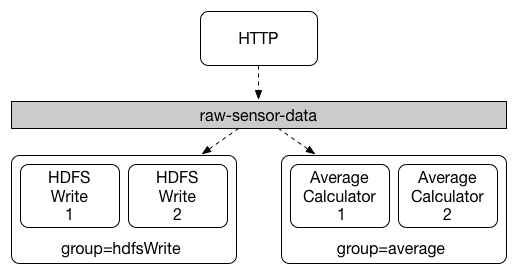
由于发布－订阅模型使得共享主题的应用之间连接更简便，创建给定应用的不同实例来进行弹性扩张的能力也同样重要。如果存在多个应用实例，那么同一应用的不同实例便会成为相互竞争的消费者，其中应该只有一个实例处理给定消息。   
Spring Cloud Stream通过消费者组的概念给这种情况进行建模。每一个单独的消费者可以使用spring.cloud.stream.bindings.input.group属性来指定一个组名字。下图中展示的消费者们，这一属性被设置为spring.cloud.stream.bindings.input.group=hdfsWrite或者spring.cloud.stream.bindings.input.group=average。   
   
所有订阅给定目标的组都会收到发布消息的一个拷贝，但是每一个组内只有一个成员会收到该消息。默认情况下，如果没有指定组，Spring Cloud Stream 会将该应用指定给一个匿名的独立的单成员消费者组，后者与所有其他组都处于一个发布－订阅关系中。