Real-time Web

adam.horcica.cz


F4
F5
F6


admin / Switch b06
602b-esw
host: Blok 6/245 • Séric
AAAB3NzaC1yc2EAAAA
Překonfigurovat switch
Zobrazit stav switche b0602b-esw
Získání stavu switche bylo vytvořeno jako úkol na pozadí. Aktuální stav bude zobrazen po dokončení úkolu.
Zavřít
Procesy na pozadí
Získávání stavu switche b0602b-esw
Oblast/Místnost
Switch port
IP adresa
Poznámka
Blok 6/245
10.6.120.2

traffic.sh.cvut.cz

- Twitter
- Facebook
- GMail
- ...


http://w

HTTP

Hyper Text Transfare Protocol

- Protokol typu *Request – Response*
- **Komunikaci vždy začíná klient**
- Beze stavový protokol

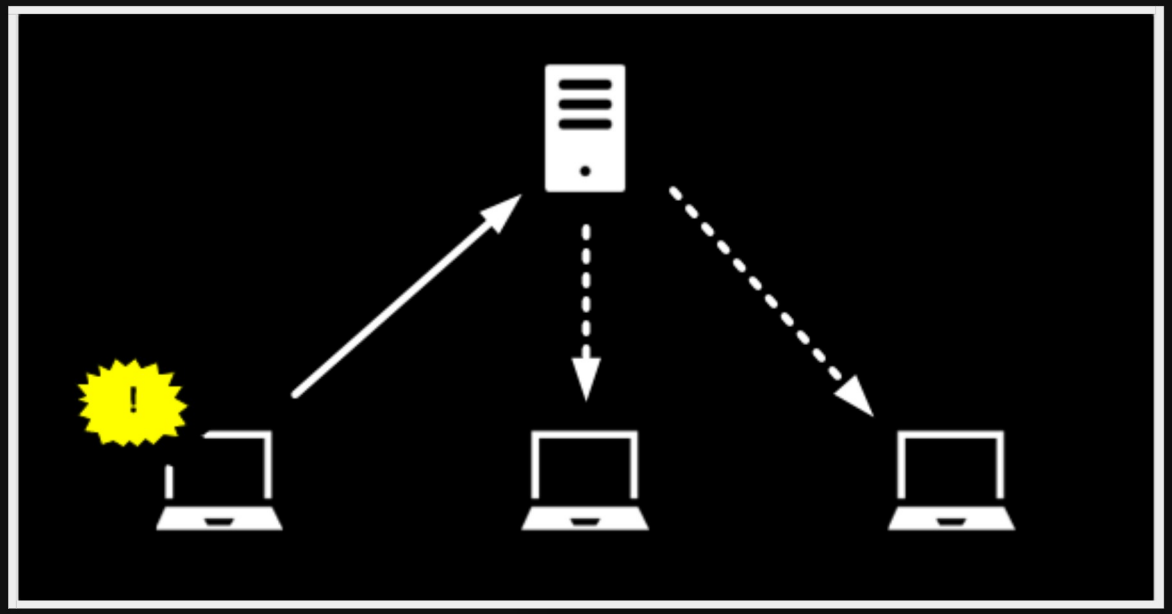
Jak to funguje...


```
GET /index.html HTTP/1.1
Host: www.example.com
```

<--- response ---

```
HTTP/1.1 200 OK
Date: Fri, 22 Mar 2013 17:59:12 GMT
Server: Apache/2.2.16 (Debian
Last-Modified: Fri, 01 Mar 2013 20:37:58 GMT
Content-Length: 123
Connection: close
Content-Type: text/html; charset=UTF-8

<html> … </html>
```



Technologie/techniky

1. Polling
2. Long Polling
3. Streaming
4. WebSockets

JavaScript

The World's Most Misunderstood Programming Language

“ JavaScript is not interpreted Java! Java is interpreted Java. JavaScript is a different language. ”

Douglas Crockford (http://www.crockford.com/javascript/javascript.html)

1 Polling

Periodické dotazování

„Školní“ Chat

Ukázková aplikace

```
<input type="text" id="message">
<button id="button">Send!</button>

<div id="history">
    <!-- messages -->
</div>
```

Odeslání zprávy

```
$("#button").click(function () {
    var msg = {
        'time': + new Date(),
        'text': $("#message").val()
    };
    $.post("/send", msg);
});
```

Příjem zprávy

```
/*
    Some Black Magic
    → messageReceived(…)
*/


function messageReceived(msg) {
    $.…
}
```

```
var last = + new Date();

function update() {
    $.get("/changes", {"from": last}, function (result) {
        if (result) {
            buttonPressed(result);
            last = result.serverTime;
        }
        setTimeout(update, 5000);
    });
}
update();
```

```
var express = require('express'), app = express();

var lastEvent = {time: 0, msg: null};

app.post('/send', function (req, res) {
    lastEvent = {serverTime: + new Date(),
                 text: req.body.text, time: req.body.time };
    res.send("ok");
});

app.get('/changes', function (req, res) {
    res.json(lastEvent.serverTime > req.query.from
             ? lastEvent
             : null);
});

app.listen(8081);
```

Ukázka

Výhody vs. nevýhody

+ Funguje v každém prohlížeči
− Není opravdové „real-time“

Další možnosti: XHR, JSONP

2 Long Polling

Periodické dotazování, ale...

```
function update() {
    $.get("/changes", {rnd: Math.random()}, function (result) {
        if (result) {
            buttonPressed(result);
        }
        setTimeout(update, 10);
    }).fail(function () {
        setTimeout(update, 100);
    })
}

update();
```

```
var express = require('express'), app = express();

var button = new (require('events').EventEmitter)();

app.post('/send', function (req, res) {
    var msg = {
        time: req.body.time,
        text: req.body.text
    };
    button.emit("click", msg);
    res.send("ok");
});

app.get('/changes', function (req, res)
{
    button.once("click", function(msg) {
        if (res.connection.writable) {
            res.json(msg);
        }
    });
});

app.listen(8082);
```

Ukázka

Výhody vs. nevýhody

+ Malá latence
− HTTP doporučuje 2 spojení na doménu
− Každý ~~pes~~ prohlížeč jiná ves

Další možnosti: XHR, JSONP, Comet

3 Streaming

Nešlo by to jen s jedním spojením?

Hádanka

Co to je?

Ukázka

```
$(document).load(function () {
    $("<iframe>")
        .attr("src", "/changes")
        .css("display", "none")
        .appendTo("body");
});
```

```
app.get('/changes', function (req, res)
{
    res.writeHead(200, {
        'Content-Type': 'text/html',
    });

    res.write("<!DOCTYPE html>\n");
    res.write("<html><head>\n");
    res.write("<title>Hidden Frame</title>\n");
    res.write("</head><body>\n");

    button.on("click", function(msg) {
        if (res.connection.writable) {
            var js = "parent.buttonPressed("+JSON.stringify(msg)+")";
            res.write("<script>" + js + "</script><br>\n");
            res.write((new Array(1000).join(" ")) + "\n");
        }
    });

});
```

Ukázka

Výhody vs. nevýhody

+ Jedno trvalé spojení
– Problém s vyrovnávací pamětí prohlížeče
– HTTP doporučuje 2 spojení na doménu
– Každý ~~pes~~ prohlížeč jiná ves

Další možnosti: Flash XMLRequest, XHR Streaming, BOSH, **Server-Sent Events**

4 Web Socket

Nešlo by to obousměrně?


```
GET /chat HTTP/1.1
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: x3...JH==
Sec-WebSocket-Protocol: chat
Sec-WebSocket-Version:  13
```


```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: HSm…GWk=
Sec-WebSocket-Protocol: chat
```

```
var ws = new WebSocket("ws://" + location.host);

$("#button").click(function () {
    var data = {
        'time': + new Date(),
        'text': $("#message").val()
    };
    ws.send(JSON.stringify(data));
});

ws.onmessage = function (event) {
    var msg = JSON.parse(event.data);
    buttonPressed(msg);
}
```

```
var express = require('express'), app = express(),
    ws = require('websocket.io'),
    http = app.listen(8084),
    server = ws.attach(http);

var button = new (require('events').EventEmitter)();

server.on('connection', function (socket) {

    var handler = function (msg) {
        socket.send(msg);
    };

    button.on("click", handler);

    socket.on("message", function (msg) {
        button.emit("click", msg);
    });

    socket.on("close", function () {
        button.removeListener("click", handler);
    });

});
```

Ukázka

Výhody vs. nevýhody

+ Technologie navržená pro tyto účely
− Pouze v moderních prohlížečích

Závěr

1. Pokud můžete, použijte WebSocket
2. Pokud nemůžete, použijte specializovanou knihovnu
3. Nebo použijte SaaS

Knihovny

- Automatický výběr nejlepší komunikační metody
- Skupiny, broadcast, priorita, potvrzování
- Zabezpečení, session
- Fronty (redis)

Např. Socket.IO, Faye, SignaR, …

Socket.IO

```
<script src="/socket.io/socket.io.js"></script>
```

```
var socket = io.connect('http://' + location.host);

$("#button").click(function () {
    var data = {
        'time': + new Date(),
        'text': $("#message").val()
    };
    socket.emit('msg', data);
});

socket.on("show", buttonPressed);
```

```
var express = require('express'), app = express(),
    http = app.listen(8085),
    io = require("socket.io").listen(http);

io.sockets.on('connection', function (socket) {
    socket.on('msg', function (msg) {
        io.sockets.emit("show", msg);
    });
});
```

Ukázka

traffic.sh.cvut.cz

SaaS

- Pokud nemáme vlastní server
- Nebo používáme technologii, která neumožňuje dlouho otevřené spojení (např. PHP)

Např. **pusher.com**, Google Channel API, ...

Shrnutí

?