

配置参数	默认值	描述
<code>optional.&lt;hdr-value&gt;</code>	-	与 <code>mapping</code> 参数相同，但是在这个列表中指定的 Channel 是可选的，写入失败会忽略
<code>default</code>	-	事件必须写入的 Channel 列表，如果报头不存在或者报头的值没有指定映射列表

配置多路复用 Channel 选择器与配置复制选择器颇有几分不同。与往常一样，Source 写入的所有 Channel 必须在 Source 的 `channels` 参数中指定。为了使 Source 的多路复用 Channel 选择器生效，该 Source 的 `selector.type` 参数必须设置为 `multiplexing`。传递到 Channel 选择器的所有参数使用 Source 前缀，该 Source 后面跟着 `selector.`，如下所示：

```
agent.sources.avroSrc.type = avro
agent.sources.avroSrc.channels = c1 c2 c3
agent.sources.avroSrc.selector.type = multiplexing
agent.sources.avroSrc.selector.default = c3
```

对于每个事件，选择器查找配置中 `header` 参数指定的键的报头。接下来，它检查报头的值是否是配置中使用 `mapping` 前缀指定值的其中一个。如果是 `mapping` 的一个匹配，那么它通过映射将事件写出到 Channel。可选的映射也可以使用 `optional` 前缀而不是用 `mapping` 前缀进行指定。任何指定为可选的 Channel 列表写失败，将会被忽略。如果选择器没有找到匹配或报头本身不存在，那么它写事件到 `default` 参数指定的 Channel 中。如果一个事件没有映射到任何所需的 Channel，但是映射到一个或多个可选的 Channel，那么该事件被写出到可选 Channel 和默认 Channel，任何写出到默认 Channel 的失败将会导致抛出 `ChannelException` 异常。

下面展示了一个配置了多路复用 Channel 选择器的 Source 的实例：

```
agent.sources.avroSrc.type = avro
agent.sources.avroSrc.channels = c1 c2 c3 c4 c5
agent.sources.avroSrc.selector.type = multiplexing
agent.sources.avroSrc.selector.header = priority
agent.sources.avroSrc.selector.mapping.1 = c1 c2
agent.sources.avroSrc.selector.mapping.2 = c2
agent.sources.avroSrc.selector.optional.1 = c3
agent.sources.avroSrc.selector.optional.2 = c4
agent.sources.avroSrc.selector.optional.3 = c4
agent.sources.avroSrc.selector.default = c5
```

在这个例子中，Avro Source 写事件到四个 Channel。但是不像复制 Channel 选择器，不是所有的事件都被写入到所有的四个 Channel。对于每个事件，Channel 选择器查找使用