THE UNIVERSITY *of* ADELAIDE

Web and Database Computing

adelaide.edu.au

Client-Server Basics: Serving Content Dynamically

Serving Dynamic Content

Dynamic view counter

Say we want to create a counter that shows how many times a button has been pushed.

- Write a basic HTML page
    - Include a counter and a button that increments the counter.
- What will happen if we now serve this from express as a static page?
- What if we wanted the counter to represent the number of times the page was loaded rather than the number of times the user pressed the button?
- What if we wanted the counter to represent the number of times anyone in the world pressed the button on the page?

State information

The information needs to be remembered somewhere!

Options:

- Store the value of the variable on the web server in our Javascript
- Save the value in a file on the server and read/write as needed
- Store the value in a database and query/update as needed

However the data is stored, it needs to be inserted into the web page before we send it to the client.

Our express files

- `routes` contains Node.js code files for serving dynamic content
- `routes/index.js` contains basic dynamic routes

Routes

Routes are special functions that we can define on our server to perform actions when a given path is requested.

```
router.get('/some/path', function(req, res) {
    // Do stuff
});
```

http://expressjs.com/en/api.html#router.METHOD

Our first route

Demo

- Modify the `routes/index.js` file to add a new custom `GET` route at `/count` that prints the number of page visits

Sending a response

We don’t want to just print the output, we want to send an HTTP response to the client!

- The `res` object that express passes to our function helps us here.
    - `router.get('/test', function(req,res) {`
    - `res` represents the HTTP response that express sends.
- One of the methods of the response object res, is the send method.
    - `res.send([body])`
    - The body is the HTTP response body. Can be a string (html web page), or data.

```
router.get('/test', function(req, res) {
    res.send("This is a test");
});
```

Let's use `res.send()` to send our message to the browser instead of the console.

http://expressjs.com/en/api.html#res

A route that sends a response

Demo

- Modify the `routes/index.js` file's route at `/count` to send the number of page visits as a response.

Dynamic webpages

Welcome to the dot-com era

Dynamic webpages

Dynamic webpages were the cornerstone of the modern internet, allowing us to make complex websites.

1. A user makes a request
2. The request is processed
3. Custom content is generated
4. The content is combined with HTML to create the webpages
5. The **whole** page is sent in response

Dynamic webpage example

Demo

- Modify the `routes/index.js` file's route at `/count` to send a full html page as a response.

SUB CRUCE LUMEN

THE UNIVERSITY of ADELAIDE