作者：徐子岩  
链接：https://www.zhihu.com/question/23015432/answer/115094859  
来源：知乎  
著作权归作者所有，转载请联系作者获得授权。

服务端是[http://Socket.io](//link.zhihu.com/?target=http://Socket.io" \t "_blank)的集群，供客户端（Web、移动端）连接。集群后面是一个Redis服务器，保存集群中每个节点（我们称之为Cluster）连接的客户端ID。同时Redis里面为每一个Cluster分配了一个队列，保存要推送到这个Cluster的消息。  
  
当有消息从某个客户端发出后，所连接的Cluster从Redis里面获取这个消息的目标客户端ID（由于我们同时支持一对一私聊和群组，因此一条消息可能会被推送到多个客户端），然后把消息Push到每个Cluster的消息队列里面。  
  
每一个Cluster都会以阻塞方式读取它所对应的消息队列，一旦发现有消息，就获取并且查看其目标客户端ID是不是连接在这个Cluster上。如果是，就通过[http://Socket.io](//link.zhihu.com/?target=http://Socket.io" \t "_blank)发送；如果不是就丢弃。然后继续阻塞读取，知道下一条消息到达。

一个Cluster（集群）找到对应的机架（假如一个机架40台服务器）在通过服务器看目标客户端ID是不是连接在上面，如果是通过socket发送，不是就丢弃，然后继续阻塞直到下一条消息到达。  
  
PS：我们之前使用过XMPP，但是如果是产品开发，要考虑XMPP是基于Erlang语言写的。如果需要二次开发，目前市面上Erlang工程师不是很多。  
  
PPS：对于Cluster间如何互通，也可以考虑使用Redis的Pub/Sub。如果没有特殊的场景，推荐用[http://Socket.io](//link.zhihu.com/?target=http://Socket.io" \t "_blank)自带的 [GitHub - socketio/socket.io-redis: Adapter to enable broadcasting of events to multiple separate socket.io server nodes.](//link.zhihu.com/?target=https://github.com/socketio/socket.io-redis/" \t "_blank) 我们Worktile基础版的消息服务就是基于这个组件实现的Cluster之间消息通知。或者让Cluster之间通过TCP协议通信。不过在高并发的情况下，需要考虑TCP包的分割合并问题，因此不是特别推荐。