

# EMBED WRAPPER: TWO PARTS

```
class EmbedWrapper {  
  
  markPageAsLoaded() {  
    window.postMessage('pageLoaded');  
  }  
  
  onScroll(callback) {  
    window.addEventListener('message', evt => {  
      const scrollPosCheck = evt.data.match(/scrollPos:(.+)/);  
      if (scrollPosCheck) {  
        callback(scrollPosCheck[1]);  
      }  
    });  
  }  
}
```

**Embed wrapper (inside iframe)**

**Host page**

```
// wrapper helper - can go inline in host page  
window.addEventListener('scroll', pos => {  
  window.postMessage(`scrollPos:${pos}`);  
});  
  
window.addEventListener('message', msg => {  
  if (msg.data === 'pageLoaded') {  
    removeLoadingSpinner();  
  }  
});
```

### STANDALONE WRAPPER

### EMBED WRAPPER

```
class StandaloneWrapper {  
  
  markPageAsLoaded() {  
    removeLoadingSpinner();  
  }  
  
  onScroll(callback) {  
    window.addEventListener('scroll', callback);  
  }  
  
}
```

```
// wrapper helper - can go inline in host page  
window.addEventListener('scroll', pos => {  
  window.postMessage(`scrollPos:${pos}`);  
});  
  
window.addEventListener('message', msg => {  
  if (msg.data === 'pageLoaded') {  
    removeLoadingSpinner();  
  }  
});
```

```
class EmbedWrapper {  
  
  markPageAsLoaded() {  
    window.postMessage('pageLoaded');  
  }  
  
  onScroll(callback) {  
    window.addEventListener('message', evt => {  
      const scrollPosCheck = evt.data.match(/scrollPos:(.+)/);  
      if (scrollPosCheck) {  
        callback(scrollPosCheck[1]);  
      }  
    });  
  }  
  
}
```