

```

    }

    @Override
    public void setChannels(List<Channel> channels) {
        this.channels = channels;
    }

    @Override
    public synchronized void setName(String name) {
        this.name = name;
    }

    @Override
    public synchronized String getName() {
        return name;
    }

    protected Map<String, Channel> getChannelNameMap() {
        Map<String, Channel> channelNameMap = new HashMap<String, Channel>();
        for (Channel ch : getAllChannels()) {
            channelNameMap.put(ch.getName(), ch);
        }
        return channelNameMap;
    }

    protected List<Channel> getChannelListFromNames(String channels,
        Map<String, Channel> channelNameMap) {
        List<Channel> configuredChannels = new ArrayList<Channel>();
        if(channels == null || channels.isEmpty()) {
            return configuredChannels;
        }
        String[] chNames = channels.split(" ");
        for (String name : chNames) {
            Channel ch = channelNameMap.get(name);
            if (ch != null) {
                configuredChannels.add(ch);
            } else {
                throw new FlumeException("Selector channel not found: "
                    + name);
            }
        }
        return configuredChannels;
    }
}

```