

Slide URL

<https://vu5.sfc.keio.ac.jp/slide/>

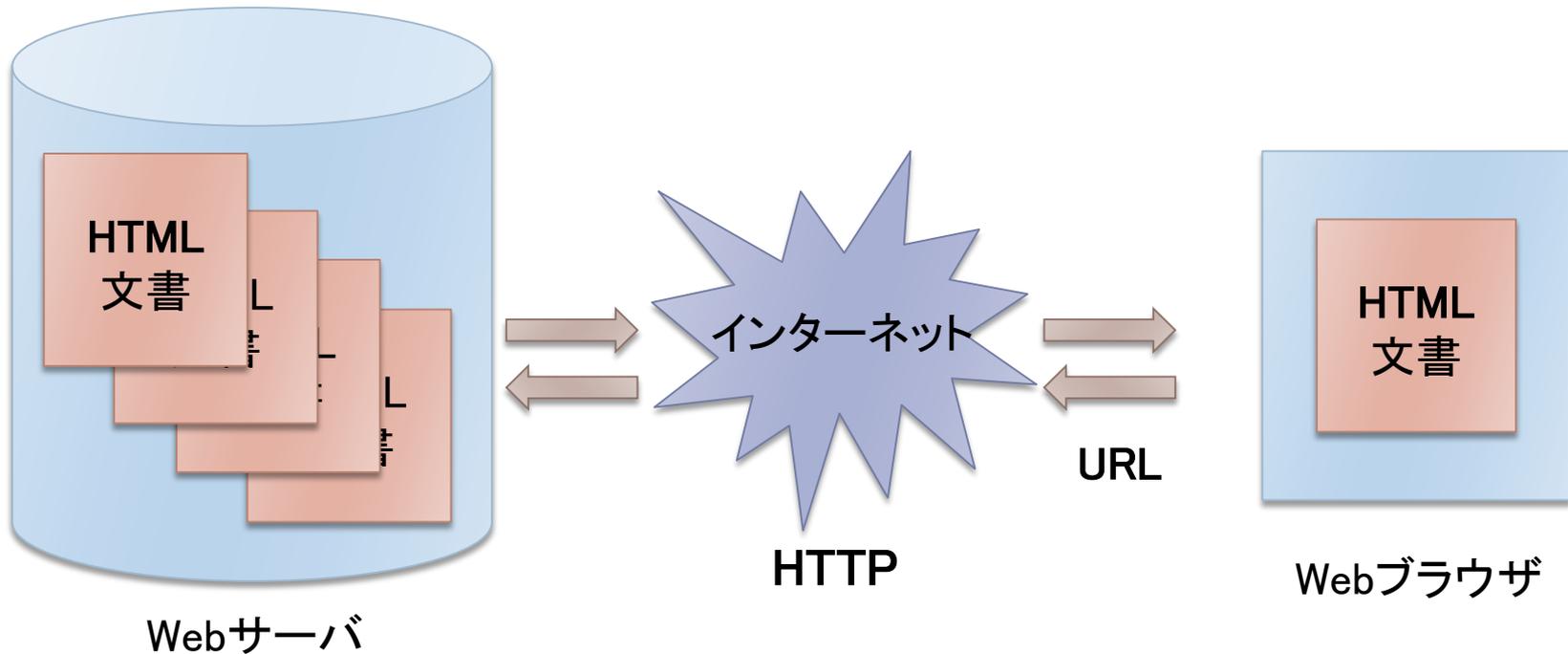
Web情報システム構成法

第4回 Webサーバ

萩野 達也 (hagino@sfc.keio.ac.jp)

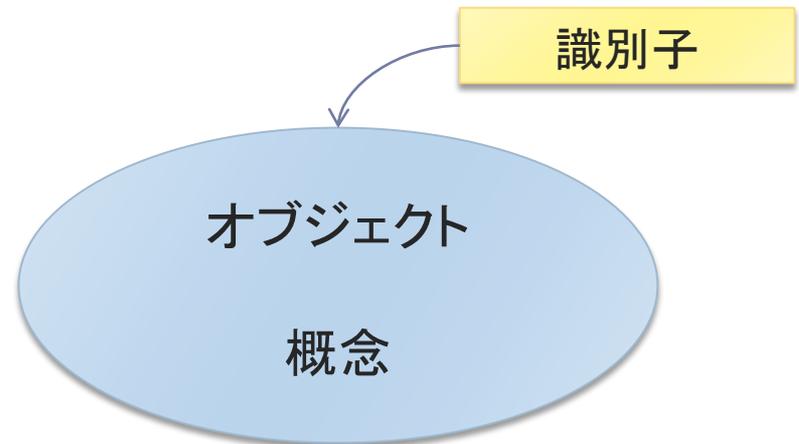
Webの基本構成要素

- ▶ 文書をHTMLとCSSで準備する
- ▶ 文書をHTTPで取得・操作する
- ▶ URLを使って文書を指し示す



URI (Uniform Resource Identifier)

- ▶ 識別子
 - ▶ モノ(オブジェクト)や概念を名前や数字で識別する
- ▶ 身近な識別子:
 - ▶ 学生を識別
 - ▶ 本を識別
 - ▶ 電話を識別
 - ▶ PCを識別
 - ▶ インターネットでの識別



URIのシンタックス

`http://www.sfc.keio.ac.jp/teacher/hagino.html?title=web#lecture`

スキーマ

オーソリティ

パス

問い合わせ

フラグメント

▶ スキーマ

- ▶ URIの種類
- ▶ プロトコル

▶ オーソリティ

- ▶ ホスト名
- ▶ サーバ名

▶ パス

- ▶ オーソリティ内での場所
- ▶ ファイルのパス

▶ 問い合わせ

- ▶ 問い合わせパラメータ
- ▶ インタラクション

▶ フラグメント

- ▶ 文書内での位置

URIに関する公理

- ▶ 普遍性(Universality)
 - ▶ すべてのWeb資源はURIを持つ
- ▶ 大域性(Global Scope)
 - ▶ URIはどこでも同じ意味を持つ
 - ▶ 一意性
- ▶ 同一性(Sameness)
 - ▶ URIは常に同じものを意味する
 - ▶ 意味が同じであり, 内容は異なることもある
- ▶ 不透明性(Opacity)
 - ▶ URIだけから資源の種類を知ることはできない
 - ▶ 資源の表現を調べないと種類等はわからない

Webにおける概念・識別子・表現

▶ 概念

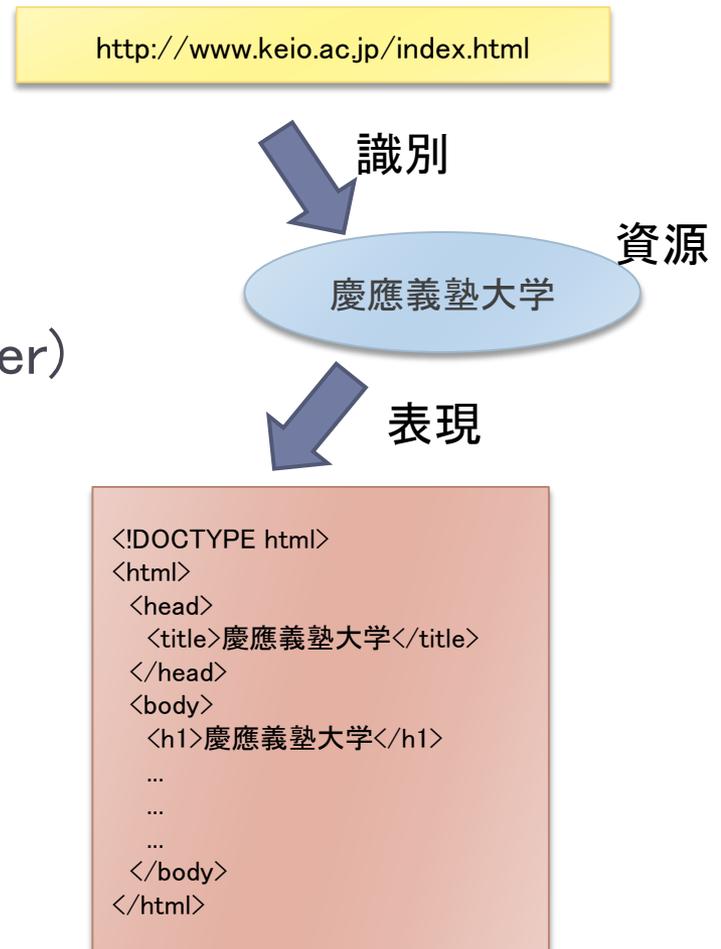
- ▶ Web資源(Resource)

▶ 識別子

- ▶ URI (Uniform Resource Identifier)

▶ 表現(Representation)

- ▶ HTML+CSS
- ▶ XML
- ▶ RDF
- ▶ GIF



URL • URN • IRI

▶ URL (Uniform Resource Locator)

- ▶ 資源の場所を識別するURI
- ▶ 例: http, ftp, file, mailto

▶ URN (Uniform Resource Name)

- ▶ urn:<nid>:<nss>
- ▶ NIDはIANAに登録
- ▶ <http://www.iana.org/assignments/urn-namespaces/urn-namespaces.xml>
- ▶ 79 URNが登録されている(2023-03-14)
- ▶ 3gpp, 3gpp2, adid, alert, bbf, broadband-forum-org, cablelabs, ccscs, cdx, cgi, clei, ddi, dev, dgiwg, doi, dslforum-org, dvb, ebu, eic, eidr, epc, epcglobal, etsi, eurosystem, example, fdc, fipa, gdst, geant, globus, gsma, hbbtn, ieee, ietf, iptc, isan, isbn, iso, issn, itu, ivis, lei, lex, liberty, mace, mef, meta, mpeg, mrn, nato, nbn, nena, newsml, nfc, nzl, oasis, ogc, ogf, oid, oipf, oma, onem2m, onf, pin, publicid, pwid, reso, s1000d, schac, service, smpte, swift, tva, uci, ucode, uuid, web3d, xmlorg, xmpp

▶ IRI (Internationalized Resource Identifier)

- ▶ 国際化されたURI
- ▶ URIではアスキー文字しか使うことができない
- ▶ IRIではユニコード文字を使うことができる

URIの利用

▶ Webページの場所

▶ URL

```
https://www.sfc.keio.ac.jp/about_sfc/video.html
```

▶ 仕様

▶ DTD

```
http://www.w3.org/TR/html4/loose.dtd  
urn:ietf:rfc:2141
```

▶ RFC

▶ Web資源

▶ 人

```
https://ja.dbpedia.org/page/東京都
```

▶ 都市

```
https://ja.dbpedia.org/page/福澤諭吉
```

HTTPとFTPの違い

▶ FTP

- ▶ 遠隔ファイル操作
- ▶ インターネットの初期の段階から利用
- ▶ ユーザは認証のためFTPサーバにログインする
- ▶ フリーソフトなどの配布のためにAnonymous FTPが存在
- ▶ 制御とデータ転送のための2つのTCPコネクションを利用
- ▶ データの種類は2つのみ:テキスト・バイナリ

▶ HTTP

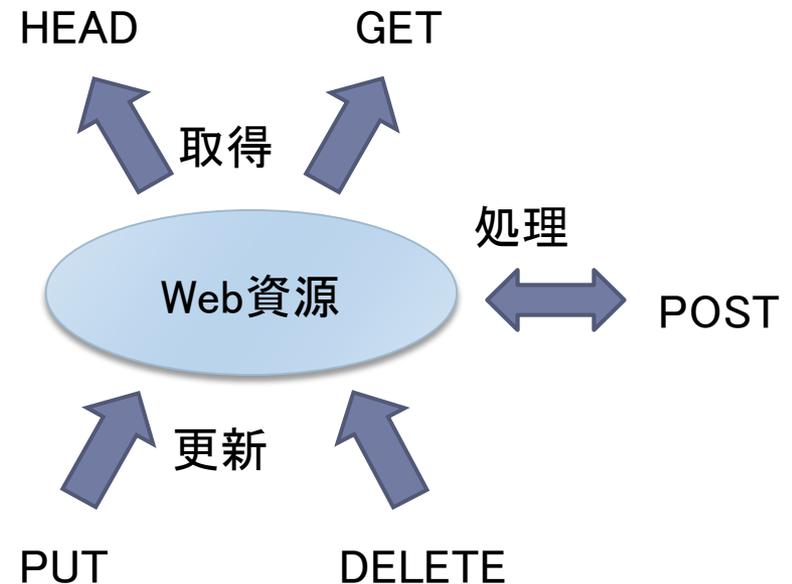
- ▶ Web資源の操作
- ▶ 認証は原則行わない
- ▶ 一つのTCP接続を利用
- ▶ マルチメディアをサポート

HTTP (Hypertext Transfer Protocol)

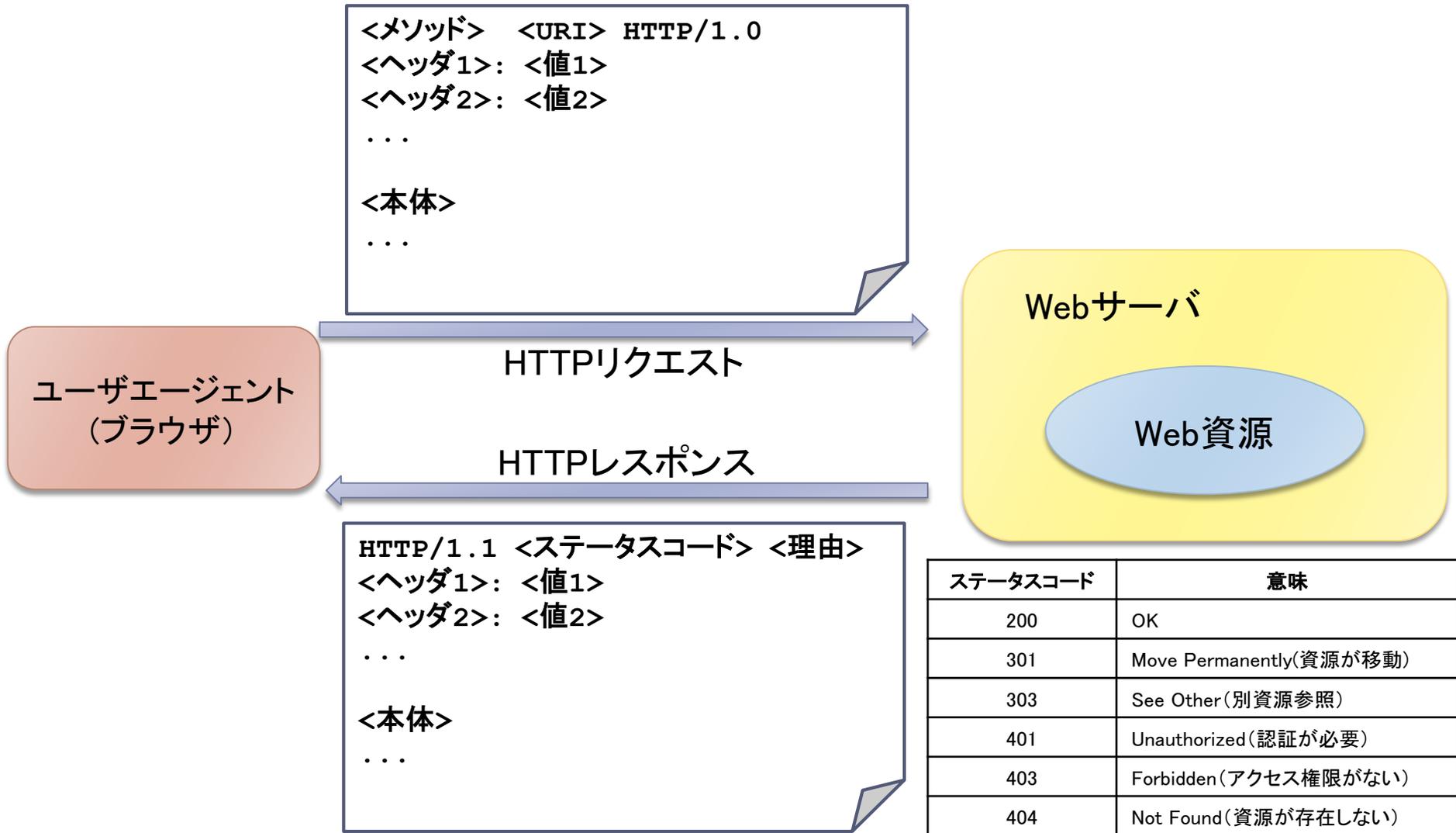
▶ Web資源を操作するプロトコル

▶ 5つの主なメソッドを持つ

- ▶ HEAD
 - ▶ 資源の情報を得る
- ▶ GET
 - ▶ 資源の表現を取得する
- ▶ PUT
 - ▶ 資源を作成あるいは更新する
- ▶ DELETE
 - ▶ 資源を削除する
- ▶ POST
 - ▶ データを処理するために送る



HTTPリクエストとレスポンス



GETとHEADメソッド

▶ GET

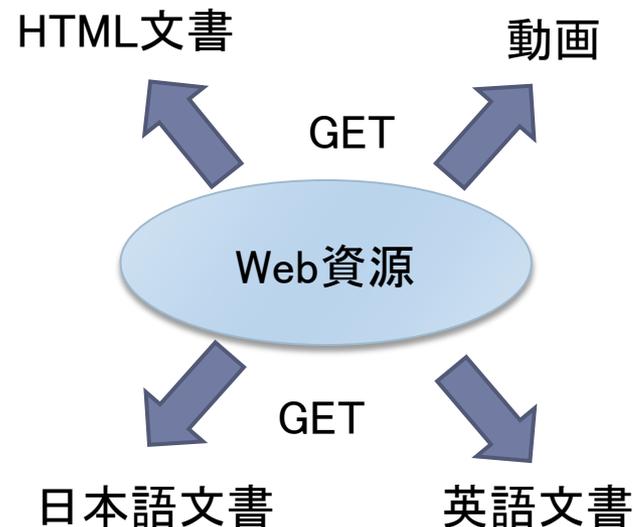
- ▶ 資源の一つの表現を取得する
- ▶ コンtentネゴシエーション
- ▶ 言語ネゴシエーション

▶ HEAD

- ▶ 資源およびその表現のに関する情報を取得する
- ▶ GETの一部

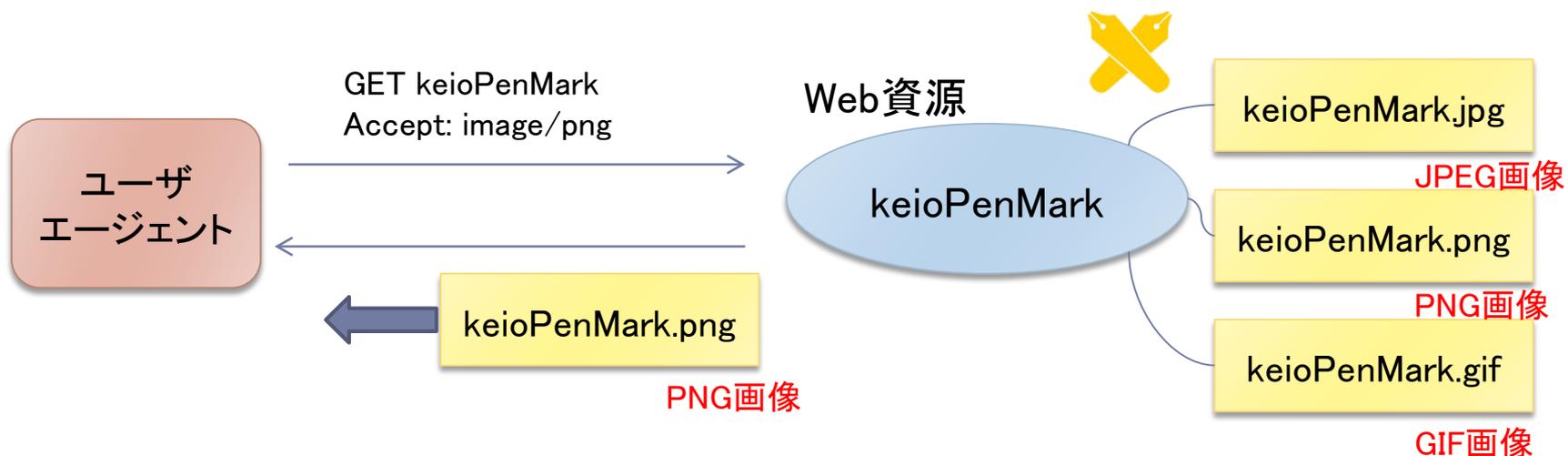
▶ GETの性質

- ▶ GETは何度行っても安全
- ▶ GETは冪等 (Idempotent)
- ▶ GETは副作用がない

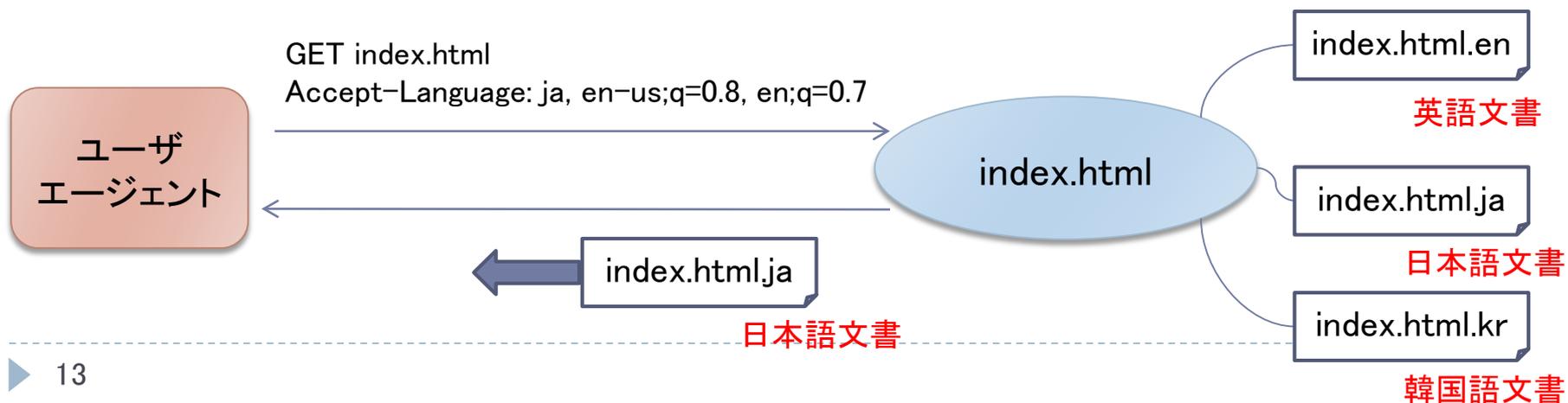


Contentネゴシエーション

▶ 文書/画像形式



▶ 言語



PUTとPOSTメソッド

▶ PUTメソッド

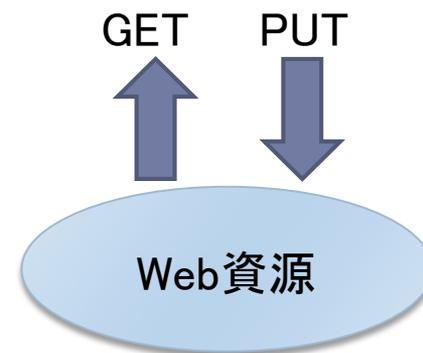
- ▶ 資源を作成あるいは更新する
- ▶ GETの逆
- ▶ 通常のブラウザはPUTを使わない

▶ POSTメソッド

- ▶ 資源にデータを送る
- ▶ 資源はデータを処理する
- ▶ HTMLのFORMで利用

▶ GET vs POST

- ▶ FROMのmethod属性によりGETまたはPOSTを指定
- ▶ 副作用がない場合にGET
- ▶ 資源の更新など副作用があるときにはPOST
- ▶ POSTは幂等ではない



Other Features of HTTP

- ▶ 転送
 - ▶ 資源が別のところに移動
- ▶ 認証
 - ▶ ユーザ名とパスワードによる認証
 - ▶ Basic認証
 - ▶ Digest認証
- ▶ Virtual Host
 - ▶ 一つのサーバで複数のWebサイトをホスト
 - ▶ DNSの別名を利用
- ▶ TCP/IPの効率的利用
 - ▶ Persistent connection
 - ▶ keep-alive
 - ▶ パイプライン処理
 - ▶ 並列処理
- ▶ Proxyキャッシュ制御
 - ▶ max-age
 - ▶ no-cache
 - ▶ public あるいはprivate
- ▶ WebDAVへの拡張
 - ▶ COPY, MOVE, LOCK, UNLOCK

課題：多言語対応

- ▶ 自分の好きな海外の街を紹介のトップページを日本語と英語の両方で準備しなさい。
 - ▶ 両方とも同じHTML5で記述すること。(CSSやJavascriptは用いないこと)
 - ▶ できるだけ同じ内容にしなさい。
- ▶ CNSでの設定
 - ▶ 現在のトップページ `town.html` とは別に用意する。
 - ▶ `multi.html` を日英で用意する場合には,
 - ▶ `multi.html` は作成してはいけない(存在する場合には削除する)
 - ▶ `multi.html.ja` に日本語版(`town.html` をコピーする)
 - ▶ `multi.html.en` に英語版を新たに作成する
 - ▶ `multi.html.ja` から `multi.html.en` への言語切替のリンクを作成(逆も)
 - ▶ Webサーバがフォルダ内を探ることができるように, `other/`に`read`パーミッションを与える
 - `chmod o+r ~/public_html/web23`

- ▶ 提出
 - ▶ 作成したHTMLファイルをCNS上に置き、トップページの言語自動切換えURLを提出してください。
 - ▶ 上の例の場合には, `https://web.sfc.keio.ac.jp/~ログイン名/web23/multi.html`

HTML5によるページの構造化

multi.html.ja

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8" />
  <title>私の好きは街</title>
</head>
<body>
  <header>
    <h1>エジンバラ</h1>
  </header>
  <nav>
    <ul>
      <li><a href="landmark.html">名所</a></li>
      <li><a href="park.html">公園</a></li>
      <li><a href="multi.html.en">English</a></li>
    </ul>
  </nav>
  <article>
    <p>世界遺産にもなっている美しい街です. </p>
  </article>
  <footer>
    <p>問い合わせ: <a href="help.html">ヘルプ</a></p>
  </footer>
</body>
</html>
```

ヘッダ
ロゴなど

メニュー

このページの本文

フッター
ヘルプナビなど

まとめ

▶ URI

- ▶ Web資源の識別
- ▶ URL, URN, IRI

▶ HTTP

- ▶ Web資源を操作するプロトコル
- ▶ HTTP URI
- ▶ メソッド: HEAD, GET, PUT, DELETE, POST