



Data Science Academy

www.datascienceacademy.com.br

Data Lake – Design, Projeto e Integração

O Que é Rack Awareness no Hadoop HDFS?

Em um grande cluster Hadoop, para melhorar o tráfego de rede durante a leitura/gravação do arquivo no HDFS, o namenode escolhe o datanode que está mais próximo do mesmo rack para a solicitação de leitura/gravação. O namenode obtém informações sobre o rack mantendo os ids do rack de cada datanode. Esse conceito de escolher datanodes mais próximos com base nas informações do rack é chamado de Rack Awareness no Hadoop.

O Rack Awareness é ter conhecimento da topologia do cluster ou, mais especificamente, como os diferentes nós de dados são distribuídos entre os racks de um cluster do Hadoop. A instalação padrão do Hadoop pressupõe que todos os nós de dados pertencem ao mesmo rack. O Rack Awareness é necessário pelas razões abaixo:

- Melhora a alta disponibilidade e confiabilidade dos dados.
- Melhora o desempenho do cluster.
- Melhora a utilização de largura de banda da rede.
- Evitar perda de dados se o rack inteiro falhar, embora a chance de falha do rack seja muito menor do que a da falha de um nó.
- Em geral, quanto maior a largura de banda no rack, menor latência.

Posicionamento de réplica via Rack Awareness no Hadoop

O posicionamento da réplica dos blocos de dados é fundamental para garantir alta confiabilidade e desempenho do HDFS. A otimização do posicionamento da réplica por meio do Rack Awareness diferencia o HDFS de outros Sistemas de Arquivos Distribuídos. A replicação de blocos em vários racks no HDFS é feita usando uma política da seguinte forma:

“Não mais do que uma réplica é colocada em um nó. E não mais do que duas réplicas são colocadas no mesmo rack. Isto leva a uma restrição pois o número de racks usados para replicação de blocos deve ser menor que o número total de réplicas de blocos”.

Quando um novo bloco é criado a primeira réplica é colocada no nó local. O segundo é colocado em um rack diferente e o terceiro é colocado em um nó diferente no rack local.

Ao replicar novamente um bloco, se o número de uma réplica existente for um, o segundo bloco é colocado em um rack diferente. Se o número de uma réplica existente for dois e se as duas réplicas existentes estiverem no mesmo rack, a terceira réplica será colocada em um rack diferente e assim por diante.

Uma política simples, mas não ótima, é colocar réplicas nos diferentes racks. Isso evita a perda de dados quando um rack inteiro falha e nos permite usar a largura de banda de vários racks enquanto fazemos a leitura dos dados. Essa política distribui uniformemente os dados entre as réplicas no cluster, o que facilita o balanceamento de carga em caso de falha do componente. Mas a maior desvantagem dessa política é que ela aumentará o custo da operação de gravação porque um gravador precisa transferir blocos para vários racks e a comunicação entre os dois nós em racks diferentes precisa passar por switches.

Na maioria dos casos, a largura de banda da rede entre máquinas no mesmo rack é maior que a largura de banda da rede entre máquinas em racks diferentes. É por isso que usamos a política de substituição de réplica. A chance de falha do rack é muito menor do que a da falha do nó. Isso não afeta a confiabilidade e a garantia de disponibilidade de dados. No entanto, ele reduz a largura de banda de rede agregada usada ao ler dados, pois uma réplica de bloco é colocada em apenas dois racks exclusivos, em vez de três.

E o desempenho?

Operação de replicação mais rápida: como as réplicas são colocadas no mesmo rack, elas usam largura de banda mais alta e latência mais baixa, tornando-a mais rápida.

Se o YARN não puder criar um contêiner no mesmo nó de dados no qual os dados consultados estão localizados, ele tentará criar o contêiner em um nó de dados dentro do mesmo rack. Isso é mais eficiente devido à maior largura de banda e menor latência dos nós de dados dentro do mesmo rack.



Vantagens de implementar o Rack Awareness

Minimiza o custo de gravação e maximiza a velocidade de leitura - o Rack Awareness coloca solicitações de leitura/gravação em réplicas no mesmo rack ou próximo, assim minimizando o custo de gravação e maximizando a velocidade de leitura.

Fornece largura de banda de rede maximizada e baixa latência - o Rack Awareness maximiza a largura de banda da rede por meio da transferência de blocos dentro de um rack. Especialmente com o Rack Awareness, o YARN é capaz de otimizar o desempenho de Jobs MapReduce. Ele atribui tarefas a nós que estão "mais próximos" de seus dados em termos de topologia de rede. Isso é particularmente benéfico nos casos em que as tarefas não podem ser atribuídas a nós nos quais seus dados são armazenados localmente.

Proteção de dados contra falha de rack - Por padrão, o namenode atribui as 2ª e 3ª réplicas de um bloco a nós em um rack diferente da primeira réplica. Isso fornece proteção de dados mesmo contra falhas de rack; no entanto, isso é possível somente se o Hadoop tiver sido configurado com Rack Awareness.

A configuração é feita através da manipulação de parâmetros dos arquivos de configuração do Hadoop. Aqui você encontra mais detalhes:

<https://hadoop.apache.org/docs/r3.0.3/hadoop-project-dist/hadoop-common/RackAwareness.html>

Aqui alguns exemplos de topologia que podem ser usadas:

<https://community.hortonworks.com/articles/43057/rack-awareness-1.html>