

インターネットラジオにおけるプライベートチャンネルとコンテンツ配信の分散の提案

2G-03

市川 快† 荒川 健介† 和野 恵介† 村山 優子†

†岩手県立大学 ソフトウェア情報学部

1 はじめに

World-wide Web(WWW) はインターネット上のマルチメディア通信の基盤として機能している。インターネットラジオは WWW アプリケーションであり、最近広く普及し始めている。

本研究では 2000 年 4 月よりインターネットラジオサービス Flip Over Radio(略称 FOR) を提供し、岩手県立大学の学生が運営している [1]。本研究の目的はインターネットラジオ局を実際に運営し、問題点を見つけ改善することである。さらに、インターネット上において柔軟で快適な音楽配信サービスを実現することである。ここで言う柔軟とは、これまでのインターネットラジオ局のように、ユーザが予め用意されたサービスだけを利用するのではなく、自分の好みにあったサービスを利用することである。また、快適とはコンテンツがストリーミング配信される際に、音楽などを途切れずに配信するシステムを提供するという意味である。

本稿では、プライベートチャンネルおよびコンテンツ配信の分散の提案を行う。プライベートチャンネルとは、ユーザの好みに応じて音楽を配信するというインターネットラジオの新しいサービスである。

2 プライベートチャンネル

本節ではプライベートチャンネルの提案として、先ず現状のシステムを紹介し、次にプライベートチャンネルの設計について述べる。

現在、インターネットラジオの配信サービスには 2 つの形態がある。ユーザ全員が同じ一つのチャンネルを聞くものと、一方は一つのインターネットラジオ局がジャンルごとに個別のチャンネルを複数用意し、ユーザはその複数あるチャンネル (マルチチャンネル) の中から好きなものを選ぶものである。現在 FOR は前者のサービスを行っている。

FOR では、ユニキャストでユーザと接続している。従ってそれぞれのユーザごとに専用のチャンネルを作り、違う音楽を配信することも可能である。そこでマルチチャンネルの技術を応用してプライベートチャンネルを開発することにした。

2.1 現在の FOR のシステム

FOR で使用しているインターネットラジオアプリケーションは主に Icecast[2] と Shout[2] である。Icecast はインターネット上でユーザに MP3[3] 形式の音楽データを配信し、Shout はプレイリストと呼ばれるテキスト形式の楽曲のリストに従い Icecast に mp3 の音楽データを送る。

Shout には Mountpoint と呼ばれるチャンネルを生成する機能が備わっている。動的にチャンネルを作る為にこの Shout の機能を利用する。

2.2 プライベートチャンネルのシステム設計

システム設計は現在のところ図 1 の通りで考えている。サーバのデータベースにはユーザ、音楽家、楽曲情報などが登録されている。ユーザのプライベートチャンネルへの登録処理は WEB インターフェースによって提供される。

ユーザは WEB 上から自分専用のチャンネル名 (チャンネル ID) や好みの音楽情報を登録する。選曲データベースエンジンにチャンネル ID と好みの音楽情報が送られ、どの曲を配信するかをリストにしたプレイリストを作成する。その後プレイリストとチャンネル ID を makechannel と呼ばれるチャンネル生成プロセスに渡し、Shout を使いユーザの専用チャンネルを作成する。選曲された音楽データは Shout によりストリーミングサーバに送信され、そこからインターネット上に音楽配信が行われる。チャンネル生成時に makechannel は、ユーザがアクセスする自分専用チャンネルの URL (例: <http://radio.comm.soft.iwate-pu.ac.jp:8000/UserA/>) を返し、WEB 上にそれを表示する。ユーザは MP3 プレイヤを使いその URL にアクセスし音楽を聞く。ストリーミングサーバとは、ここでは Icecast を用いる。

プライベートチャンネルで重要なのは、選曲である。ユーザの履歴情報や感性情報検索の処理 [5] を参考にしながら、設計を進めたい。

“A proposal of the private channel and distributed contents streaming in the Internet radio”

Yasushi Ichikawa, Kensuke Arakawa, Keisuke Wano and Yuko Murayama

†Faculty of Software and Information Science, Iwate Prefectural University

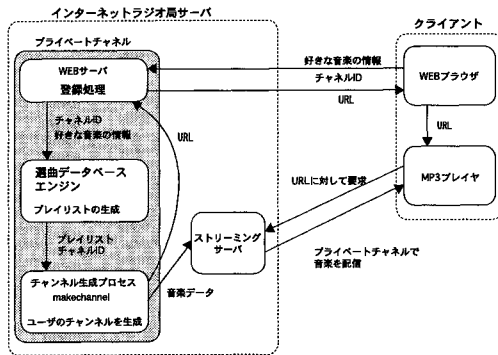


図1: プライベートチャネルの設計図

3 コンテンツ配信の分散

音楽データのような実時間型の連続データを配信する場合、問題となりやすいのはインターネット回線の速度である。それは、音楽の場合音声の途切れやノイズにつながるからである。

この問題を解決するために、インターネットラジオの音楽配信の分散化を考えている。局の音楽配信データを分散サーバに送り、複数のサーバや回線から配信出来るようにする。

分散配信は icecast の中継 (relay) 機能により実現できる。今後は複数の中継サーバを設置し、ユーザにより近く回線が太い中継サーバを自動で選択しつながるようにしていきたい。関連技術として CDN (Contents Distribution Network) がある [4]。CDN ではインターネット上の異なった場所に分散サーバを置き、そこにオリジナルのサーバのコンテンツなどをキャッシュまたは中継する。ユーザから一番近い分散サーバへ自動的にアクセスさせるリクエストルーティングの技術により負荷を分散する。

リクエストルーティングには、クライアント選択型、サーバ選択型、中間システム選択型がある [6]。クライアント選択型では、クライアントが複数のサーバから任意の一台を選択する。クライアントには負荷分散に必要な機能が備わっている必要がある。サーバ選択型では、要求を受け取るサーバが状態に応じて複数のサーバに要求を転送する。柔軟性は高いが、要求を受けるサーバがボトルネックになる場合があり、規模の大きさに限界がある。

中間システム選択型では、複数のサーバに同一 IP を付け、選択をネットワークの経路制御にまかせるものや、DNS を用いて一つのホスト名に複数 IP を付け、選択するものなどがある。DNS は、ほぼ全てのアプリケーションが利用するため汎用性があるが、DNS サーバはユーザのアドレスを知ることはできないため、ユーザの利用しているローカル DNS のアドレスを近似値として判断せざるを得ないなどの問題がある [7]。

今後はこの CDN の技術を使い、インターネットラジオへ応用していきたい。一案として、プライベート

チャネルと組み合わせ、ユーザの好みの音楽だけを一番近いサーバから配信させることも可能である。実現方法としては、以下の方法が考えられる。オリジナルサーバのコンテンツを分散サーバにキャッシュする。プライベートチャネル専用サーバを用い、ユーザのリクエストをもとにプレイリストを作成し、プレイリストを分散サーバに送る。分散サーバはそのプレイリストをもとにユーザへ音楽配信を行う。

ユーザは最初に、好みの音楽情報やチャンネル名の情報を送るため、プライベートチャネル専用サーバにアクセスする。プライベートチャネル専用サーバは、ユーザからのアクセスの後にサーバ選択型のリクエストルーティングを行い、ユーザから一番近い分散サーバを選択し、プレイリストをその分散サーバへ送る必要がある。

4 まとめ

本稿では、インターネットラジオ局を例題としてし、それぞれのユーザに対して好みの音楽を配信するプライベートチャネルの提案と、音楽配信のストリームデータを中継サーバを使う事によって分散させる提案をした。今後はプライベートチャネルとコンテンツ配信の分散を組み合わせ、ユーザに一番近いサーバからユーザの好みの音楽を配信していきたい。

参考文献

- [1] Flip Over Radio : <http://radio.comm.soft.iwate-pu.ac.jp> †
- [2] Icecast and Shout : <http://www.icecast.org> †
- [3] Fraunhofer Research : <http://www.iis.fhg.de/amm/> †
- [4] Balachander Krishnamurthy, Craig Wills and Yin Zhang, On the Use and Performance of Content Distribution Networks, ACM SIGCOMM INTERNET MEASUREMENT WORKSHOP 2001
- [5] 三石大, 佐々木 淳, 船生 豊: ユーザの利用履歴を利用した動的なインデックス半自動生成手法の提案, 情報処理学会研究会報告 2000-DBS-121, Vol.2000, No.44, pp.53-60(2000).
- [6] 下川俊彦, 吉田紀彦, 牛島和夫: ネームサーバを用いた柔軟な負荷分散, インターネットコンファレンス '99, 107-116 (December,1999).
- [7] A. Barbir, B. Cain, F. Douglass, M. Green, M. Hofmann, R. Nair, D. Potter, O. Spatscheck: Known CDN Request-Routing Mechanisms, draft-cain-cdn-request-routing-04.txt(Nov 2001)

† 最終アクセス日: 2001 年 12 月 15 日