



## УВЕЛИЧЕНИЕ СКОРОСТИ ОБРАБОТКИ ДАННЫХ В СИСТЕМЕ ЭЛЕКТРОННОГО ОБУЧЕНИЯ

Бондарик В.М.<sup>1</sup>, Тюменцев А.Д.<sup>1</sup>, Камлач П.В.<sup>1</sup>, Будник А.В.<sup>2</sup>, Алтавилл Н.М.<sup>1</sup>, Тавгенъ Т.А.<sup>3</sup>

<sup>1</sup> Белорусский государственный университет информатики и радиоэлектроники, г. Минск, Беларусь, [bondarik@bsuir.by](mailto:bondarik@bsuir.by);

<sup>2</sup> Белорусская государственная академия связи, г. Минск, Беларусь, [a.budnik@bsac.by](mailto:a.budnik@bsac.by);

<sup>3</sup> Белорусский национальный технический университет, г. Минск, Беларусь, [ttavgen@bntu.by](mailto:ttavgen@bntu.by)

Abstract. A method is proposed for scaling the database of an e-learning system using the MapReduce model to increase the speed of data processing, reduce the load on the server and increase the performance of the system as a whole.

В современных условиях организации учебного процесса с использованием системы электронного обучения (СЭО) создается большое количество неструктурированных данных, таких как текстовые документы, изображения, видеозаписи, машинные коды, таблицы и т.п. Вся эта информация может храниться в различных репозиториях, даже за пределами учреждения образования. Пользователи СЭО могут иметь доступ к огромному массиву собственных данных и не иметь необходимых инструментов для организации взаимосвязи между этими данными, что затрудняет получение значимых выводов о результатах учебного процесса. Использование традиционных методов анализа информации не всегда дает возможность контролировать огромные объемы постоянно растущих и обновляемых данных. Применение технологий *Big Data* позволяет эффективно обрабатывать структурированные и неструктурированные данные больших объемов и значительного многообразия горизонтально масштабируемыми программными инструментами.

Горизонтальное масштабирование предполагает увеличение количества вычислительных ресурсов за счет увеличения количества серверов. Если при работе с приложением пользователь обратился к большому объему данных и вычислительный сервер не справляется с нагрузкой, то время отклика может заметно увеличиться. Для решения проблемы быстрого доступа к большим объемам информации предлагается использовать модель *MapReduce* [1].

*MapReduce* – это модель распределенной обработки данных, предложенная компанией *Google* для обработки больших объемов данных на компьютерных кластерах.

Предполагается, что данные организованы в виде некоторых записей. Поэтому обработка данных происходит в две стадии:

1. Стадия *Map*. На этой стадии данные преобразуются при помощи функции *Map*. Работа этой стадии заключается в преобразовке и фильтрации данных. Функция *Map*, примененная к одной входной записи, выдает множество пар ключ-значение. Множество может выдать только одну запись, может не выдать ничего, а может выдать несколько записей.

2. Стадия *Reduce*. На данной стадии данные, полученные из функции *Map*, будут обработаны и возвращены в соответствии с изначально сформулированной задачей.

Модель *MapReduce* может быть применена при разработке приложений для систем электронного обучения, позволяющих автоматизировать процесс усвоения знаний, текущей, промежуточной и итоговой аттестации с заполнением соответствующих документов. Эффективность использования предложенной модели обусловлена большим количеством данных, которые необходимо фильтровать и возвращать, что занимает довольно значительную часть времени при обработке запроса [2].

Для современной СЭО предложена асинхронная обработка данных с разделением их между несколькими процессами и агрегированием на уровне бизнес-логики. Для полноценной работы модели *MapReduce* в СЭО необходимо реализовать так называемую «сетку» с добавлением в нее нескольких виртуальных серверов. Чем больше «сетка», тем серьезней проблему можно решить при обработке данных, разбивая ее на более мелкие составляющие, которые нужно решить, и тем выше уровень параллелизма, который потенциально может быть достигнут. Асинхронный конвейер *ASP.NET*, объединенный с несколькими конвейерами на сервере, обеспечивает параллелизм в ядрах одного сервера. Еще одно преимущество архитектурной серверной «сетки» заключается в том, что она не полагается на главный узел для работы. В предложенной нами структуре в «сетке» серверов нет главного узла. Любой серверный узел может инициировать запрос *MapReduce*. При сбое любого из серверов, обработка данных может быть передана на любой другой сервер при условии, что входные данные для проводимой операции доступны.

Использование функции *MapReduce* позволяет увеличить скорость обработки данных в СЭО приблизительно в четыре раза без каких-либо серьезных изменений в архитектуре проекта и без значительных вложений в серверное оборудование.

### Литература

1. Горизонтальное масштабирование базы данных реального проекта с помощью SQL Azure Federations [Электронный ресурс]. – Режим доступа: <https://msdn.microsoft.com/ru-ru/dn458578.aspx>.

2. Принцип работы с большими данными, парадигма MapReduce [Электронный ресурс]. – Режим доступа: <https://msdn.microsoft.com/en-us/magazine/mt147240.aspx>.