволяет эффективно организовывать взаимодействие веб-сервисов. Эффективность достигается за счет параллельной обработки взаимодействующих веб-сервисов, возможности продолжить работу при отказе одного или нескольких веб-сервисов или их недоступности.

Для обработки ситуации отказа в работе или недоступности веб-сервисов были введены альтернативные веб-сервисы, возможность задания значения по умолчанию для искомого параметра и весовые функции для выбора одного из альтернативных веб-сервисов.

## 5. Благодарности

Исследование было частично выполнено в рамках проектов РФФИ # 05-01-00151 и # 06-07-89242, проекта # 16.2.35 программы «Математическое моделирование и интеллектуальные системы» Президиума РАН, проекта # О-1.9 программы «Фундаментальные основы информационных технологий и компьютерных систем» ОИТВС РАН.

## Литература

1. Humanitarian Logistics: Getting the Right Relief to the Right People at the Right Time // Fact Sheets, Fritz Institute, 2005.
2. Смирнов А. В., Шилов Н. Г., Кашевник А. М. Интеллектуальная поддержка принятия решений в спасательных операциях // Труды XIV Международной конференции «Проблемы управления безопасностью сложных систем». Российский государственный гуманитарный университет, 2007. С. 380–383.
3. Aalst W. M. P. van der, Hofstede A. H. M. ter, Kiepuszewski B., and Barros A. P. . Workflow Patterns // Distributed and Parallel Databases. 2003. Vol. 14(3), P. 5–51.
4. Russell N., Hofstedel A. H. M. ter, Edmond D., Aalst W. M. P. van der. Workflow Data Patterns // QUT Technical report, FIT-TR-2004-01. Brisbane: Queensland University of Technology, 2004.
5. Russell N., Hofstedel A. H. M. ter, Edmond D., Aalst W. M. P. van der. Workflow Resource Patterns // BETA Working Paper Series, WP 127, Eindhoven: Eindhoven University of Technology, 2004.
6. Russell N., Aalst W. M. P. van der, and Hofstedel A. H. M. ter. Exception Handling Patterns in Process-Aware Information Systems // BPM Center Report BPM-06-04, 2006.