

# 小型・低廉デバイスを用いた自律統合型デジタルサイネージシステムの実現

代表研究者      三 島 和 宏      東京農工大学総合情報メディアセンター 助教  
共同研究者      櫻 田 武 嗣      東京農工大学総合情報メディアセンター 講師

## 1 本研究開発の背景と目的

本研究開発では、Raspberry Pi などに代表される小型かつ低廉な情報デバイスを利用したデジタルサイネージシステムと、これらデバイスを複数運用する際に必要となる「集約管理」を可能とする統合型デジタルサイネージシステムの実現を低コストにて目指す。

近年、数千円程度で入手可能な情報デバイスが性能面でも高度化してきたことから、センサーシステムや簡易的な計算機システムの一部として広く利用されるようになってきた。このようなデバイスを公共エリア等で多様な情報を自律的に提供する「デジタルサイネージシステム」に応用することで、これまでバックエンドとして PC や専用機等を必要としてきたサイネージシステムの「低価格化」と「低消費電力化」をより進めることを可能とする。

従来のデジタルサイネージシステムでは、コストを対価にして導入した専用システムを利用することで複数台の表示装置の集約管理を実現している。本研究開発では、ネットワークを通じて小型・低廉デバイスを制御し、集約管理型の表示装置として利用できる仕組みを、オープンシステムとして「低コスト」にて実現する。本システムにより、表示装置自体の低コスト化に加え、表示装置導入時の構築コスト低減と運用管理時のコスト低減までも図ることが可能となる。本システムにより、これまでの専用システムに依らない手軽なデジタルサイネージシステムを実現し、デジタルサイネージの一般化をより目指す。

## 2 本研究開発を取り巻く環境

本学では、電力の省エネルギー化を目指すべく、早期より「電力の見える化」に積極的に取り組んできた。これに関連して、全学規模での「受電電力量を計測する設備」、「計測したデータを蓄積する設備」、ならびに、「計測データの表示を行う装置（既存表示システム）」を導入し、活用を行っている。右図は、既存表示システムにて電力量を表示した例である。これら表示装置は全学に複数設置されている。既存表示システムでは、表示装置に通常の PC を利用する形態となっており、通常の PC であるがゆえに、機器自体の消費電力や OS のサポート終了等、長期にわたり機器を利用することに対するコストが徐々に高くなっていくという課題の対処に追われる状況となっている。この課題がテーマ発想の原点となった。

これに加えて、電力見える化以外の場面でも、学内でのデジタルサイネージ利用に対する要望（本館等での教育研究紹介ビデオの定時表示など）も高くなってきたことから、今回応募を行っている「小型かつ低廉な情報デバイスを利用したデジタルサイネージシステム」のプロトタイプとなるシステムを設計・構築し、運用を開始している。本プロトタイプシステムでは、端末ごとに構築・設定を行い単独での動作を前提とした構成となっている。今後、表示装置の置き換えを視野に入れ、提供規模の拡大を想定した場合、表示端末に対する設定の簡素化・自動化（ゼロコンフィグレーション）と複数台の表示端末の集約管理（デバイスマネージメント）を行う機能が必須の状況となっている。

デジタルサイネージシステムは、計算機の高度化と小型化に伴い、動画や静止画を自由に表示可能な機器を壁面などに容易に組み込めるようになったことから、研究ベースに加えてビジネスベースでの開発も活発化している。今日、デジタルサイネージシステムは、駅や公共エリアの施設等で広く利用されるようになってきており、一般ユーザの目に付くことも多くなってきた。デジタルサイネージシステムに関連した研究では、いわゆるシステム自体に関連した取り組みの他、ロケーションをベースとしたもの[1]やユーザーインタラクションを取り入れたもの[2]など表示方法に関連した応用研究などがある。

この状況下において、パッケージングされた商用製品も多く存在しており、このようなデジタルサイネージシステムとして、パナソニック社製の NMstage[3]や三菱電機社製の MEDIAWAY[4]、トレインビジョン[5]などがある。トレインビジョンは、JR などの鉄道網での案内表示に利用されている。また、小型かつ低廉な情報デバイスを利用したシステムとして、Screenly[6]が挙げられる。[6]についても、全ての機能を利用す

る場合、導入コストが発生する商用のパッケージソフトウェアとなっている。

本研究開発では、導入・運用に対する「機器コスト」、「構築コスト」、「運用コスト」を考慮し、小型かつ低廉な情報デバイスを利用したデジタルサイネージシステムならびに低コストにて利用可能な集約管理システムをターゲットとする。本ターゲットにおける事例として、フリーソフトウェアの Raspberry Digital Signage[7]がある。しかし、[7]は、単独での構築・運用を想定したシステムであるため、複数の表示装置を集約的に管理することができない。このように、複数の表示装置を安価かつ集約的に管理することを想定した場合、コスト面からユーザが容易に利用可能なデジタルサイネージシステムは非常に限定的なものになってしまう。このため、本研究開発にて実現を目指すシステムの重要度は非常に高いものであると言える。

### 3 デジタルサイネージデバイスの自律統合を促す集約管理アーキテクチャの開発

#### 3-1 小型・低廉デバイスを用いたデジタルサイネージデバイスの実装

本学では、表示デバイスとして安価なシングルボードコンピュータを利用したデジタルサイネージデバイスの開発を行ってきた。このデジタルサイネージシステムは、今回集約管理を行う対象となるものである。本デジタルサイネージデバイスは、可能な限りコストを低減することを目的とし、数多く販売されるシングルボードコンピュータの中でも、入手の容易さ、性能、価格等のバランスが良い Raspberry Pi を表示用のデバイスとして利用している。Raspberry Pi は、英国ラズベリーパイ財団が開発したシングルボードコンピュータである。CPU プロセッサとして、ARM プロセッサを搭載し、基本機能のみを搭載した「Model-A」、ネットワーク接続等の機能も拡張した「Model-B」の2つのモデルがある。教育用途として容易に取得可能なように Model-A は 20～25 ドル、Model-B は 35 ドル程度で販売されている。OS としては Linux (Raspbian 等のディストリビューションが存在する) を用いることができ、開発の容易さもある。



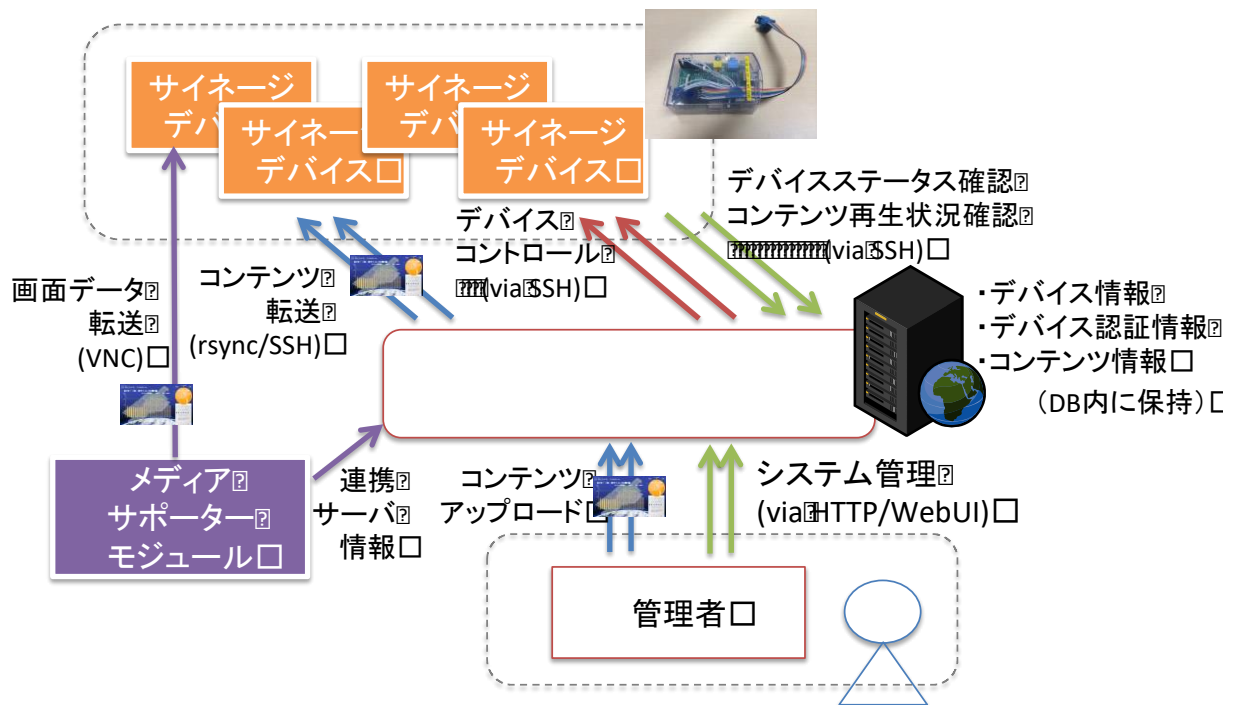
図：Raspberry Pi を利用したサイネージ表示デバイス

本デジタルサイネージデバイスには、サイネージ表示機能として、1) MPEG-4 ビデオ表示機能と 2) Flash ムービーなどのリッチコンテンツ表示機能がある。MPEG-4 ビデオ表示機能は、MPEG-4 形式で生成された動画コンテンツを連続的に再生する機能である。所定のフォルダや設定に基づき、複数の動画コンテンツを順次、継続して表示することも可能となっている。コンテンツの再生には、Raspberry Pi の持つ動画再生支援機能を利用する。動画再生支援機能により安定した動画再生が可能のため、MPEG-4 ビデオの再生時には、Raspberry Pi にて構築した本デバイス単体で動作する。動画の再生は支援機能を利用可能な omxplayer をアプリケーションソフトウェアとして利用するが、複数コンテンツの連続再生は当該アプリケーション単体では実現できないため、omxplayer を拡張する形でこの機能を実装している。Flash ムービーなどのリッチコンテンツを表示する機能は、ビデオ以外の各種コンテンツの再生を行う際に利用されるものである。Raspberry Pi は、安価に入手できる反面、得られる計算機能力はそれほど高いものとは言えない。このため、Flash ムービーなどのブラウザとプラグイン機能を必要とするようなコンテンツの再生は、計算機能力の限界を超えてしまい困難となる。そこで、本デバイスでは、これらのコンテンツ再生のための再生支援機能として、メディアサポーターを併用する形態をとる。メディアサポーターに関しては、後述する。

### 3-2 デジタルサイネージデバイス集約管理アーキテクチャの設計と実装

#### (1) 全体構成

各サイネージデバイスとこれらを集約管理するためのアーキテクチャの設計を行った。以下に本アーキテクチャの全体構成を示す。構成として、まず各サイネージデバイスがある。これは、先述した Raspberry Pi を用いて本学が実装したサイネージデバイスで構成される。次に、各デバイスを集約管理と管理者からの各種設定を可能とする集約管理アーキテクチャがある。これには、モジュールとして、1) デバイスマネージャ・セッションコントローラモジュール、2) メディアサポーターモジュールがそれぞれ含まれる。デバイスマネージャ・セッションコントローラモジュールは管理者からの各種設定を受け付けるため、Web ベースのユーザインタフェース (WebUI) を合わせて搭載する。



図：アーキテクチャの全体構成

#### ● デバイスマネージャ・セッションコントローラモジュール：

各サイネージデバイスの個別設定ならびに各デバイスで表示されるコンテンツの管理を行うためのモジュールである。これらモジュールにはフロントエンドとしての WebUI とバックエンドのデータストレージ (RDBMS) を搭載し各種情報を保持する。

デバイスマネージャでは各デバイスの情報を、セッションコントローラではコンテンツに関する情報をそれぞれ管理する。保持するデータについては以下に詳述するが、デバイスに関しては、固有 ID・ホスト名・IP アドレス (デバイスとの接続に利用する識別子)・デバイス表示名 (各 WebUI で表示する際の名称)・デバイスの運用モード・デバイスの疎通状況といった情報を、コンテンツに関しては、固有 ID・コンテンツ表示名 (各 WebUI で表示する際の名称)・コンテンツタイプ・コンテンツオプションといった情報を持つ。これに加えて、複数コンテンツ (動画コンテンツなど) に対するために、複数のコンテンツをひとつにまとめるためのコンテンツセット情報 (固有 ID・コンテンツセット表示名 (各 WebUI で表示する際の名称)・コンテンツタイプ・コンテンツリスト) も持つこととしている。さらに、各サイネージデバイスとコンテンツセットの関連づけを行い、コンテンツ再生時間等の制御を行うためのアサインメント情報 (固有 ID・コンテンツタイプ・稼働時間帯など) についてもデータストレージ上に保持する。

表：デバイス情報のデータ構造 (SQL)

```

`id` int(11) NOT NULL AUTO_INCREMENT,
`devid` varchar(64) NOT NULL,
`hostname` varchar(128) NOT NULL,
`ipaddr` varchar(128) NOT NULL,
`dispname` varchar(64) NOT NULL,
`mode` int(11) NOT NULL,
`status` varchar(16) NOT NULL,
`disabled` tinyint(1) NOT NULL,
`created` datetime NOT NULL,
`updated` datetime NOT NULL,
PRIMARY KEY (`id`)

```

表：コンテンツ情報のデータ構造 (SQL)

```

`id` int(11) NOT NULL AUTO_INCREMENT,
`metaid` varchar(64) NOT NULL,
`dispname` varchar(64) NOT NULL,
`type` int(11) NOT NULL,
`contentopt` text NOT NULL,
`disabled` tinyint(1) NOT NULL DEFAULT '0',
`created` datetime NOT NULL,
`updated` datetime NOT NULL,
PRIMARY KEY (`id`)

```

表：コンテンツセット情報のデータ構造 (SQL)

```

`id` int(11) NOT NULL AUTO_INCREMENT,
`csetid` varchar(64) NOT NULL,
`dispname` varchar(64) NOT NULL,
`type` int(11) NOT NULL,
`contents` text NOT NULL,
`disabled` tinyint(1) NOT NULL DEFAULT '0',
`created` datetime NOT NULL,
`updated` datetime NOT NULL,
PRIMARY KEY (`id`)

```

表：アサインメント情報のデータ構造 (SQL)

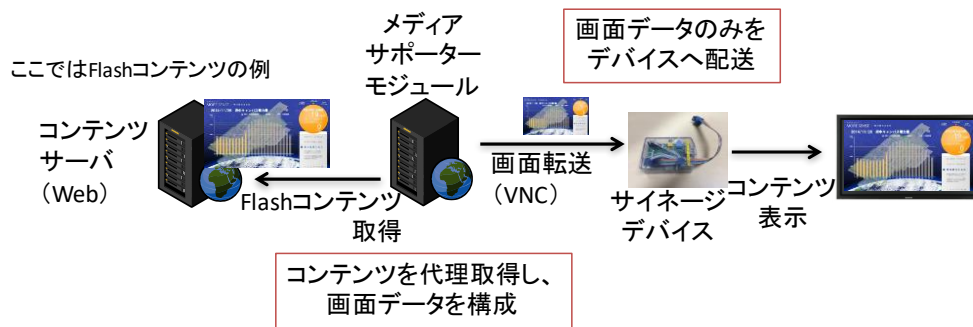
```

`id` int(11) NOT NULL AUTO_INCREMENT,
`devid` varchar(64) NOT NULL,
`type` int(11) NOT NULL,
`csetid` varchar(64) NOT NULL,
`starttime` time NOT NULL,
`endtime` time NOT NULL,
`enableday` int(11) NOT NULL,
`disabled` tinyint(1) NOT NULL DEFAULT '0',
`created` datetime NOT NULL,
`updated` datetime NOT NULL,
PRIMARY KEY (`id`)

```

- メディアサポーターモジュール：

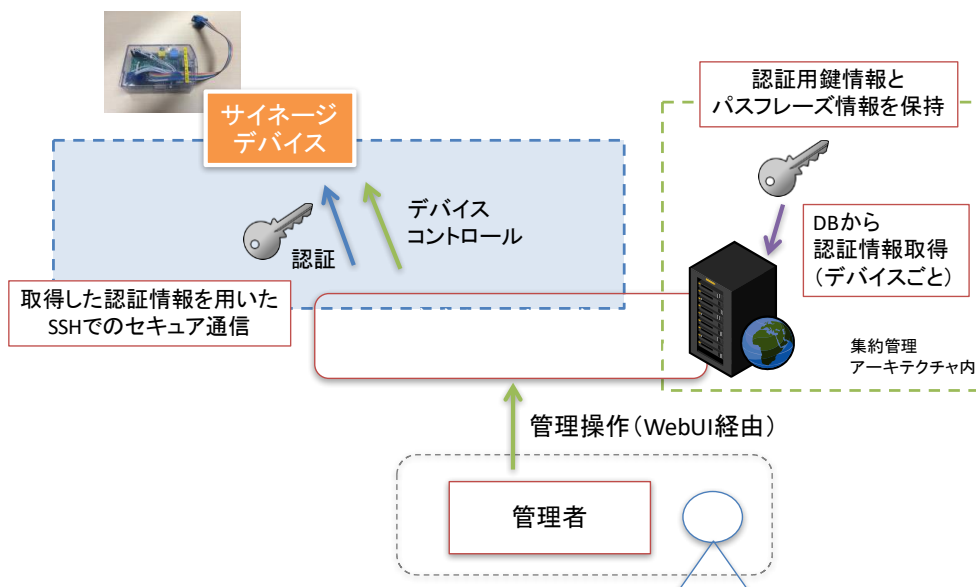
メディアサポーターモジュールは、VNC サーバ機能を有し、モジュール自身が表示する画面情報をサイネージデバイスに対して提供することができ、メディアサポーター上で表示できるものはいずれもサイネージデバイス上で表示できるようになる。メディアサポーター自身が集約管理アーキテクチャ上ではひとつのコンテンツとして認識され、メディアサポーターへの接続に必要な情報がメタ情報としてコンテンツデータベース上に登録される。



図：メディアサポーターでの処理例

## (2) 機器間メッセージング

デバイスの管理やコンテンツの管理のため、集約管理アーキテクチャと各サイネージデバイス間において設定・制御のためのメッセージングを行う必要がある。メッセージングには当初、REST/JSON をデータ構造とした独自の API 設計を行うことを検討していた。しかし、サイネージデバイス上で動作する各種モジュールはコマンドライン動作を前提とした構成となっており、新たなメッセージング API を開発するより、機器間でのメッセージングプロトコルとメッセージ自体の独立を図ることで開発コストの低減を図ることとした。機器間でのメッセージングプロトコルには、SSH(Secure Shell)を用いることとし、機器間通信におけるセキュリティ的な考慮は SSH プロトコル自体に委ねる形とした。実際には、各サイネージデバイスに対してコマンド的なコントロールを加える際には、SSH を経由してサイネージデバイス上のコマンドを実行する形式としている。また、設定変更については、SSH を経由してサイネージデバイス上の設定ファイルをリモートで変更するための実装を加えている。



図：メッセージング処理の概要

SSHでの通信の際には、通信前の認証が必要である。SSHにおける認証にはユーザ ID とパスワードを用いる従前よりあるプレーンな認証の他、公開鍵暗号認証を用いることができる。集約管理アーキテクチャでは複数のデバイスに対してメッセージングを行う必要があり、これらに対して共通のユーザ ID とパスワードを設定し、これを用いてメッセージングを行うことも可能である。しかし、特定の認証情報を使い回すことはセキュリティレベルの低下を招くことから、今回の仕組みでは、デバイスごとに認証情報を生成し、集約管理アーキテクチャとデバイス間で個別の認証を行うこととしている。実際には、パスフレーズ付き公開鍵暗号認証を用い、これに必要な認証情報をサイネージデバイスごとに生成し、認証に係る情報は集約管理アーキテクチャ上の別のデータベース上で管理するようにしている。機器に対して認証を実行できるように、SSH 認証のための鍵情報とパスフレーズをデータベース上で保持し、特定のプロセスからその認証情報を読み出し、自動的に SSH コマンドに対して提供するためのデバイス認証保持機能を通常の UNIX 系 OS に実装されている SSH クライアントのためのサポートモジュールを新たに実装している。

### (3) 管理用インタフェース (WebUI)

集約管理アーキテクチャは、個別のデバイスを制御する機能の他に、これらの設定やコンテンツ管理のために管理者が管理作業を実施する必要がある。これらの機能を有している。管理者による操作は原則として Web ベースのユーザインタフェースを通じて行うこととし、サーバや各デバイスへの個別のコマンドライン実行を行うことを必要としない。管理用のインタフェース (WebUI) には、1) デバイス管理インタフェース、2) コンテンツ管理インタフェース、3) コンテンツアサインインタフェースがそれぞれあり、これに加えて、管理者ユーザ自体の管理を行ったり設定を変更したりするインタフェースも別途用意している。1 から 3 の各インタフェース上で設定された項目は、設定情報のデータベースへの書き込みが終了次第、各サイネージデバイスに対して上述したメッセージングによって伝搬される。たとえば、コンテンツを登録し、そのコンテンツをデバイスに対してアサインした場合、当該のコンテンツ情報がターゲットとなるデバイスに対して機器間メッセージングを通じて設定がなされる。

管理用インタフェースについては、以下の環境にて実装、動作確認がなされている。なお、WebUI の開発には Perl の MVC フレームワークの一つである CGI-Application を用い、各種データの保持は SQL サーバ上にて行い、MySQL を RDBMS として用いている。また、Web ページ自体は、Bootstrap を採用し、シンプルかつ見やすいデザインとなるよう考慮している。

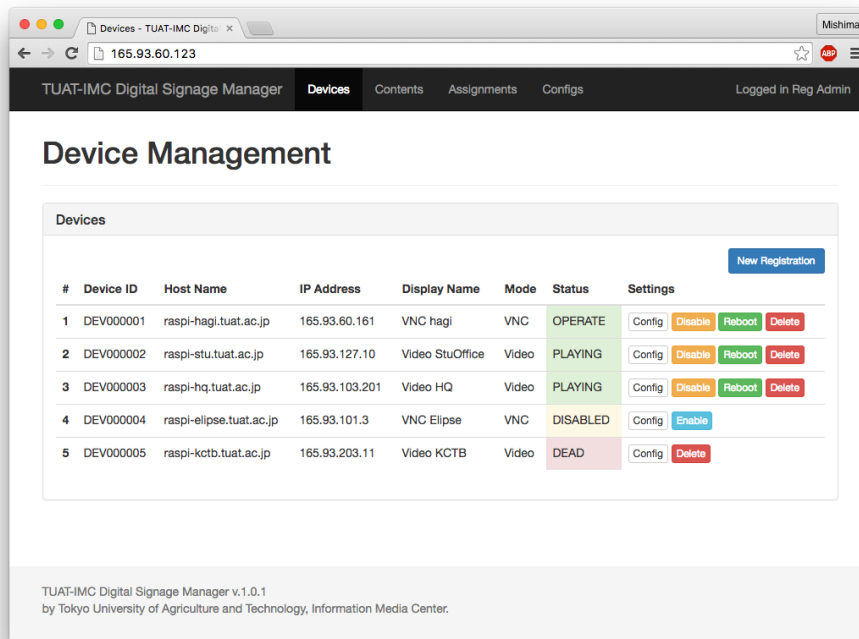
表：集約管理アーキテクチャの動作環境

OS	Linux OS (開発には CentOS 6.5 を利用)
Web サーバ	Apache Web Server (バージョン：2.2 系)
開発言語	Perl (バージョン：5.10) Bootstrap テンプレート
動作に必要なモジュール	CGI, DBD-MySQL 等の Perl モジュール (これ以外についてはローカルライブラリにモジュールを保持)
データベース	RDBMS として MySQL が必要

以下に各管理用インタフェースの詳細について記述する。合わせて、管理画面の例示も行う。

- デバイス管理インタフェース：

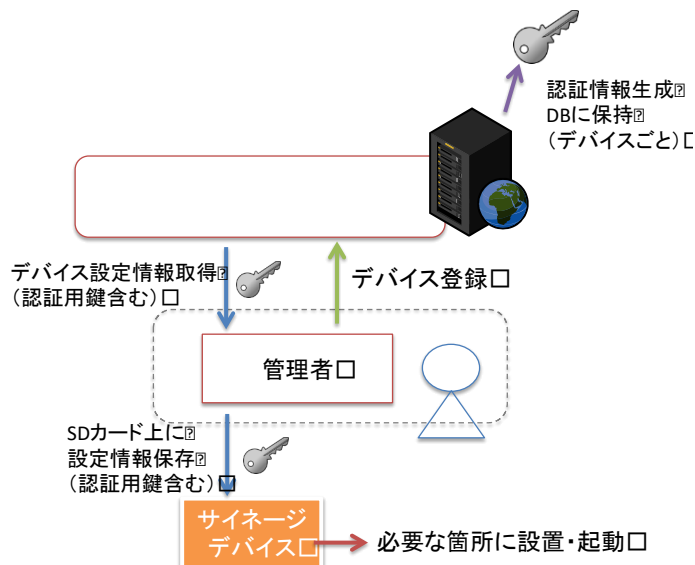
デバイス管理インタフェースは、各サイネージデバイスの登録と管理を行うためのインタフェースである。画面例では、デバイスの一覧表示が行われている形となる。管理者は、まずサイネージデバイスを集約管理アーキテクチャに登録するところから始まる。この作業が行われるのが本モジュールであり、サイネージデバイス管理にあたって最初に操作されることとなる。デバイス管理インタフェースでは、登録されたデバイスのユニーク ID・ホスト名・IP アドレス等の情報を登録し、一覧表示ではデバイスとの疎通状態・コンテンツの表示状態について表示する形となっている。疎通状態やコンテンツ表示状態については、各サイネージデバイスから一定時間ごとにデバイス間メッセージングを通じて情報を取得したものが表示される。



図：デバイス管理インターフェースの例

デバイス管理インターフェースでは、デバイスの登録のほか、デバイスの一時停止 (disable)、デバイスの再起動、デバイスの削除 (サイネージデバイス自体が初期化されるわけではない) を一覧画面から行うことができる。これ以外の設定項目に関しては、詳細設定の画面に遷移してから行う。

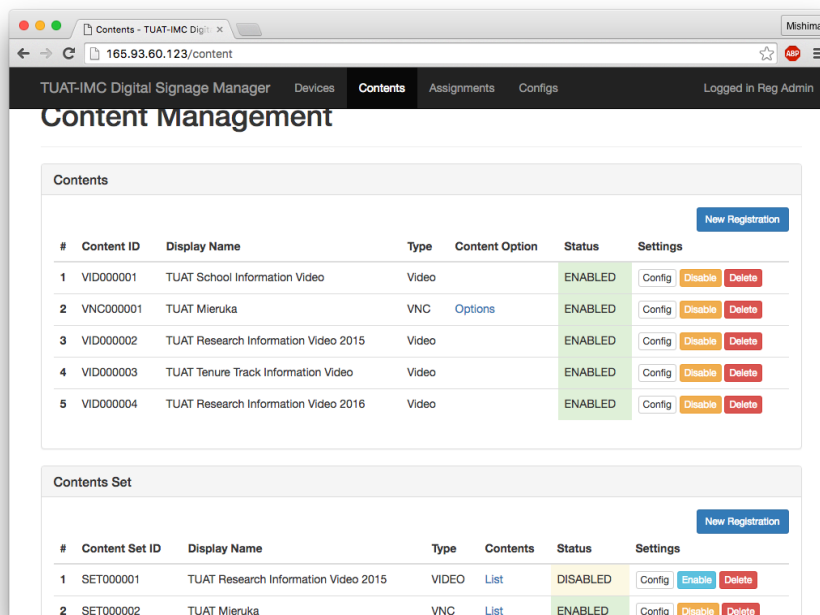
また、サイネージデバイスの登録はデバイス管理インターフェースより行う。デバイスを登録すると、デバイスマネージャ内でサイネージデバイスに保持させるべき設定と認証に係る情報が自動生成される。管理者はインターフェースを通じてこれらの情報をダウンロードでき、ダウンロードした設定情報を新たに登録したサイネージデバイスの SD カード上に保存することで初期設定を行うことが可能となっている。



図：サイネージデバイスの登録例

- コンテンツ管理インタフェース :

コンテンツ管理インタフェースは、各サイネージデバイスで表示されるべきコンテンツ自身を管理するためのインタフェースである。どのデバイスに何を表示するかは後述するコンテンツアサインインタフェースを用いるため、本インタフェースはあくまでもコンテンツ自体を登録・管理するために存在する。コンテンツ管理インタフェースでは、登録されているビデオファイルやメディアサポーター情報が表示される。実際に登録されているひとつひとつのコンテンツを登録した上で、各デバイスで表示させるコンテンツセット(例えば複数のビデオファイルをまとめて表示させるときにコンテンツセットを形成する)を合わせて登録する形となる。ここで登録されたコンテンツセット情報はコンテンツアサインの際に利用される。



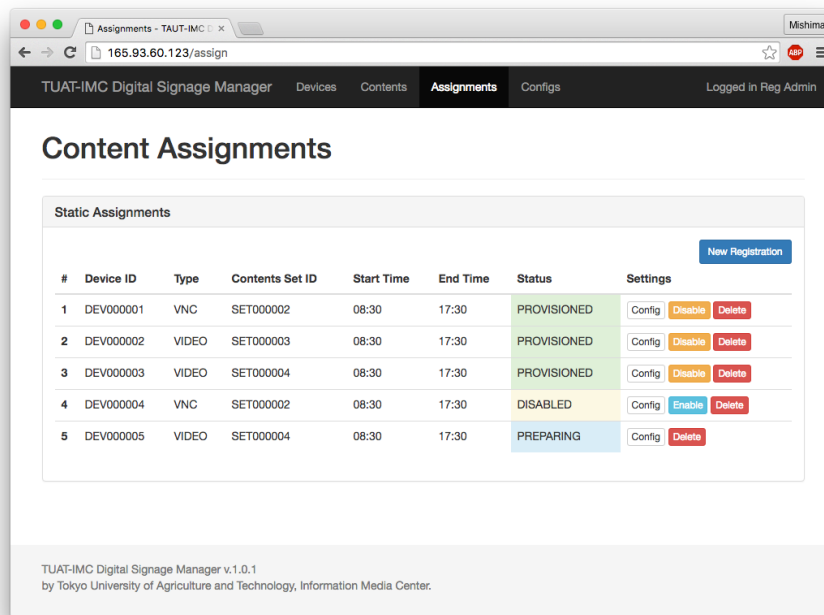
図：コンテンツ管理インタフェースの例

なお、コンテンツ自体は集約管理アーキテクチャ(セッションコントローラモジュール)が配置されるサーバ上にファイルとして保持される。ビデオファイルの場合は、コンテンツ登録時に別途用意されるファイルアップローダを通じてファイルをアップロードすることで、所定のフォルダに所定の名前(アップロード時のファイル名の揺れに対応するため、ファイルを保持する際はコンテンツ固有IDをベースとしたファイル名に自動的に変更される)として配置される。メディアサポーターを用いるコンテンツの場合は、コンテンツファイル自体が存在しないため、コンテンツ登録の際にあわせて登録されるコンテンツオプション(メディアサポーターへの接続に必要な情報)をデータベースに登録する形となり、ファイルの形でデータがサーバ上に保持されることはない。

- コンテンツアサインインタフェース :

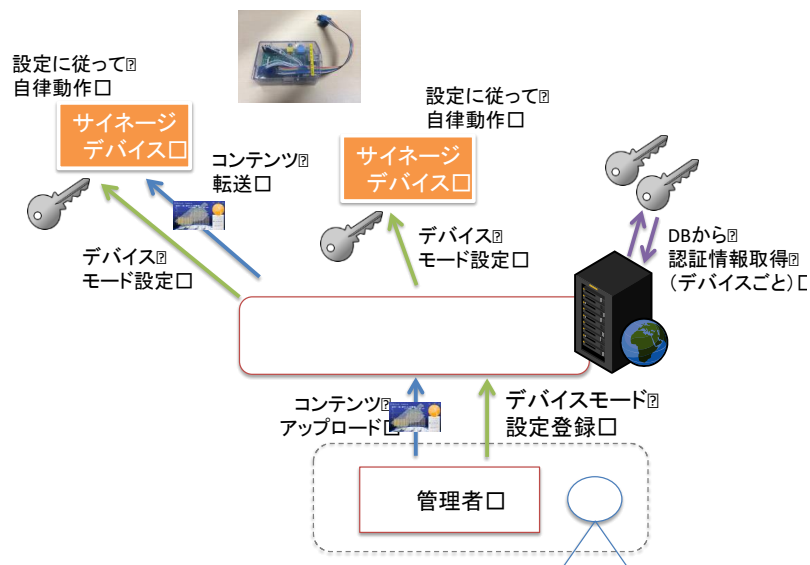
コンテンツアサインインタフェースでは、各サイネージデバイスが何をいつ表示するかを設定する。ここで、すでに登録されたサイネージデバイス情報とコンテンツセット情報のマッチングが図られ、この情報を元に各サイネージデバイスが設定されることとなる。ここでは、コンテンツをどう表示するかについても設定が行われる。現時点では、コンテンツをいつからいつまで表示するかについての設定を可能としており、ここで設定された時刻情報に基づき、各サイネージデバイスではコンテンツ表示の開始・停止といった処理が各デバイスの時計をベースに実行される形となる。





図：コンテンツアサインインターフェースの例

デバイスモードの設定は、コンテンツセット登録とコンテンツアサインが完了した後に自動的に実施される。デバイスモードは、動画ファイルを再生するモードとメディアサポーターを介したコンテンツ再生を行うモードに大別され、動画ファイルを再生するモードでは設定に合わせて、コンテンツファイル自体が各サイネージデバイスに対して転送される。コンテンツファイルの転送には、rsyncを用いる。この際、機器間での通信にはメッセージングで利用されるSSHを同様に利用し、認証に必要な情報についてもメッセージング同様、認証情報データベースより取得し、パズフレーズを用いる公開鍵認証を通じて接続が実施される。こうしてモード設定された各デバイスは、自律的な動作が可能となり、設定変更が行われるまではコンテンツアサインで設定された時間にて自動的に動作をする。



図：デバイスモードの設定

### 3-3 集約管理を実現するサイネージデバイスの改修

先述した Raspberry Pi をベースとした小型かつ低廉なサイネージデバイスは、各デバイス単独でコンテンツ表示と表示に係る制御を実施することができるよう実装されている。当初設計・実装されたサイネージデバイスは、あくまでも個別に稼働することを目的としたデバイスであり、各機器に対して個別の実装をデバイスごとに行い、設定についても個別に行う形態となっていた。集約管理機能の実装に合わせ、これらデバイスの集約管理対応を行う必要があり、これに対応した実装をすでにあるサイネージデバイスに対して行った。また、この実装に合わせて、従来の Raspberry Pi をベースとしたサイネージデバイスの抱えるいくつかの問題についても改修を行った。今回、改修を実施した内容については、大別すると、1) サイネージデバイス実装の統合、2) サイネージデバイスの遠隔管理対応機能の実装となり、それぞれについて以下にて詳述する。

- サイネージデバイス実装の統合：

これまでに実装されていたサイネージデバイスは、動画ファイルの再生を行う機能を有するものと、Flash コンテンツ等高度なメディア再生を行う機能（リッチコンテンツ提供サーバと連携し当該コンテンツの画面データを表示する機能）を有するものがあった。これらは、それぞれ別の実装物として存在しており、内部に開発実装されたモジュールならびに設定ファイル等は個別のものとなっていた。これでは、サイネージの集約管理対応を行った際に、連携するモジュールをそれぞれ用意する必要があり、効率的ではないと判断し、各機能をひとつのサイネージデバイス実装に統合した。この統合に伴い、サイネージデバイスに含まれる再生モジュールや設定ファイル等の共通化を図ることができ、集約管理時の連携機能についてもシンプルな構成とできるようになった。

- サイネージデバイスの遠隔管理対応機能の実装：

これまでの実装物であるサイネージデバイスは、単独での動作を想定した構成となっており、自身の持つ時計をベースにデバイスの再起動やコンテンツの再生・停止を行うようになっていた。集約管理に対応するためには、これらの実装に対して、遠隔からのコントロールが可能となるように調整することと、単独で勝手に動作しないよう機能抑制を行う必要がある。そこで、各デバイスが自律的に動作するモジュールについて、遠隔からのコントロールに基づく動作統制を図る機能を実装した。実際は、特定の設定ファイルを参照し、それに従って動作するかどうかを決定する実装を行い、集約管理時は当該ファイルを適切に変更するようにすることで、単独動作も集約管理も可能とする構成とした。

また、集約管理アーキテクチャからのコントロールに対するため、遠隔管理に対応する必要がある。後述する集約管理アーキテクチャと各サイネージデバイスとの間でのメッセージングを受信する構成とする。なお、メッセージングには UNIX 系 OS のリモート管理に利用される SSH(Secure Shell) の機能を応用する形とした。これにより、メッセージングに関するモジュール自体の実装は不要となり、受信したメッセージに応じて内部で処理するモジュールの開発を行う形にとどめている。

### 3-4 一斉同報を実現するためのモジュール実装

本研究開発では、これまで挙げた機能の実装を行うことで、サイネージデバイスの集約管理を可能とするアーキテクチャの実現を図った。また、平成 27 年度計画上で設けられていた「一斉同報」に係る機能は、各サイネージデバイス上に特定の画像ファイルを自動表示するモジュールを実装している。そして、これらを集約管理アーキテクチャ上から遠隔操作でコンテンツ（画像ファイル）配信とモジュール実行を行うことを可能としているが、非常に簡素な機能実装にとどまっているため、引き続き機能の高度化が求められる。これについては継続的に取り組んでいきたい。

#### 4 本研究開発のまとめと今後の課題

本年度における研究開発は、実施計画に基づき、小型・低廉なサイネージデバイスとその集約管理に係る機能実装を実施した。集約管理アーキテクチャは、各サイネージデバイスを遠隔からコントロールし、機器・コンテンツ・実行状況について管理するためのインタフェースを設け、機器間はセキュアな通信によってコントロールメッセージを送受信する。また、各サイネージデバイスについてもこれまで実装されていた単独動作するシステムを遠隔管理対応なものに切り替える実装を行っており、これによって集約管理アーキテクチャとの連携を図る。これらに対して、システム関連の設計に対するフィードバックを得ることと、その他要望調査のために、国内・外の研究発表の場にて口頭発表・デモ発表を実施している。発表の際の抄録等については別途添付資料にて提示させていただく。論文誌掲載については当該年度中に再録されたものがなかったため、引き続き投稿・再録に向けた活動も行っていく必要があると考えている。

なお、本研究開発における今後の課題として挙げられるものが、引き続きの開発継続と広くユーザに利用してもらうためのパッケージングである。現時点で開発されている各モジュールについて完成度が高い状況にあるわけではなく、ブラッシュアップが必要であると考えている。また、一斉同報に係る機能についても継続対応が必要であると考えており、取り組みを継続していく。なお、平成28年度についても助成を継続いただき、引き続きの活動を行えることとなっており、初年度計画における集約管理アーキテクチャの実現に向けた活動に加え、次年度計画として、(1) 集約管理アーキテクチャの高度化：約管理アーキテクチャの高度化として、サイネージデバイスが室内に展開されることを想定し、GPS などではなく、近隣に設置された無線LANシステムとの協調により、位置情報を推測し、推測された情報に基づくサイネージデバイスの各種設定項目（ネットワークや表示コンテンツの設定）の簡素化、ならびに、自動化（ゼロコンフィグレーション）を実現する、ならびに、(2) 初期構築コスト低減を図るためのアプリケーションパッケージ開発：サイネージシステム構築パッケージの実現として、本研究開発の成果物を公開し、利用するにあたって、ユーザビリティを高めるために各種環境に向けたアプリケーションのパッケージ化を行う。これにより、ユーザは提供されたパッケージをOS等の実行環境に展開することで容易に自律統合型サイネージシステムの運用を開始できるようになる、を行いたいと考えている。

初年度計画の成果により、オープンシステムとしてより低コストにサイネージデバイスを構築し（機器コストの低減）、かつ集約管理を行うことを可能とした（運用コストの低減）。これに加え、本次年度計画の成果により、構築コストと運用コストの両面からさらなるコスト低減を可能とする。これによりさらなる手軽さを有したデジタルサイネージシステムを実現し、デジタルサイネージをより広く一般的なものとするのが可能となる。

#### 【参考文献】

- [1] Yasuhito Tsukahara, Takahiro Nemoto, Masaaki Sato, Atsushi Kawami, Hiroshi Kobayashi, Kazunori Sugiura: "LORDS": Design and Implementation of Location-Based Digital Signage System, Proc. of ITS World Congress 2013, Tokyo, 2013, pp.31-40
- [2] Q. Chen, F. Malric, Y. Zhang, M. Abid, A. Cordeiro, E.M. Petriu and N.D. Georganas, "Interacting with Digital Signage Using Hand Gestures," in International Conference on Image Analysis and Recognition (ICIAR 2009), Montreal, 2009, pp. 347-358.
- [3] NMStage: URL: <http://panasonic.biz/it/nmstage/>
- [4] MEDIAWAY:  
URL: [http://www.mitsubishielectric.co.jp/visual/digital\\_signage/l\\_scale/mediaway/index.html](http://www.mitsubishielectric.co.jp/visual/digital_signage/l_scale/mediaway/index.html)
- [5] トレインビジョン: URL: <http://www.mitsubishielectric.co.jp/society/traffic/product/syaryou/s10.html>
- [6] Screenly: URL: <https://www.screenlyapp.com/>
- [7] Raspberry Digital Signage: URL: <http://www.binaryemotions.com/raspberry-digital-signage/>

〈発表資料〉

題名	掲載誌・学会名等	発表年月
Cost Effective Digital Signage System using Low Cost Information Device	13th IEEE Annual Consumer Communications & Networking Conference (CCNC2016)	2016/01
小型・低価格情報デバイスによるデジタルサイネージデバイスの集約管理システムの設計	電子情報通信学会 2016 年総合大会	2016/03