



SMDに対する リソース指向アーキテクチャの提案

1

Group-J4

2009SE026 福田雄介

2009SE093 岩下 翔

2008MI160 中根拓也

目次

- 研究の背景
- 発表の経緯
- リソースとは
- URIとは
- アドレス可能性
- ステートレス性
- 接続性
- 統一インタフェース
- SMDでの課題
- 参考文献

研究の背景

- 近年,SMDの普及に伴い,SMDに対応したwebサービス開発の要求が高まっている
- 特に,webサービス開発で,RESTが広まってきている

しかし...

RESTに基づくアーキテクチャが,明確に標準化されていない問題点



RESTに基づくSMDに適したアーキテクチャを提案



- RESTの単純な構造を用いることは,デバイスの処理能力の制約上での開発に,効果があるのではないかと?

発表の経緯

- 前回SMDに対するSOAについて調べた.
- 今回は,ROAに関して知識を得る必要があったので,調べてみた.

リソースとは

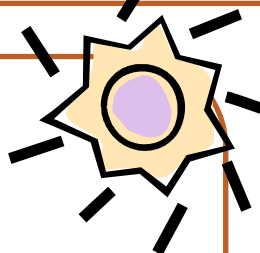
リソース

- Web 上に存在する情報/データを指す



例

- 東京の天気予報
- 1,024の次の素数
- 1,024から5つ目の素数
- クラゲに関する情報
- “REST”に関する検索結果
- etc...



このような情報が、リソースに成り得る。

しかし...

- これだけでは”リソース”とは言えない。



リソースがリソースであるための条件とは??

- リソースは少なくともURIを1つ持っていなければならない。

URIとは

“URI=リソースの名前+アドレス” で表される



- URIを持たない物はリソースではない
- リソースを示すもの以外として,URIは存在しない

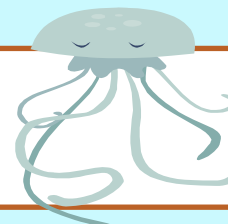


- リソースとURIは,誰もが分かりやすいように対応させるべきである.

例

クラゲに関する情報

<http://www.example.com/wiki/Jellyfish>



- 開発者側から見ても,利用者から見ても,
“クラゲに関する情報”だと一目見て分かること出来る



アドレス可能性

- URIはアドレス可能である.

例 クラゲに関する情報を検索する

クラゲ 検索

検索ページ



クラゲとは……



結果ページ

★ブックマーク

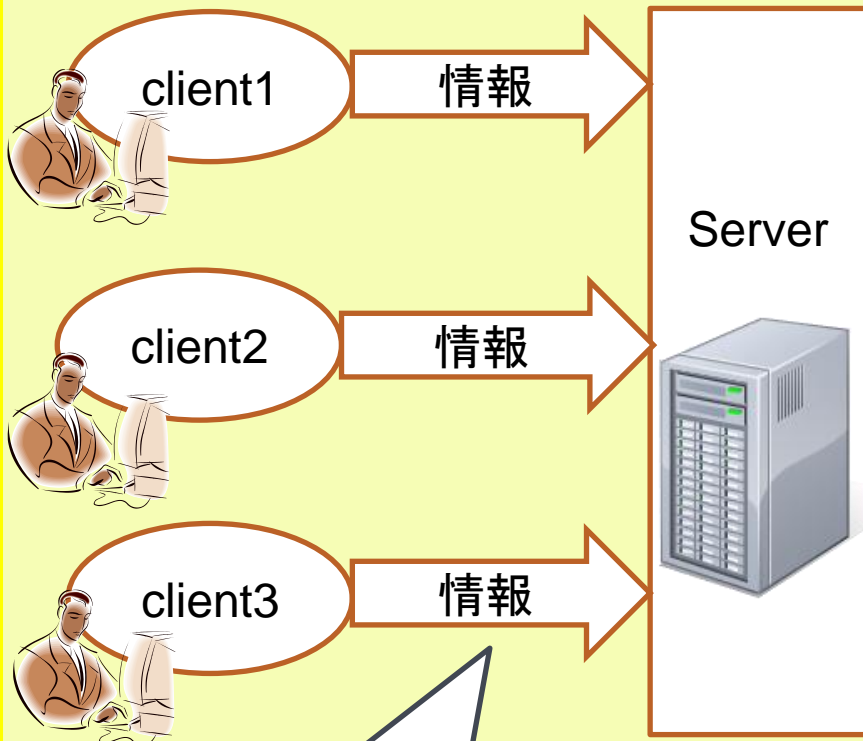
- お気に入りのサイトをブックマークすることで、一発でアクセス可能



ステートレス性

- ステートレス: サーバにセッション状態がない事を指す
- 制約: リクエストには処理に必要な全ての情報を含まなければならない

ステートレスの状態



1つの情報でやり取りが終了

メリット

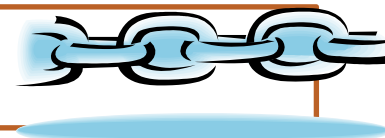
- 1つのリクエストだけをモニタリングすればよい.
- 何か不具合が起こっても、最初からやり直さなくても良い
- スケーラビリティの向上

デメリット

- 認証情報などを繰り返し送ることによるパフォーマンスの減少

接続性

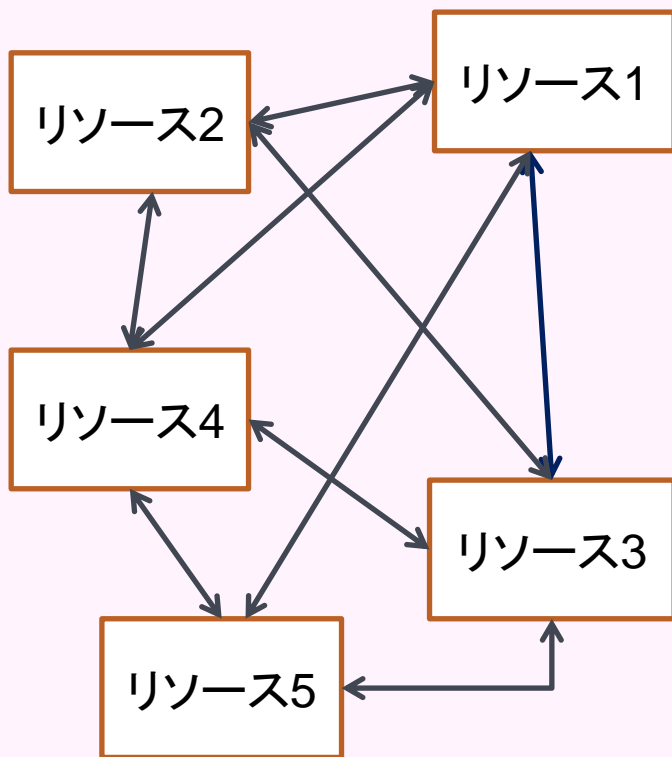
- リソース自身がリンクをもっており,それらをたどってアプリケーションの状態遷移が可能.



- リソース同士は単純に **URI** と **HTTP (リンク)** で結びついている.

- リンクを管理するサーバである,中央リンク管理サーバや,ローカルリンクサーバ等が不要.

- URIさえあれば,アプリケーションの連携が簡単に行える.
 - 疎結合の実現
 - 拡張性も有り

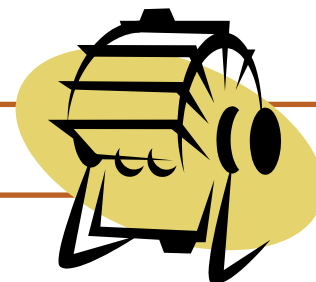


各リソースは,接続されているべき



統一インターフェース

- コンポーネント間のインターフェースを統一する.



| | |
|---------|---------------------|
| GET | リソースの所得(Retrieve) |
| PUT | リソースの更新(Update) |
| DELETE | リソースの消去>Delete) |
| POST | リソースの作成(Create) |
| HEAD | リソースのメタデータの所得 |
| OPTIONS | リソースがサポートするメソッドを調べる |

SMDでの課題

- SMDに対してRESTに基づくWebサービスを開発する上で考える点.

最適な要求の生成

- 頻繁に起こる要求について最適化
- データペイロードについて最適化
- ユーザの待ち時間を考慮

このSMDでの課題を解決するような、アーキテクチャを考えていきたい。

参考文献

- RESTful Webサービス
 - 著:Leonard Richardson ・ Sam Ruby
- E-Words
 - <http://e-words.jp>
- RESTとROA
 - http://blogs.ricollab.jp/webtech/wp-content/uploads/2008/03/rest_and_roa_for_ricoh.pdf