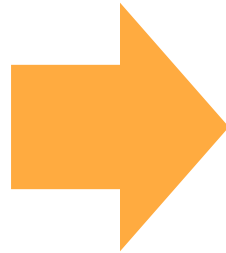


Realtime Web mit RethinkDB und Websockets

Chemnitzer StudentenNetz
Tobias Gall

Ziel

Switch-Logs in Echtzeit auf der Webseite



Wie?

RethinkDB

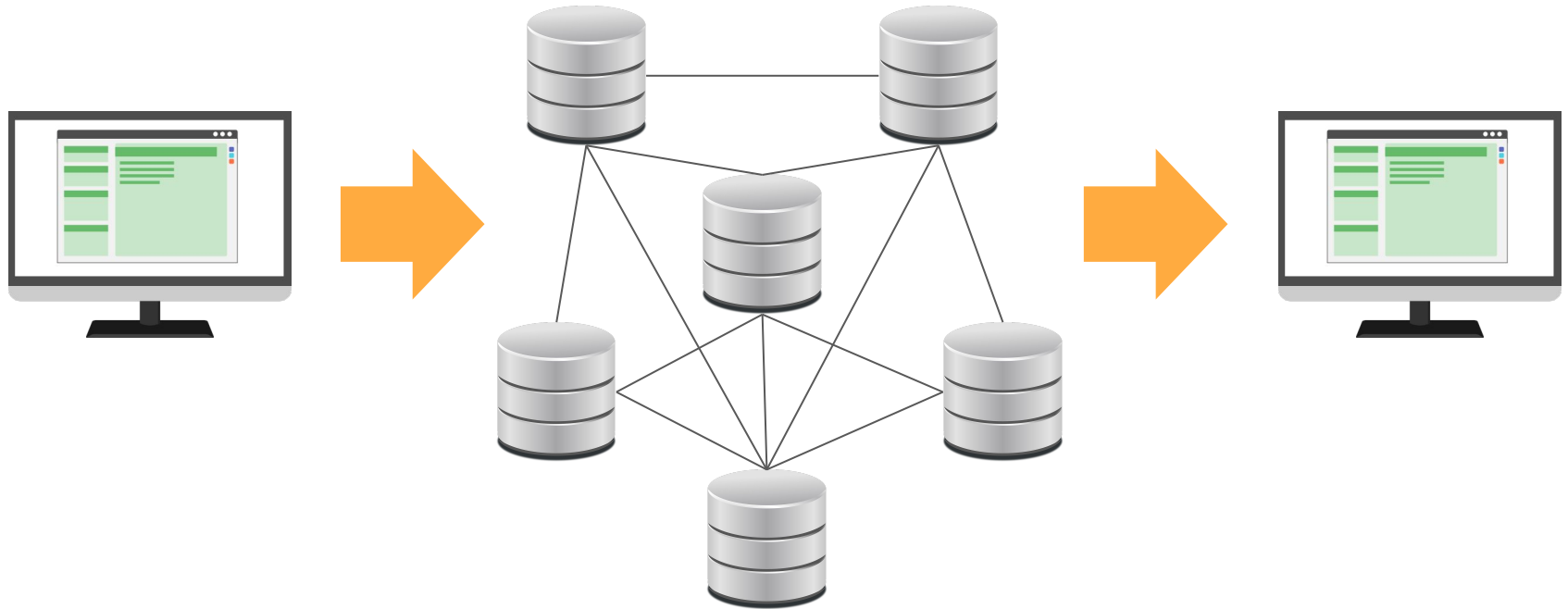
Websockets

RethinkDB

Was ist RethinkDB?

- JSON Datenbank
- Geschrieben in C++
- Konzipiert für Echtzeit-Anwendungen
- Eigene mächtige Query-Language: ReQL

Was ist RethinkDB?



Anwendungsgebiete

- Kollaborative Web- und mobile Anwendungen
- Analytische Anwendungen
- Spiele

ReQL Crashkurs

(python)

Connection

```
import rethinkdb as r
```

```
r.connect("localhost", 28015).repl()
```

```
# r.connect(host, port, db, user, password, timeout, ssl)
```

Datenbank Anlegen

```
r.db_create('superheroes').run()
```

```
{ "config_changes": [{  
  "new_val": {  
    "id": "e4689cfc-e903-4532-a0e6-2d6797a43f07",  
    "name": "superheroes"},  
  "old_val": None}],  
  "dbs_created": 1  
}
```

Tabelle Anlegen

```
r.db('superheroes').table_create('marvel').run()
```

```
{  
  "config_changes": [  
    # <table configuration data>  
  ]  
  "tables_created": 1  
}
```

Daten Einfügen

```
heroes = [  
    { "name": "Captain America", "real_name": "Steve Rogers"  
    },  
    { "name": "Daredevil", "real_name": "Matthew Murdock" },  
]  
r.table('marvel').insert(heroes).run()
```

Daten Einfügen

```
{  
  "unchanged": 0, "skipped": 0, "replaced": 0,  
  "inserted": 2,  
  "generated_keys": [  
    "7644aaf2-9928-4231-aa68-4e65e31bf219",  
    "064058b6-cea9-4117-b92d-c911027a725a"  
  ],  
  "errors": 0, "deleted": 0  
}
```

Daten Updaten

```
r.table('marvel').update({"type": "fictional"}).run()
```

```
r.table("marvel")  
  .filter(r.row['real_name'] == "Peter Parker")  
  .update({'city': 'New York'}).run()
```

Daten Abholen

```
# All Data
```

```
for hero in r.table('marvel').run():  
    print(hero)
```

```
# By Primary Key
```

```
daredevil =  
r.table('marvel').get('064058b6-cea9-4117-b92d-c911027a725a'  
)  
.run()
```

Daten Filtern

```
heroes = r.table('marvel')  
        .filter(r.row["name"] == "Daredevil").run()  
  
for hero in heroes:  
    print(hero)
```


Daten Löschen

```
r.table('marvel')  
  .filter(r.row["city"] == "New York")  
  .delete().run()
```

Daten Feeds

```
heroes = r.table('marvel').changes().run()
```

```
for hero in heroes:
```

```
    print(hero)
```

```
{
```

```
    "new_val": { # <new data> },
```

```
    "old_val": { # <old data if updated else None> }
```

```
}
```

Wie?

RethinkDB

Websockets

Websockets

Was sind Websockets?

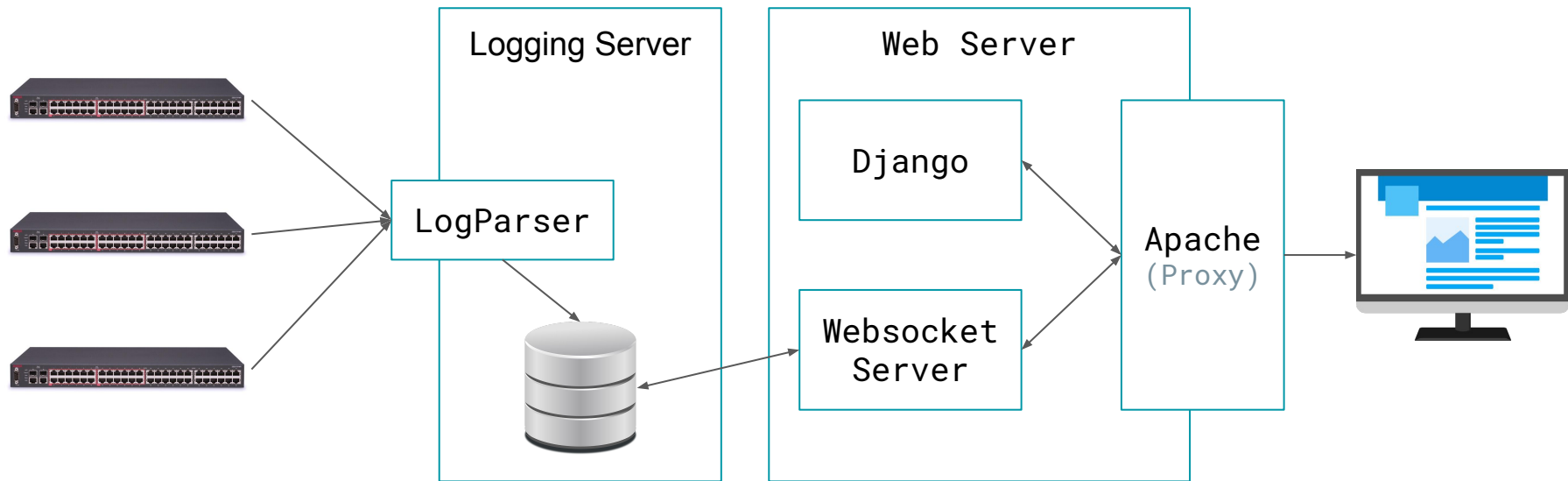
- Bidirektionale Verbindung zwischen einem Server und dem Browser
- Verbindungsaufbau über HTTP
- Kommunikation über TCP

Javascript

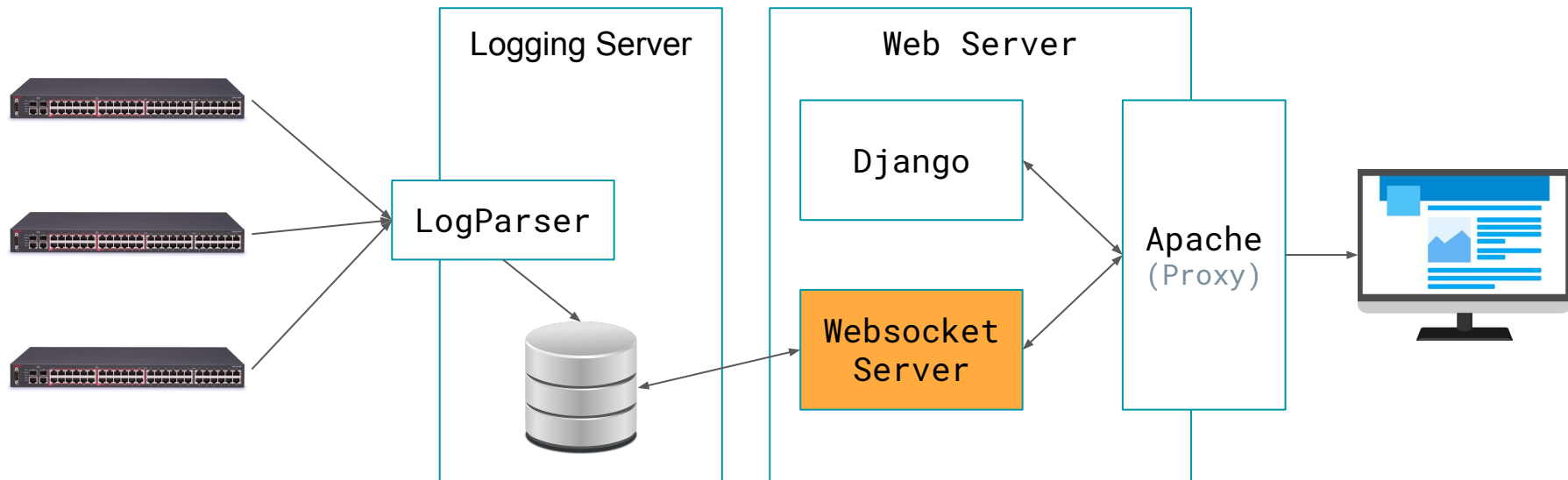
```
const connection = new WebSocket('wss://localhost:9000/');
connection.onopen = function () {
    connection.send('Ping'); // Send the message 'Ping' to the server
};
connection.onerror = function (error) {
    console.error('WebSocket Error ' + error);
};
connection.onmessage = function (e) {
    console.log('WebSocket Message: ' + e.data);
};
```

Umsetzung im Chemnitzer StudentenNetz

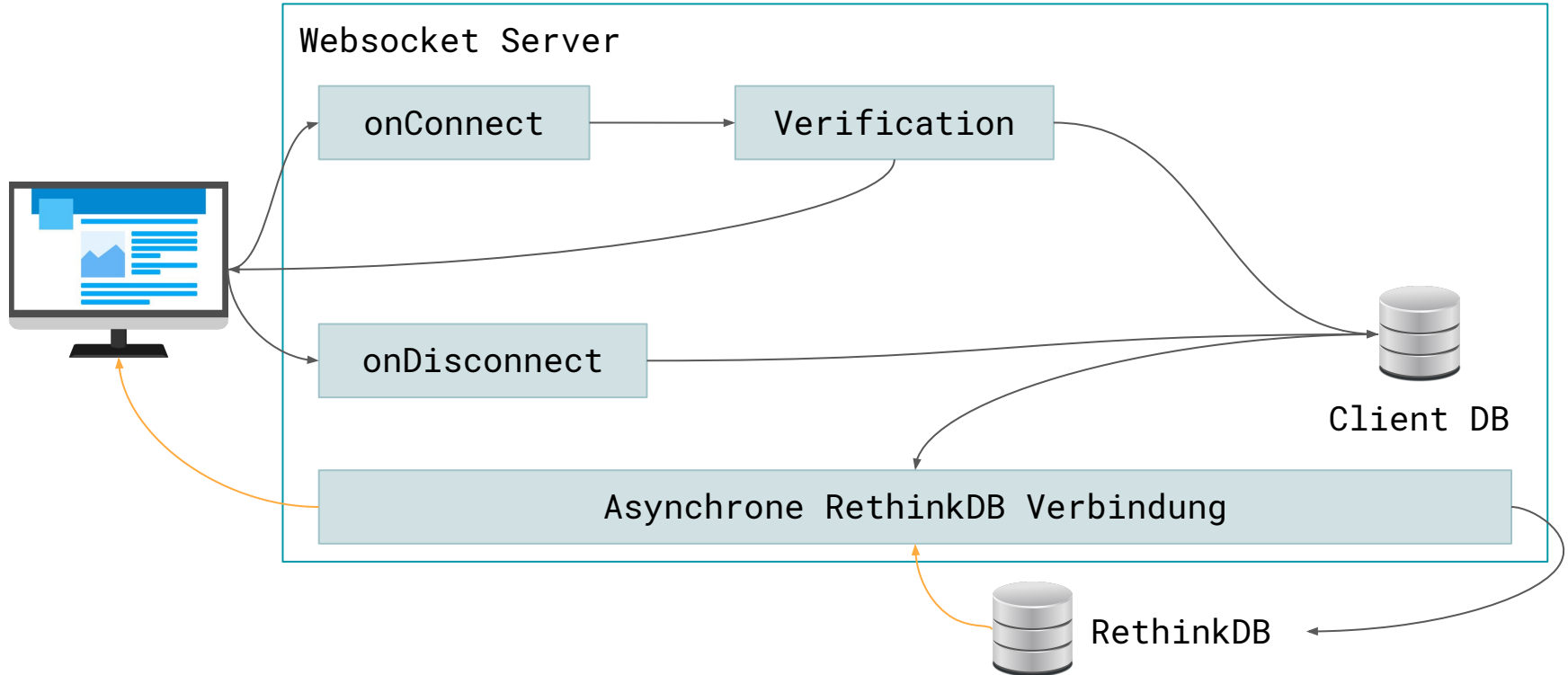
Setup



Setup



Websocket Server



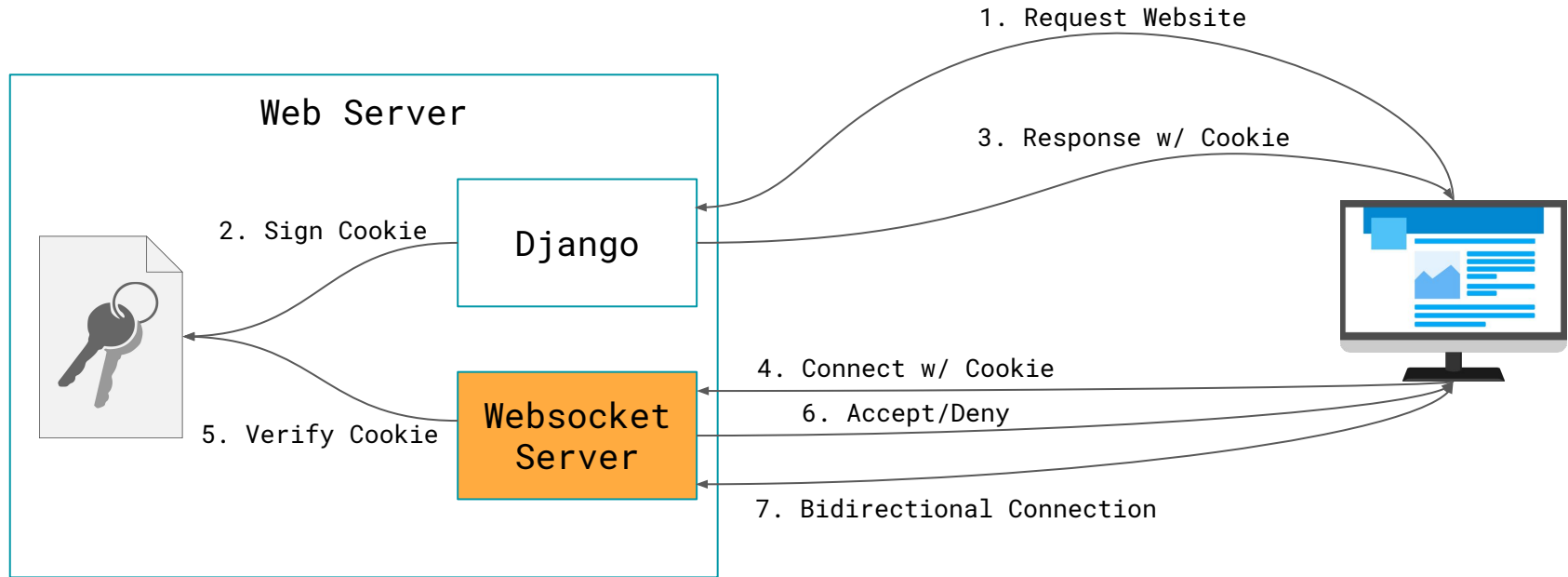
Demo

Sicherheit?!

Websockets sind Unsicher!

- Spoofing der 'Origin'-Policy
- Keine eingebaute Authentifizierung/Autorisierung
- Daten über TCP (kein definiertes Datenformat)

Autorisierung via Cookie



Happy Coding!