# 視聴者参加型番組のリアルタイム連動 Web を実現するサーバ連携ミドルウェア

†中島一彰、松元繁明、大芝崇、小池雄一、田淵仁浩、神場知成
‡NEC インターネットシステム研究所

# Servers Linkage Middle-ware for Real-time Synchronized Web Page Associated with Viewer Participation Programs

Kazuaki Nakajima, Shigeaki Matsumoto, Takashi Oshiba, Yuichi Koike
Masahiro Tabuchi and Tomonari Kamba
NEC Corporation, Internet Systems Research Laboratories

**概要**

本報告では、インターネット放送やラジオ放送における視聴者参加型番組に、リアルタイム連動した Web ページから視聴者が参加することを可能にするサーバ連携ミドルウェアについて述べる。ライブ放送の視聴者参加型番組では視聴者とのインタラクションにおいてリアルタイム性が重視される。本ミドルウェアの特徴は、(1) 番組進行中に動的に指定される司会者の指示や視聴者からのリアクションをリアルタイムに Web サーバや配信サーバに自動的に反映できる連携ミドルウェアであること、(2)番組進行中、アドホックに作成される最新情報を、多人数の視聴者の Web ページにリアルタイムプッシュ更新して効果的な演出を与えられる機能を備えていること、である。

## 1. はじめに

広帯域常時接続のブロードバンドインターネットの普及は、映像や音声などのストリーミングコンテンツの配信だけでなく、Web ページと連動したチャネルミックス戦略を可能にした。映像や音声を一方的に送るだけではなく、文字や画像のコンテンツを付加価値として提供し、さらにユーザからのリアクションを直接収集できる。その中でもユーザの顧客情報と顧客ニーズが収集できる番組コンテンツが視聴者参加型のライブ放送である。しかし、ライブ放送の番組運用は難しく、従来は、あらかじめ決まった内容に限定し、かつリアルタイム性を低下させるほかなかった。

そこで、著者らは視聴者参加型番組向けのリアルタイム連動 Web を実現するためのサーバ連携ミドルウェアを開発した。本報告では、特に、視聴者のリアクションをリアルタイムに番組全体に反映させることができるサーバ連携ミドルウェアと、多数の視聴者の Web ページをリアルタイムに同時プッシュ更新可能な Web プッシュ更新機能について報告する。

## 2. 視聴者参加型番組

### 2.1 視聴者参加型番組とは

インターネット放送またはラジオ・テレビ放送における視聴者参加型番組においてライブ放送では、翌日以降の放送に反映させるのではなく、進行中の番組に反映させることが望ましい。視聴者参加型番組の例を示す。

(a) アンケート・クイズ番組

アンケートやクイズを出して、連動した Web で視聴者に答えてもらう。アンケートやクイズの内容は、視聴者からの回答や司会者の感性で出題するたびに随時定義され、番組の進行とともに変化していく。アンケートやクイズは司会者が告げるタイミングで初めて Web ページにプッシュ更新され、ほかの視聴者の回答状況が刻々と表示される。

(b) インターネットスポーツ中継、ドラマ

スポーツ中継で複数のカメラアングルを小さく表示し、視聴者にもっとも人気があるアングルの映像を、広い帯域で送り大きく表示する。インターネットドラマでは、放送されるドラマのストーリーを選択式アンケートの視聴者投票で決定する。状況に応じてアドホックに司会者の指示と視聴者の投票で番組を進行する。

### 2.2 視聴者参加型番組のシステム的な課題

リアルタイム性を重視する視聴者参加型番組では、運用システム全体のリアルタイム連動が必要である。Web サーバ、応答サーバ、プッシュ更新サーバ、あるいは映像配信サーバをリアルタイムに連携（同期）する必要がある。しかし、インターネット上ではサーバの運用システムが ASP、ハウジングなどで独立している場合が多い。既存の方式は固定スケジュールに限定して各サーバを独立動作させて限定的な番組にするか、人海戦術で対応しなければならず、リアルタイム性が確保できない。一般的には FAX やメールを司会者の手元に印刷してアナログで放送するにとどまっており、Web サーバなどへリアルタイムに反映できない。

また、Web サーバから視聴者の Web ブラウザへのリアルタイム配信にも問題がある。ライブ放送型の映像・音声コンテンツと Web ページをリアルタイム連動するには、多くの視聴者の Web ページをプッシュ更新する必要がある。SMIL[1]を基本とする既存の技術ではインターネットのストリーミングに同期して、Web ページを紙芝居的に更新することができる。しかし、既存のシステムだけでは、前もって確定できない動的な視聴者からの応答を、更新タイミングや更新内容に自動的にリアルタイムに反映させることができない。既存のプッシュ更新では、紙芝居的なプッシュ更新以外に詳細なプッシュ更新が難しいので、更新領域が大きくなり、人数が増えるほどリアルタイム性が低下してしまう。

## 3. 設計方針と全体構成

### 3.1 設計方針

解決するべき課題は、ライブ放送で動的に変わり得る番組進行のリアルタイム性の保証である。保証すべきリアルタイム性には運用システム内および多数の視聴者間の 2 種類を同時に確保する必要がある。したがって、大きく 2 つの設計方針を設けることができる。

(1) **視聴者のリアクションやライブ放送での指示を自動的に反映するミドルウェアを提供することで、システム内のリアルタイム性を向上する**

(2) **視聴者 Web ページへの軽量なリアルタイムプッシュ更新を備えることで、放送と視聴者とをリアルタイムに結ぶ**

### 3.2 全体構成

本システムは**図 1** に示す 4 層のレイヤ構成をしている。それぞれのレイヤではサーバシステムが独立しており、用途に応じてシステムを柔軟に変更できるように API が定義されている。

†連絡先　nakajima@ah.jp.nec.com
‡住所　神奈川県川崎市宮前区宮崎 4-1-1

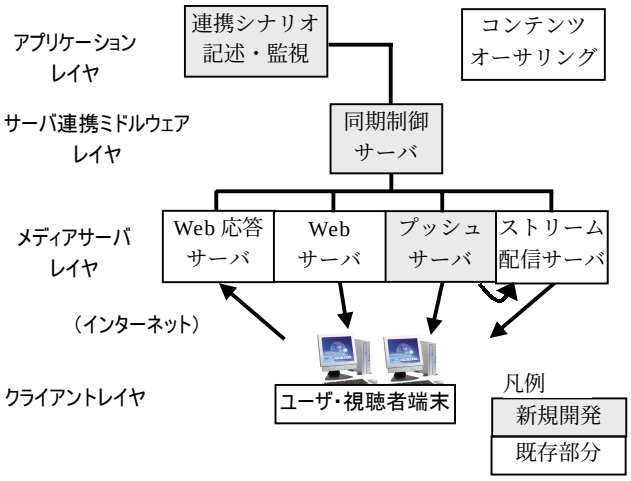



**図１　システムのレイヤ構成**

アプリケーションレイヤでは可換可能な UI を提供する。既存のオーサリングツールで静的な Web コンテンツあるいは映像コンテンツを用意し、連動のタイミングは連携シナリオ記述監視ツールで時系列のオーサリングを行う。連携シナリオ記述監視ツールは番組の種類と用途に応じて UI や機能を追加、変更できる。

サーバ連携ミドルウェアレイヤでは、同期制御サーバが連携シナリオにしたがって下位の複数のメディアサーバレイヤを同期制御することで、システム全体の連携動作を制御する。

メディアサーバレイヤでは既存システムのメディアサーバを用途に応じて追加可能である。同期制御をする連携ドライバをプラグインすることで簡単に既存のサーバシステムが追加できる。

クライアントレイヤには既存の Web ブラウザおよび映像やプッシュ通知を受信するためのプラグインモジュールが組み込まれる。多数の視聴者を収容できるサーバが必要となる。

プッシュサーバは新規開発したメディアサーバである。クライアントの Web ブラウザに対して、Web コンテンツの差分プッシュ更新を可能にする。更新方法として、５章で述べる新たに開発した Web プッシュ更新と、ストリームのフレーム連動のプッシュ更新が可能である。

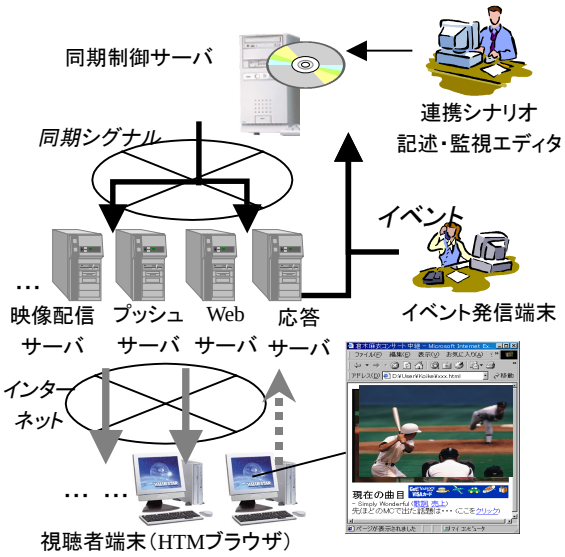



**図２　システム構成例**

## 4. サーバ連携ミドルウェア

### 4.1　連携シナリオを可能にする集中型アーキテクチャ

サーバ連携ミドルウェア[2]は同期制御サーバが各メディアサーバのタイミングを一括して制御することで、サーバの連携動作を可能にする。**図２**にサーバ連携ミドルウェアのシステム構成例を示す。同期制御サーバは各メディアサーバ、連携シナリオ記述・監視エディタ、およびイベント発生端末と連結されている。連携シナリオでは後述する方式で時間によるスケジュール駆動とイベント駆動が記述できる。同期制御サーバからは各メディアサーバに動作のトリガーとなる同期シグナルが発行される。

**図３(a)**に示す連携シナリオ・記述監視エディタ[3]では、各メディアサーバの連携シナリオを一括記述する。従来の独立したスケジュール記述に比べて、細かい緊密した連携動作が容易に記述できる。各メディアサーバにまたがる動作について動作タイミングを一括して相対時間で記述でき、「Ａサーバで動作した１分後にＢサーバで動作」などの記述ができる。

また、同期制御サーバで集中制御することでユーザリアクションをフィードバックできる。Web 応答サーバから発行されるイベントを同期制御サーバが受け取り、連携シナリオに基づいて、別のサーバへの動作を指示することができる。

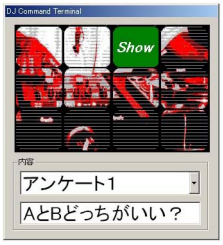
この連携シナリオ・記述監視エディタでは、動作時には動作状況が刻々と表示される。システム全体のリアルタイム監視ができるだけでなく、連携シナリオの動作直前のリアルタイムな変更が可能である。不確定要素が多い生放送に対応することができ、スケジュールの停止や再開などの操作が可能となる。

### 4.2　連携シナリオの記述

連携シナリオはスケジュールデータ群とイベント動作データ群からなる。イベントとスケジュールが状態変数を媒介にして柔軟に連携する。**図４**に連携シナリオの概念図を示す。スケジュールデータは ID、タイミング、コマンドから構成される。タイミングが成立したらコマンドを実行する。

タイミングはサーバ間にまたがる動作やイベント駆動型のシナリオによる連携動作が柔軟に記述できる。タイミングには時間、

**(a) 連携シナリオ記述・監視エディタの実行画面**




**(b) イベント発生端末の一例**

**図３　システム管理の画面例**

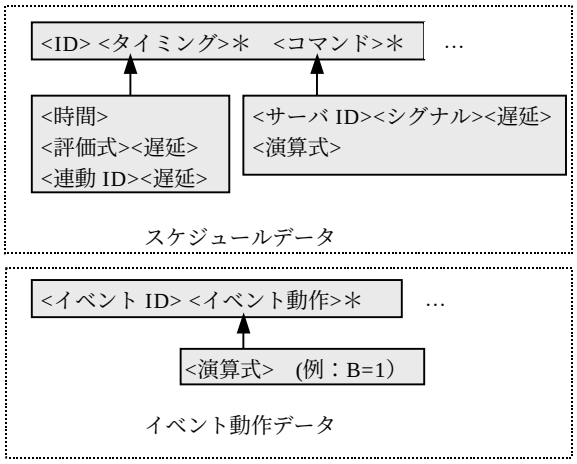



**図４　連携シナリオデータ**

評価式、連動 ID を複数選択することができる。時間は番組開始時点からの相対時間、または絶対時間が指定できる。評価式は状態変数を媒介にしてイベント駆動型のスケジュールに利用する。イベントなどによって変数が演算され、評価式が成立した瞬間から遅延時間後にタイミングが成立する。連動 ID はほかのスケジュールデータと連動するときに指定する。連動 ID のスケジュールのタイミングが成立した瞬間から遅延時間を加算した時点でタイミングが成立する。このようにタイミング間の連携記述ができるようになっている。

コマンドは複数記述することが可能である。同じタイミングで複数のサーバへ同期シグナルを発行できる。なお、各シグナルの発行タイミングは、それぞれさらに遅延時間を加算できるようになっているので、サーバごとに数十秒ずらすこともできる。なお、コマンドはイベントと同様に変数演算を指定でき、イベント動作と柔軟に連携した記述ができる。

指定された ID のイベントを受信するとイベント動作データにある演算式が動作し、変数が変更される。そのときにスケジュールデータのタイミングの評価式が成立したら、タイミングが動作する。単純なイベント駆動に関しては変数をフラグとしてオンオフで行う。複雑なイベント駆動ではたとえば、二つのイベントが発生したらタイミングを発動する場合には、[イベント 91]で A=1、[イベント 92]で B=1 と変数演算を定義して、スケジュールのタイミング評価式で A*B=1 と定義する。

### ４.３　視聴者応答のリアルタイムフィードバックの例

視聴者の応答などをフィードバックする方法として、連携シナリオではイベント駆動型の動作を提供している。具体例としてアンケートの募集と結果表示の例を示す。

**(1)アンケートの募集**

アンケート募集 A のスケジュールデータでは、Web ページに視聴者へのアンケート画面を表示させて、応答サーバでアンケートの受け取りを開始する２つのコマンドを作成する。タイミングはイベント駆動にして、司会者の指示で任意のアンケートを表示できるようにする。変数 R をフラグとしてイベントの駆動に利用する。そのときのイベント発生端末は**図３(b)**である。

**(2)回収結果の通知**

応答サーバにおいて一定時間に達したら、応答サーバが結果を保存し、イベントを同期制御サーバに送付する。途中経過を表示するためのイベントの発行タイミングは一定間隔ごとあるいは回答が一定数になったらイベントを発行する。イベントを同期制御サーバが受け取ると、変数Ｅに回答数を代入する。

**(3)回答のプッシュ配信**

アンケート結果表示 B のスケジュールデータでは、変数評価式 E*R>0 があり、対応するコマンドをプッシュサーバに対して発行する。R はアンケートを募集していない段階でプッシュ更新しないための安全装置である。プッシュサーバでは、結果を読み込んで Web ページの生成および差分データを生成して、回答結果をユーザのブラウザでプッシュ更新して表示する。

### ４.４　メディアサーバへのシグナル送信制御

メディアサーバに連携ドライバがインストールされており、同期制御サーバが発行する同期シグナルを連携ドライバが受け取り指定された動作をする。同期シグナルに割り当てられた ID に対応する動作が連携ドライバにプログラムされている。ID と動作はあらかじめ連携ドライバが定義し、連携シナリオ記述・監視エディタで同期シグナルを指定する。

分散されたメディアサーバ間で完全に同期タイミングを一致させる先行配信プロトコルを提供する。インターネット運用で問題となる伝送時間の不安定さと、各メディアサーバの動作準備時間を吸収する。先行配信プロトコルでは同期シグナルをメディアサーバに向けて先行配信して、同期シグナルがメディアサーバ内で正確な時間で動作する。先行仮想時間はインターネットの推定遅延時間とコンテンツ切替え時間の合計である。メディアサーバ内でコンテンツ切替えの時間が必要な場合は、動作時間までにコンテンツを準備する。

なお、先行配信されたあとの緊急事態の対応にために、先行配信制御プロトコルがある。先行配信シグナルの取消し、中断、再開がある。先行配信された同期シグナルを制御できるので、緊急ニュースや、急なスケジュール変更があった場合の先行配信リスクを回避できる。

## 5. Web プッシュ更新

### ５.１　プッシュサーバのアーキテクチャ

プッシュサーバは複数の更新方法を提供することで、映像ストリームの種類を問わない統一的な Web プッシュ更新が提供できる。**図５**に示すように、プッシュ更新のエンジンとして、独自開発の軽量プッシュ更新システム(LiveComm)と、ストリーム連動のプッシュ更新である Windows Media Technology(以下 WMT)サーバ[4]、Real[5]のエンジンに対応している。いずれもプッシュ更新用の HTML コンテンツを用意すれば、自動的にコンテンツ更新用のエンジンを介してプッシュ配信され、初回ダウンロードの初期 HTML ページが自動更新される。インターネット配信される映像や音声のバッファリングを考慮した連動にはストリーム連動を利用する。視聴者間で時間的なリアルタイム同期が必要な場合は、LiveComm を利用する。

### ５.２　軽量プッシュ更新システム LiveComm

軽量プッシュ更新システムは HTML コンテンツのリアルタイム更新に特化した軽量差分更新システムである。ストリーム連動

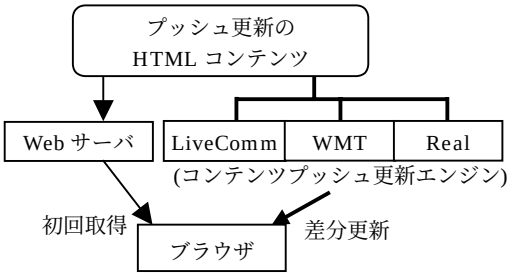



**図５　プッシュサーバのアーキテクチャ**

の更新システムが、ユーザごとに異なるストリームのバッファリング時間分だけ更新タイミングがずれてしまうのに対して、この軽量プッシュ更新システムは、同時接続された大多数のクライアントに対して、リアルタイムに更新コマンドを発行できる。

実装としては、プッシュサーバとブラウザにプラグインされた通信モジュールで差分を転送して、ブラウザにダウンロードされた差分更新用スクリプト言語が HTML ページを書き換える。通信経路として、永続的な双方向 HTTP トンネリングプロトコル[6]を備えており、ファイヤウォールを超えてイントラネット内からも利用可能である。プッシュ更新のトリガーを任意のタイミングでリアルタイムに配信することができる。なお、更新方法として、5.3 節で示す領域とアイテムのプッシュ更新を備え、更新コマンドの組合せで汎用的にかつ容易に外部からシステムから利用できる。コンテンツオーサリングに関しては、更新対象 HTML の更新個所のタグに ID 属性を追加して、対応した更新コマンドを用意するだけであり、既存のオーサリングツールで容易に対応できる。

### 5.3 領域のプッシュ更新

領域更新では更新 HTML コンテンツをあらかじめ指定された任意の ID のタグに挿入する。WMT、Real の差分更新にも対応している。具体的にはブラウザ内の HTML コンテンツに

<DIV ID='TargetID'></DIV>

のように記述されていたときに、この DIV タグの中身を更新 HTML コンテンツに差し替える。**表 1** に示すように、いくつかの更新方法が用意されている。差分更新データの大きさと、更新表示までの時間的余裕によって、更新コマンドを自動選択することもできる。また、同時接続ユーザ数が多くなったら、ダウンロード通知を利用することでトリガーを小さくして、更新タイミングのずれを極力少なくする。更新コンテンツ先行ダウンロード通知は、更新 HTML がリンクする大きな画像などを、前もって送信することにも利用できる。

### 5.4 詳細アイテムのプッシュ更新

軽量プッシュ更新システムでは ID 属性のついたすべてのタグに対して、詳細なプッシュ更新ができる[7]。タグの追加、変更、削除、属性変更が可能となっている。アイテム更新コマンドを**表 2** に示す。色の変更、文字の書き換え、画像の差し替えなどを柔軟にできる。さらに、クイズの正解発表のリアルタイム性を向上する場合には、正解発表の大枠を数十秒前に領域更新でプッシュ更新しておき、正解の文字列をアイテム更新コマンドで送付する、という使い方ができる。なお、複数のアイテム更新コマンドを連続して送信することで、ひとつの更新操作とする場合には、コマンド列をパッケージとして送付して、コマンド間のタイムラグをなくすことができる。

**表 1　領域プッシュ更新コマンド一覧**

| コマンド名 | 機能 |
|---|---|
| 更新 HTML<br>トリガー内直接通知 | 比較的小さい差分コンテンツをトリガー内に含めて送信して、すぐに表示する。 |
| 更新 HTML<br>通常ダウンロード通知 | トリガーで URL を送信し、差分コンテンツを HTTP ダウンロードして、終了後すぐに挿入、表示する |
| 更新コンテンツ<br>先行ダウンロード通知 | トリガーで URL を送信し、差分コンテンツを HTTP ダウンロードして内部保有しておく。後に表示。 |

**表 2　アイテムプッシュ更新コマンド一覧**

| コマンド名 | 機能 |
|---|---|
| AppendElement | 新規タグの追加 |
| ChangeElement | タグ名の変更 |
| RemoveElement | タグの削除 |
| SetAttribute | タグに属性をセット |
| RemoveAttribute | タグの属性を削除 |
| TextNodeChange | ノードにテキストを追加 |

## 6. 実現と評価

本システムは、Microsoft Windows 環境でプロトタイプが実現されている。同期制御サーバおよび連携ドライバ、連携シナリオ記述・監視エディタが実現されている。また、軽量プッシュ更新システムとして、プッシュ更新サーバおよびブラウザのクライアントモジュールとして ActiveX Control プラグインとして実装した。連携ドライバとして、現在、Web サーバにはコンテンツファイルのコピーとアクセス権限を変更する連携ドライバがあり、応答サーバには、回収期限の設定と結果をイベント発行する連携ドライバがある。またストリームドライバとして WMT サーバを制御するストームドライバがある。プッシュサーバはドライバ機能が内蔵されている。

サーバ連携ミドルウェアの評価として、連携ドライバのリアルタイム性の確認を行い、先行配信プロトコルによって時間通りに正確に動作することを確認している。Web プッシュ更新の評価として、同時接続数 2,500 の場合に 5 秒以内にサーバ内において更新通知トリガーの配信が終了することを確認した。サーバからクライアントへの通知時間は回線品質に依存するが、Web のブラウジングの品質と同等であることを確認した。回線品質に余裕がありプロキシを 2 段経由した環境においては、平均 150ms で更新できることを確認した。数万人が同時にアクセスするサーバ群を低コストで導入できる見込みがあることが分かった。

## 7. おわりに

本報告では、ライブ放送における視聴者参加型番組のリアルタイム連動 Web を実現するためのシステムとして、サーバ連携ミドルウェアおよびその軽量 Web プッシュ機能を開発したことについて述べた。今後、インターネットにおける本格運用のための実施試験を行い、リアルタイム性と運用面の実用性を検証する。

## 参考文献


[1] SMIL http://www.w3.org/AudioVideo/

[2] 中島、松元、大芝、吉坂、小池、田淵、神場：視聴者参加型インターネット放送の自動制御を可能にするチャネル連携システム、情報処理学会第 64 回全国大会、2G-04、2002 年 3 月

[3] 松元、中島、大芝、吉坂、小池、田淵、神場：インターネット放送における連動情報配信シナリオ編集、実行監視システム、情報処理学会第 64 回全国大会、2G-05、2002 年 3 月

[4] Windows Media Technology http://www.microsoft.com

[5] リアルネットワーク社 http://www.real.com

[6] 中島、阿部、田淵：インターネット品質に対応したマルチメディア情報リアルタイム共有制御システム、情報処理学会 DiCoMo2000 シンポジウム、pp.265-270、2000 年 6 月

[7] K. Nakajima, T. Oshiba, M. Tabuchi, T. Kamba, *A Real-time E-Marketplace System for Advertising Opportunities on Datacasting*, Tenth International World Wide Web Conference, 2001.3 (http://www10.org/cdrom/posters/p1133/index.htm)