

IPFIXによるIPTVマルチキャスト・ トラフィック監視手法の提案

小林 淳史 (akoba@nttv6.net)
NTT情報流通プラットフォーム研究所

背景

- マルチキャスト運用の課題に関する報告は様々なところで指摘されている†.
- IPTVなどのNW品質に敏感なサービスの提供により、更に運用が困難になる.
- マルチキャスト・ルーティングとは独立な仕組みで、トラヒックの状態が把握できる仕組みが必要とされる.
- パケットロスなどを測定することで、映像品質を推定する技術も報告されている‡.

† “IP/MPLSマルチキャストの運用管理技術への期待”, MPLS JAPAN 2007

‡ “IPTVにおけるインサービス映像品質評価モデルの検討” CQ2007-94

課題に向けたアプローチ

課題：品質把握までを可能とするIPTVマルチキャストトラフィック監視システムの検討。

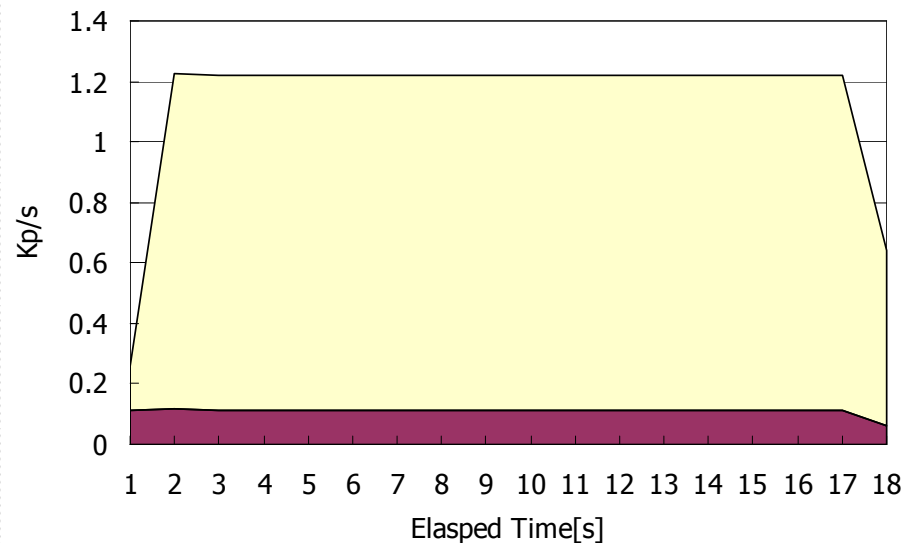
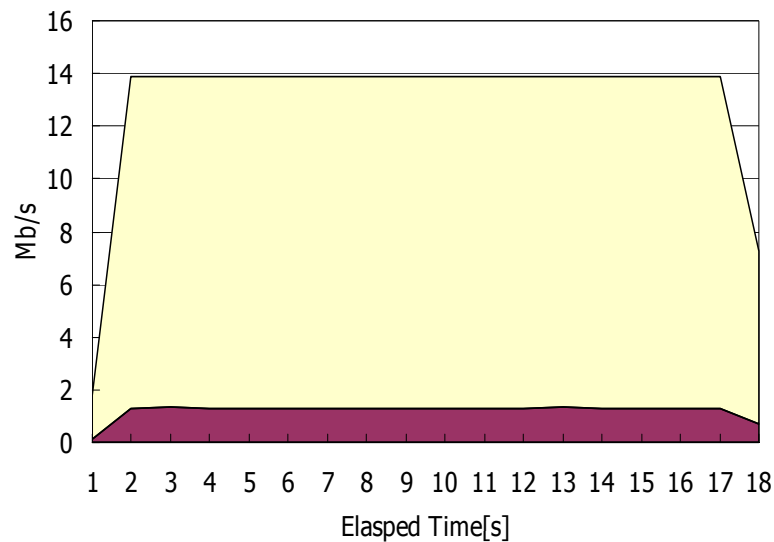
- IPFIX, PSAMPによるトラフィック監視技術を用いて、IPTVマルチキャスト配信サービスのトラフィック配信状況、品質状況を把握するシステムの検討。
- マルチキャストのルーティングプロトコルと独立に動作させることで、トラブルシュートなどに活用可能。
- 将来的にRouter, Switchより収集したトラフィック情報をもとに実現を可能とする。
 - 大規模網も含めて、有効性や実現可能性を評価する。
 - まだ実際に適用した報告例はない。

提案内容

- IPTVマルチキャストへのIPFIX,PSAMPの適用方式の提案
- IPFIXプロトコルを用いて配信する情報の検討.
- システム構成のプロトタイプ of 提案.
- プロトタイプをもとにしたシステム評価.

IPTVマルチキャスト・トラフィック

- 1チャンネル分のトラフィック流量は?
 - H.264にて12 Mb/s 程度.
 - 14Mb/s,1.2kp/s 程度.
- 50チャンネル分程度は、同時に品質を見えるようにしたい.
 - 全パケットをIPFIXで配信するのは、かなりのボリュームになる.



監視要件

□ 前提条件

- 大規模なNW上を流れるIPTVストリームのトラフィック動向を把握可能とする.
 - 主にトラブルシューティング、ユーザ申告による問題解析
 - ユーザ課金管理,Zapping性能を測定する目的ではない.

□ 要件1:マルチキャスト・ツリーの可視化.

- マルチキャストGrアドレス毎にNW上に流れる状態が可視化できること.

□ 要件2:影響度の大きい品質劣化の検出.

- RTPのSeq.番号のトレースによりロス検知可能であること.
- パケットロス1/1000 の発生が1分継続で検出可能なレベル.

□ 要件3:ツリー上でのトラフィック・品質状態の描画.

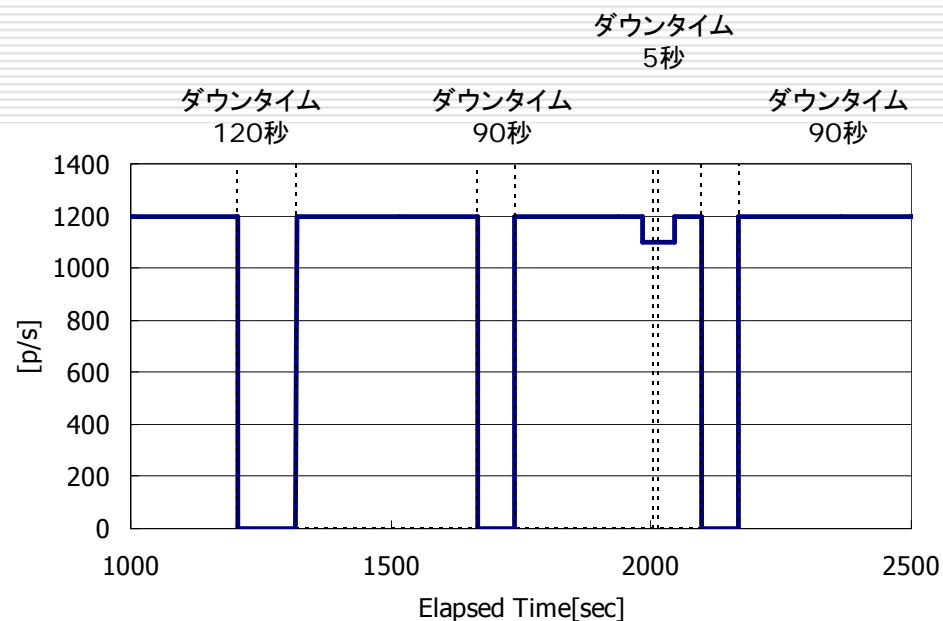
現状のNetFlow機能の課題

□ 要件2は、現状のNetFlowでは困難.

- 要件1:マルチキャスト・ツリーの可視化.
- 要件2:影響度の大きい品質劣化の検出.
- 要件3:ツリー上でのトラヒック・品質状態の描画.

□ フロー情報のタイムアウトの問題.

- ActiveTimeout(最大通信監視時間) 1分以上
- InactiveTimeout(無通信監視時間) 10秒以上

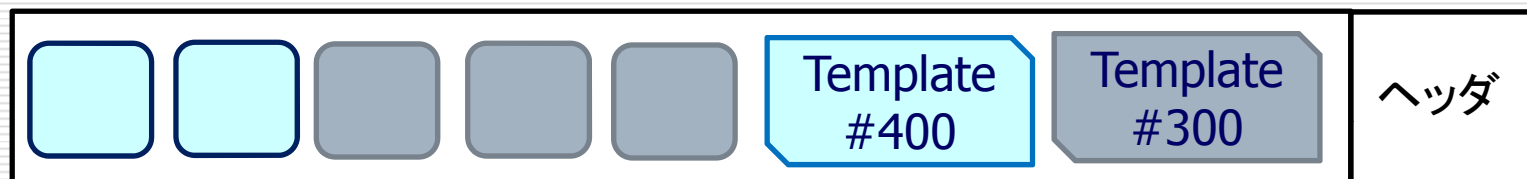


IPFIXの特徴

□ IPFIX (RFC5101)

- NetFlow ver.9 を母体に検討.
 - Templateによりユーザにより配信するフローの情報をcfg することが可能.
- SCTP,TCP,UDPによる配信が可能.
- Enterprise Id の設定により、ベンダ毎の情報要素の設定が可能.
- Variable-Lengthの情報要素の配信が可能.(パケット単位の情報配信を可能にする.)

IPFIXパケット

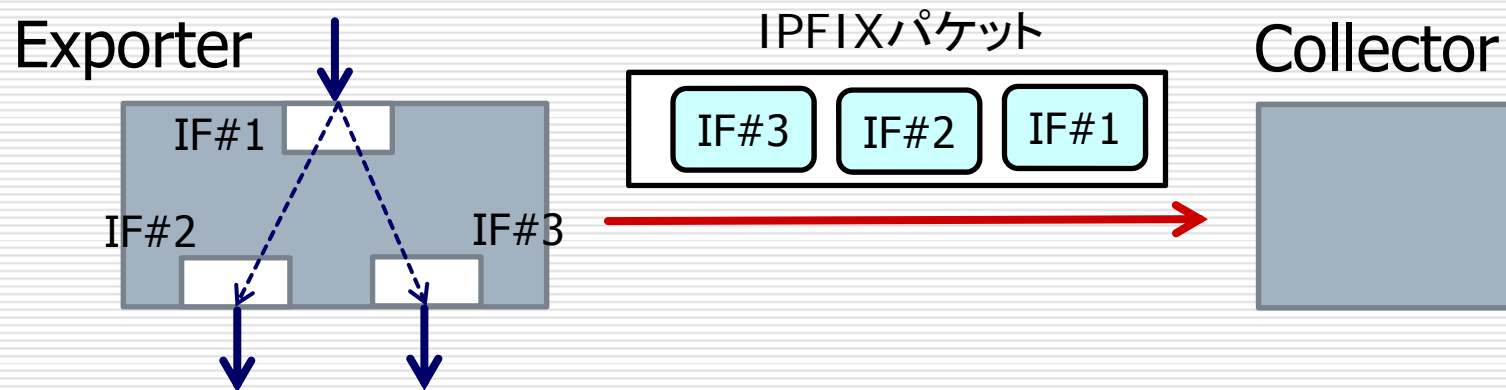


PSAMPの特徴

- PSAMP (draft-ietf-psamp-sample-tech-10.txt)
 - パケット単位のサンプリング,フィルタリング手法を定義.
 - サンプリング: random, deterministic, systematic time-base, systematic count-based
 - フィルタリング: 情報要素単位のフィルタ(ACL-like), Hash, uRPF etc

IPFIX マルチキャスト・フロー

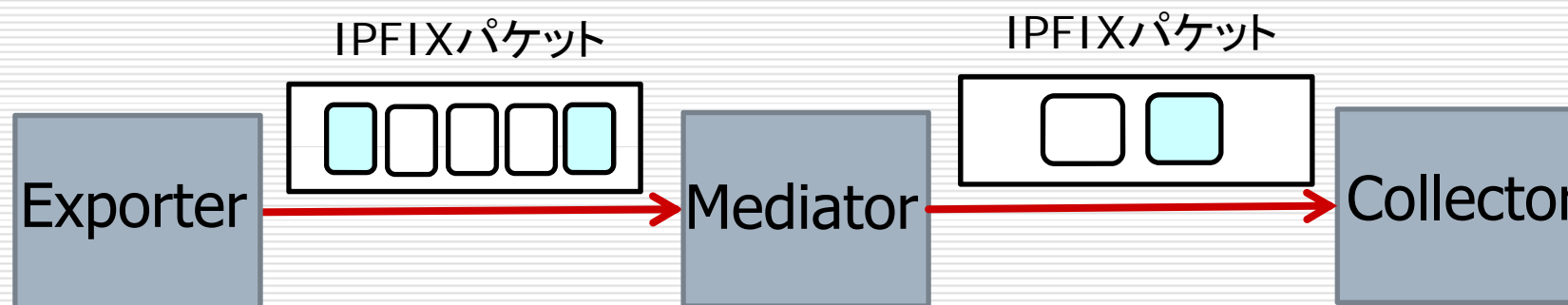
- マルチキャストの扱い(RFC3917)
- 複製されたパケットのフローも管理.
 - Ingress IFのフロー情報とEgressIFのフロー情報は、個別に管理してIPFIXを用いて配信.



数千ユーザを収容するアクセス側ルータでは、フローの配信が膨れ上がるという懸念.

IPFIX Mediation の機能検討

- フロー仲介装置 (Mediator) の検討
 - フロー情報の振り分け・フィルタ配信.
 - フロー情報内に情報要素追加
 - BGP経路情報, 顧客情報
 - フロー情報のanonymize
 - 集約(フロー粒度の変更)



要件1:マルチキャスト・ツリーの可視化

□ IPFIXオプション情報を活用.

- 通常は、フローのメタデータの配信に利用.
- IPFIXマルチキャスト・グループアドレス毎のEgressIF index の情報をIPFIXオプション情報として配信.
- Scope情報にルータを識別するExporter IPアドレスを追加.

| | | | |
|-----------------------|----|-----------------|--------------------------------|
| 0 | 15 | 16 | 31 |
| Set Id = 3 | | Length | |
| Template Id = 256 | | Field Count = 6 | |
| Scope Field Count = 5 | | 0 | ExporterIPv6Address Id = 130 |
| Field length = 16 | | 0 | DestinationIPv6Address Id = 27 |
| Field length = 16 | | 0 | SourceIPv6Address Id = 130 |
| Field length = 16 | | 0 | SourceIPv6PrefixLength Id = 27 |
| Field length = 16 | | 0 | ingressInterface Id = 27 |
| Field length = 4 | | 0 | egressInterface Id = 27 |
| Field length = 4 | | Padding | |

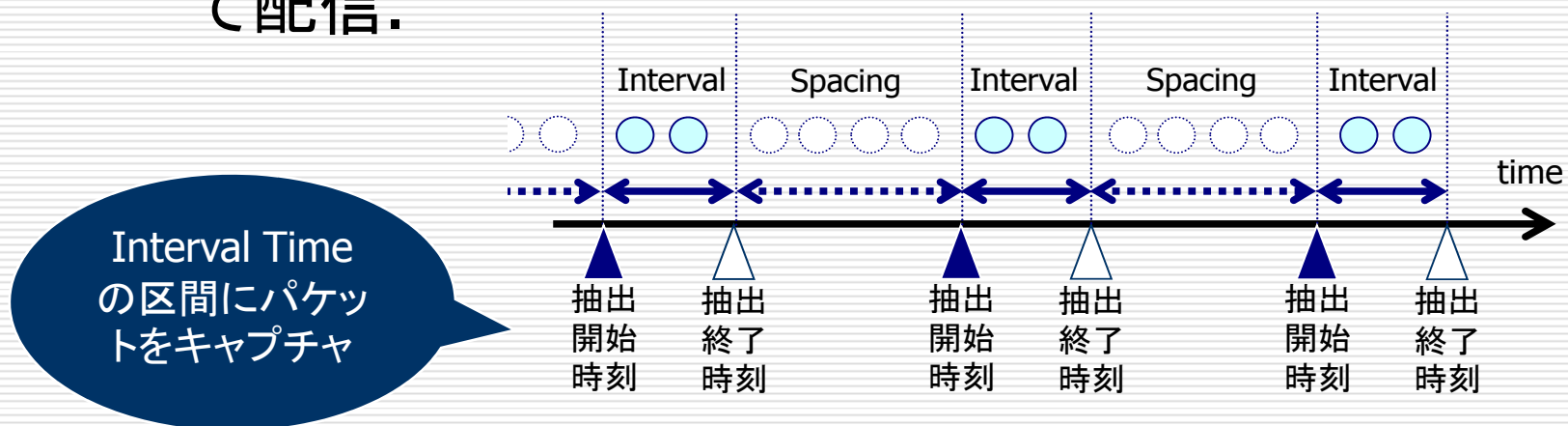
要件2:影響度の大きい品質劣化の検出

□ PSAMP フィルタリング

- PSAMPの技術を用いて、マルチキャストパケットに絞りこんでフィルタ.

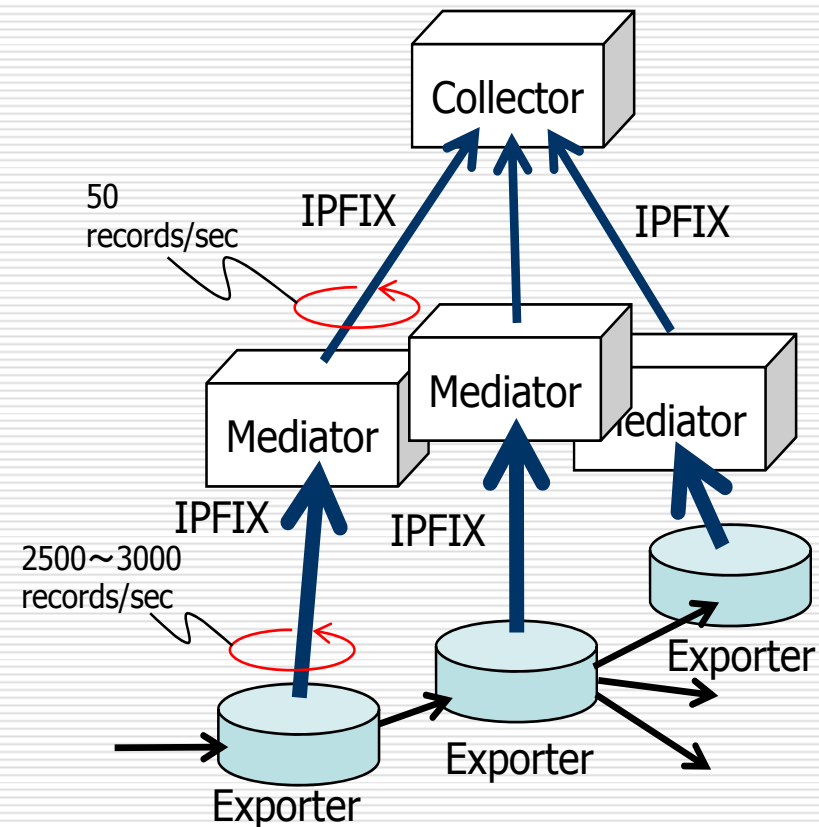
□ PSAMP systematic time-based sampling

- その後、パケットロスの把握のために、ある一定時間内のパケットを全てキャプチャしてIPFIXとして配信.

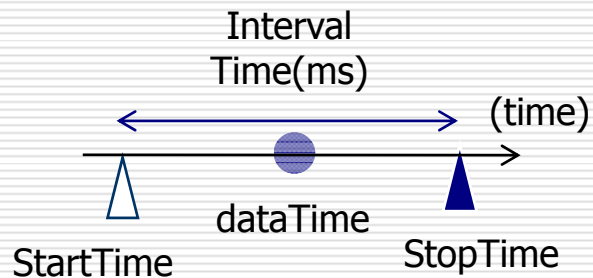
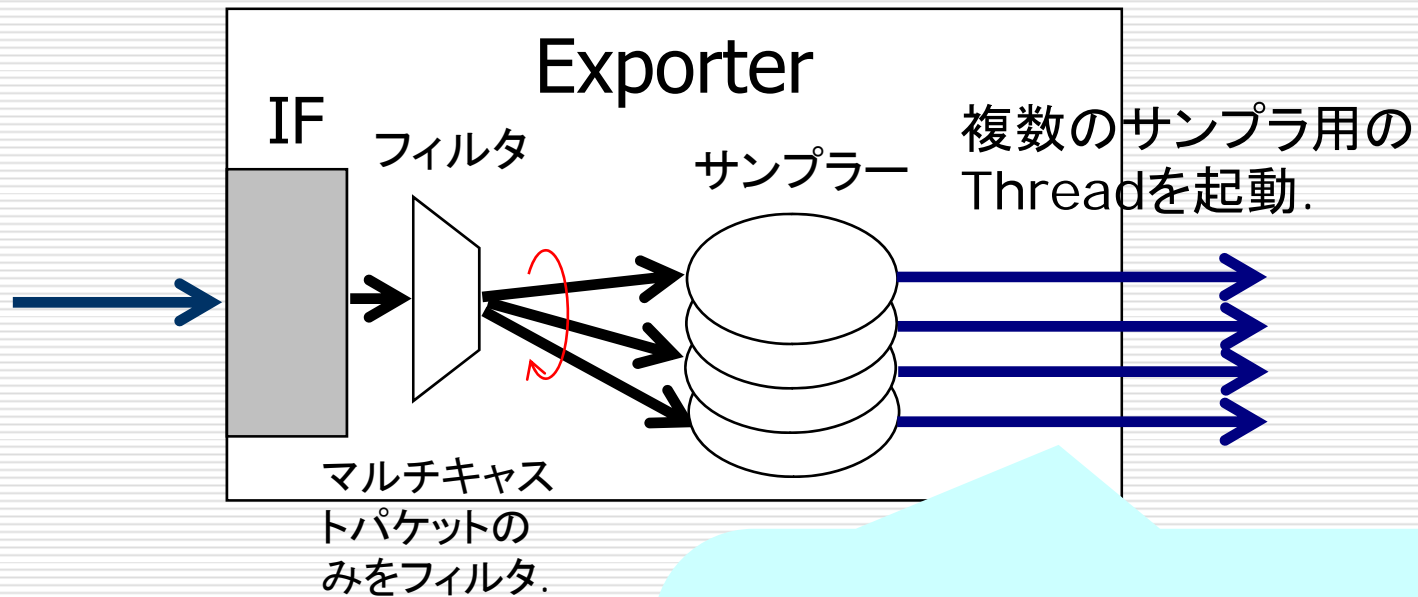


要件3:ツリー上でのトラフィック・品質状態の描画

- 大量のトラフィック情報を単一のコレクタに集約.
 - NW全体から情報を収集.
- IPFIX Mediatorの活用.
 - Exporter:パケット単位の情報配信.
 - Mediator:ロスをカウントして、フロー単位の情報として再配信.
 - Collector:トラフィック量、ロスをツリー上に描画.



Exporterの動作概要

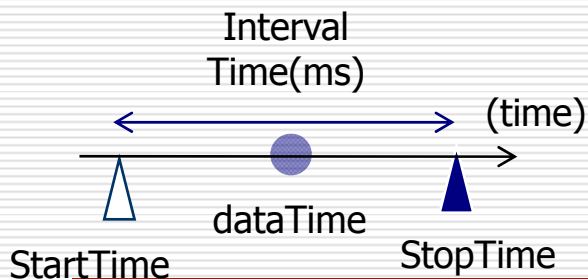
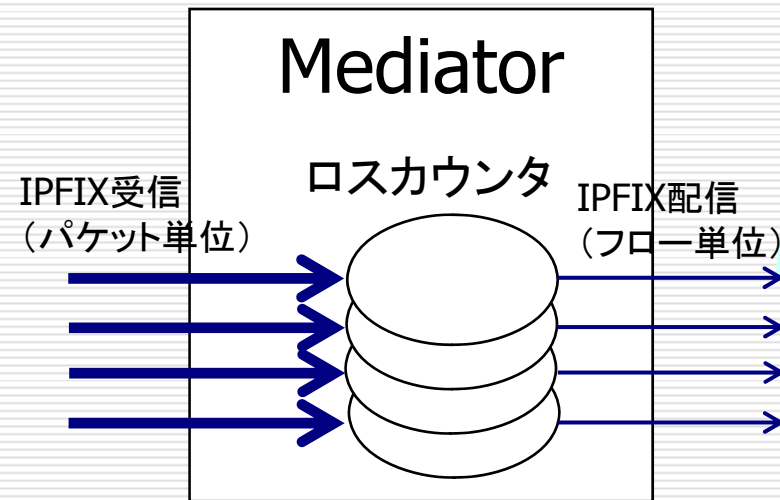


Interval Timeの開始/終了時間をフロー情報内に追加.

| 0 | | 15 | 16 | 31 |
|-------------------|--------------------------------|----------------------|----|----|
| Set Id = 2 | | Length | | |
| Template Id = 257 | | Field Count = 5 | | |
| 0 | ipHeaderPacketSection Id = 313 | Field Length = 65535 | | |
| 0 | ingressInterface Id = 27 | Field Length = 4 | | |
| 0 | dateTimeMiliSeconds Id = 323 | Field Length = 4 | | |
| 1 | startPointMillisecond Id = 1 | Field Length = 4 | | |
| Enterprise Number | | | | |
| 1 | EndPointMillisecond Id = 2 | Field Length = 4 | | |
| Enterprise Number | | | | |

Mediatorの動作概要

- StartTimeとStopTimeが同じデータを収集.
- PacketSectionからRTPヘッダのSeq.番号を取り出し、パケットロスをカウント.

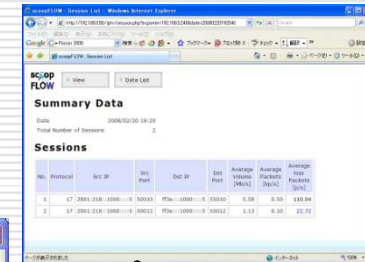


| 0 | | 15 | 16 | 31 |
|-------------------|----------------------------------|-------------------|----|----|
| Set Id = 2 | | Length | | |
| Template Id = 258 | | Field Count = 13 | | |
| 0 | Bytes Id = 1 | Field Length = 4 | | |
| 0 | Packets Id = 2 | Field Length = 4 | | |
| 0 | protocolIdentifier Id = 4 | Field Length = 4 | | |
| 0 | sourceTransportPort Id = 1 | Field Length = 2 | | |
| 0 | ingressInterface Id = 10 | Field Length = 4 | | |
| 0 | destinationTransportPort Id = 11 | Field Length = 2 | | |
| 0 | sourceIPv6Address Id = 27 | Field Length = 16 | | |
| 0 | destinationIPv6Address Id = 28 | Field Length = 16 | | |
| 0 | ipVersion Id = 60 | Field Length = 1 | | |
| 0 | exporterIPv4Address Id = 130 | Field Length = 4 | | |
| 0 | droppedPacketDeltaCount = 133 | Field Length = 4 | | |
| 0 | flowStartMilliSeconds = 152 | Field Length = 4 | | |
| 0 | flowEndMilliSeconds = 153 | Field Length = 4 | | |

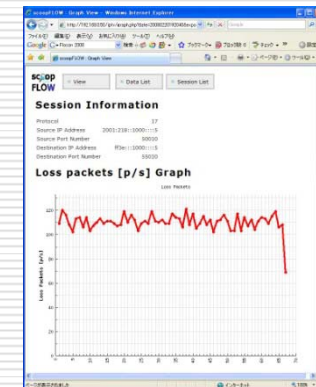
Collectorの描画機能

セッション状況

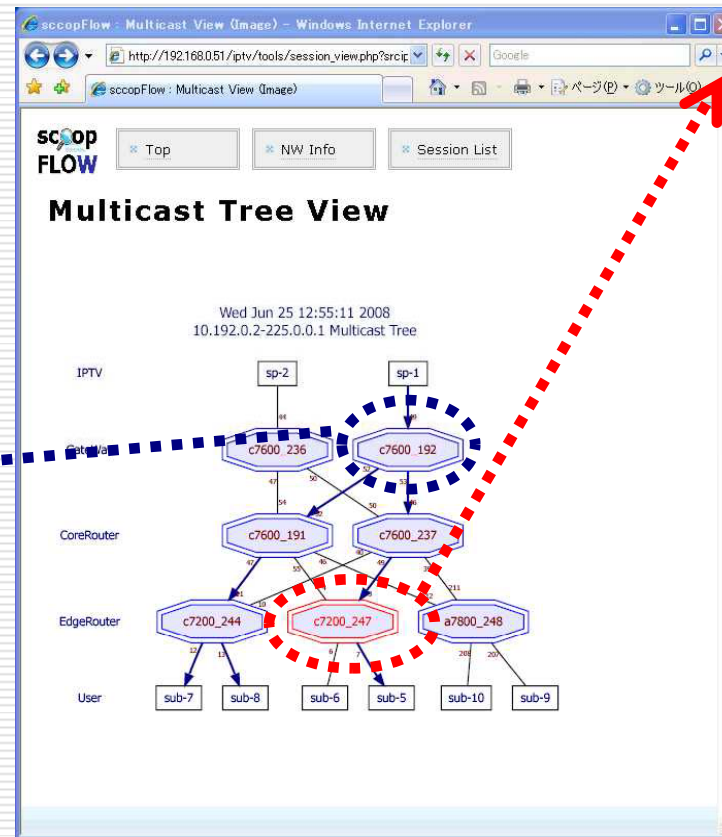
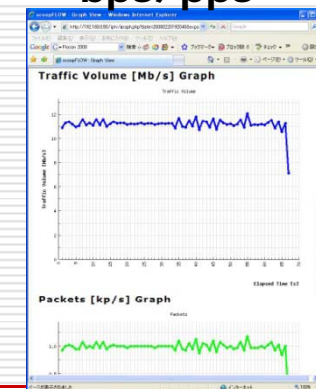
- Mediator側から配信される情報をもとにマルチキャストパス情報とパケットロスの情報を描画。



パケットロス

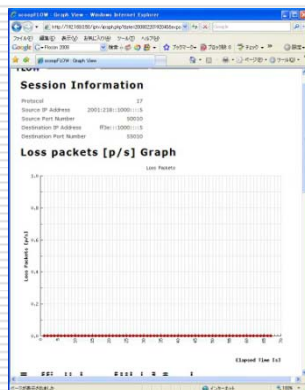


bps/pps

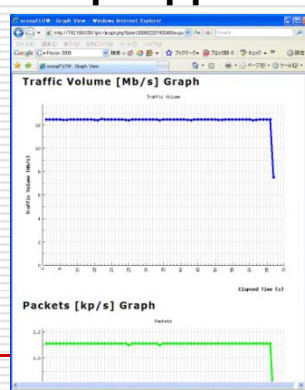


正常な場合

パケットロス

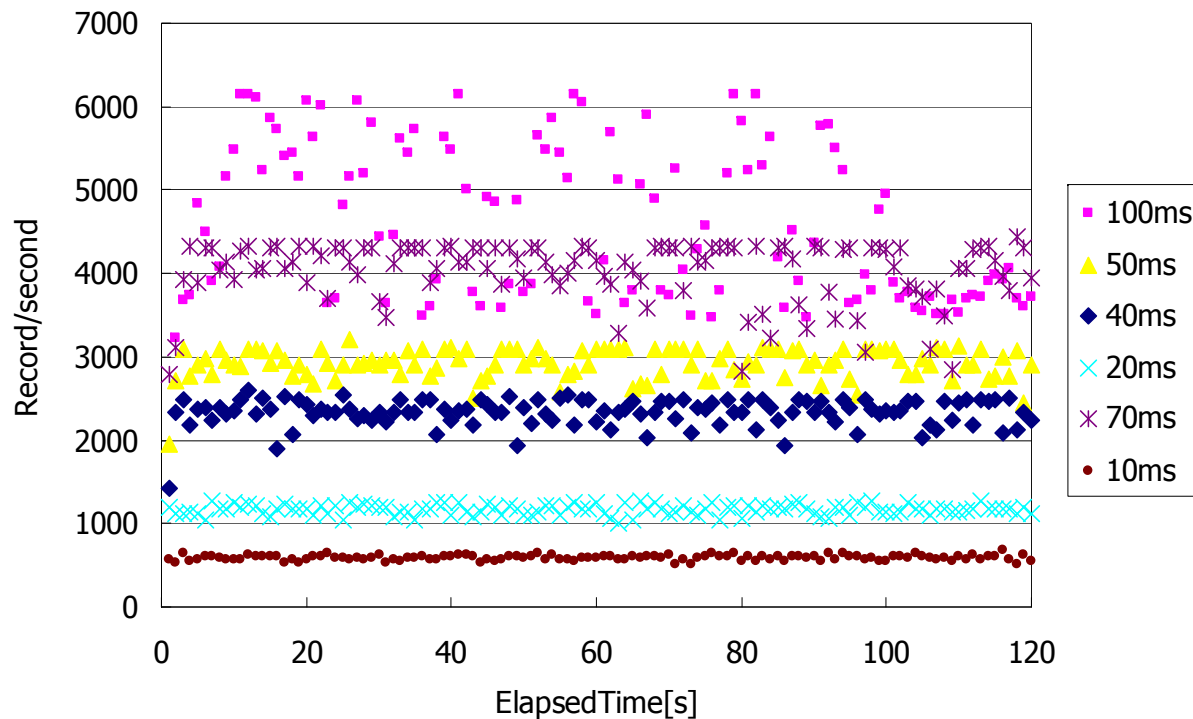


bps/pps



Exporterの評価

- IPTV50チャンネル分のトラフィックが収集可能か確認。
 - Interval Timeの区間を長くして、IPFIXの配信量を確認。
 - IntervalTimeとSpacingTimeは、合算すると1秒。

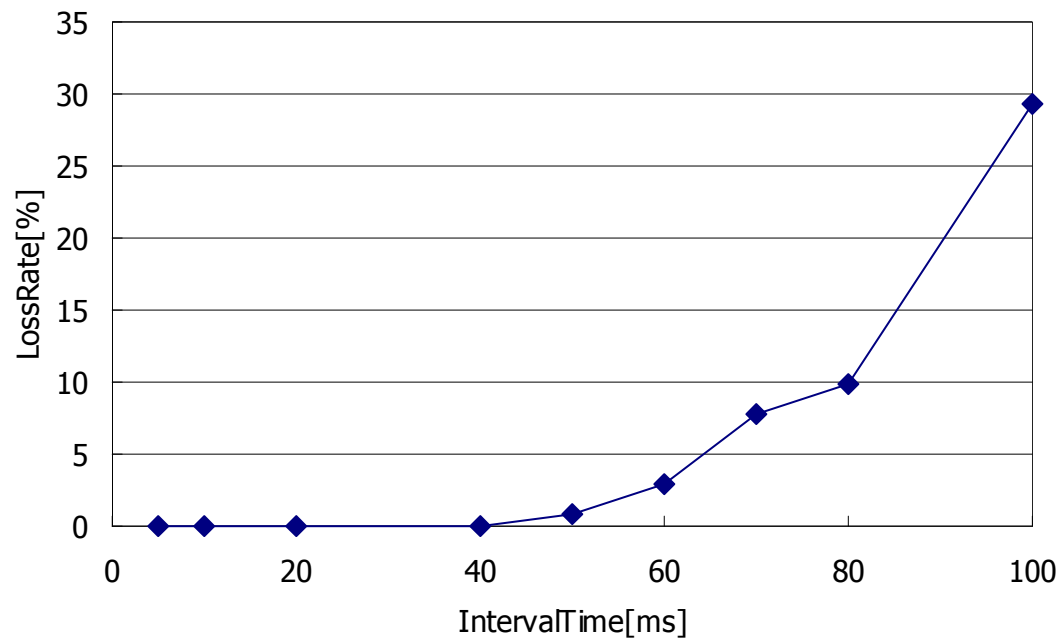


3000 records/sec
が上限.

- CPU: Quad Core2.66GHz
- メモリ: 4GB
- OS: FreeBSD 6.2

Exporterの評価

- Interval Time=50 msぐらいで、パケットのキャプチャの際に取りこぼし発生.
- IPTV 50チャンネル分では、40msのキャプチャ時間が限界.



Exporterの評価

- 1秒間に40msキャプチャするとパケットロスほどの程度検出可能か?
 - 監視時間単位:n分
 - インターバル時間比率: $\Delta/1000$
 - パケットロス率:p
 - IPTVパケットレート: r [p/s]

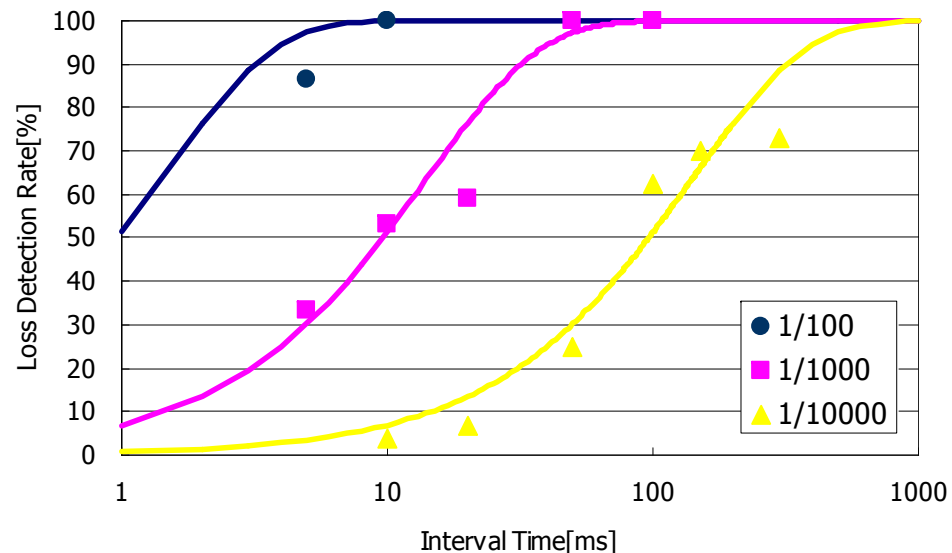
パケットロス検出確率

$$1 - (1 - p)^{r \cdot n \cdot \frac{\Delta}{1000}}$$

監視時間単位を1分として、実動作を確認。

ほぼ期待値どおり。

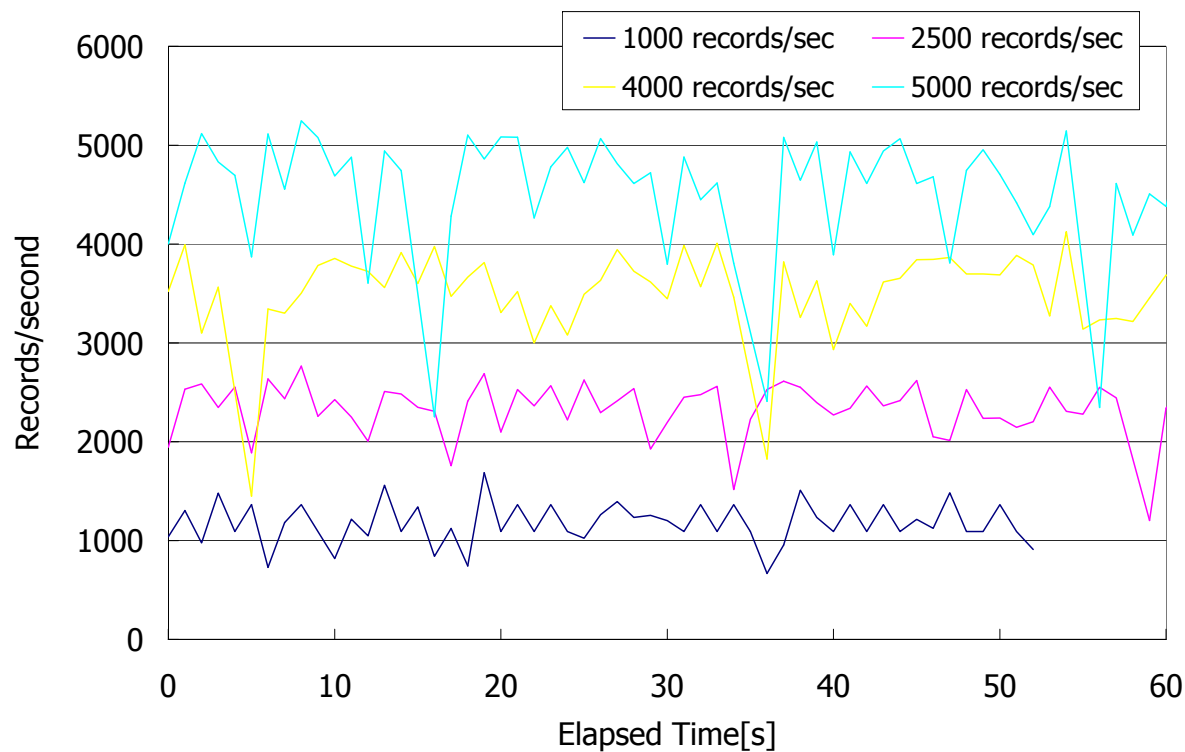
40 msだと80%ぐらいの確率で1/1000のロスを検出。



Mediatorの評価

□ 2500 record/sec が限界.

■ この構成では、Exporter:Mediator=1:1で配置.



- CPU: Core Duo 2.00 Hz
- メモリ: 4GB
- OS: FreeBSD 6.2

まとめ

- IPFIX, PSAMPによるIPTVトラフィック監視手法の有効性をプロトタイプをもとに検証.
- Exporterにおいては、実NW環境にも耐えうるProbeの開発も必要.
 - 同時チャンネル数の拡大、精度向上
- ルータ等への実装を促進するための働きかけの実施.