

АРХИТЕКТУРА ИНТЕЛЛЕКТУАЛЬНОЙ ТРАНСПОРТНОЙ СИСТЕМЫ НА ОСНОВЕ ТЕХНОЛОГИЙ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ

© 2015 О.Н. Сапрыкин¹, О.В. Сапрыкина², А.А. Федосеев¹

¹ Самарский государственный аэрокосмический университет имени академика С.П. Королева
(национальный исследовательский университет)

² ООО «NetCracker», г. Самара

Поступила в редакцию 30.07.2015

В статье рассматриваются принципы архитектурного проектирования распределенных интеллектуальных транспортных систем (ИТС) посредством интеграции узкоспециализированных приложений. Предлагается интеграционный подход на основе паттерна «обмен сообщениями» примененного к службам, расположенным в виртуальных облачных средах. Рассмотрено применение принципа многовариантной персистентности к хранению разнотипных данных ИТС.

Ключевые слова: интеллектуальная транспортная система, облачные вычисления, многовариантная персистентность.

ВВЕДЕНИЕ

Современные технологии организации вычислительных процессов сложных систем базируются на хорошо изученных и апробированных методиках и паттернах проектирования, внося незначительные модификации в них. Однако даже незначительное изменение, сделанное в верном направлении, приводит к увеличению эффективности всей системы. Одним из примеров подобных изменений может служить переход к технологиям облачных вычислений. Наибольшую эффективность применения этого подхода можно достигнуть в сложных распределенных гетерогенных системах, примером которых могут служить интеллектуальные транспортные системы (ИТС).

Интеллектуальные транспортные системы широко распространены в настоящее время и используются в большинстве крупных городов мира. Под данным термином понимается широкий спектр аппаратно-программных средств, позволяющих собирать информацию с объектов транспортной инфраструктуры, проводить анализ и принимать решения (оперативные или стратегические) [1]. ИТС представляет собой гибридную систему, состоящую из множества разнородных и взаимодействующих между собой систем, объединенных для достижения единой цели. Системы выполняют функции мониторин-

га, сбора данных, управления, классификации, прогнозирования, поддержки процессов принятия решений [2]. В разных реализациях ИТС используются различные функциональные подмножества. Однако большинство разработчиков ИТС сталкивается со следующими задачами:

- территориальная распределенность технических средств мониторинга и управления дорожным движением;
- разнородность программных средств, используемых для решения узкоспециализированных задач;
- необходимость обработки больших массивов разнотипных неструктурированных данных.

Для их решения в статье предлагается подход организации архитектуры ИТС, основанный на виртуализации разнородных программных модулей, размещении их в облачной инфраструктуре и организации интеграционного пространства с помощью паттерна «обмен сообщениями». Задачу обработки разнотипных данных предлагается решить за счет применения подхода многовариантной персистентности. Данные решения позволят спроектировать стабильную, масштабируемую и интероперабельную систему, удовлетворяющую новейшим требованиям к программному обеспечению.

СИНЕРГЕТИЧЕСКИЙ ЭФФЕКТ В АРХИТЕКТУРЕ ИТС

Проектирование интеллектуальной транспортной системы является сложной задачей, поскольку оно связано с необходимостью сочетания множества разнотипных компонентов. Все элементы архитектуры взаимодействуют между собой как в пределах содержащих их компонентов, так и между компонентами. Для оптимального распределения элементов архи-

Сапрыкин Олег Николаевич, кандидат технических наук, доцент кафедры организации и управления перевозками на транспорте. E-mail: saprykinon@gmail.com

Сапрыкина Ольга Валерьевна, кандидат технических наук, инженер-программист ООО «NetCracker».

E-mail: olga_grieva@mail.ru

Федосеев Александр Андреевич, аспирант кафедры информационных систем и технологий.

E-mail: fedoseevale@gmail.com

тектуры между компонентами необходимо определить скрытые фундаментальные особенности взаимодействия объектов системы и свойства их формирующие. В качестве методологической основы проектирования выбран синергетический подход, который базируется на максимальном использовании имманентных свойств объектов в процессе синтеза сложных систем. В синергетической системе в процессе ее целенаправленной самоорганизации происходит последовательное уменьшение степеней свободы и их подстройка к макропеременным [3].

При проектировании ИТС синергетический эффект достигается благодаря использованию в качестве управляющих воздействий архитектурной организации паттернов проектирования *GRASP (General Responsibility Assignment Software Patterns)*, общих паттернов распределения ответственности в программном обеспечении) «Информационный эксперт», «Низкая связность» и «Высокое зацепление». Паттерн «Информационный эксперт» обеспечивает адекватное распределение обязанностей между компонентами, возлагая выполнение той или иной функциональности на элемент, обладающий необходимой информацией для ее выполнения [4]. Паттерн «Высокое зацепление» обеспечивает высокую степень сфокусированности обязанностей элемента. Паттерн «Слабое связывание» поддерживает функциональную и информационную независимость элементов друг от друга.

Инфраструктурный модуль ядра ИТС поддерживает данную концепцию на физическом уровне посредством использования паттернов «Внедрение зависимости» (*Dependency Injection*) и «Локатор служб» (*Service Locator*). Паттерн «Внедрение зависимости» обеспечивает независимость между компонентами, каждый из которых решает узкоспециализированную задачу. Паттерн «Локатор служб» дает унифицированный способ взаимодействия между слабосвязанными компонентами [5]. Данный факт позволяет использовать ядро ИТС в качестве интеграционной среды, реализовывая дополнительные компоненты в виде подключаемых к ИТС модулей, что так же дает возможность опираться на базовую функциональность ядра ИТС при решении узкоспециализированных задач.

Если позволить модулям ИТС эволюционировать, придав им в качестве управляющих воздействий перечисленные выше паттерны, то взаимоотношение между элементами ИТС выходит на качественно новый уровень, что ведет к проявлению синергетического эффекта. Разбиение на модули основывается на смысле содержащихся в нем агрегатов и классов. Модули, с одной стороны, позволяют просмотреть общую картину архитектуры, не вдаваясь в детали их внутреннего устройства. С другой стороны, каждый отдельный

модуль показывает некоторую часть проекта, по минимуму ссылаясь на остальную его часть.

После того как у модулей ИТС появляется возможность отражать понимание предмета, агрегаты и классы внутри них могут более гибко эволюционировать. Находясь в одном модуле, элементы могут указать на понятия неявно присутствующие в предметной области, но отсутствующие в модели. Добавление этих понятий через несколько классов или взаимосвязей приведет к повышению качества модели предметной области, при этом универсальность модели возрастает, а сложность сокращается [6].

СЕТЬ ИНТЕГРИРОВАННЫХ ОБЛАЧНЫХ СЛУЖБ

После того как определены границы каждого модуля системы необходимо определить технологии их хостинга и взаимодействия друг с другом. В качестве среды выполнения систем целесообразно выбрать облачные службы, в которых информация хранится и обрабатывается на сетевых масштабируемых виртуальных ресурсах, доступ к которым осуществляется по запросу. Согласно парадигме облачных вычислений все используемые ресурсы представляются в виде служб: аппаратное обеспечение (*Hardware as a Service, HaaS*), программное обеспечение (*Software as a Service, SaaS*), платформа (*Platform as a Service, PaaS*) и база данных (*Data as a Service, Daas*).

В последнее время наблюдается рост интеграции ГИС в глобальное информационное пространство Интернет. Повышение эффективности данного процесса достигается путем взаимодействия с сервисами в глобальной сети. Например, сервисами полезными для большинства систем ИТС являются сервисы электронных карт и сервисы спутниковых снимков. Между тем, организовав собственную систему как веб-сервис, можно предоставить к ней доступ третьих систем (рис. 1).

Структуру и механизмы межсистемного взаимодействия определяют паттерны интеграции

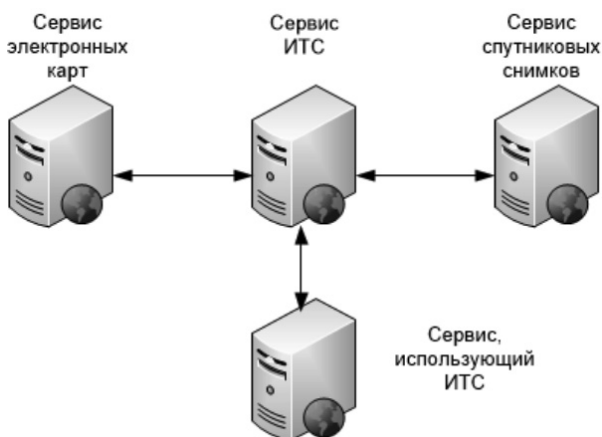


Рис. 1. Интеграция облачных служб

информационных систем. При проектировании применяется сервис-ориентированный подход. Сервис-ориентированное проектирование – это архитектурный подход, при котором интеграция слабосвязанных компонентов распределенных приложений обеспечивается посредством сообщений и контрактов [7]. В то же время интеграция с внешними системами построена с учетом архитектурных паттернов, обеспечивающих синергетический эффект – «Информационный эксперт», «Высокое зацепление» и «Слабое связывание». Логика интеграционного решения сосредоточена на определенном уровне системы, не содержит иной логики и по минимуму зависит как от остальной части системы, так и от приложений с которыми происходит интеграция.

В архитектуре разрабатываемой системы применяется паттерн проектирования «Обмен Сообщениями». В сервис-ориентированных приложениях фундаментальной единицей взаимодействия являются сообщения. Для реализации паттерна необходимо распределить между участниками процесса следующие роли: исходный отправитель сообщений, посредник и конечный приемник сообщений. Принимающей и отправляющей сторонам необходима следующая информация для установки контакта: адрес конечной точки, кодировка и контракт.

Совместимость осуществляется за счет поддержки системами таких стандартов как *HTTP*, *URL*, *XML* и *WS**. *WS** – это семейство стандартов, описывающих формат сообщений, их передачу, шифрование, транзакционность пересылки и адресацию конечных точек. При отсутствии поддержки данного стандарта используется паттерн «Адаптер» для преобразования одного формата сообщений в другой.

Для обеспечения слабой связности при проектировании доступа к внешним сервисам, уровень взаимодействия с ними организован как источник данных с открытым интерфейсом. Для уменьшения зависимости ИТС от качества соединения и доступности сервисов используется паттерн проектирования «Репозиторий» [5] с двумя уровнями кэша. Информация из внешних сервисов сохраняется в репозиторий для последующего анализа. Кэш первого уровня репозитория располагается в оперативной памяти, а кэш второго уровня в локальной базе данных. Репозиторий инкапсулирует механизмы запроса к внешним источникам данных и хранение ссылок на полученные объекты, таким образом, с клиентской стороны репозиторий виден как набор коллекций объектов, хранящихся в оперативной памяти.

Для доступа репозитория к системе обмена сообщениями используется паттерн проектирования «Шлюз обмена сообщениями» [5], инкапсулирующий вызовы методов, специфичных для

обмена сообщениями, и предоставляющий интерфейс, специфичный для предметной области. Таким образом, паттерн «Шлюз обмена сообщениями» устраняет прямые зависимости между приложением и системой обмена сообщениями.

«Шлюз обмена сообщениями» также используется уровнем представления для предоставления функций приложения третьим системам. При пересылке результата вычислений третьим системам могут возникнуть проблемы связи и результат потеряется. В связи с чем, к системе обмена сообщений предъявляются повышенные требования по надежности. Необходимый уровень надежности может обеспечить паттерн проектирования «Гарантированная доставка» [7]. Надежность пересылки обеспечивается путем сохранения очереди сообщений в локальном хранилище данных. Сообщение удаляется из очереди только после того, как от получающей стороны получено уведомление об успешной доставке сообщения.

Применение паттернов интеграции значительно повышает уровень интероперабельности ИТС, что расширяет границы ее применения. Различное конфигурирование и развертывание системы в облачной инфраструктуре позволяет кардинально изменить варианты ее использования и настроить на решение актуальных задач конкретного ИТС.

МНОГОВАРИАНТНАЯ ПЕРСИСТЕНТНОСТЬ ПРИ ДОСТУПЕ К ДАННЫМ

Составляющие интеллектуальную транспортную систему модули обладают разными требованиями к хранению данных, поэтому многие из них имеют собственный слой доступа к данным. В ИТС использованы следующие технологии хранения данных:

- файловые хранилища;
- реляционные СУБД;
- документальные СУБД.

Используемый подход носит название многовариантной персистентности [8] и позволяет хранить данные в форме близкой той, что используется в программном коде. Это исключает объектно-реляционное отображение в звеньях системы, где производительность является критическим фактором. Подход, основанный на многовариантной персистентности использовался на всех этапах проектирования базы данных: концептуальном, логическом и физическом. На рис. 2 изображена структура взаимодействия модулей данных.

Модули импорта данных имеют возможность читать данные из внешних источников, для каждого из них разработан специальный компонент, способный разобрать информацию в предоставленном формате.

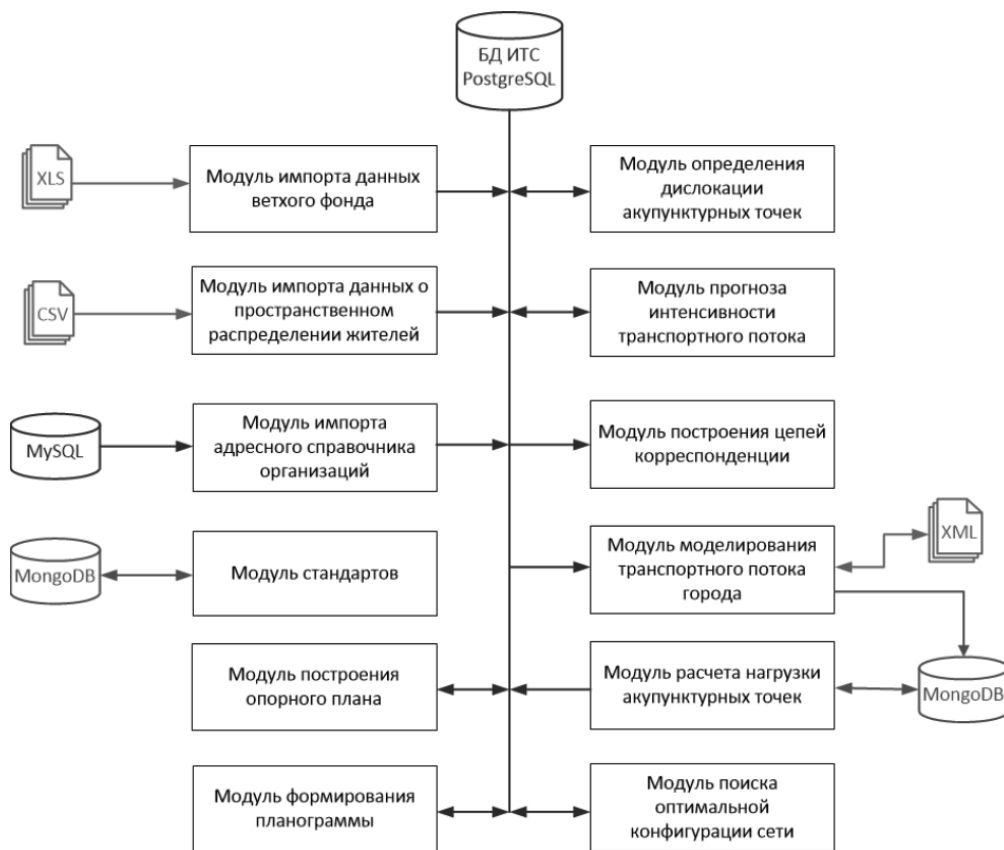


Рис. 2. Структура взаимодействия модулей данных

В частности, разработаны следующие компоненты:

- провайдер Excel-файлов, хранящий данные ветхого жилищного фонда;
- провайдер Csv-файлов неперсонализированной информации о пространственном распределении жителей города;
- провайдер к базе данных MySQL с адресным справочником организаций и предприятий города.

Каждый модуль импорта записывает преобразованную информацию в основную базу ИТС посредством провайдера, предоставленного ядром ИТС [6].

Модуль ведения стандартов в электронном виде хранит информацию в стандартизованном виде, и в то же время, имеет возможность гибко изменять структуру хранения. Данными возможностями обладает документальная СУБД MongoDB.

Модуль моделирования транспортного потока основан на библиотеке компонентов многоагентного моделирования транспорта MATSim. Библиотека имеет собственный формат хранения транспортной сети и агентов, который основан на формате XML. Перед запуском моделирования производится экспорт электронной карты, стандартов проектирования УДС, данных об интенсивности транспортных потоков, информации о плотности населения и адресного справочника

организаций в XML-формат MATSim. Результатом работы системы моделирования являются XML-файлы, содержащие пути и время перемещения агентов. Данные файлы импортируются в систему, после чего информация, хранящаяся в них, используется для визуализации и анализа [9].

В модуле моделирования транспортного потока каждый агент, перемещаясь по сети, формирует новые знания в виде информации о напряженности на участках улично-дорожной сети и окружающей инфраструктуре. Данные знания имеют сложную структуру, и в то же время, они должны быстро сохраняться в базе данных. Этого нельзя достичь при помощи отображения в реляционную базу данных. Поэтому в данном случае используется СУБД документального типа MongoDB. Это позволяет моментально сохранять снимок текущего состояния интеллектуального объекта в любой момент времени, не отвлекая его от основных обязанностей. Скорость достигается за счет быстрого отображения агрегатов и возможности широкого использования кластеров.

Для консолидации знаний используется процедура, основанная на паттерне «Отображение-свертка» (Map-reduce) [8]. Данная процедура позволяет структурировать полученную от объектов информацию и записать ее в реляционную базу данных. Реляционное представление данных позволяет использовать алгоритмы интеллектуального анализа для их обработки.

Результатом работы модуля оптимизации конфигурации УДС является оптимальная конфигурация транспортной сети урбанизированной территории [10]. В рамках программного комплекса предусмотрено параллельное хранение нескольких версий конфигураций улично-дорожной сети для последующего их сравнения.

ЗАКЛЮЧЕНИЕ

Применение описанных в статье подходов позволило создать стабильную, масштабируемую и интероперабельную интеллектуальную транспортную систему. Проект системы рассчитан на постоянное развитие, в ней упрощена процедура добавления новых модулей и источников данных. Интероперабельность системы обеспечивается использованием общепризнанных стандартов интеграции приложений. В результате применения предложенного подхода, исключено перепроектирование системы, вносящее существенные затраты при выполнении проекта.

Разработанная архитектура хранения данных интеллектуальной транспортной системы дает возможность унифицированного доступа к разнотипной информации. Данный подход формирует единую информационную платформу, на которой создается расширяемая система, способная решать аналитические задачи исследования транспортной инфраструктуры.

СПИСОК ЛИТЕРАТУРЫ

1. Сапрыкин О.Н. Паттерны проектирования в архитектуре ИТС / Математика. Компьютер. Образование // Тезисы докладов XV международной конф. М.-Ижевск: НИЦ «Регулярная и хаотическая динамика», 2008. С. 116.
2. Михеева Т.И., Сапрыкин О.Н. Паттерны проектирования подсистемы анализа состояния пространственно-координированных объектов интеллектуальной транспортной системы // Информационные технологии. 2010. №7. С. 64–70.
3. Построение архитектуры аналитического инструментария интеллектуальной транспортной системы на основе паттернов проектирования / Т.И. Михеева, О.Н. Сапрыкин, О.В. Сапрыкина // Вестник Самарского гос. техн. ун-та. Серия «Технические науки». 2010. №4 (27). С. 27–35.
4. Ларман К. Применение UML и шаблонов проектирования [пер. с англ.]. М.: Издательский дом «Вильямс», 2004. 624 с.
5. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон и др. СПб: Питер, 2001. 368 с.
6. Управление транспортной инфраструктурой / Т.И. Михеева, С.В. Михеев, О.Н. Сапрыкин // Самара: НИЦ ИТС, 2015. 173 с.
7. Фаулер М. Архитектура корпоративных программных приложений [пер. с англ.]. М.: Вильямс, 2006. 544 с.
8. Фаулер М., Садаладж П.Дж. NoSQL. Новая методология разработки нереляционных баз данных. М.: Вильямс, 2013. 192 с.
9. The Multi-Agent Transport Simulation MATSim / A. Horni, K. Nagel, K. W. Axhausen, et al. // Zurich and Berlin, 2015. 604 p.
10. Сапрыкина О.В., Сапрыкин О.Н. Адаптирующийся меметический алгоритм оптимизации конфигурации улично-дорожной сети [Электронный ресурс] // Современные проблемы науки и образования. 2014. № 5. URL: <http://www.science-education.ru/119-15256> (дата обращения 05.06.2015).

ARCHITECTURE OF THE INTELLIGENT TRANSPORT SYSTEM BASED ON CLOUD COMPUTING TECHNOLOGIES

© 2015 O.N. Saprykin¹, O.V. Saprykina², A.A. Fedoseev¹

¹ Samara State Aerospace University named after Academician S.P. Korolyov
(National Research University)

² NetCracker Technology, Samara

The article discusses the architectural design principles for distributed intelligent transport systems (ITS) by the integration of highly specialized applications. Integration approach based on the “messaging” design pattern that is applied to services that are located in virtual cloud environments is proposed. The application of the multivariate persistence principle for storing different types of ITS data is discussed.
Keywords: Intelligent Transport System, Cloud Computing, Multivariate Persistence.

Oleg Saprykin, Candidate of Technics, Associate Professor at the Transportation Organization and Management Department. E-mail: saprykinon@gmail.com

Olga Saprykina, Candidate of Technics, Software Engineer. E-mail: olga_grineva@mail.ru

Alexander Fedoseev, Graduate Student at the Department of Information Systems and Technology. E-mail: fedoseevale@gmail.com