ソフトウェアアーキテクチャ
第12回　WORLD WIDE WEB

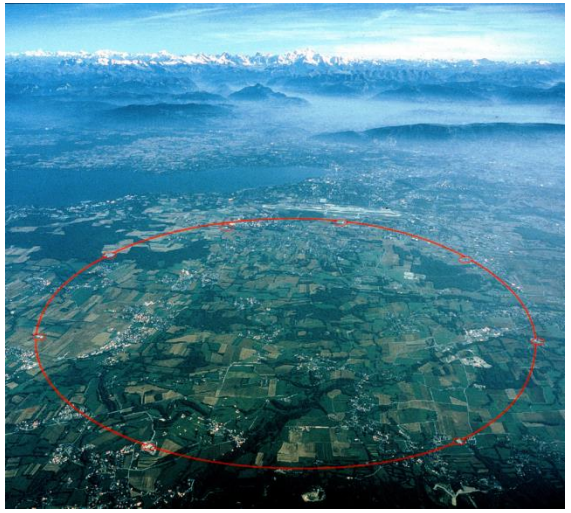
環境情報学部
萩野　達也

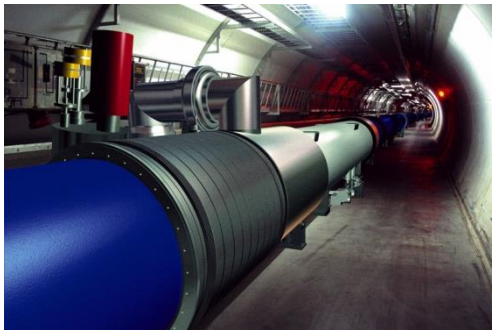
スライドURL

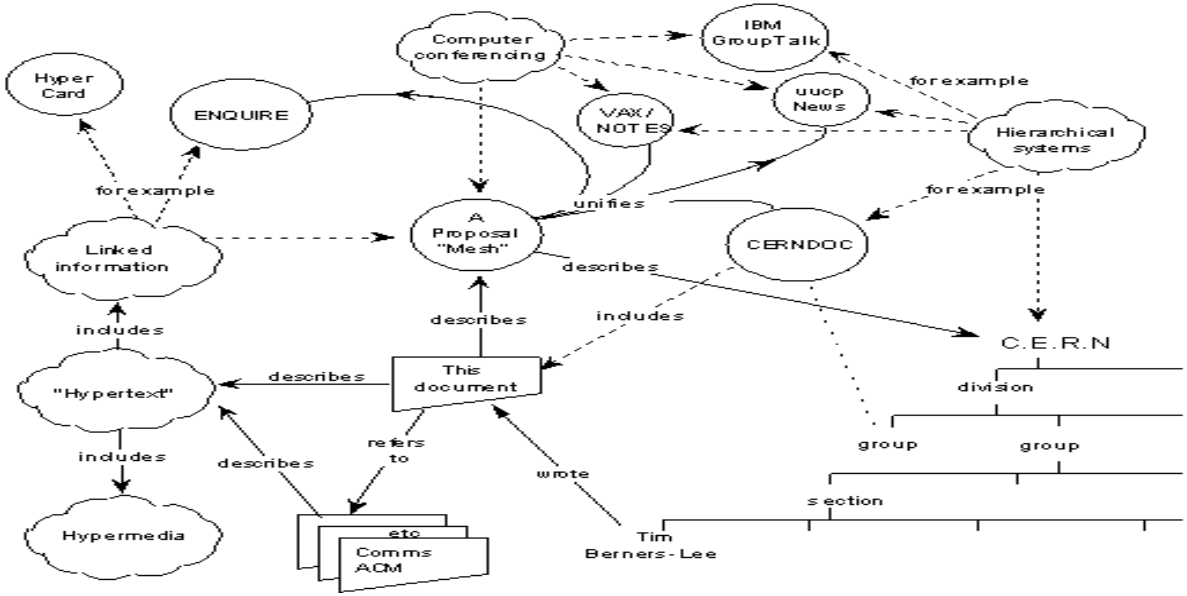
https://vu5.sfc.keio.ac.jp/slide/

Webの誕生

- 1989年にスイスジュネーブCERNで誕生
  - Tim Berners-Lee
  - CERNにおける情報管理のため

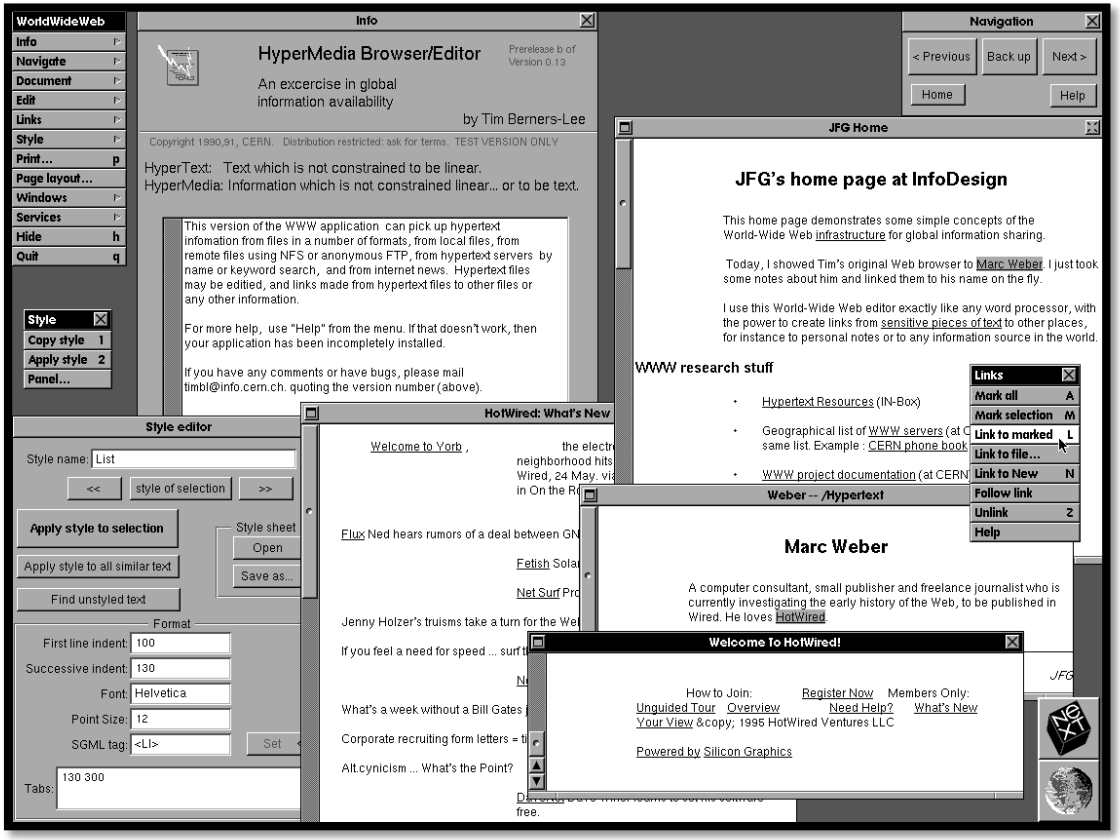






Web = ハイパーテキスト + インターネット

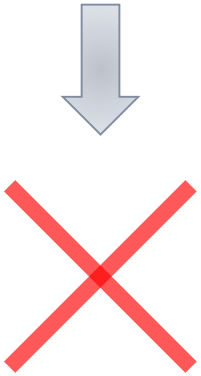
最初のWebサーバとWebブラウザ



3つの方法の比較

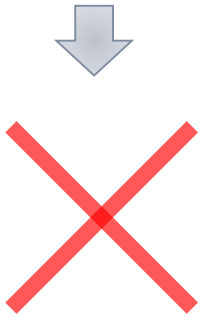
木構造

管理はしやすい
現実の関係を表せない



キーワード

検索などは速い
キーワードを前もって決めないといけない
利用者はキーワードが分からない



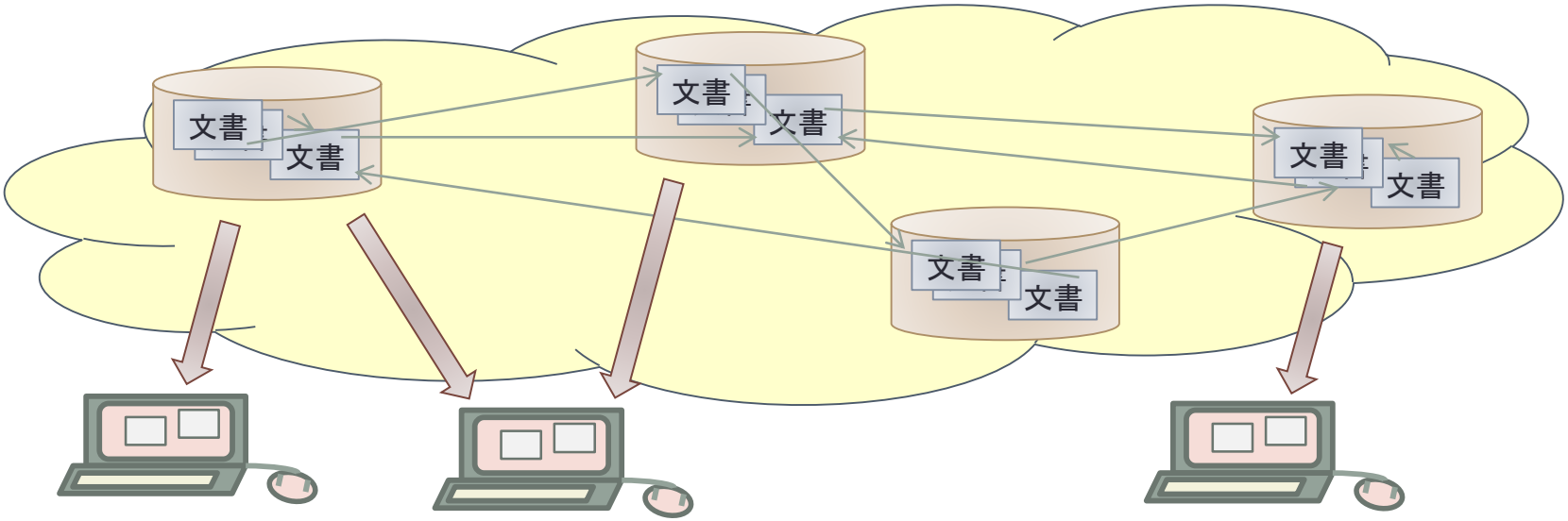
ハイパーテキスト

自由度が高い
現実の関係を表すことができる
キーワードをノードとして表す

Webとは何か

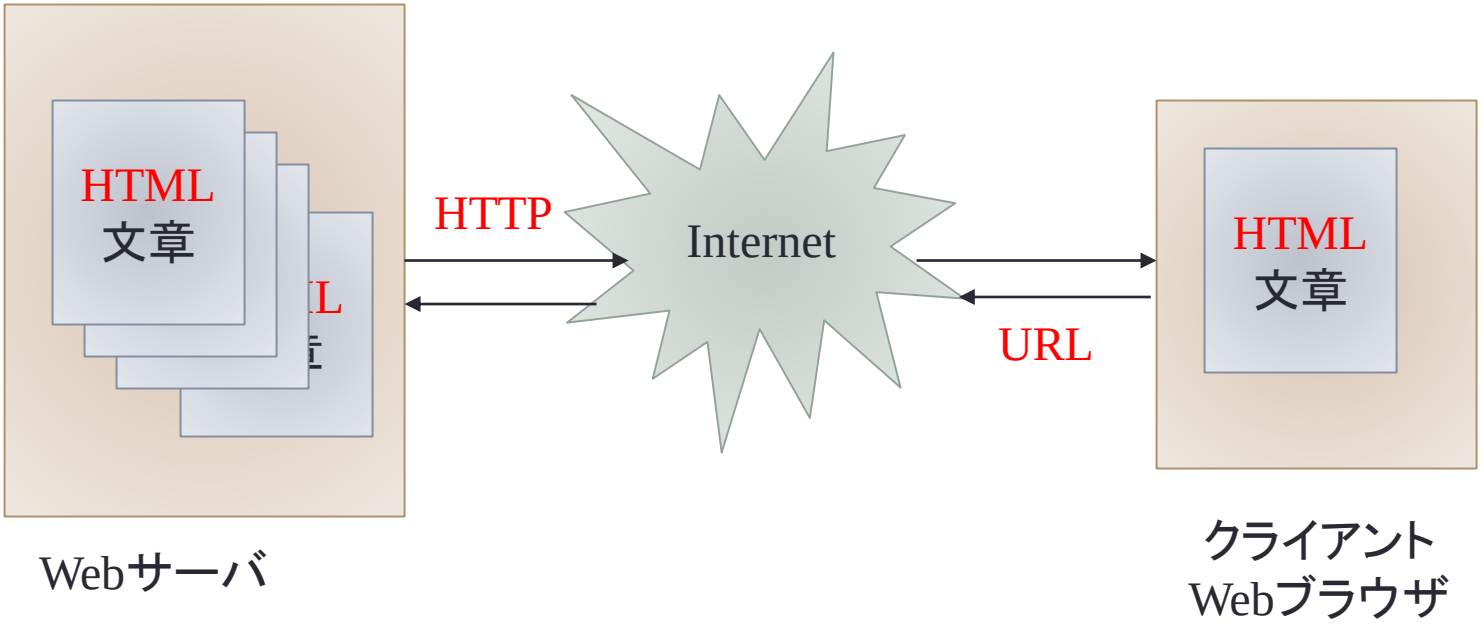
Web = インターネット + ハイパーテキスト

- インターネット
  - 世界中のネットワークを結ぶネットワーク
- ハイパーテキスト
  - 他のテキストへのリンクを含むテキスト



Webの基本的仕組み

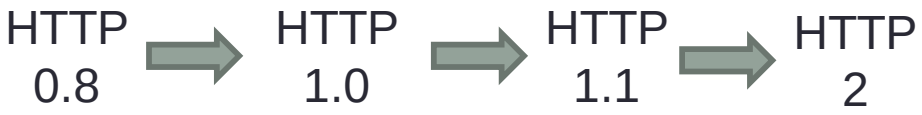
- 文章をハイパーテキストとしてHTMLで記述
- 文章の場所をURLで指定
- HTTPによりサーバからブラウザに転送



Webの最初の重要な発明

- HTML + CSS
  - Webページの記述言語
  - HTML: Hypertext Markup Language
    - 内容の記述
  - CSS: Cascading Style Sheet
    - スタイルの記述
    - 1990年にはなく，後で追加

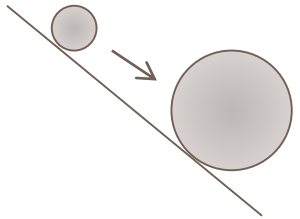
- HTTP: Hypertext Transfer Protocol
  - Webページの転送
  - Anonymous FTPの単純化
  - マルチメディアへの対応
  - 言語ネゴシエーション機能



- URL: Uniform Resource Locator
  - Webページの位置を示す
  - Hypertextのポインタ

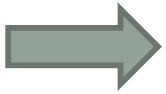
Webはどうしてこんなに普及したのか？

- 無料にした
  - Gopherはライセンスの問題で普及しなかった(?)
- オープンなシステム
  - だれでもが参加可能
  - 検索ロボットが自動的に追加
  - ネットワーク効果



雪だるま式に普及

- 厳密さにこだわらなかった
  - リンクが切れていることも容認(404 Not Found)
  - HTMLの文法エラーも多数



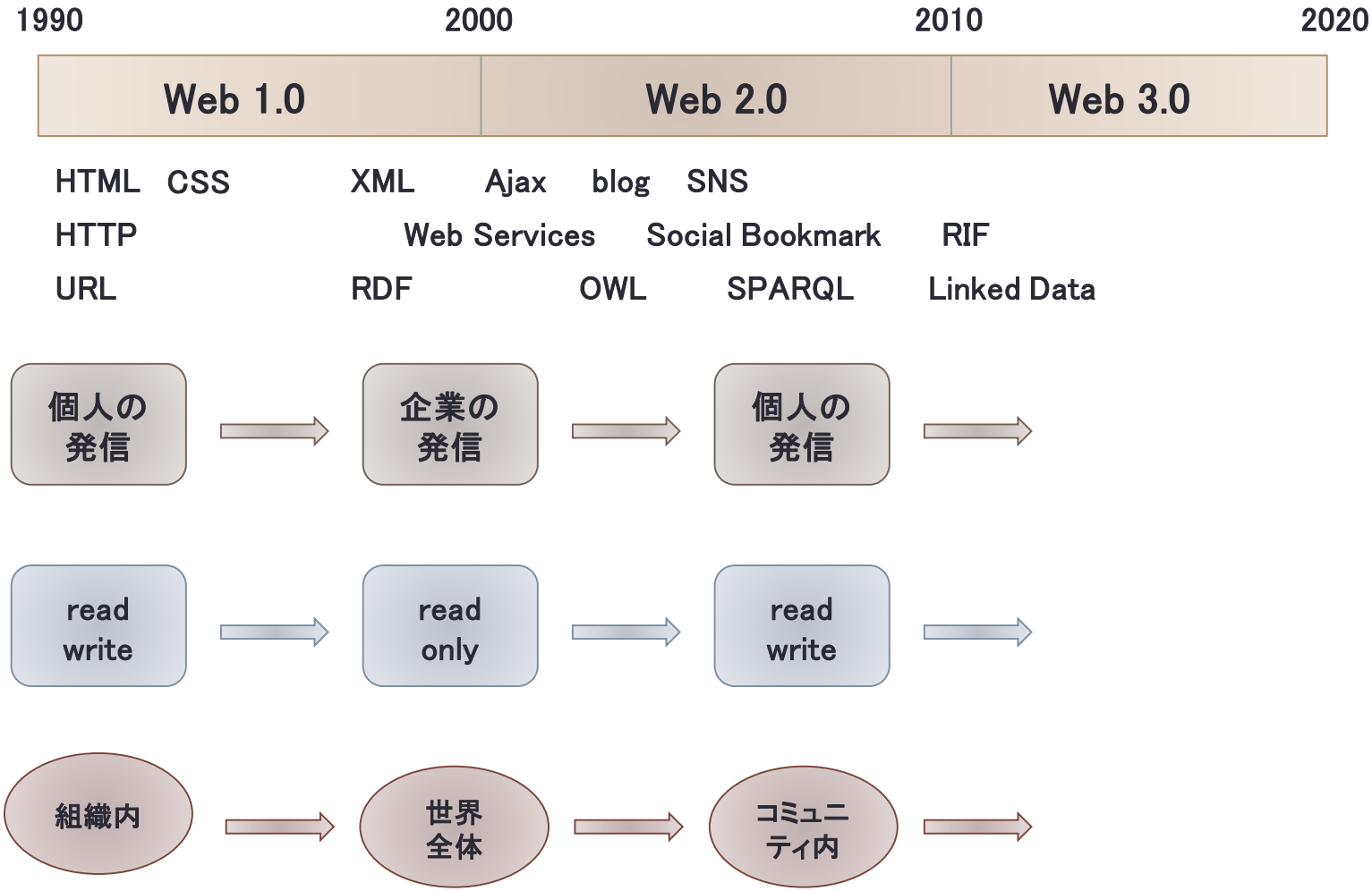
ハイパーテキストとしては欠陥

- 標準化への努力
  - IETF
  - World Wide Web Consortium

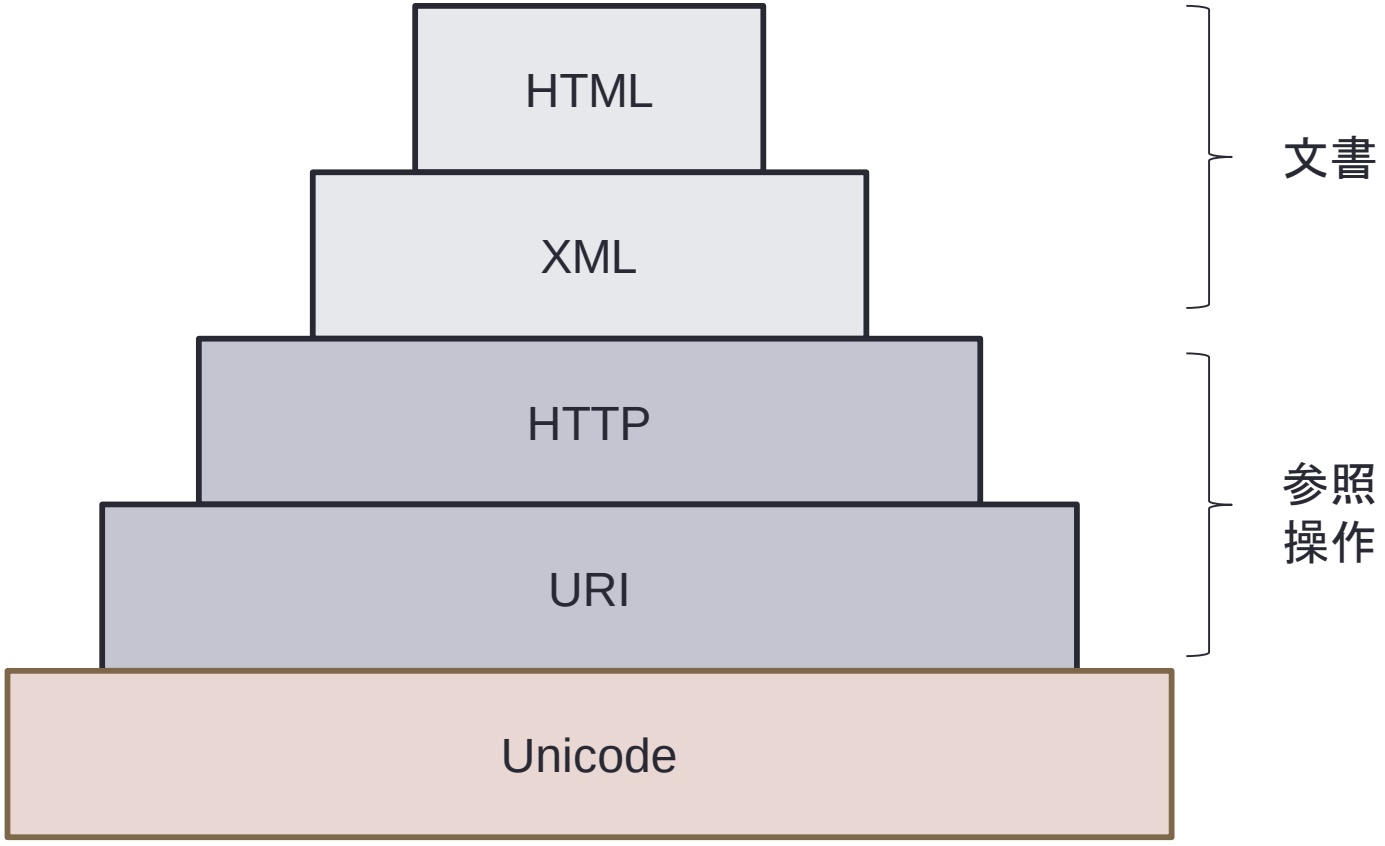
IETF

W3C WORLD WIDE WEB consortium

Web 1.0からWeb 2.0へ



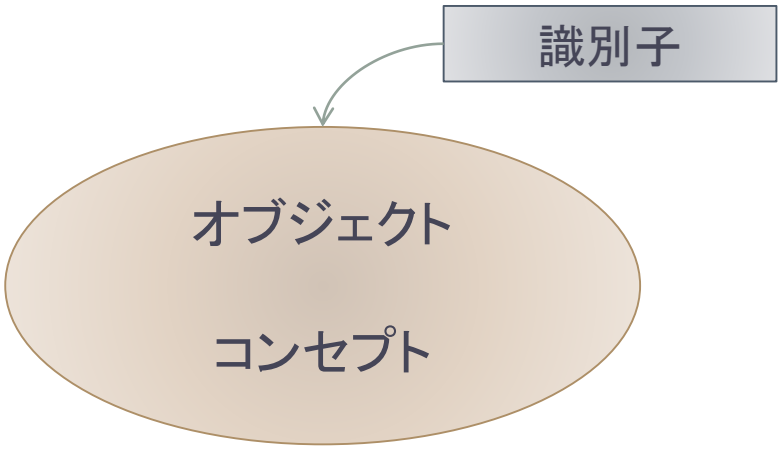
Webの全体構成要素



URI (Uniform Resource Identifier)

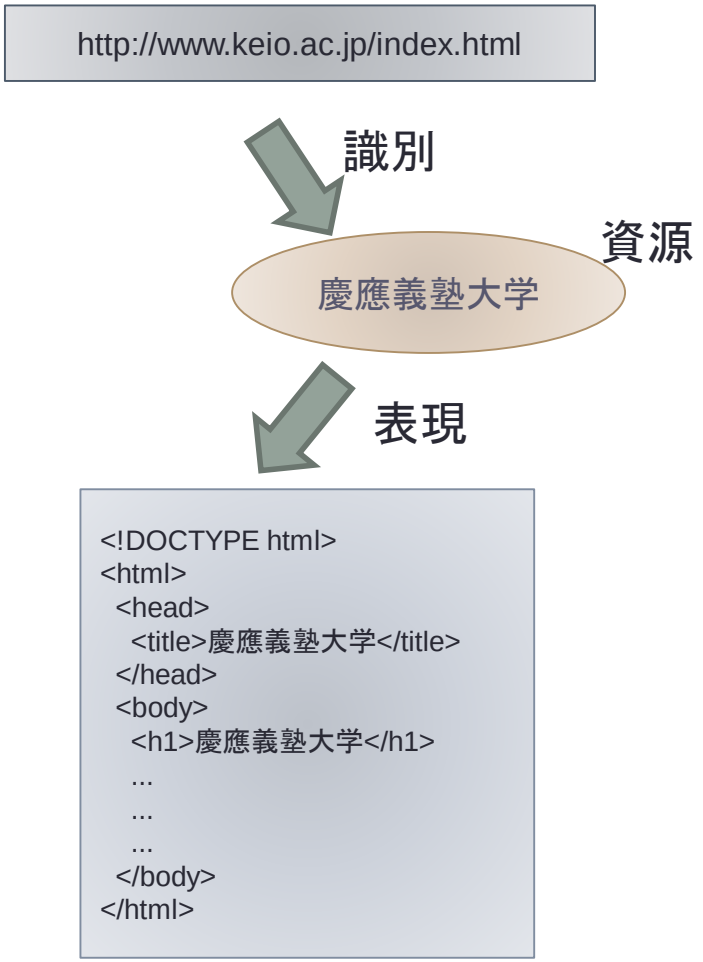
- 識別子
  - 物（オブジェクト）や概念（コンセプト）を識別する名前や番号

- 身近な識別子をあげてみよう？
  - 個人の識別子
  - 本の識別子
  - PCの識別子
  - 場所の識別子
  - ・・・・



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
  - GIF



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(Schema)
  - URIの種類
  - プロトコル
- オーソリティ(Authority)
  - ホスト名
  - サーバ名
- パス(Path)
  - オーソリティ内の位置
  - ファイル名
- 問い合わせ(Query)
  - 検索などの検索語
- フラグメント(Fragment)
  - 文書内の位置

httpでは問い合わせまでWebサーバ，フラグメントはブラウザが処理

URIの公理

- 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

URL, URN, IRI

- URL (Uniform Resource Locator)
  - 資源の場所を表すURI
  - http
  - ftp

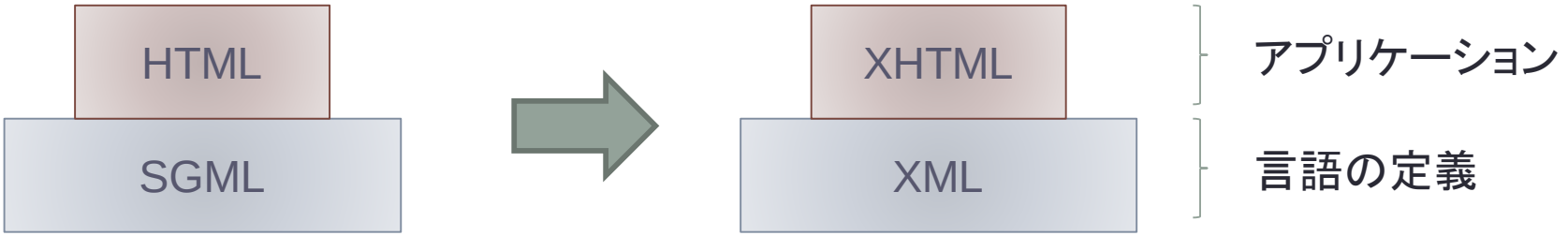
- URN (Uniform Resource Name)
  - urn:<nid>:<nss>
  - NIDをIANAに登録することによって一意性を確保
  - https://www.iana.org/assignments/urn-namespaces/urn-namespaces.xhtml
  - 2017年6月26日現在62個が登録
  - 例：ISBN

- IRI (Internationalized Resource Identifier)
  - 国際化されたURI
  - URIではパスに漢字等は利用できず16進数を使う
  - ホスト名，パスにUnicodeを利用

XMLとは

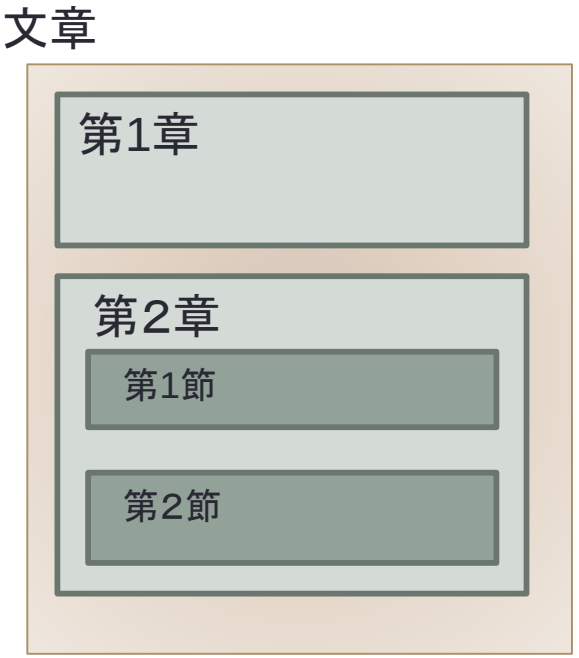
- XML (Extensible Markup Language)
  - マークアップ言語の1つ
  - JIS: 拡張可能なマーク付け言語
  - 1998年2月W3Cが1.0を制定
  - 現在は1.0と1.1の2つのバージョン

- XMLの必要性
  - SGMLが古すぎる. インターネット時代には即していない
  - HTMLだけですべての文書が記述できるわけではない
  - 文書だけでなくデータも記述したい
  - 複数の文書を混ぜたい



構造化文書

- 文章は構造を持つ
  - 段落，章，節，目次
  - 履歴書，申請書
- マークアップ（markup）
  - 文章の構造の指定
  - marking upが語源
  - SGMLにより定着
  - SGMLではタグ(tag)によりマークアップ



<coffee price="250">カフェラテ</coffee>

マークアップ　　マークアップ

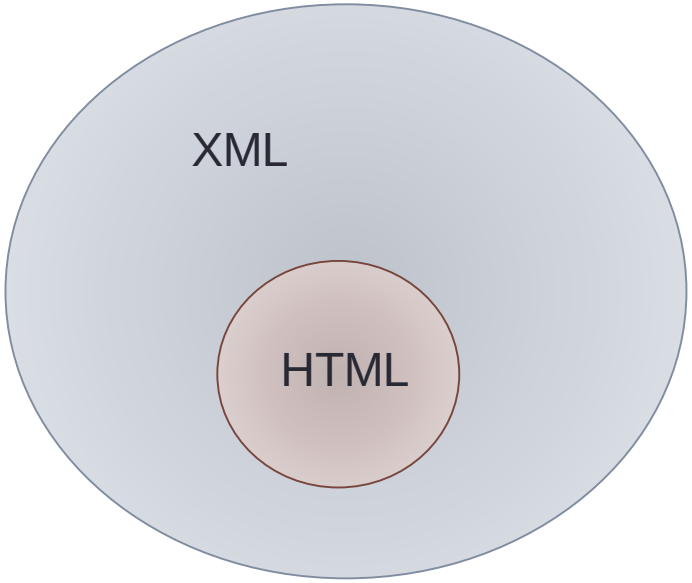
XML文書の構成要素

- XML宣言
  - XML文書であることを示す
  - 文字コードの指定
- Element
  - 開始タグと終了タグでマークアップ
  - 空要素タグ
- CharData
  - 文字データ
- Reference
  - <やAなどの参照
- CDSect
  - CDATAセクション
- PI
  - 処理命令
- Comment
  - コメント

```
<?xml version="1.0" encoding='EUC-JP'?>
<!-- 慶応SFCのレストラン -->
<restaurant>
  <name>慶応レストラン</name>
  <place>SFCキャンパス内</place>
  <menu>
    <item price="150">コーヒー</item>
    <item price ="250">カフェラテ</item>
    <item price="400">サンドイッチ</item>
    <item price="700">スパゲッティ</item>
  </menu>
  <open from="10:00" to="17:00" />
  <networks>
    <network type="無線LAN" />
    <network type="有線LAN" />
  </networks>
  <misc>
    快適なひとときを味わえます.
  </misc>
</restaurant>
```

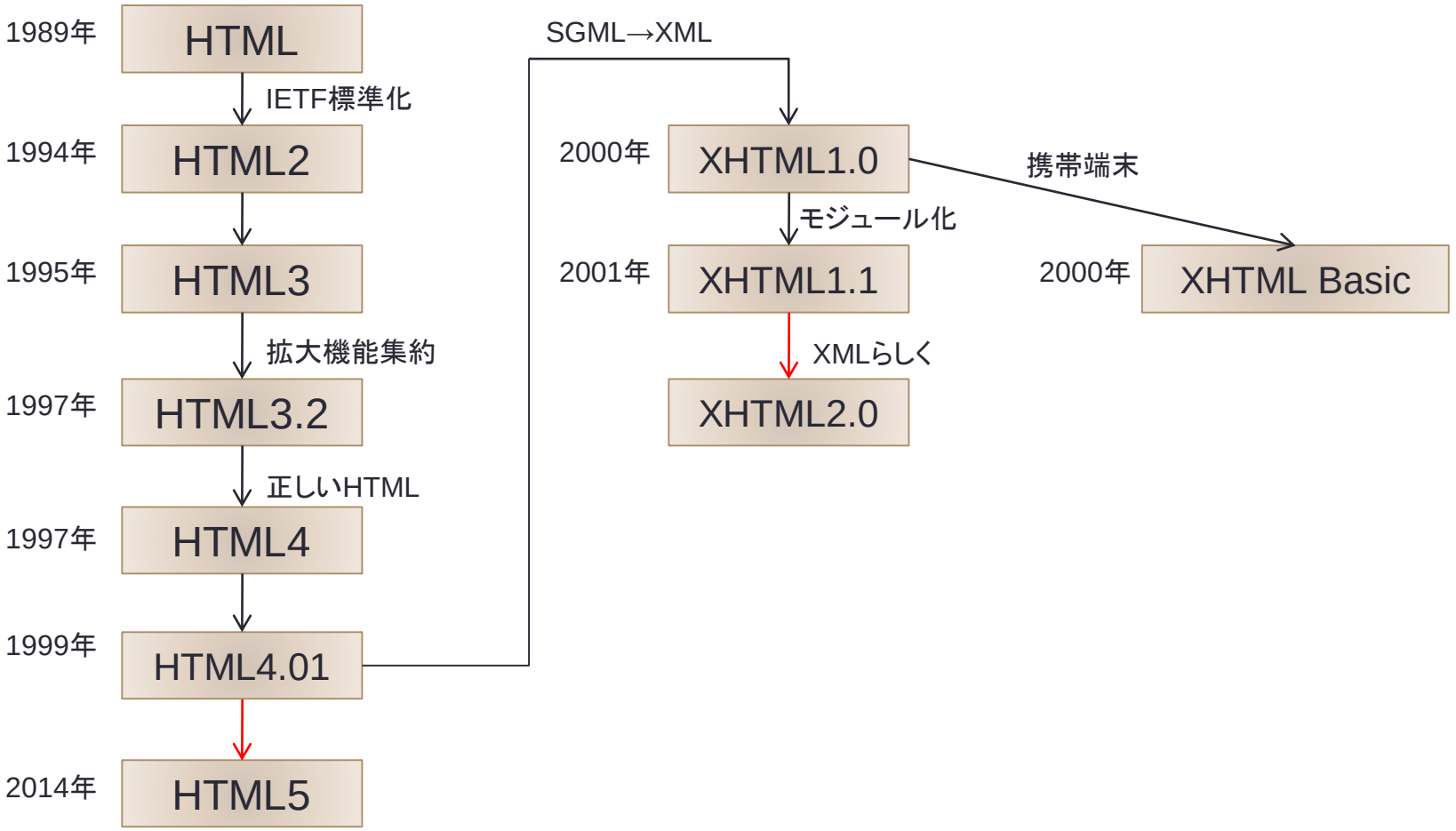
HTML

- HTML
  - XML(SGML)アプリケーション
  - ハイパーテキスト文書
- HTMLの特徴
  - 内容と表現の分離
  - スタイルはCSSで指定
  - 直交する技術を用いる
    - 内容: HTML
    - スタイル: CSS
    - プログラミング: Javascript



直交技術　ビデオ
- 映像
- 音声
- サブタイトル

HTMLの変遷


1989年
HTML
IETF標準化
1994年
HTML2
1995年
HTML3
拡大機能集約
1997年
HTML3.2
正しいHTML
1997年
HTML4
1999年
HTML4.01
2014年
HTML5
SGML→XML
2000年
XHTML1.0
携帯端末
モジュール化
2001年
XHTML1.1
2000年
XHTML Basic
XMLらしく
XHTML2.0

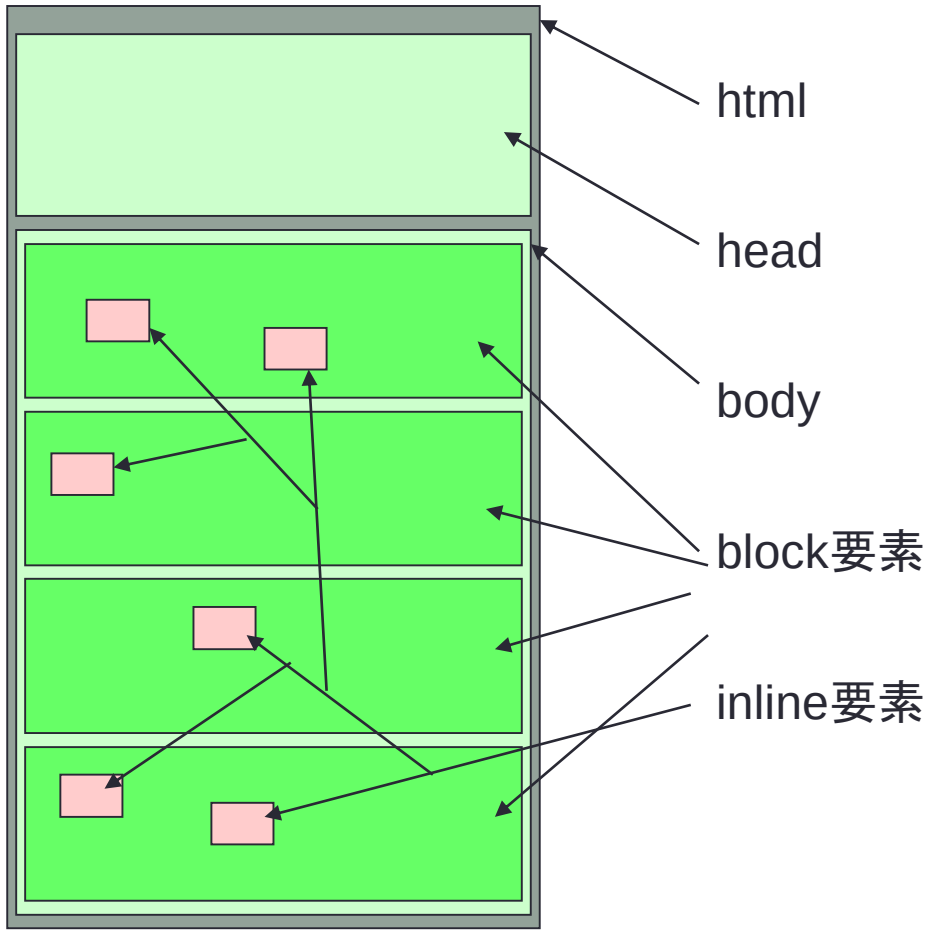
簡単なHTML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
  <head>
    <title>My first HTML document</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
    <link rel="stylesheet" href="sample.css" type="text/css"/>
  </head>
  <body>
    <h1>簡単なHTML</h1>
    <p>HTMLは簡単にだれでもが書くことができます. </p>
    <p>複数の段落から構成されています. </p>
  </body>
</html>
```

- DOCTYPE宣言によりバージョン指定する
  - 初心者にはDOCTYPE宣言は長くて覚えにくい
  - HTML5では単純化
    - <!DOCTYPE html>

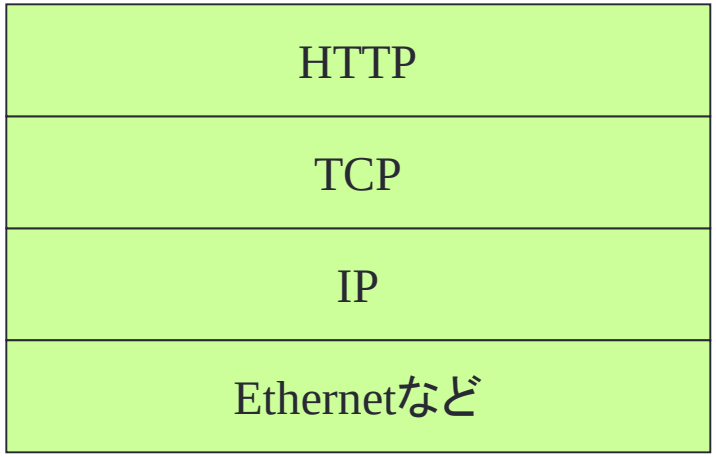
HTMLの要素の分類

- 文書全体を構成する要素
  - html, head, bodyなど
  - section, articleなど
- 段落を構成する要素
  - block要素
  - h1, h2, ul, ol, tableなど
- 文の中で用いられる要素
  - inline要素(text要素)
  - i, b, em, strongなど



HTTP

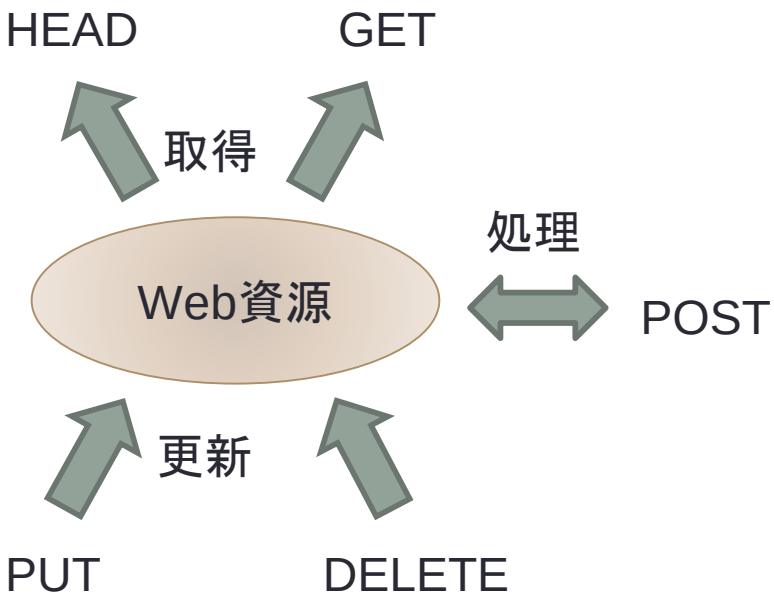
- Hyper Text Transfer Protocol
- TCP/IP上のプロトコル
  - ポート80
- Anonymous FTPを単純化したもの



HTTP (Hypertext Transfer Protocol)

- Web資源を操作するプロトコル

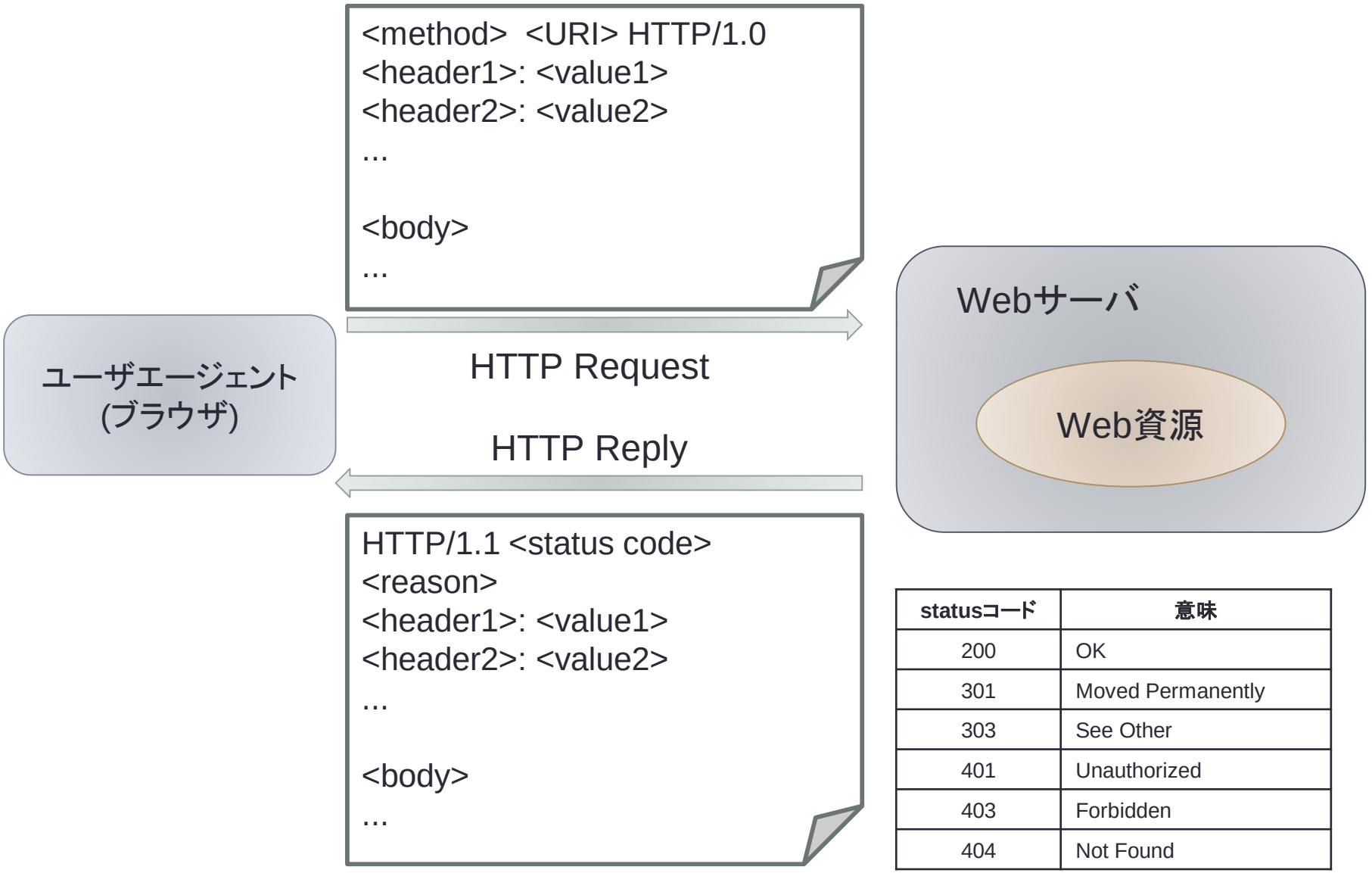
- 主なメソッド
  - HEAD
    - 資源の情報を取得する
  - GET
    - 資源の表現を取得する
  - PUT
    - 資源の作成または更新
  - DELETE
    - 資源の削除
  - POST
    - データを資源に送り処理する



HTTPの変遷

- HTTP 0.9
  - 1991年
  - Request
    GET <uri> CR LF
  - Reply
    <html document>
- HTTP 1.0
  - 1992年, 1996年 RFC1945
  - Request
    <Method> <uri> HTTP/1.0 CR LF
    [<request header>*]
    [CR LF <data>*]
  - Reply
    HTTP/1.0 <status code> <reason> CR LF
    [<response header>*]
    [CR LF <data>*]
- HTTP 1.1
  - 1999年11月 RFC2616
    - Virtual Hosting
    - TCP/IP Persistent Connection
    - TCP/IP Pipelining
    - Proxy Cache Control
- HTTP/2
  - 2015年5月 RFC7540
    - Googleによって開発されたSPDYを拡張
    - HTTP/1.1を置き換えるものではない
    - HTTP/1.1を高速にする
    - 多重接続を可能にする
    - フロー制御
    - ヘッダ圧縮
    - サーバプッシュ

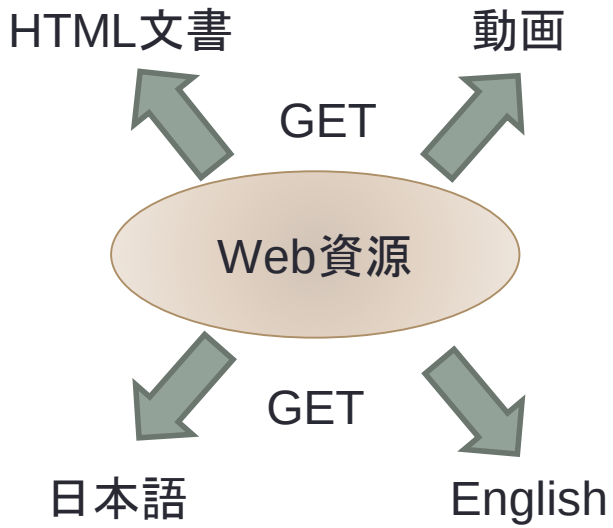
HTTPのRequestとReply



statusコード	意味
200	OK
301	Moved Permanently
303	See Other
401	Unauthorized
403	Forbidden
404	Not Found

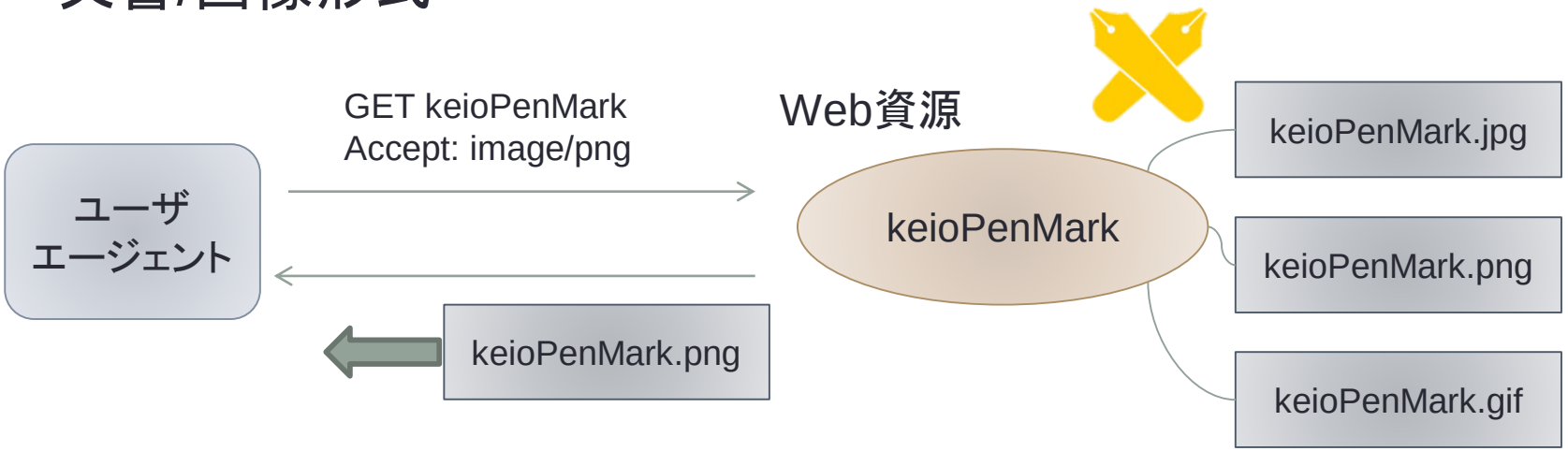
GETとHEADメソッド

- GETメソッド
  - Web資源の表現の一つを取得する
  - Contentネゴシエーション
  - 言語ネゴシエーション
- HEADメソッド
  - Web資源あるいはその表現に関する情報を取得する
  - GETの一部分
- GETの性質
  - GET何度使っても安全
  - GETは冪等（idempotent）
  - GETに副作用はない

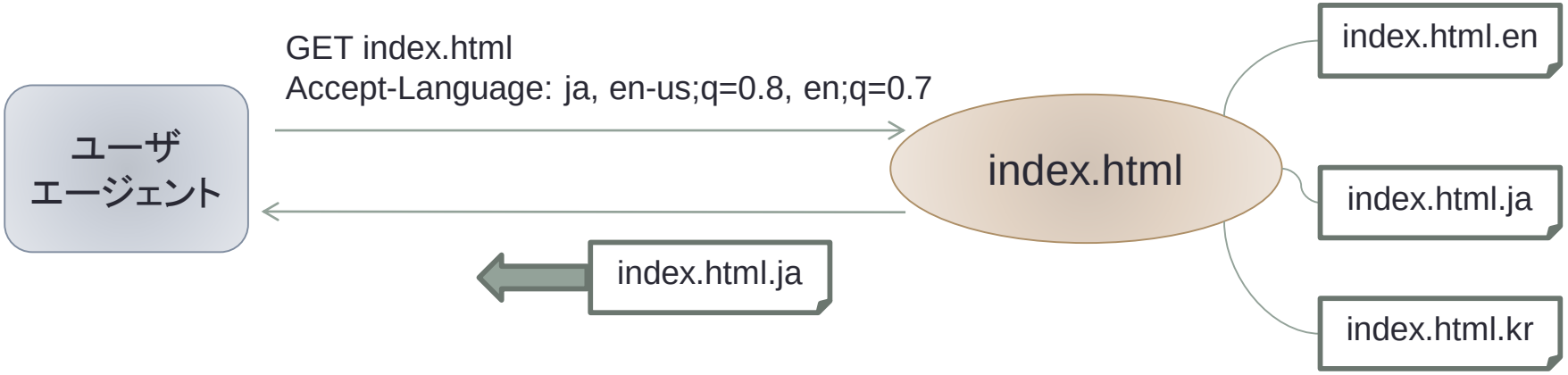


Contentネゴシエーション

- 文書/画像形式

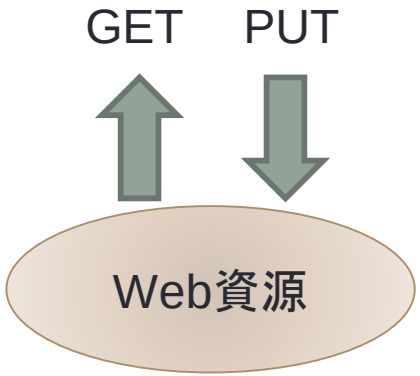


- Language format



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
  - FORM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は冪等ではない

HTTP Header

- General Header
  - Date
  - Pragma
- Request Header
  - Authorization
  - From
  - If-Modified-Since
  - Referer
  - User-Agent
- Response Header
  - Location
  - Server
  - WWW-Authenticate
- Entity Header
  - Allow
  - Content-Encoding
  - Content-Length
  - Content-Type
  - Expires
  - Last-Modified
- Additional Header
  - Accept
  - Accept-Charset
  - Accept-Encoding
  - Accept-Language
  - Content-Language
  - Link
  - MIME-Version
  - Refer-After
  - Title
  - URI

HTTPのStatusコード

statusコード	意味
200	リクエスト成功
201	作成成功
202	リクエスト受理
204	内容なし
301	恒久的に移動した
302	一時的に移動している
303	他を参照せよ
305	Proxyを利用しなくてはいけない

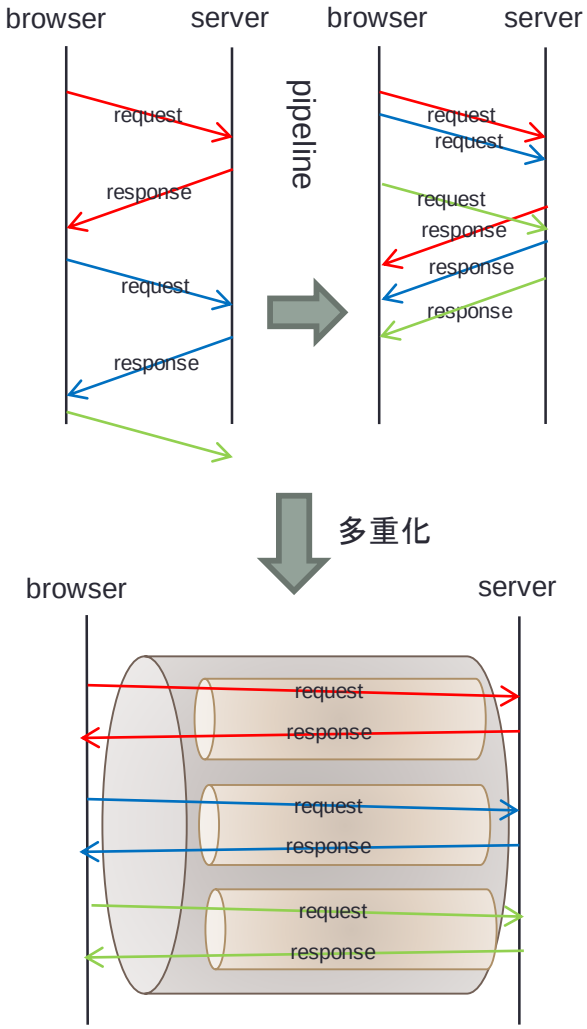
statusコード	意味
400	リクエストが不正である
401	認証が必要である
402	支払いが必要である
403	アクセスが禁止されている
404	資源が存在しない
405	メソッドが許されない
500	サーバ内部エラー
501	実装されていない

HTTPのその他の機能

- 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あるいは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
- 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

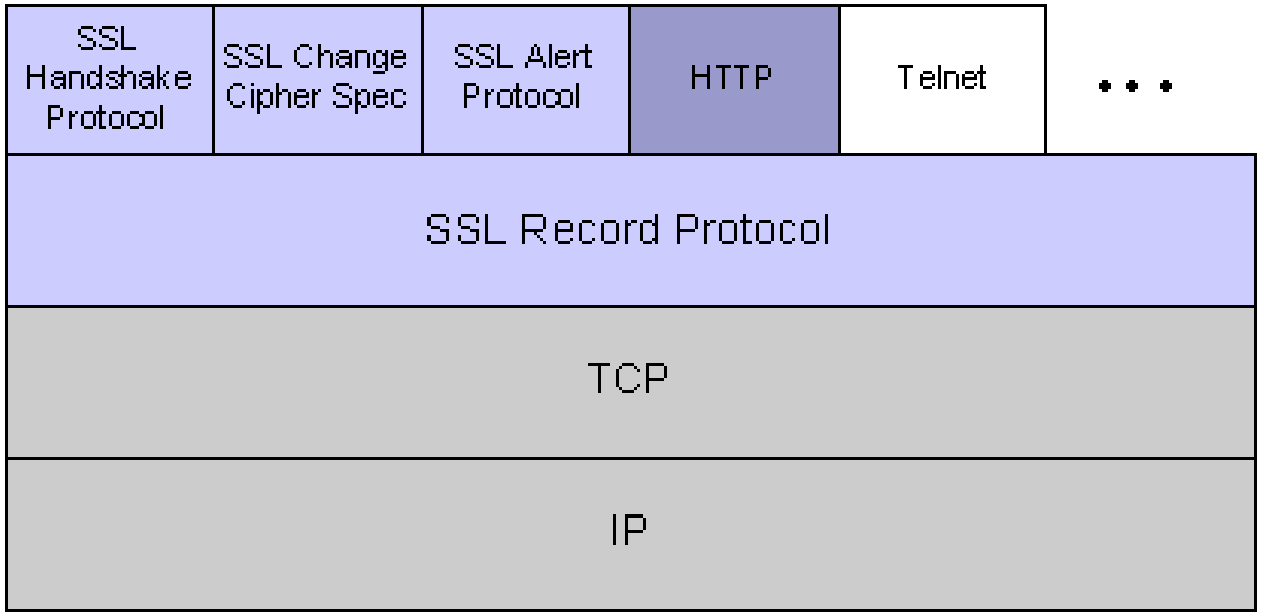
HTTP/2

- 2015年5月 RFC7540
  - Googleが開発したSPDYが元になっている
- 特徴
  - HTTP/1.1を置き換えない
    - HTTP/1.1を高速にする
  - 多重化
    - パイプラインでは要求が順番に処理されていた
    - 多重化では優先度に応じて要求は処理される
  - フロー制御
    - 返信がオーバフローしないように制御
  - ヘッダ圧縮
    - HPACKにより圧縮
  - サーバプッシュ
    - サーバがクライアントのキャッシュに前もってデータを注入しておくことができる



HTTPS

- SSL (Secure Sockets Layer)を利用した通信上にHTTPを実現
  - ホストの認証
  - 通信の暗号化



http://www.modssl.org/docs/2.8/ssl_intro.html

まとめ

- Webについて
  - 歴史
  - URL
  - XML
  - HTML
  - HTTP
- Webのその他の話題
  - 検索エンジン（Page Rank）
  - Webサイトデザイン（情報アーキテクチャ）
  - Webデータ（セマンティックWeb, Linked Data）