

WWWにおける時間的制約のある プッシュ型情報配信機構の設計とプロトタイプ実装

田中 久夫 河合 栄治 小山 琢己 山口 英 山本 平一

奈良先端科学技術大学院大学 情報科学研究科

概要

番組連動型情報サービスのよう、多数のユーザに対する同報性の高いサービスをインターネット上で実現するためには、スケーラビリティの問題を解決しなければならない。インターネット上での同報型コンテンツ配信にはマルチキャストが適している。しかし、技術的な諸問題などで、現在ほとんどのインターネットユーザが利用できないのが現状である。そこで、本研究では、従来のプッシュ型とプル型のそれぞれの情報配信機構を応用した、ユニキャストによる高いスケーラビリティを有する情報配信機構を提案する。プッシュ型で配信するデータは、リアルタイム性の要求が高いものに限定する。また、リアルタイム性の低い配信は従来の HTTP によるプル型配信を行う。その際、サーバへのアクセスの集中を回避するために、ダウンロードタイミングの時間的分散を行う機構を導入することで、提案配信機構のスケーラビリティを確保する。本稿では提案システムの全体設計、そして、録画放送番組の場合についてプロトタイプシステムの実装を行ったので報告する。

A Push Event Delivery Mechanism for Web Content Service Synchronized with TV programs

Hisao Tanaka Eiji Kawai Takumi Koyama Suguru Yamaguchi Heiichi Yamamoto

Graduate School of Information Science, Nara Institute of Science and Technology

Abstract

This paper proposes a push event delivery mechanism to realize content data service synchronized with TV programs on the Internet. To realize such content service, multicast seems to be a good solution. However, the system requires TCP-like reliable data transfer, which is rather difficult to provide on multicast. In addition, multicast is not suitable for service personalization. The proposed delivery mechanism in our system solved the problems mentioned above by dividing the content service into two parts. One is for real-time client control data, which are delivered to the clients in a pure push style. The other is for content data, delivered to the clients in a traditional pull style with HTTP. This paper describes the details of proposed system design and implementation. Discussion has been done for appropriate evaluation metrics.

1 はじめに

デジタル放送開始に伴って提供されているサービスとしてデータ放送がある。データ放送サービスのうち、特に、番組と連動してリアルタイムに番組関連情報を提供できる番組連動型サービスは、テレビの本番組をより魅力的にできるものとして期待されている。

しかし、放送規制やコンテンツ作成コストが高いため、提供コンテンツがかなり限定されていること

から、データ放送の利用者数は、一部の番組を除いて、なかなか伸びていないのが現状である。データ放送をより魅力的なものにするために、筆者らはこのような番組連動型情報サービスをインターネット上で実現するフレームワークの検討を進めている。

インターネット上の新しいサービスとして、Point Cast Network に代表されるプッシュ型情報配信サービス（90 年代後半）[1] が非常に多くの注目を集めた。これらのサービスの大部分はクライアントソフトによって自動的かつ定期的に情報配信サーバに対

して更新データをダウンロードしてユーザに自動的に情報提示するものであり、疑似プッシュ（スマートプル）と呼ばれる。本研究で提案する番組連動型情報サービスでは、放送番組と同期して番組関連情報もあわせて番組視聴者に提供するものであり、情報の自動提示という点では従来のプッシュ型情報配信に類似していると言える。しかし、従来型の配信方式であるスマートプルの欠点として、不必要なデータ問い合わせにより無駄なトラフィックの発生や、更新されたデータが即時にクライアントに提供できないなどの問題点がある。また、クライアントの数が増えた場合に対するスケーラビリティの問題が解決されていない。

現在、インターネット上で同報型通信を実現する技術として、マルチキャスト技術に期待が寄せられている。マルチキャストには、IP マルチキャスト [2] とオーバーレイネットワークなどを用いたアプリケーションレベルマルチキャスト [3, 4] がある。本研究の場合、通信のリアルタイム性を考慮すると、IP マルチキャストを用いた方が望ましいが、IP マルチキャストのアーキテクチャに本質的に起因する多数の問題で、インターネットへの広域な展開が為されていない問題がある [5]。そのため、ほとんどのインターネットユーザは IP マルチキャストを利用できないのが現状である。また、本提案サービスはサービスのパーソナル化を目指しているため、マルチキャスト上でこのようなことを実現することには限界があり、問題が多い。そのため、本研究ではユニキャストを用いる。本提案サービスの利用者数は、番組視聴率によっては 100 万もの規模があり得るため、本提案サービスを実現するには、大規模なサービスユーザに耐えるスケーラブルな同報型の情報配信機構を構築しなければならない。

そこで、本研究では、プッシュ型とプル型のそれぞれの情報配信機構を併用した、高いスケーラビリティを有するサービスシステムを提案する。具体的に、プッシュ型で配信するデータは、リアルタイム性の要求が高いもの（例えば、生放送番組に連動する特定のコンテンツに対する表示命令）に限定する。また、多くのコンテンツは本放送が開始する前に用意可能であることから、表示コンテンツは従来の HTTP によるプル型配信を行う。本稿では、提案サービスシステムの全体設計、ドラマなどのような録画放送番組に限定したプロトタイプシステムの実装につい

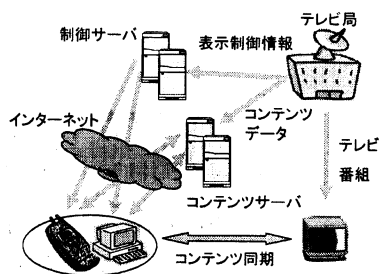


図 1: 番組連動型情報サービスの概略

て説明し、評価手法の議論を行う。

以下に本稿の構成について述べる。2 節では、本提案番組連動型情報サービスを実現するためのシステムの概要について述べる。3 節では、提案システムにおける配信系の設計について述べる。4 節では、提案システムのプロトタイプ実装について述べる。5 節では、システムのスケーラビリティに着目した評価項目について議論を行う。6 節では、実装が未完了である生放送と緊急放送の場合について解決すべき課題を明らかにし、今後の予定について述べる。

2 提案サービスシステム

2.1 番組連動型情報サービス

本研究で実現する番組連動型情報サービスの概略を図 1 に示す。本提案サービスはテレビで放送されている番組と、インターネット上で配信するコンテンツなどを同期させるものである。サービス利用者はパーソナルコンピュータや携帯電話、PDA などの情報端末を操作することで、本番組放送を視聴しながら同時に表示されるコンテンツもチェックすることができる。具体的に提供されるコンテンツは、たとえば CM における広告商品に関連する簡単な説明や Web サイトへのリンクなどを現在のところ想定している。将来的には、ネットワーク帯域の増加や配信システムの性能向上により、よりリッチな関連コンテンツの提供を目指す。

2.2 システムの概要

次に本提案サービスを実現するためのシステムの概略を図 2 に示す。このように、本提案システムはコンテンツ生成・管理、コンテンツ配信（サーバ）、コンテンツ受信（クライアント）からなる。本稿では、これらのうち、コンテンツ配信とコンテンツ受信に焦点をあてる。

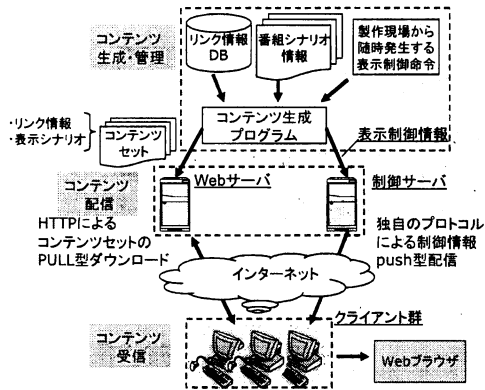


図2: 提案システムの概略

提案システムのデータフローについて説明する。まず、クライアント上に表示されるコンテンツは、URL データベースと番組シナリオによりパッケージングされ、そして、コンテンツサーバ (Web サーバ) に格納することによってダウンロード可能になる。表示コンテンツの表示開始終了時刻、コンテンツの識別子といった制御情報は、独自プロトコルにより制御サーバからクライアントに対して純粋にプッシュ配信される。制御情報の配信タイミングは番組前や番組放送中のような場合が考えられる。クライアントは、コンテンツサーバからダウンロードしたコンテンツと、制御サーバから受信した表示制御情報に基づいて自動的に Web ブラウザ上でコンテンツの表示を行う。以下にシステムの構成要素について説明する。

2.3 プッシュ型配信 (制御サーバ)

テレビ放送番組はたまに、ドラマなどの録画放送番組、ニュース番組、スポーツ中継などの生放送、緊急ニュース速報などの緊急放送に分けることができる。以下に放送番組のそれぞれの特徴について説明する (表1)。

録画放送の場合、録画放送番組は、放送開始時刻が変更されたり、番組途中で緊急ニュースなどの番組が挿入しない限り、あらかじめ決められたシナリオ通りに進行する。生放送は番組シナリオがある程度存在する場合とほとんど存在しない場合がある。番組シナリオが放送前に製作されれば、表示コンテンツを事前に用意することが可能である。そして、番組の進行にあわせてコンテンツの表示タイミングを時系列の前後調整だけで済む場合が多い。また、ス

ポーツ中継のようなシナリオが存在しない場合でも、チームや選手などの情報を放送前にある程度入手できるため、表示コンテンツ自体は事前に用意することが可能である。しかし、シナリオがないため、次にどのようなイベントが発生するのか事前に決定することができない。よって、コンテンツ表示イベントを任意のタイミングで生成できる手法が必要になる。緊急ニュース速報の場合はコンテンツを事前に用意できない、かつ表示制御情報とコンテンツは共に緊急度が高く、できるだけ早くユーザーに送信しなければならない。

本提案サービスでやり取りされるデータは、上述した放送番組のそれぞれの特徴を考慮して、リアルタイム性の高低に分類できる (表2)。本提案システムでは、リアルタイム性の高いデータについては制御サーバからクライアントに対して完全プッシュ型配信される。具体的には、制御サーバとクライアント間に TCP セッションを保持した上で、必要に応じて自由なタイミングでクライアントに対して送信する。プッシュ型配信により送られる表示制御情報には、コンテンツの識別子と表示開始終了時刻などが含まれる。これらの情報は、可能な限り小さくすることがシステムのスケーラビリティの確保のために重要である。たとえば、100 万人に対して、3 秒で 100 バイトのデータを配信することを考えると、プロトコルオーバーヘッドを除いても、266Mbps の帯域が必要となる。そのため、このプッシュ型配信はバイナリフォーマットによる独自プロトコルを設計し用いる。

2.4 プル型配信 (コンテンツサーバ&クライアント)

リアルタイム性の低いデータは Content Delivery Network (CDN) や P2P 技術といった従来のコンテンツ配信技術を活用できるように、従来のプル型の HTTP を用いた配信を行う。コンテンツサーバは通常の Web サーバを用いる。前節で説明したように、連動型コンテンツは事前に用意できることから、本提案システムでは、HTTP プロトコルのオーバーヘッドをできるだけ減らすために、一定時間分の配信コンテンツをパッケージ化してコンテンツサーバより配信する。ここで、パッケージされたコンテンツをコンテンツセットと呼ぶ。

本提案システムにおいて、放送番組の特徴にあわせて最適なコンテンツセットを作成することが望ましい。本稿では、録画放送番組に適したコンテンツ

表 1: 代表的なテレビ放送番組の特徴

番組	特徴
生放送	ニュース速報 コンテンツを事前に用意できない、表示緊急度が一番高い。
	スポーツ中継 コンテンツを事前に用意できる、次の表示イベントは事前に決定できず、細粒度の表示調整を行う。
	一般ニュース コンテンツを事前に用意できる、番組シナリオにより、時系列への前後表示調整を行う。
非生放送	ドラマ コンテンツと表示制御情報を事前に用意できる。

表 2: リアルタイム性の高いデータと低いデータ

テレビ番組	リアルタイム性高いデータ	リアルタイム性低いデータ
緊急放送 (災害時のニュース速報など)	緊急メッセージ	コンテンツ表示制御情報
シナリオのある生放送番組 (ニュースなど)	表示制御調整情報	表示コンテンツ
シナリオのない生放送番組 (スポーツ中継など)	表示制御情報	表示コンテンツ
非生放送番組 (ドラマなど)	放送時間の変更	表示制御情報・表示コンテンツ

セットの構造 (図 3) について述べる。図のように、コンテンツセットは基本的に階層構造になっており、一つのコンテンツセットにはさらに複数のコンテンツセットの情報 (上位コンテンツセット (親) の識別子、自分自身の識別子、表示開始終了時間、下位コンテンツセット (子) の識別子) を、最下位のコンテンツセットには具体的な表示コンテンツ情報 (表示コンテンツ識別子、表示開始終了時間、テキスト、リンクなどのコンテンツ情報) を格納している。これらの情報はすべて XML の形式でファイルに記述されるものとする。クライアント側では必要に応じて XML ファイルを解析したり、ブラウザに表示できるよう加工したりする。

一方、番組連動型コンテンツ配信では、クライアントが同期してサーバに対してダウンロードを行う可能性が高いため、サーバへの負荷の一極集中を避ける手法を導入する必要がある。前節で述べたように、放送番組はそれぞれ番組の特徴があり、その特性に応じて一極集中を回避するダウンロード手法を適用しなければならない。3.2 節で録画放送の場合のダウンロード手法について詳説する。

3 配信系の設計

図 2 に示したように、本提案システムの配信系の部分はコンテンツ配信 (サーバ)、コンテンツ受信 (ク

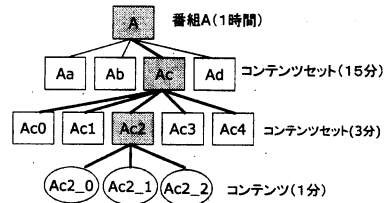


図 3: 録画放送の場合のコンテンツセット構造

ライアント) からなる。プッシュ型配信する際、システムのスケーラビリティを確保するためには、配信される表示制御情報などのデータサイズは可能な限り小さくすることが重要である。また、プル型によるコンテンツのダウンロードを行う際、クライアント主導による最適な時間的分散を行うダウンロード手法もアクセスの一極集中を避けるために重要である。以下にプッシュ型配信に用いられるプロトコルと時間的分散を行うダウンロード手法の詳細設計について述べる。

3.1 プッシュ型配信

3.1.1 データフォーマット

制御サーバからクライアントへのプッシュ配信に用いられるプロトコルのデータフォーマットについて述べる。ここでは、リアルタイム性の低い場合に分類される録画放送番組に適したデータ構造 (図 4)

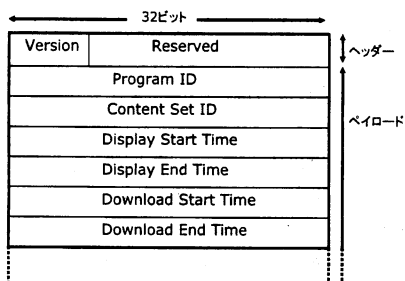


図 4: 録画放送の場合のデータ構造

について説明する。

まず、ヘッダー部は 4 バイトの固定長フィールドを持つ。そして、ペイロード部は一つのコンテンツセット情報サイズ (24 バイト) を基本単位とし、個々のコンテンツセット情報をリスト構造により可変数構成することが可能である。ペイロード全体のサイズは 24 バイト × コンテンツセット数になる。

2.3 節で述べたように、録画放送番組の場合、放送開始時刻が変更されたり、番組途中で緊急ニュースなどの番組が挿入しない限り、あらかじめ決められたシナリオ通りに番組を進行する。そのため、表示シナリオが変更されない場合では、クライアントのサービス開始時に、即時表示すべきコンテンツセットの情報をクライアントに対してプッシュ型配信すればよい。その後は、クライアント側で主導的にコンテンツダウンロードを制御することが可能である。また、放送時刻の変更があった場合は、即座に変更情報をクライアント側に送信し、その後は表示タイムテーブルを書き換えて変更を反映するようにする。

3.2 コンテンツダウンロードの時間的分散

コンテンツダウンロード時のサーバへの負荷の集中を防ぐために、クライアントは決められた時間内に時間的な分散を行う。ダウンロードの時間的分散の概念図を図 5 に示す。ダウンロードの開始タイミングはダウンロード開始可能時間から最終期限までの間にランダムに決定される。もし、最終期限を過ぎてしまっている場合は即時ダウンロードを開始する。実際のダウンロード手法は番組の特性に応じて考慮する必要がある。ここでは、録画放送の場合のダウンロード手法について説明する。

録画放送の場合、すべてのコンテンツセットは表示開始時刻と表示終了時刻があらかじめ決定されている。図 6 は図 5 のコンテンツセットの表示時間の部

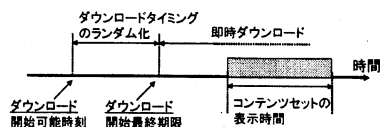


図 5: コンテンツダウンロードの時間的分散

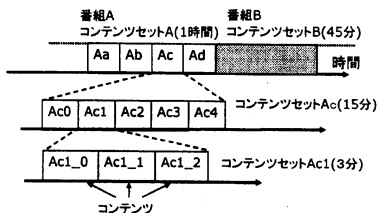


図 6: 録画放送のダウンロード手法

分を詳細化したものである。ここでは録画放送番組 A と B を想定している。図 6 に示すように、今クライアントの接続時刻がちょうどコンテンツ Ac1_0 の表示時間内であると仮定すると、制御サーバはクライアントに対してコンテンツセット Ac1, Ac2, Ac3, Ac4, Ad の表示制御情報をプッシュ配信を行う。そして、クライアント側では受信した制御情報に基づき、今すぐ表示しなければならないコンテンツセット Ac1 の即時ダウンロードを行う。そして、ダウンロードしたコンテンツをブラウザに表示する。また、それと並行して、次のコンテンツの表示開始時刻を考慮し、バックグラウンドで番組 A の残りのコンテンツセットを即時ダウンロードか、時間的分散ダウンロードをするようスケジューリングを行う。このように、ダウンロードタイミングは制御サーバからのヒントをもとに、すべてクライアント側のみで決定することで、処理の効率化を図る。

4 プロトタイプシステムの実装

前節で述べたシステムの設計モデルに基づき、録画放送番組の場合についてプロトタイプシステムの実装を行った。実装では OS に Linux (kernel2.4.20) を使い、C 言語とライブラリには pthread, XML ファイル解析に libxml2 を用いた。

4.1 実装の概要

プロトタイプシステム全体は大きくサービス諸機能を提供するサービススレッド群と、通信機能を提供する通信スレッドに分かれる。制御サーバとクラ

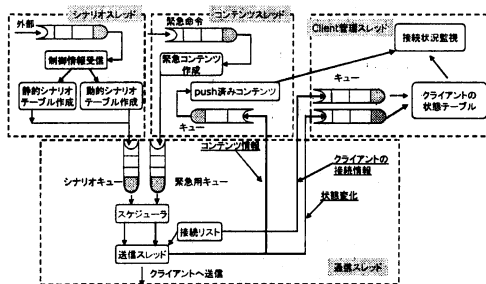


図 7: 設計の詳細 (制御サーバ)

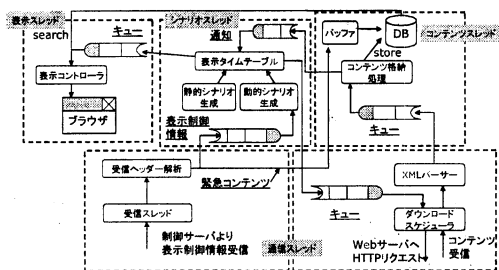


図 8: 設計の詳細 (クライアント)

クライアントの設計の詳細をそれぞれ図 7 と図 8 に示す。

4.1.1 制御サーバ

制御サーバは大きくサービス処理を行う部分と通信処理を行う部分に分けられる。ここで、プロトタイプ実装した制御サーバの各モジュールについて述べる。各モジュールはイベント駆動型で独立的に動作している。

サービススレッド群

サービススレッド群はシナリオスレッド (今回は静的なシナリオのみについてしか考慮していない)、コンテンツスレッド、Client 管理スレッドで構成されている。

シナリオスレッドは外部より発生したシナリオの受信イベントによりシナリオテーブルを作成する。テーブルのデータ構造は今回ヘッダーフォーマットを考慮してリスト型を用いた。もし、クライアントから接続がきた場合、接続情報が接続リストに追加され、接続時刻により、現在放送中の番組に関連するコンテンツセットの表示制御情報をクライアントに対してプッシュ配信する。プッシュ配信されるデータ量は現時点でのシナリオテーブル上に存在する情報の合計サイズとする。そして、次のイベント発生

(新たなシナリオ情報到着) までシナリオスレッドは待機する。

client 管理スレッドは接続してきたクライアントの状態管理を行う。上で述べたように、本サービスの利用者数は場合によって 100 万人以上の規模もあり得るため、効率的な状態管理手法を実現することが重要である。今回の実装では、IP アドレスのツリー構造によって個々のクライアントを特定できるようにし、そして、ツリーの各末端ノードにクライアントの接続情報 (ポート番号、ファイルディスクリプタ、フラグなど) を一意的に関連づけを行う方法を用いた。また、接続が終了した場合、クライアントに関する情報を状態テーブルから開放する。

イベント管理

今回の実装コストを考慮して、図 7 に示すように、サービスモジュールから通信モジュールへのリクエストの通知はリクエストキューを通じて直接やり取りする手法を用いた。こちらはクライアント側の実装においても同様に適用される。

通信スレッド

通信インタフェースとして、UNIX の標準的なソケットインタフェースを使用している。また、リクエストキューには優先度付けをし、サービススレッドからのリクエストの緊急度に応じてリクエストキューへの振り分けを行う。こうすることで、放送局からの緊急命令を割り込ませることが可能になる。通信スレッドは基本的にクライアントからのリクエストを受け付けないことで、制御サーバのスケラビリティを高める。

4.1.2 クライアント

ここで、プロトタイプ実装したクライアントの各モジュールについて述べる。

サービススレッド群

サービススレッド群は表示スレッド、シナリオスレッド (今回は静的なシナリオのみについてしか考慮していない)、コンテンツスレッドで構成されている。

表示スレッドはシナリオスレッドで作られる表示タイムテーブルに基づき、適切なタイミングで番組関連コンテンツを web ブラウザ上で表示する。シナリオスレッドにおける表示タイムテーブル作成方法は制御サーバの場合と同様に適用できる。表示コンテンツの例外に緊急ニュース速報情報の表示がある。この場合は、現在の表示コンテンツを一旦退避させ

るなどの工夫が必要になる。また、表示コンテンツは専用のコンテンツデータベース（DB）で一元管理されることが望ましいが、今回の実装では個々のファイルで管理しており、DB 構築は今後の課題とする。

通信スレッド

通信スレッドは制御サーバからの制御データ受信スレッドと HTTP によるダウンロードスレッドによって構成されている。表示コンテンツがどこまでダウンロードしたかといったステータスや、つぎにダウンロードすべき項目のスケジューリングはすべてダウンロードスレッドによって自己管理される。このような独立性を持たせることで、制御サーバへの負担を軽減でき、システム全体のスケーラビリティを高めることができる。

5 評価手法

本提案サービスシステムはある保証された時間内に配信でき、かつ大規模なサービスユーザに耐える高いスケーラビリティを実現しなければならない。以下に本提案システムの性能を表す評価項目について述べる。なお、実際の評価は今後の課題である。

- 応答遅延分布（クライアント）

クライアントが接続を開始してから、コンテンツを受信してブラウザに表示するまでの時間分布を示す。大きく制御サーバから表示制御情報の受信時間と Web サーバからの表示コンテンツ受信時間に分けられる。応答遅延時間分布はシステム全体の性能の許容範囲を表す主要な指標であると言える。特に、初回接続時の場合や途中での緊急放送の割り込みが発生する場合の応答遅延時間が重要である。

- 受信成功率（クライアント）

配信されたコンテンツ全体の正しく時間内に受信できた割合を示す。これは特に、システムの過負荷時において、応答遅延時間と合わせて、システムの性能の最大許容範囲を明らかにする重要な指標となる。

- ネットワークの消費帯域（制御、Web サーバ）

本提案システムはユニキャストを用いているため、システムの性能はサービス利用者数の著しい増加により大きく影響される可能性がある。したがって、限られたネットワーク帯域の消費を計測することで、本提案サービス

の同時利用者数をどこまでサポートできるかをあきらかにすることができる。

- クライアントの接続状態管理（制御サーバ）

接続数の変化に応じて、クライアントの接続状態を正しく管理できているかを示す。接続状態には、接続時間、受信状態（受信済、待機中、終了）などがある。これらはクライアントの接続状態をモニタリングするために計測する必要があると考えられる。

6 今後の課題

今回のプロトタイプ実装では、ドラマなどの録画放送番組の場合のみにしか考慮していない。録画放送以外に生放送や緊急放送について考慮しなければならない。今後はこれらすべての場合を考慮したシステムの実装を行い、現実のインターネット環境を模擬した環境で本提案システムの性能評価を行う予定である。

6.1 生放送の場合についての考慮

6.1.1 シナリオがある場合

生放送の場合、放送スケジュールは予定より常に変化する可能性があるため、番組のシナリオある、ないに関係なくコンテンツの表示制御情報を番組の進行に合わせて逐次ブッシュ配信する必要があると考えられる。それに対して、表示コンテンツ情報はそれぞれの場合について検討する必要がある。番組シナリオが放送前に製作されれば、表示コンテンツを事前に用意することが可能である。したがって、この場合、コンテンツの配信手法は基本的には録画放送の場合と同様に扱える。

6.1.2 シナリオがない場合

スポーツ中継のようなシナリオが存在しない場合でも、チームや選手などの情報を放送前にある程度入手できるため、表示コンテンツ自体は事前に用意することが可能である。しかし、シナリオがないため、次のようなイベントが発生するのか予測することができない。よって、コンテンツ表示イベントを任意のタイミングで生成できる手法に合わせて、クライアント側における最適なコンテンツダウンロードスケジュール手法を検討しなければならない。

6.2 緊急放送の場合についての考慮

ニュース速報の場合は、何秒か以内にコンテンツを表示しなければならないため、制御情報と共にコ

ンテンツデータ自体も一緒に制御サーバからクライアントへと直接プッシュ配信される。この場合、システムのスケーラビリティを高めるために配信されるデータサイズ（ヘッダーを含めて）をできるだけ小さくするといった工夫をしなければならない。

7 関連研究

関連研究である [6] は放送番組に連動するインターネット利用サービス提供手法について検討している。これはあらかじめ作られた台本による想定されたシナリオに対してのみの番組連動型サービスであり、番組台本の変更が生じた場合や生放送などリアルタイム性の高い事象を考慮していない。また、システムのスケーラビリティを高める手法 [7, 8] を提案しているが、大規模なサービスユーザに対してサービスを提供できるかは疑問に残る。これに対して、本研究では、放送番組において想定しえるすべての事象を考慮しており、実際の番組放送現場と親和性の高い番組連動型サービス、かつ高いスケーラビリティを有するサービスシステム構築を目指している。

8 まとめ

本論文では、インターネット上でテレビの本放送番組と連動した情報配信サービスを実現するための情報配信機構の提案を行い、提案システムの全体設計、録画放送の場合のプロトタイプ実装の詳細に関して説明した。そして提案システムの性能評価手法に関して議論を行った。

本提案システムは、従来のプッシュ型とプル型のそれぞれの基本技術を併用することにより、高いスケーラビリティを実現する。具体的に、プッシュ型で配信するデータは、リアルタイム性が高いものに限定し、同報的な情報配信のオーバーヘッドを可能な限り小さくする。リアルタイム性が低いコンテンツ配信は、従来の HTTP によるプル型配信を行う。その際、サーバへのアクセスの集中を回避するために、ダウンロードタイミングの時間的分散を行う機構を導入する。また、従来の CDN や P2P 技術といったコンテンツ配信技術を併用することを可能にすることで、システム全体のスケーラビリティを高める。今後はシステム機能を強化した本実装を行い、システムの性能評価を通じて、本提案サービスの実現可能性について検討していきたい。

参考文献

- [1] Manfred Hauswirth and Mehdi Jazayeri. A Component and Communication Model for Push Systems. *ESEC/FSE 99*, September 1999.
- [2] S. Deering. Multicast Routing in Internetworks and Extended LANs. *Proc. of ACM SIGCOMM'88*, Vol. 14, No. 1, August 1988.
- [3] M. Castro, P. Druschel, A. M. Kermarrec, and A. Rowstron. SCRIBE: A large-scale and decentralized application-level multicast infrastructure. *IEEE Journal on Selected Areas in communications (JSAC)*, 2002.
- [4] D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel. ALMI: An Application Level Multicast Infrastructure. *Proc. 3rd USENIX Symposium on Internet Technologies & Systems*, March 2001.
- [5] C. Diot, B. N. Levine, B. Lyles, H. Kassem, and D. Balensiefen. Deployment Issues for the IP Multicast Service and Architecture. *IEEE Network*, Vol. 14, No. 1, January 2000.
- [6] 中川俊夫, 中岡範之, 山本真. テレビ視聴しながらのネット利用”に適したコンテンツナビゲーションサービスシステム. 信学総大 B-16-27, p. 636, March 2004.
- [7] 中岡範之, 中川俊夫, 山本真. 放送連動インターネットサービスにおける時間方向アクセス負荷分散手法の検討と評価. 信学技報 CQ2004-56, July 2004.
- [8] 中岡範之, 中川俊夫, 山本真. サーバー負荷軽減を考慮した時刻同期コンテンツ提供手法の検討. 信学総大 B-16-26, p. 635, March 2004.