

## ВЛИЯНИЕ АЛГОРИТМОВ СЖАТИЯ ДАННЫХ НА ПРОИЗВОДИТЕЛЬНОСТЬ ВЕБ-ПРИЛОЖЕНИЙ

*Анциферова Е.И.*

*гр.467041*

*Белорусский государственный университет информатики и радиоэлектроники,  
г. Минск, Республика Беларусь*

*Научный руководитель: Шевчук О.Г. – кандидат технических наук, доцент кафедры ИКТ*

**Аннотация.** В материалах доклада рассматривается влияние алгоритмов сжатия данных на производительность веб-приложений. Происходит сравнение основных методов сжатия, таких как Gzip, Brotli и Zstandard их эффективность в снижении объема передаваемых данных и влияние на скорость загрузки страниц. Рассматриваются экспериментальные результаты и рекомендации по выбору оптимального алгоритма в зависимости от особенностей веб-приложения.

**Ключевые слова:** сжатие данных, производительность, Gzip, Brotli, Zstandard, скорость загрузки, lossless алгоритмы, lossy алгоритмы

**Введение.** В настоящее время улучшение клиентского опыта имеет первостепенное значение для повышения вовлеченности и достижения успеха в бизнесе. Одним из важнейших аспектов, который существенно влияет на удовлетворенность клиентов, является скорость загрузки веб-сайта [1].

В условиях растущей конкуренции и стремления к удержанию пользователей, улучшение клиентского опыта становится одной из ключевых задач бизнеса. Пользователи ожидают мгновенного отклика и плавной работы веб-сайтов. Несомненно, скорость загрузки контента играет решающую роль в восприятии веб-ресурса и влияет на его успех.

Веб-приложения передают огромное количество данных, включая текстовые файлы, изображения, видео и другие ресурсы. Высокая нагрузка на сеть может замедлить загрузку страниц, особенно на устройствах с низкой пропускной способностью соединения. Для преодоления этих ограничений разработчики все чаще прибегают к использованию алгоритмов сжатия данных.

Использование алгоритмов сжатия позволяет уменьшить объем передаваемых файлов, что сокращает время загрузки страниц и снижает потребление сетевого трафика. Это особенно важно при

передаче текстовых данных, таких как HTML, CSS, JavaScript и JSON.

Сжатие данных позволяет существенно сократить объем передаваемых файлов, что приводит к уменьшению времени загрузки, оптимизации потребления сетевого трафика и повышению общей производительности приложения. Эта технология особенно актуальна для мобильных подключений, где скорость интернета может быть ограничена.

На сегодняшний день наиболее популярными алгоритмами сжатия для веб-приложений являются Gzip, Brotli и Zstandard. Каждый из них имеет свои особенности, такие как уровень сжатия, скорость декомпрессии и нагрузка на вычислительные ресурсы. Выбор подходящего алгоритма зависит от множества факторов, включая характеристики сервера, требования к скорости обработки данных и поддержку браузерами.

Правильный выбор алгоритма сжатия зависит от многих факторов, включая характеристики оборудования сервера, требования к скорости обработки данных и поддержку алгоритмов браузерами. Кроме того, важным аспектом является нахождение баланса между степенью компрессии и возможной нагрузкой на серверное и клиентское оборудование.

**Основная часть.** Сжатие данных представляет собой процесс уменьшения их объема без потери информации с целью оптимизации передачи и хранения. В контексте веб-приложений это позволяет снизить нагрузку на сеть, уменьшить время загрузки страниц и, как следствие, повысить удобство взаимодействия пользователя с системой.

Существуют два типа сжатия: без потерь (lossless), с потерями (lossy).

Lossless алгоритмы сокращают размер данных без изменения их структуры (Gzip, Brotli, Zstandard). Lossy алгоритмы используют частичное удаление информации для значительного снижения объема (JPEG, WebP, MP3).

Рассмотрим алгоритм сжатия веб-контента Gzip. Он является одним из самых популярных и давно используемых алгоритмов сжатия в веб-разработке. Он был разработан в 1990-х годах и с тех пор зарекомендовал себя как надежный и эффективный инструмент для уменьшения объема текстовых данных в интернет-приложениях.

Gzip обеспечивает значительное сокращение объема данных, особенно для текстовых файлов, таких как HTML, где происходит уменьшение избыточности за счет удаления повторяющихся символов и структур; CSS, где сжатию поддаются стили, содержащие часто повторяющиеся ключи и значения; JavaScript, где упаковываются длинные цепочки символов, комментарии и пробелы. В результате, использование Gzip может сократить размер файлов на 60–80% в зависимости от структуры данных.

Рассмотрим разработанный Google алгоритм Brotli. Его преимуществами является поддержка большинством современных браузеров, оптимизация для веб-ресурсов, улучшенное сжатие по сравнению с Gzip, особенно для текстовых файлов.

Одной из ключевых особенностей Brotli является его способность использовать кодирование Huffman — метод, который основывается на вероятностном анализе. Часто встречающиеся символы и фразы получают короткие коды, а редкие — более длинные, что позволяет минимизировать общий объем данных.

Также Brotli использует заранее заданный словарь, содержащий часто встречающиеся фразы и выражения, которые особенно актуальны для веб-контента. Например, фразы из HTML-разметки или JavaScript-кода могут быть упакованы в более компактный формат. Эта особенность позволяет алгоритму работать быстрее и эффективнее, чем Gzip, особенно с текстовыми файлами.

Zstandard — сравнительно новый алгоритм, разработанный Facebook. Он ориентирован на быструю компрессию и декомпрессию, что делает его удобным для серверных приложений.

Этот алгоритм сочетает гибкость и высокую производительность, что делает его универсальным инструментом для различных сценариев. Одной из ключевых особенностей алгоритма является возможность регулировать уровни компрессии в диапазоне от 1 до 22, позволяя разработчикам

адаптировать процесс под задачи. Низкие уровни обеспечивают максимальную скорость обработки данных, что важно для приложений, работающих в реальном времени.

Высокие уровни компрессии позволяют достичь значительного уменьшения объема файлов, что особенно полезно при передаче больших объемов данных. Благодаря высокой скорости компрессии и декомпрессии, Zstandard демонстрирует одну из лучших производительностей среди современных алгоритмов, успешно справляясь с обработкой информации в реальном времени.

Использование сжатия данных позволяет достичь значительного уменьшения объема передаваемой информации, что приводит к следующим улучшениям: сокращение времени загрузки — уменьшение размера файлов на 30–80% снижает задержки при передаче данных по сети; снижение нагрузки на сервер — передача сжатых данных уменьшает объем передаваемого трафика, снижая использование пропускной способности; оптимизация мобильных подключений — компрессия особенно важна для пользователей с медленными или нестабильными соединениями. Однако чрезмерное сжатие может привести к увеличению нагрузки на процессор сервера и клиента, что особенно критично для ресурсоемких приложений. Поэтому важно находить баланс между степенью сжатия и скоростью обработки данных.

Gzip обеспечивает средний уровень сжатия при высокой скорости компрессии и декомпрессии. Он широко поддерживается браузерами и серверами и подходит для динамических данных.

Brotli демонстрирует более высокий уровень сжатия, чем Gzip, особенно для текстовых данных. Он менее быстрый, но оптимален для статического контента, такого как CSS и JavaScript, и поддерживается современными браузерами.

Zstandard сочетает высокий уровень сжатия с самой высокой скоростью обработки данных. Он идеален для серверных приложений и обработки больших массивов данных, но имеет ограниченную поддержку браузерами.

При выборе алгоритма компрессии следует учитывать специфику веб-приложения и его пользователей. Для большинства случаев рекомендуется Brotli из-за его высокой эффективности при передаче статических файлов [2]. Gzip остается актуальным для обратной совместимости и быстрого сжатия в реальном времени. Zstandard подходит для серверных приложений и обработки больших объемов данных.

Для оценки влияния алгоритмов сжатия данных на производительность веб-приложений были проведены тестирования с различными типами контента и в разных сценариях. Основная цель экспериментов заключалась в определении эффективности алгоритмов Gzip, Brotli и Zstandard в снижении объема передаваемых данных, ускорении загрузки страниц и минимизации нагрузки на серверы.

Для исследования использовались различные типы контента, которые часто встречаются в веб-приложениях. Среди них текстовые файлы, такие как HTML, CSS, JavaScript, а также JSON и XML, которые широко применяются для структурированных данных. HTML позволяет уменьшить объем за счет оптимизации повторяющихся элементов структуры, CSS выигрывает благодаря сокращению ключей и значений, а в JavaScript эффективно сжимаются длинные цепочки, пробелы и комментарии.

Изображения в формате SVG, содержащие текстовые элементы и метаданные, также поддаются сжатию, тогда как форматы JPEG и PNG имеют встроенные методы компрессии, из-за чего дополнительные алгоритмы почти не улучшают их размер.

Веб-шрифты, такие как WOFF и TTF, успешно сжимаются при использовании Brotli, что особенно полезно для ускорения загрузки страниц. Видео и аудиофайлы, например, MP4, MP3 и WebM, уже используют сжатие с потерями, поэтому дополнительная компрессия для них не дала значительных результатов.

Также были протестированы API-ответы в форматах JSON и XML, которые играют важную роль в обмене данными между клиентами и серверами. Эти форматы продемонстрировали высокий уровень

уменьшения объема из-за предсказуемой структуры. Отдельное внимание было уделено текстовым лог-файлам серверов, так как они имеют значительные размеры, и Zstandard показал лучшие результаты по скорости сжатия и минимальным задержкам при их обработке.

Каждый тип контента дал различные результаты. Brotli оказался оптимальным для CSS и JavaScript благодаря высокой степени сжатия, Zstandard продемонстрировал исключительную производительность при работе с большими данными, такими как логи или API-ответы, а Gzip эффективно справился с динамическими текстовыми файлами, минимизируя нагрузку на процессор. Такой подход показал, что выбор алгоритма зависит от структуры и особенностей обрабатываемого контента.

По влиянию на скорость загрузки Brotli обеспечил сокращение времени загрузки статического контента, такого как CSS и JavaScript, на 25–35% по сравнению с Gzip. Благодаря этому он становится оптимальным выбором для заранее сжатых файлов. Zstandard продемонстрировал высокую производительность при компрессии динамических данных, минимизируя задержки в высоконагруженных серверных приложениях. Gzip сохранил свою эффективность для обработки мелких текстовых файлов, обеспечивая минимальную нагрузку на сервер.

По коэффициенту сжатия Gzip уменьшил размер текстовых данных на 60–70%, что обеспечивает значительное сокращение объема. Brotli превзошел Gzip, сократив размер на 70–80%, особенно для больших текстовых массивов. Zstandard достиг аналогичных результатов, сравнимых с Brotli, но обеспечил значительно более высокую скорость компрессии, что делает его предпочтительным для серверных задач.

Приведем сравнение алгоритмов сжатия по коэффициенту сжатия (рис. 1).

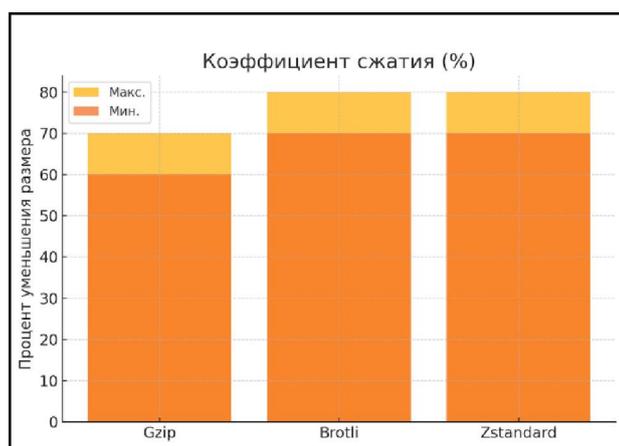


Рисунок 1 – Сравнение алгоритмов сжатия по коэффициенту сжатия

Еще одним важным показателем является скорость загрузки (рис.2).

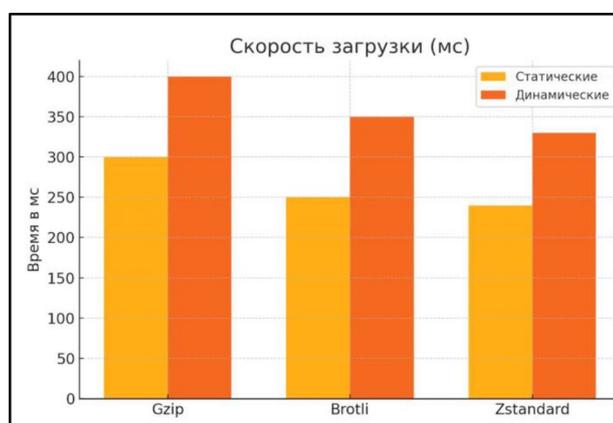


Рисунок 2 – Сравнение алгоритмов сжатия по скорости загрузки

Для того, чтобы сравнить скорость обработки при компрессии и декомпрессии обратимся к диаграмме

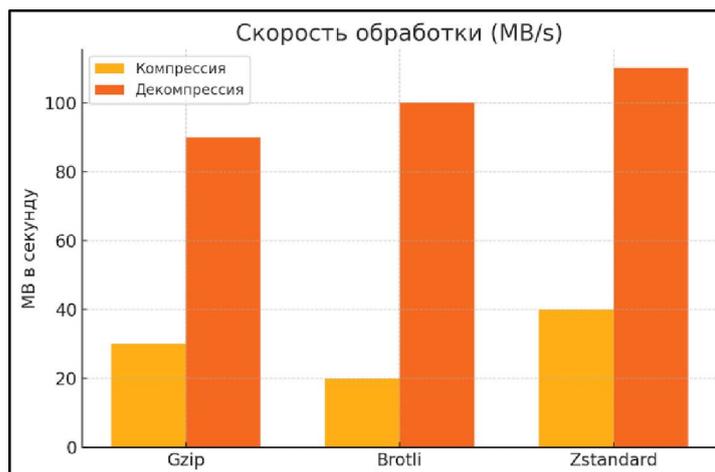


Рисунок 3 – Сравнение алгоритмов сжатия по скорости обработки при компрессии и декомпрессии

**Заключение.** Алгоритмы сжатия данных играют важную роль в оптимизации веб-приложений, позволяя уменьшить объем передаваемых файлов и улучшить производительность системы. Gzip, Brotli и Zstandard обладают разными характеристиками, и выбор метода сжатия должен учитывать баланс между степенью компрессии, скоростью обработки и нагрузкой на сервер. Грамотное применение компрессии способствует улучшению пользовательского опыта, снижению нагрузки на сеть и более эффективному использованию ресурсов.

### **Список литературы**

1. *Brotli vs. GZIP: A Comprehensive Comparison of Compression Algorithms.* / Abdul Rehman 2023. – 17 Min Read.
2. *How Image Compression Affects Your Website's Loading Times* – John Hughes 2023.

UDC 004.75

## **THE IMPACT OF DATA COMPRESSION ALGORITHMS ON WEB APPLICATION PERFORMANCE**

*Antsiferova E.I.*

*gr.467041*

*Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus*

*Shevchuk O.G. – Ph. D*

**Annotation.** The report examines the impact of data compression algorithms on the performance of web applications. It compares key compression methods such as Gzip, Brotli, and Zstandard, analyzing their effectiveness in reducing the volume of transmitted data and their influence on page load speed. Experimental results and recommendations for selecting the optimal algorithm based on the characteristics of the web application are discussed.

**Keywords:** Data compression, performance, Gzip, Brotli, Zstandard, load speed, lossless algorithms, lossy algorithms