

# 情報の共有・検索を行うサービス指向ルータ・アーキテクチャ

鯉淵 道紘<sup>†</sup> 原島 真悟<sup>††</sup> 永富 泰次<sup>††</sup> 牧野 友昭<sup>††</sup> 石田 慎一<sup>††</sup>

井上 恒一<sup>††,†</sup> 川島 英之<sup>†††</sup> 西 宏章<sup>††,†</sup>

<sup>†</sup> 国立情報学研究所, 東京都

<sup>††</sup> 慶應義塾大学大学院理工学研究科, 横浜市

<sup>†††</sup> 筑波大学大学院システム情報工学研究科, 筑波

E-mail: [†sr@west.sd.keio.ac.jp](mailto:†sr@west.sd.keio.ac.jp)

あらまし 我々はパケット転送に加え, カプセル化された転送データからリアルタイムで情報を収集し, インターネット・アプリケーションの価値向上, 高度化を支援するサービス指向ルータを提案してきた. 本報告では, フォワーディングエンジンなどの既存のルータの機能に加えて, サービス指向ルータの実現に必要な不可欠な研究要素技術である (1) メモリ使用量を抑える TCP ストリームの再構築技術, (2) 対象とする正規表現を頻繁に更新可能な処理エンジン, (3) 抽出情報を高速にデータベースに格納する DB インサージョンエンジンの詳細を述べる.

キーワード サービス指向ルータ, TCP ストリーム再構築, 文字列検索, DB インサージョン, 新世代ネットワーク

## A Service-oriented Router Architecture that Shares and Search Information

Michihiro KOIBUCHI<sup>†</sup>, Shingo HARASHIMA<sup>††</sup>, Yasutsugu NAGATOMI<sup>††</sup>, Tomoaki MAKINO<sup>††</sup>, Shin-ichi ISHIDA<sup>††</sup>, Koichi INOUE<sup>††,†</sup>, Hideyuki KAWASHIMA<sup>†††</sup>, and Hiroaki NISHII<sup>††,†</sup>

<sup>†</sup> National Institute of Informatics, Tokyo

<sup>††</sup> Department of System Design, Keio University, Yokohama

<sup>†††</sup> Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba

E-mail: [†sr@west.sd.keio.ac.jp](mailto:†sr@west.sd.keio.ac.jp)

**Abstract** We have proposed a service-oriented router (SoR), that supports not only packet transfer but also the collection of rich real-time information that cannot be obtained at end hosts, in order to make highly-developed value creation of Internet applications. In this technical report, in addition to functions of existing routers, such as forwarding engine, we describe the following required functions of service-oriented routers in detail: (1) a TCP reconstruction technique that reduces the amount of required memory, (2) a regular expression engine that can immediately update target patterns, and (3) a high-speed DB insertion engine.

**Key words** Service-oriented Router(SoR), TCP stream reconstruction, string search, DB insertion, new generation network

### 1. はじめに

近年, インターネット上に存在するあらゆる情報は, マッシュアップなど複数発信源からの情報を組み合わせる手法により, 多角的な価値とコンテンツとしての意義を有するようになった. 例えば Google や Amazon が次々と生み出す新技術が Web アプリケーション・サービスの高度化に寄与し, 新たなビジネスを掘り起こし広げている. 現在, Web アプリケーション・サービ

スのさらなる高度化のための 1 つの方法として, インターネット・インフラストラクチャであるルータやゲートウェイが取得可能な情報を積極的に活用する, あるいはルータが担うスイッチングをサービスに追加する研究が進められている [1] [2] [3]. 例えば Cisco ISR (Integrated Services Router) 向けに提供されている AXP (Application eXtension Platform) はルータ上で Linux アプリケーションを実行するための API を提供している. また, Active Network では, ネットワークノードが

キャッシュを持ち、株式市況やオンラインオークションのサーバーの負荷を軽減するために、トラフィックを解析してコンテンツに応じてパケット処理を最適化することなどが検討されてきた [1]。また、リコンフィギュラブルなハードウェアを用いることでアプリケーション層に及ぶパケット解析を高速に行い、IP ベースではなくコンテンツベースのルーティングを行う研究も行われている [2]。これらの研究はルータが単なる通信基盤に留まらず、次世代のインターネットにおけるサービスの中核になりうることを示している。

我々は、ルータをパケットデータの管理基盤として考え、この管理基盤から有用な情報を正規表現により抽出し、その情報をルーティングや新しいサービスの提供に生かすサービス支援型ルータ (SoR) を提案してきた [3]。このルータは、既存のルータに高速な情報抽出を効率的に行うための正規表現プロセッサとオンメモリデータベースへの高速なデータインサージョンを行うハードウェアなどを加えた構成を持つ。

トラフィック情報は、「ある URL にアクセスしたユーザは、他のどんな URL にアクセスするのか」、「ユーザがある URL にどのくらいの時間滞在していたのか」、「その情報がいつ、どこからネットワーク上に現れたのか」といった、検索サービスや人気調査にとって重要な情報を含んでいる。これらの情報は従来のネットワークシステムでは利用が困難であったが、我々が提案を行っている SoR アーキテクチャではリアルタイムでサービスに必要な情報をネットワークトラフィックから抽出することができる [3]。

本研究報告では、我々が提案している SoR における情報抽出、提供に関する 3 つの要素研究技術について詳細を述べる。

- メモリ使用量を抑える TCP ストリームの再構築技術: 受信したパケットから TCP ストリームの再構築情報を抽出し、到着した TCP パケット毎に正規表現による文字列検索が可能な部分 TCP 再構築法を述べ、評価する。部分 TCP 再構築法は、部分的に TCP パケットを再構築するのみで文字列の抽出情報を判定し、必要なペイロードのみを抽出することでメモリ使用量を抑える。また、複数パケットに渡る文字列検索を可能とするために部分 TCP 再構築法では各パケットの処理内容をコンテキストとして格納する [4, 5]。

- 対象とする正規表現を高速に更新可能な処理エンジン: (部分的に) 再構築した TCP ストリームから該当する情報を探索するために、迅速なパターン更新を可能とするプロセッサタイプの正規表現プロセッサを述べ、評価する。本正規表現プロセッサは、ネットワークプロセッサにおける複数のプロセッシングユニットアレイ (PU) に付随するコプロセッサとして搭載される [6]。

- 抽出情報を高速にデータベースに格納する DB インサージョンエンジン: 正規表現処理エンジンにより抽出されたデータを、データベースへの書き込みを行うハードウェアを述べ、評価する。(一時的な保存を行うために) 高速なメモリ書き込み処理を実現する Stream Input Adopter と、永続化処理を実現する Archiving Engine などで構成される。ホスト PC からの

要求により、IMDB もしくは Disk-Based DB の内容を求められるデータ形式に変換後、ホスト PC のメモリへ DMA 転送する [7]。

本研究報告の構成は次のとおりである。2 章では、サービス指向ルータの狙いを述べる。3 章以下では、サービス指向ルータの要素技術を順に述べる。具体的には 3 章では、コンテキストスイッチを用いた TCP 再構築法を提案し、4 章では、正規表現処理エンジンを提案する。5 章では DB インサージョンエンジンを提案し、6 章では、関連研究を述べ、7 章では結論と課題を述べる。

## 2. サービス指向ルータの意義

### 2.1 社会的意義: 情報オープンイノベーションの実現

既存のインターネットでは、クライアントサーバモデルにより、エンドホストであるサーバから提供されるサーチエンジンなどのコンテンツやサービスを利用している。インターネットのクライアントサーバモデルによるソフトウェアのディストリビューションは中間コストを必要とせず、企業の設備投資をサーバとソフトウェア開発に集中することが可能となった。中間コストが無くなり、明解になった費用対効果の算出より、企業は利益の回収手法について様々な挑戦を行い、それがソフトウェアのオープンソース化など様々なビジネスモデルイノベーションを創出してきた。しかしながら、このモデルは、競争力がサーバ側の設備投資を可能とする資金力あるいは資金調達力に依存してしまい、プラットフォームと言われる根幹サービスの多くはベンチャーキャピタルの発達した米国が主導権を握り続けている。

さらに、今後、プラットフォーム・サービスが行動履歴と呼ばれる重要なデータを独占することが懸念される。インターネットによって安価かつ効率的に行えるようになったダイレクトマーケティングは今後、行動履歴によってターゲティングされ、より精度が増す方向にあり、これは情報爆発時代の一般消費者にとっての情報の選別ニーズに応えるコンテンツやサービスのパーソナライズとして利用価値の高いものになると考えられている。従って、プラットフォームを資金力の優れた企業に依存することを助長するクライアントサーバモデルに対抗し得るサービスモデルの開発が急務である。

サービス指向ルータは、情報のオープンイノベーションを可能とするネットワークを目指している。ここでの提供価値は、上記のプラットフォーム提供企業が蓄積している行動履歴を、プラットフォームを持たない多くの新興企業にも公平に提供可能とすることにある。さらに、ルータより提供されるユニークなデータは、プラットフォーム提供企業にとっても有用であり、公正な競争の促進を行うことができる。

### 2.2 技術的意義: データのユニーク性とリアルタイム性

技術的意義として、サービス指向ルータが扱うデータのユニーク性とリアルタイム性を挙げる。

まず、ルータが獲得可能なユニークなデータとしてはエンドホスト横断的な行動履歴がある。例えば、ブログやクチコミ

の普及により、消費者の購買活動は検索>購買に留まらず、検索>比較>購買>クチコミへと変化しており、今後も新しいメディアの登場により変化するものと考えられる。このような購買活動を一括してトラッキングする手法は現在存在せず、米 Amazon が行っているようにエンドホスト内あるいは提携した企業同士の取り組みに留まる。このような一連の行動履歴は、メーカーの商品開発ならびに販売戦略への活用ニーズが高いだけでなく、一般消費者の購買活動を支援することも可能であり、ルータがこれらの情報を収集、提供することの意義は大きい。

また、サービス指向ルータの方向性の一つとして、インターネットを介して普及するコンピュータウィルスやスパムメールへの対策もエンドホスト横断的あるいはルータ横断的に行うことを利用した抜本的解決方法の提案も検討している。

次に、リアルタイム性についてであるが、これは情報検索での活用可能性が高い。エンドホストに位置する既存の検索エンジンサービスがクローリングという手法によりリアルタイム性を追求する限り、少なからず Web サーバの負荷を高めてしまう。具体的な提案は見られないが、今後、検索エンジン側のクローリング手法の改善、Web サーバ側の計算能力が期待されたとしても、将来ますます情報発信が個人レベルまで普及すること、あるいは新興国のみならずアフリカ諸国を含む発展途上国まで情報通信技術が広がった場合の全人類情報の収集として最適なアーキテクチャとはいえない。

一方で、(サービス指向)ルータは情報収集を受動的に行うことができるため、リアルタイム性が高いといえる。パケットがルータを経由する際に情報収集する仕組みは、リアルタイム性を追求しても Web サーバの負荷を高めることがない。この点については、検索エンジンサービスと対抗する技術という位置付けではなく、リアルタイム情報を提供する側として既存検索エンジンサービスと共存することが可能である。

### 3. サービス指向ルータにおける情報抽出機構

本章では、フォーワーディングエンジンなどの既存のルータの機能に加えて、サービス指向ルータの実現に必要な不可欠な技術である情報抽出機構について述べる。

#### 3.1 サービス指向ルータ・アーキテクチャ

サービス指向ルータ・アーキテクチャを図 1 に示す。パケット転送に必要なルーティングブロック、Backplane スイッチブロックは従来のルータと同様である。サービス指向ルータの特徴はネットワークプロセッサ・ブロックが従来のルーティングの機能だけでなく、ハードウェアにおいて TCP ストリームの再構築を行い、レイヤ 7 情報を展開し、文字列検索、抽出を行うことである。

図 1 において、上段の Hardware で囲まれた部分がハードウェアで実装され、トラフィックから情報を抽出する部分である。一方下段の Software で囲まれた部分はソフトウェアとして動作し、ユーザからの情報取得条件の受付、ルーティングの変更などを行う。また、対象となる検索文字列はユーザが SQL を基にした処理系を用いて記述することでルータへ登録する [8]。

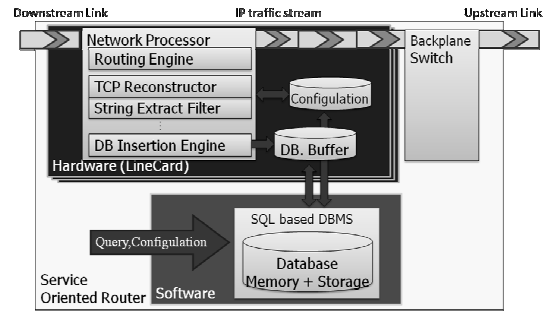


図 1 サービス指向ルータ・アーキテクチャ

#### 3.2 情報抽出を行うネットワークプロセッサ

ネットワークプロセッサ・ブロックにおける情報選択・抽出は図 2 に示した通り、以下の 5 つの処理エンジンにより行われる。

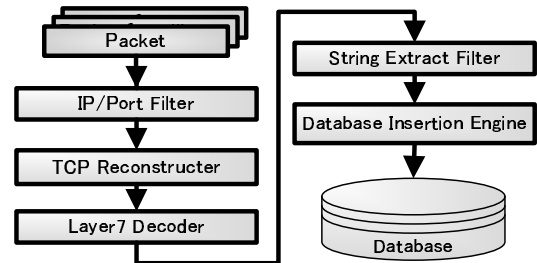


図 2 ネットワークプロセッサの情報選択・抽出処理の流れ

##### (1) IP/ポートフィルタ

ルータの各ポートを通過するパケットは、ネットワークプロセッサ内でまずヘッダ解析される。そして IP/ポートフィルタ内において IP/ポート情報から解析するパケットを限定する。具体的にはパケットヘッダ情報から抽出保存が必要ないと判断できるパケットを廃棄する。

##### (2) TCP 再構築

IP/ポートフィルタを通過したパケットは TCP ストリームへ再構築される。

##### (3) レイヤ 7 デコーダ

アプリケーションプロトコルのデコードを行う。TCP ストリームからレイヤ 7 デコーダによって、HTTP/1.1 や MIME エンコード等のアプリケーションプロトコルがデコードされる。

##### (4) 正規表現処理エンジン (文字抽出フィルタ)

レイヤ 7 ペイロードから文字抽出フィルタによって文字列探索が行われる。

##### (5) データベース挿入エンジン

抽出された文字列をメモリデータベースへと保存する。

データベースに格納されたデータは、プライバシー保護を行った後、ユーザに提供される。

### 4. TCP 再構築技術

本章ではサービス指向ルータにおける TCP ストリーム再構築技術について述べる。

#### 4.1 既存の TCP 再構築法の問題点

libnids などの代表的な既存の TCP 再構築法の実装は、ストリーム全体を保存するバッファが必要となる。そのため多数のストリームを扱うルータにおける TCP 再構築には向かない。

libnids の実装の場合、ストリームバッファと呼ばれるメモリ領域を用意し、ストリーム毎の TCP パケットのペイロードを書き込んでいく。そして、ストリームプロセッサ部は、ストリームのペイロードを直接読み込み、情報抽出を行う。

また、同様の処理を Snort Stream5 で行う場合、メモリ領域に、ストリームを構成する TCP パケットへのポインタを格納するストリームポインタリストと、TCP パケットを格納するバッファを用いる。そして、ストリームプロセッサ部は、そのポインタと TCP パケットの両方を読み込み、情報抽出を行う。

したがって、多数のコネクションが同時に通信を行うルータにおける情報取得手法として、これらを採用した場合、各ポートは多数のストリーム全体が揃うまでペイロードを格納する必要がある。これはメモリ使用量の点で以下の 2 点の無駄を生む。

##### (1) 非選択 TCP パケットの構築の無駄

例えば IDS に関するパケットを破棄する問合せ集合  $Q_D$  を  $n$  個の IP パケットから構成される TCP パケット  $t$  に適用することを考える。  $i(1 \leq i \leq n)$  番目の IP パケット  $ip$  が IDS の特徴を有する場合には、  $ip$  到着時に  $t$  は破棄可能であることが  $D$  の条件から判明する。しかし、既存の TCP 再構築の実装では  $n$  番目の IP パケットが到着するまで  $t$  を破棄できない。このとき、  $j(i + 1 \leq j \leq n)$  番目の IP パケットを保持するメモリ空間は無駄になる。

##### (2) TCP パケット中の非抽出ペイロードの構築の無駄

例えばペイロードの一部のみを抽出する問合せ集合  $Q_E$  を考える。String Extract Filter において  $Q_E$  で示されるペイロードの一部が抽出される。一方、残りの非抽出ペイロード部を保持するメモリ空間は無駄となる。

#### 4.2 部分 TCP 再構築法

本節では、メモリ使用量を削減するために、ストリームを構成するパケットが到着した時点で文字列検索の解析を開始し、解析の途中状態をコンテキストとして保存する部分 TCP 再構築法を述べる [4, 5]。

部分 TCP 再構築法では、TCP 再構築において、各 TCP パケットに TCP ストリームの再構築情報を付加する。TCP パケットに SYN フラグがあった場合、3 ウェイハンドシェイクが行われることを確認して新しいストリームの開始を検出する。そして、新しいストリームのためのエントリを Context Switch Information Table に作成する。Context Switch Information Table におけるコンテキストの詳細を表 1 に示す。

図 3 に部分 TCP 再構築法におけるパケット処理例を示す。図 3 においてストリーム A は処理が終了し、削除されている一方、ストリーム B, C については抽出情報に関するコンテキストが保存されている。

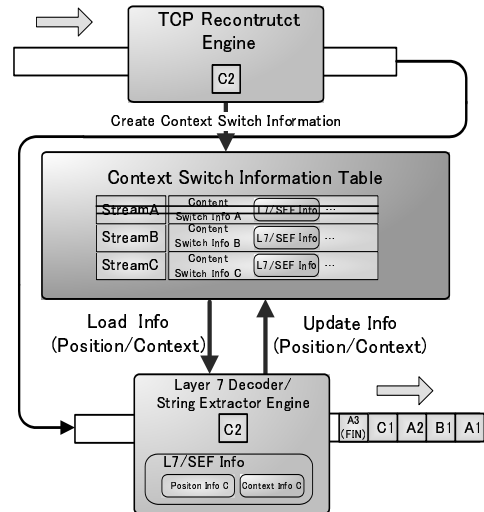


図 3 部分 TCP 再構築法におけるパケット処理

表 1 パケットのコンテキスト情報

変数	サイズ (Byte)	説明
ストリーム	160	TCP ストリーム状態保存用クラス。タイムスタンプや送信元/先の IP/ポートの組を保存する。
文字抽出情報	48	文字抽出フィルタの状態保存用クラス。以下の 2 つの状態保存領域へのポインタを持つ。サイズはポインタの管理に必要なメモリ量である。
active_rule_list	16 × 条件数	検索条件のポインタのリストを保存する。また各条件が未判定、判定中、判定終了のいずれの状態にあるかを保存する。
string_buf	文字列数	検索途中の文字列を保存する。検索条件の最も長い文字列長分必要となる。

なお、TCP 再構築エンジンを経たパケットはレイヤ 7 デコーダによって、HTTP/1.1 や MIME エンコード等のアプリケーションプロトコルがデコードされる。HTTP/1.1 のデコードを含め多くの場合、到着したパケットの途中状態のみで、処理中のパケットがデコードできる。よって後継のパケットに依存する処理がないため、ストリーム全体を保存する前にデコード処理が可能である。

#### 4.3 評価

部分 TCP 再構築法と Snort Stream 5 方式に基づく TCP ストリーム再構築 (以後、従来法と呼ぶ) の比較を行った。従来法においても TCP 再構築エンジン、レイヤ 7 デコーダ、文字抽出フィルタ以外の部分は部分 TCP 再構築法と同じである。

実験に用いた計算機は Intel(R) Xeon(TM) L5520 MP CPU 2.27GHz, 12GB RAM, CentOS release 5.4 である。評価項目として、各手法においてパケットの保存に必要なメモリ容量と計算時間とした。

評価におけるパラメータを表 2 に示す。特に記述がない場合は、左端の数字をデフォルト値として用いる。

検索条件は Snort の攻撃検知ルールセットから文字列検索を

表 2 パラメータ

パケットサイズ	1,500(デフォルト), 500 or 9,000Byte
ストリームサイズ	100(デフォルト) or 1,000 パケット
(送信) ホスト数	1,000(デフォルト), 1 or 10,000

行っているルール 68 個を利用した。なお、文字列検索を対象としたため、メモリ使用量はそのパターンの内容はほとんど影響されず、パターン数とパターンの文字列数に強く影響される。そのため本評価では Snort のみがサービス指向ルータの対象アプリケーションではないがメモリ量の初期評価のために、公開され、既知である Snort のパターンを使用した。

本評価においてメモリ使用量は、公平な比較を行うため、パケットを保存するために必要なメモリ使用量と、コンテキストスイッチのための中間情報を保存するメモリ使用量の和とした。

トライフィックパターンは各ホストが独立に TCP ストリームをポアソン分布に従った注入間隔で生成する合成トラフィックとした。なお、従来法と部分 TCP 再構築法ともに、文字列検索終了後にただちに、抽出情報をデータベースに保存し、メモリ上から削除される。そのため、合成トラフィックにおけるペイロードの中身は、メモリ使用量にほとんど影響しない。同様に宛先などの多くのヘッダ情報もメモリ使用量にほとんど影響しない。

#### 4.3.1 1つのストリームにおけるメモリ使用量

1つのストリームにおけるメモリ使用量の評価を図 4 に示す。図 4 において、1 台のクライアントは 1 台のサーバストリームを 1 つずつ順番に送信している。

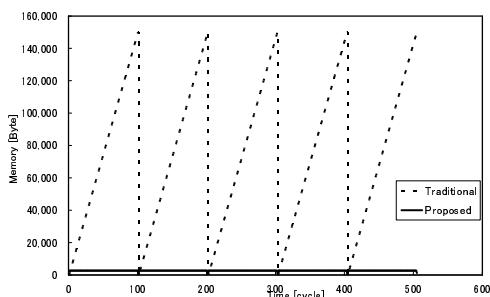


図 4 1つのストリームの場合のメモリ使用量

図 4 において、“Traditional Method” は従来法を用いた場合，“Partially TCP Reconstruction” は部分 TCP 再構築法のメモリ使用量を各々表す。

従来法の最大メモリ使用量が 150,102 バイトである一方、部分 TCP 再構築法は 2,672 バイトとメモリ使用量を 98%削減できていることが分かる。図 4 より、従来法ではストリームの最終パケットの到着までパケットを蓄積した後、パターンマッチ処理を行う。そして、処理が終わるとメモリ資源を開放していることが分かる。

一方、部分 TCP 再構築法はストリームの最終パケットを待つことなく、コンテキストスイッチ処理に必要な途中状態のみが常に保存される。そのため、パケットが到着する度に途中状態が更新されるため、ストリームの転送状態による影響が少な

く、必要となる最大メモリ使用量が少ないことが分かる。

なお、この他にもパケットサイズが大きい、あるいは、1 つのストリームを構成するパケット数が多い場合について評価を行ったが、従来法はグラフの形状は変わらず、メモリ使用量が大きくなった。一方、部分 TCP 再構築法ではさほど変化がなかった。

#### 4.3.2 複数ストリームにおけるメモリ使用量

本節では複数コネクションが同時に通信を行う場合の評価結果を図 5 に示す。各ホストは高々 1 つのストリームを同時に送信する。

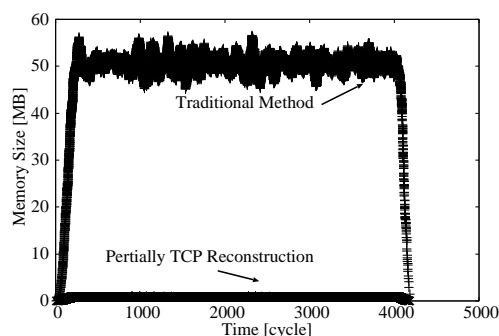


図 5 100 パケット・ストリーム、1500 Byte パケット、1000 ホストの場合のメモリ使用量

図 5 より、部分 TCP 再構築法は既存法に比べてメモリ使用量を大幅に削減できていることが分かる。

#### 4.3.3 文字列検索数の影響

検索条件として利用している Snort の攻撃検知ルールセットの文字列検索ルール数を変化させた場合の比較を図 6 に示す。

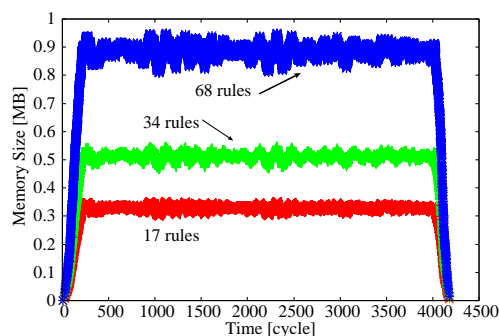


図 6 部分 TCP 再構築法におけるルール数ごとのメモリ使用量

図 5 に示した通り、既存法は極めて必要とするメモリ使用量が大いいため、部分 TCP 再構築法において抽出するルール数を 17, 34, 68 個と変化させて比較を行った。図 6 において、ルール数が増えるほど、メモリ使用量が大きくなるが、既存法のメモリ使用量と比べると、微々たるものであることが分かった。

## 5. 正規表現処理エンジン

サービス指向ルータにおいてレイヤ 7 デコーダを経たパケットは文字抽出フィルタによって正規表現に基づく文字列探索が行われる。文字抽出フィルタでは対象とする文字列の開始位置

を検出することで TCP パケット中の非抽出ペイロードを削除する．例えば、`<title>(.*?)</title>` という文字列抽出では、`<title>` と `</title>` のそれぞれの開始位置を検出することで、その前後のペイロードは破棄される．文字列検索では、検索対象となる文字列を含む TCP ストリームが複数のパケットに分割され、他の TCP ストリームと混ざった状態で受信されることを考慮しなくてはならない．そこで文字抽出フィルタは処理中のパケットが所属するストリームを同定し、ストリームにおける処理対象のパケットの位置を得る必要がある．そのパケットがストリームの先頭であった場合は各ルールに記述されている文字列の探索を開始し、マッチしたかどうかの結果とマッチした位置を結果を保存する．パケットの末尾までに検索が完了しなかった場合、検索の途中であるというフラグと検索の途中状態をそのストリームに保存する．処理対象のパケットがストリームの途中であった場合は、前述の途中状態を参照し、検索を再開する．TCP ストリームが終了する前であっても条件にマッチしないことが分かった場合、その後到着する同じ TCP ストリームに所属するパケットに対しては文字抽出フィルタは何もしない．

ここで、我々は、検索対象のストリームに対して、迅速なパターン更新を可能とするプロセッサタイプの正規表現プロセッサを提案、検討した [6]．

正規表現プロセッサは、ネットワークプロセッサにおける複数のプロセッシングユニットアレイ (PU) に付随するコプロセッサとして搭載される．パケットは途中で PU と、正規表現プロセッサの両方に送られる．

正規表現プロセッサコアは一般的な RISC プロセッサが備えるうち基本的な命令セットに加え、正規表現処理と抽出処理に特化した命令を拡張したプロセッサである．正規表現プロセッサは正規表現におけるマッチング処理を専用に行う複数のマルチファンクションコンパレータを集中管理し、パレルシフト内のある範囲の文字列を割り当て比較する．

パケットはストリームと呼ばれるまとまった単位で順次正規表現プロセッサのパレルシフトに情報が送られる．パレルシフトは、16 文字までの文字列を任意の範囲で取り出すことが可能である．また並列取り出しも可能である．

取り出した文字列はマルチファンクションコンパレータ内に複数あるビットマップコンパレータアレイとバッファに送られる．抽出された文字列は、ポストプロセッサに送られる．そして、ポストプロセッサは抽出すべき文字列の最終選択を行う．その結果はハードウェアベースのデータベースインサージョンエンジンがデータベースへ書き込む．

1 つのビットマップコンパレータアレイは、128 個のフリップフロップのアレイで構成され、各フリップフロップを 1 つの文字に対応させ、その値 (0 or 1) によりその文字が抽出対象文字かどうかを判定する．これにより、パイプ (||) を用いた 7-bit ASCII 文字の比較を 1 ステップで行うなどの高速化が可能である．

我々は、Verilog HDL を用いてビットマップコンパレータを

設計した．そして、これらを 45nm の CMOS プロセスを用いて Design Compiler を用いて論理合成した．その結果を表 3 に示す．MCPS は Mega Character per Second の略である．これらの結果より、使い方を限定することにより 1Gbps 程度のスループットがあれば許容されるエッジルータなどで用途を限定することで本正規表現プロセッサアーキテクチャを導入することは可能であるといえる．

表 3 論理規模及び動作遅延

回路面積 ( $\mu\text{m}^2$ )	動作遅延 (ns)	スループット
4060.8	1.34	7.46MCPS, 5.97Gbps

## 6. データベースインサージョンエンジン

### 6.1 概要

一般にデータベースシステムでは信頼性を獲得するためにハードディスクへの書き込みが原則とされるが、全てのデータをメモリ上に格納し、高速なデータ処理が可能である In-Memory (IM)DB が新しいアプリケーションを開拓している．しかし、IMDB はディスク型データベースと比較し容量が小さい．ストレージ容量の制限を考えれば、すべてのトラフィックを書き込むのではなく、ユーザにより選択的に絞り込んだ情報を書き込むことが望ましい．また、高速な書き込み処理の実現と記憶容量の確保・障害によるデータ消失の回避を両立するため、サービス指向型ルータにおけるデータベースは IMDB とハードディスクストレージ DB の階層的なアーキテクチャが必要である [7]．

サービス指向ルータにおいて我々が提案する図 7 に示した DB インサージョンエンジンは、データベースへの書き込みを行うハードウェアであり、高速なメモリ書き込み処理を実現する Stream Input Adopter と、永続化処理を実現する Archiving Engine などで構成される．ホスト PC からの要求により、IMDB もしくは Disk-Based DB の内容を求められるデータ形式に変換後、ホスト PC のメモリへ DMA 転送する．

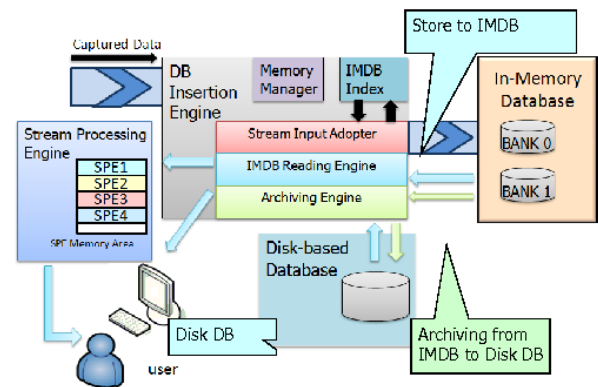


図 7 データベースインサージョンエンジン

### 6.2 IMDB 領域への書き込み手法

トラフィック情報を IMDB へ書き込む際、セグメントという単位で管理され、そのインデックス情報を作成・管理しなけれ

ばならない．インデックスの作成方法を以下の2種類検討した．

a) インデックス構造

もっとも単純な方法は，図8に示した逐次アクセス方式である．インデックスは，データの到着時間のみを利用して作成する．構成が単純でありメモリのRAS-to-CAS遅延を隠蔽できるため，書き込み処理が高速に実行できるが，一つのセグメントに複数クエリのデータが含まれるため，検索コストが大きくなる．

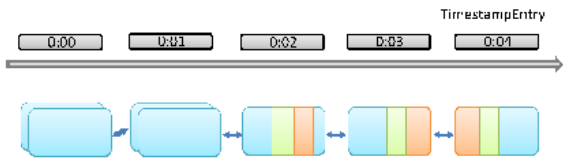


図8 逐次アクセス

第2の方法は，図9で示したLeaf別逐次アクセス方式である．データをLeaf-IDと時間の両方で管理する．高速な検索が可能であるが，インデックスメモリで管理できるエントリの構成が固定である場合各セグメントは多くて1つのトラフィックしか含めることができない．セグメントのサイズを小さくできない場合はメモリ利用効率が低下する．

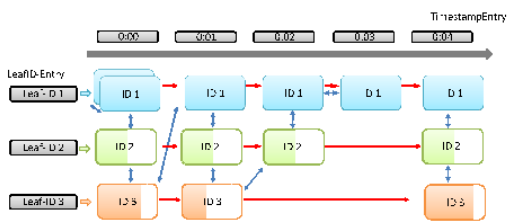


図9 Leaf-ID別逐次アクセス

6.3 評価

前節で述べた2種類の基本的なインデックス構造を評価した．インデックス作成部分をVerilogにより設計し，論理規模及び動作遅延を評価した．評価には5.1節で述べた3方式について専用命令を含む独自マイクロコードで記述し動作遅延を求めた．IMDBメモリへの書き込み処理は専用命令により並列処理されるため，メモリスループットがネットワークスループットよりも十分高いと仮定した．結果，ボトルネックとなるのはインデックス作成部位となる．また設計環境は正規表現プロセッサと同様である．

表4に論理規模及び動作遅延の結果を示す．

表4 論理規模及び動作遅延

回路面積 ( $\mu\text{m}^2$ )	回路面積 (NAND 換算)	動作遅延 (ns)
79872	100091	2.80

表5において，取得を想定している内容を含むパケットでは最小に近いサイズと考えられる50Bのパケットが連続して届

表5 ネットワークスループットの比較

	スループット (Gbps)			
	オンチップメモリ		オンチップメモリ	
	case1	case2	case1	case2
逐次アクセス	267	10.2	53.3	0.95
Leaf-ID別逐次アクセス	104	3.97	25.2	0.83

いた場合の性能をcase1に，WIDE MAWIプロジェクトで取得された2009年7月12日14:00-14:15のトラフィックにおけるHTMLパケットの平均サイズを利用した場合の性能をcase2に示す．これらの結果から，1Gbps以上の性能を獲得するためには，オンチップメモリよりインデックスメモリを構成することが必要である．

7. 関連研究

ハイパフォーマンスルータに関する基礎・応用研究が産学共に盛んに行われており，強力なバックボーンルータベンダ，それらを支えるプロセッサ・デバイスベンダ，独創的な周辺チップを提供するベンチャー企業が存在する．学術分野ではスタンフォード大学のマッケオンらによるルータに関する研究が著名であるが，これらの研究は従来の枠組みに則ったアーキテクチャに基づいている．コンテンツを扱うルータの基礎となるXMLルータに関しては，ワシントン大学セントルイス校のロックウッドらの研究が著名であるが，近年マッケオンとロックウッドの共同研究がスタートしたのは注目すべき点である．これは，宛先IPアドレスよりむしろパケットペイロード中のコンテンツによってルーティングを行う点が特徴である．

2009年度，日本電気(株)よりSMTPをリコンフィギュラブルハードウェアで解析する迷惑メールフィルタエンジンが発表されたが，サービス指向ルータを実現するにあたって前提となる，ネットワークの基調をなす技術の1つと考えることができる．類似研究としてトラフィックデータを用いた侵入検知システムの一つであるSNORTがある．SNORTはあらかじめ登録されている正規表現を含むルールセットとトラフィックデータを比較し，合致すれば警告を出すシステムである．SNORTはソフトウェアで処理されるために，数Gbitのスループットを満足させることは困難であった．現在は専用ハードウェアの実装により10Gbps程度のスループットを達成した研究も存在する．

システム管理，課金などのために，ルーター・スイッチのフロー情報 (Netflow, sflow, peakflow) の高速収集とフロー情報を用いた帯域監視が行われているが，そのパケットのペイロードをリアルタイムで解析，情報抽出することは，Deep packet inspectionの分野において盛んになりつつある [9]．その他，サーバなどのホストにおいてトラフィック解析を行うソフトウェア [10] が存在するが，トラフィック量の測定が主な目的であり，パケットストリーム中のコンテンツの解析を行わない．

一般的に，リアルタイムで，あるいは膨大な貯蔵されたトラフィックから情報抽出を行うためには，正規表現処理が必要となる．NFA/DFAによるFPGAを用いたハードウェア正規表

現エンジンについての研究は近年の Snort [11] の普及により再び活発化している。しかし、対象とする正規表現が頻繁に更新される場合は FPGA の再構成が必要となり、これらのシステムを本研究と同様の目的で使うことは困難である。

近年のルータは高い計算能力や交換能力の獲得により、単なる IP パケットの中継機器には留まらず、ユーザや ISP 向けのサービス提供が可能となってきた。Cisco ISR (Integrated Services Router) に拡張して搭載できる AXP (Application eXtension Platform) では、ルータ上において Linux アプリケーションを実行するための API が提供されている。しかし、トラフィックを解析するための高度な処理エンジンは持ち合わせていないためサービス支援型ルータとは目的が異なる。また、Active Network では、ネットワークノードがキャッシュを持ち、株式市況やオンラインオークションのサーバーの負荷を軽減するために、トラフィックを解析してコンテンツに応じてパケット処理を最適化することが検討されてきた [1]。しかし、当時のインターネット・アプリケーションは限定的であり、我々のサービス指向型ルータのようにパケット全体にわたる処理をほどこしたアプリケーション高度化についてはあまり議論されていない。

## 8. 結 論

我々はインターネット・アプリケーションの更なる価値創成、高度化を支援するために、カプセル化された転送データから自動的にリアルタイムで情報を収集し、活用するサービス指向ルータを提案してきた。

本研究報告では、フォワーディングエンジンなどの既存のルータの機能に加えて、サービス指向ルータの実現に必要な不可欠な、以下の 3 つの要素技術について詳細を述べた。

- メモリ使用量を抑える TCP ストリームの再構築技術: 受信したパケットから TCP ストリームの再構築情報を抽出し、到着した TCP パケット毎に正規表現による文字列検索が可能な部分 TCP 再構築法を述べ、評価した。部分 TCP 再構築法は、部分的に TCP パケットを再構築するのみで文字列の抽出情報を判定し、必要なペイロードのみを抽出することでメモリ使用量を抑える。また、複数パケットに渡る文字列検索を可能とするために部分 TCP 再構築法では各パケットの処理内容をコンテキストとして格納する。

- 対象とする正規表現を高速に更新可能な処理エンジン: (部分的に) 再構築した TCP ストリームから該当する情報を探索するために、迅速なパターン更新を可能とするプロセッサタイプの正規表現プロセッサを述べ、評価した。本正規表現プロセッサは、ネットワークプロセッサにおける複数のプロセッシングユニットアレイ (PU) に付随するコプロセッサとして搭載される。

- 抽出情報を高速にデータベースに格納する DB インサージョンエンジン: 正規表現処理エンジンにより抽出されたデータを、データベースへの書き込みを行うハードウェアを述べ、評価した。(一時的な保存を行うために) 高速なメモリ書き込み

処理を実現する Stream Input Adopter と、永続化処理を実現する Archiving Engine などで構成される。ホスト PC からの要求により、IMDB もしくは Disk-Based DB の内容を求められるデータ形式に変換後、ホスト PC のメモリへ DMA 転送する。

今後、サービス指向ルータの FPGA プロトタイプを日立情報通信エンジニアリング (株) LogicBench において実装し、性能を測定する予定である。

## 謝 辞

本研究の一部は情報通信研究機構「新世代ネットワーク技術戦略の実現に向けた萌芽的研究」、科研費 (21013047) を受けたものである。

## 文 献

- [1] David J. Wetherall and Ulana Legedza and John Guttag: "Introducing New Internet Services: Why and How", IEEE Network Magazine (1998).
- [2] J. Moscola, Y. H. Cho and J. W. Lockwood: "A reconfigurable architecture for multi-gigabit speed content-based routing", HOTI '06: Proceedings of the 14th IEEE Symposium on High-Performance Interconnects, Washington, DC, USA, IEEE Computer Society, pp. 61-66 (2006).
- [3] K. Inoue, D. Akashi, M. Koibuchi, H. Kawashima and H. Nishi: "Semantic router using data stream to enrich services", Proc. of the 3rd International Conference on Future Internet Technologies (CFI08), pp. 20-23 (2008).
- [4] 石田, 原島, 川島, 鯉淵, 西: "パケットデータ管理基盤における抽出処理の効率化技法", 電子情報通信学会技術研究報告 CPSY2009-86 (2010).
- [5] 石田, 原島, 鯉淵, 川島, 西: "コンテキストスイッチを利用したルータにおける tcp ストリーム再構築のメモリ削減手法", 情報処理学会研究報告 2010-ARC-190, No.5 (2010).
- [6] 永富, 石田, 三野, 川島, 鯉淵, 西: "リッチなユーザーサービスを提供するセマンティックルータにおける正規表現プロセッサの提案", 電子情報通信学会, ネットワークシステム研究会 (NS) (2008).
- [7] 牧野, 辻, 川島, 鯉淵, 西: "サービス指向型ルータにおける高速な書き込み機構の提案", 電子情報通信学会技術研究報告, 電子情報通信学会, コンピュータシステム研究会 (CPSY2009-23), Vol. 109, No.168, pp. 79-84 (2009).
- [8] 川島, 鯉淵, 西: "パケットストリーム処理における正規表現選択演算を含む問合せ最適化", 電子情報通信学会技術研究報告 CPSY2009-87 (2010).
- [9] OpenDPI: [www.opendpi.org](http://www.opendpi.org).
- [10] IPTraf: [iptraf.seul.org](http://iptraf.seul.org).
- [11] SNORT: [www.snort.org](http://www.snort.org).