

```
agent.channels.memChannel.type = memory
agent.channels.memChannel.capacity = 10000
agent.channels.memChannel.transactioncapacity = 5000
```

配置文件配置了 Source 从 ActiveMQ 消息队列拉取数据，ActiveMQ 消息队列在主机 `usingflume.oreilly.com:61616` 读取 JMS 队列。队列信息通过很多参数来传递。对于每个 ActiveMQ 文件，初始化上下文工厂设置为 `org.apache.activemq.jndi.ActiveMQInitialContextFactory`，`JsonMessageConverter` 使用的是 ISO-8859-1 字符集。

转换 JMS 消息为 Flume 事件 *

JMS Source 可以配置为使用自定义代码将 JMS 消息转换成 Flume 事件，就像 HTTP Source 的处理程序。这使得 JMS Source 极其灵活，允许用户将 JMS 消息解析为专有格式的数据。与 Flume 绑定在一起的是 JMS 消息转换器（JMS Message Converter），支持默认 JMS 格式。JMS Source 可以传递配置到消息转换器，就像 HTTP Source。这个配置可以用来设置任何初始配置或转换器所需的状态。Source 处理所有的 JMS 事务；转换器不需要任何担心。

与 Flume 一起绑定的默认 JMS 消息转换器，基于 JMS 消息的格式和内容，分别将它们转换成 Flume 事件。JMS `ByteMessage` 只是读取，从消息中读取的字节被放入 Flume 事件的 body。对于 JMS `TextMessage`，转换器编码文本为字节数组并把它放入 Flume 事件的 body。通过 Source 配置，要使用的编码是可配置的。如果 `ObjectMessage` 从 JMS 队列读出，转换器在 `ObjectOutputStream` 封装了对象并且写到 `ByteArrayOutputStream`。然后会从流中读取到字节作为 Flume 事件的 body。在大多数情况下，这就是用户所需要的内容。

有时候，从消息中解析出为任何模式，并将其转换为一种可以在终端 Sink 更容易被序列化的格式，这也许更有意义。例如，如果有几个不同的应用程序以不同的序列化格式（例如 JSON 或 XML）写入 JMS 队列，那么可以编写一个转换器将这些转换成统一的格式，这样可以更容易在终端 Sink 被解析。

转换器必须实现 `JMSMessageConverter` 接口，在例 3-12 中有提及。JMS Source 实例化 `JMSMessageConverter$Builder` 的类，然后通过一个 Context 实例传递配置给 Builder 类的 `build` 方法，这个方法会返回一个完全配置好的转换器实例。

例3-12 JMS转换器接口

```
package org.apache.flume.source.jms;
public interface JMSMessageConverter {
    public List<Event> convert(Message message)
```