УДК 004.056.3

ИВАНОВ Д.Г.

ОРГАНИЗАЦИЯ РЕЗЕРВИРОВАНИЯ В СИСТЕМАХ РАСПРЕДЕЛЕННОГО ХРАНЕНИЯ ДАННЫХ

В статье предлагается версия специальных кодов для защиты от потери информации при отказе в доступе к отдельным дискам при распределенном хранении информации. Разработанные коды представляют собой избыточные коды, позволяющие восстановить утерянные данные. Предложенная версия линейных кодов оптимизирована для задачи восстановления информации с двух или более отказавших дисков. Предложенная методология является достаточно общей и может быть применима к другим кодам восстановления, но наиболее целесообразна для кодов, основанных на операциях логического суммирования.

In paper the version of erasures codes for protecting against multiple failures in disk arrays for distributed networked storage is proposed. Developed erasures codes provide a codespecific means for reconstruction of erased data. Proposed version of linear erasures codes is optimized to reconstruct of information from two or more disks. The methodology we propose for both problems is completely general and can be applied to any erasure code, but is most suitable for XOR-based codes.

Введение

Динамичное развитие Интернета имеет логичным следствием появление и распространение технологий “облачных вычислений” – удаленного предоставления пользователю вычислительных ресурсов, ресурсов хранения данных и программного обеспечения как сервисных платных услуг. Эти ресурсы динамически реконфигурируются для адаптации к постоянно меняющейся нагрузке, что позволяет их оптимизировать их использование.

Эффективность практического использования прогрессивных технологий “облачных вычислений” напрямую зависит от решения ряда проблем, связанных с распределением ресурсов, обеспечением информационной безопасности и надежности. Если рассматривать только технологии удаленного предоставления ресурсов хранения данных, то именно проблема надежности стоит наиболее остро. Фактически речь идет об удаленном и не контролируемом пользователем хранении его данных на различных носителях доступных для постороннего вмешательства и подверженных риску потери хранящейся на них информации или отказе в доступе к ней.

Таким образом, проблема повышения надежности удаленного хранения данных на распределенных носителях является важной и актуальной для современного этапа развития технологий распределенной обработки информации.

Анализ существующих методов организации избыточного хранения данных на дисках

Информационный файл разделяется на n блоков, каждый из которых сохраняется на своем удаленном носителе. Блок, который хранится на j-том носителе, где $j \in \{1,..,n\}$, состоит из последовательности слов установленной длины : $a_{j1},a_{j2},\ldots,a_{jm}$.

При удаленном хранении информации существует ненулевая вероятность того, что на одном или нескольких носителях, по тем или иным причинам, произойдет потеря данных. Для того, чтобы исключить для пользователя возможность при этом потери информации, необходимо организовать резервирование ее хранения с использованием дополнительных носителей. Исходя из того, что в большинстве случаев, потери данных удаленными носителями в вероятностном плане имеют биномиальное распределение [1], наиболее вероятным является вариант потери данных на одном носителе. Вероятность потери данных сразу на двух носителях – на порядки меньше.

К настоящему времени предложено и активно используется на практике ряд систем удаленного разнесенного хранения данных с резервированием [1-4]. Помимо обеспечения надежности, резервирование позволяет повысить производительность доступа к данным, поскольку позволяет реализовать доступ к данным даже при условии возникновения временной задержки доступа к одному из носителей. При оценке эффективности резервирова-

ния наиболее часто [1] используются такие критерии:

- количество h носителей, при отсутствии доступа к которым можно реализовать доступ ко всему информационному файлу;
- количество k дополнительных носителей, используемых для резервирования;
- вычислительная сложность восстановления данных на основных носителей с использованием информации, хранящейся на дополнительных носителях, а также затраты времени на восстановление данных, доступ к которым отказан;
- повышение скорости доступа за счет использования дополнительных носителей.

Исходя из того, что наиболее вероятным является случай потери данных одним носителем, наиболее простым и эффективным способом резервирования [1] является использование помимо n основных одного дополнительного носителя, получившего название носителя четности. Хранящаяся на этом носителе последовательность слов $s_{11},s_{12},\ldots,s_{1m}$ представляет собой сумму по модулю два одноименных слов, хранящихся на основных носителях:

$$\forall j = 1,...,m : s_{1j} = \bigoplus_{i=1}^{n} a_{ji} \qquad (1)$$

При таком типе резервирования фактически не достигается повышения скорости доступа к данным, поскольку при резервировании и без такового время доступа к информационному файлу определяется максимальным временем получения доступа к n носителям.

С ростом объемов информационных файлов возрастает вероятность того, что возникнет отказ в доступе сразу для двух b более носителей. Наиболее известной системой с полным дублированием является Intermemory [2]. В системе используется n дополнительных носителей. Каждый блок разделяется на секции, которые хранятся по одной на каждом из дополнительных носителей. Основным достоинством такой системы является повышение скорости доступа к данным. Основным недостатком – большая избыточность использования объема памяти. Система гарантирует простое восстановление данных одного блока при потере доступа к любому из $2 \cdot n$ носителей, но используемое в ней дублирование данных не гарантирует восстановление файла при потере данных на двух носителях.

Близкое к описанному решение задачи резервирования данных при их хранении на удаленных носителей используется в системе Farsite [3]. В этой системе также используется дублирование данных с разнесением дублирующей информации по разным носителям. В системе с высокой вероятностью восстанавливается информационный файл при потере доступа к двум носителям, но это свойство не гарантируется: при отказе в доступе к одному из основных носителей и одному из тех, на котором хранится дублирующая информация – восстановление полностью исходного файла не возможно.

Наиболее перспективным путем организации резервирования разнесенных данных считается [3] использование корректирующих кодов или, точнее – их разновидности – erasures codes [4]. Такие коды достаточно хорошо изучены в теории связи для ситуации с потерей передачи символов сообщения. Наиболее часто в качестве таких кодов используются коды Рида-Соломона.

Однако, имеются существенные отличия в использовании корректирующих кодов при передаче данных и при хранении информации на разнесенных носителях. В классических корректирующих кодах Хемминга, БЧХ, Рида-Соломона для исправления возникающих при передаче ошибок четко выделяются фазы локализации искаженных единиц данных (битов или символов) и их непосредственного исправления принятого кода. При применении корректирующих кодов для восстановления информации в случае отказа в доступе двух носителей нет проблемы локализации. С другой стороны, отсутствуют принятые данные от этих носителей, которые обычно используются для исправления.

Таким образом, использование корректирующих кодов потенциально позволяет гарантированно восстанавливать информационный файл при потере данных на одном или двух носителях, используя при этом меньшее, по сравнению с дублированием, количество дополнительных накопителей. Однако, наличие существенных особенностей в восстановлении данных требует модификации существующих корректирующих кодов или создания новых их разновидностей, ориентированных на решение указанной выше задачи.

Целью исследований является разработка эффективного способа восстановления информации, хранящейся на распределенных носи-

телях при отказе в доступе к двум и более из них.

Способ избыточного кодирования для восстановления данных при отказе в доступе к двум дискам

Теоретически, для того, чтобы восстановить данные, хранящиеся на n дисках при отказе в доступе к h из них, информация каждого из n блоков должна сохраняться на h дисках (либо непосредственно, либо в виде линейной комбинации с данными других блоков). Исходя из этого, теоретически минимальное число дополнительных дисков равно h.

На практике требование использования минимального числа дополнительных дисков не является доминирующим [2]. Чаще учитываются факторы, связанные с временем восстановления исходя из вероятности отказа в доступе к определенному числу дисков. Так как наиболее вероятной является ситуация, при которой доступ ко всем дискам разрешен, то для удобства доступа, на основных дисках информация n блоков должна храниться в явном виде. Соответственно, это означает, что на дополнительных дисках, информация о каждом n блоках должна храниться минимум на h-1 дисках.

Учитывая, что при возникновении отказае в доступе, наиболее вероятное число недоступных дисков равно единице, то на первом дополнительном диске наиболее целесообразным является хранение сумм (1) данных всех n дисков, что обеспечивает эффективное восстановление любого одного блока с использованием одного дополнительного диска.

При отказе в доступе к двум носителям с номерами q и r , где $q,r\in\{1,\ldots,n\}$ необходимо восстановить значения слов $a_{q1},a_{q2},\ldots,a_{qm}$ и $a_{r1},a_{r2},\ldots,a_{rm}$ по информации, хранящейся в n-2 основных и k дополнительных носителях. Для обеспечения возможности простого восстановления данных при потере доступа к одному носителю, как указывалось выше, на первом дополнительном носителе целесообразно хранить суммы по модулю 2 всех одноименных слов основных носителей (1).

Если рассматривать задачу восстановления слов, хранящихся на q-том и r-том основных носителях слов $a_{q1},a_{q2},\ldots,a_{qm}$ и $a_{r1},a_{r2},\ldots,a_{rm}$, то их значения могут быть найдены в результате решения систем двух линейных булевых уравнений вида:

$$\begin{cases} a_{qj}+a_{rj}=z_{1j} \\ a_{qj}=z_{2j} \end{cases} \text{ или } \begin{cases} a_{qj}+a_{rj}=z_{1j} \\ a_{rj}=z_{3j} \end{cases}, \quad \forall j\in\{1,\ldots,m\} \qquad (2)$$

где символом ‘+’ обозначена поразрядная операция суммирования по модулю 2. Первое уравнение в системах (2) трансформируется из (1), то есть для его получения используются данные, хранящиеся в первом дополнительном накопителе. Получение второго уравнения систем (2) существенно сложнее, поскольку априори значения q и r неизвестны. Очевидно, что второе уравнение системы (2) может быть получено из системы линейных булевых уравнений, которая при любых значениях q и r содержит уравнение, в которое в качестве слагаемого входит только a_{q1} или только a_{r1}. Если n является степеню 2, то пример такой системы может иметь вид:

$$\begin{cases} a_{1j}+a_{2j}+\ldots+a_{m/2,j}=y_j \\ a_{1j}+a_{2j}+\ldots+a_{m/4,j}+ \\ a_{m/2+1,j}+\ldots+a_{3m/4,j}=y_2 \\ \ldots \\ a_{1j}+a_{3j}+a_{5j}+\ldots+a_{m-1,j}=y_{\log_2 n,j} \end{cases} \quad \forall j\in\{1,..,m\} \qquad (2)$$

В общем случае произвольного значения n число уравнений системы вида (3) равно ближайшему целому, равному или большему $\log_2 n$: $\lceil \log_2 n \rceil$.Для восстановления информационного файла при отказе любых двух накопителей (основных или дополнительных) необходимо в одном дополнительном носителе хранить сумму по модулю 2 одноименных слов основных накопителей (1), суммы, определяемые системой (3) на $\lceil \log_2 n \rceil$ дополнительных накопителях и на еще одном дополнительном носителе дублировать блок с последнего основного носителя.

Таким образом, если каждое слово из пары носителей, доступ к которым отказан восстанавливается независимо, то число k_1 дополнительных носителей составляет:

$$k_1 = 2+\lceil \log_2 n \rceil \qquad (4)$$

Например, если n=4, то есть информационный файл сохраняется на 4-х основных накопителях, то для восстановления файла при отказе в доступе к любым двум файлам требу-

ется $k_1 = 4$ дополнительных носителя, в которых хранятся m слов, формируемых в вид

$$\begin{cases} s_{1,j} = a_{1j} + a_{2j} + a_{3j} + a_{4j} \\ s_{2j} = a_{1j} + a_{2j} \\ s_{3j} = a_{1j} + a_{3j} \\ s_{4j} = a_{4j} \end{cases}, \forall j = 1,...,m \qquad (5)$$

Очевидно, что при любом выборе пары $q,r \in \{1,2,3,4\}$ система (5) преобразуется к виду (2). Например, если q=2 и r =4, то система (5) преобразуется к виду:

$$\begin{cases} a_{2j} + a_{4j} = z_{1j} = s_{1j} + a_{1j} + a_{3j} \\ a_{2j} = z_{2j} = s_{2j} + a_{1j} \end{cases}, \forall j = 1,...,m \qquad (6)$$

Из системы (6) все m значений слов на 2-м и 4-м носителях, к которым отказано в доступе восстанавливаются достаточно просто.

Если будет отказано в доступе к одному из основных накопителей и одному из дополнительных, то информационный файл также легко восстанавливается. Действительно, в случае невозможности доступа к q-тому $q \in \{1,\ldots,n\}$ из основных накопителей и первому из дополнительных, на одном из оставшихся дополнительных накопителей всегда хранятся данные из q-того основного накопителя. Например, если произойдет потеря данных на 3-м основном и первом дополнительном носителях, то данные $a_{31},a_{32},\ldots,a_{3m}$ с 3-го носителя достаточно просто восстанавливаются с использованием информации с 3-го дополнительного носителя:

$$a_{3j} = s_{3j} + a_{1j}, \forall j \in \{1,...,m\}$$

Таким образом предложенная организация резервирования данных, хранящихся на разнесенных накопителях гарантирует восстановление информации при отказе в доступе к двум любым носителям. Вычислительные процедуры восстановления данных предельно просты и не сказываются заметным образом на времени доступа к информации.

Очевидным недостатком этой организации является большая избыточность резервирования. Для устранения этого недостатка предлагается перейти от независимо процедуры восстановления каждой пары из m слов к процедуре, объединяющей восстановление сразу h пар слов. Действительно, в описанном простейшем варианте решением системы вида (2) из двух линейных булевых уравнений восстанавливаются значения одноименных слов из 2-х носителей. Существует возможность усложнить систему уравнений с тем, чтобы она содержала $2 \cdot h$ линейных булевых уравнений и решением которой является h слов с каждого из 2-х носителей, к которым отказано в доступе.

Рассмотрим такую возможность для h=2. При отказе в доступе к q-му и r-тому основным носителям рассмотрим задачу восстановления пары слов с каждого из этих носителей: a_{ql} ,a_{ql+1} и a_{rl} , a_{rl+1}, где $l \in \{1,3,\ldots,m\text{-}1\}$. По аналогии с системой (2) для восстановление данных на паре носителей, к которым отказано в доступе может быть осуществлено путем решения для каждого значения l одной из 4-х возможных систем линейных булевых уравнений. Первыми двумя уравнениями системы являются суммы по модулю 2 l-тых и (l+1)-тых слов всех основных носителей:

$$\begin{cases} \begin{cases} a_{ql} + a_{rl} = g_{1l} \\ a_{q,l+1} + a_{r,l+1} = g_{2l+1} \\ a_{ql} = g_{3l} \\ a_{rl} + a_{r,l+1} = g_{4l} \end{cases}, \forall l \in \{1,3,...,m-1\} \\ \begin{cases} a_{ql} + a_{rl} = g_{1l} \\ a_{q,l+1} + a_{r,l+1} = g_{2l+1} \\ a_{ql} + a_{q,l+1} = g_{3l} \\ a_{rl} = g_{4l} \end{cases}, \forall l \in \{1,3,...,m-1\} \end{cases} \qquad (7)$$

Получения одной из означенных систем (7) при каждых конкретных значениях q и r по аналогии с изложенным выше осуществляется следующим образом. Первые два уравнения систем (7) получаются из (1), то есть для их получения используются данные, хранящиеся в первом дополнительном накопителе.

Получение второго уравнения систем (2) существенно сложнее, поскольку априори значения q и r неизвестны. Очевидно, что второе уравнение системы (2) может быть получено из системы линейных булевых уравнений, которая при любых значениях q и r содержит уравнение, в которое в качестве слагаемого входит только a_{ql}+$a_{q,l+1}$ или только a_{r1}.

Рассмотрим возможность восстановления двух пар слов на примере. Пусть, как и в рассмотренном выше случае информационный файл сохраняется на 4-х основных накопителях, то есть n=4. Информационный файл сохраняется на 4-х основных накопителях. Для восстановления файла при отказе в доступе к любым двум файлам требуется $k_1 = 4$ дополни-

тельных носителя, в которых хранятся m слов, формируемых в виде:

$$\begin{cases} s_{1,l} = a_{1l} + a_{2l} + a_{3l} + a_{4l} \\ s_{2,l+1} = a_{1,l+1} + a_{2,l+1} + a_{3,l+1} + a_{4,l+1} \\ s_{3l} = a_{1l} + a_{2l} + a_{2,l+1} \\ s_{4l} = a_{1l} + a_{l+1} + a_{3j} \\ s_{5l} = a_{4l} + a_{4,l+1} \end{cases}, \forall l = 1,3,\ldots,m-1 \quad (8)$$

Очевидно, что при любом выборе пары $q,r \in \{1,2,3,4\}$ система (8) преобразуется к виду (7). Например, если q=1 и r =4, то система (8) преобразуется к виду:

$$\begin{cases} a_{1l} + a_{4l} = g_{1l} = s_{1l} + a_{2l} + a_{3l} \\ a_{1,l+1} + a_{2,l+1} = g_{2l} + a_{2,l+1} + a_{3,l+1} \\ a_{1l} = g_{3l} = s_{3l} + a_{2l} + a_{2,l+1} \\ a_{4l} + a_{4,l+1} = g_{4l} = s_{5,l} \end{cases}, \forall l = 1,3,\ldots,m-1 \quad (9)$$

Очевидно, что система (9) совпадает с первой из систем (7) и ее решение позволяет однозначно восстановить значения слов a_{1l}, $a_{1,l+1}$ и $a_{4,l}$, $a_{4,l+1}$ для всех значений l.

Как видно из системы (8) для получения первых двух уравнений используется информация, хранящаяся на первом дополнительном накопителе. Для этого, на первом дополнительном носителе должно храниться m слов сумм по модулю 2 одноименных слов всех основных носителей. На остальных дополнительных накопителях достаточно хранить только половину, то есть m/2 слов.

Таким образом, при h=2 несколько увеличивается вычислительная сложность процедуры восстановления данных: вместо двух независимых систем уравнений решается одно, состоящее из 4-х линейных булевых уравнений. Число дополнительных носителей не изменяется и соответствует оценке (4), однако только первый из них хранит m слов, а остальные – хранят вдвое меньше слов. Таким образом, избыточность использования объема дополнительной памяти резервирования удалось уменьшить практически вдвое.

Предложенный подход к уменьшению избыточности может быть развит на случай объединения в одном уравнении 3-х, 4-х и большего числа слов разных блоков. Если, как было показано выше, переход к объединению в одной системе уравнений (8) двух смежных слов всех блоков позволял уменьшить число используемых дополнительных дисков вдвое, то, объединение в одной системе уравнений четырех смежных слов всех блоков позволяет снизить количество дополнительных дисков в 4 раза. В пределе, при объединении в одной системе линейных уравнений всех слов всех дисков достигается оговоренный выше теоретический минимум числа дополнительных дисков при усложнении процедуры восстановления.

Выводы

В результате проведенных исследований предложен простой способ резервирования данных, хранящихся на удаленных разнесенных носителях с использованием дополнительных носителей. Показано, предложенный способ обеспечивает гарантированное восстановление информационного файла при потере данных или отказе в доступе двух произвольных носителей (основных или дополнительных). Предложен также подход к уменьшению объема информации, хранящейся на дополнительных носителях.

Список литературы

1. Patterson D., Gibson G., Katz R. Case of Redundant Array of Inexpensive Disk (RAID).- Berkeley: University of California.-1987.
2. Chen Y., Edler J., Goldberg A., Gottlieb A., Sorti S., Yianilos P. A Prototype Implementation of Archival Intermemory.- Pricenton: NEC Research Institute.-1999.
3. Plank J. S., Thomasson M.G., On Practical Use of LDPC Erasure Codes for Distribute Storage Application: Technical Report UT-CS-03-510.- Department of Computer Science, University of Tennessee.-2004.
4. Pierre-Ugo Tournoux. On-the-fly erasure coding for real-time video applications / Pierre-Ugo Tournoux Emmanuel Lochin, J. Lacan, Amine Bouabdallah, and Vincent Roca // IEEE Transactions on Multimedia, – 2011.- Vol. 13,- № 4.- P. 797–812.