Services

Web Engineering

Prof. D. König

SIR TIM BERNERS-LEE

United Kingdom – 2016

CITATION

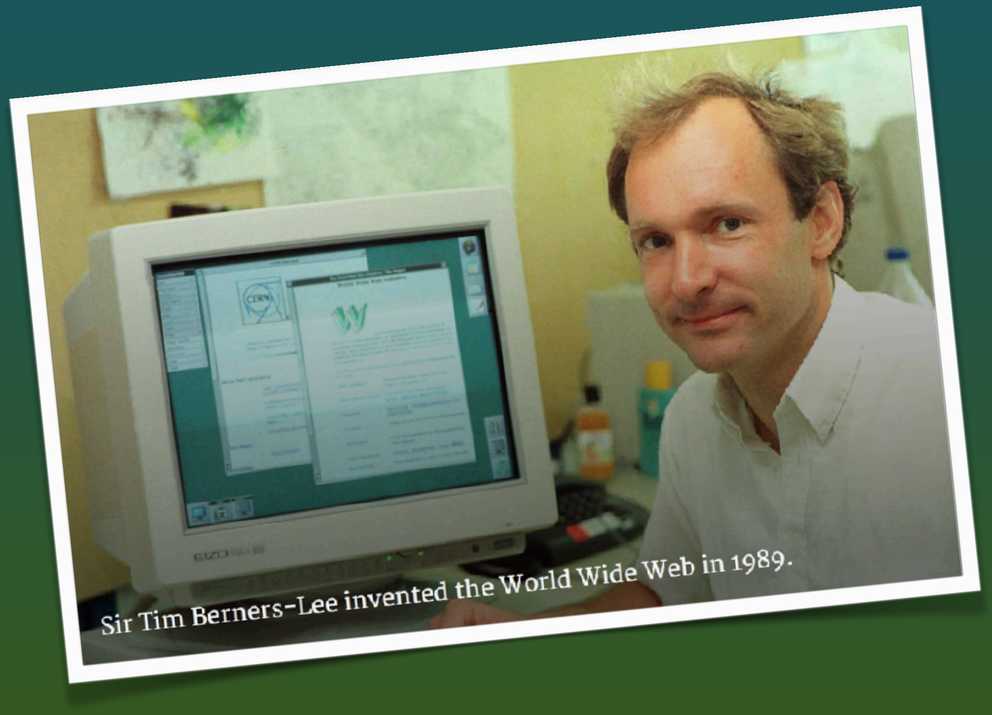
For inventing the World Wide Web, the first web browser, and the fundamental protocols and algorithms allowing the Web to scale.

ACM DL AUTHOR PROFILE

RESEARCH SUBJECTS

Web Resilience

The Web has never gone down
since its inception
in 1989.



Web Resilience

BE RESTRICTIVE
WHAT YOU EXPOSE

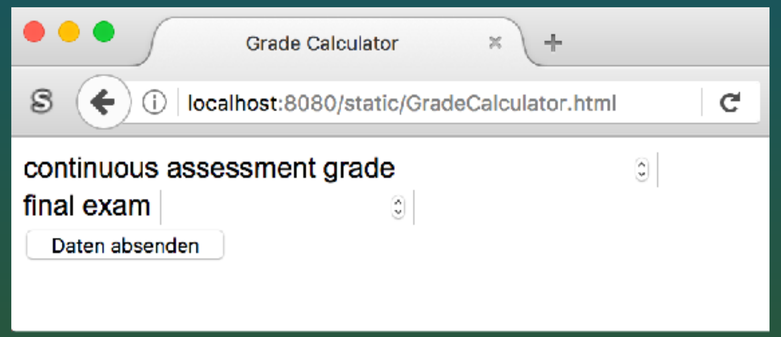
BE TOLERANT
WHAT YOU CONSUME

Three Tier Web Applications

Client

Server

DB



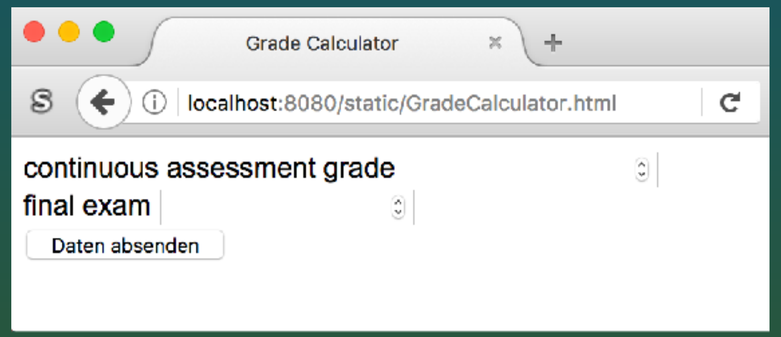
persist

Server with Service

Client

Server

Service



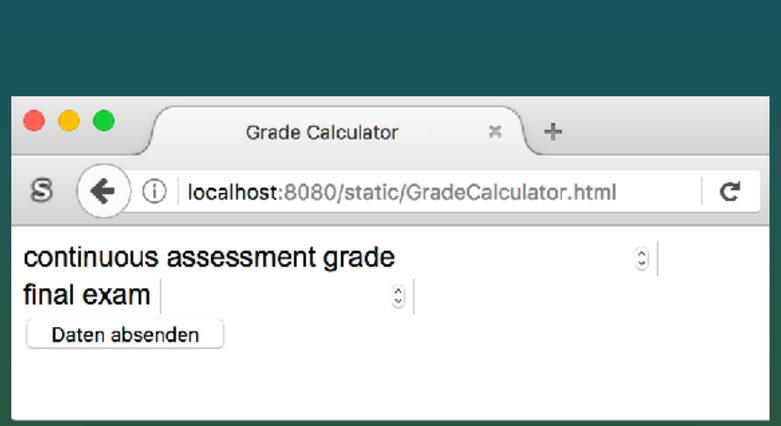
req/resp

"Smart UI" with Service

Client

Server

Service



req/resp

JavaScript Service APIs

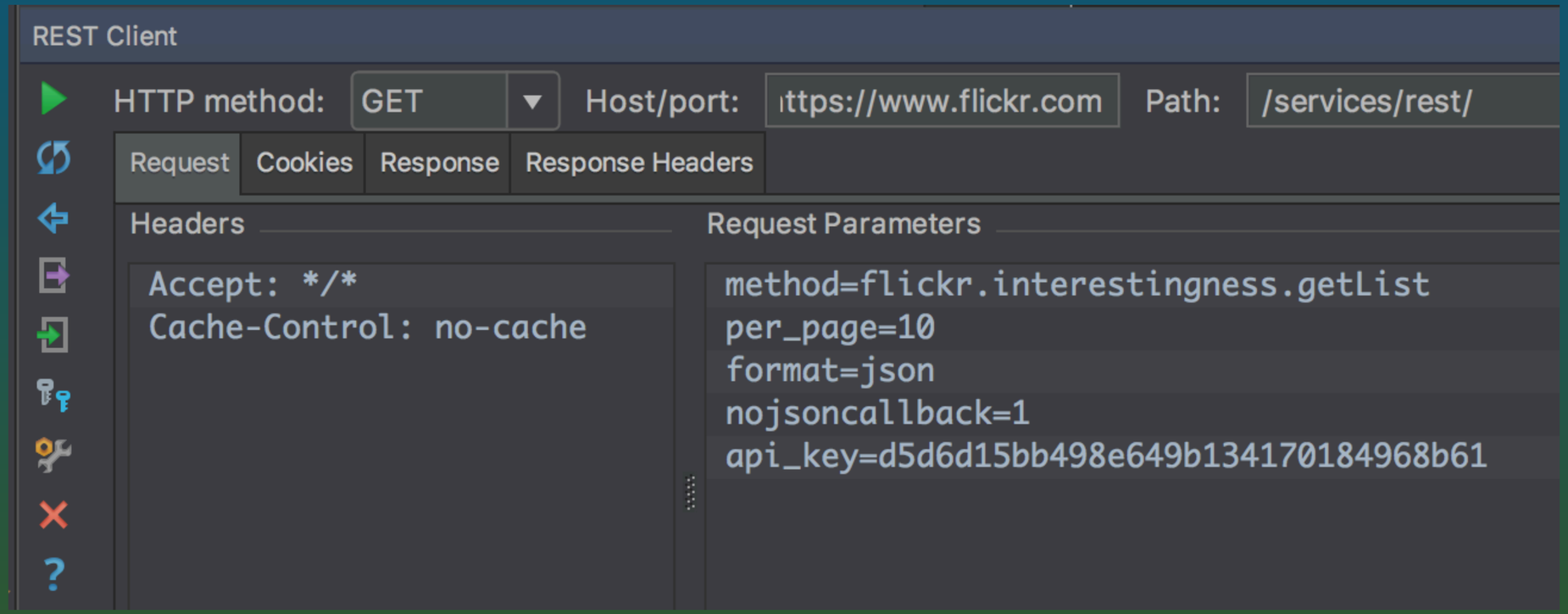
https://www.flickr.com/services/api/request.rest.html

Get a developer key (use mine carefully)

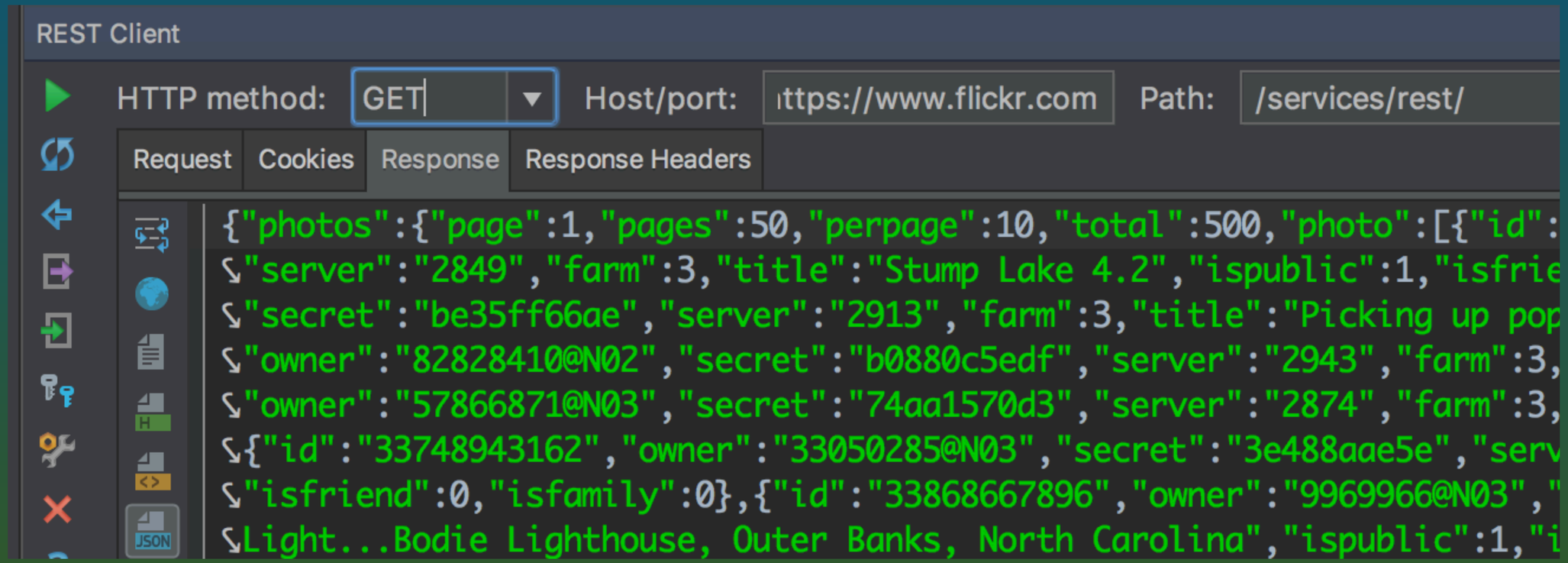
Use key to request a secret

Use secret to request a resource

Integrated REST Client



Result


REST Client
HTTP method:
GET
Host/port:
ttps://www.flickr.com
Path:
/services/rest/
Request
Cookies
Response
Response Headers
{"photos":{"page":1,"pages":50,"perpage":10,"total":500,"photo":[{"id":
"server":"2849","farm":3,"title":"Stump Lake 4.2","ispublic":1,"isfrie
"secret":"be35ff66ae","server":"2913","farm":3,"title":"Picking up pop
"owner":"82828410@N02","secret":"b0880c5edf","server":"2943","farm":3,
"owner":"57866871@N03","secret":"74aa1570d3","server":"2874","farm":3,
{"id":"33748943162","owner":"33050285@N03","secret":"3e488aae5e","serv
"isfriend":0,"isfamily":0},{"id":"33868667896","owner":"9969966@N03","
Light...Bodie Lighthouse, Outer Banks, North Carolina","ispublic":1,"i

Result


```
{"photos": {
 "page":1,"pages":50,"perpage":10,"total":500,
   "photo"[ {
     "id"     :"33090986453",
     "owner" :"133058989@N03",
     "secret":"5ce24df9f7",
     "server":"2849",
     "title" :"Stump Lake 4.2"
}, …
```

Fetch Resource URL Pattern

http://static.flickr.com/
server-id/id_secret_m.jpg

URL

Uniform Resource Locator

protocol, host, port, path, query, fragment

scheme:[//[user:password@]host[:port]][/]path[?query][#fragment]

HTTP(s)

Hyper Text
Transfer Protocol

Very simple

Line/delimiter based

```
# First request
GET /quote?symb=YHOO HTTP/1.1
Host: www.example.org

HTTP/1.1 200 OK
Content-Type: text/plain;charset=UTF-8

15.96

# Second request 10 minutes later
GET /quote?symb=YHOO HTTP/1.1
Host: www.example.org

HTTP/1.1 200 OK
Content-Type: text/plain;charset=UTF-8

16.10
```

HTTP Verbs

Verb	Change	Use
GET	No	Single or collective read, may be cached
PUT	Yes	Modify resource in place
POST	Yes	Create new resource
DELETE	Yes	Delete resource
HEAD	No	
OPTIONS	No	

REST

Representational State Transfer

- Resource addressing by URL
- Uniform operations by HTTP verbs
- Self-containment (no conversational state)
- Choice of format (XML, JSON, …)

REST Issues

Easy to get started - easy to get lost.

Type safety, Versioning, Documentation

Addressing Operations as Resources

REST Alternatives

SOAP (WSDL),

CORBA (IDL),

EJB, *-RPC, RMI,

local services

REST in Grails

http://docs.grails.org/latest/guide/webServices.html

Note the usage of @Resource in domain classes and "respond" instead of "render"

in the controller actions.