Information Network I

Web 2.0

Youki Kadobayashi
NAIST

They're revolving around the web, after all

- Name any Internet-related buzz:
  - Cloud computing
  - Smartphone
  - Social media
  - ...

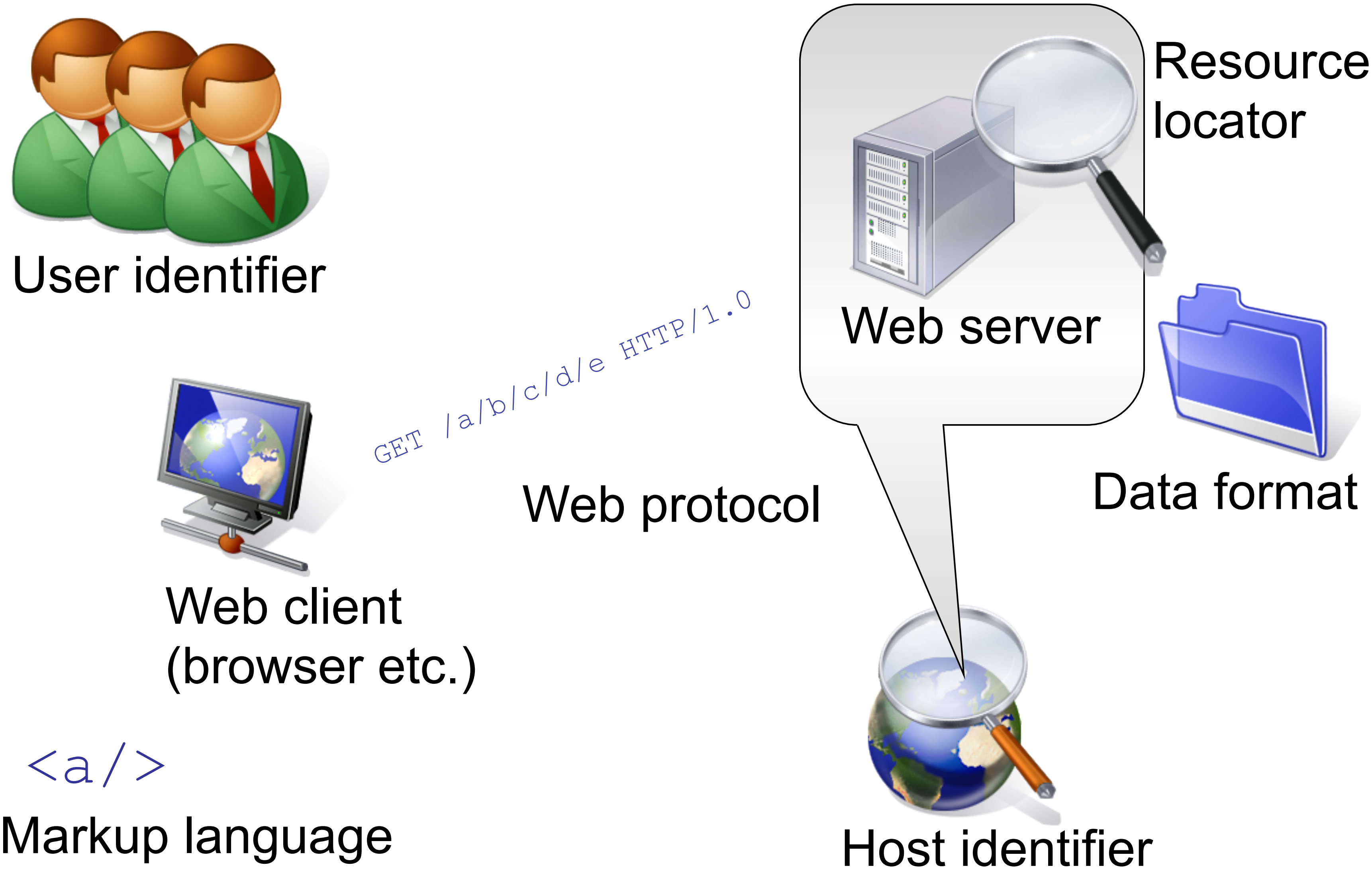
- You'll end up struggling with the web

Outline

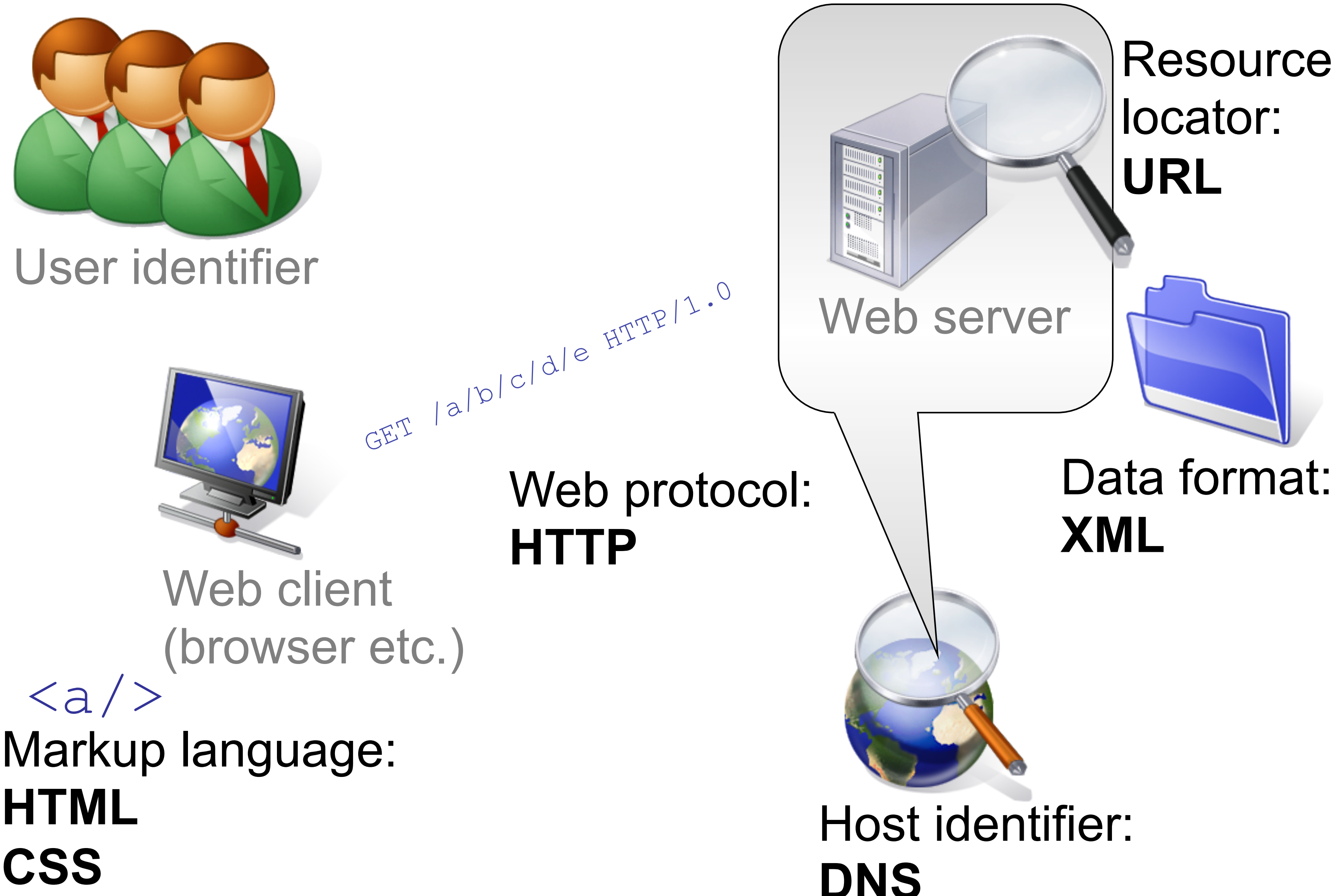
- Web technology stack
- Web technology standards
  - URI, URL
- The web protocol: HTTP
- Web browser, server

- Web 2.0 standards, implications
- Web 2.0 browsers
  - JSON, DOM
- HTTP in Web 2.0
- Mashups

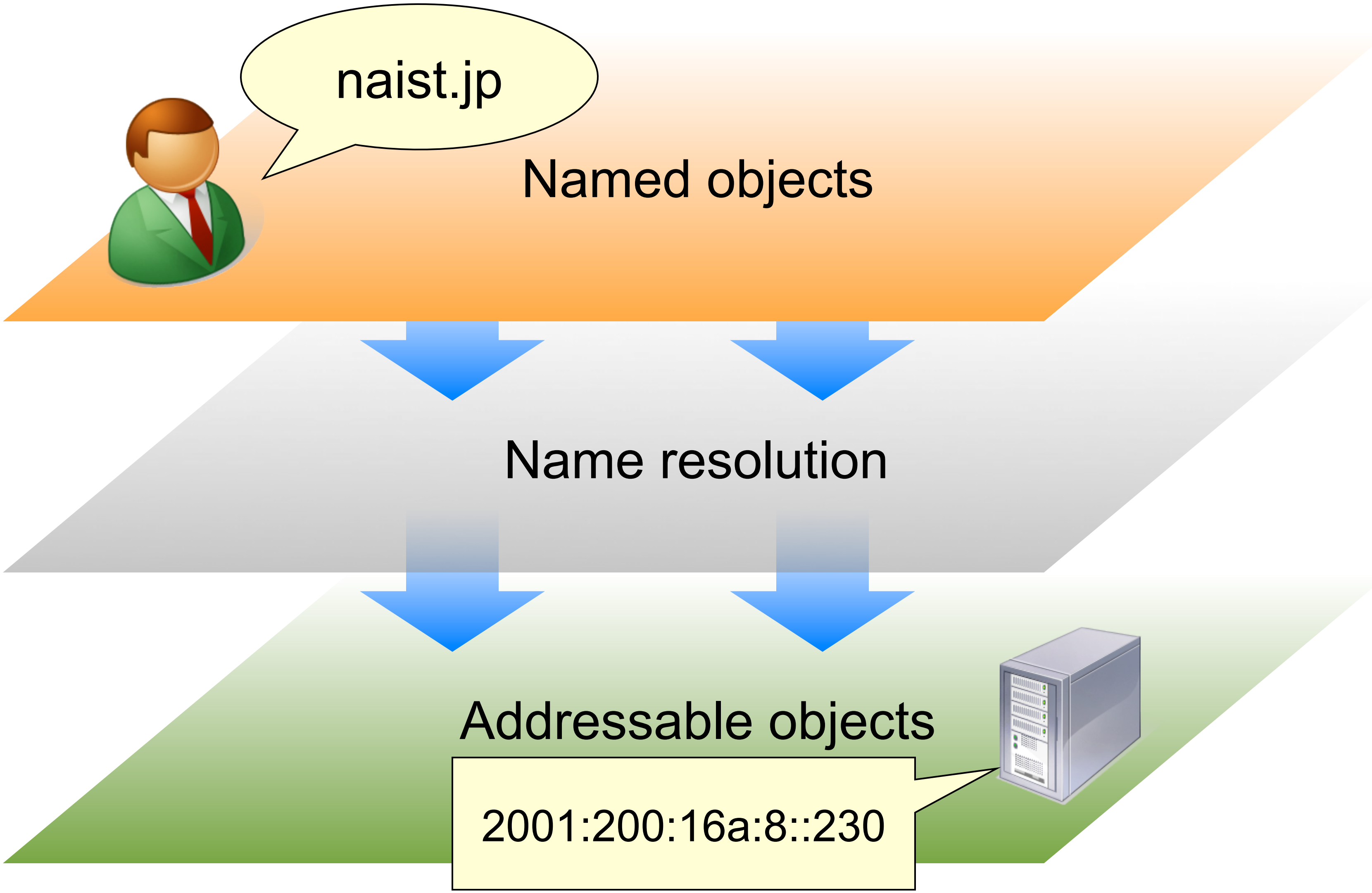
Web technology stack



Web technology standards



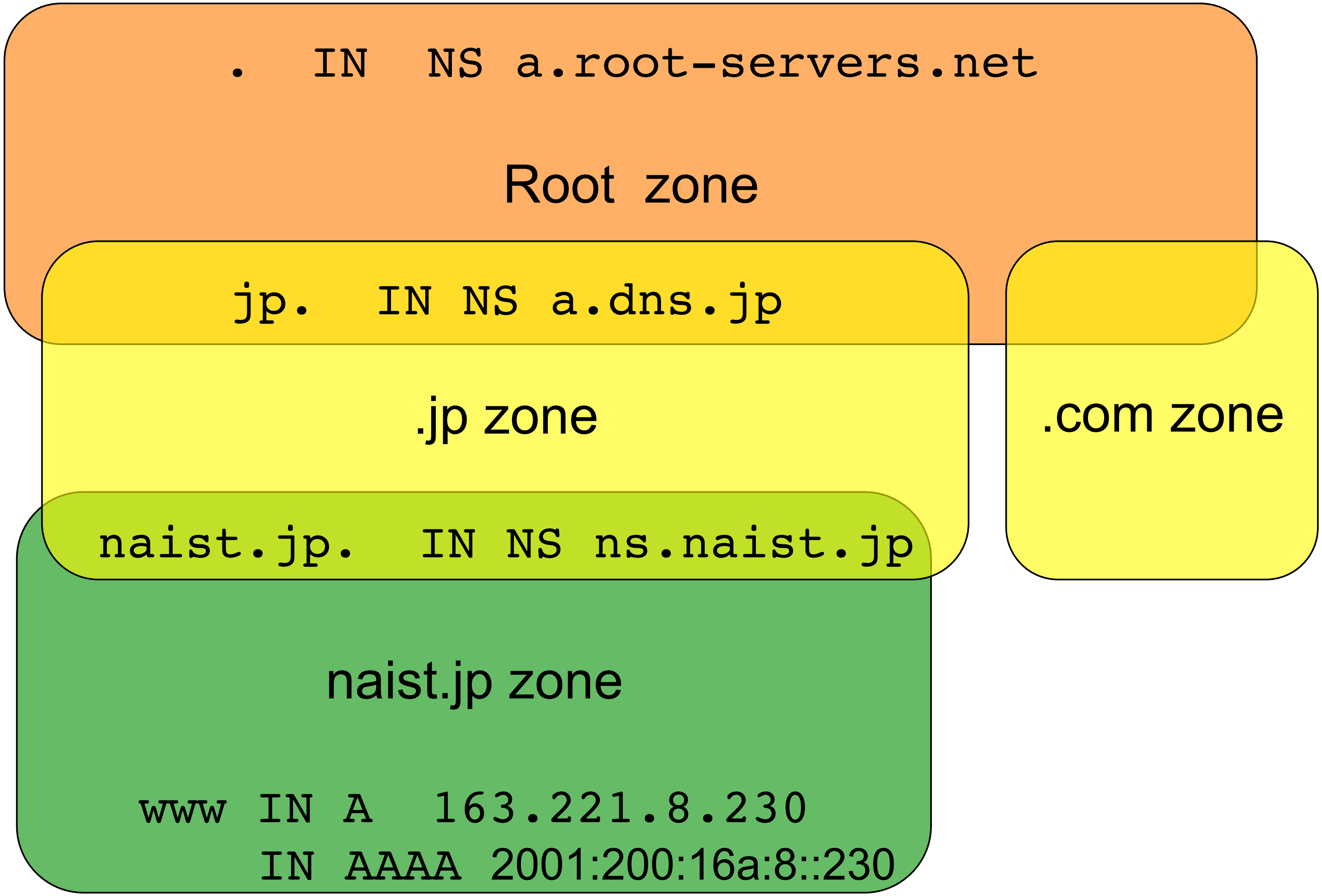
Addressing vs naming: its resolution



DNS: domain name system

- Single global tree for hostname to IP resolution
- Hierarchical delegation
  - Root
    - .
  - gTLD: generic top-level domain
    - .com .net .org .biz .info …
  - ccTLD: country code top-level domain
    - .jp .th .ch .fr …
  - Secondary level domains
    - co.jp, ac.jp, go.jp …

DNS: distributed DB of records



Uniform Resource Locator

- RFC 1738: Uniform Resource Locators
- "locate" resources by providing an abstract identification of the resource location

```
<scheme>:<scheme-specific-part>
   ftp                          File Transfer protocol
   http                         Hypertext Transfer Protocol
   gopher                       The Gopher protocol
   mailto                       Electronic mail address
   file                         Host-specific file names

Common Internet Scheme Syntax
  //<user>:<password>@<host>:<port>/<url-path>
```

Uniform Resource Identifier

- RFC 3986: Uniform Resource Identifier
- URI, URN and URL

```
URI           = scheme ":" hier-part [ "?" query ] [ "#" fragment ]

hier-part     = "//" authority path-abempty
              / path-absolute
              / path-rootless
              / path-empty
```

```
     foo://example.com:8042/over/there?name=ferret#nose
     \_/   \______________/\_________/ \_________/ \__/
      |           |            |            |        |
   scheme     authority       path        query   fragment
      |   _____________________|__
     / \ /                        \
     urn:example:animal:ferret:nose
```

The web protocol: HTTP

- RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1

Request method
URI
Protocol version
Request modifier
Client information
Body content

Status line
protocol version
success/error code
Server information
Entity meta-information
Entity-body content

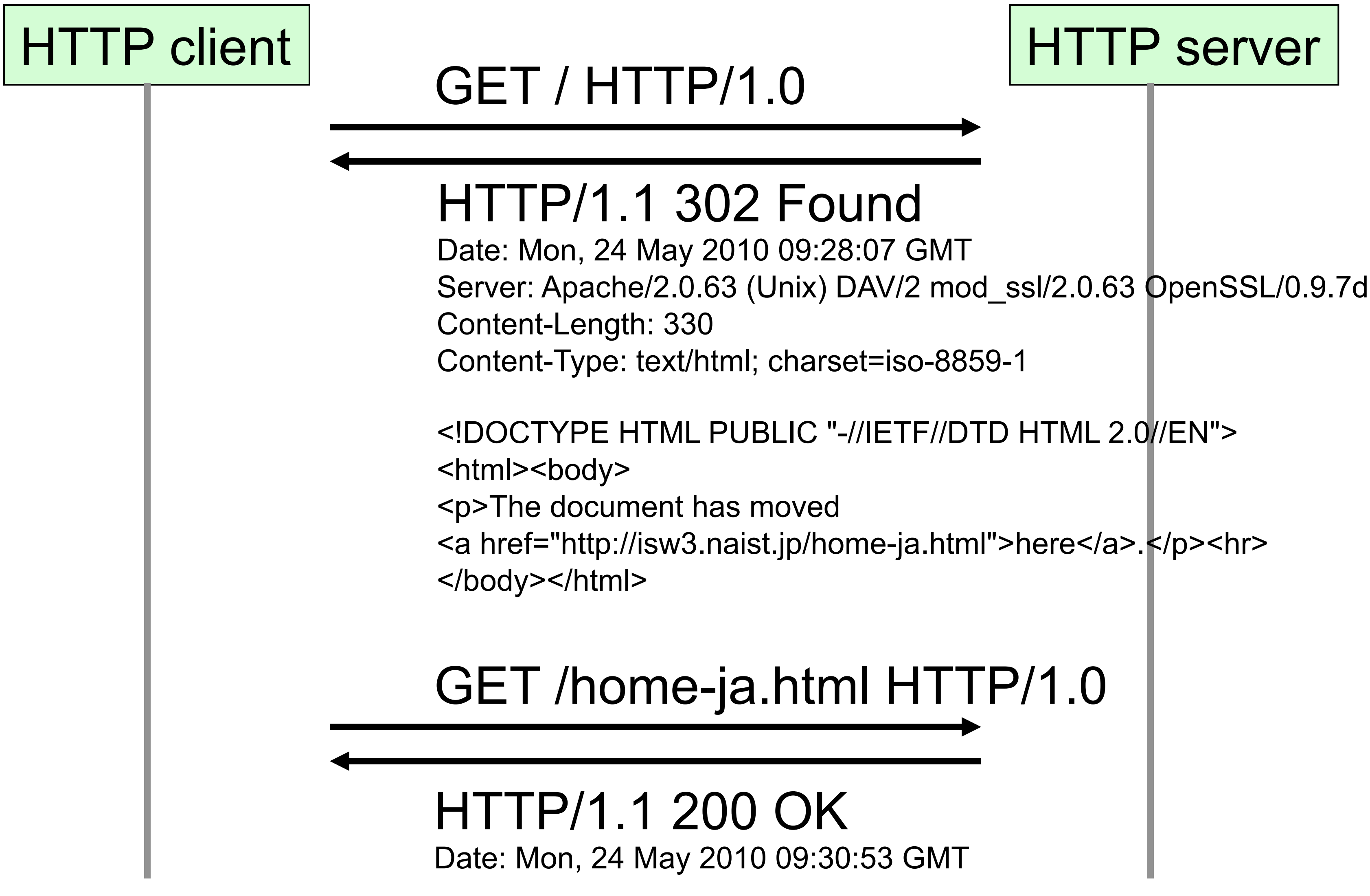
Initiator

Responder

Protocol characteristics of HTTP

- Almost stateless
  - Safe methods – meant to be free from side effects
    - GET, HEAD
  - Idempotent methods – repeated requests w/o side effects
    - GET, HEAD, PUT, DELETE
- State management with cookies -- RFC 6265
- Multimedia
  - Media Types -- RFC 2046
- Content adaptation through Request modifier
  - Multilingual: Accept-Language, Accept-Charset
  - Tailoring: User-Agent

HTTP in action

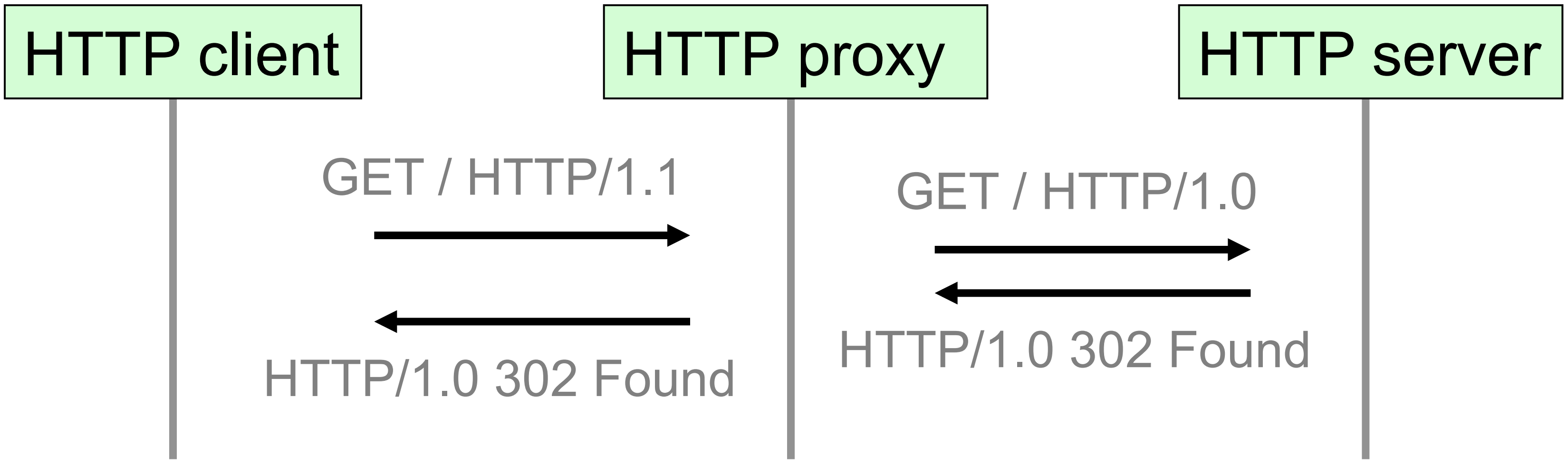


HTTP eases implementation & debugging

Clients:
- Web browser
- telnet
- 1-line program
- Test program

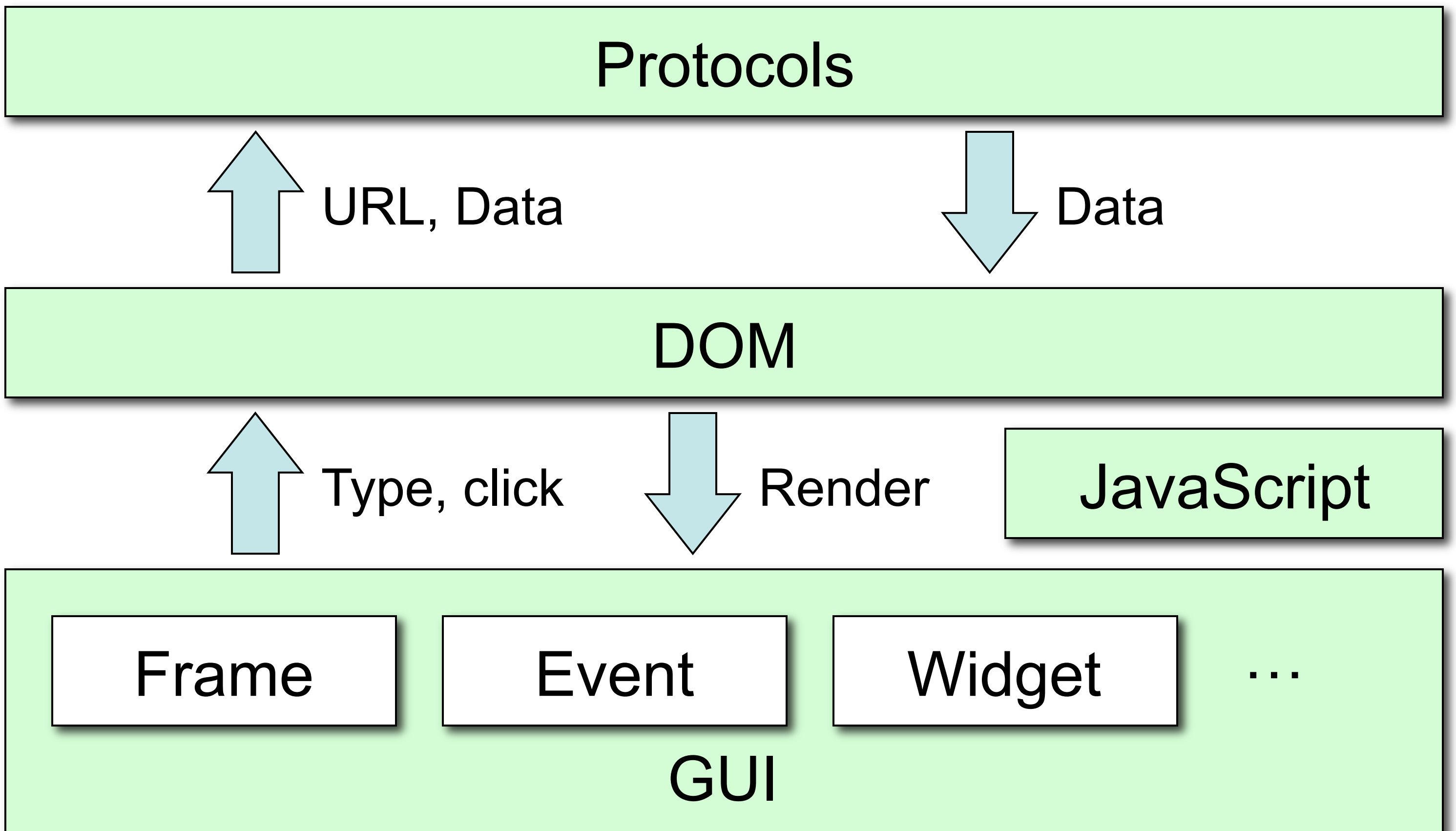
Proxies:
- Logging
- Instrumentation
- Testing
- Analysis

Servers:
- Web server
- 1-line program
- Test program



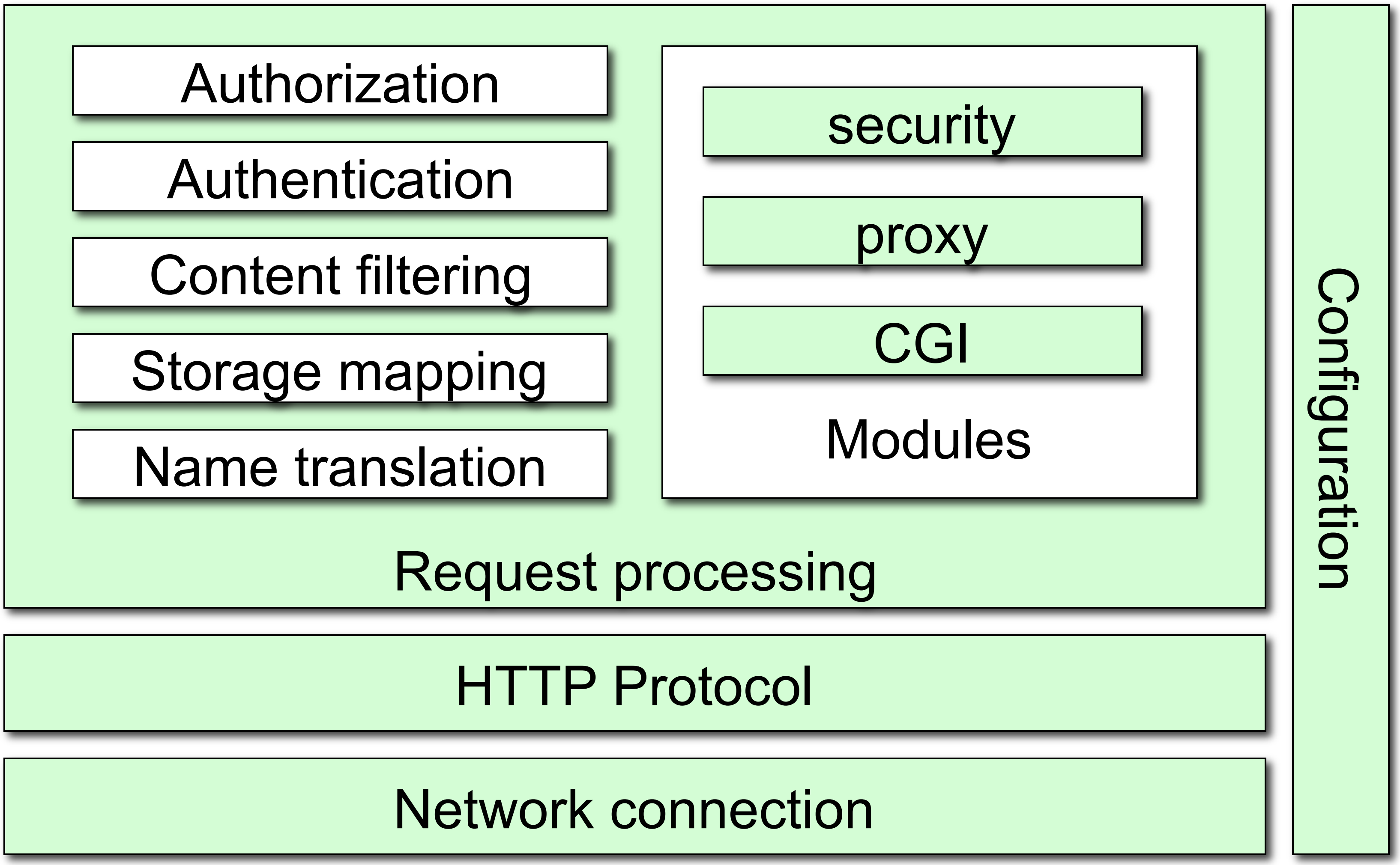
Example: WebScarab

Functional components of Web browser



DOM: Document Object Model

Functional components of Web server



For more detailed study, see for example “The Apache Modeling Project” from Hasso-Plattner Institute

Web 2.0

Web 2.0 technology standards

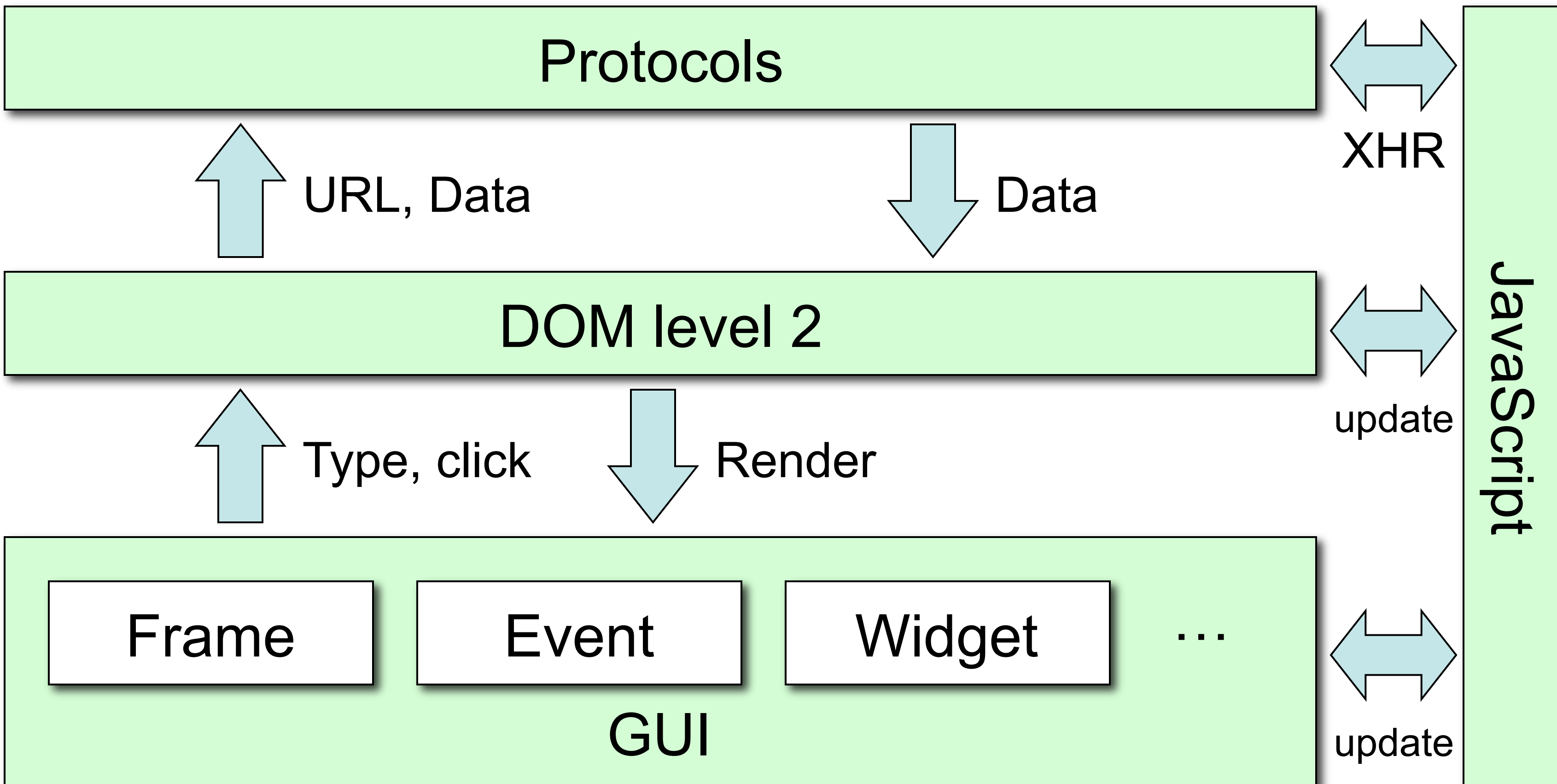
- Web client programming model
  - JavaScript language: ECMA-262
  - Browser as a collection of component objects
    - Document object: W3C DOM level 2
    - Protocol object: W3C XMLHttpRequest
- Data types
  - XML
  - JSON: JavaScript Object Notification
- Collectively called AJAX
  - Asynchronous JavaScript and XML

Web 2.0 technology implications

- Rich web application made possible
  - Progressive update of displayed information
    - DOM level 2
  - Client-server interaction without page transition
    - JavaScript, XMLHttpRequest
  - Structured data
    - XML, JSON

- Emergence of Software as a Service
  - SaaS

Functional components of Web 2.0 browser



For more detailed study, see for example “Rapid Application Development with Mozilla” by Nigel McFarlane

JSON

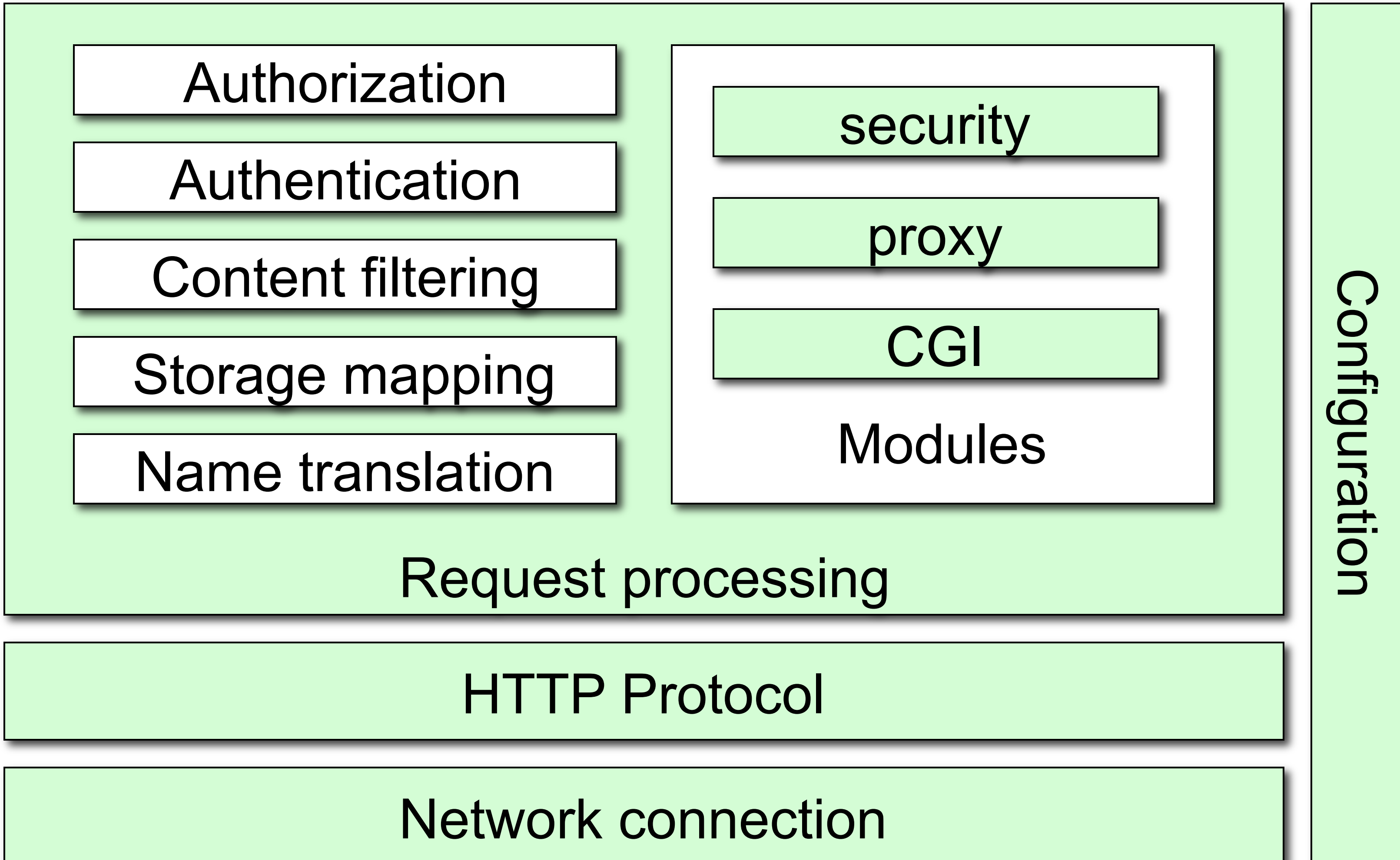
- RFC 4627: The application/json Media Type for JavaScript Object Notation
- A text format for the serialization of structured data
- Four primitive types: string, number, boolean, null
- Two structured types: object, array

```
JSON object:
   {
      "Image": {
          "Width":  800,
          "Height": 600,
          "Title":  "View from 15th Floor",
          "Thumbnail": {
              "Url":    "http://bit.ly/",
              "Height": 125,
              "Width":  "100"
          },
          "IDs": [116, 943, 234, 38793]
      }
   }
```

JSON array

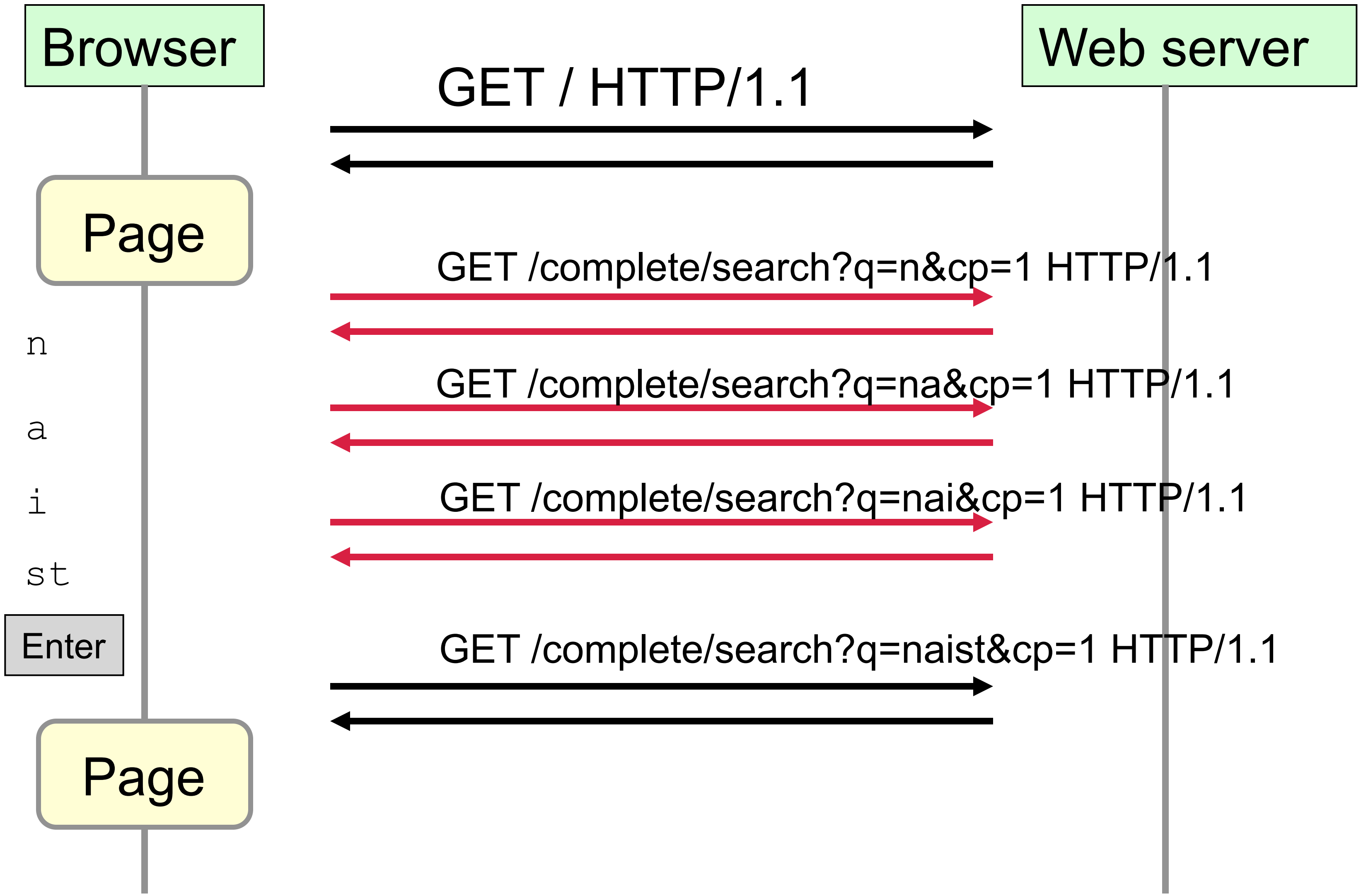
DOM example by Firebug

Functional components of Web 2.0 server



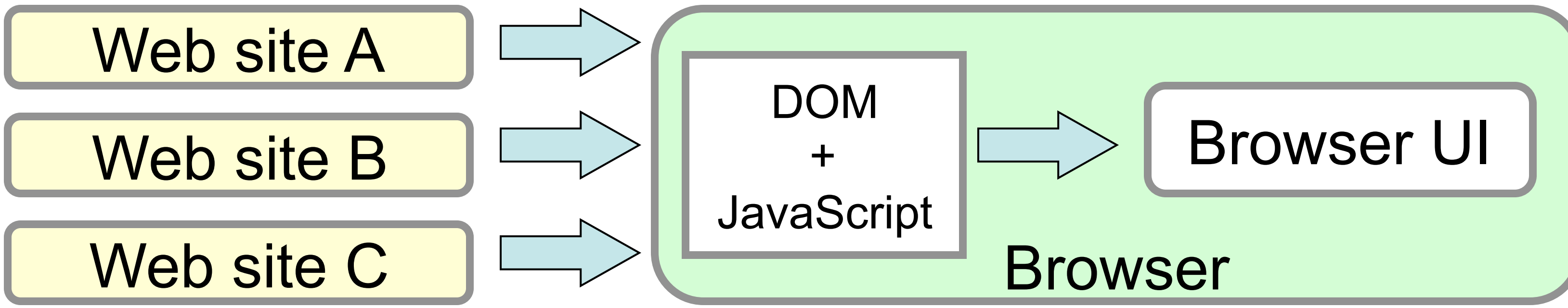
No difference at server side

HTTP in action, Web 2.0

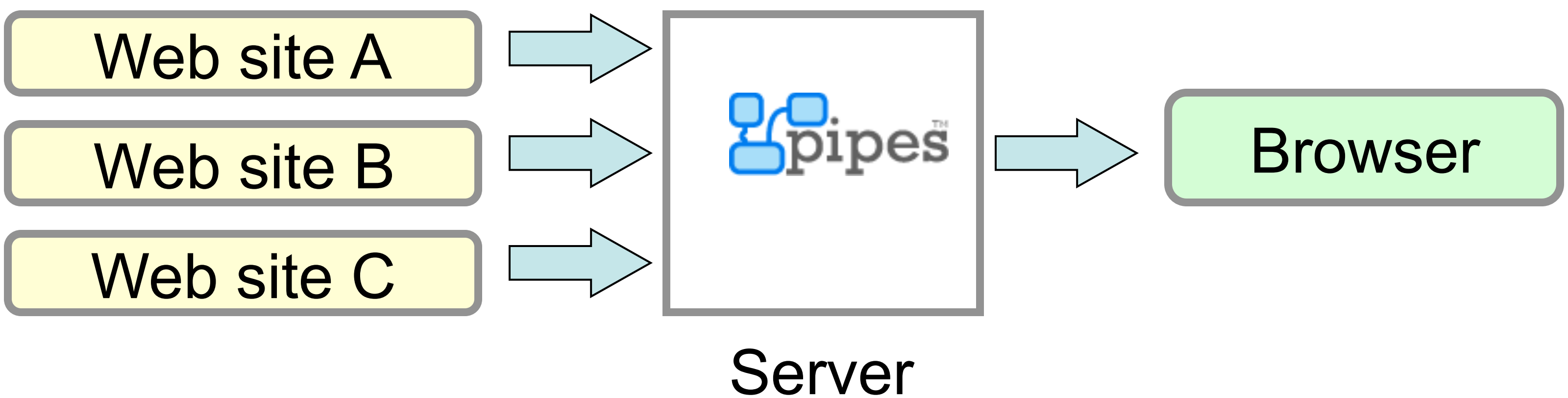


Mashups

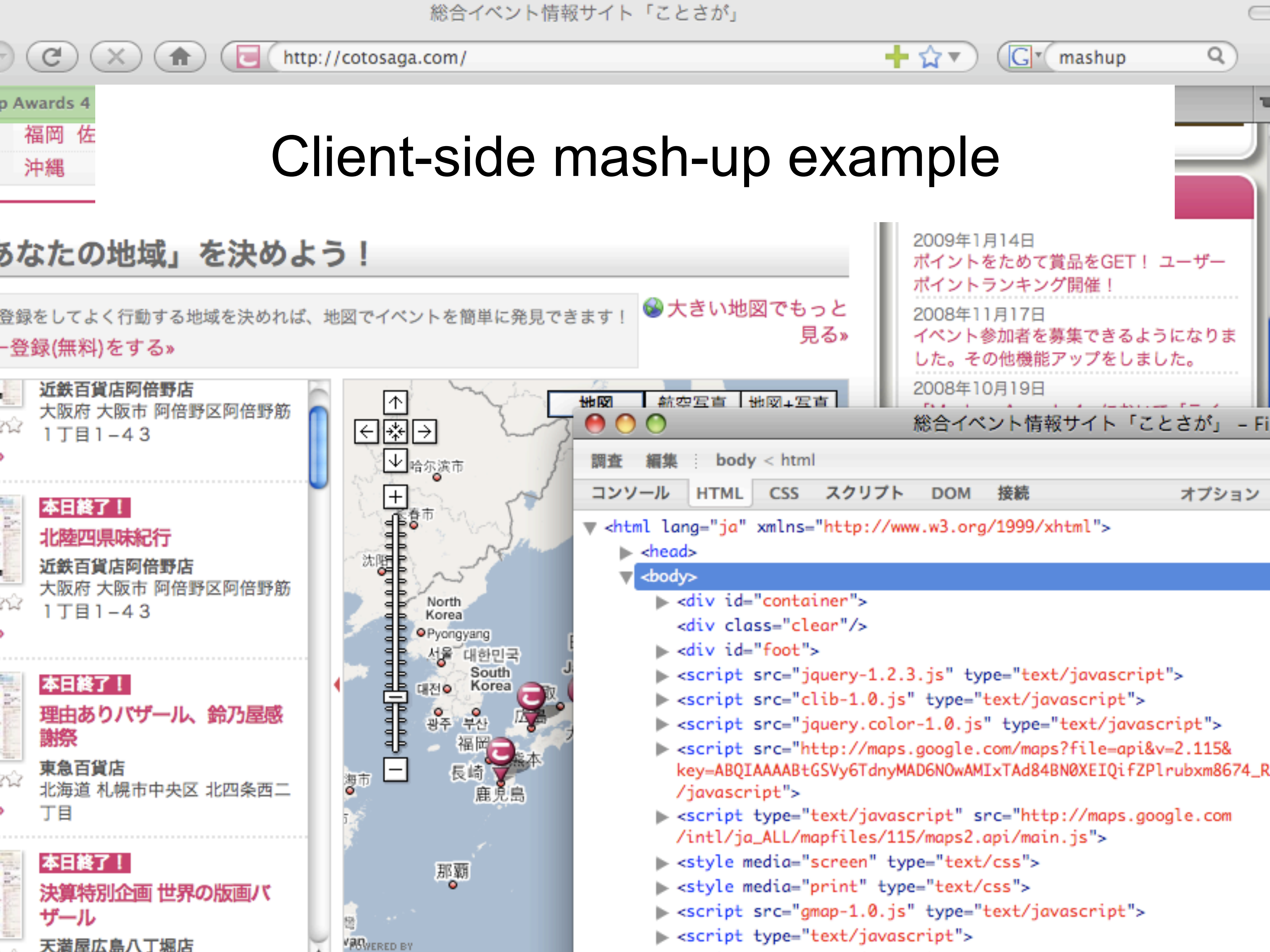
- Client-side mashup



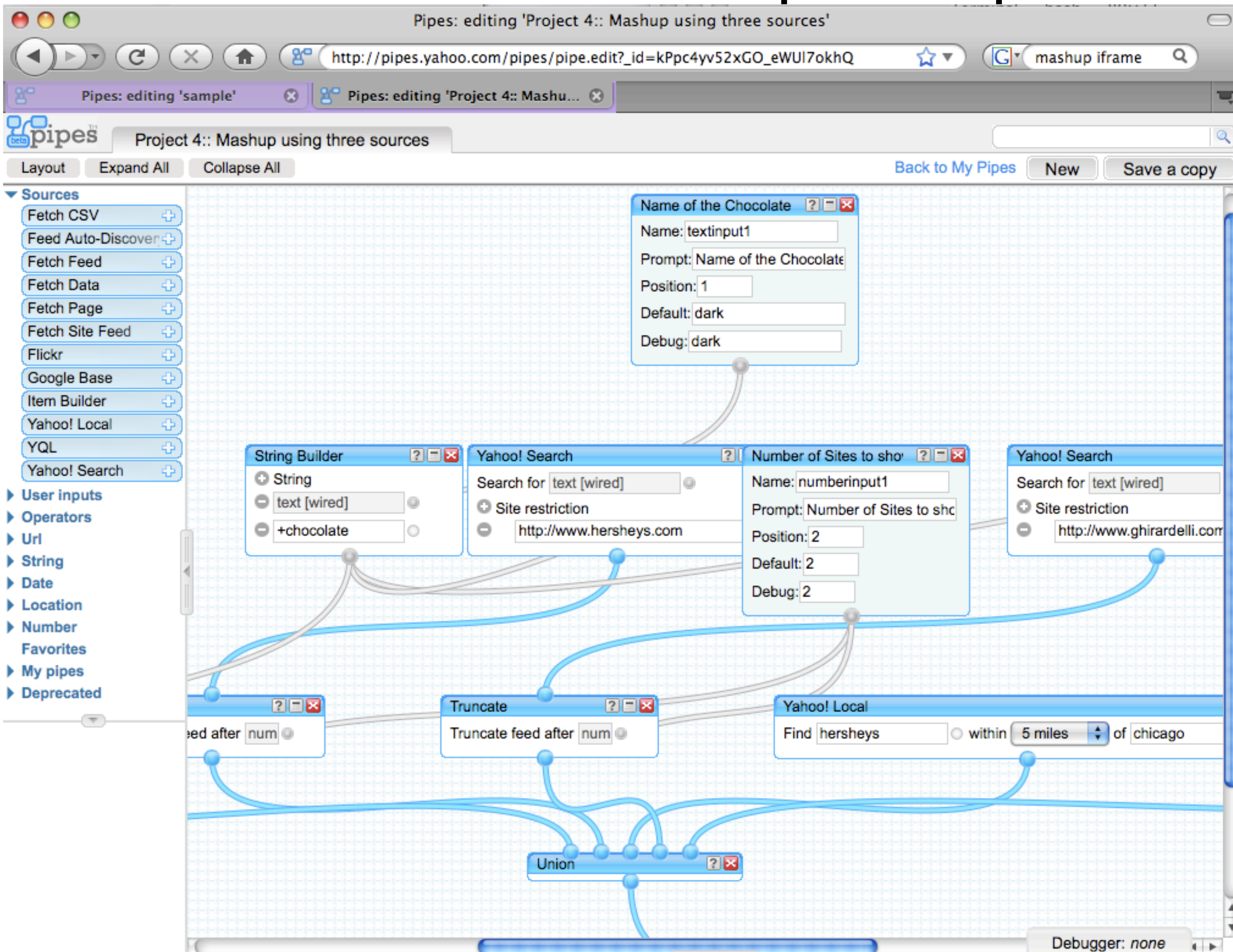
- Server-side mashup



Client-side mash-up example



Server-side mash-up example



Web 2.0 technology standards

User identifier:
OpenID

GET /a/b/c/d/e HTTP/1.0

Web client:
JavaScript
DOM
XHR

<a/>

Markup:
HTML
CSS

Web protocol:
HTTP 1.1

Web server

Resource locator:
URL

Data format:
XML
JSON

Host identifier:
DNS

Evolving web standards

- IETF
- W3C

- WHATWG

