

Pepperディスプレイ のちよつとしたコツ

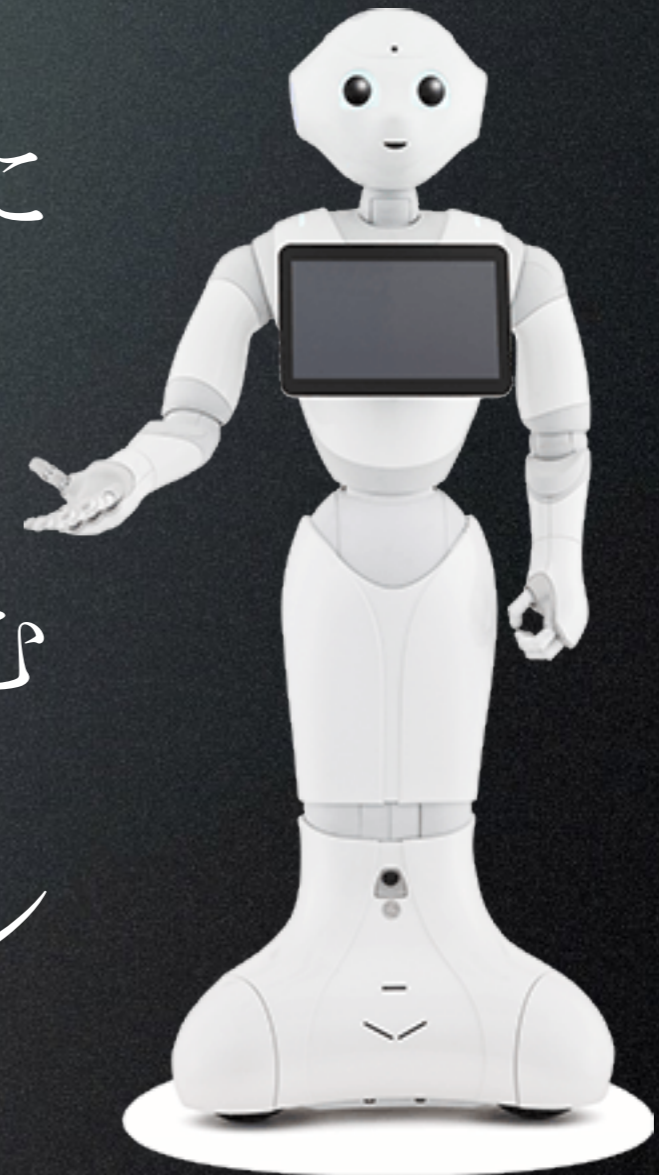


Headwaters Co., Ltd.

うちではこんな感じで
やっています

ディスプレイ開発の流れ

- PC上でタブレット画面に表示するHTMLを作成
- Choregrapheに取り込む
- Pepperのアニメーション動作と合わせて調整



デバッグ

- PCブラウザからタブレットに接続

`http://[pepperIP]/apps/[appId]`

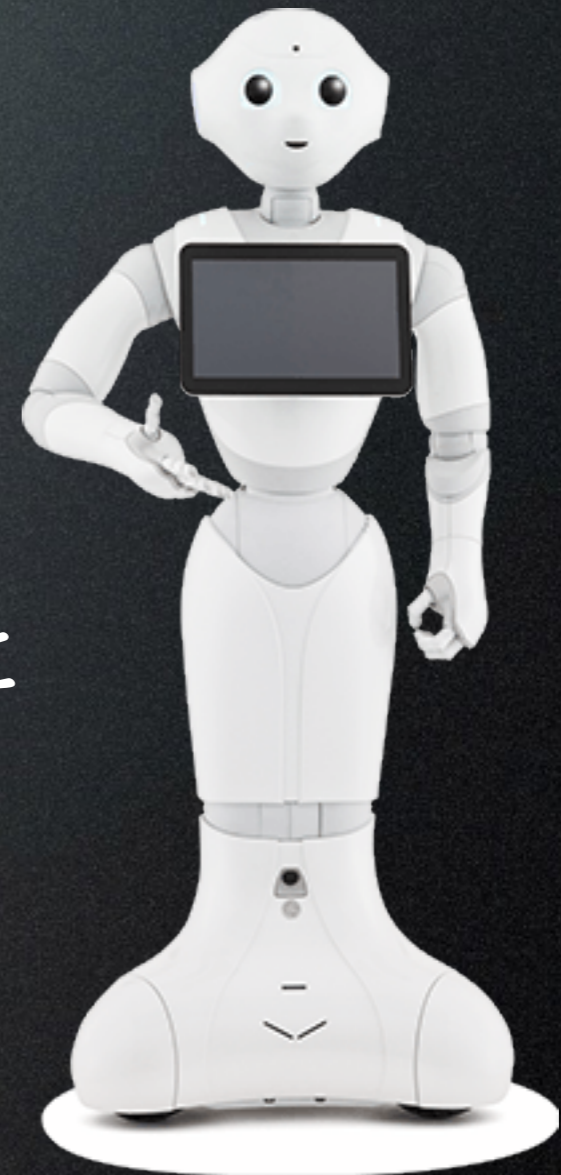
- Chromeのデベロッパーツールを利用

- Logcatも活用

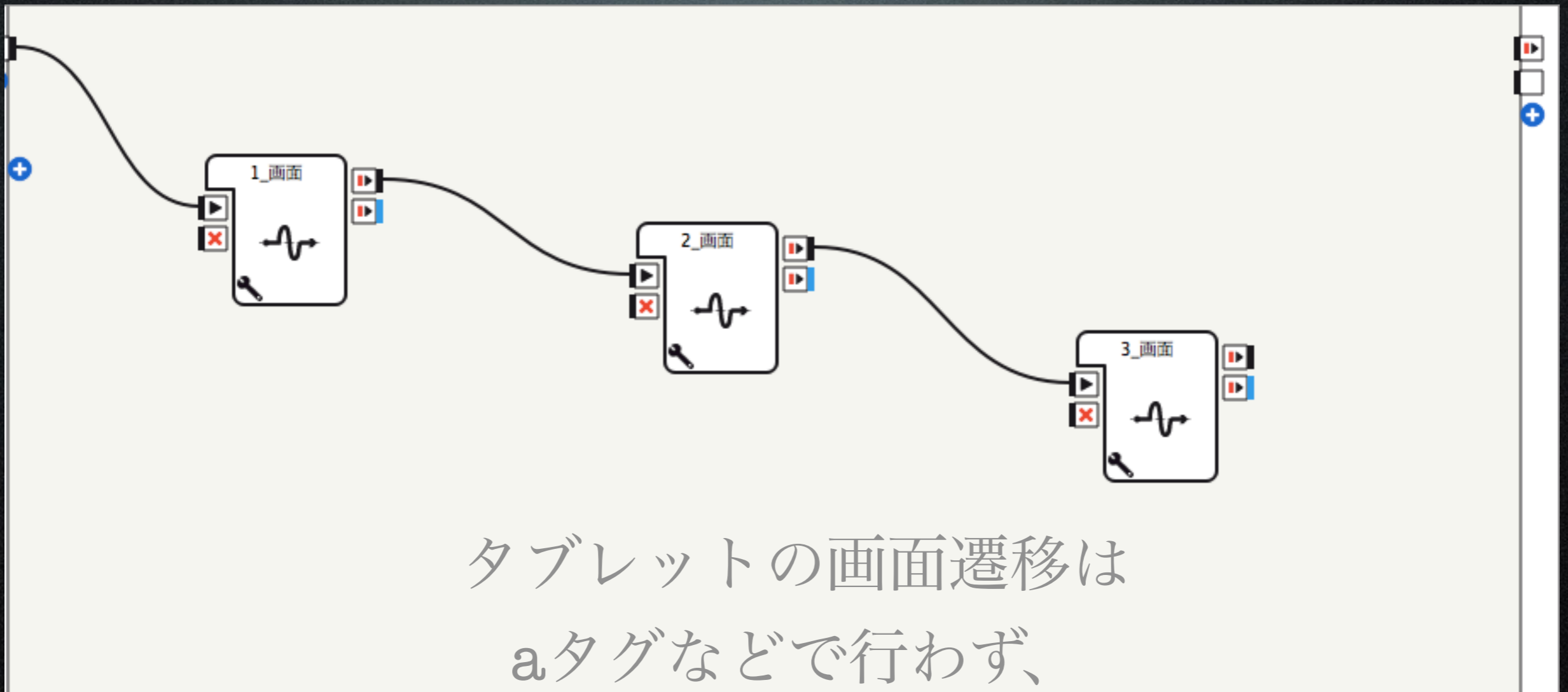
`~/.local/share/PackageManager/apps/j-tablet-browser/adb logcat`

設計のポイント

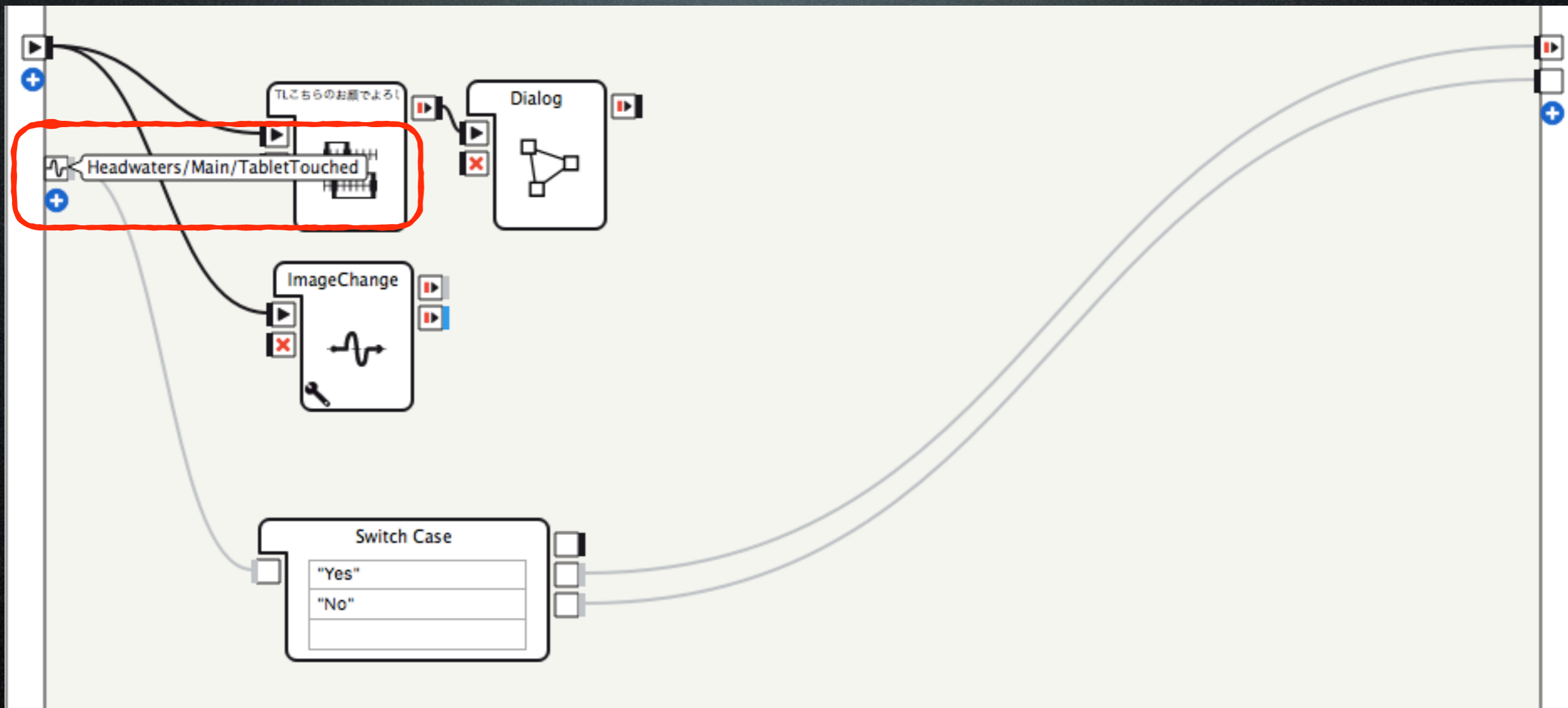
- Pepper側を主体として
タブレットを操作すること
を意識する



つまり？



Pepper側からRaiseEventで操作する

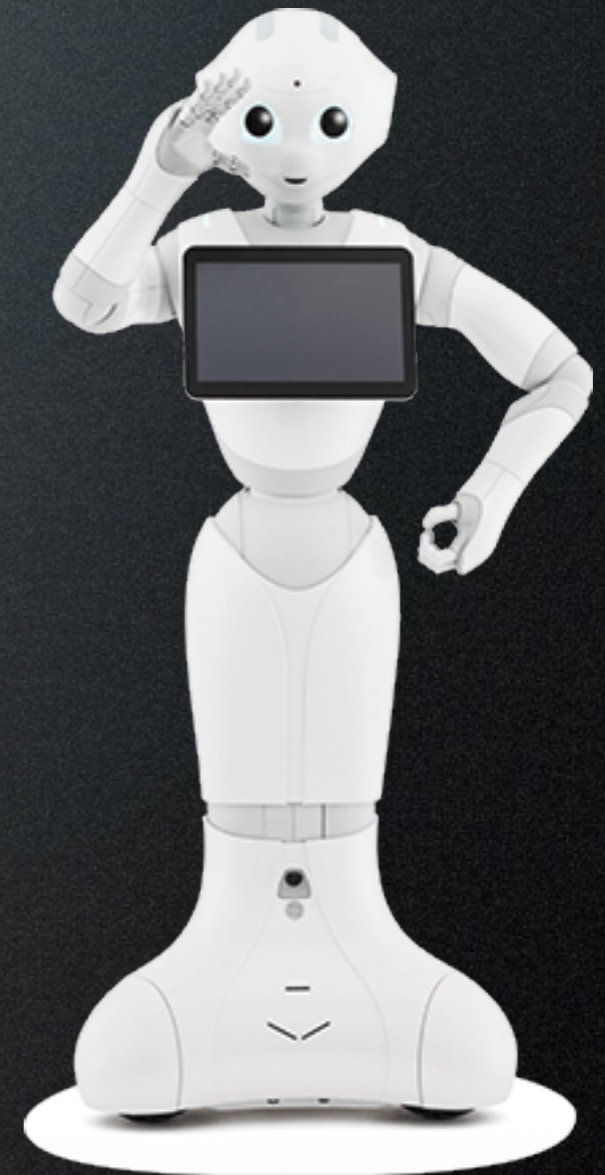


Htmlのボタンなどのタッチイベントも、
必ずPepper側の処理を通すようにする

实例

求められること

- 目を引く画像や動画
- リッチなアニメーション
- Pepperの動作との
自然な連携



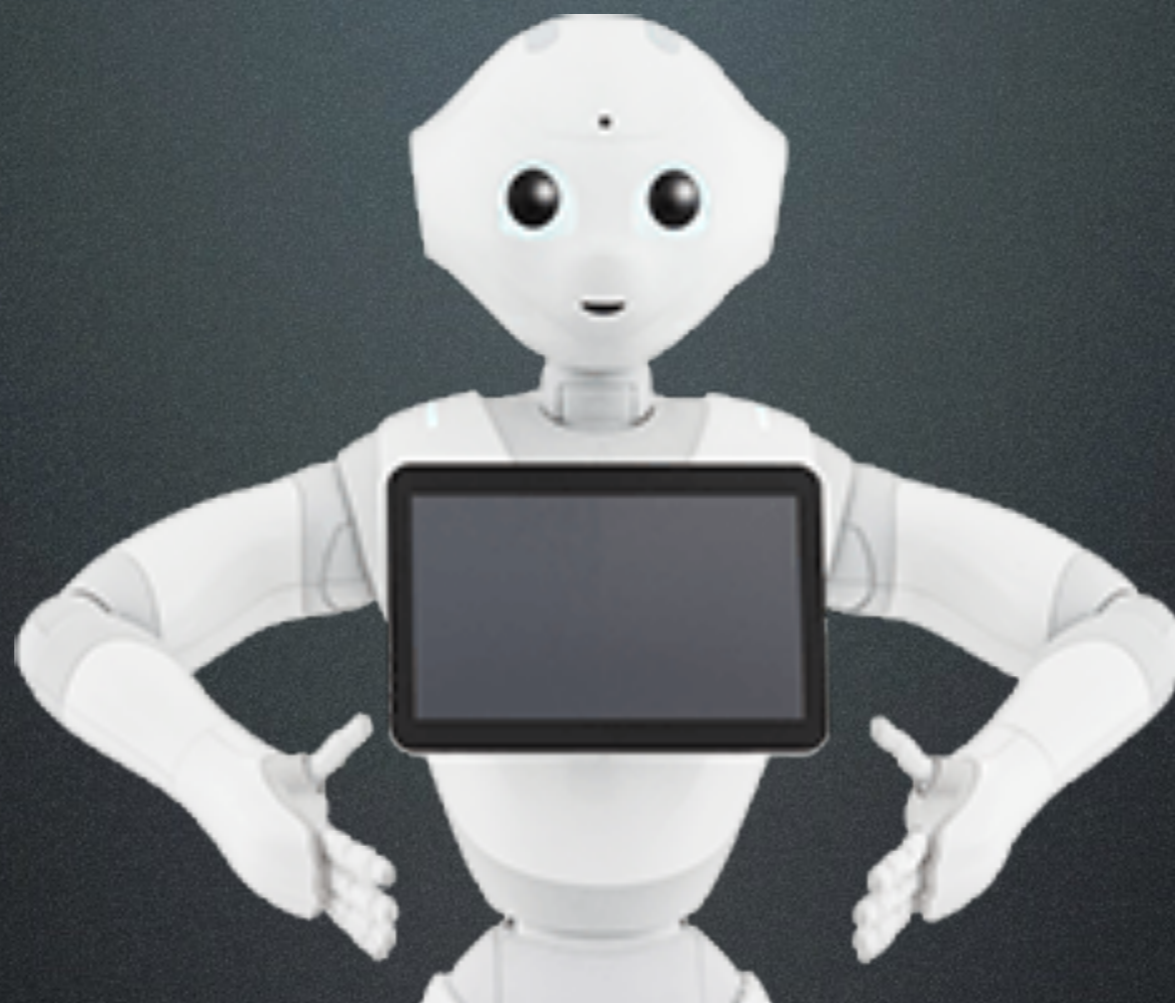
悩み

- 画面、画像の切り替えが遅い
- アニメーションが遅い、カクカクする
- タブレットの処理が重くなり、Pepperの動作とズレる
- WiFiが遅いことがある

どうすれば？

画面、画像の切り替えが遅い

- 画面遷移の前に白画面が表示されてしまい、見た目が悪くなる
- 画像の読み込みでチラツキが発生する



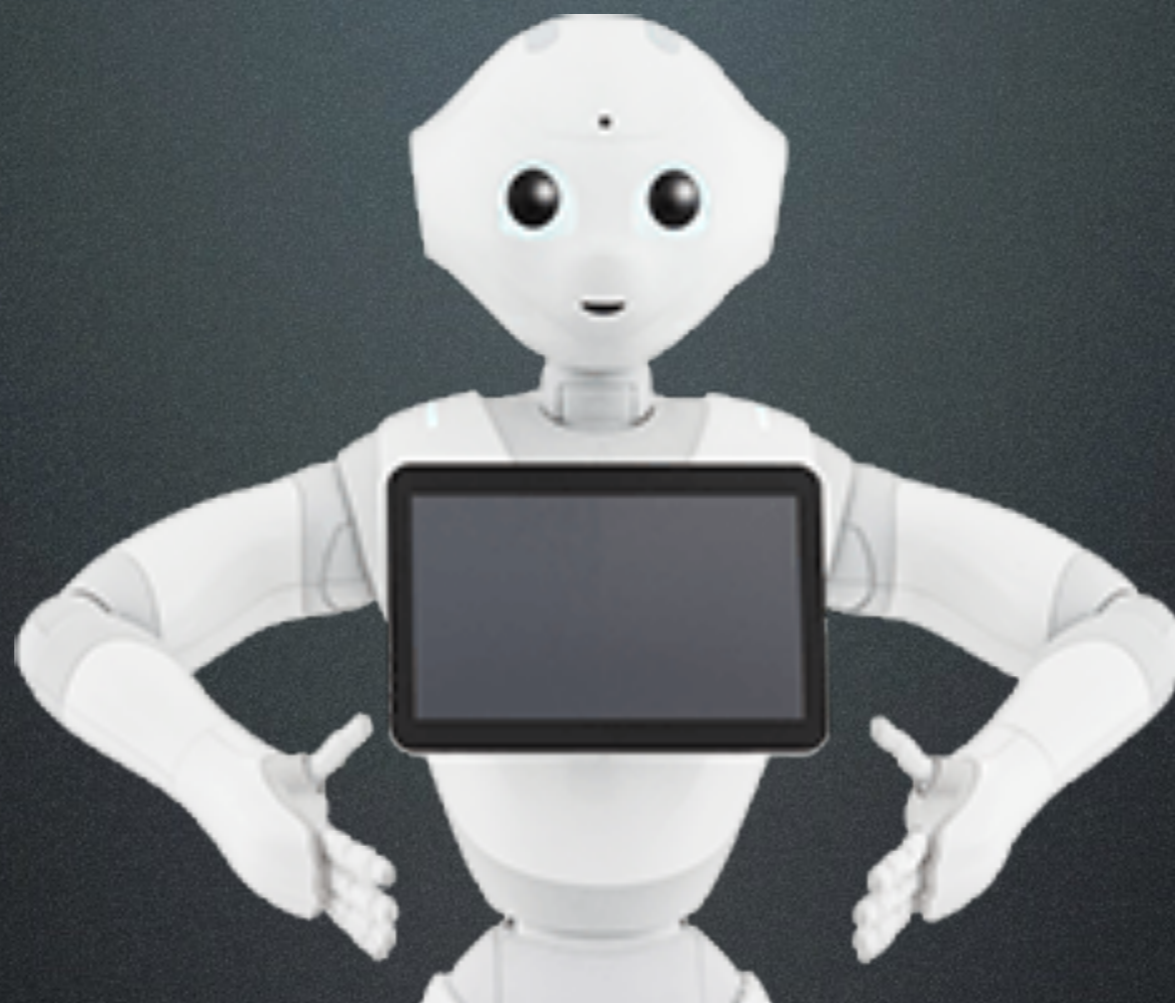
かいけつ！！

画面、画像の切り替えが遅い

- Htmlそのものを読み込まず、変更が必要な箇所のみ切り替える
- 画像のプリロード
(`ALTabletService::preLoadImage`)

アニメーションが遅い、 カクカクする

- UFOが飛ぶアニメーションをjQueryのanimateを使ってやろうとしたら、凄くカクカクしてしまった
- 結果画面でキラキラするエフェクトを表示しようとしたら、凄くカクカクしてしまった



かいけつ！！

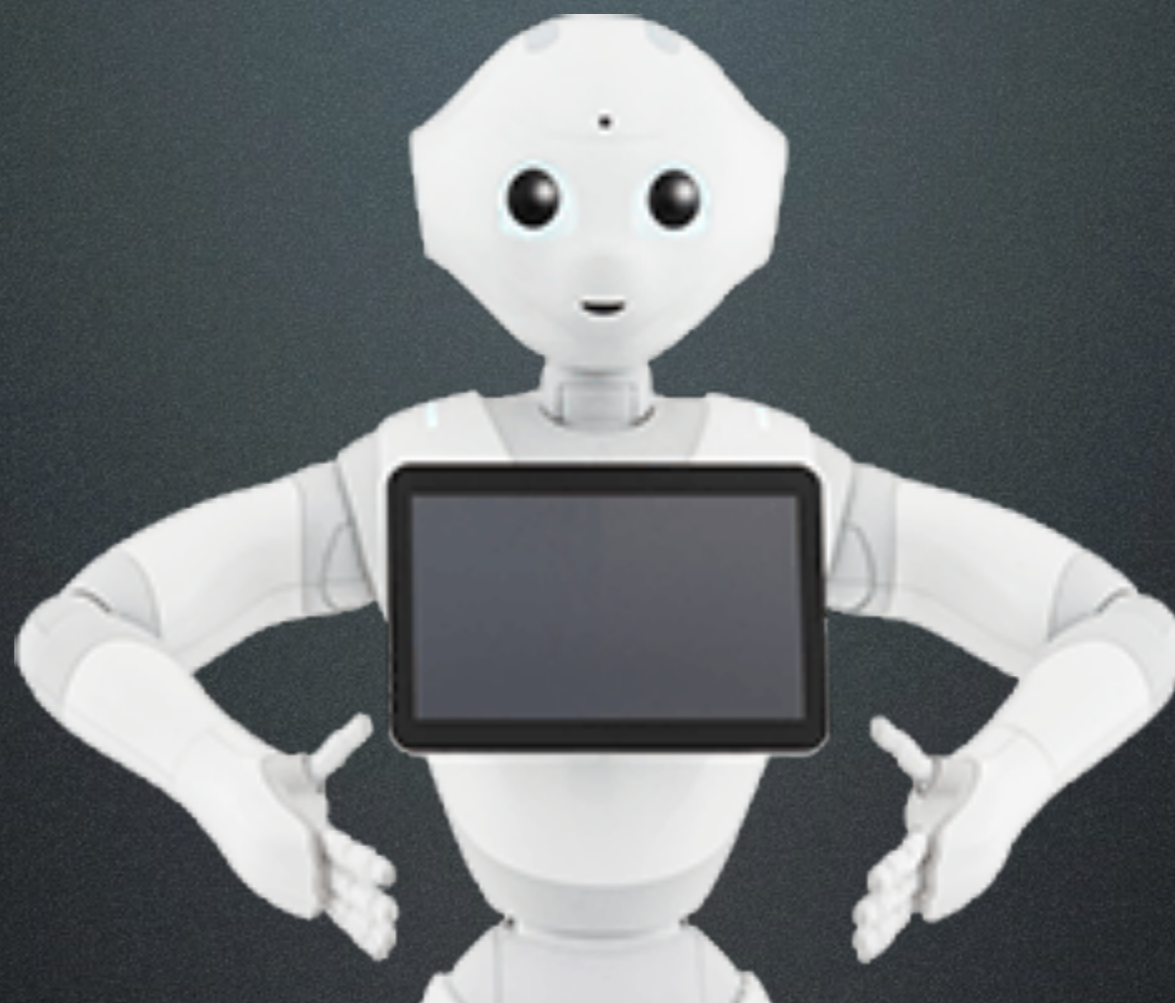
アニメーションが遅い、 カクカクする

- シンプルなアニメーションを心がける
- transision: transformを利用する
- アニメーションさせる要素の数を減らす(3個まで)

タブレットの処理が重くなり、

Pepperの動作とズレる

- 聞き取りの後にボタンが押せたり、
ボタンを押した後に聞き取ったり
- 操作しているうちにタブレットと
Pepperのアニメーションがズレてしま
う



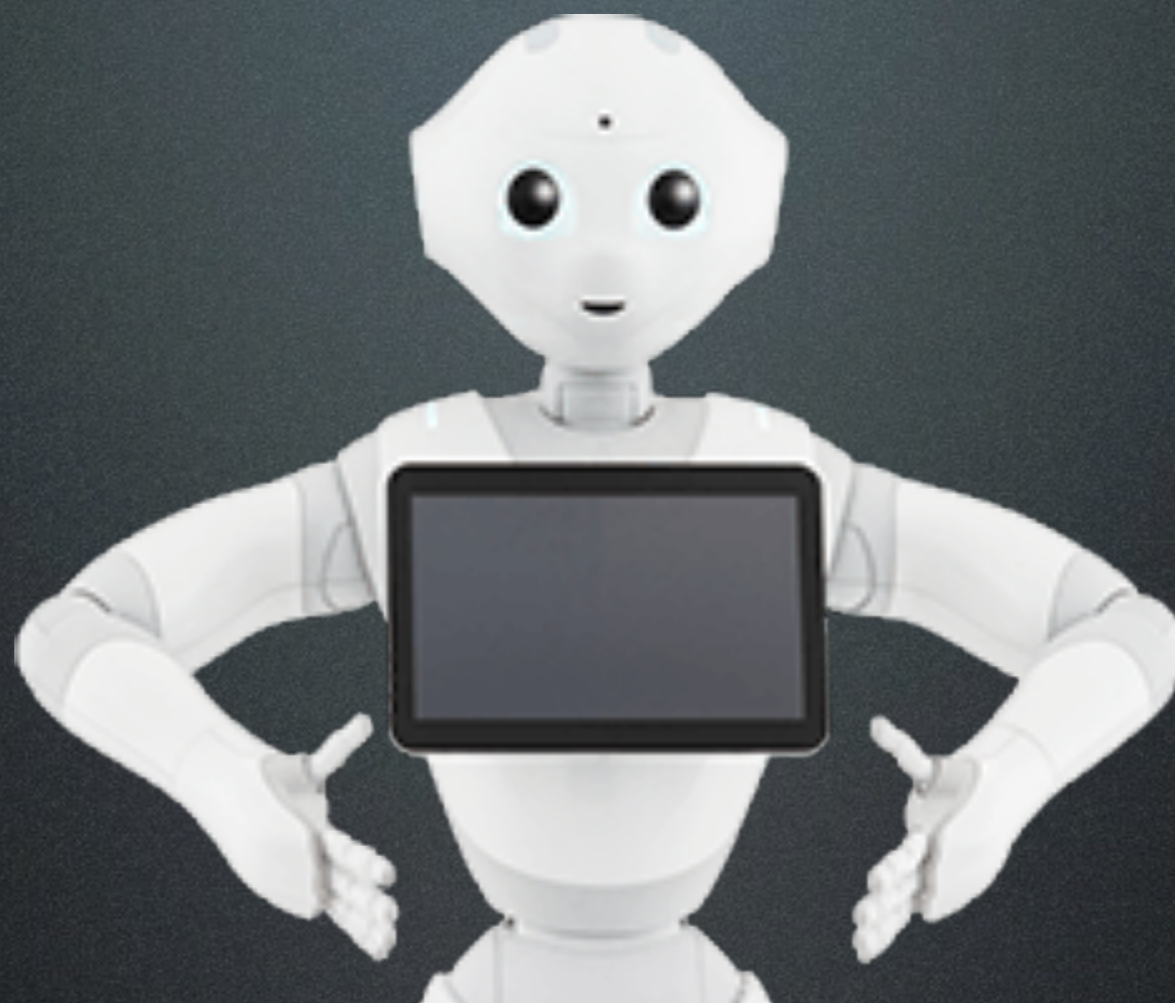
かいけつ！！

タブレットの処理が重くなり、 Pepperの動作とズレる

- 画面遷移やタブレットからのイベント
処理はPepper側で制御する

WiFiが遅いことがある

- Pepper側と比較してタブレット側のWiFiが不安定になりやすい傾向
- タブレット側のWiFiは設定されていないこともある



かいけつ！！

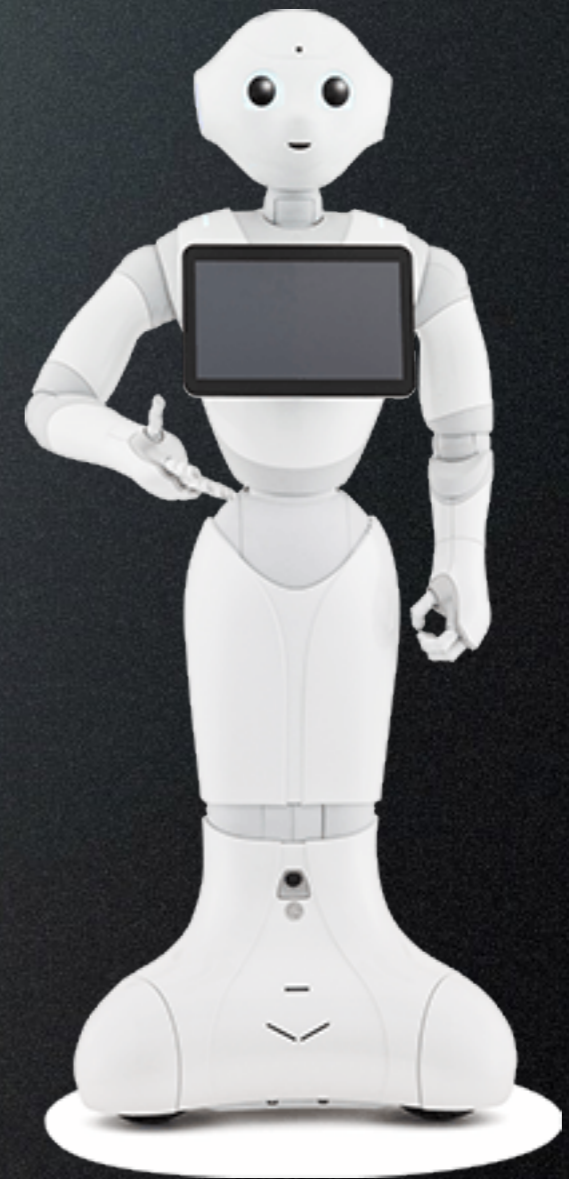
WiFiが遅いことがある

- Pepper側で画像をダウンロードして、タブレット側はローカルから読み込む
- タブレットのWiFi設定情報を取得して処理を分岐させる

```
qicli call ALTabletService.getWifiStatus
```


Tips

- 解像度の問題
- タブレットが反応しない



タブレットの解像度

- 現在のPepperタブレットの解像度は
1707 * 1067
- 1280 * 800を1.335倍に引き伸ばして
使用するのがオススメ

```
<meta name="viewport" content="initial-scale = 1.335,  
minimum-scale = 1.335, maximum-scale = 1.335"/>
```

タブレットが反応しない

- 通常は、`nao restart`で解決
- それでもダメなら胸のボタンで再起動
- それでもダメなら . . .

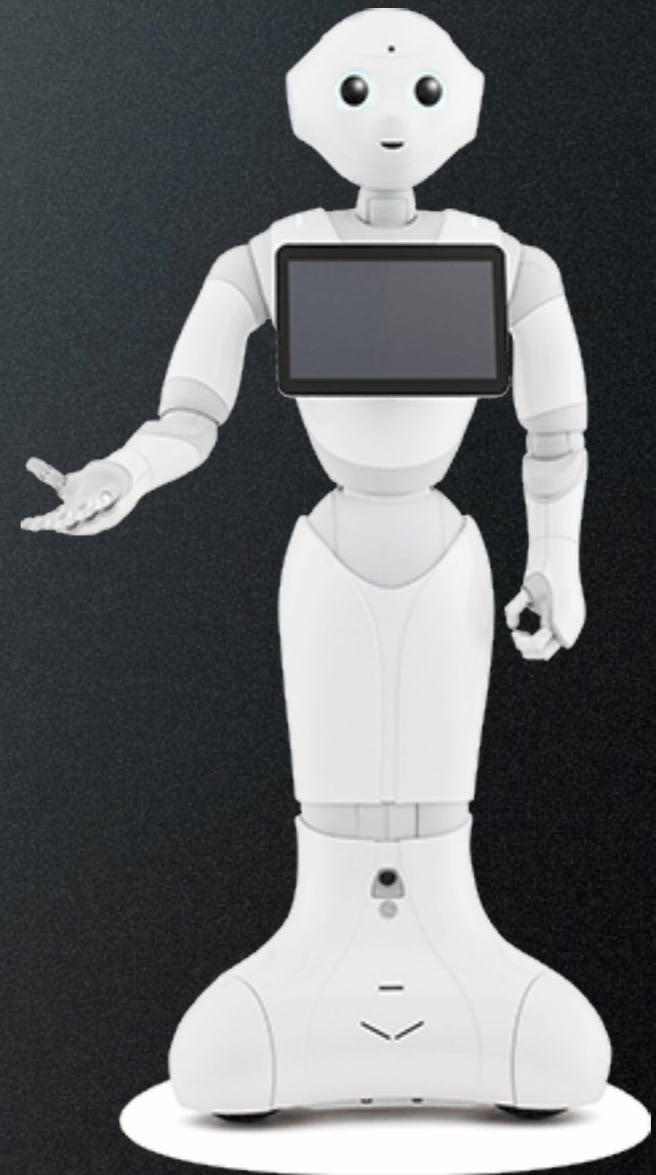
Resetボタン



最後の手段として
ご利用ください

Frameworkの紹介

- Tapper
- QiTalk



Tapperとは？

- Pepperタブレット開発の雛形生成ツール
- QiMessaging JavaScript, jQueryとALProxyを連携
- <https://github.com/pepper-dev/tapper>

Tapperとは？

```
tapper create
```

コマンドで雛形作成

```
Tapper create.
```

```
Scene 1
```

```
mkdir: ./html
```

```
mkdir: ./html/css
```

```
mkdir: ./html/js
```

```
mkdir: ./html/img
```

```
mkdir: ./html/img/preloads
```

```
create: ./html/index.html
```

```
create: ./html/css/contents.css
```

```
create: ./html/css/normalize.css
```

```
create: ./html/js/tapper.js
```

```
create: ./html/js/contents.js
```

```
Succeeded.
```


Tapperとは？

index.html

```
</head>
<body>
  <section id="scene1" style="display: block;">
  </section>
  <section id="scene2" style="display: block;">
  </section>
</body>
</html>
```

show/hideで切り替え

// javascriptでコールする場合

```
Tapper.view.changeScene(1);
```

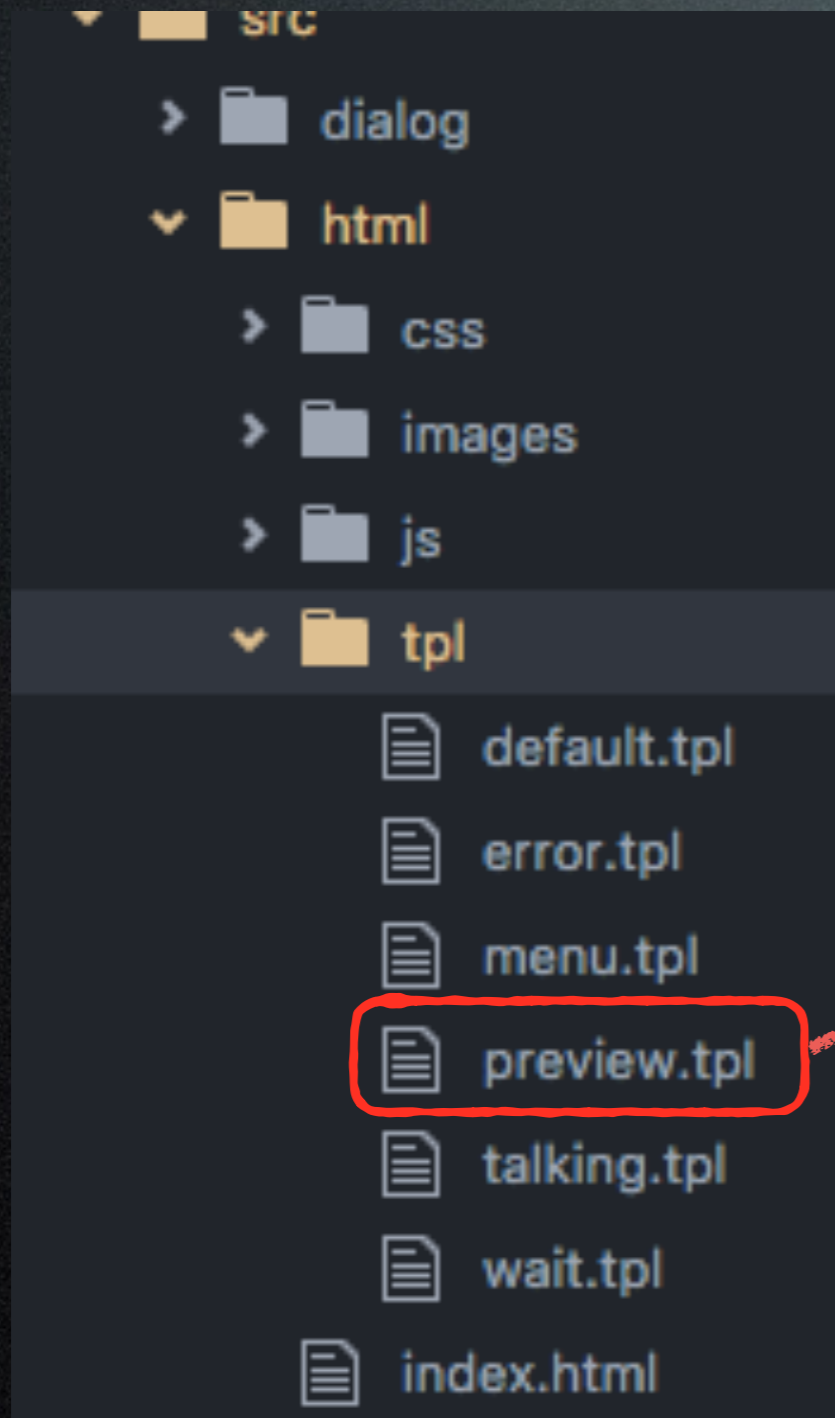
// ALMemoryのイベントでシーンの切り替えを行う場合

```
Tapper.utl.raiseEvent("Tapper/View/ChangeScene", 1);
```

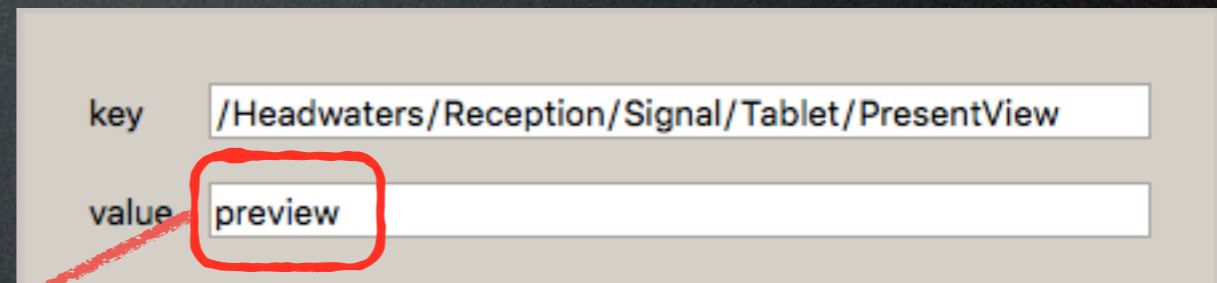
QiTalkとは？

- QiMessaging JavaScript, jQuery と ALProxy を連携
- Pepperからの呼び出しイベント毎に対応したページ設計が可能
- <https://github.com/hikouki/qitalk>

QiTalkとは？



ALMemoryのvalue値と
ファイル名でテンプレートの関連付け



QiTalkとは？

preview.tpl

```
<script>
$(function() {
  $('#filter').hide();
  Qitalk.proxy.ALMemory.getData('Headwaters/Reception/TakePicture').then(function(str) {
    $('#pictureImg').attr('src', 'images/' + str);
  });
  $("#business-confirm--contents-confirm--yes").on('touchend click', function() {
    Qitalk.send("Headwaters/Main/TabletTouched", "Yes");
  });
  $("#business-confirm--contents-confirm--no").on('touchend click', function() {
    Qitalk.send("Headwaters/Main/TabletTouched", "No");
  });
});
</script>
<div id="content" class="page">
  <div id="confirmPicture">
    <div id="pictureImgArea">
      <img id="pictureImg" src="" />
    </div>
    <div id="business-confirm--contents-confirm">
      <div id="business-confirm--contents-confirm--yes" class="se_click push_btn"></div>
      <div id="business-confirm--contents-confirm--no" class="se_click push_btn"></div>
    </div>
  </div>
</div>
```

指定されたテンプレートに切り替え

ディスプレイ開発に幸あれ

