エコーネットライト オンライン制御デモアプリ  
設計書

v2.1

インターネット・ビジネス・ジャパン株式会社

目次

[コンセプト 3](#_Toc62808135)

[本アプリとWeb UI間の通信 3](#_Toc62808136)

[開発フレームワーク 3](#_Toc62808137)

[エコーネットライトサーバー（UDP通信）の処理 4](#_Toc62808138)

[エコーネットライト電文の処理 4](#_Toc62808139)

[自ノードのエコーネットオブジェクト 5](#_Toc62808140)

[Web UIサーバー（HTTP通信、WebSocket通信）の処理 6](#_Toc62808141)

[Web UIのソースコード 6](#_Toc62808142)

[Web UIのステート管理 6](#_Toc62808143)

[Web UIのビルド 7](#_Toc62808144)

## コンセプト

下記のユースケースを踏まえ、「エコーネットライトコントローラーとしての処理」と「GUIに係る処理」を、それぞれ「本アプリ」（バックエンド）と「Web UI」（フロントエンド）に関心分離した。

* 本アプリを起動するマシン（サーバーマシン）と、そのWeb UIにアクセスするマシン（クライアントマシン）が異なるケース。
* 複数のWeb UIアクセスが同時発生するケース。
* Web UIにアクセスするWebブラウザーのJavaScript処理性能が低いケース。

## 本アプリとWeb UI間の通信

|  |  |
| --- | --- |
| 通信方式 | 通信内容 |
| HTTP通信（Vue Router） | Web UIの画面表示 |
| WebSocket通信 | 設定データやエコーネットライトデータの送受信 |

## 

## 開発フレームワーク

|  |  |
| --- | --- |
| フレームワーク | 主要用途 |
| npm | 依存パッケージ管理。 |
| Node.js、Express | エコーネットライトサーバー、Web UIサーバーのコアライブラリー。 |
| ws | WebSocketサーバーのコアライブラリー。 |
| Vue CLI | Vue.jsを中心としたフロントエンド開発環境セットアップ。 |
| connect-history-api-fallback | SPA向けのWebサーバールーティング支援ライブラリー。 |
| Axios | Ajax通信支援ライブラリー。 |

## 

## エコーネットライトサーバー（UDP通信）の処理

app.js の els モジュールがその責任を担う。

|  |  |  |
| --- | --- | --- |
| メソッド | 返り値 | 機能 |
| start() | void | エコーネットライト通信のためのUDPソケット作成とNICバインド。 |
| stop() | void | UDPソケットの終了。 |
| restart() | void | UDPソケットの再作成とNICバインド。 |
| send(ip: string, msg: Buffer) | void | 指定したip（IPアドレス）に向けてmsg（バイト列）をUDP送信。 |
| multicast(msg: Buffer) | void | エコーネットのマルチキャストアドレスに向けてmsg（バイト列）をUDP送信。 |
| notifyUI(dir: string, ip: string, msg: Buffer) | void | エコーネットライト通信の内容をWeb UIにPUSH通知。 |

## 

## エコーネットライト電文の処理

app.js の el モジュールがその責任を担う。

|  |  |  |
| --- | --- | --- |
| メソッド | 返り値 | 機能 |
| isELFrame(msg: Buffer) | bool | msg（バイト列）がエコーネットライト電文の書式に合致しているか否かを返す。 |
| parse(msg: Buffer) | object | null | msg（バイト列）をエコーネットライト電文としてパースし、後処理しやすいオブジェクトにして返す。エラー等で破棄された場合は、nullを返す。 |
| serialize(data: object) | Buffer | null | data（パース済み電文）をエコーネットライト電文（バイト列）にシリアライズして返す。エラー等で破棄された場合は、nullを返す。 |
| respond(data: object) | Buffer | null | data（パース済み電文）の内容を確認し、応答用のエコーネットライト電文を返す。応答不要であればnullを返す。 |
| getMyDeviceList() | Buffer | 自ノードの機器オブジェクト一覧をマルチキャスト通知するためのエコーネットライト電文（バイト列）を返す。 |
| searchAllNodes() | Buffer | null | ネットワーク上の全ノードを検索するためのエコーネットライト電文（バイト列）を返す。 |

## 自ノードのエコーネットオブジェクト

app.json の app.el.self プロパティーに設定。下記は設定例。

"node": {  
 "0x0EF0": {  
 "0x01": {  
 "0x80": ["0x30"],  
 "0x82": ["0x01", "0x0C", "0x01", "0x00"],  
 "0x83": **null**,  
 "0x8A": ["0x00", "0x00", "0x77"],  
 "0x9D": ["0x01", "0x80"],  
 "0x9E": ["0x00"],  
 "0x9F": ["0x0C", "0x80", "0x82", "0x83", "0x8A", "0x9D", "0x9E", "0x9F", "0xD3", "0xD4", "0xD5", "0xD6", "0xD7"],  
 "0xD3": ["0x00", "0x00", "0x03"],  
 "0xD4": ["0x00", "0x02"],  
 "0xD5": ["0x01", "0x05", "0xFF", "0x01"],  
 "0xD6": ["0x01", "0x05", "0xFF", "0x01"],  
 "0xD7": ["0x01", "0x05", "0xFF"]  
 }  
 },  
 "0x05FF": {  
 "0x01": {  
 "0x80": ["0x30"],  
 "0x81": ["0x00"],  
 "0x82": ["0x00", "0x00", "0x4C", "0x00"],  
 "0x83": **null**,  
 "0x88": ["0x42"],  
 "0x8A": ["0x00", "0x00", "0x77"],  
 "0x8C": "ONLINE\_DEMO\_",  
 "0x9D": ["0x02", "0x80", "0x88"],  
 "0x9E": ["0x00"],  
 "0x9F": ["0x0A", "0x80", "0x81", "0x82", "0x83", "0x88", "0x8A", "0x8C", "0x9D", "0x9E", "0x9F"]  
 }  
}

※ 0x83 の値を null に設定すると、本アプリ起動時にランダムに設定される（0xFE000077 + 乱数（13B））。  
※単一の文字列を値に設定すると、UTF-8文字列として処理される。

## Web UIサーバー（HTTP通信、WebSocket通信）の処理

app.js の下記モジュールがその責任を担う。

|  |  |
| --- | --- |
| モジュール | 責任範囲 |
| app | HTTP通信（Vue Router）。 |
| server | Expressサーバーの参照や終了。 |
| wss | WebSocket通信。 |

## 

## Web UIのソースコード

主要ファイルは下記の通り。

|  |  |
| --- | --- |
| パス | 内容 |
| public/index.html | トップページ。SPAなので、HTMLファイルはこの1ファイルのみ。 |
| public/\* | その他の静的ファイル。 |
| src/shims-vue.d.ts | TypeScriptによる型宣言。 |
| src/App.vue | 最上位のVueコンポーネント。最外殻のレイアウトもここで定義。 |
| src/components, src/views | 各パーツや画面ごとのVueコンポーネント。 |
| src/router/index.ts | Vue Router（ルーティング管理）の定義。 |
| src/store/index.ts | Vuex（ステート管理）の定義。 |

## 

## Web UIのステート管理

Web UI上のデータ管理は、Vuexのステート管理機構に依る。el-demoapp/src/store/index.ts がそのメインファイル。

|  |  |  |
| --- | --- | --- |
| 主なデータ | データ型 | 内容 |
| state.settings | object | Web UIの設定。設定画面の各値はここから取得する。 |
| state.text | object | Web UIで使用する全テキスト。別途指定した言語分のみ。 |
| state.log | Array | エコーネットライト通信のログ。 |
| state.nodes | object | 通信した全ELノードのEDT等のキャッシュ。 |

## Web UIのビルド

基本的には、el-demoapp/public 内の静的ファイル群と el-demoapp/src 内のVue.jsファイル群を、リント、コンパイル、圧縮し、公開可能な状態にすること。

npm run build

※ビルド成果物は el-demoapp/dist フォルダー内に出力される。