

## 再生統計情報に着目したストリーミング品質評価方式とその効果

大浴 孝治<sup>†</sup> 山本 文治<sup>††</sup> 中山 雅哉<sup>‡</sup> 門林 雄基<sup>‡‡</sup>

<sup>†</sup>株式会社PFU 〒105-0123 東京都港区山田 1-2-3

<sup>††</sup>株式会社アイアイジェイメディアコミュニケーションズ 〒101-0051 東京都千代田区神田神保町 1-105

<sup>‡</sup>東京大学 〒113-8658 東京都文京区弥生 2-11-16

<sup>‡‡</sup>奈良先端科学技術大学院大学 〒630-0101 奈良県生駒市高山町 8916-5

E-mail: <sup>†</sup>ooeki@pfu.fujitsu.com, <sup>††</sup>bunji@iij-mc.co.jp,  
<sup>‡</sup>nakayama@nc.u-tokyo.ac.jp, <sup>‡‡</sup>youki-k@is.aist-nara.ac.jp

**あらまし** 本稿では従来のサーバ監視方式やネットワーク監視方式では達成できなかったストリーミングの品質評価方式と品質監視装置を提案する。提案方式はこれまで目視によって行われてきた主観的な監視を、プレイヤーから取得できるフレームレートの数値に置き換えるものであり、その根拠を示す。提案する品質監視装置は擬似プレイヤーとして動作し、再生時の品質データを数値化できる。この品質監視装置について改良を加え、実環境下での実験を通じて本提案方式の有効性を示す。

**キーワード** ストリーミング, 受信品質, アクティブ型品質監視, フレームレート

## A streaming quality evaluation method based on player's statistics information, and its effect

Koji OOEKI<sup>†</sup> Bunji YAMAMOTO<sup>††</sup> Masaya NAKAYAMA<sup>‡</sup> and Youki KADOBAYASHI<sup>‡‡</sup>

<sup>†</sup>PFU Limited 1-2-3 Yamada, Minato-ku, Tokyo, 105-0123 Japan

<sup>††</sup>IJ Media Communications Inc. 1-105 Kanda Jinbo-cho, Chiyoda-ku, Tokyo 101-0051 Japan

<sup>‡</sup>The University of Tokyo 2-11-16 Yayoi-ku, Bunkyo, Tokyo 113-8658, Japan

<sup>‡‡</sup>Nara Institute of Science and Technology 8916-5 Takayama, Ikoma, Nara 630-0101, Japan

E-mail: <sup>†</sup>ooeki@pfu.fujitsu.com, <sup>††</sup>bunji@iij-mc.co.jp,  
<sup>‡</sup>nakayama@nc.u-tokyo.ac.jp, <sup>‡‡</sup>youki-k@is.aist-nara.ac.jp

**Abstract** Health check of server is important and many methods have been developed. Previous methods are based on layer 3 or layer 4, however, and they are not suited for streaming applications. Meanwhile, previous methods only provide periodic monitoring function but streaming applications need continuous monitoring function. Furthermore, CDN operators had to watch streaming program and confirm the health check of quality by themselves. In this paper, we propose a new streaming quality evaluation method based on player's statistics information and present its evaluation results. It uses frame-rate information of pseudo-client and it fairly matches user-perceived quality.

**Keyword** Streaming, Receiving Quality, Active quality monitoring, Frame rate

### 1. はじめに

インターネットが普及し、職場や家庭でも広域なネットワーク環境が利用可能になるにつれ、VoD やライブによるストリーミングの大規模配信

が実用になってきた。単一の ISP 内のユーザに対するストリーミング配信では Content Delivery Network(CDN)を ISP 内で構成してユーザを単一の CDN に誘導する技術を用いればよいが、複数の ISP のユーザを対象とした大規模配信においては、

それぞれのISP内で構成されるCDN同士をピアリング(接続)することで、広範囲にストリームを配信するContent Routing Network(CRN)アーキテクチャを構成する必要がある。筆者らが活動を進めているCRN Forum[1]では、ストリーミングの大規模配信技術に関する各種の研究を行っており、ユーザに提供するストリーミングの受信品質の向上技術について考察する受信品質部会の調査結果では、CDNをピアリングした際の親サーバから子サーバへのストリーミングフィードの品質の確保と監視の重要性が明らかになってきた。これは、親サーバから子サーバへのフィード部でストリーミングの品質が低下するファクターが存在すると、子サーバの先のすべてのユーザに対する受信品質の低下が発生してしまうためである。

そこで我々は、クライアント側の再生品質を監視する新たな方式の検討を進めている。本稿では、ストリーミングにおける品質評価としてのクライアント主導型の監視方式の概要とその有効性について紹介する。

## 2. 既存のサーバ監視方法に対するストリーミングにおける品質評価としての問題

インターネット上に設置されたサーバの動作状況を遠隔から監視する手法として、これまではネットワーク層(Layer 3)でのICMPによる監視、トランスポート層(Layer 4)でのサービスポート監視やサービスプロトコルを用いたサービス機能監視などが広く用いられてきた。

ICMPを使った監視では、監視対象のサーバにecho request messageを送出し、そのecho reply messageのステータスによって監視対象の状態を判定する方法をとる。また、サービスポート監視では、サーバで動作するサービスプロトコル(HTTPやRTSPなど)に動作チェック用のパケットを送出し、その応答status codeや一定のトランザクション結果から監視対象の状態を判定する。これらの手法による代表的な監視ツールには、Hewlett-Packard社のOpenView[2]、Ethan Galstad氏によるNagios[3]などがある。

これらの監視手法では、間歇的に監視対象となるサーバへアクセスを行って、その瞬間毎のサーバの状態を監視していることになるが、ストリーミングにおける品質評価の観点では、ある時刻のサーバの動作状況を監視するだけでは十分でなく、ストリーミングの受信開始から受信終了まで連続して受信者にパケットが到着しているかを

監視する必要がある。即ち、アプリケーション層(Layer 7)での監視を必要とする。

## 3. ストリーミングに対する監視方法

前節で述べたように、ストリーミングにおいては、「受信品質の達成度」の把握により総合的な監視を達成できる。これはストリーミングが連続的にパケットをサーバからクライアントへ送信することで成立する技術であるからである。受信開始から受信終了に至るまでの監視を行わないと、受信側での「再生フレームスキップ」や「リバッファリング」がいつどこで発生したのか把握することができない。

これを達成するには以下の二つのアプローチが考えられる。

1. ネットワークに流れるパケットに着目した品質測定および監視(パッシブ監視方式)
2. ストリーミングを受信し実際の再生状況に着目した品質測定および監視(アクティブ監視方式)

以下、それぞれの手法について特徴を述べる。

### 3.1. パッシブ監視方式

パッシブ監視方式は、サーバとクライアントで送受信されるパケットをタップやミラーポートなどを用いてネットワーク上でキャプチャし、該当するストリーミングデータをピックアップして解析をおこなう方法である。TCPの挙動やストリーミングプロトコルの制御パケット(method)を観測し、これらの振る舞いやメッセージ内容から対象システムの状態を推定する。

この方法の利点は観測システムをサーバやクライアントに依存せず独立して設置できることにある。サーバやクライアントに観測システムを導入する手法では、観測システムが観測対象に干渉や影響を及ぼす可能性があるが、本方式では経路上の機器でパケットをモニタできるため、サーバやクライアントの負荷を上昇させることもない。また、キャプチャしたパケットを取捨選択しながら情報の再構成が実施できるため、観測システムの応用性に富んでいる。

### 3.2. アクティブ監視方式

アクティブ監視方式は、ストリーミングプロト

コルを解すクライアントソフトウェアを用意して、サーバに接続し受信状況を直接監視する方法である。

クライアントソフトウェアは、ユーザが利用するプレイヤー、例えば RealPlayer や Windows Media Player を稼働させ、目視での主観評価を実施しながらプレイヤーから一定のデータを取得する。クライアントから取得できるデータには以下の種類の情報がある。

- 受信プロトコル
- エンコード時のビットレート
- 受信時のビットレート
- エンコード時のフレームレート
- 実際に再生できているフレーム数
- スキップフレーム数
- 受信したパケット数
- 再送されたパケット数
- 喪失されたパケット数

この方式のメリットは、実際にサーバからのストリーミングを受信することでアプリケーション層のデータを正確に取得することができることにある。これらのデータを、受信ストリーミングの主観評価と比較評価することが容易に実施できる。

### 3.3. アクティブ監視方式によるプロトタイプ実験

先に示した様に、パッシブ監視方式はサーバやクライアントに依存せず独立した観測システムを設置できるため有効な方法であるが、ネットワーク上を流れる制御パケットからクライアントの状態を推定するため実際のストリーミングの品質評価との相関を取ることが困難な状況にある。

そこで本稿では、アクティブ監視方式を用いたプロトタイプ実験を行い、クライアントから取得できるデータと目視による主観評価の比較を行い、最も一致度の高かったフレームレートの変化による配信品質監視の有効性の検証を行った。

プロトタイプ実験では、図 1 に示す様に Windows Media エンコーダ 9 および、Windows Media サービス 9 を使用してライブ配信を行い、同一 L2 スイッチに接続した監視端末で擬似クライアントを動作させて定期的にフレームレートを取得するとともに視聴端末で目視によるストリーミングの主観品質評価を実施する。この環境で負荷端末上から Windows Media Load Simulator

の負荷を掛けることで過渡的な過負荷状態を生成し、視聴端末および監視端末での品質評価を行った。

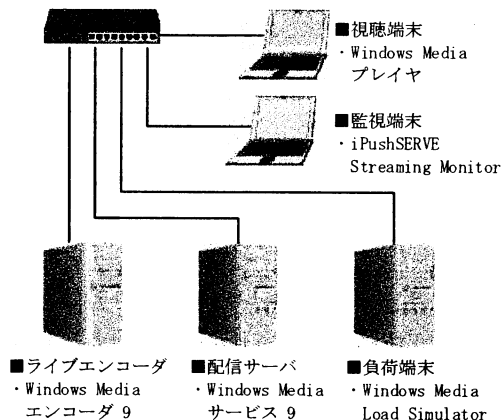


図 1 プロトタイプ実験の構成

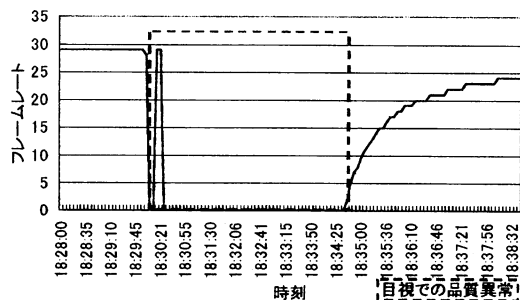


図 2 プロトタイプ実験結果

図 2 は、ストリーミングの制御プロトコルに RTSP を使い、データ配信に UDP を使用した時の実験結果を示している。実線は監視端末で取得した 10 秒毎の各時刻のフレームレート値を、斜線のブロックは視聴端末での目視による品質異常と判断した時間帯を表している。この実験では、負荷端末からの負荷は 18:30 から 18:35 までの 5 分間である。この実験結果から、監視端末で取得したフレームレート値は、サーバの負荷が増した時の視聴端末の受信品質劣化を短時間で把握するのに適していることがわかる。しかし、フレームレート値は受信品質が一旦劣化すると、サーバの配信状態が正常に復旧した後も正常なフレームレート値に復帰するのに時間がかかるとい

う問題があることが確認できた。これは、フレームレート値が過去一定時間の平均値として計算されて扱われるためである。

#### 4. 改良型ストリーミング品質評価方式

前節のプロトタイプ実験結果を元に、監視端末上で動作させる監視ツールをPFU社のストリーミング監視・管理ソフトウェア iPushSERVE Streaming Monitor[4] をベースに改良を行った。

同製品は、擬似プレイヤーとして動作し、Windows Media サービスや RealServer からのストリーミングを受信・再生しながら各配信サーバの稼動および配信品質を監視する機能が提供されている。擬似プレイヤーは SDK 経由で再生統計情報が取得でき、予め指定したしきい値と比較することで監視者に警告する機能を有する。擬似プレイヤーの機能を持つため、モニタ上での実際の描画は行わないが、プレイヤーのコーデックを利用しているため、描画した結果と変わらぬ統計データの取得が可能となる。

同製品の特長を以下に、監視項目を表 1、表 2 に示す。

1. 擬似プレイヤーで実際の監視対象ストリームを受信・再生することで監視を行うため、実際の視聴者観点で品質を監視することができる。
2. SDK 経由で取得した再生統計情報を逐次ロギングできるため、受信品質の遷移を把握することができる。
3. ストリーミングの制御に任意のプロトコル(MMS, RTSP, HTTP)が選択できる。またデータのトランスポートにも TCP, UDP, HTTP が指定できる。
4. アラートパスとして一般的に使用されている、SMTP, snmp-trap を標準装備していることに加え、任意のコマンド呼出にも対応しているため、他システムとの連携が容易である。

表 1 Windows Media 監視項目一覧

監視項目	説明
Availability	稼動状況をアプリケーションレベルで監視し、ストリーミングメディアを配信できない状態になった場合、通知する。
Lost	ロストパケットを受信パケット数に対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
Recover	回復パケットを受信パケット数に対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
CurrentFrameRate	フレームレートをコンテンツのフレームレートに対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
FramesSkipped	スキップフレームをコンテンツのフレームレートに対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。

表 2 Real Media 監視項目一覧

監視項目	説明
Availability	稼動状況をアプリケーションレベルで監視し、ストリーミングメディアを配信できない状態になった場合、通知する。
AverageBandwidth	平均帯域幅をコンテンツ帯域幅に対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
CurrentBandwidth	瞬間帯域幅をコンテンツ帯域幅に対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
Late	遅延パケットを受信パケット数に対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
Lost	ロストパケットを受信パケット数に対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
CurrentFrameRate	フレームレートをコンテンツのフレームレートに対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
FramesDropped	破棄フレームをコンテンツのフレームレートに対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。
FramesLost	喪失フレームをコンテンツのフレームレートに対する割合(%)で監視し、指定したしきい値を超えた場合、通知する。

サーバの配信状態が正常に復旧し、視聴端末に

おける配信品質が改善した後も、監視端末上の擬似プレイヤーから取得されるフレームレートの復旧に時間がかかる問題に対しては、擬似プレイヤーの受信セッションを一度切断し、次の監視データを取得する際に再度サーバへの接続を試みて受信品質データを取得するアプローチをとることとした。これにより、接続開始時刻の平均フレームレートを監視データとして取得できるようになり、回復時の応答性が向上することになる。この改善手法には以下の二つの方法が考えられる。

1. 擬似クライアントで、一定時間毎に定期的に受信ストリーミングのセッションを切断し、再接続を実施する方法
2. 擬似クライアントで、フレームレート値が一定の閾値を下回った（品質劣化を検出した）時点で、受信ストリーミングのセッションを切断し、再接続を実施する方法

方法1では、サーバへの再接続要求を一定時間毎に制御することができるため、サーバが過負荷な状態が連続する様な苛酷な環境では有用であると考えられるが、再接続要求が出る間隔が疎になると復旧後の低いフレームレート値を記録する可能性が高くなる問題がある。逆に方法2では、サーバが過負荷な状態が続いた時に再接続要求が頻発することになるが、復旧後のフレームレート値は早期に回復する監視が可能となる。

方式2の改良を iPushSERVE Streaming Monitor の擬似クライアントを加えた後、図1の実験環境で行った実験結果を図3に示す。実験に用いた各種パラメータはプロトタイプ実験と同様である。ただし、負荷端末からの負荷は18:30から18:34までの4分間与えている。

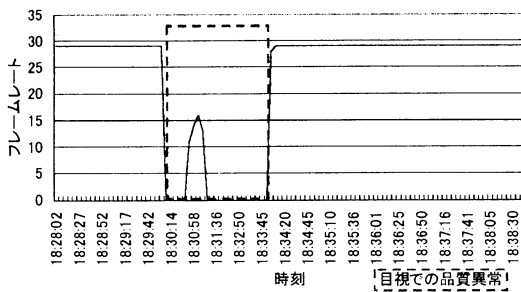


図3 改良方式による実験結果

図3の実験結果からも分かるように、負荷端末からの負荷を停止し、サーバの配信状態を正常に戻すと同時に目視での品質評価も向上し、フレームレートも即時に元の値に復旧することが確認できる。

この様に改良方式は、サーバの配信劣化の発生、復旧に高い追従性を示す監視が可能になったといえる。

## 5. 大規模配信環境での検証

提案方式の有用性を検証するため、2004年10月14日に実施されたLIVE! ECLIPSE 2004 partial[5]の部分日食インターネットライブ中継で実証実験を行った。この実証実験では図4に示す様に実験用配信サーバを用意し、CRNの一サイトとして全アクセスの1/50を受け持った。サーバの脇に監視端末と視聴端末を置くことでプロトタイプ実験と同様にサーバにおける受信品質の測定を行った。

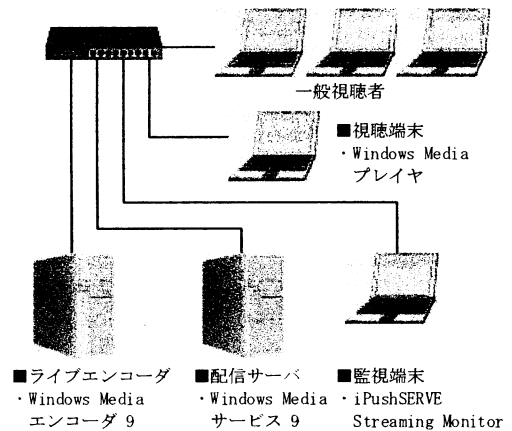


図4 実証実験の構成

実験結果は図5に示すとおりである。本実証実験結果でも視聴端末による目視による品質劣化の状況と監視端末によるフレームレートの値が減少する時刻はほぼ一致している結果が得られており、提案方式の有用性が検証され、目視によらない受信品質監視を実現できることが示された。

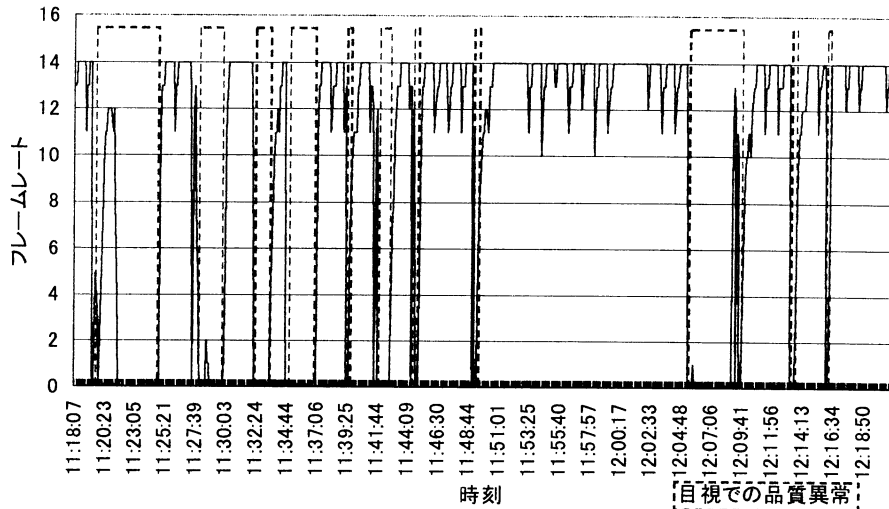


図5 実証実験結果

## 6. まとめと今後の課題

提案方式により、ストリーミングの受信品質の劣化検出が可能になったと考えられる。また監視ソフトウェアの拡張により品質劣化時にオペレータへの通知を行うシステムの構築も可能となる。これまで視聴による受信品質監視に多大な労力を払っていた運用管理者は、本提案方式を導入することで監視コストの抑止や、誤検出のリスクを回避することが可能となり、フレームレートの遷移を確認することで、過去の配信状況を把握することも可能になると考える。

今後、本提案方式をCDN/CRNシステムに適用し、サーバ間フィードに隣接して本方式を用いた品質監視を導入することで、各フィードの品質状態を把握できるようになると期待する。

**謝辞** 本提案内容の議論および測定に加わっていただいたCRN Forum受信品質部会のメンバ、特に株式会社アイアイジェイメディアコミュニケーションズ 高山将孝氏、岡淳一氏、株式会社IRIコミュニケーションズ BBX 事業部 芝野智誉氏、エヌ・ティ・ティ・コミュニケーションズ株式会社 佐野由紀子氏、若林進氏、エヌ・ティ・ティ・スマートコネクト株式会社 遠藤論氏、北海道総合通信網株式会社 池野桂司氏、松下電器産業株

式会社 東正美氏の各氏に感謝します。

## 参考文献

- [1] <http://www.crnf.net/>
- [2] <http://www.openview.hp.com/>
- [3] <http://www.nagios.org/>
- [4] 伊藤，今福，大浴，星野，”ストリーミング監視ソフトウェア iPushSERVE Streaming monitor”，PFU Tech. Rev. Vol. 14, No. 1, pp.1-10, May 2003.
- [5] <http://www.live-universe.org/le2004p/>

## 正 誤 表

pp.105 第一著者の所属

誤：

株式会社 PFU 〒105-0123 東京都港区山田 1-2-3

正：

株式会社 PFU 〒194-8510 東京都町田市鶴間 658-1