

Questão 1

Fernanda Maria de Souza

31 de Agosto de 2020

(a) Como funciona a alocação baseada em extents, e como ela se diferencia da alocação indexada?

A alocação baseada em extents usa um esquema similar, porém modificado e otimizado de alocações contíguas. Um extent é um único descritor que representa um conjunto de blocos físicos contíguos, sendo representado como um par endereço-comprimento, que identifica o endereço do bloco inicial e o comprimento do extent. Os mesmos são alocados para cada arquivo, com cada arquivo sendo composto de um ou mais extents. Na Figura 1 é apresentado um exemplo da alocação por extents.

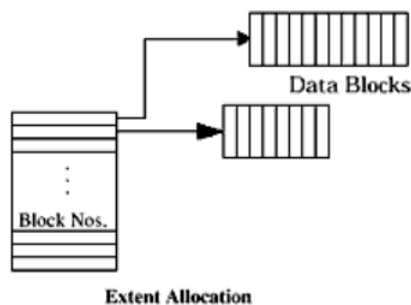


Figura 1: Representação da alocação baseada em extents

Por sua vez, sistemas de arquivos baseados em extents alocam blocos de disco em grandes grupos de uma só vez. Conforme um arquivo é gravado, um grande número de blocos é alocado. Os metadados do sistema de arquivos são escritos quando o arquivo é criado pela primeira vez. Após, as gravações subsequentes na primeira alocação de extents de blocos não requerem gravações de metadados adicionais, até que o próximo extent seja alocado.

Com relação a diferença entre a alocação baseada em extents e a alocação indexada, na última, um número de endereço de bloco é necessário para cada bloco

lógico, resultando em muitos metadados para cada arquivo. Para a alocação baseada em extents, apenas é necessário saber o número e o comprimento do bloco inicial para cada extent contíguo de blocos de dados. Um arquivo com alguns extents, porém muito grandes, requer apenas uma pequena quantidade de metadados.

(b) Qual a justificativa para introduzir um esquema de alocação diferente na evolução do ext3 para o ext4?

Com o aumento da capacidade de disco e requisitos de recursos de última geração, o ext4 foi criado. O sistema de arquivos ext3 usa o mapeamento de bloco indireto, o qual consiste em um esquema um-para-um de blocos lógicos para blocos do disco. Esse mapeamento indireto é muito eficiente para arquivos pequenos, mas acaba tendo muito overhead para arquivos maiores, acabando por ter um desempenho ruim especialmente na exclusão de arquivos grandes. No ext4, o uso de extents é usado de forma a mapear eficientemente blocos lógicos para físicos para grandes arquivos contíguos. Logo, foi introduzido um esquema de alocação diferente principalmente para diminuição do overhead em grandes arquivos.

(c) Até o ext3, o alocador de blocos retornava um bloco a cada vez que era invocado. Qual o problema que isso acarreta para o uso de extents? Que modificações são necessárias na alocação de blocos para que o esquema de extents funcione bem?

Para o uso de extents, o modo de retornar um bloco a cada vez que for invocado implica um alto índice de overhead principalmente para grandes arquivos, facilitando a fragmentação do disco quando vários gravadores são abertos simultaneamente. Para que o esquema de extents funcione bem é necessário implementar algumas funcionalidades como foi no ext4:

- Delayed allocation: permite que o ext4 espere para alocar os blocos atuais nos quais gravará os dados até que esteja pronto para enviar esses dados ao disco. Ao contrário, o ext3 alocaria blocos imediatamente, mesmo enquanto os dados ainda estavam fluindo para a cache de gravação.
- Multiblock allocation: por meio da Delayed allocation, é permitido juntar as gravações e tomar melhores decisões sobre como alocar blocos para as gravações que ainda não foram confirmadas.
- Tentar manter os blocos de dados de um arquivo no mesmo grupo de blocos que seu inode.
- Tentar manter todos os inodes em um diretório no mesmo grupo de blocos do diretório.
- O volume do disco é dividido em grupos de blocos de 128 MB.

- No caso de todos esses mecanismos falharem, pode-se usar um desfragmentador online.

(d) Qual dos esquemas introduz um overhead menor no acesso a arquivos pequenos (com 10 blocos ou menos), a alocação indexada ou a baseada em extents? Justique sua resposta, considerando o caso em que os blocos são contíguos e o caso em que os blocos estão totalmente espalhados (i.e., não há blocos adjacentes).

A baseada em extents introduz um overhead menor. Para arquivos grandes, altamente fragmentados ou espalhados, mais extents são necessários, logo, uma árvore de extent de profundidade constante é usada para armazenar as extents do mapa de um arquivo. Por isso, extents não são eficientes para representar arquivos espalhados. Para arquivos pequenos ou contíguos, é possível representar diretamente na estrutura ext4 inode, o que não provoca custos adicionais para o sistema de arquivos.

Referências

Ext4 Disk Layout. Wiki Kernel, 2019. Disponível em: https://ext4.wiki.kernel.org/index.php/Ext4_Disk_LayoutExtent_Tree. Acesso em: 29 de Ago. de 2020

Getting to know the Solaris filesystem, Part 1. SunWorld, 1999. Disponível em: <http://sunsite.uakom.sk/sunworldonline/swol-05-1999/swol-05-filesystem.html>. Acesso em: 29 de Ago. de 2020.

MATHUR, A. et al. The new ext4 filesystem: current status and future plans, IN: Proceedings of the Linux Symposium, 2007, Ottawa. Disponível em: <https://www.kernel.org/doc/ols/2007/ols2007v2-pages-21-34.pdf>. Acesso em: 29 de Ago. de 2020.

POMERANZ, Hal. Understanding EXT4 (Part 1): Extents, 2019. Disponível em: <https://www.sans.org/blog/understanding-ext4-part-1-extents/>. Acesso em: 29 de Ago. de 2020.