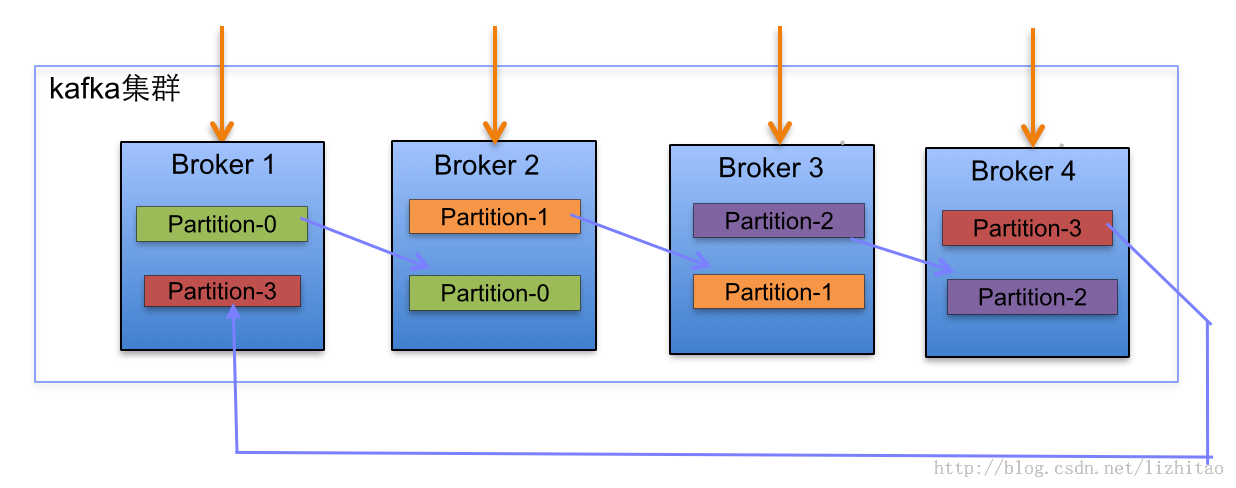
https://www.cnblogs.com/lytwajue/p/7205538.html

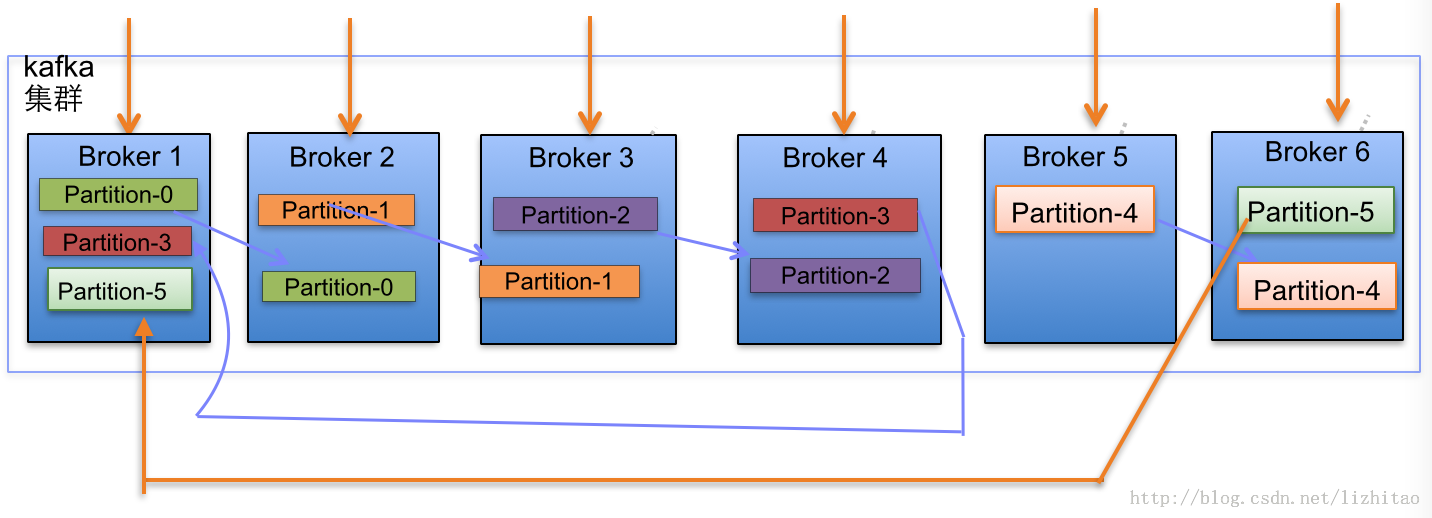
kafka分区及副本在broker的分配

部分内容參考自：<http://blog.csdn.net/lizhitao/article/details/41778193>

以下以一个Kafka集群中4个Broker举例，创建1个topic包括4个Partition，2 Replication；数据Producer流动如图所看到的：   
(1)   


pic

(2)当集群中新增2节点，Partition添加到6个时分布情况例如以下：



副本分配逻辑规则例如以下：   
在Kafka集群中，每一个Broker都有均等分配Partition的Leader机会。   
上述图Broker Partition中，箭头指向为副本。以Partition-0为例:broker1中parition-0为Leader。Broker2中Partition-0为副本。

上述图种每一个Broker(依照BrokerId有序)依次分配主Partition,下一个Broker为副本，如此循环迭代分配，多副本都遵循此规则。

副本分配算法例如以下：   
将全部N Broker和待分配的i个Partition排序.   
将第i个Partition分配到第(i mod n)个Broker上.   
将第i个Partition的第j个副本分配到第((i + j) mod n)个Broker上.

其实以上的算法是有误的，由于非常明显。每一个topic的分区0都会被分配在broker 0上。第1个分区都分配到broker 1上。直到partition的id超过broker的数据才開始从头開始反复，这样会导致前面几台机器的压力比后面的机器压力更大。

因此。kafka是先随机挑选一个broker放置分区0，然后再按顺序放置其他分区。

例如以下图的情况：

Topic:ljh\_test3 PartitionCount:10 ReplicationFactor:2 Configs:

Topic: ljh\_test3 Partition: 0 Leader: 5 Replicas: 5,6 Isr: 5,6

Topic: ljh\_test3 Partition: 1 Leader: 6 Replicas: 6,7 Isr: 6,7

Topic: ljh\_test3 Partition: 2 Leader: 7 Replicas: 7,2 Isr: 7,2

Topic: ljh\_test3 Partition: 3 Leader: 2 Replicas: 2,3 Isr: 2,3

Topic: ljh\_test3 Partition: 4 Leader: 3 Replicas: 3,4 Isr: 3,4

Topic: ljh\_test3 Partition: 5 Leader: 4 Replicas: 4,5 Isr: 4,5

Topic: ljh\_test3 Partition: 6 Leader: 5 Replicas: 5,7 Isr: 5,7

Topic: ljh\_test3 Partition: 7 Leader: 6 Replicas: 6,2 Isr: 6,2

Topic: ljh\_test3 Partition: 8 Leader: 7 Replicas: 7,3 Isr: 7,3

Topic: ljh\_test3 Partition: 9 Leader: 2 Replicas: 2,4 Isr: 2,4

这里分区0放到了broker5中。分区1–broker6。分区2—broker7….

再看一个样例：

Topic:ljh\_test2 PartitionCount:10 ReplicationFactor:2 Configs:

Topic: ljh\_test2 Partition: 0 Leader: 2 Replicas: 2,7 Isr: 2,7

Topic: ljh\_test2 Partition: 1 Leader: 3 Replicas: 3,2 Isr: 3,2

Topic: ljh\_test2 Partition: 2 Leader: 4 Replicas: 4,3 Isr: 4,3

Topic: ljh\_test2 Partition: 3 Leader: 5 Replicas: 5,4 Isr: 5,4

Topic: ljh\_test2 Partition: 4 Leader: 6 Replicas: 6,5 Isr: 6,5

Topic: ljh\_test2 Partition: 5 Leader: 7 Replicas: 7,6 Isr: 7,6

Topic: ljh\_test2 Partition: 6 Leader: 2 Replicas: 2,3 Isr: 2,3

Topic: ljh\_test2 Partition: 7 Leader: 3 Replicas: 3,4 Isr: 3,4

Topic: ljh\_test2 Partition: 8 Leader: 4 Replicas: 4,5 Isr: 4,5

Topic: ljh\_test2 Partition: 9 Leader: 5 Replicas: 5,6 Isr: 5,6