

УДК 004.056.3:004.451.5

## СПОСОБ АДАПТИВНОЙ ДЕДУПЛИКАЦИИ С ПРИМЕНЕНИЕМ МНОГОУРОВНЕВОГО ИНДЕКСА РАЗМЕЩЕНИЯ КОПИРУЕМЫХ БЛОКОВ ДАННЫХ

<sup>1</sup>Казakov В.Г., <sup>2</sup>Федосин С.А., <sup>2</sup>Плотникова Н.П.

<sup>1</sup>НОУ ВПО «Московский технологический институт «ВТУ», Москва, e-mail: vgzakakov@gmail.com;

<sup>2</sup>ФГБОУ ВПО «Мордовский государственный университет им. Н.П. Огарёва», Саранск, e-mail: fedosinsa@mrsu.ru.

Данное исследование посвящено применению технологии дедупликации в системах хранения данных. В работе представлен способ адаптивной дедупликации, отличающийся применением многоуровневого индекса размещения копируемых блоков данных. Данная особенность позволяет унифицировано работать с различными применяемыми способами дедупликации на файловом и блочном уровнях, что обуславливает возможность балансировать вычислительную нагрузку системы при регулируемом уровне дедупликации. Источниками данных для модуля автоматического выбора метода и параметров дедупликации являются файловые метаданные, системные настройки, а также накапливаемая статистическая информация. Для повышения производительности разработан специализированный алгоритм группировки индексов, основанный на принципе объединения файлов в классы согласно производимой оценке подобия. Проведенные экспериментальные исследования эффективности предложенного подхода произведены с применением разработанного специализированного программного обеспечения.

**Ключевые слова:** восстановление данных, дедупликация, резервное копирование, системы хранения

## METHOD OF ADAPTIVE DEDUPLICATION WITH MULTILEVEL BLOCK INDEXING

<sup>1</sup>Kazakov V.G., <sup>2</sup>Fedosin S.A., <sup>2</sup>Plotnikova N.P.

<sup>1</sup>Moscow Technological Institute «WTU», Moscow, e-mail: vgzakakov@gmail.com;

<sup>2</sup>Mordovia State University n.a. N.P. Ogarev, Saransk, e-mail: fedosinsa@mrsu.ru

Data deduplication is a lossless compression technology that has been widely used in the large-scale storage systems. In a data storage system, the performance of data deduplication algorithms are varying on the condition of file contents. This paper presents a method of an adaptive deduplication that uses both file-level and block-level deduplication techniques. This approach gives the ability to balance computational workload with regulated dedupe ratios. The module of adaptive deduplication uses file metadata, system settings and statistics. For improving the performance of the method we developed a specialized algorithm of multilevel block indexing and file classification. Files are automatically separated into classes according to the degree of similarity, that is based on application of Jaccard index and Broder theorem. The experimental evaluation driven by the real world datasets and developed software shows that our proposed approach effectively balance workload.

**Keywords:** data, deduplication, backup, recovery, storage

Постоянно увеличивающийся научный интерес к процессам эффективного хранения, обработки и защиты информации, наблюдаемый в настоящее время, обусловливается, прежде всего, беспрецедентным ростом данных. Совершенствование систем хранения и резервного копирования в настоящее время в большей степени направлено на использование вычислительных ресурсов для повышения эффективности работы. Активно развивающаяся технология дедупликация данных (англ. deduplication) служит для снижения объема хранимых данных посредством определения и удаления избыточных идентичных элементов в разных копиях и версиях данных. В то же время, как правило, процессы дедупликации создают дополнительную вычислительную нагрузку.

Дедупликация данных производится на различных уровнях: на уровне файлов (англ. file-level deduplication, далее FD), когда дублирующие файлы исключаются

из хранения, а также на уровне блоков. Дедупликация на блочном уровне может быть разделена на два основных направления: дедупликация по блокам фиксированного размера (англ. static chunking, далее SC); дедупликация по блокам различной длины, которая зависит от содержания данных (англ. content defined chunking, далее CDC). Считается, что дедупликация на файловом уровне в среднем снижает объем данных резервного копирования в три, четыре раза. Одно из исследований [7], основанное на масштабном, комплексном анализе данных, выявило, что применение дедупликации на уровне файлов вместе с использованием разреженных файлов (функциональная особенность файловых систем, англ. sparse files) является эффективным средством снижения объема хранимых данных, в том числе и в случае резервного копирования. Так, дедупликация с блоками переменной длины с использованием метода Рабина позволяет получить на 18–20% лучшую эф-

фективность, чем файловая дедупликация, а блочная дедупликация с фиксированной – на 11%. При исследовании дедупликации с применением инкрементного резервного копирования с созданием еженедельных полных копий блочная дедупликация с использованием метода Рабина оказалась на 32% эффективнее пофайловой. Результаты данного исследования позволяют сделать вывод о том, что эффективность файловой дедупликации данных была в целом недооценена ранее. В то же время файловая дедупликация требует намного меньшую вычислительную нагрузку и более проста в реализации.

Увеличение производительности дедупликации является актуальной целью многочисленных исследований в области вычислительной техники. В представленной работе предлагается оригинальная технология организации дедупликации при копировании данных. В отличие от существующих решений [3, 6] в разрабатываемом подходе предлагается гибридный адаптивный способ дедупликации, нацеленный на совместное использование пофайловой и различных вариантов блочной дедупликации для нахождения эффективного баланса снижения вычислительной нагрузки при сохранении достаточно высокого уровня дедупликации.

#### Способ адаптивной дедупликации данных

Эффективность дедупликации зависит от типа обрабатываемых данных [5]. Так, применение дедупликации с блоками различной длины (CDC) наименее эффективно в случае с зашифрованными и упакованными данными, например, в случае с аудио- и видеоформатами лучшим вариантом является дедупликация на файловом уровне (FD) или с фиксированными блоками (SC) большого размера. В случае, когда известно, какие данные передаются на вход в систему дедупликации, в зависимости, например, от типа файла, размера и его атрибутов система может различным образом включаться в его обработку. Таким образом, адаптация параметров дедупликации в зависимости от типа файловых данных предоставляет возможности для повышения эффективности работы системы.

В целях исследования эффективности предложенного способа применения гибридного механизма дедупликации разработана программная система [1], в которой источниками данных для модуля адаптивного выбора метода и параметров дедупликации являются файловые метаданные, пользовательские и системные настройки, опреде-

ляющие методы обработки для известных типов файлов, статистическая информация, накопленная системой (рис. 1). Параметры копирования и дедупликации доступны для коррекции пользователем в настраиваемых профилях каждой задачи копирования.

Адаптация системы происходит с использованием статистики, основанной на информации каталога системы резервного копирования. Например, в случае, если при резервировании некоторого файла была автоматически выбрана FD-дедупликация, но при выполнении последующих операций резервирования файл подвергается частым изменениям, системой в результате анализа может быть принято решение об изменении применяемого способа дедупликации.

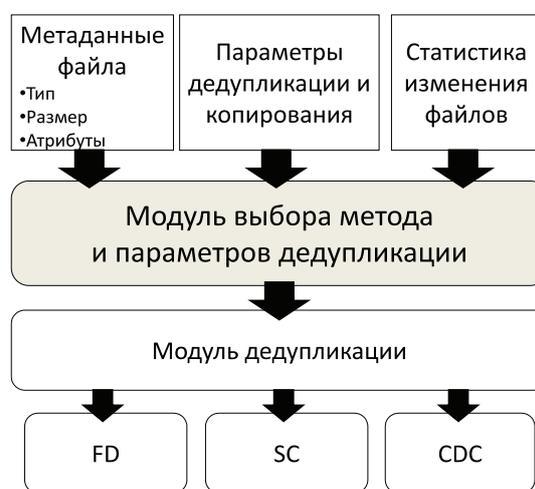


Рис. 1 Иллюстрация работы модуля выбора метода и параметров дедупликации

В предложенном способе блочная дедупликация данных с размером блоков, зависящих от данных (CDC), выполняется с использованием метода Рабина. При выполнении разметки каждого нового блока используется «скользящее окно» (рис. 2), для которого рассчитывается хеш-значение  $f$ . Выполнение критерия в общем виде  $f \bmod D = r$  определяет конец нового блока [3]. Используются следующие обозначения,  $D$  – делитель,  $r$  – предустановленные значения, зависящие от алгоритма.

На рис. 2 изображено «скользящее окно» при разметке нового блока с применением плавающих контрольных сумм. Для разбиения файла на блоки используется окно постоянного размера. В каждой позиции рассчитывается значение сигнатуры  $f$ . Процесс повторяется до завершения разметки всего файла. Для расчета значений хеша по блокам в системе используется алгоритм SHA.



Рис. 2 Процедура разметки файла на блоки

### Индекс размещения данных

В целях повышения производительности процесса дедупликации разработан специализированный алгоритм группировки индексов, основанный на принципе объединения файлов в классы. Так, индекс размещения данных разработанной системы разбивается по уровням: индекс файлов, классов подобия и индекс блоков данных (рис. 3). Для каждого файла в системе соз-

дается запись в индексе файлов, хранящая значение хеш-функции, ссылку на индекс блоков данных, где в случае, когда используется блочная дедупликация, хранится информация об имеющихся блоках с указанием хеш-значений. При использовании файловой дедупликации данные хранятся в едином блоке для обеспечения унификации работы системы и индекса размещения данных.

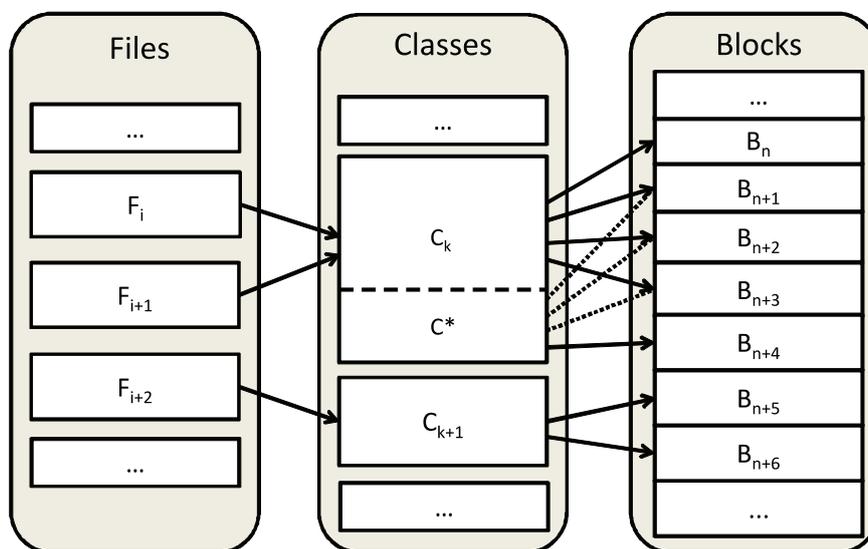


Рис. 3 Принципиальная схема индекса размещения данных

Ключевым элементом построения индекса блоков данных является применяемый принцип разбиения множества обрабатываемых файлов на классы. Так, в случае, когда логическое разделение блоков по файлам известно, предлагается применение группировки блоков по классам подобных файлов. При этом для оценки степени подобия используется индекс Жаккарда и обобщение теоремы Бродера [4]. Так, если  $S_1$  и  $S_2$  – два множества блоков двух файлов,  $H(S_1)$  и  $H(S_2)$  – хеш-блоков, тогда:

$$P(\min(H(S_1)) = \min(H(S_2))) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}.$$

Данное выражение показывает, что вероятность совпадения минимальных хеш-элементов множеств  $S_1$  и  $S_2$  равна индексу Жаккарда. Для возможности определения подобных файлов информация о мини-

мальном хеш-элементе хранится в файловом индексе. Имеющиеся исследования [1, 2] показывают, что описанный выше подход может быть использован с высокой точностью без необходимости применения техники перебора. Пример, изображенный на рис. 3, иллюстрирует частный случай размещения класса подобных файлов. Так, при появлении нового файла  $F_{i+1}$ , относящегося к классу  $C_k$ , дополнительного элемента в индексе классов не создается. Вместо этого различающиеся блоки ( $B_{n+4}$  на рисунке) помещаются в хранилище, а в индексе блоков добавляются необходимые ссылки. Для повышения производительности индекс файлов постоянно располагается в оперативной памяти, в то же время индексы блоков данных в целях обеспечения экономии памяти загружаются и остаются в кэше в зависимости от потребности системы.

Группировки индексов блоков, которые с высокой вероятностью будут затребованы вместе, последовательно или с небольшим интервалом также позволяют увеличить производительность. Так, при поиске идентификатора нового блока выполняется поиск в группе предыдущего совпадения. В отличие от известного подхода [3] нами предлагается схема, где разбиение по блокам является опциональным. Это обобщение позволяет реализовать гибридный подход, где в одном ряду может быть использована как файловая, так и блочная дедупликация в зависимости от работы заданных параметров.

### Заключение

Для возможности проведения комплексного анализа характеристик эффективности дедупликации и производительности разработан прототип программной системы, средствами которой можно проводить исследование разных аспектов процессов резервного копирования и восстановления как на моделируемых, так и на реальных данных [1]. В системе реализовано полное, инкрементное, дифференциальное и другие основные виды резервного копирования, предусмотрены возможности архивирования резервных копий, автоматизированное восстановление. В результате проведенных экспериментов было подтверждено, что уровень дедупликации предложенного способа при производстве инкрементного резервного копирования в среднем выше файловой и SC-дедупликации на 18%, и на некоторых исходных файловых данных системы приближается к уровню CDC-дедупликации. Применение принципа группировки блоков и подобных файлов в индексе показало повышение скорости работы поиска по индексу данных в среднем на 24% по проведенным экспериментам.

Важнейшими направлениями дальнейшей работы являются формальное описание алгоритма адаптированного изменения параметров дедупликации, детальное экспериментальное исследование количественных зависимостей характеристик эффективности предложенного способа дедупликации от массива используемых данных для копирования и публикация результатов. Помимо этого предложенная концепция применения накопленной статистики работы системы резервного копирования для адаптивного управления системой дедупликации является также малоизученной и нуждается в экспериментальных данных.

Описанный в работе подход позволяет применять технологию дедупликации не только с использованием выделенных вычислительных серверных мощностей на крупных предприятиях, но и в бытовых персональных компьютерах и даже на мобиль-

ных устройствах. Требуемый баланс между уровнем вычислительной нагрузки и эффективностью дедупликации достигается за счет адаптивного применения различных способов дедупликации на файловом и блочном уровнях. Практическая значимость работы заключается в возможности внедрения предложенного способа при разработке систем копирования и хранения данных.

### Список литературы

1. Казаков В.Г. Исследование эффективности и отказоустойчивости резервного копирования цифровых данных / В.Г. Казаков, С.А. Федосин, Д.П. Сидоров // Инфокоммуникационные технологии. – 2010. – Т. 8, № 4. – С. 13–19.
2. Bhagwat D. Content-based document routing and index partitioning for scalable similarity-based searches in a large corpus / D. Bhagwat, K. Eshghi, P. Mehra // KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. – 2007. – P. 105–112.
3. Bhagwat D. Extreme binning: scalable, parallel deduplication for chunk-based file backup / Bhagwat D., Eshghi K., Long D., Lillibridge M // In Proc. 17th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems. – 2009.
4. Broder A.Z. On the resemblance and containment of documents // SEQUENCES '97: Proceedings of the Compression and Complexity of Sequences. – 1997. – P. 21–29.
5. Jung H.M. Efficient Data Deduplication System Considering File Modification Pattern / Sang Y.P., Jeong G.L., Young W.K. // International Journal of Security and Its Applications. – 2012. – Vol. 6. – № 2.
6. Lee W. An Adaptive Chunking Method for Personal Data Backup and Sharing / Lee W., Park C. // In Proc. 8th USENIX Conference on File and Storage Technologies. – 2010.
7. Meyer D. A study of practical deduplication / Meyer D., Bolosky W. // In Proc. 9th USENIX Conference on File and Storage Technology. – 2011.

### References

1. Kazakov V.G. Issledovanie effektivnosti i otkazoustoichivosti rezervnogo kopirovaniya cifrovih dannih / V.G. Kazakov, S.A. Fedosin, D.P. Sidorov // Infokommunikacionnye tehnologii. 2010. T. 8, no. 4. pp. 13–19.
2. Bhagwat D. Content-based document routing and index partitioning for scalable similarity-based searches in a large corpus / D. Bhagwat, K. Eshghi, P. Mehra // KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. 2007. pp. 105–112.
3. Bhagwat D. Extreme binning: scalable, parallel deduplication for chunk-based file backup / Bhagwat D., Eshghi K., Long D., Lillibridge M // In Proc. 17th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems. 2009.
4. Broder A.Z. On the resemblance and containment of documents // SEQUENCES '97: Proceedings of the Compression and Complexity of Sequences. 1997. pp. 21–29.
5. Jung H.M. Efficient Data Deduplication System Considering File Modification Pattern / Sang Y.P., Jeong G.L., Young W.K. // International Journal of Security and Its Applications. 2012 Vol. 6 no. 2.
6. Lee W. An Adaptive Chunking Method for Personal Data Backup and Sharing / Lee W., Park C. // In Proc. 8th USENIX Conference on File and Storage Technologies. 2010.
7. Meyer D. A study of practical deduplication / Meyer D., Bolosky W. // In Proc. 9th USENIX Conference on File and Storage Technology. 2011.

### Рецензенты:

Щенников В.Н., д.ф.-м.н., профессор, заведующий кафедрой дифференциальных уравнений, ФГБУ ВПО «Мордовский государственный университет им. Н.П. Огарева», г. Саранск;

Панфилов С.А., д.т.н., профессор, заведующий кафедрой теоретической и общей электротехники, ФГБУ ВПО «Мордовский государственный университет им. Н.П. Огарева», г. Саранск.

Работа поступила в редакцию 19.07.2013.