

**Самойлов Дмитрий Станиславович**

E-mail: [duma@yandex.ru](mailto:duma@yandex.ru)

Студент гр. М-44.

**Beliacov Stanislav Leonidovich**

Taganrog Institute of Technology - Federal State-Owned Educational Establishment of Higher Vocational Education «Southern Federal University»

E-mail: [beliacov@yandex.ru](mailto:beliacov@yandex.ru)

44, Nekrasovskiy, Taganrog, 347928, Russia, phone 371-743

Professor, Dr. of Eng. Sc.

**Samoilov Dmytri Stanislavovich**

E-mail: [duma@yandex.ru](mailto:duma@yandex.ru)

Student of M-44.

УДК 528

**В.С. Василенко**

### **ПРИМЕНЕНИЕ КОНЦЕПЦИИ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ В ГИС**

*Рассмотрены особенности и ограничения применения концепции облачных вычислений в ГИС. Предложены методы обхода этих ограничений на основе применения высокоуровневых стратегий управления взаимодействием сервера и клиента.*

*Геоинформационная система; облачные вычисления; картография.*

**V.S. Vasilenko**

### **APPLICATION OF THE CLOUD COMPUTING CONCEPT IN GIS**

*Features and restrictions of application of the cloud computing concept in GIS are considered. Methods of detour of these restrictions on the basis of application high-level strategy of management by server and client interaction are offered.*

*Geoinformation system; cloud computing; cartography.*

В настоящее время широкое распространение получает концепция «облачных вычислений», суть которой состоит в том, что вычислительные ресурсы и среды предоставляются пользователям удаленно через сеть Internet. Это освобождает пользователей от необходимости обслуживать задействованное в вычислениях аппаратное и программное обеспечение («облако») и позволяет им сосредоточиться непосредственно на решении прикладных задач [1].

Наиболее известными проектами с применением облачных вычислений на сегодняшний день являются, прежде всего, набор Web-сервисов и систем разработки Web-приложений от Google известный как Google Web. Система Synaptic Hosting от AT&T, используя которую пользователь получает виртуальную среду с разными вариантами хранения данных, сетевых архитектур, с поддержкой приложений и обеспечением удаленных вычислительных мощностей для них. Система легко масштабируется как в плане новых сервисов, так и в плане нагрузок. Основанный на концепции «Everything as a Service» проект компании Hewlett-Packard, предоставляющей свои вычислительные мощности Министерству обороны США через набор виртуальных машин, работающих через один web-портал от Hewlett-Packard. Альянс Intel, Yahoo! и HP под названием Cloud Computing Test Bed представляющий собой тестовый полигон для облачных вычислений, который состоит из шести дата-центров, в каждом из которых находится «облако» из нескольких тысяч процессоров.

Применение концепции облачных вычислений в области геоинформационных технологий дает ряд очевидных преимуществ по сравнению со стандартным протоколом работы распределенных геоинформационных (ГИС) приложений. Это, прежде всего, независимость от платформы [2] – пользователи могут использовать любые удобные для работы мобильные устройства и операционные системы. Отсутствует необходимость в наличии большого количества прикладных приложений ориентированных на работу с определенными web-сервисами или предназначенными для решения конкретных задач, а также в их обновлении – все необходимые приложения клиент получает, своевременно подключившись к «облаку». Автоматическая поддержка масштабируемости «облака» представляет удобную среду для расширения ГИС системы.

Основным недостатком и ограничением применения концепции «облачных вычислений» в распределенных ГИС является узкополостность информационного канала, связывающего пользователя системы с ее сервисами, характерного для распределенных ГИС систем, особенно для мобильных ГИС. При этом стандартных алгоритмов сжатия информации, передаваемой по информационному каналу, уже недостаточно для обеспечения нормальной работы пользователя в системе в процессе решения большинства геоинформационных задач [3]. Для решения этой проблемы необходим ряд высокоуровневых стратегий, направленных на уменьшение количества обращений пользователя к сервисам геоинформационной системы.

В ряде случаев такой стратегией может служить строгая формулировка пользователем задачи на понимаемом системой метаязыке с последующей отправкой данной формулировки на обработку соответствующему web-сервису, и получении решения задачи в готовом виде сразу или постранично. Однако такая стратегия базируется на наличии ограниченного ряда высокоуровневых процедур, предназначенных для выполнения строго определенных операций над геоданными, которые в общем случае соответствуют операторам соответствующего метаязыка. Кроме того, такая стратегия требует исчерпывающего понимания пользователем алгоритма решения конкретной задачи с использованием данного инструментария (web-приложения и языка определения запросов). Примером применения данной стратегии может служить использование технологии OpenGIS, расширяющей возможности стандартных СУБД на операции с картографическими данными. Пока решение некоторой картографической (например, поисковой) задачи может быть представлено в виде SQL запроса и пользователь в состоянии сформулировать этот запрос диалог пользователя с системой может быть сведен к передаче SQL запроса web-сервису и получении ответа в виде выборки или модификации соответствующих данному запросу картографических объектов.

В большинстве случаев пользователь имеет лишь намерение решить некоторую задачу, представляемую им в виде конечных результатов, и имеет лишь частичное представление об инструменте ее решения. А ограниченного числа высокоуровневых процедур системы может быть недостаточно для четкого описания решения данной задачи. Что вынуждает пользователя искать решение стандартными средствами ГИС – манипулированием тематическими слоями, что в случае ограниченного информационного канала неприемлемо.

В этом случае целесообразно использовать другую стратегию по уменьшению количества обращений по информационному каналу – рассматривать алгоритм решения картографической задачи в виде последовательности элементарных операций, которые бы выполнил пользователь эксперт в процессе нахождения решения данной задачи. Тогда решение задачи пользователем не экспертом будет представлять собой накопление опыта (возможно путем проб и ошибок) по преобразованию или получению картографических данных с помощью имеющегося в

его распоряжении web-приложения с последующим применением данного опыта для получения алгоритма решения задачи приближенного к алгоритму, который бы получил эксперт. Стратегией ограничения числа обращений пользователя к web-сервису в данном случае будет исключение или ограничение продолжительности этапа получения решающим задачу пользователем опыта по решению данной задачи с использованием данного ему инструментария и предоставлением пользователю готовых блоков конечного алгоритма, к которому он стремится в процессе решения своей задачи.

Для реализации данной стратегии геоинформационная система должна предоставлять ряд достаточно низкоуровневых процедур для того, чтобы их последовательностью могли бы быть описаны решения максимально широкого круга задач. Картографические задачи должны быть типизированы и классифицированы, а их решения представлены в виде иерархического дерева, в котором основная задача разбита на ряд подзадач, выполняемых в определенной последовательности для получения ее решения. Каждая подзадача в свою очередь разбивается на ряд подзадач. Такое разбиение продолжается до представления каждой листовой подзадачи в виде ряда элементарных процедур, предоставляемых данной геоинформационной системой.

Таким образом, в процессе решения своей задачи пользователь может взять за основу похожую задачу из заранее подготовленных шаблонных задач. После этого он может выполнить детализацию и подстройку алгоритма решения взятой задачи под свою, путем замены или заимствования ветвей ее дерева решения – причем это возможно без загрузки картографических данных.

Очевидно, что даже при применении такой стратегии пользователь будет вынужден загрузить с сервера ветви дерева решения подобной задачи, разобраться в том какие ветви подходят для его собственной задачи, заменить неподходящие ветви подходящими из набора предоставляемых системой ветвей решения или создать свои. Кроме того, ему понадобится просмотреть промежуточные результаты решения. Все это требует наличия у пользователя определенных навыков работы с подобными системами и в худшем случае может потребовать столько же сетевого трафика сколько требуют стандартные методы решения задач в ГИС.

Недостатки перечисленных выше стратегий сокращения сетевого трафика и тот факт, что решением любой картографической задачи является тематическая карта, получаемая определенной последовательностью поддерживаемых ГИС операций, приводят к понятию декларативного решения картографических задач, при котором конечный пользователь определяет только то, что он хочет видеть на результирующей тематической карте. Основной идеей такой стратегии является освобождение конечного пользователя от создания алгоритма решения картографической задачи и перенос всей вычислительной нагрузки на «облако». При этом нагрузка на информационный канал оказывается минимальной, так как по нему передается только определение требуемых данных – декларация, и готовый ответ.

Для автоматического получения решения картографической задачи требуется сгенерировать алгоритм решения, после чего применить его к исходным данным. Для этого, прежде всего, нужна особая структура базы данных ГИС – необходимо отделение собственно картографических данных (таких как значения свойств объектов) от метаданных описывающих их в виде объектов (города, реки и т.д.) и их свойств (население, протяженность и т.п.). При этом важно выполнять классификацию, ориентируясь на семантический аспект данных, а не на их представление. Так, в векторных ГИС правильно будет выделить города, реки и страны, а не точки, линии и полигоны, поскольку город, например, в зависимости от масштаба может быть представлен точкой, полигоном или иерархической структурой.

После получения метаданных в виде множества объектов и их свойств, следует определить множество элементарных преобразований, каждое из которых оп-

ределено на некотором подмножестве метаданных и переводит их из одного элементарного состояния в другое.

При такой организации структуры базы данных ГИС картографическая задача может быть представлена в виде двух множеств элементарных состояний метаданных – исходного и целевого. А алгоритмом решения задачи будет последовательность элементарных преобразований, трансформирующая исходное множество состояний в целевое. Определение этой последовательности проводится с помощью нисходящего алгоритма поиска, при котором к целевому множеству элементарных состояний метаданных присоединяются подходящие по типу возвращаемых значений элементарные преобразования, что приводит к генерации нового множества элементарных состояний. К полученному множеству вновь присоединяются подходящие элементарные преобразования – итерации продолжают до тех пор, пока не будет получено исходное состояние, что означает успешное нахождение решения либо дальнейший поиск будет невозможен из-за отсутствия подходящих элементарных преобразований, что означает, что решения не существует.

В случае успешного завершения итераций алгоритмом решения картографической задачи будет найденная последовательность элементарных преобразований, приводящая исходное множество элементарных состояний в целевое. Решение задачи получится после применения этого алгоритма к исходным данным.

Рассмотрим пример автоматического поиска решения.

Пусть в базе данных ГИС определены следующие метаданные:

- объект hotel («отель») со свойством «туристическая компания»;
- объект monument («памятник») со свойством «количество посетителей в день»;
- объект path («путь») со свойством «тип транспорта»;
- объект route («маршрут»).
- Кроме того, определены следующие элементарные операции:
- list Get(conditions) – операция выборки объектов list некоторого типа по значениям их свойств conditions;
  - route Route(object, object, list) – операция вычисления маршрута route из одного пункта object в другой пункт object по сети путей list;
  - image Display(object, scale, area) – операция генерирования представления image объекта object в области area в масштабе scale;
  - list Extend(list, method) – операция расширения действия метода method на все объекты в списке list.

Необходимо построить карту туристических маршрутов от отелей некоторой туристической кампании к наиболее популярным памятникам (с количеством посещений в день выше определенного уровня).

Исходными данными к этой задаче будут исследуемая область area, масштаб целевой карты scale, объекты hotel и monument, соединенные линией – объектом route.

На первой итерации нужно перейти от маршрутов к метаданным следующего уровня. Так как операция Route требует список путей, его необходимо получить с помощью операции list Get({ «area, тип транспорта = автомобильный»}), которая опирается на известные данные и не требует дальнейших итераций.

Маршруты от всех отелей до всех памятников можно получить единственным способом – расширением операции Route на списки отелей list Get({area, туристическая компания = A}) и памятников list Get({area, количество посетителей в день > B}).

Таким образом, искомый алгоритм в терминах данной ГИС будет выглядеть следующим образом:

```
Extend(Get({туристическая компания = A}), Display(scale,
area));
Extend(Get({area, количество посетителей в день > B}),
Display(scale, area));
Extend(Extend({Get({туристическая компания = A}),
Get({area, количество посетителей в день > B})}),
Route(Get({«area, тип транспорта = автомобильный»}))),
Display(scale, area)).
```

Ответ сервера получится при применении найденного алгоритма для заданных значений исходных данных.

При применении такого подхода количество доступных для решения задач определяется количеством объектов, их свойств и элементарных операций, определенных в системе. Поэтому в реальных ГИС их число должно быть достаточно большим.

Использование концепции «облачных вычислений» в реализации распределенных геоинформационных систем обладает рядом важных достоинств – платформенной независимостью, освобождением конечных пользователей от поддержки вычислительных систем, масштабируемостью и гибкостью. Возникающую при этом проблему недостаточной пропускной способности информационного канала можно решить путем применения ряда высокоуровневых стратегий по ограничению количества «тяжелых» запросов от пользователя web-сервису – стратегии полной формализации решения, стратегии ручного и автоматического поиска алгоритма решения.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Chappell D.* A Short Introduction to Cloud Platforms. – MA: Addison-Wesley, 2008. – 154 p.
2. *Jones M., Tim A.* Cloud Computing with Linux. – NY : Diasoft, 2008. – 262 p.
3. *Slocum H., Terry A.* Thematic Cartography and Visualization. – NY : Diasoft, 2006. – 427 p.

**Василенко Виталий Сергеевич**

ООО Программные Технологии, г. Таганрог, Россия

E-mail: [vittech@mail.ru](mailto:vittech@mail.ru)

347900, г. Таганрог ,пер. Красногвардейский, №13, кв. 54, моб. тел.: 8-904-443-93-40

Инженер

**Vasilenko Vitaly Sergeevich**

Software Technologis Ltd., Taganrog, Russian Federation

E-mail: [vittech@mail.ru](mailto:vittech@mail.ru)

Appartment 54, № 13, Krasnogvardeyskiy, Taganrog, 347900, cell 8-904-443-93-40

Software engineer

УДК 338.27

**Л.А. Гинис**

#### **ОБЗОР МЕТОДОВ НАУЧНОГО ПРОГНОЗИРОВАНИЯ**

*В представленной работе рассматривается классификационная группировка наиболее применяемых и широко известных методов моделирования и прогнозирования социально-экономических процессов. Представлена их краткая характеристика и анализ.*