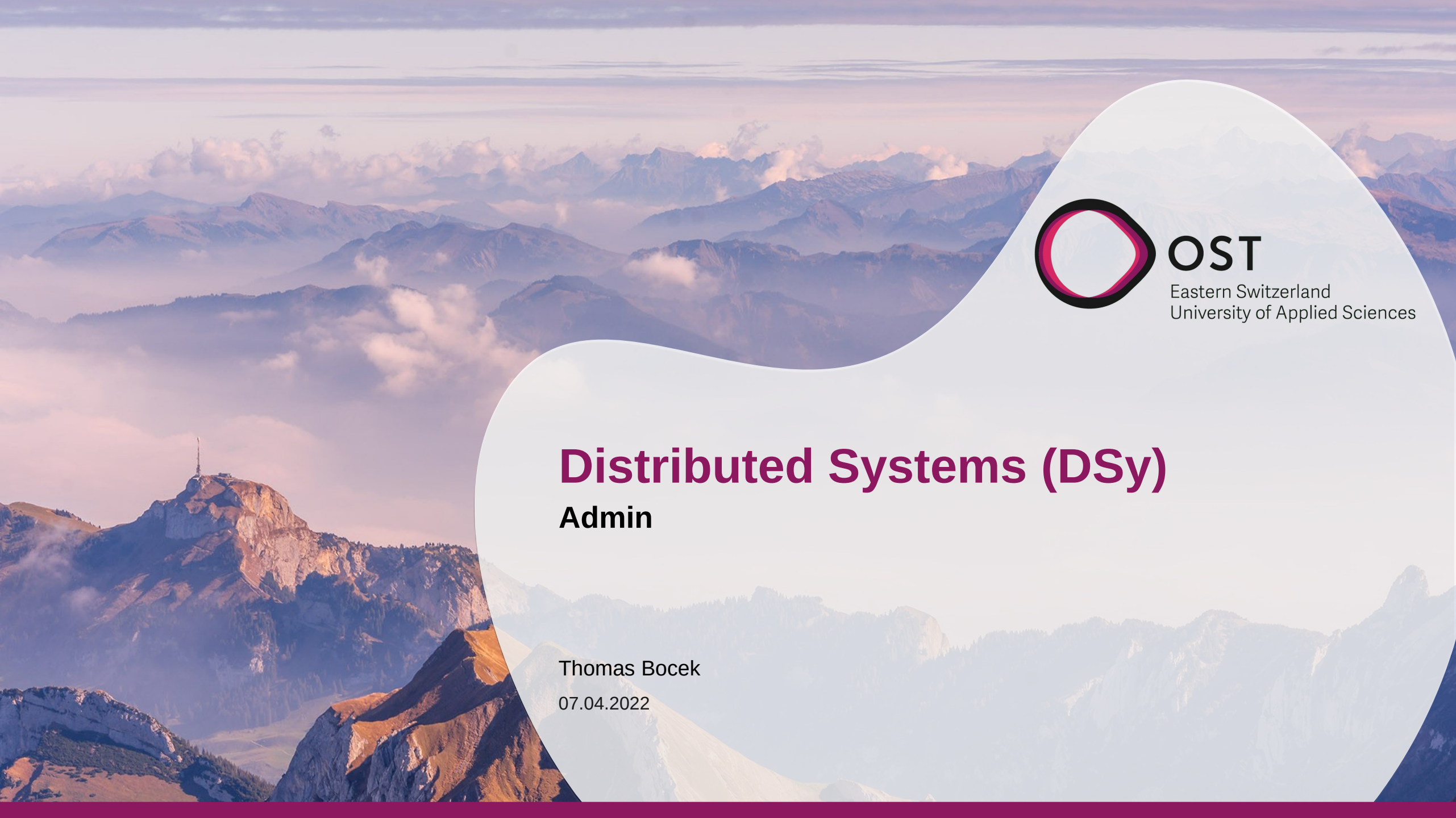


OST
Eastern Switzerland
University of Applied Sciences

Distributed Systems (DSy)

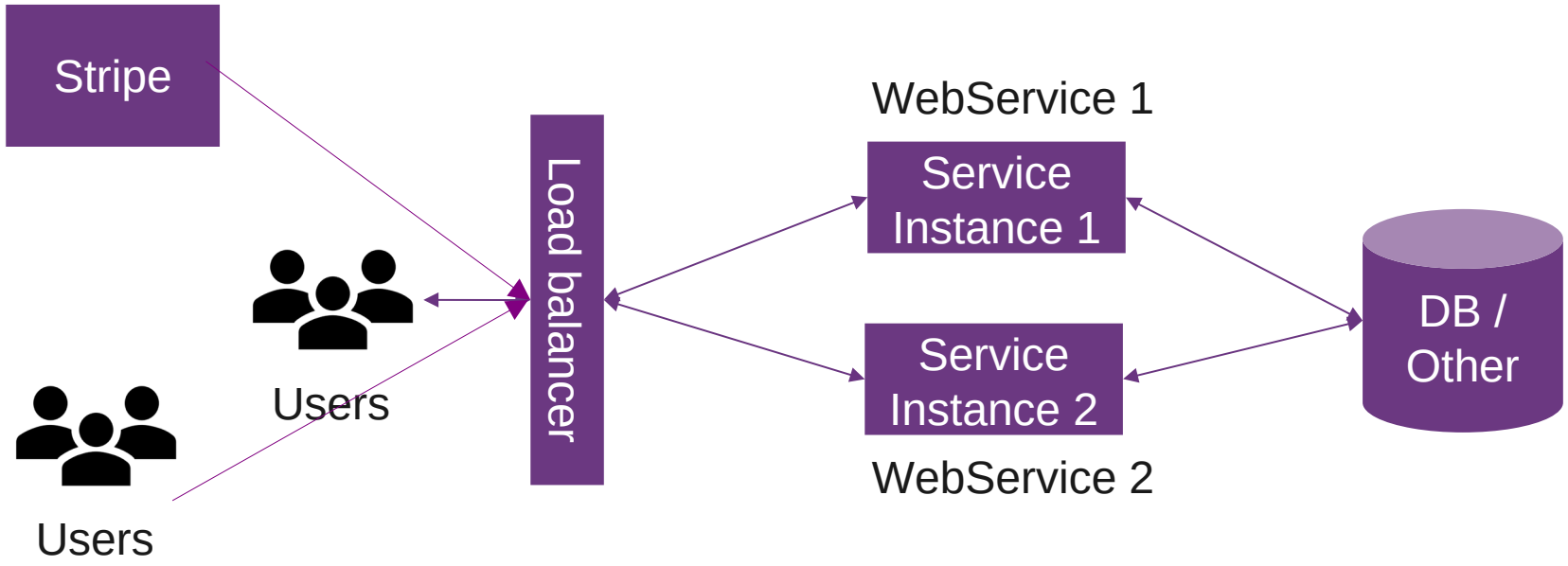
Admin

Thomas Bocek

07.04.2022

Services

- WebSockets – arch decisions
  - Scenario: callback, e.g. stripe – to which service?
  - Scenario: other user – to which service?
    - Solutions: a) store connection to user in the DB, a1) use distrtibuted memory libraries – memcached, redis, b) same user to same service (group users), c) Message Queue → send connection infos to all services (all services have all connection infos), d) use only 1 service (flatfeestack)



XOVIS Sensor Integration

- Real-time counter
  - Polling - latency
  - WebSocket! (well-supported)
- WebSocket
  - Full duplex communication with TCP
  - Text or binary
  - Ports 80/443, compatible with HTTP, uses upgrade header to change from HTTP to WebSocket
  - HTTP request/reply
    - WebSocket does not need request
  - wss://localhost…

```
let connection = new WebSocket('wss://localhost:8080/api');
connection.binaryType = "arraybuffer";
```

```
GET ws://localhost:8080/ws HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: tvdBI26pvLeIgFiDC3H5Sg==
Sec-WebSocket-Version: 13
Origin: http://localhost:8080
```

```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: Myz76wyg5WStKRkpSCKhTt/tWAM=
```

WebSockets

- Reverse Proxy?
  - Nginx / Apache
  - Proxy need to set header again, otherwise lost
  - Connection timeout
- Go implementation
  - Basic: x/net
  - Advanced: github.com/gorilla/websocket
- Example:

```
location /chat/ {
    proxy_pass http://backend;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
    proxy_read_timeout 60s;
}
```

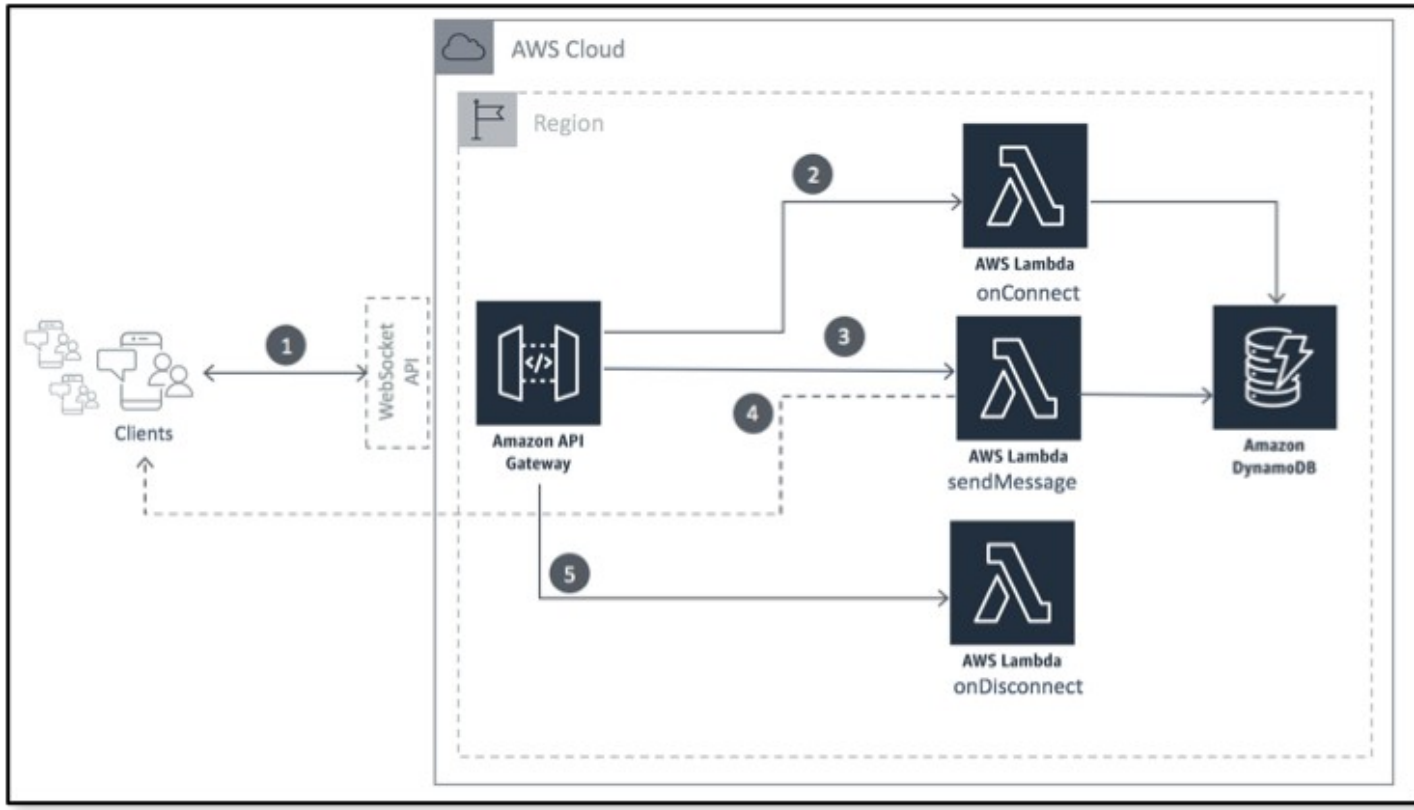
```
mod_proxy_wstunnel:

ProxyPass "/ws2/"  "ws://echo.websocket.org/"
ProxyPass "/wss2/" "wss://echo.websocket.org/“
ProxyWebsocketIdleTimeout 60
```

```
var connection = new WebSocket('ws://localhost:8080/ws');
    connection.onopen = function () {
        connection.send('Ping');
    };
    let counter = 0;
    connection.onmessage = function (e) {
        console.log('update: ' + counter++);
        let string = 'Markets:<br>';
        for(let s of JSON.parse(e.data)) {
            string += s.name + ', price=' + s.price_usd + '<br>';
        }
        let element = document.getElementById('text');
        element.innerHTML = string;
    };
```

```
func ws(w http.ResponseWriter, r *http.Request, _
httprouter.Params) {
    conn, _ := upgrader.Upgrade(w, r, nil)
    messageType, _, _ := conn.NextReader()
    for {
        res, _ :=
http.Get("https://api.coinmarketcap.com/v1/ticker/?limit=10")
        w, _ := conn.NextWriter(messageType)
        io.Copy(w, res.Body)
        time.Sleep(2 * time.Second)
    }
}
```

AWS Websockets

- 18.12.2018: Announcing WebSocket APIs in Amazon API Gateway
- Sample App on github
  - ~200 lines configuration
  - ~70 line of code



Services

- Caddy health checks
  - In the lecture: passive health checks
  - Nignx vs Caddy: nginx resends request if upstream fails, caddy does not
- Example ThorwalletDEX – we use caddy as a LB
  - Yesterday, one upstream went down (out of disk)
  - Degraded performance, since every 60s, the offline upstream is tried again, results in timout
- Solution
  - Switch to Nginx
  - Use active health check as suggested also in the exercises
  - Example

```
reverse_proxy {
        to localhost:8087 localhost:8088
        health_uri /status
        health_interval 5s
        health_timeout 500ms
    }
```


OST

DSy 2022: Plan

Nr	Date	Lecture*	Lecturer
01	25.02.2022	Admin, Introduction/Motivation	Bocek
02	04.03.2022	Introduction/Categorization	Bocek
03	11.03.2022	Distributed Applications: Tor, WebSockets, Protocols 1	Bocek
04	18.03.2022	Mulirepo / Monorepo, Protocols 2	Bocek
05	25.03.2022	Containers and VMs	Bocek
06	01.04.2022	Load Balancing	Bocek
07	08.04.2022	Application Protocols and RPC	Bocek
-	15.04.2022	- No Lecture (Spring Vacation) – **Hand-in 1 (14.04.2022, 23:59 (CET))**	-
08	22.04.2022	Authorization (Basic, JWT, Oauth, …), Deployment (Docker Swarm, Kubernetes),	Bocek
09	29.04.2022	Blockchain, Bitcoin, Ethereum Introduction	Bocek
10	06.05.2022	Smart Contracts	tba
11	13.05.2022	Challenge Task Presentations I	you
12	20.05.2022	Challenge Task Presentations II	you
-	27.05.2022	- No Lecture	-
13	03.06.2022	Guest lecture	tba
14	10.06.2022	Challenge Task Award Winner Announcement, Q&A	Bocek

* topics may change