

# REST を用いた軽量 Web サービスアーキテクチャの提案と評価

池崎 崇十 永橋 陽一郎† 森 晃‡ 井垣 宏† 青山 幹雄†

南山大学 数理情報学部 情報通信学科† 南山大学 大学院 数理情報研究科‡

## 1. はじめに

Web サービスのメッセージ交換は、主に SOAP が利用されているが、SOAP 構文の生成や解析に処理時間を要する性能低下の問題がある。

本稿では、この問題に対し、REST (REpresentational State Transfer)を利用したアーキテクチャを提案し、プロトタイプを用いて、応答時間を比較し、REST の有効性を検証する。

## 2. メッセージ交換の問題点

### 2.1. SOAP メッセージによる性能の問題

SOAP のメッセージ交換は、SOAP エンベロープに XML 構文を用いる。一般的に SOAP メッセージの処理は、図 1 に示すように、生成(シリアライズ)や解析(デシリアライズ)に多くの処理時間を要する。

そのため、SOAP メッセージの処理は、XML 処理に伴う性能低下を招いている。

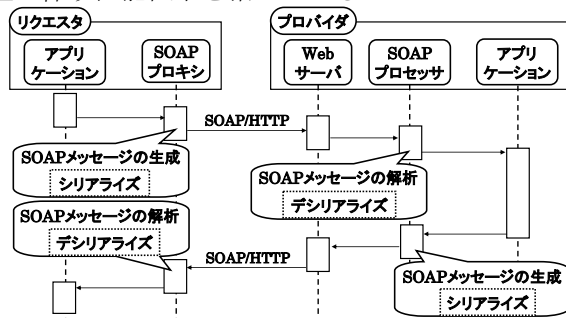


図 1: SOAP メッセージアーキテクチャ

### 2.2. 性能低下を招くメッセージ交換

SOAP メッセージ交換による性能低下が顕著となる場合を、以下に示す。

#### 1) 少量の情報を含むメッセージ交換

SOAP のボディに少量の情報しか含まないメッセージ交換は、SOAP 構文に関するデータが余分に付加される。そのため、データ量が増加し、通信の負荷が余分にかかり、性能低下を招く。

#### 2) 類似したメッセージを複数回交換

繰り返し類似したメッセージ交換を行う場合、SOAP ヘッダに冗長な情報が含まれる。これにより、メッセージの生成や解析に余分な処理時間を要するため、メッセージ交換の性能が低下する。

近年、少量の類似した情報を複数回交換する Web サービスが増加傾向にあるため、上記の状況は頻繁に発生すると考えられる。

## 3. REST の Web サービスアーキテクチャ

本論文では、SOAP メッセージ交換の性能問題に対して、REST の概念と制約に基づくメッセージ技術と REST の制約を適用したアーキテクチャを提案する [2]。

特に、図 2 に示すように、REST の概念と制約に基づくメッセージ技術の適用を中心に、性能の問題に対する改善を図る。

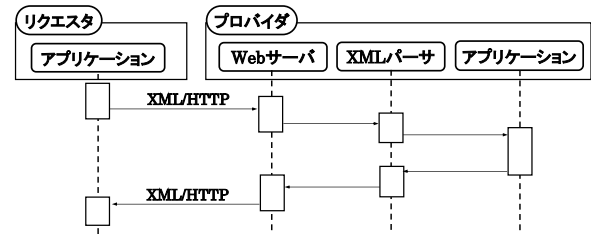


図 2: REST メッセージアーキテクチャ

### 3.1. REST のメッセージ技術の適用

REST のメッセージ技術は、SOAP と異なり、標準化されていないため、REST のメッセージ交換には様々な方式がある。一般的には、HTTP 上で、XML 文書そのまま交換する方式である。

REST のメッセージ交換は、SOAP エンベロープのような構文がないため、図 1 に示す SOAP メッセージ構文の生成や解析を行う必要がない。そのため、REST のメッセージ技術を、少量の情報を送信する Web サービスのメッセージ交換に利用することは、性能向上に有効であると思われる。

### 3.2. REST の制約の適用

REST の制約にキャッシュがある。キャッシュは HTTP GET リクエストに対してのみ有効であり、Web サービスにキャッシュを適用することは、性能向上に有効な手段である。

しかし、SOAP メッセージは、HTTP リクエストに POST を利用し、サービスに対するターゲット URL が SOAP プロセッサを指す。これらの点から、Web サービスキャッシュを効果的に適用することはできない。

そこで、REST の Web サービスアーキテクチャの適用により、効果的にキャッシュを利用できる。

A Lightweight Web Services Architecture with REST.  
† Takashi Ikezaki, Yoichiro Nagahashi, Hiroshi Igaki, Mikio Aoyama, Faculty of Mathematical Sciences and Information Engineering, Nanzan University  
‡ Akira Mori, Graduate School of Mathematical Sciences and Information Engineering, Nanzan University

## 4. 評価

### 4.1. 評価方法

提案したアーキテクチャを検証するために、SOAP メッセージ交換による性能低下を招く場合を想定し、文字列をそのまま返す Echo サービスのプロトタイプを開発し、SOAP と REST の性能を比較した。図 3 に示すように、性能比較の評価尺度として、応答時間と要求処理時間、スループットがある。本稿ではメッセージ交換の性能を比較するためリクエスタ側の応答時間を用いて測定する。

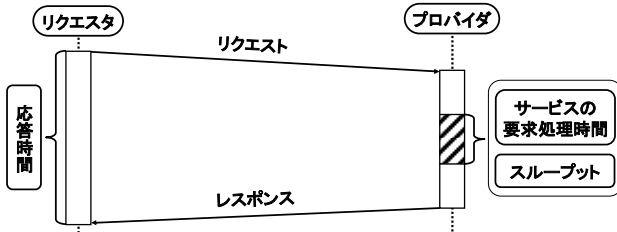


図 3: 評価尺度

### 4.2. 実行回数に対する応答時間の変化

Echo サービスのプロトタイプは、文字列データ (10 バイト) を送信し、受け取ったデータをそのまま返信する。実行回数を変化させて測定し、SOAP と REST のメッセージ交換の性能を比較した。参考に Ajax の平均処理時間も測定し比較した。

図 4 に、SOAP, REST, Ajax[3] の実行回数による平均応答時間を表す。

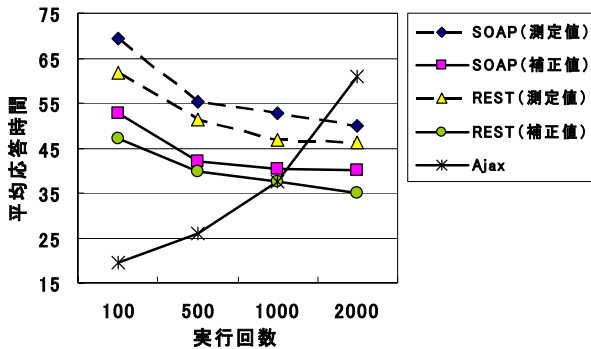


図 4: サービス呼び出しの平均応答時間

図 4 の SOAP, REST の測定値は、応答時間の総和を各実行回数で割った値である。しかし、この値には、ほぼ一定周期で異常な遅延を含む。これは Java のガベージコレクションと思われるため、この特異値を除いた平均応答時間を補正值とする。

Ajax は非同期通信であるため、最初のリクエストを送信後、最後のレスポンスを受信するまでの時間を実行回数で割った値を測定値とする。

図 4 から、REST の応答時間は、SOAP の応答時間に対して、約 10% 短縮されたことが分かる。

また、Ajax は実行回数が 100 回程度の場合、SOAP や REST と比較し高速であるが、1000 回を上回ると逆に平均応答時間が低下する。

### 4.3. データ量に対する応答時間の変化

送信するデータ量の増加による SOAP メッセージと REST メッセージの応答時間の変化を検証した。変化の検証には、図 5 に示すデータ量を用いてそれぞれ 100 回実行し、1 回の平均応答時間を比較した。図 5 にその検証結果を示す。

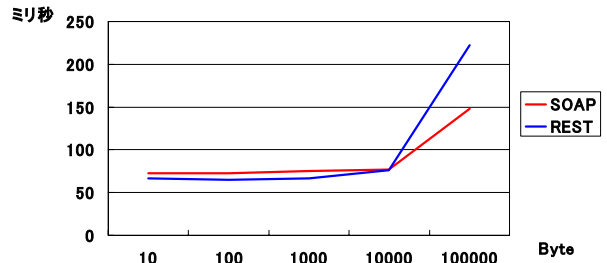


図 5: データ量における応答時間の差

図 5 から、少量のデータのメッセージ交換には、REST メッセージの応答時間が短いため、効率的にメッセージ交換が行える。

一方、データ量が増加するにつれて、REST メッセージよりも SOAP メッセージの応答時間が短くなることから、状況に応じて SOAP メッセージと REST メッセージを使い分ける必要がある。

## 5. 関連研究

SOAP メッセージにおける性能問題の関連研究として、SOAP の形式(スタイル)や送るデータの種類における処理速度の違いに関する検証を行っている [1]。しかし、SOAP のメッセージ交換では、プロバイダ側の XML シリアライズ処理時間は総処理時間の 10.7%、XML デシリアライズ処理時間は総処理時間の 30.7% となっていることから、SOAP 構文の生成や解析はメッセージ交換のボトルネックとなっている。

## 6. まとめ

本研究では、SOAP のメッセージ交換における性能の問題を解決する方法として、REST を用いた Web サービスアーキテクチャを提案した。

SOAP のメッセージ交換が性能低下を招く場合において、プロトタイプを用いて、SOAP と REST の応答時間を比較し、REST の有効性を検証した。

## 参考文献

- [1] A. Ng, et al., An Evaluation of Contemporary Commercial SOAP Implementations, Proc. of the 5<sup>th</sup> Australasian Workshop on Software and System Architecture, 2004, pp.64-71.
- [2] R. T. Fielding, Architectural Styles and the Design of Network-based Software Architectures, PhD Dissertation, 2000, <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>.
- [3] D. Crane, et al., Ajax in Action, Manning Pub., 2005.