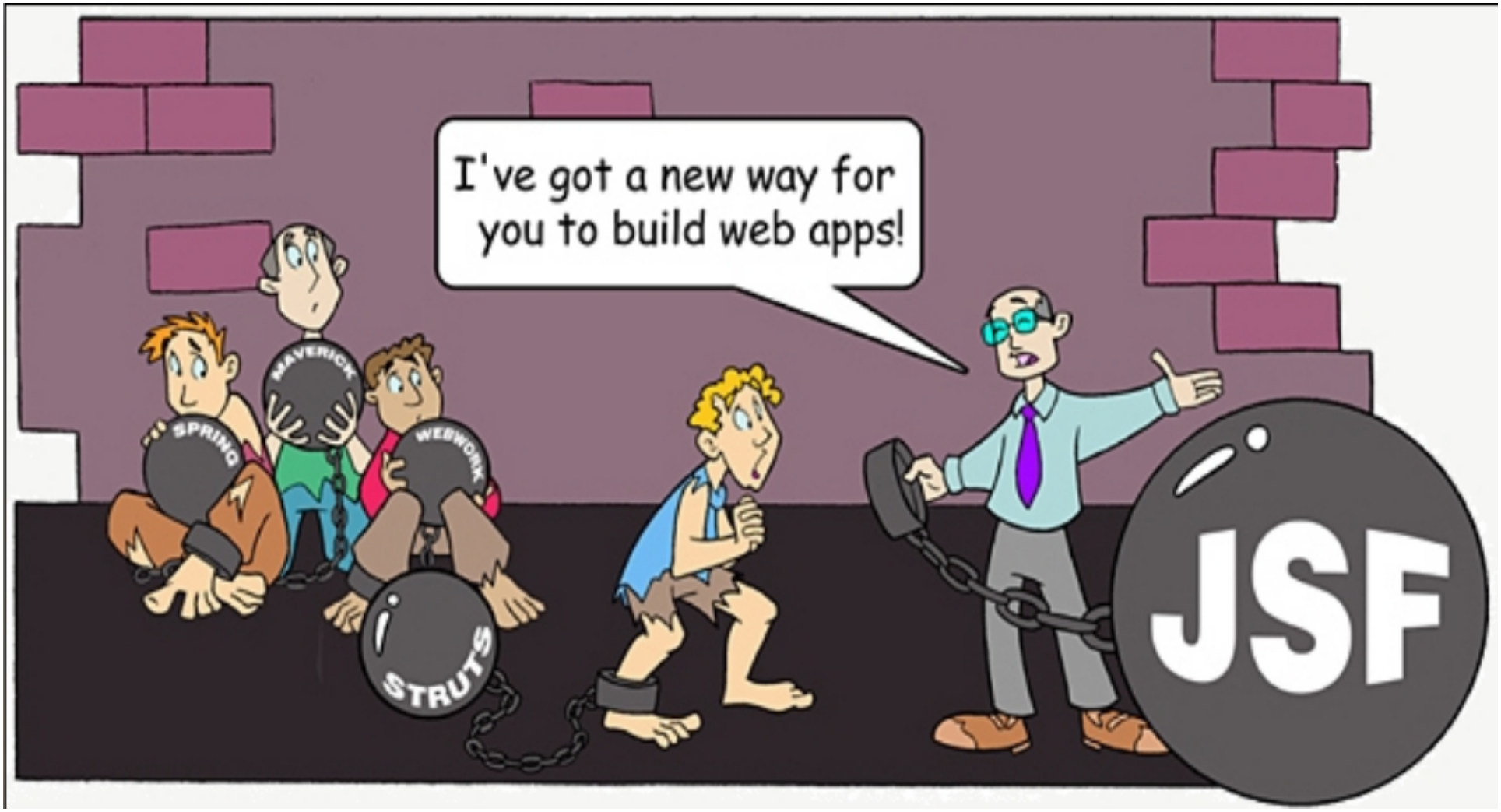
tREST

framework pre webové aplikácie a služby

Peter Rybár

Situácia v korporátnej sfére

- Dominuje technológia



Situácia na Webe

- Dominuje architektúra **REST** – štýl softvérovej architektúry pre distribuované hypermediálne systémy
- REST Webové aplikácie
  - Google
    - http://code.google.com/webtoolkit/
  - Amazon
    - http://developer.amazonwebservices.com/
  - Yahoo!
    - http://developer.yahoo.com/

CENTAUR


tREST

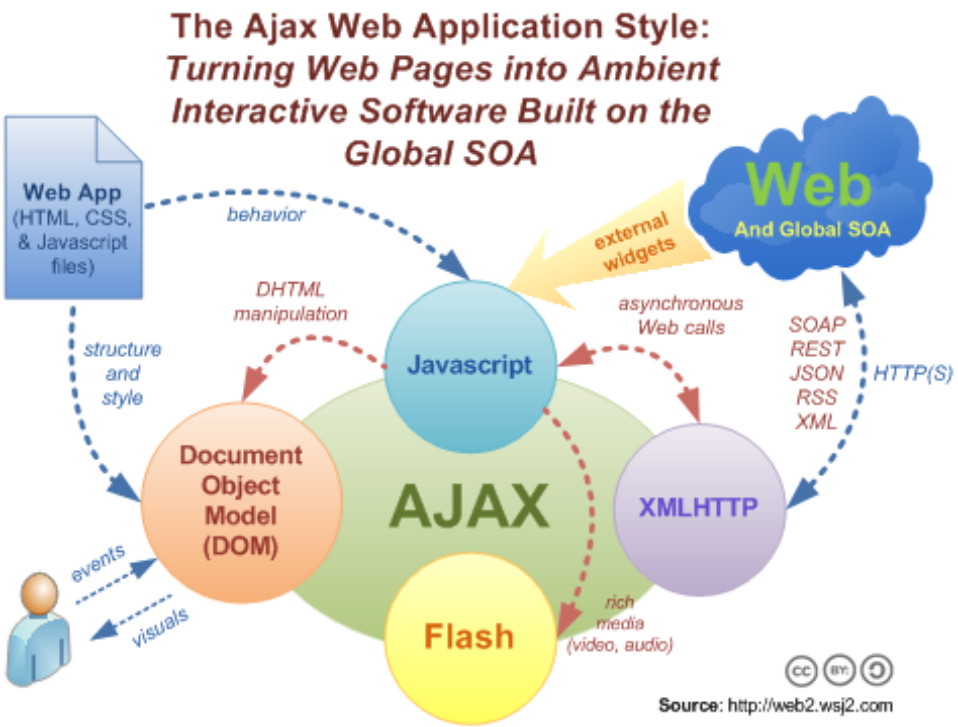
tREST – čo to je?

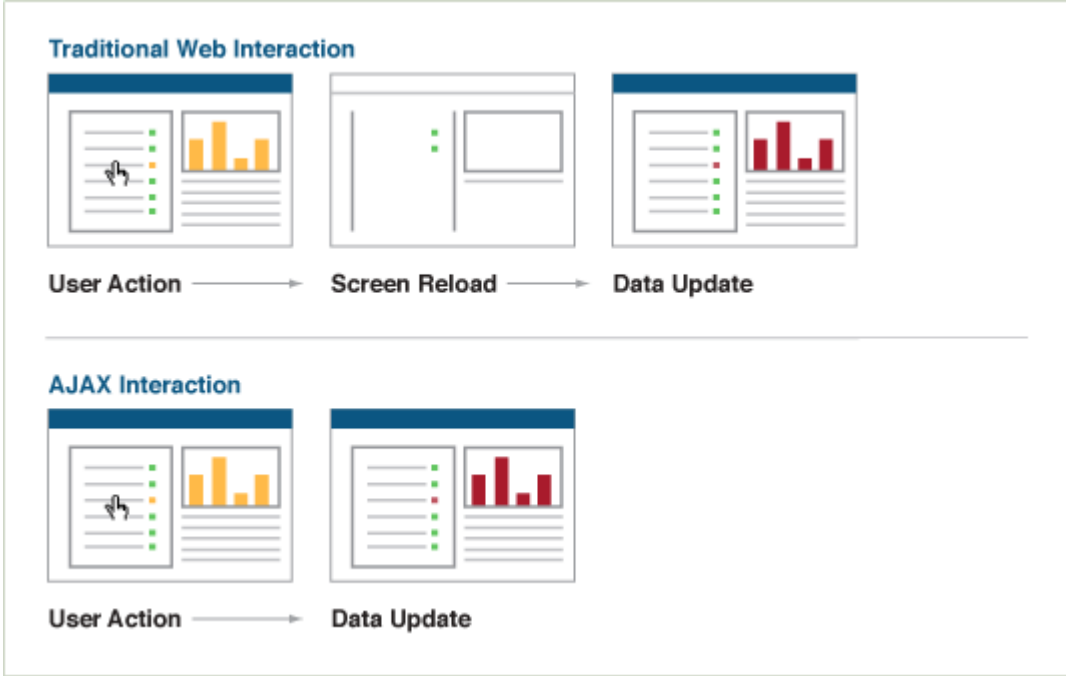
- Web framework – softvérovy framework
- Navrhnutý na vývoj
  - Web aplikácií
  - Webových služieb
- Určený pre vývoj architektonickým štýlom **REST** (**Re**presentational **S**tate **T**ransfer)
- Kladie dôraz na
  - Jednoduchosť vývoja
  - Efektivitu vývoja

CENTAUR

tREST – čo to je?

- Jednotný spôsob ako vytvárať
  - Tradičné Web aplikácie
  - AJAX Web aplikácie





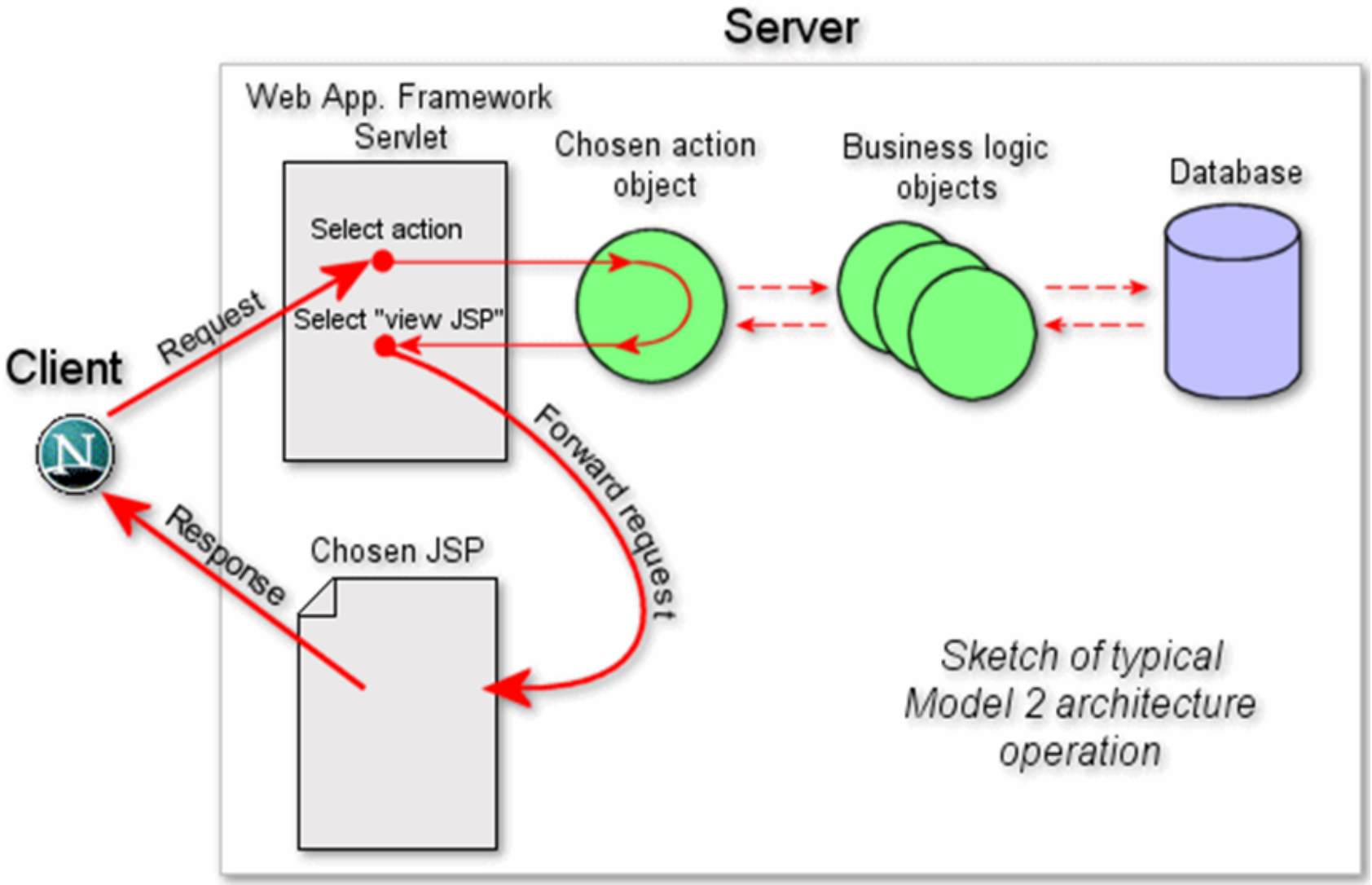
CENTAUR

tREST – o čo v ňom ide

- Modularita – modulárna architektúra
- Extenzibilita – jednoduchá možnosť integrácie iných technológií
- Vývoj REST Web aplikácií a služieb – AJAX, RIA (**R**ich **I**nternet **A**pplications)
- Oddelenie vývoja
  - prezentačnej vrstvy
  - serverového backendu
  - aplikačnej logiky

CENTAUR

tREST – o čo v ňom ide



CENTAUR

tREST – o čo v ňom ide

- Minimálna doba nábehu vývojára do vývoja vo frameworku – čas rádovo v hodinách
- Horizontálny vývoj aplikácií – bežný vývojár ovláda iba svoju doménu, nemusí ovládať všetky technológie naprieč aplikáciou, čoho dôsledkom je vyššia kvalitu kódu, efektivita
- Voľná väzba – nezávislosť frameworku od veľkého počtu technológií, stabilita
- Platformová nezávislosť – Java 5 a viac

CENTAUR

tREST – viac do hĺbky

- Framework pozostáva z dvoch komponentov
  - Serverový komponent
    - Kontróler pre servletový kontajner so sadou rozšírení
  - Klientský komponent
    - Javascript knižnica nezávislá na serverovej komponente – je možné ju použiť v kombinácii s ľubovolnou technológiou na strane servera.

CENTAUR

tREST – Serverový komponent

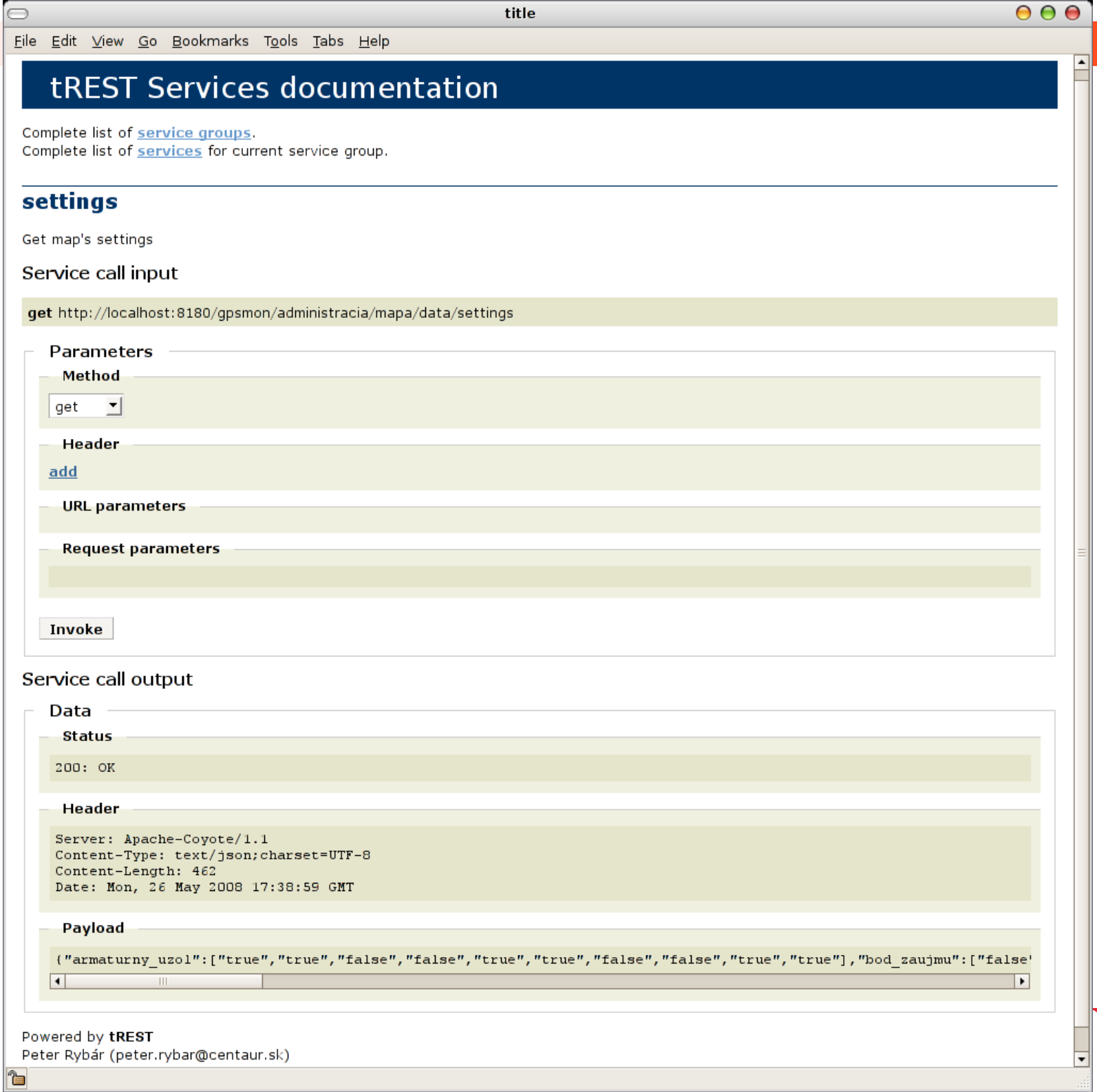
- Implementuje vlastnosti
  - Automatická konverzia Webových vstupov do natívnych dátových typov
  - Automatická serializácia výstupov z natívnych dátových typov
  - Dynamické Mapovanie kontrólerov
  - Mapovanie URL parametrov
  - Jednoduché rozhranie pre implementáciu vlastných rozšírení (extensions)
  - Deklaratívny prístup pre pri použití rozšírení realizovaný Java anotáciami

CENTAUR

tREST – Serverový komponent

- Podpora internacionalizácie (i18n)
- Role based user access management
- Podpora pre lineárny workflow
- Dokumentácia Webových služieb pomocou Java anotácií priamo v kóde
- Webové rozhranie pre prístup k dokumentácii Webových služieb na základe Java reflexie
- Možnosť testovania implementovaných konrólerov, Webových služieb, s využitím vstavaného Webového rozhrania

CENTAUR


title
File Edit View Go Bookmarks Tools Tabs Help
tREST Services documentation
Complete list of service groups.
Complete list of services for current service group.
settings
Get map's settings
Service call input
get http://localhost:8180/gpsmon/administracia/mapa/data/settings
Parameters
Method
get
Header
add
URL parameters
Request parameters
Invoke
Service call output
Data
Status
200: OK
Header
Server: Apache-Coyote/1.1
Content-Type: text/json;charset=UTF-8
Content-Length: 462
Date: Mon, 26 May 2008 17:38:59 GMT
Payload
{"armaturny_uzol":["true","true","false","false","true","true","false","false","true","true"],"bod_zaujmu":["false
Powered by tREST
Peter Rybár (peter.rybar@centaur.sk)

tREST – Serverový komponent

- Oddelenie prezentačnej vrstvy od aplikačnej logiky pri zachovaní slabej väzby vzhľadom k použitej prezentačnej technológii (JSP, FreeMarker, SiteMesh, …)
- Zabudovaná podpora interceptingu pre podporu aspektovo orientovaného programovania
- Sada rozšírení modulárne doplňujúcich funkcionalitu kontrólera/filtra
- Rozšíriteľná sada validátorov validujúcich syntax a sémantiku s podporou konverzie do natívnych dátových typov

tREST – Klientský komponent

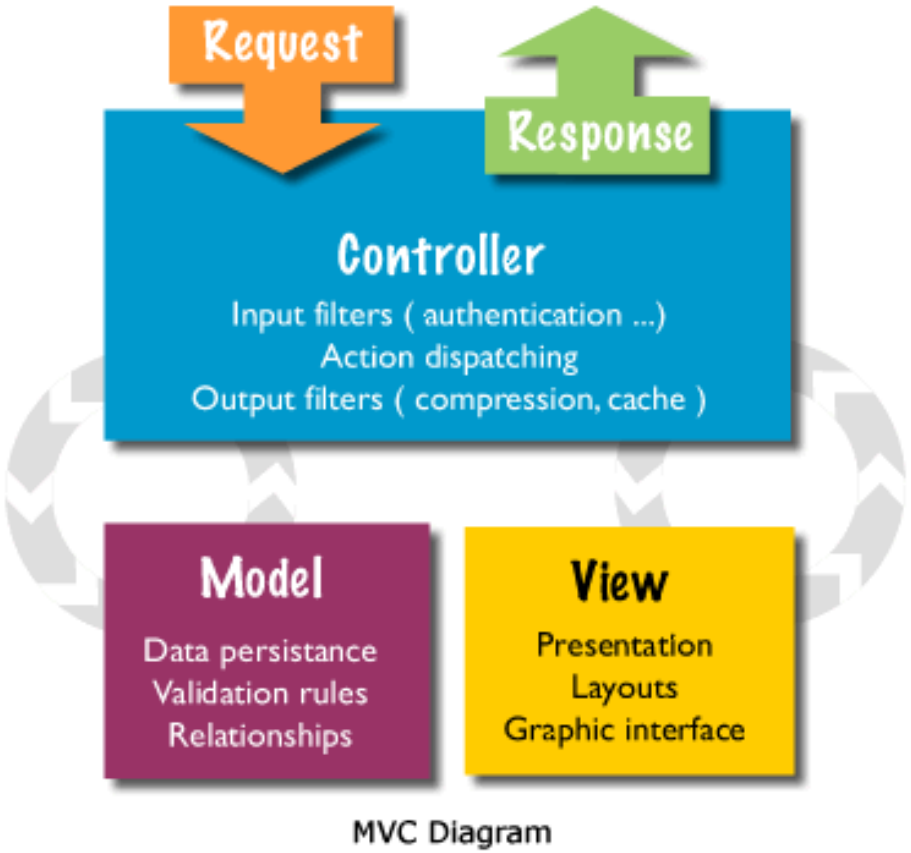
- Implementuje vlastnosti
  - Signál-Slot návrhový vzor
  - Objektové API
    - vizuálnych komponentov (tabuľky, záložky, …)
    - formulárových komponentov
    - formulárových validátorov
  - Podpora AJAX
  - Logovacie API

CENTAUR

tREST

Lepšie **raz vidieť**
ako **100-krát počuť**

tREST – kontróler



MVC Diagram

- **Kontróler** extenduje vždy spoločného predka (*TrestCntroller*), alebo jeho potomka (napríklad *WebController*)

```
public class PartsDepotController extends WebController {

}
```

CENTAUR

tREST – mapovanie kontrólera

- Kontroler mapujeme na **URL** od rootu aplikácie
- http://<server>/<app_root>/<controller_path>/

- Píklad
  - http://server.net/company/depot/

```
mount("/depot/", PartsDepotController.class);
```

tREST – mapovanie metód

- metóda je mapovaná na časť **URL**
  - http://<server>/<app_root>/<controller_path>/method
- Príklad:
  - http://server.net/company/depot/parts

tREST – mapovanie metód

- http://company.net/company/depot/parts
- http://company.net/company/depot/part/13

```
public class PartsDepotController extends WebController {

    public String parts() {
        Parts parts = new BL().getParts();

        String xmlOutput = Tools.toXML(parts);

        return xmlOutput;
    }

    public String part(int id) {
        Part part = new BL().getPart(id);

        String xmlOutput = Tools.toXML(part);

        return xmlOutput;
    }
```

CENTAUR

tREST – mapovanie vstupov

- **Vstup** z webu **len v textovej podobe**
  - **URL parametre**: <url>/param1/param2/.../paramN
    - Príklad:
      http://server.net/company/depot/parts/p1/p2/p3
  - **GET**: <url>?key1=value1&key2=value2
    - Príklad:
      http://server.net/company/depot/parts?k1=v1&k2=v2
  - **POST**: url encoded v HTTP body
    - key1=value1&key2=value2

CENTAUR

tREST – mapovanie vstupov

- URL: http://server.net/company/depot/part/13

```
public String part(int id)
```

- GET: http://server.net/company/depot/part?id=13
  - V Jave nie je možné pomocou reflexie zistiť meno parametra metódy, mená mapujeme pomocou anotácií

```
public String part(@Key("id") int id)
```

- POST: http://server.net/company/depot/part

tREST – iné možnosti mapovania

- Všetky URL parametre
  - napr. ako typ **UrlParameters**
- Všetky POST/GET parametre
  - napr. ako typ
    - **RequestParameters**
    - **PostRequestParameters**
    - **GetRequestParameters**

tREST – validácia vstupov

- Veľmi dôležitá – bezpečnosť
- potrebujeme validovať dva druhy vstupov
  - jednotlivé vstupy (webové služby)
  - celý formulár (webové aplikácie)
- dve fázy validácie
  - **syntaktická** – prevod textového reťazca na požadovaný typ
  - **sémantická** - napr. kontrola veľkosti čísla a pod.

CENTAUR

tREST – výstup (napr. XML)

- http://company.net/depot/parts
  - <?xml version="1.0"?>
    <p:Parts xmlns:p="http://company.net"
    xmlns:xlink="http://www.w3.org/1999/xlink">
    <Part id="00345" xlink:href="http://company.net/depot/part/12"/>
    <Part id="00346" xlink:href="http://company.net/depot/part/13"/>
    <Part id="00347" xlink:href="http://company.net/depot/part/14"/>
    </p:Parts>
- http://company.net/depot/part/13
  - <?xml version="1.0"?>
    <p:Part xmlns:p="http://company.net"
    xmlns:xlink="http://www.w3.org/1999/xlink">
    <Part-ID>13</Part-ID>
    <Name>Widget-A</Name>
    <Description>This part is used within the frap assembly</Description>
    <UnitCost currency="USD">0.10</UnitCost>
    <Quantity>10</Quantity>
    </p:Part>

CENTAUR

tREST – výstupy na web

- Výstup na webe môže byť len **text/stream**
- Štandardne programujeme metódy, ktoré môžu mať ako návratový typ ľubovoľný objekt
- Automatická **serializácia** výstupu (napr. JSON)

```
@Json(maxdepth = 5)
public Part part(int id)
{
    Part part = new BL().getPart(id);
    return part;
}
```

CENTAUR

tREST – výstupy na web

- Oddelenie logiky od prezentačnej vrstvy
  - Použitie šablón

```
public void parts() {
    Parts parts = new BL().getParts();
    view("templates/parts.jsp", "parts", parts);
}

@View(template = "templates/parts.jsp")
public Part part(int id)
{
    Part part = new BL().getPart(id);
    return part;
}
```

CENTAUR

Ďakujem

Otázky