

Exercise 3.d

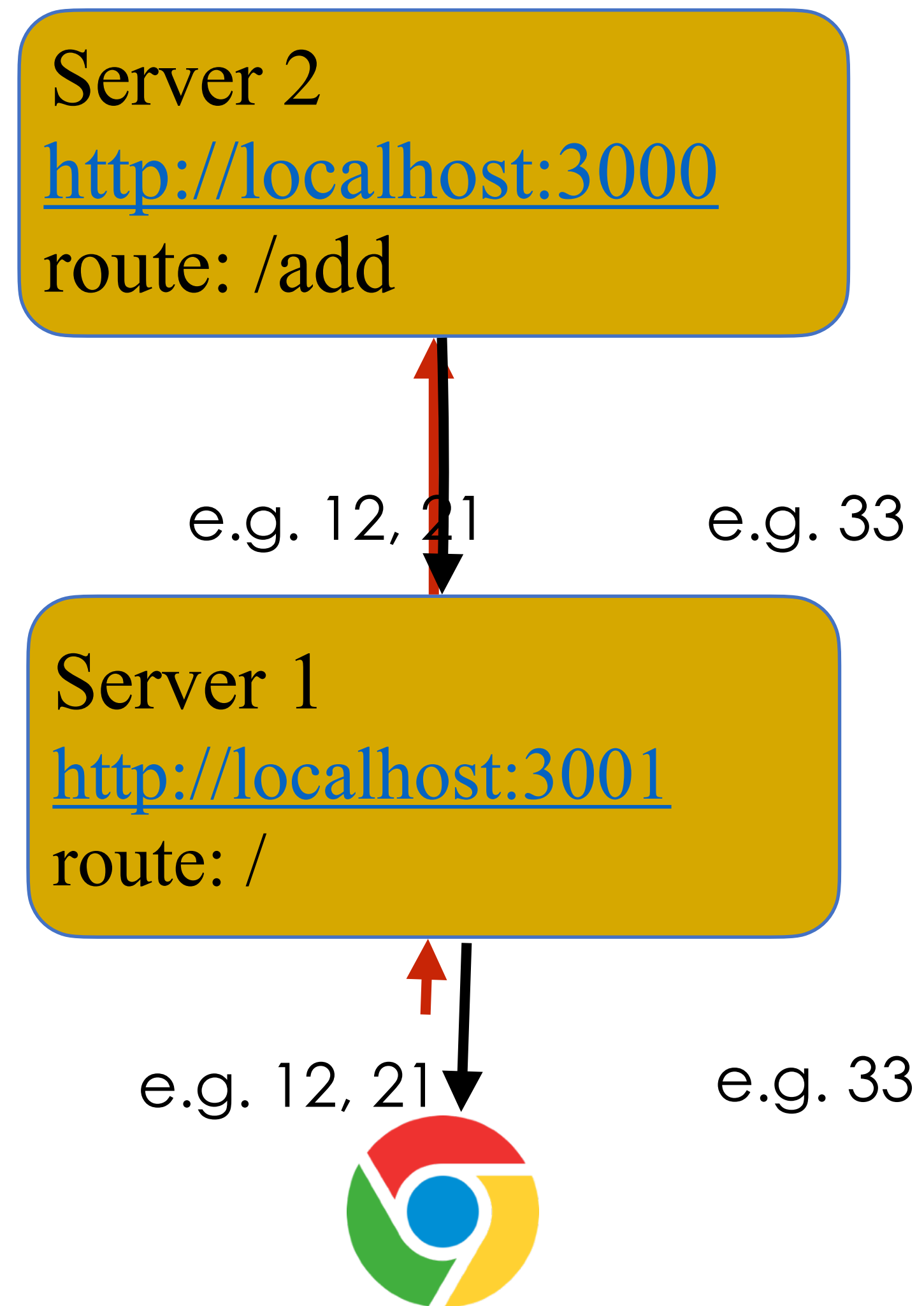
Axios on both Client and Server

Prof. Fabio Ciravegna
Department of Computer Science
The University of Sheffield
f.ciravegna@sheffield.ac.uk

Using Axios

- This is the same exercise we saw two days ago
- Have a client receives two numbers in input and sends them to server1
- server1 receives the two numbers and ships them to server 2
- server 2 adds the numbers and returns them to server 1
- server 1 return the numbers to the client
- We used fetch. This time instead will use
 - Axios on the client and on the server

Example



Replacing JQuery

- add the dependency on Axios in the ejs file

```
<script src="https://cdn.jsdelivr.net/npm/axios/dist/axios.min.js"></script>
```
- and insert the axios call (see lecture slides) in javascripts/index.js
- Solution in the next slide but try to work it out yourself

Solution for the client

```
function sendAxiosQuery(url, data) {  
  axios.post(url, data)  
    .then(function (dataR) {  
      document.getElementById('results').innerHTML= "The result is: "+JSON.stringify(dataR.data);  
    })  
    .catch(function (response) {  
      alert (response.toJSON());  
    })  
}
```

•

Replacing Fetch on teh server

- Axios is simpler to use than Fetch

- remove this part

```
fetch('http://localhost:3001/add', {  
  method: 'post',  
  body: JSON.stringify({firstNumber: firstNo, secondNumber: secondNo}),  
  headers: {'Content-Type': 'application/json'},  
})
```

- with the corresponding Axios call
- Solution in the next slide but try to work it out yourself using the lecture slides

Solution: the final route

```
router.route('/')  
  .get(function(req, res) {  
    res.render('index', {title: 'Express'});  
  })
```

```
  .post(function (req, res) {  
    let firstNo = req.body.no1;  
    let secondNo = req.body.no2;  
    if (isNaN(firstNo) || isNaN(secondNo)) {  
      res.setHeader('Content-Type', 'application/json');  
      res.status(503).json({error: 403, reason: 'One of the numbers is invalid'});  
    } else {  
      axios.post('http://localhost:3000/add',  
        {firstNumber: firstNo, secondNumber: secondNo})  
        .then(json => {  
          res.json(json.data.result)  
        })  
        .catch(err => {  
          res.setHeader('Content-Type', 'application/json');  
          res.status(403).json(err)  
        })  
    }  
  })  
});
```

You should be able to do
this exercise in 15 minutes