

# Web について

SecPrj Intro-phase

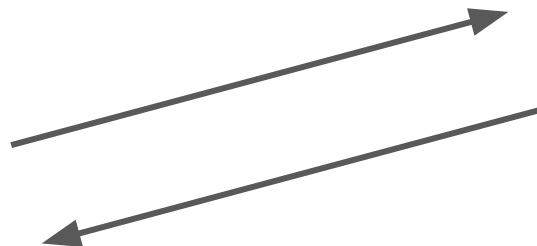
# Web (WWW / World Wide Web) とは

**Web**とは、インターネット上で標準的に用いられている文書の公開・閲覧システム。文字や図表、画像、動画などを組み合わせた文書を配布することができる。文書内の要素に別の文書を指し示す参照情報(**ハイパーリンク**)を埋め込むことができる「**ハイパーテキスト**」(hypertext)と呼ばれるシステムの一つである。

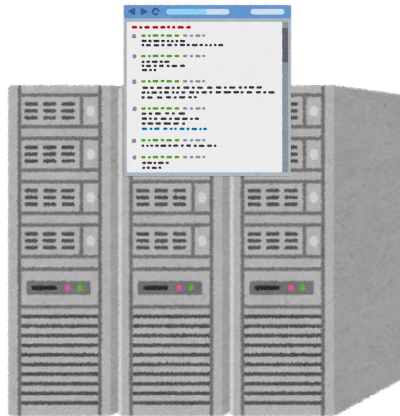
(Web【ウェブ】 WWW / World Wide Web / ワールドワイドウェブ とは - IT用語辞典 e-Words より)

『インターネットで文書などを公開／閲覧するための仕組み』

ブラウザで閲覧



サーバで公開



サーバで公開



インターネット

ブラウザで閲覧



# Webコンテンツの構成要素

- **HTML - HyperText Markup Language**
  - Webページの**構造**を記述
  - 必須(一番重要)
- **CSS - Cascading Style Sheet**
  - Webページの**装飾**を記述
  - なくてもいいし, HTMLに埋め込むこともできる
- **JavaScript**
  - 動的な処理を記述
  - これもHTMLに埋め込むこともできる
- 静的ファイル(画像／音楽／動画／文書 など)



# HTML

ファイルの拡張子は .html (ex: index.html)

<>で囲まれたタグの入れ子によって構造を記述する

```
<html>
  <head>
    <meta charset="UTF-8">
    <title>sample page</title>
  </head>
  <body>
    <h1>Hello, this is sample page.</h1>
    
  </body>
</html>
```



# CSS

ファイルの拡張子は .css (ex: index.css)

それぞれのタグやクラスなどについて装飾(見た目)を記述

```
body {  
    max-width: 500px;           /*最大横幅を500pxに*/  
    margin: auto;               /*marginはよしなに*/  
}  
h1 {  
    color: blue;                /*色を青に*/  
    background-color: black;    /*背景色を黒に*/  
}
```

# JavaScript



ファイルの拡張子は .js (ex: index.js)

HTMLで記述した構造について動きや機能を追加するもの

JAVA言語とは何の関係もない

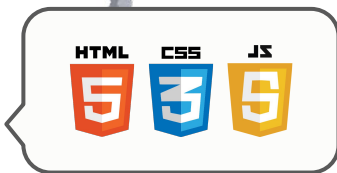
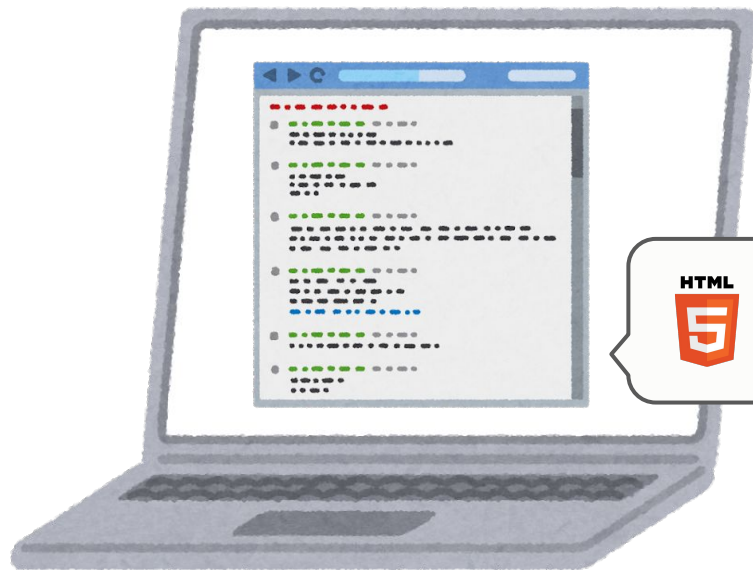
```
var hoge = document.getElementById("hoge");
var fuga = document.getElementById("fuga");

hoge.onclick = function() {
    console.log("clicked hoge");
}

fuga.onclick = function() {
    console.log("clicked fuga");
}
```



ブラウザで  
解釈&描画



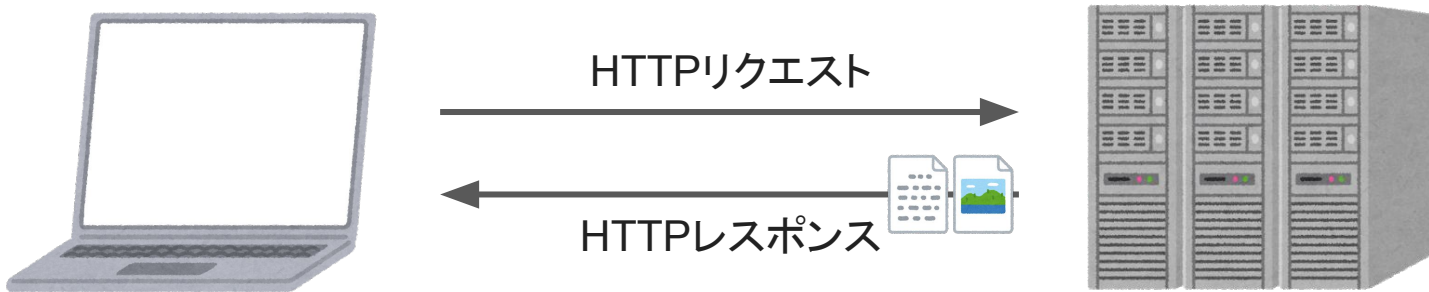
サーバで公開

# Webページのリクエストに使われるプロトコル(L7)

HTTP : HyperText Transfer Protocol

URL ([http://www.google.com/...](http://www.google.com/))とメソッド(GET/POST...)によって  
HTTPサーバに対してリクエストする

HTTPサーバはリクエストに応じてコンテンツを返送する



# IPアドレスとドメイン名

Web上のサービスにアクセスする時, IPアドレスは指定しない(してもいいけど)

ドメイン名の方が人間にわかりやすい

ドメイン名

**www.google.com**

IPアドレス

**216.58.196.228**

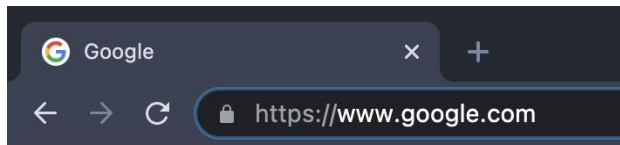
# IPアドレスとドメイン名

Web上のサービスにアクセスする時, IPアドレスは指定しない(してもいいけど)

ドメイン名の方が人間にわかりやすい

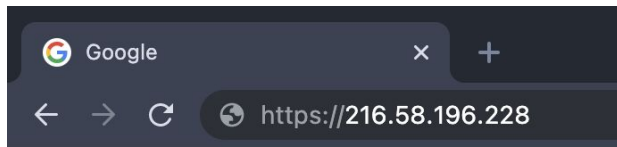
ドメイン名

**www.google.com**



IPアドレス

**216.58.196.228**



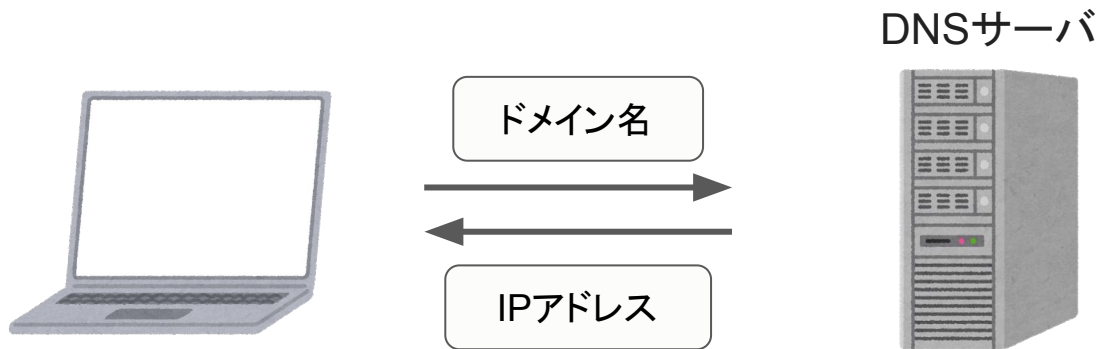
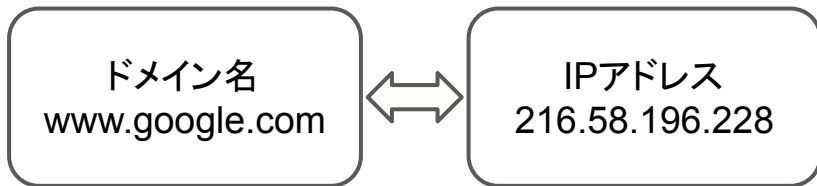
どちらもGoogle検索のトップページにつながる

# ドメイン名とIPアドレスの変換

DNS: Domain Name System

ドメイン名とIPアドレスとを変換する

L7プロトコル



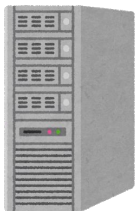
# ドメイン名とIPアドレスの変換

ドメイン名  
www.google.com



IPアドレス  
216.58.196.228

DNSサーバ



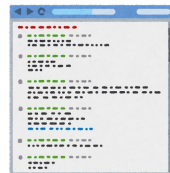
クライアントに入力されたドメイン名は  
DNSサーバでIPアドレスに変換されてリクエストされる

2.ドメイン名

3.IPアドレス

1.ドメイン名

4.IPアドレス



<http://www.example.com:80/hoge/index.html?key=fuga#doc>

http://www.example.com:80/hoge/index.html?key=fuga#doc

スキーマ

ドメイン名

ポート番号

リソース  
へのパス

パラメータ

アンカー

What is a URL? - Learn web development | MDN

[https://developer.mozilla.org/en-US/docs/Learn/Common\\_questions/What\\_is\\_a\\_URL](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/What_is_a_URL)



`http://www.example.com:80/hoge/index.html?key=fuga#doc`



スキーマ

スキーマ (Scheme)

通信に使用するプロトコルを記述する

ブラウザはこれを見て使用するプロトコルを認識する

Webサイトの場合はHTTPかHTTPSが使われる

ftp: や mailto: など也可以使用できる

List of URI schemes - Wikipedia

[https://en.wikipedia.org/wiki/List\\_of\\_URI\\_schemes](https://en.wikipedia.org/wiki/List_of_URI_schemes)

http://**www.example.com**:**80**/hoge/index.html?key=fuga#doc



ドメイン名

ポート番号

ドメイン名 (Domain Name) とポート番号 (Port)

リクエストするWebサーバを指定

ドメイン名は人間が便利のように生み出された. IPアドレスでもいい

ポート番号はサーバにアクセスするためのゲートのようなもの

プロトコルによってデフォルトが決まっている (80 - HTTP, 443 - HTTPS, ...)

デフォルトの場合は省略可

List of TCP and UDP port numbers - Wikipedia

[https://en.wikipedia.org/wiki/List\\_of\\_TCP\\_and\\_UDP\\_port\\_numbers](https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers)

http://www.example.com:80/**hoge/index.html**?key=fuga#doc



リソース  
へのパス

## リソースへのパス

ドメイン名が指し示すWebサーバ上のコンテンツの場所を示す

初期の頃はサーバ上の物理的なファイルの場所を示していたが  
最近では論理的なパスが提供されていることが多い

論理的なパス: Webサーバ側でコンテンツの実体へのルーティングを行う

Webサーバごと独自に好きなパスを設定することができる

http://www.example.com:80/hoge/index.html?key=fuga#doc

パラメータ

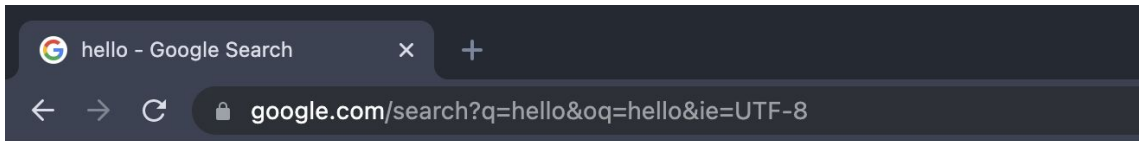
パラメータ(parameters)

Webサーバに渡す追加の情報

?から始まる Key=Valueの形式になっており, &で連結できる

これもWebサーバ内のコンテンツを提供する上での処理に使われる

例) Google検索でhelloと検索



`http://www.example.com:80/hoge/index.html?key=fuga#doc`



アンカー

## アンカー (Anchor)

リソース内部の別の場所 (デフォはトップ) へのアンカー

リソースの中の任意の場所 (小タイトルや再生時間など) を指定できる

この部分はWebサーバに送信されるのではなく  
受け取ったリソースに対して適用される

リソース側にAnchor記述しておく必要はある

<http://www.example.com:80/hoge/index.html?key=fuga#doc>

スキーマ

ドメイン名

ポート番号

リソース  
へのパス

パラメータ

アンカー

<http://www.example.com:80/hoge/index.html?key=fuga#doc>

<http://www.example.com:80/hoge/index.html?key=fuga#doc>

インターネット



DNSサーバ



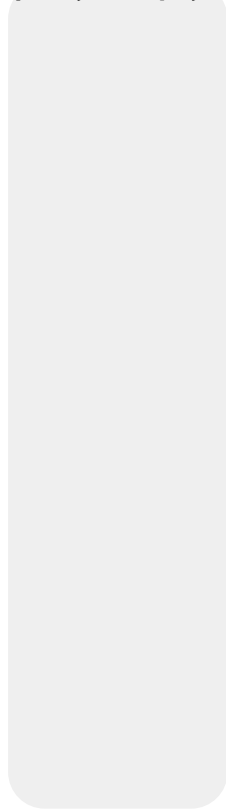
HTTPサーバ



ユーザ



ブラウザ





<http://www.example.com:80/hoge/index.html?key=fuga#doc>



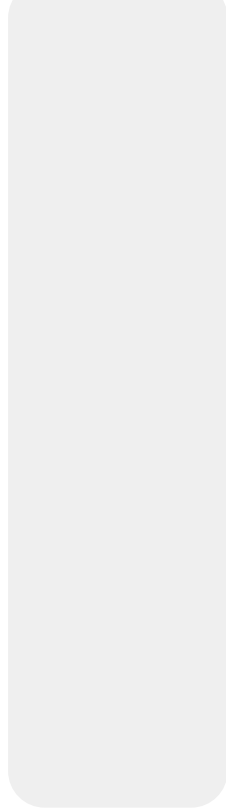
ユーザ

URL (ドメイン名)



ブラウザ

インターネット

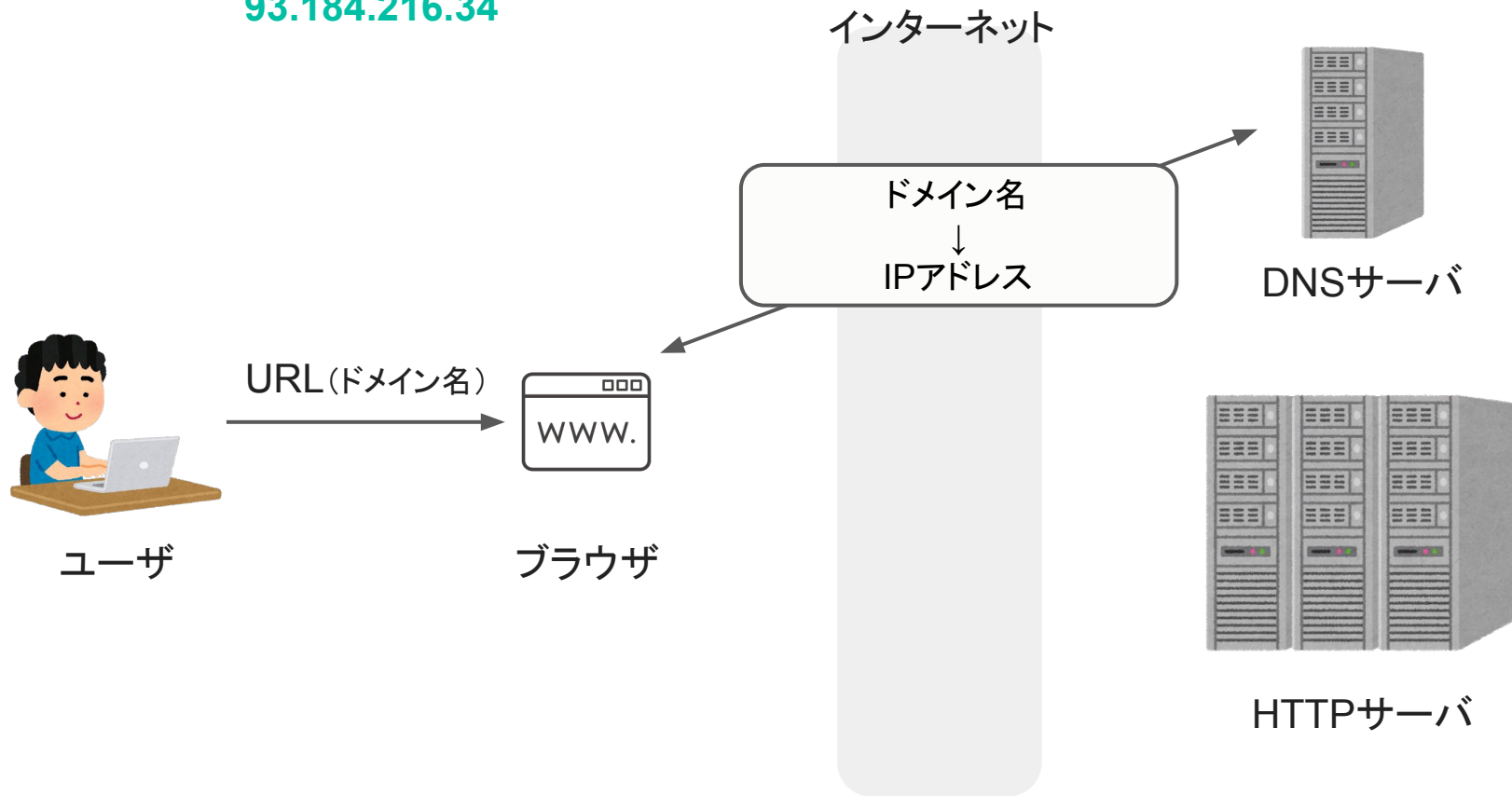


DNSサーバ

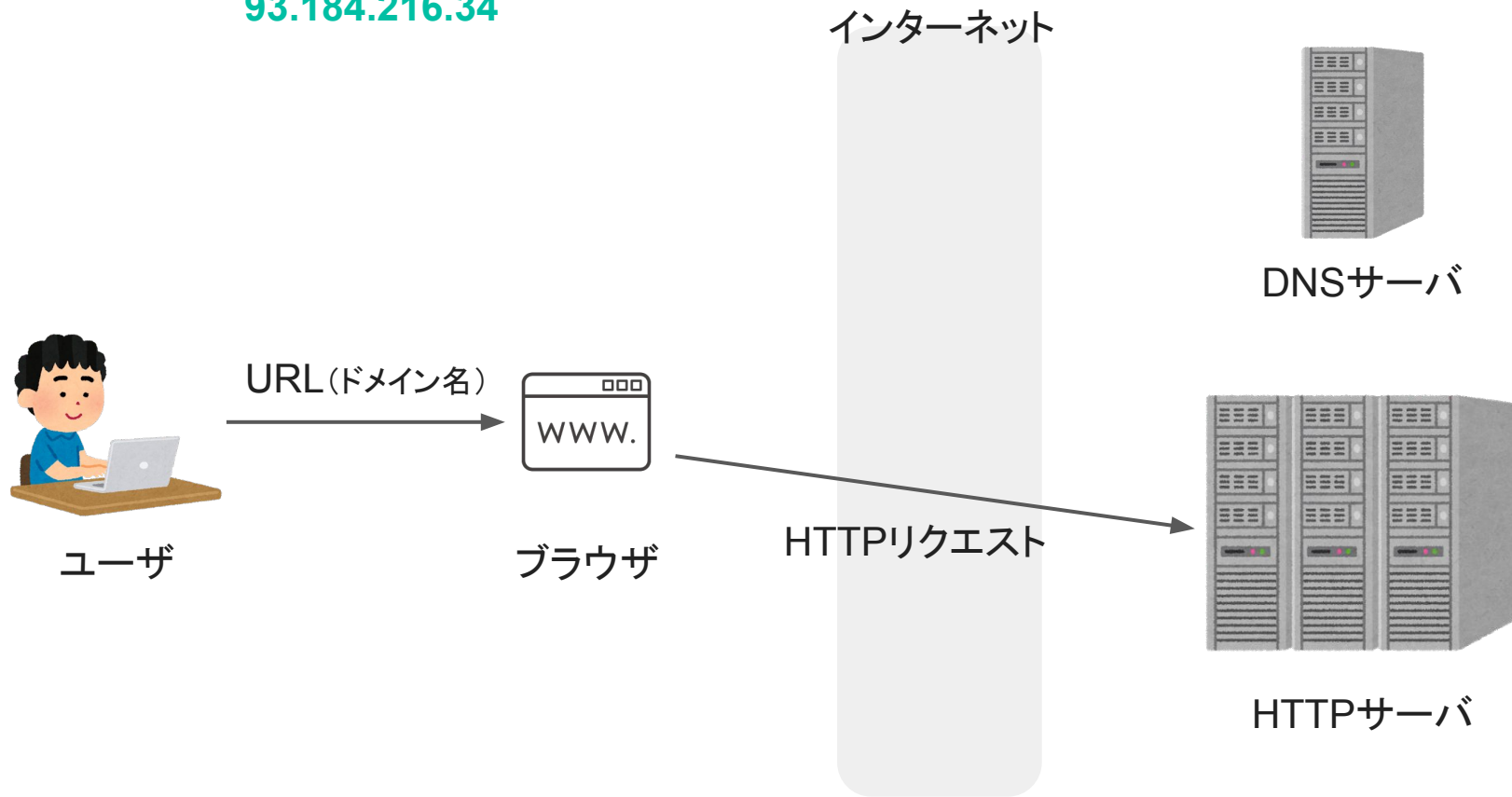


HTTPサーバ

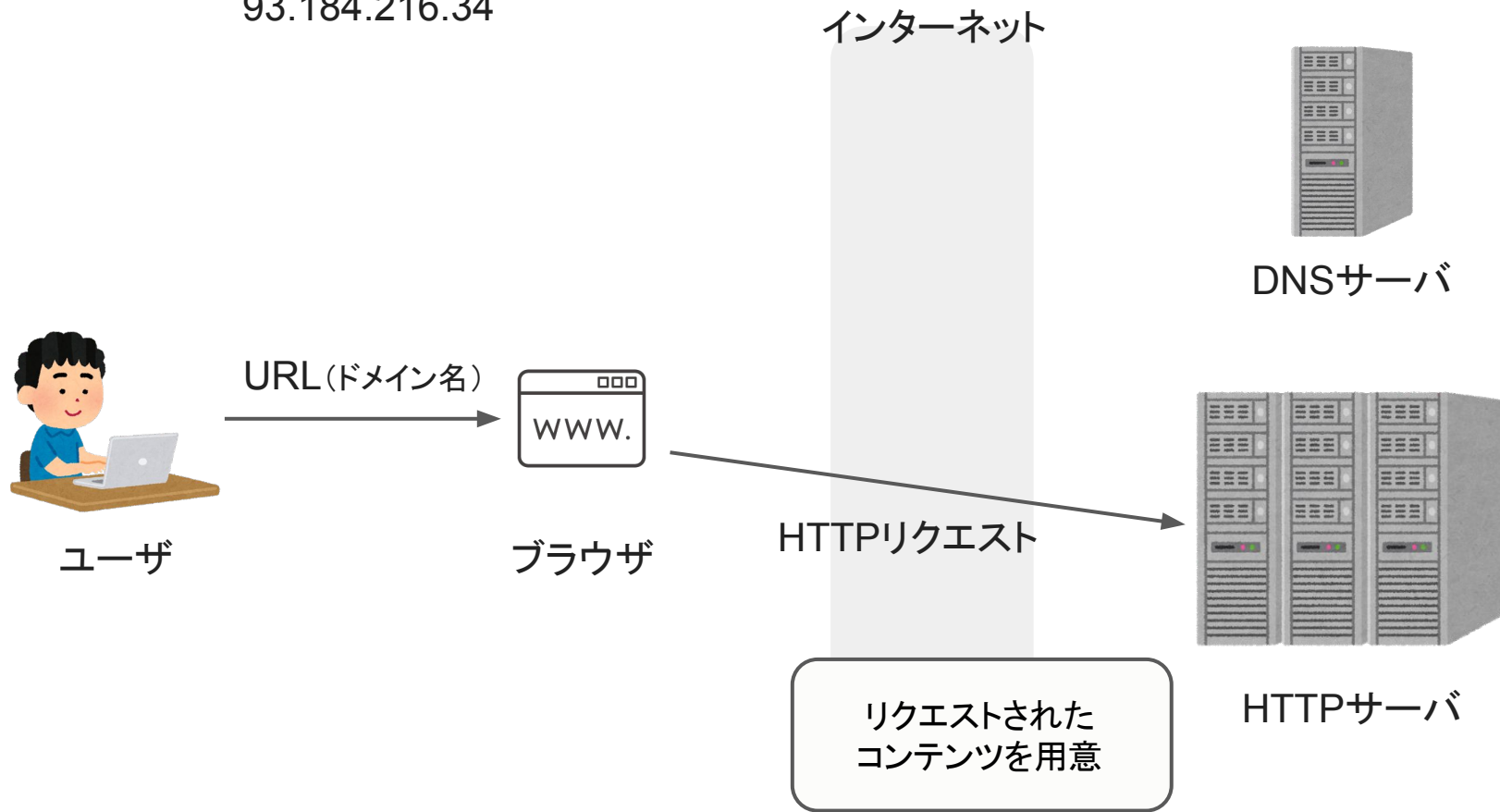
<http://www.example.com:80/hoge/index.html?key=fuga#doc>  
93.184.216.34



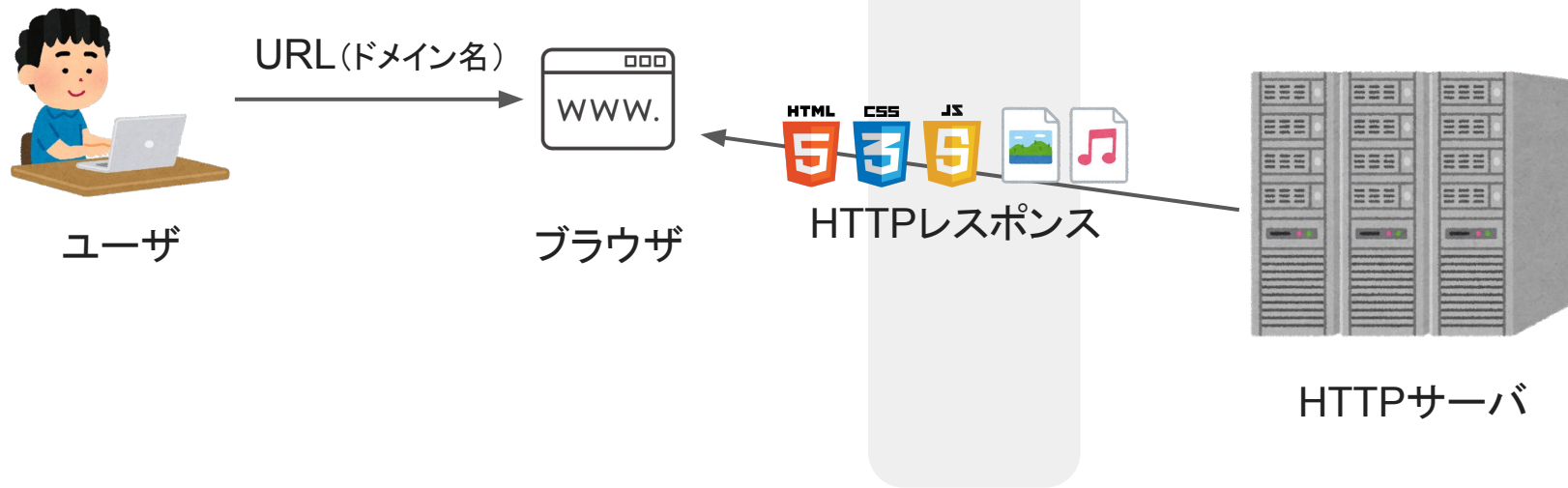
<http://www.example.com:80/hoge/index.html?key=fuga#doc>  
93.184.216.34



http://www.example.com:80/hoge/index.html?key=fuga#doc  
93.184.216.34



<http://www.example.com:80/hoge/index.html?key=fuga#doc>  
93.184.216.34



http://www.example.com:80/hoge/index.html?key=fuga#doc

93.184.216.34

インターネット



DNSサーバ



HTTPサーバ



ユーザ

URL(ドメイン名)



ブラウザ



http://www.example.com:80/hoge/index.html?key=fuga#doc

93.184.216.34

インターネット



DNSサーバ

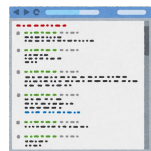


HTTPサーバ



ユーザ

URL (ドメイン名)



ブラウザ

コンテンツを読み込んで  
ページを描画

http://www.example.com:80/hoge/index.html?key=fuga#doc

93.184.216.34

インターネット



DNSサーバ

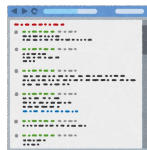


HTTPサーバ



ユーザ

コンテンツを享受



ブラウザ

コンテンツを読み込んで  
ページを描画



# まとめ

Web (World Wide Web) はインターネット上で  
コンテンツを公開／閲覧するための仕組み

WebページはHTML／CSS／JavaScriptで構成される

主なプロトコルはHTTP／HTTPS

URLにはちゃんと全部意味がある