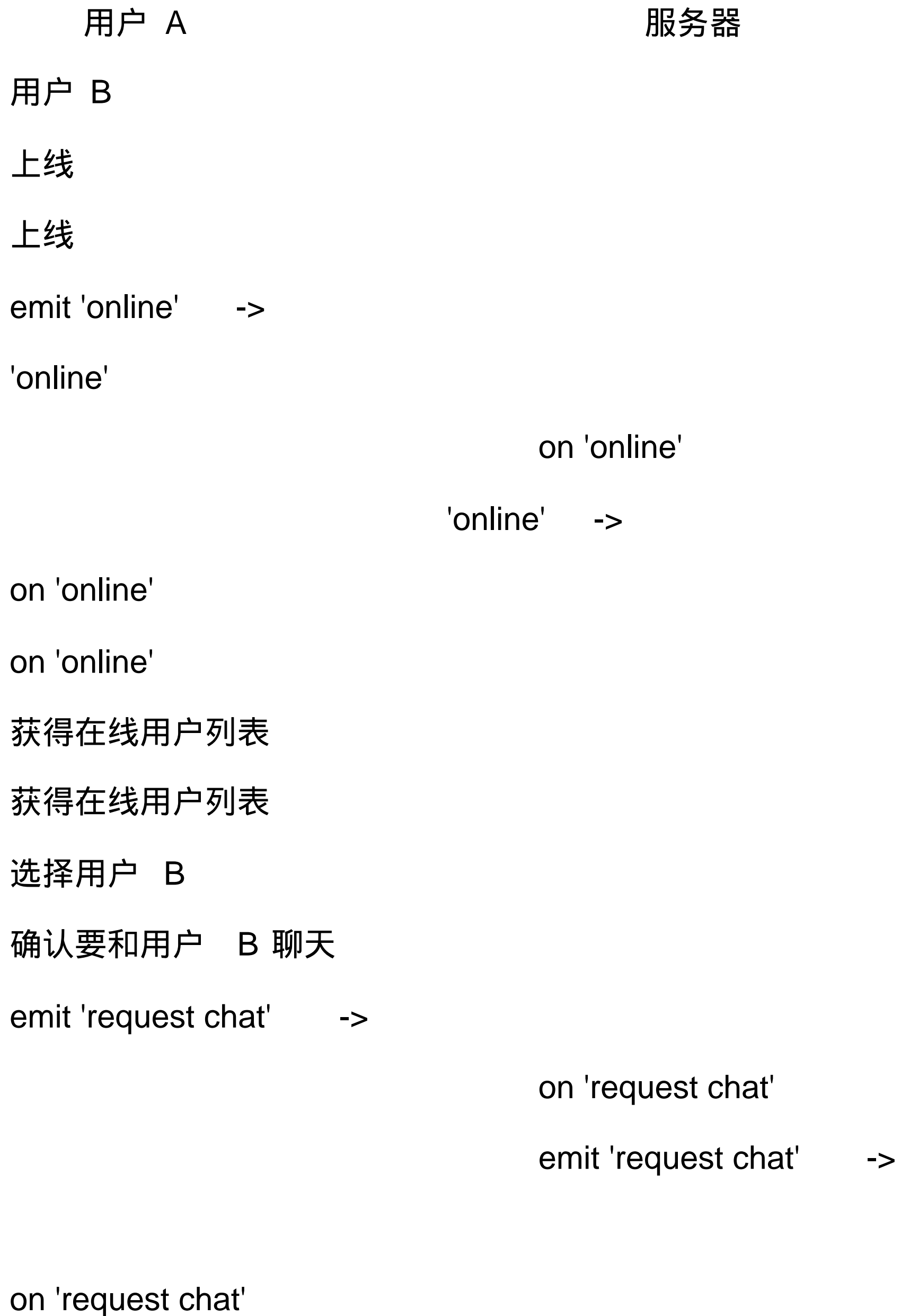


WebRTC 工作流程



getUserMedia()

'stream ok'

on 'stream ok'

'stream ok'

on 'stream ok'

getUserMedia()

emit 'stream ok' ->

on 'stream ok'

emit 'stream ok' ->

on 'stream ok'

createPeerConnection()

pc.createOffer()

pc.setLocalDescription()

'offer'

on 'offer'

'offer'

```
on 'offer'
```

```
createPeerConnection()
```

```
pc.setRemoteDescription
```

```
pc.createAnswer()
```

```
pc.setLocalDescription()
```

```
emit 'answer' ->
```

```
on 'answer'
```

```
emit 'answer' ->
```

```
on 'answer'
```

pc.setRemoteDescription() 这样就能进行视频聊天了。 其中

createPeerConnection 过程如下：全选 <button

```
href="javascript:void(0);" class="copyCode btn btn-xs"
```

```
data-clipboard-text="pc = new
```

```
RTCPeerConnection(config)
```

```
// 向 pc 中加入需要发送的流
```

```
pc.addStream(localStream)
```

```
// onicecandidate 处理器会在网络候选可用的时候调用。
```

```
pc.onicecandidate = function (event) {
```

```
    if (event.candidate) {
```

```
        socket.emit('candidate', requestSocketId, {
```

```
        type: 'candidate',
        label: event.candidate.sdpMLineIndex,
        id: event.candidate.sdpMid,
        candidate: event.candidate.candidate
    });
} else {
    console.log('End of candidates.');
```

```
};
```

// 如果检测到流媒体流到本地，就把它显示出来，同时把流赋值给 remoteStream

```
pc.onaddstream = function (event) {
    attachMediaStream(remoteVideo, event.stream);
    remoteStream = event.stream;
};
```

```
" data-toggle="tooltip" data-placement="top" title=""
style="color: rgb(255, 255, 255); font-style: inherit;
font-variant: inherit; font-stretch: inherit; font-size: 12px;
line-height: 1.5; font-family: inherit; margin: 0px 0px 0px
5px; overflow: visible; cursor: pointer; vertical-align: middle;
border: 1px solid transparent; white-space: nowrap;
padding-right: 5px; padding-left: 5px; border-radius: 3px;
```

```
-webkit-user-select: none; box-shadow: rgba(0, 0, 0, 0.2)
0px 1px 2px; background-image: none; background-color:
rgba(0, 0, 0, 0.74902);"> 复制放进笔记  pc = new
RTCPeerConnection(config)

// 向 pc 中加入需要发送的流
pc.addStream(localStream)

// onicecandidate 处理器会在网络候选可用的时候调用。
pc.onicecandidate = function (event) {
    if (event.candidate) {
        socket.emit('candidate', requestSocketId, {
            type: 'candidate',
            label: event.candidate.sdpMLineIndex,
            id: event.candidate.sdpMid,
            candidate: event.candidate.candidate
        });
    } else {
        console.log('End of candidates.');
```

```
attachMediaStream(remoteVideo, event.stream);  
  
remoteStream = event.stream;  
  
};
```